



Technical Document - Confidential

GSM PROTOCOL STACK

TEST SPECIFICATION

DL

Document Number:	6147.405.97.101
Version:	0.3
Status:	Draft
Approval Authority:	
Creation Date:	1997-Mar-21
Last changed:	2015-Mar-08 by XGUTTEFE
File Name:	dl.doc

Important Notice

Texas Instruments Incorporated and/or its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products, software and services at any time and to discontinue any product, software or service without notice. Customers should obtain the latest relevant information during product design and before placing orders and should verify that such information is current and complete.

All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment. TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI products, software and/or services. To minimize the risks associated with customer products and applications, customers should provide adequate design, testing and operating safeguards.

Any access to and/or use of TI software described in this document is subject to Customers entering into formal license agreements and payment of associated license fees. TI software may solely be used and/or copied subject to and strictly in accordance with all the terms of such license agreements.

Customer acknowledges and agrees that TI products and/or software may be based on or implement industry recognized standards and that certain third parties may claim intellectual property rights therein. The supply of products and/or the licensing of software does not convey a license from TI to any third party intellectual property rights and TI expressly disclaims liability for infringement of third party intellectual property rights.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products, software or services are used.

Information published by TI regarding third-party products, software or services does not constitute a license from TI to use such products, software or services or a warranty, endorsement thereof or statement regarding their availability. Use of such information, products, software or services may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

No part of this document may be reproduced or transmitted in any form or by any means, electronically or mechanically, including photocopying and recording, for any purpose without the express written permission of TI.

Change History

Date	Changed by	Approved by	Version	Status	Notes
1997-Mar-21	Stefan Lemke et al		0.1		1
1997-Sep-24	Volker Klann		0.2		2
2003-May-13	XGUTTEFE		0.3	Draft	

Notes:

1. Initial version
2. Updates: DL004

Table of Contents

1.1	References	8
1.2	Abbreviations	11
1.3	Terms	13
2	Overview	14
3	Parameters	15
4	TEST CASES	133
4.1	Routing (internal)	133
4.1.1	DL000: Setup the routing and PCO view for the DL test	133
4.2	Unacknowledged Information Transfer	136
4.2.1	DL002: Transmit empty frame (SDCCH)	136
4.2.2	DL003: Transmit empty frame (SACCH)	137
4.3	Contention Resolution from Idle mode (SDCCH)	138
4.3.1	DL011: Initiation by Radio Resource	138
4.3.2	DL012: UA response received from Base Station (UA = SABM)	139
4.3.3	DL013: UA response received from Base Station (UA not = SABM)	140
4.3.4	DL014: DM response received from Base Station - establishment rejected	141
4.3.5	DL015: Expiry of T200 timer - max. number of retransmissions not reached	142
4.3.6	DL016: Expiry of T200 timer - maximum number of retransmissions reached	143
4.3.7	DL017: DL012 – Engineering mode	145
4.4	Contention Resolution from Idle mode (FACCH Fullrate, SACCH)	146
4.4.1	DL021: Initiation by Radio Resource	146
4.4.2	DL022: UA response received from Base Station (UA = SABM)	147
4.4.3	DL023: UA response received from Base Station (UA not = SABM)	148
4.4.4	DL024: DM response received from Base Station - establishment rejected	149
4.4.5	DL025: Expiry of T200 timer - max. number of retransmissions not reached	150
4.4.6	DL026: Expiry of T200 timer - maximum number of retransmissions reached	152
4.4.7	DL027: Initiation and establishment of multiple frame state (SACCH, SAPI 3)	158
4.5	Contention Resolution from Idle mode (FACCH Halfrate, SACCH)	159
4.5.1	DL421: Initiation by Radio Resource	159
4.5.2	DL422: UA response received from Base Station (UA = SABM)	160
4.5.3	DL423: UA response received from Base Station (UA not = SABM)	161
4.5.4	DL424: DM response received from Base Station - establishment rejected	162
4.5.5	DL425: Expiry of T200 timer - max. number of retransmissions not reached	163
4.5.6	DL426: Expiry of T200 timer - maximum number of retransmissions reached	164
4.5.7	DL427: Initiation and establishment of multiple frame state (SACCH, SAPI 3)	170
4.6	Error Conditions	171
4.6.1	DL031: Error frame received (SDCCH)	171
4.6.2	DL032: Error frame received (FACCH Fullrate)	172
4.6.3	DL033: Error frame received (FACCH Halfrate)	173
4.7	Suspension and Resumption of Multiframe Operation (SAPI 0, SDCCH)	174
4.7.1	DL041: Suspend multiframe operation	174
4.7.2	DL042: Initiate resumption of multiframe operation	175
4.7.3	DL043: Expiry of T200 timer - max. no. of retransmissions not reached	176
4.7.4	DL044: Expiry of T200 timer - max. no. of retransmissions reached	177
4.7.5	DL045: UA response from Base Station - information field empty	179
4.7.6	DL046: UA response from Base Station - information field set	180
4.7.7	DL047: DM response from Base Station - establishment rejected	181
4.7.8	DL051: Initiate reconnection of multiframe operation	182

4.7.9	DL052: Expiry of T200 timer - max. no. of retransmissions not reached	183
4.7.10	DL053: Expiry of T200 timer - max. no. of retransmissions reached	184
4.7.11	DL054: UA response from Base Station - information field empty	186
4.7.12	DL055: UA response from Base Station - information field set.....	187
4.7.13	DL056: DM response from Base Station - establishment rejected.....	188
4.8	Resumption of Multiframe Operation (SAPI 0, FACCH Fullrate).....	189
4.8.1	DL061: Initiate resumption of multiframe operation	189
4.8.2	DL062: Expiry of T200 timer - max. no. of retransmissions not reached	190
4.8.3	DL063: Expiry of T200 timer - max. no. of retransmissions reached	192
4.8.4	DL064: UA response from Base Station - information field empty	198
4.8.5	DL264: UA response from Base Station - information field empty (II).....	199
4.8.6	DL065: UA response from Base Station - information field set.....	201
4.8.7	DL066: DM response from Base Station - establishment rejected	202
4.8.8	DL071: Initiate reconnection of multiframe operation	203
4.8.9	DL072: Expiry of T200 timer - max. no. of retransmissions not reached	204
4.8.10	DL073: Expiry of T200 timer - max. no. of retransmissions reached	206
4.8.11	DL074: UA response from Base Station - information field empty	212
4.8.12	DL075: UA response from Base Station - information field set.....	213
4.8.13	DL076: DM response from Base Station - establishment rejected.....	214
4.9	Resumption of Multiframe Operation (SAPI 0, FACCH Halfrate)	215
4.9.1	DL461: Initiate resumption of multiframe operation	215
4.9.2	DL462: Expiry of T200 timer - max. no. of retransmissions not reached	216
4.9.3	DL463: Expiry of T200 timer - max. no. of retransmissions reached	218
4.9.4	DL464: UA response from Base Station - information field empty	224
4.9.5	DL465: UA response from Base Station - information field set.....	225
4.9.6	DL466: DM response from Base Station - establishment rejected	226
4.9.7	DL471: Initiate reconnection of multiframe operation	227
4.9.8	DL472: Expiry of T200 timer - max. no. of retransmissions not reached	228
4.9.9	DL473: Expiry of T200 timer - max. no. of retransmissions reached	230
4.9.10	DL474: UA response from Base Station - information field empty	236
4.9.11	DL475: UA response from Base Station - information field set.....	237
4.9.12	DL476: DM response from Base Station - establishment rejected.....	238
4.10	Normal Establishment (SAPI 3)	239
4.10.1	DL081: Initiation by Radio Resource	239
4.10.2	DL082: UA response received from Base Station (information field empty)	240
4.10.3	DL083: UA response received from Base Station (information field not empty)	241
4.10.4	DL084: DM response received from Base Station -	242
4.10.5	DL085: Expiry of T200 timer - maximum number of retransmissions not reached	243
4.10.6	DL086: Expiry of T200 timer - maximum number of retransmissions reached	244
4.10.7	DL087: Initiation by the Base Station - MS able to accept establishment.....	246
4.11	Release (SDCCH)	247
4.11.1	DL091: Local end release	247
4.11.2	DL092: Normal release initiated by MS (SDCCH)	248
4.11.3	DL093: Expiry of T200 timer - max. no. of retransmissions not reached.....	249
4.11.4	DL094: Expiry of T200 timer - max. no. of retransmissions reached	250
4.11.5	DL095: Response from base station	252
4.11.6	DL096: Release initiation by base station - MS in idle state	253
4.11.7	DL097: BS initiates release - MS in multiple frame established state	254
4.12	Release (FACCH fullrate)	255
4.12.1	DL101: Normal release initiated by MS (idle state).....	255
4.12.2	DL102: Normal release initiated by MS (multiframe established state)	256
4.12.3	DL103: Expiry of T200 timer - max. no. of retransmissions not reached.....	257
4.12.4	DL104: Expiry of T200 timer - max. no. of retransmissions reached	259
4.12.5	DL105: Response from base station	265
4.12.6	DL106: Release initiation by base station - MS in idle state	266
4.12.7	DL107: BS initiates release - MS in multiple frame established state	267

4.13	Release (FACCH Halfrate)	268
4.13.1	DL501: Normal release initiated by MS (idle state)	268
4.13.2	DL502: Normal release initiated by MS (multiframe established state)	269
4.13.3	DL503: Expiry of T200 timer - max. no. of retransmissions not reached	270
4.13.4	DL504: Expiry of T200 timer - max. no. of retransmissions reached	272
4.13.5	DL505: Response from base station	278
4.13.6	DL506: Release initiation by base station - MS in idle state	279
4.13.7	DL507: BS initiates release - MS in multiple frame established state	280
4.14	Collision of Unnumbered Commands and Responses	281
4.14.1	DL111: Identical commands - establish (SDCCH)	281
4.14.2	DL112: Identical commands - establish (SACCH)	283
4.14.3	DL113: Identical commands - establish (FACCH Fullrate)	285
4.14.4	DL513: Identical commands - establish (FACCH Halfrate)	287
4.14.5	DL114: Identical commands - release (SDCCH)	289
4.14.6	DL115: Identical commands - release (FACCH Fullrate)	291
4.14.7	DL515: Identical commands - release (FACCH Halfrate)	293
4.14.8	DL116: Different commands - establish (SDCCH)	295
4.14.9	DL117: Different commands - establish (SACCH)	297
4.14.10	DL118: Different commands - establish (FACCH Fullrate)	299
4.14.11	DL518: Different commands - establish (FACCH Halfrate)	301
4.14.12	DL119: Different commands - release (SDCCH)	303
4.14.13	DL120: Different commands - release (FACCH Fullrate)	305
4.14.14	DL520: Different commands - release (FACCH Halfrate)	307
4.15	Acknowledged Information Transfer	309
4.15.1	DL131: Transmit single I frame (SDCCH)	309
4.15.2	DL132: Transmit several I frames (SDCCH)	310
4.15.3	DL133: Transmit single I frame (SACCH)	312
4.15.4	DL134: Transmit several I frames (SACCH)	313
4.15.5	DL135: Transmit single I frame (FACCH Fullrate)	315
4.15.6	DL136: Transmit several I frames (FACCH Fullrate)	316
4.15.7	DL535: Transmit single I frame (FACCH Halfrate)	317
4.15.8	DL536: Transmit several I frames (FACCH Halfrate)	318
4.15.9	DL137: Receive single I frame (SDCCH)	319
4.15.10	DL138: Receive several I frames (SDCCH)	320
4.15.11	DL139: Receive single I frame (SACCH)	321
4.15.12	DL140: Receive several I frames (SACCH)	322
4.15.13	DL141: Receive single I frame (FACCH Fullrate)	324
4.15.14	DL142: Receive several I frames (FACCH Fullrate)	325
4.15.15	DL541: Receive single I frame (FACCH Halfrate)	327
4.15.16	DL542: Receive several I frames (FACCH Halfrate)	328
4.15.17	DL143: N(R) sequence error occurs during frame transmission (SDCCH)	330
4.15.18	DL144: N(R) sequence error occurs during frame transmission (SACCH)	331
4.15.19	DL145: N(R) sequence error occurs during frame transmission (FACCH Fullrate)	332
4.15.20	DL545: N(R) sequence error occurs during frame transm. (FACCH Halfrate)	333
4.15.21	DL146: Transmit I frame, acknowledged by I frame (SDCCH)	334
4.15.22	DL147: Transmit I frames, acknowledged by I frame (SDCCH)	336
4.15.23	DL148: Transmit I frames, retransmission on new channel	340
4.15.24	DL149: Retransmission of I frames, Timer Recovery	344
4.15.25	DL150: Resumption with stored messages	346
4.15.26	DL151: RR response frame loss (MS to SS), SDCCH	350
4.15.27	DL152: Customer reported problem from China	353
4.15.28	DL153: Customer reported problem from China, delayed setup	357
4.15.29	DL154: Customer reported problem from China, variation I after step 18	361
4.15.30	DL155: Customer reported problem from China, variation II after step 18	365
4.15.31	DL156: Customer reported problem from China, variant of DL152	368
4.16	Measurement Reporting	372
4.16.1	DL160: Repetition of last Measurement Report (main link = SDCCH)	372

4.16.2	DL161: Suspend and Resume Multiframe operation (SDCCH, Variant I).....	375
4.16.3	DL162: Suspend and Resume Multiframe operation (SDCCH, Variant II)	378
4.16.4	DL163: Suspend and Resume Multiframe operation (SDCCH, Variant III)	380
4.16.5	DL164: Suspend and Resume Multiframe operation (SDCCH, Variant IV)	382
4.16.6	DL170: Repetition of last Measurement Report (main link = FACCH).....	384
4.16.7	DL171: Suspend and Resume Multiframe operation (FACCH, Variant I)	387
4.16.8	DL172: Suspend and Resume Multiframe operation (SDCCH, Variant II)	390
4.16.9	DL173: Suspend and Resume Multiframe operation (FACCH, Variant III)	392
4.16.10	DL174: Suspend and Resume Multiframe operation (FACCH, Variant IV)	394
4.17	GSM 11.10 Test Cases	396
4.17.1	DL200: Normal Initialization (25.2.1.1.1), SDCCH	396
4.17.2	DL201: Normal Initialization (25.2.1.1.1), FACCH Fullrate	398
4.17.3	DL601: Normal Initialization (25.2.1.1.1), FACCH Halfrate	399
4.17.4	DL202: Loss of UA frame (25.2.1.1.2.1), SDCCH	400
4.17.5	DL203: Loss of UA frame (25.2.1.1.2.1), FACCH Fullrate	402
4.17.6	DL603: Loss of UA frame (25.2.1.1.2.1), FACCH Halfrate	404
4.17.7	DL204: UA frame with different information field (25.2.1.1.2.2), SDCCH	406
4.17.8	DL205: UA frame with different info field (25.2.1.1.2.2), FACCH Fullrate	408
4.17.9	DL605: UA frame with different info field (25.2.1.1.2.2), FACCH Halfrate	410
4.17.10	DL206: Wrong frames in response to SABM (25.2.1.1.2.2), SDCCH	412
4.17.11	DL207: Wrong frames in response to SABM (25.2.1.1.2.2), FACCH Fullrate	414
4.17.12	DL607: Wrong frames in response to SABM (25.2.1.1.2.2), FACCH Halfrate	417
4.17.13	DL208: Initialisation denial (25.2.1.1.3), SDCCH	419
4.17.14	DL209: Initialisation denial (25.2.1.1.3), FACCH Fullrate	420
4.17.15	DL609: Initialisation denial (25.2.1.1.3), FACCH halfrate	421
4.17.16	DL210: Total Initialisation failure (25.2.1.1.4), SDCCH	422
4.17.17	DL211: Total Initialisation failure (25.2.1.1.4), FACCH Fullrate	425
4.17.18	DL611: Total Initialisation failure (25.2.1.1.4), FACCH Halfrate	432
4.17.19	DL212: Initialisation without Contention Resolution (25.2.1.2.1)	439
4.17.20	DL213: Initialisation Failure (25.2.1.2.2)	443
4.17.21	DL214: Initialisation Denial (25.2.1.2.3).....	449
4.17.22	DL215: Total Initialisation Failure (25.2.1.2.4)	455
4.17.23	DL216: Normal Information Transfer (25.2.2.1), SDCCH	467
4.17.24	DL217: Normal Information Transfer (25.2.2.1), FACCH Fullrate	475
4.17.25	DL617: Normal Information Transfer (25.2.2.1), FACCH Halfrate	483
4.17.26	DL218: Receipt of I frame in Timer Recovery (25.2.2.2), SDCCH	491
4.17.27	DL219: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Fullrate	494
4.17.28	DL619: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Halfrate	498
4.17.29	DL620: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Halfrate (II)	502
4.17.30	DL220: Segmentation and Concatenation (25.2.2.3), SDCCH	504
4.17.31	DL221: Segmentation and Concatenation (25.2.2.3), FACCH Fullrate	509
4.17.32	DL621: Segmentation and Concatenation (25.2.2.3), FACCH Halfrate	513
4.17.33	DL222: Normal Layer 2 Disconnection (25.2.3), SDCCH	517
4.17.34	DL223: Normal Layer 2 Disconnection (25.2.3), FACCH Fullrate	518
4.17.35	DL623: Normal Layer 2 Disconnection (25.2.3), FACCH Halfrate	519
4.17.36	DL224: I Frame Loss (MS to SS), (25.2.4.1), SDCCH	520
4.17.37	DL225: I Frame Loss (MS to SS), (25.2.4.1), FACCH Fullrate (I)	527
4.17.38	DL250: I Frame Loss (MS to SS), (25.2.4.1), FACCH Fullrate (II)	544
4.17.39	DL625: I Frame Loss (MS to SS), (25.2.4.1), FACCH Halfrate (I).....	559
4.17.40	DL650: I Frame Loss (MS to SS), (25.2.4.1), FACCH Halfrate (II)	576
4.17.41	DL226: RR response frame loss (MS to SS), (25.2.4.3), SDCCH	586
4.17.42	DL227: RR response frame loss (MS to SS), (25.2.4.3), FACCH Fullrate	588
4.17.43	DL627: RR response frame loss (MS to SS), (25.2.4.3), FACCH Halfrate	590
4.17.44	DL228: I frame with C bit set to zero (25.2.5.1), SDCCH.....	592
4.17.45	DL229: I frame with C bit set to zero (25.2.5.1), FACCH Fullrate	594
4.17.46	DL629: I frame with C bit set to zero (25.2.5.1), FACCH Halfrate.....	595
4.17.47	DL230: SABM Frame with C bit set to zero (25.2.5.2), SDCCH	596
4.17.48	DL231: SABM Frame with C bit set to zero (25.2.5.2), FACCH Fullrate	598
4.17.49	DL631: SABM Frame with C bit set to zero (25.2.5.2), FACCH Halfrate	600

4.17.50	DL232: N(S) Sequence Error (25.2.6.1), SDCCH	602
4.17.51	DL233: N(S) Sequence Error (25.2.6.1), FACCH Fullrate	605
4.17.52	DL633: N(S) Sequence Error (25.2.6.1), FACCH Halfrate	608
4.17.53	DL234: N(R) Sequence Error (25.2.6.2), SDCCH	611
4.17.54	DL235: N(R) Sequence Error (25.2.6.2), FACCH Fullrate	612
4.17.55	DL635: N(R) Sequence Error (25.2.6.2), FACCH Halfrate	613
4.17.56	DL236: Test on receipt of invalid frames (25.2.7), SDCCH	614
4.17.57	DL237: Test on receipt of invalid frames II(25.2.7), SDCCH.....	621
4.17.58	DL238: Test on receipt of invalid frames (25.2.7), FACCH Fullrate	626
4.17.59	DL638: Test on receipt of invalid frames (25.2.7), FACCH Halfrate	634
4.18	Additional Multilayer Tests	642
4.18.1	DL300: Mobile Terminated Call	642
4.18.2	DL301: Mobile Terminated SMS (SDCCH).....	645
4.18.3	DL302: Mobile Terminated SMS (SDCCH), very long message	647
4.18.4	DL303: N(R) Sequence Error (25.2.6.2), SDCCH	651
4.18.5	DL304: N(R) Sequence Error (25.2.6.2), FACCH Fullrate	653
4.18.6	DL305: N(R) Sequence Error (25.2.6.2), FACCH Halfrate	654
4.18.7	DL306: Collision Problem Establishment and incoming Segments	655
4.18.8	DL307: SMS-Testcase 34.2.1 on the Anite (Part 1)	658
4.18.9	DL308: SMS-Testcase 34.2.1 on the Anite (Part 2)	667
4.18.10	DL309: Handover Testcase on the Anite	669
4.18.11	DL311: Mobile Terminated Call (slow network)	675
Appendices.....		679
A.	Acronyms	679
B.	Glossary	679

List of Figures and Tables

List of References

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

1.1 References

- [1] GSM 2.81, Line Identification Supplementary Services - Stage 1
ETS 300 514, ETSI, September 1994
- [2] GSM 2.82, Call Forwarding Supplementary Services - Stage 1
ETS 300 515, ETSI, September 1994
- [3] GSM 2.83, Call Waiting and Call Hold Supplementary Services - Stage 1
ETS 300 516, ETSI, September 1994
- [4] GSM 2.84, Multi Party Supplementary Services - Stage 1
ETS 300 517, ETSI, September 1994
- [5] GSM 2.85, Closed User Group Supplementary Services - Stage 1
ETS 300 518, ETSI, September 1994
- [6] GSM 2.86, Advice of Charge Supplementary Services - Stage 1
ETS 300 519, ETSI, September 1994
- [7] GSM 2.88, Call Barring Supplementary Services - Stage 1
ETS 300 520, ETSI, September 1994
- [8] GSM 3.14, Support of Dual Tone Multi Frequency Signalling via the GSM System
ETS 300 532, ETSI, April 1994
- [9] GSM 3.40, Technical Realization of the Short Message Service Point-to-Point
ETS 300 536, ETSI, January 1996
- [10] GSM 3.41, Technical Realization of Short Message Service Cell Broadcast
ETS 300 537, ETSI, June 1995
- [11] GSM 3.81, Line Identification Supplementary Services - Stage 2
ETS 300 542, ETSI, February 1995
- [12] GSM 3.82, Call Forwarding Supplementary Services - Stage 2
ETS 300 543, ETSI, February 1995
- [13] GSM 3.83, Call Waiting and Call Hold Supplementary Services - Stage 2
ETS 300 544, ETSI, November 1994
- [14] GSM 3.84, Multi Party Supplementary Services - Stage 2
ETS 300 545, ETSI, November 1994
- [15] GSM 3.85, Closed User Group Supplementary Services - Stage 2
ETS 300 546, ETSI, January 1996
- [16] GSM 3.86, Advice of Charge Supplementary Services - Stage 2
ETS 300 547, ETSI, March 1995
- [17] GSM 3.88, Call Barring Supplementary Services - Stage 2
ETS 300 548, ETSI, November 1994
- [18] GSM 4.01, MS-BSS Interface General Aspects and Principles
ETS 300 550, ETSI, September 1994
- [18a] GSM 4.03, MS-BSS Interface Channel Structures and Access Capabilities
ETS 300 552, ETSI, September 1994
- [19] GSM 4.05, Data Link Layer General Aspects
ETS 300 554, ETSI, September 1994
- [20] GSM 4.06, MS-BSS Interface Data Link Layer Specification
ETS 300 555, ETSI, September 1994
- [21] GSM 4.07, Mobile Radio Interface Signalling Layer 3 General Aspects
ETS 300 556, ETSI, February 1995

- [22] GSM 4.08, Mobile Radio Interface Layer 3 Specification
ETS 300 557, ETSI, January 1996
- [23] GSM 4.10, Mobile Radio Interface Layer 3 Supplementary Services Specification
General Aspects
ETS 300 558, ETSI, February 1995
- [24] GSM 4.11, Point-to-Point Short Message Service Support on Mobile Radio Interface
ETS 300 559, ETSI, October 1995
- [25] GSM 4.12, Short Message Service Cell Broadcast Support on Mobile Radio Interface
ETS 300 560, ETSI, January 1996
- [26] GSM 4.80, Mobile Radio Interface Supplementary Services Specification Formats and Coding
ETS 300 564, ETSI, February 1995
- [27] GSM 4.81, Line Identification Supplementary Services - Stage 3
ETS 300 565, ETSI, February 1995
- [28] GSM 4.82, Call Forwarding Supplementary Services - Stage 3
ETS 300 566, ETSI, February 1995
- [29] GSM 4.83, Call Waiting and Call Hold Supplementary Services - Stage 3
ETS 300 567, ETSI, February 1995
- [30] GSM 4.84, Multi Party Supplementary Services - Stage 3
ETS 300 568, ETSI, February 1995
- [31] GSM 4.85, Closed User Group Supplementary Services - Stage 3
ETS 300 569, ETSI, February 1995
- [32] GSM 4.86, Advice of Charge Supplementary Services - Stage 3
ETS 300 570, ETSI, February 1995
- [33] GSM 4.88, Call Barring Supplementary Services - Stage 3
ETS 300 571, ETSI, February 1995
- [34] GSM 5.01, Physical Layer on the Radio Path General Description
ETS 300 573, ETSI, October 1995
- [35] GSM 5.02, Multiplexing and Multiple Access on the Radio Path
ETS 300 574, ETSI, January 1996
- [36] GSM 5.08, Radio Sub-system Link Control
ETS 300 578, ETSI, January 1996
- [37] GSM 5.10, Radio Sub-system Synchronisation
ETS 300 579, ETSI, October 1995
- [38] Service Access Point MMREG
6147.100.96.100; Condat GmbH
- [39] Service Access Point MNCC
6147.101.96.100; Condat GmbH
- [40] Service Access Point MNSS
6147.102.96.100; Condat GmbH
- [41] Service Access Point MNSMS
6147.103.96.100; Condat GmbH
- [42] Service Access Point MMCC
6147.104.97.100; Condat GmbH
- [43] Service Access Point MMSS
6147.105.97.100; Condat GmbH
- [44] Service Access Point MMSMS
6147.106.97.100; Condat GmbH

- [45] Service Access Point RR
6147.107.97.100; Condat GmbH
- [46] Service Access Point SIM
6147.108.97.100; Condat GmbH
- [47] Service Access Point MPH
6147.109.96.100; Condat GmbH
- [48] Service Access Point DL
6147.110.96.100; Condat GmbH
- [49] Service Access Point MDL
6147.111.96.100; Condat GmbH
- [50] Service Access Point PH
6147.112.97.100; Condat GmbH
- [51] Service Access Point MMI
6147.113.96.100; Condat GmbH
- [52] Message Sequence Charts CC
6147.200.97.100; Condat GmbH
- [53] Message Sequence Charts SS
6147.201.97.100; Condat GmbH
- [54] Message Sequence Charts SMS
6147.202.97.100; Condat GmbH
- [55] Message Sequence Charts MM
6147.203.97.100; Condat GmbH
- [56] Message Sequence Charts RR
6147.204.96.100; Condat GmbH
- [57] Message Sequence Charts DL
6147.205.96.100; Condat GmbH
- [58] Users Guide
6147.300.96.100; Condat GmbH
- [59] Test Specification CC
6147.400.97.100; Condat GmbH
- [60] Test Specification SS
6147.401.97.100; Condat GmbH
- [61] Test Specification SMS
6147.402.97.100; Condat GmbH
- [62] Test Specification MM
6147.403.97.100; Condat GmbH
- [63] Test Specification RR
6147.404.97.100; Condat GmbH
- [64] Test Specification DL
6147.405.97.100; Condat GmbH
- [65] Test Specification CCD
6147.406.97.100; Condat GmbH
- [66] SDL Specification CC
6147.500.97.100; Condat GmbH
- [67] SDL Specification SS
6147.501.97.100; Condat GmbH

- [68] SDL Specification SMS
6147.502.97.100; Condat GmbH
- [69] SDL Specification MM
6147.503.97.100; Condat GmbH
- [70] SDL Specification RR
6147.504.97.100; Condat GmbH
- [71] SDL Specification DL
6147.505.97.100; Condat GmbH
- [72] Message Specification CC
6147.600.97.100; Condat GmbH
- [73] Message Specification SS
6147.601.97.100; Condat GmbH
- [74] Message Specification SMS
6147.602.97.100; Condat GmbH
- [75] Message Specification MM
6147.603.97.100; Condat GmbH
- [76] Message Specification RR
6147.604.97.100; Condat GmbH
- [77] Message Specification DL
6147.605.97.100; Condat GmbH
- [78] Technical Documentation CC
6147.700.97.100; Condat GmbH
- [79] Technical Documentation SS
6147.701.97.100; Condat GmbH
- [80] Technical Documentation SMS
6147.702.97.100; Condat GmbH
- [81] Technical Documentation MM
6147.703.97.100; Condat GmbH
- [82] Technical Documentation RR
6147.704.97.100; Condat GmbH
- [83] Technical Documentation DL
6147.705.97.100; Condat GmbH
- [84] Technical Documentation CCD
6147.706.97.100; Condat GmbH

1.2 Abbreviations

AGCH	Access Grant Channel
BCCH	Broadcast Control Channel
BS	Base Station
BSIC	Base Station Identification Code
CBCH	Cell Broadcast Channel
CBQ	Cell Bar Qualify
CC	Call Control
CCCH	Common Control Channel
CCD	Condat Coder Decoder
CKSN	Ciphering Key Sequence Number
C/R	Command / Response
C1	Path Loss Criterion
C2	Reselection Criterion

DCCH	Dedicated Control Channel
DISC	Disconnect Frame
DL	Data Link Layer
DM	Disconnected Mode Frame
EA	Extension Bit Address BEGINARRAY
EL	Extension Bit Length BEGINARRAY
EMMI	Electrical Man Machine Interface
F	Final Bit
FACCH	Fast Associated Control Channel
FHO	Forced Handover
GP	Guard Period
GSM	Global System for Mobile Communication
HPLMN	Home Public Land Mobile Network
I	Information Frame
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
Kc	Authentication Key
L	Length Indicator
LAI	Location Area Information
LPD	Link Protocol Discriminator
M	More Data Bit
MCC	Mobile Country Code
MM	Mobility Management
MMI	Man Machine Interface
MNC	Mobile Network Code
MS	Mobile Station
NCC	National Colour Code
NECI	New Establishment Causes included
N®	Receive Number
N(S)	Send Number
OTD	Observed Time Difference
P	Poll Bit
PCH	Paging Channel
PDU	Protocol Description Unit
P/F	Poll / Final Bit
PL	Physical Layer
PLMN	Public Land Mobile Network
RACH	Random Access Channel
REJ	Reject Frame
RNR	Receive Not Ready Frame
RR	Radio Resource Management
RR	Receive Ready Frame
RTD	Real Time Difference
SABM	Set Asynchronous Balanced Mode
SACCH	Slow Associated Control Channel
SAP	Service Access Point
SAPI	Service Access Point Identifier
SDCCH	Slow Dedicated Control Channel
SIM	Subscriber Identity Module
SMS	Short Message Service
SMSCB	Short Message Service Cell Broadcast
SS	Supplementary Services
TCH	Traffic Channel
TCH/F	Traffic Channel Full Rate
TCH/H	Traffic Channel Half Rate
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity
UA	Unnumbered Acknowledgement Frame
UI	Unnumbered Information Frame

VPLMN Visiting Public Land Mobile Network
V(A) Acknowledgement State Variable
V@ Receive State Variable
V(S) Send State Variable

1.3 Terms

Entity: Program which executes the functions of a layer

Message: A message is a data unit which is transferred between the entities of the same layer (peer-to-peer) of the mobile and infrastructure side. Message is used as a synonym to protocol data unit (PDU). A message may contain several information elements.

Primitive: A primitive is a data unit which is transferred between layers on one component (mobile station or infrastructure). The primitive has an operation code which identifies the primitive and its parameters.

Service Access Point: A Service Access Point is a data interface between two layers on one component (mobile station or infrastructure).

2 Overview

The Protocol Stacks are used to define the functionality of the GSM protocols for interfaces. The GSM specifications are normative when used to describe the functionality of interfaces, but the stacks and the subdivision of protocol layers does not imply or restrict any implementation.

The base of the Protocol Stack rests on the physical layer.

The Data Link Layer (DL) is used to handle an acknowledged connection between mobile and base station. The LAPDm protocol is used.

Radio Resource (RR) manages the resources of the air-interface. That means configuration of physical layer, cell selection and cell reselection, data transfer, RR-Connection handling.

Mobility Management (MM) handles registration aspects for the mobile station. It detects changes of location areas and updates a mobile station in the new location area.

Call Control (CC) provides the call functionality. This includes call establishment, call maintenance procedures like Hold, Retrieve or Modify, and call disconnection.

Supplementary Services (SS) handles all call independent supplementary services like call forwarding or call barring.

Short Message Services (SMS) is used for sending and receiving point-to-point short messages. Additionally the reception of cell broadcast short messages is included.

The man machine interface (MMI) is the interface to the user. Normally it is connected with a keypad as input device and a display as output device.

Between the several entities data interfaces are defined. These data interfaces are called Service Access Points (SAPs), indicating that an upper layer uses the services of a lower layer.

The GSM specification do not set out any implementation of the Protocol Stack. The following diagrams show the implementation described in all these documents for the mobile station. All entities except the Man Machine Interface and Physical Layer are implemented as part of the Protocol Stack.

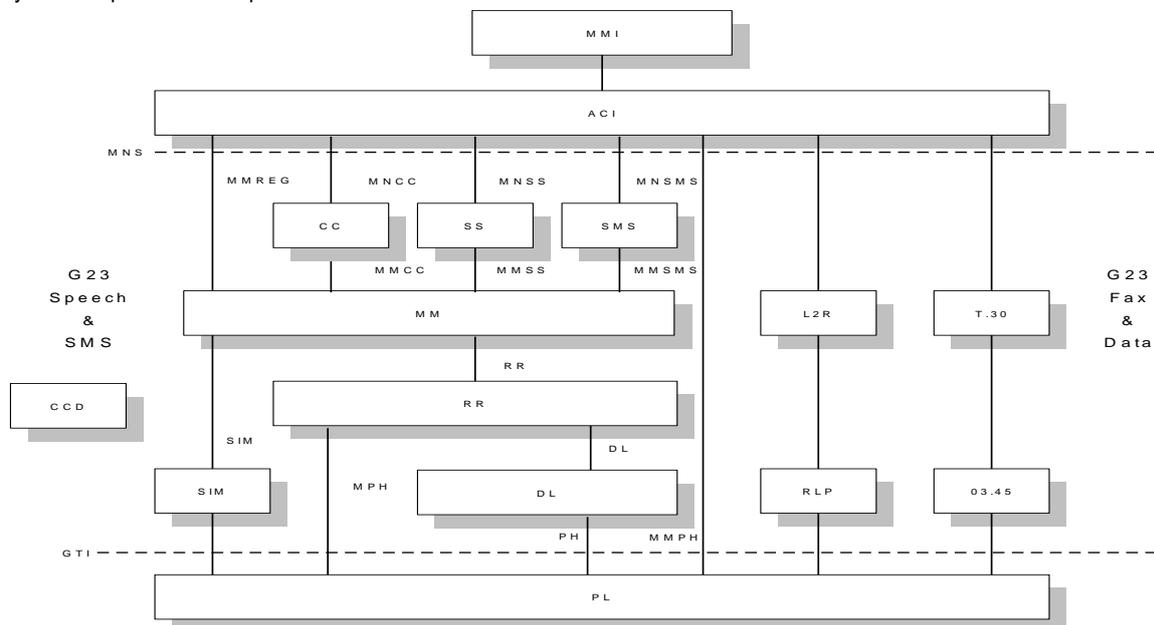


Figure 1: Mobile-station protocol architecture

This document describes the tests for the data link layer.

3 Parameters

/* DECLARATIONS */

DECLARATION (BBIS_UNACK_DL)
DECLARATION (DM_A_DL)
DECLARATION (L3_MSG_L0_UL)
DECLARATION (L3_MSG_L06_1_DL)
DECLARATION (DL311_I_CMD_0_0_UL)
DECLARATION (DL311_I_CMD_0_0_DL)
DECLARATION (DL311_L3_SETUP)
DECLARATION (DL311_L3_CALL_CONF)
DECLARATION (DL311_I_CMD_1_0_P_UL)
DECLARATION (DL311_RR_RSP_1_DL)
DECLARATION (DL311_REJ_RSP_1_F_DL)
DECLARATION (DL311_I_CMD_1_1_UL)
DECLARATION (DL311_RR_RSP_1_F_DL)
DECLARATION (DL311_I_CMD_1_1_P_UL)
DECLARATION (DL311_I_CMD_2_1_DL)
DECLARATION (DL311_I_CMD_2_1_P_UL)
DECLARATION (DL311_REJ_RSP_2_F_DL)
DECLARATION (UI_SACCH_DL)
DECLARATION (L3_MSG_UI_SACCH)
DECLARATION (L3_MSG_L06_UL)
DECLARATION (L3_MSG_1)
DECLARATION (L3_MSG_3)
DECLARATION (L3_MSG_4)
DECLARATION (L3_MSG_5)
DECLARATION (L3_MSG_6)
DECLARATION (L3_MSG_HO)
DECLARATION (L3_MSG_7)
DECLARATION (L3_MSG_8)
DECLARATION (L3_MSG_9)
DECLARATION (L3_MSG_10)
DECLARATION (L3_MSG_11)
DECLARATION (L3_MSG_12)
DECLARATION (SABM_A_S3_SA_UL)
DECLARATION (SABM_B_I06_UL)
DECLARATION (SABM_MSG_1)
DECLARATION (SABM_MSG_11)
DECLARATION (SABM_MSG_13)
DECLARATION (I_MSG_3)
DECLARATION (I_MSG_3B)
DECLARATION (I_HO_1)
DECLARATION (I_HO_2)
DECLARATION (I_MSG_4)
DECLARATION (I_MSG_5)
DECLARATION (I_MSG_5A)
DECLARATION (I_MSG_5B)
DECLARATION (I_MSG_5C)
DECLARATION (I_MSG_5D)
DECLARATION (I_MSG_6)
DECLARATION (I_MSG_6B)
DECLARATION (I_MSG_7)
DECLARATION (I_MSG_8)
DECLARATION (I_MSG_9)
DECLARATION (I_MSG_10)

DECLARATION (I_MSG_11)
DECLARATION (I_MSG_12)
DECLARATION (I_MSG_12B)
DECLARATION (I_MSG_14)
DECLARATION (UA_A_S3_SA_DL)
DECLARATION (UA_B_I06_1_DL)
DECLARATION (UA_MSG_1)
DECLARATION (UA_MSG_11)
DECLARATION (UA_MSG_13)
DECLARATION (UA_B_I06_2_DL)
DECLARATION (ERR_UN SOL_UA_DL)
DECLARATION (ERR_UN SOL_DM_DL)
DECLARATION (ERR_U_PARAM_DL)
DECLARATION (ERR_S_PARAM_DL)
DECLARATION (ERR_I_M_BIT_DL)
DECLARATION (ERR_I_FR_LEN)
DECLARATION (ERR_I_FR_NOT_IMP)
DECLARATION (SABM_A_UL)
DECLARATION (UA_A_DL)
DECLARATION (I_I06_UL)
DECLARATION (RR_A_NR1_DL)
DECLARATION (SABM_A_S3_UL)
DECLARATION (UA_A_S3_SD_DL)
DECLARATION (UA_B_S3_I04_DL)
DECLARATION (DM_A_S3_DL)
DECLARATION (SABM_A_S3_SD_DL)
DECLARATION (SABM_A_S3_SA_DL)
DECLARATION (UA_A_S3_UL)
DECLARATION (DISC_A_UL)
DECLARATION (DISC_A_DL)
DECLARATION (DM_A_UL)
DECLARATION (UA_A_UL)
DECLARATION (SABM_B_I06_DL)
DECLARATION (UA_A_S3_SA_UL)
DECLARATION (DISC_A_S3_UL)
DECLARATION (DM_A_S3_SA_DL)
DECLARATION (L3_MSG_L16_1_UL)
DECLARATION (L3_MSG_L24_UL)
DECLARATION (L3_MSG_L16_2_UL)
DECLARATION (L3_MSG_L24_DL)
DECLARATION (I_I16_UL)
DECLARATION (I_I20_UL)
DECLARATION (I_I04_UL)
DECLARATION (I_I06_S3_DL)
DECLARATION (I_I16_S3_UL)
DECLARATION (I_I18_S3_UL)
DECLARATION (I_I06_S3_UL)
DECLARATION (I_I16_DL)
DECLARATION (I_MSG_2)
DECLARATION (L3_MSG_2)
DECLARATION (RR_MSG_2)
DECLARATION (RR_MSG_3)
DECLARATION (RR_MSG_5)
DECLARATION (RR_MSG_12B)
DECLARATION (RR_MSG_5C)
DECLARATION (RR_MSG_5D)
DECLARATION (RR_MSG_6)
DECLARATION (RR_MSG_7)
DECLARATION (RR_MSG_8)

DECLARATION (RR_MSG_9)
DECLARATION (RR_MSG_11)
DECLARATION (RR_MSG_12)
DECLARATION (RR_MSG_14)
DECLARATION (I_HO_RETRANSMIT)
DECLARATION (I_I20_DL)
DECLARATION (I_I20_DL3)
DECLARATION (I_I18_DL3)
DECLARATION (I_I04_NS1_DL)
DECLARATION (I_I04_NS1_DL3)
DECLARATION (I_I06_NS1_DL3)
DECLARATION (I_I16_S3_DL)
DECLARATION (I_I18_S3_DL)
DECLARATION (RR_A_S3_NR1_DL)
DECLARATION (RR_A_UL)
DECLARATION (RR_A_UL3)
DECLARATION (RR_A_NR2_UL)
DECLARATION (RR_A_NR2_UL3)
DECLARATION (RR_A_NR2_DL)
DECLARATION (RR_A_S3_UL)
DECLARATION (RR_A_S3_NR2_DL)
DECLARATION (EMPTY_FRAME_SACCH)
DECLARATION (EMPTY_FRAME_DCCH)
DECLARATION (I_SETUP)
DECLARATION (L3_SETUP)
DECLARATION (L3_SETUP_MSG)
DECLARATION (I_SETUP_MSG_P0)
DECLARATION (I_SETUP_MSG_P1)
DECLARATION (I_HO_CMD_P0_S1)
DECLARATION (I_HO_CMD_P1_S1)
DECLARATION (I_HO_CMD_P0_S2)
DECLARATION (RR_HO_CMD)
DECLARATION (REJ_HO_CMD)
DECLARATION (L3_HO_CMD)
DECLARATION (I_CIPH_REQ)
DECLARATION (RR_CIPH_REQ)
DECLARATION (L3_CIPH_REQ)
DECLARATION (L3_CIPH_RSP)
DECLARATION (I_CIPH_RSP)
DECLARATION (RR_CIPH_RSP)
DECLARATION (RR_SETUP)
DECLARATION (I_CALL_CONFIRM)
DECLARATION (I_CALL_CONFIRM_2)
DECLARATION (I_CALL_CONFIRM_3)
DECLARATION (RR_CALL_CONFIRM_3)
DECLARATION (RR_CALL_CONFIRM_4)
DECLARATION (L3_CALL_CONFIRM)
DECLARATION (I_ASSIGN_CMD)
DECLARATION (L3_ASSIGN_CMD)
DECLARATION (RR_ASSIGN_CMD)
DECLARATION (I_ASSIGN_FAILURE)
DECLARATION (ASSIGN_FAILURE)
DECLARATION (RR_ASSIGN_FAILURE)
DECLARATION (I_ALERT)
DECLARATION (I_ALERT_2)
DECLARATION (L3_ALERT)
DECLARATION (RR_ALERT)
DECLARATION (I_CONNECT)
DECLARATION (L3_CONNECT)

DECLARATION (RR_CONNECT)
DECLARATION (I_ASSIGN_CMP)
DECLARATION (L3_ASSIGN_CMP)
DECLARATION (RR_ASSIGN_CMP)
DECLARATION (UI_FRAME_DOWNLINK)
DECLARATION (UI_FRAME_UPLINK)
DECLARATION (MEAS_REPORT_1)
DECLARATION (MEAS_REPORT_1_L1)
DECLARATION (MEAS_REPORT_0_L1)
DECLARATION (MEAS_REPORT_2)
DECLARATION (MEAS_REPORT_2_L1)
DECLARATION (MEAS_REPORT_INV_L1)
DECLARATION (MEAS_REPORT_NO_NCELL)
DECLARATION (UI_FRAME_UP_SACCH)
DECLARATION (UI_FRAME_UP_SACCH_MEAS)
DECLARATION (UI_FRAME_UP_SACCH_M2)
DECLARATION (SABM_PAGING_RESP)
DECLARATION (UA_PAGING_RESP)
DECLARATION (UA_NO_PAGING_RESP)
DECLARATION (L3_PAGING_RESP)
DECLARATION (LONG_MESSAGE)
DECLARATION (I00)
DECLARATION (I01)
DECLARATION (I02)
DECLARATION (I03)
DECLARATION (I04)
DECLARATION (I05)
DECLARATION (I06)
DECLARATION (I07)
DECLARATION (I00_FINAL)
DECLARATION (SMS_MSG)
DECLARATION (RR00)
DECLARATION (RR01)
DECLARATION (RR02)
DECLARATION (RR03)
DECLARATION (RR04)
DECLARATION (RR05)
DECLARATION (RR06)
DECLARATION (RR07)
DECLARATION (I_FRAME_1)
DECLARATION (RR_FRAME_1)
DECLARATION (REJ_FRAME_1)
DECLARATION (DM_FRAME)
DECLARATION (I_ASSIGN_CMD_SDCCH)
DECLARATION (ASSIGN_CMD_SDCCH)
DECLARATION (I_ASSIGN_CMD_FACCH)
DECLARATION (I_ASSIGN_CMD_2)
DECLARATION (ASSIGN_CMD_FACCH)
DECLARATION (BS_I_0_0)
DECLARATION (BS_I_0_0_P)
DECLARATION (BS_I_0_0_C)
DECLARATION (BS_I_0_1)
DECLARATION (BS_I_0_7)
DECLARATION (BS_I_0_1_P1)
DECLARATION (BS_I_0_7_P1)
DECLARATION (BS_I_0_1_P)
DECLARATION (BS_I_0_4)
DECLARATION (BS_I_1_0)
DECLARATION (BS_I_1_1_M1)

DECLARATION (BS_I_2_1_M2)
DECLARATION (BS_I_2_0)
DECLARATION (BS_I_3_0)
DECLARATION (BS_I_4_0)
DECLARATION (BS_I_5_1)
DECLARATION (BS_I_6_2)
DECLARATION (BS_I_7_3)
DECLARATION (BS_RR_0)
DECLARATION (BS_RR_0_P)
DECLARATION (BS_RR_1)
DECLARATION (BS_RR_1_P)
DECLARATION (BS_RR_1_F)
DECLARATION (BS_RR_2)
DECLARATION (BS_RR_2_F)
DECLARATION (BS_RR_3)
DECLARATION (BS_RR_5)
DECLARATION (BS_RR_6)
DECLARATION (BS_RR_7)
DECLARATION (ID_REQUEST)
DECLARATION (ID_RESPONSE)
DECLARATION (MS_I_0_1_M1)
DECLARATION (MS_I_1_1_M2)
DECLARATION (MS_I_1_2_M2)
DECLARATION (MS_I_1_0_M1)
DECLARATION (MS_I_2_0_M2)
DECLARATION (MS_I_0_1)
DECLARATION (MS_I_0_1_P)
DECLARATION (MS_I_0_2_P)
DECLARATION (MS_I_0_5)
DECLARATION (MS_I_1_2)
DECLARATION (MS_I_1_6)
DECLARATION (MS_I_2_7)
DECLARATION (MS_I_3_0)
DECLARATION (MS_I_4_1)
DECLARATION (MS_I_5_1)
DECLARATION (MS_I_6_1)
DECLARATION (MS_I_7_1)
DECLARATION (MS_RR_0)
DECLARATION (MS_RR_0_F)
DECLARATION (MS_RR_1)
DECLARATION (MS_RR_1_F)
DECLARATION (MS_RR_2)
DECLARATION (MS_RR_3)
DECLARATION (MS_RR_4)
DECLARATION (MS_RR_5)
DECLARATION (MS_RR_6)
DECLARATION (MS_RR_7)
DECLARATION (DISC_FRAME)
DECLARATION (UA_FRAME)
DECLARATION (BS_SABM_C)
DECLARATION (MS_REJ_1)
DECLARATION (MS_REJ_1_F)
DECLARATION (INV_1)
DECLARATION (INV_2)
DECLARATION (INV_3)
DECLARATION (INV_4)
DECLARATION (INV_5)
DECLARATION (INV_6)
DECLARATION (INV_7)

DECLARATION (INV_8)
 DECLARATION (INV_9)
 DECLARATION (INV_10)
 DECLARATION (INV_11)
 DECLARATION (INV_12)
 DECLARATION (INV_13)
 DECLARATION (INV_14)
 DECLARATION (INV_15)
 DECLARATION (I_DL151_UL)
 DECLARATION (I_DL151_DL)
 DECLARATION (I_DL151_ULa)
 DECLARATION (I_DL151_DLa)
 DECLARATION (I_DL151_ULb)
 DECLARATION (I_DL151_ULd)
 DECLARATION (I_DL151_DLb)
 DECLARATION (REJ_DL151_ULc)
 DECLARATION (I_DL151_DLc)
 DECLARATION (I_DL151_DLd)

/* For testcase DL152ff >>> */

DECLARATION (L3_MSG_CM_SERV_REQ)
 DECLARATION (SABM_MSG_CM_SERV_REQ)
 DECLARATION (UA_MSG_CM_SERV_REQ)
 DECLARATION (L3_MSG_CM_SERV_ACC)
 DECLARATION (I_MSG_CM_SERV_ACC)
 DECLARATION (L3_MSG_CLASSM_CHG)
 DECLARATION (I_MSG_CLASSM_CHG_P0)
 DECLARATION (I_MSG_CLASSM_CHG_P1)
 DECLARATION (RR_RSP_DL_NR1_F0)
 DECLARATION (RR_RSP_DL_NR2_F0)
 DECLARATION (RR_RSP_DL_NR3_F0)
 DECLARATION (REJ_RSP_DL_NR1_F1)
 DECLARATION (REJ_RSP_DL_NR2_F1)
 DECLARATION (RR_RSP_DL_NR1_F1)
 DECLARATION (I_I20_UL_NR1_NS1)
 DECLARATION (I_I20_UL_NR1_NS1_P)
 DECLARATION (I_I04_UL_NR1_NS2)
 DECLARATION (L3_MSG_SETUP)
 DECLARATION (I_SETUP_UL_NR1_NS1)
 DECLARATION (I_SETUP_UL_NR1_NS1_P)

/* <<< For testcase DL152ff */

/* ARRAY DEFINITIONS */

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x1D	pseudo length
0x11, ...	7 Byte content
0x2B, ...	rest octets

*/

BEGINARRAY (BBIS_UNACK_DL, 27)
 0xB8, 0x00,
 0x00, 0x00,
 0x1D,

0x11, 0x0F, 0x0E, 0x0D, 0x0C, 0x0B, 0x0A,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x1F	DM frame, F= 1
0x01	length 0
0x2B	fill bits

*/

BEGINARRAY (DM_A_DL, 8)

0xB8, 0x00,
0x00, 0x00,
0x01,
0x1F,
0x01,
0x2B

ENDARRAY

/*

0x00, 0x00	length in bits
0x00, 0x00	offset in bits

*/

BEGINARRAY (L3_MSG_L0_UL, 4)

0x00, 0x00,
0x00, 0x00

ENDARRAY

/*

0xB0, 0x00	length in bits
0x08, 0x00	offset in bits
0x1D	pseudo length (7 Byte)
0x11, ...	content
0x2B, ...	rest octets

*/

```
BEGINARRAY (L3_MSG_L06_1_DL, 27)
    0xB0, 0x00,
    0x08, 0x00,
    0x1D,
    0x11, 0x0F, 0x0E, 0x0D, 0x0C, 0x0B, 0x0A,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command (UL)
0x00	I, Nr = 0, N(S)=0, P=0
0x29	length = 10, M=0
0x06, 0x16, ...	CLASSM CHG
0x2B	fill bits

*/

```
BEGINARRAY (DL311_I_CMD_0_0_UL, 27)
    0xB8, 0x00, 0x00, 0x00,
    0x01, 0x00, 0x29,
    0x06, 0x16, 0x03, 0x33, 0x19, 0x81, 0x20, 0x02, 0x60, 0x14,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command (DL)
0x00	I, Nr = 0, N(S)=0, P=0
0x4d	length = 19, M=0
0x03, 0x05	SETUP
0x2B	fill bits

*/

```
BEGINARRAY (DL311_I_CMD_0_0_DL, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x03, 0x00, 0x4d,
    0x03, 0x05, 0x04, 0x01, 0xa0, 0x5c, 0x08, 0x11, 0x81, 0x94,
    0x03, 0x93, 0x90, 0x44, 0x41, 0x7d, 0x02, 0x91, 0x81, 0x2b
```

ENDARRAY

```
BEGINARRAY (DL311_L3_SETUP, 26)
    0x98, 0x00,          /* length in bits */
    0x18, 0x00,          /* offset in bits */
    0x00, 0x00, 0x00,    /* free for layer 2 */
    0x03, 0x05, 0x04, 0x01, 0xa0, 0x5c, 0x08, 0x11, 0x81, 0x94,
    0x03, 0x93, 0x90, 0x44, 0x41, 0x7d, 0x02, 0x91, 0x81
```

ENDARRAY

```
BEGINARRAY (DL311_L3_CALL_CONF, 15)
    0x40, 0x00,          /* length in bits */
```

0x18, 0x00, /* offset in bits */
 0x00, 0x00, 0x00, /* free for layer 2 */
 0x83, 0x08, 0x04, 0x04, 0x60, 0x02, 0x00, 0x81

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command (UL)
0x30	I, Nr = 1, N(S)=0, P=1
0x29	length = 10, M=0
0x06, 0x16, ...	CLASSM CHG
0x2B	fill bits

*/

BEGINARRAY (DL311_I_CMD_1_0_P_UL, 27)

0xB8, 0x00, 0x00, 0x00,
 0x01, 0x30, 0x29,
 0x06, 0x16, 0x03, 0x33, 0x19, 0x81, 0x20, 0x02, 0x60, 0x14,
 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response (DL)
0x21	RR, Nr = 1, F=0
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DL311_RR_RSP_1_DL, 27)

0xB8, 0x00, 0x00, 0x00,
 0x01, 0x21, 0x01,
 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response (DL)
0x39	REJ, Nr = 1, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DL311_REJ_RSP_1_F_DL, 27)

0xB8, 0x00, 0x00, 0x00,
 0x01, 0x39, 0x01,
 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command (UL)
0x22	I, Nr = 1, N(S)=1, P=0
0x21	length = 8, M=0
0x83, 0x08,	CALL CONF
...	

*/

0x2B	fill bits
------	-----------

*/

BEGINARRAY (DL311_I_CMD_1_1_UL, 27)
 0xB8, 0x00, 0x00, 0x00,
 0x01, 0x22, 0x21,
 0x83, 0x08, 0x04, 0x04, 0x60, 0x02, 0x00, 0x81, 0x2b, 0x2b,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response (DL)
0x31	RR, Nr = 1, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DL311_RR_RSP_1_F_DL, 27)
 0xB8, 0x00, 0x00, 0x00,
 0x01, 0x31, 0x01,
 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command (UL)
0x32	I, Nr = 1, N(S)=1, P=1
0x21	length = 8, M=0
0x83, 0x08,	CALL CONF
...	
0x2B	fill bits

*/

BEGINARRAY (DL311_I_CMD_1_1_P_UL, 27)
 0xB8, 0x00, 0x00, 0x00,
 0x01, 0x32, 0x21,
 0x83, 0x08, 0x04, 0x04, 0x60, 0x02, 0x00, 0x81, 0x2b, 0x2b,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command (DL)
0x42	I, Nr = 2, N(S)=1, P=0
0x53	length = 20, M=1
0x06, 0x2e, ...	ASSIGN CMD
0x2B	fill bits

*/

BEGINARRAY (DL311_I_CMD_2_1_DL, 27)
 0xB8, 0x00, 0x00, 0x00,
 0x03, 0x42, 0x53,
 0x06, 0x2e, 0x0d, 0x70, 0x1f, 0x05, 0x05, 0x10, 0x00, 0x00,
 0x80, 0x00, 0x00, 0x00, 0x00, 0x88, 0x00, 0x00, 0x00, 0x00

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command (UL)

*/

0x52	I, Nr = 2, N(S)=1, P=1
0x21	length = 8, M=0
0x83, 0x08,	CALL CONF
...	
0x2B	fill bits

*/

BEGINARRAY (DL311_I_CMD_2_1_P_UL, 27)

0xB8, 0x00, 0x00, 0x00,
 0x01, 0x52, 0x21,
 0x83, 0x08, 0x04, 0x04, 0x60, 0x02, 0x00, 0x81, 0x2b, 0x2b,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response (DL)
0x59	REJ, Nr = 2, F=1
0x01	length = 10, M=0
0x2B	fill bits

*/

BEGINARRAY (DL311_REJ_RSP_2_F_DL, 27)

0xB8, 0x00, 0x00, 0x00,
 0x01, 0x59, 0x01,
 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0x30, 0x00	Length in bits
0x18, 0x00	Offset in bits
0x03	Sapi 0
0x03	UI frame, P=0
0x19	content (6 Byte)
0x0F, ...	content

*/

BEGINARRAY (UI_SACCH_DL, 15)

0x48, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x03,
 0x03,
 0x19,
 0x0F, 0x0E, 0x0D, 0x0C, 0x0B, 0x0A

ENDARRAY

/*

0x30, 0x00	length in bits
0x18, 0x00	offset in bits
0x03	sapi 0
0x03	UI frame, P=0
0x19	content (6 Byte)
0x0F, ...	content

*/

BEGINARRAY (L3_MSG_UI_SACCH, 15)

0x30, 0x00,
 0x28, 0x00,
 0x00, 0x00,
 0x03,

0x03,
 0x19,
 0x0F, 0x0E, 0x0D, 0x0C, 0x0B, 0x0A

ENDARRAY

/*

0x30, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ..	free for layer 2 header
0x0A, ...	content

*/

BEGINARRAY (L3_MSG_L06_UL, 13)

0x30, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F

ENDARRAY

BEGINARRAY (L3_MSG_1, 20)

0x68, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x27, 0x00, 0x03, 0x33, 0x18,
 0x81, 0x05, 0xF4, 0x12, 0x34, 0x56,
 0x78

ENDARRAY

BEGINARRAY (L3_MSG_3, 13)

0x30, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x05, 0x14, 0x5F, 0x69, 0xF6, 0x3F

ENDARRAY

BEGINARRAY (L3_MSG_4, 10)

0x18, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x35, 0x01

ENDARRAY

BEGINARRAY (L3_MSG_5, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x32

ENDARRAY

BEGINARRAY (L3_MSG_6, 14)

0x38, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x03, 0x05, 0x04, 0x01, 0xA0, 0x34, 0x00

ENDARRAY

BEGINARRAY (L3_MSG_HO, 36)

0xE8, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2b, 0x18, 0x28, 0x48, 0xD0, 0x40,
 0x04, 0x6f, 0x05, 0x10, 0x00, 0x02, 0x08,
 0x00, 0x00, 0x00, 0x0f, 0x02, 0x08,

0x00, 0x00, 0x20, 0x60, 0xa2, 0x20, 0x00,
 0x63, 0x00

ENDARRAY

BEGINARRAY (L3_MSG_7, 15)

0x40, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x83, 0x48, 0x04, 0x04, 0x60, 0x02, 0x00, 0x81

ENDARRAY

BEGINARRAY (L3_MSG_8, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x83, 0x01

ENDARRAY

BEGINARRAY (L3_MSG_9, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x83, 0x47

ENDARRAY

BEGINARRAY (L3_MSG_10, 15)

0x40, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2e, 0x09, 0xa0, 0x1e, 0x09, 0x63, 0x01

ENDARRAY

BEGINARRAY (L3_MSG_11, 10)

0x18, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x29, 0x00

ENDARRAY

BEGINARRAY (L3_MSG_12, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x03, 0x0F

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, command
0x3F	SABM, P=1
0x01	no content
0x2B	fill bits

*/

BEGINARRAY (SABM_A_S3_SA_UL, 10)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x0D,
 0x3F,

0x01,
 0x2B
 ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x3F	SABM, P=1
0x19	6 Byte content
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRA Y (SABM_B_I06_UL, 14)
 0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x3F,
 0x19,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
 0x2B
 ENDARRA Y

BEGINARRA Y (SABM_MSG_1, 27)
 0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x3F, 0x35,
 0x06, 0x27, 0x00, 0x03, 0x33, 0x18,
 0x81, 0x05, 0xF4, 0x12, 0x34, 0x56,
 0x78, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B
 ENDARRA Y

BEGINARRA Y (SABM_MSG_11, 27)
 0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x3F, 0x01,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B
 ENDARRA Y

BEGINARRA Y (SABM_MSG_13, 27)
 0xA 8, 0x00,
 0x00, 0x00,
 0x00, 0x00,
 0x0F, 0x3F, 0x01,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B
 ENDARRA Y

BEGINARRA Y (I_MSG_3, 27)
 0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x20, 0x19,
 0x05, 0x14, 0x5F, 0x69, 0xF6, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,

0x2B, 0x2B
ENDARRAY
BEGINARRAY (I_MSG_3B, 27)
0xB8, 0x00,
0x00, 0x00,
0x01, 0x20, 0x19,
0x05, 0x14, 0x5F, 0x69, 0xF6, 0x3F,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_HO_1, 27)
0xB8, 0x00,
0x00, 0x00,
0x03, 0x22, 0x53,
0x06, 0x2b, 0x18, 0x28, 0x48, 0xd0,
0x40, 0x04, 0x6f, 0x05, 0x10, 0x00,
0x02, 0x08, 0x00, 0x00, 0x00, 0x0f,
0x02, 0x08

ENDARRAY

BEGINARRAY (I_HO_2, 27)
0xB8, 0x00,
0x00, 0x00,
0x03, 0x24, 0x25,
0x00, 0x00, 0x20, 0x60, 0xa2, 0x20,
0x00, 0x63, 0x00, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b

ENDARRAY

BEGINARRAY (I_MSG_4, 27)
0xB8, 0x00,
0x00, 0x00,
0x03, 0x22, 0x0d,
0x06, 0x35, 0x01, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_5, 27)
0xB8, 0x00,
0x00, 0x00,
0x01, 0x42, 0x09,
0x06, 0x32, 0x2b, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_5A, 27)
0xB8, 0x00,
0x00, 0x00,
0x01, 0x22, 0x09,
0x06, 0x32, 0x2b, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRA Y (I_MSG_5B, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x32, 0x09,
0x06, 0x32, 0x2b, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_5C, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x52, 0x09,
0x06, 0x32, 0x2b, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_5D, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x02, 0x09,
0x06, 0x32, 0x2b, 0x2b, 0x2b, 0x2b,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_6, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x44, 0x1d,
0x03, 0x05, 0x04, 0x01, 0xA0, 0x34,
0x00, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_6B, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x24, 0x1d,
0x03, 0x05, 0x04, 0x01, 0xA0, 0x34,
0x00, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_7, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x64, 0x21,
0x83, 0x48, 0x04, 0x04, 0x60, 0x02,
0x00, 0x81, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_MSG_8, 27)

0xB8, 0x00,

0x00, 0x00,
0x01, 0x66, 0x09,
0x83, 0x01, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_9, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x68, 0x09,
0x83, 0x47, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2B , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_10, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0xa6, 0x21,
0x06, 0x2e, 0x09, 0xa0, 0x1e, 0x09,
0x63, 0x01 , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_11, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x00, 0x0d,
0x06, 0x29, 0x00, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_12, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x20, 0x09,
0x03, 0x0F, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_12B, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x40, 0x09,
0x03, 0x0F, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b , 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_MSG_14, 27)

0xA8, 0x00,
0x00, 0x00,
0x00, 0x00,

0x0F, 0x00, 0x4b,
0x09, 0x01, 0xa9, 0x01, 0x00, 0x06,
0x91, 0x11, 0x11, 0x11, 0x11, 0x11,
0x00, 0x9E, 0x00, 0x0A, 0x91, 0x33
ENDARRAY

/*

0xA8, 0x00	Length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, response
0x73	UA, P=1
0x01	no content
0x2B	fill bits

*/

BEGINARRAY (UA_A_S3_SA_DL, 10)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x0D,
 0x73,
 0x01,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x73	UA frame, F = 1
0x19	6 byte content, M = 0
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (UA_B_I06_1_DL, 14)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x73,
 0x19,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
 0x2B

ENDARRAY

BEGINARRAY (UA_MSG_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x73, 0x35,
 0x06, 0x27, 0x00, 0x03, 0x33, 0x18,
 0x81, 0x05, 0xF4, 0x12, 0x34, 0x56,
 0x78, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

BEGINARRAY (UA_MSG_11, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x73, 0x01,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

BEGINARRAY (UA_MSG_13, 27)

0xA8, 0x00,
0x10, 0x00,
0x00, 0x00,
0x0F, 0x73, 0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B

ENDARRAY

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x73	UA frame, F = 1
0x19	6 byte content, M = 0
0x0F, ...	wrong content
0x2B	fill bits

*/

```
BEGINARRAY (UA_B_I06_2_DL, 14)
    0x17, 0x00,
    0x00, 0x00,
    0x01,
    0x73,
    0x19,
    0x0F, 0x0E, 0x0F, 0x0E, 0x0F, 0x0E, 0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x73	ua frame, P=1
0x01	no content
0x2B	fill bits

*/

```
BEGINARRAY (ERR_UN SOL_ UA_ DL, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x01,
    0x73,
    0x01,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x0F	dm frame, P=0
0x01	no content
0x2B	fill bits

*/

```
BEGINARRAY (ERR_UN SOL_ DM_ DL, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x01,
    0x0F,
    0x01,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x53	DISC, P=1
0x03	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (ERR_U_ PARAM_ DL, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x03,
    0x53,
    0x03,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x01	RR, P=0, N@ = 0
0x03	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (ERR_S_PARAM_DL, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x01,
    0x01,
    0x03,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x00	I, P=0, N@=0, N(S)=0
0x43	length = 32 Byte, M=1
0x2B	fill bits

*/

```
BEGINARRAY (ERR_I_M_BIT_DL, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x03,
    0x00,
    0x43,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x00	I, P=0, N@=0, N(S)=0
0x01	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (ERR_I_FR_LEN, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x03,
    0x00,
    0x01,
    0x2B
ENDARRAY
```

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x37	frame not implemented
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (ERR_I_FR_NOT_IMP, 8)
    0x17, 0x00,
    0x00, 0x00,
    0x03,
    0x37,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x3F	SABM; P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (SABM_A_UL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x3F,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x73	UA; P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (UA_A_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x73,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x00	I, N@ = 0, N(S)=0, P=0
0x19	length = 6, M=0
0x0A, ...	content
0x2B	fill bits

*/

```
BEGINARRAY (L_I06_UL, 14)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x00,
    0x19,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x21	RR, N@ = 1, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (RR_A_NRI_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x21,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0D	sapi 3, command
0x3F	SABM, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (SABM_A_S3_UL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x0D,
    0x3F,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0D	sapi 3, command
0x73	UA, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (UA_A_S3_SD_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x0D,
    0x73,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0D	sapi 3, command
0x73	UA, P=1
0x11	length = 4, M=0
0x0A, ...	content
0x2B	fill bits

*/

```
BEGINARRAY (UA_B_S3_I04_DL, 12)
    0xB8, 0x00,
    0x00, 0x00,
    0x0D,
    0x73,
    0x11,
    0x0A, 0x0B, 0x0C, 0x0D,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0D	sapi 3, command
0x1F	DM, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (DM_A_S3_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x0D,
    0x1F,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, command to MS
0x3F	SABM, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (SABM_A_S3_SD_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x0F,
    0x3F,
    0x01,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, command to MS
0x3F	SABM, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (SABM_A_S3_SA_DL, 10)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0F,
    0x3F,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, response to BS
0x73	UA, F=1
0x01	length = 0, M=0
0x2B	fill bits

*/

```
BEGINARRAY (UA_A_S3_UL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x0F,
    0x73,
    0x01,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command

0x53	DISC, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DISC_A_UL, 8)

0xB8, 0x00,

0x00, 0x00,

0x01,

0x53,

0x01,

0x2B

ENDARRAY

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x53	DISC, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DISC_A_DL, 8)

0x17, 0x00,

0x00, 0x00,

0x03,

0x53,

0x01,

0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response to BS
0x1F	DM, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DM_A_UL, 8)

0xB8, 0x00,

0x00, 0x00,

0x03,

0x1F,

0x01,

0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response to BS
0x73	UA, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (UA_A_UL, 8)

0xB8, 0x00,

0x00, 0x00,

0x03,

0x73,

0x01,

0x2B

ENDARRAY

/*

0x17, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x3F	SABM, P=1
0x19	length = 6, M=0
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (SABM_B_I06_DL, 14)

0x17, 0x00,

0x00, 0x00,

0x03,

0x3F,

0x19,

0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,

0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, response to BS
0x73	UA, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (UA_A_S3_SA_UL, 10)

0xA8, 0x00,

0x10, 0x00,

0x00, 0x00,

0x0F,

0x73,

0x01,

0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, command to MS
0x53	DISC, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DISC_A_S3_UL, 10)

0xA8, 0x00,

0x10, 0x00,

0x00, 0x00,

0x0F,

0x53,

0x01,

0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, response to BS
0x1F	DM, P=1
0x01	length = 0, M=0
0x2B	fill bits

*/

BEGINARRAY (DM_A_S3_SA_DL, 10)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x0F,
 0x1F,
 0x01,
 0x2B

ENDARRAY

/*

0x80, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	free for layer 2
0x0A, ...	content (16 Byte)

*/

BEGINARRAY (L3_MSG_L16_1_UL, 23)

0x80, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D

ENDARRAY

/*

0xC0, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	free for layer 2
0x0A, ...	content (24 Byte)

*/

BEGINARRAY (L3_MSG_L24_UL, 31)

0xC0, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F

ENDARRAY

/*

0x80, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	dummies for l2 header
0x0A, ...	content (16 Byte)

*/

```
BEGINARRAY (L3_MSG_L16_2_UL, 23)
    0x80, 0x00,
    0x18, 0x00,
    0x00, 0x00, 0x00,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D
ENDARRAY
```

/*

0xC0, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	dummies for l2 header
0x0A, ...	content (24 Byte)

*/

```
BEGINARRAY (L3_MSG_L24_DL, 31)
    0xC0, 0x00,
    0x18, 0x00,
    0x00, 0x00, 0x00,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
    0x2E, 0x2F, 0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x00	I, P=0, N@=0, N(S)=0
0x41	length = 16, M=0
0x0A, ...	content
0x2B	fill bits

*/

```
BEGINARRAY (L_I16_UL, 24)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x00,
    0x41,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
    0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x00	I, P=0, N@=0, N(S)=0
0x53	length = 20, M=1
0x0A, ...	content

*/

```
BEGINARRAY (I_I20_UL, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x00,
    0x53,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
    0x2E, 0x2F, 0x3A, 0x3B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x02	I, P=0, N@=0, N(S)=1
0x11	length = 4, M=0
0x3C, ...	content

*/

```
BEGINARRAY (I_I04_UL, 12)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x02,
    0x11,
    0x3C, 0x3D, 0x3E, 0x3F,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, command to MS
0x02	I, P=0, N@=0, N(S)=1
0x19	length = 6, M=0
0x3A, ...	content

*/

```
BEGINARRAY (I_I06_S3_DL, 16)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0F,
    0x02,
    0x19,
    0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, command to BS
0x00	I, P=0, N@=0, N(S)=0
0x41	length = 16, M=0
0x0A, ...	content

*/

```
BEGINARRAY (I_I16_S3_UL, 26)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0D,
    0x00,
    0x41,
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, command to BS
0x00	I, P=0, N@=0, N(S)=0
0x4B	length = 18, M=1
0x0A, ...	content

*/

```
BEGINARRAY (I_I18_S3_UL, 27)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0D,
    0x00,
    0x4B,
```

0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
0x2E, 0x2F

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, command to BS
0x02	I, P=0, N@=0, N(S)=1
0x19	length = 6, M=0
0x3A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I06_S3_UL, 16)

0xA8, 0x00,
0x10, 0x00,
0x00, 0x00,
0x0D,
0x02,
0x19,
0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F,
0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x00	I, P=0, N@=0, N(S)=0
0x41	length = 16, M=0
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I16_DL, 24)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x41,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

BEGINARRAY (I_MSG_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x4D,
 0x05, 0x12, 0x00, 0x01, 0x23, 0x45, 0x67, 0x89,
 0xAB, 0xCD, 0xEF, 0x10, 0x32, 0x54, 0x76, 0x98,
 0xBA, 0xDC, 0xFE, 0x2B

ENDARRAY

BEGINARRAY (L3_MSG_2, 26)

0x98, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x05, 0x12, 0x00, 0x01, 0x23, 0x45, 0x67, 0x89,
 0xab, 0xcd, 0xef, 0x10, 0x32, 0x54, 0x76, 0x98,
 0xba, 0xdc, 0xfe

ENDARRAY

BEGINARRAY (RR_MSG_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x21,
 0x01,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x21,
 0x01,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,

0x2b, 0x2b, 0x2b, 0x2B
ENDARRAY
BEGINARRAY (RR_MSG_5, 27)
0xB8, 0x00,
0x00, 0x00,
0x01,
0x41,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_12B, 27)
0xB8, 0x00,
0x00, 0x00,
0x03,
0x21,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_5C, 27)
0xB8, 0x00,
0x00, 0x00,
0x01,
0x51,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_5D, 27)
0xB8, 0x00,
0x00, 0x00,
0x01,
0x41,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_6, 27)
0xB8, 0x00,
0x00, 0x00,
0x03,
0x61,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_7, 27)
0xB8, 0x00,
0x00, 0x00,
0x01,

0x61,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_8, 27)

0xB8, 0x00,
0x00, 0x00,
0x01,
0x81,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_9, 27)

0xB8, 0x00,
0x00, 0x00,
0x01,
0xA1,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_11, 27)

0xB8, 0x00,
0x00, 0x00,
0x01,
0x21,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_12, 27)

0xB8, 0x00,
0x00, 0x00,
0x03,
0x21,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2B

ENDARRAY

BEGINARRAY (RR_MSG_14, 27)

0xA8, 0x00,
0x10, 0x00,
0x00, 0x00,
0x0f,
0x21,
0x01,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
0x2b, 0x2b

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x00	I, P=1, N@=0, N(S)=0
0x41	Length = 16, M=0
0x0A, ...	Content
0x2B	fill bits

*/

BEGINARRAY (I_HO_RETRANSMIT, 24)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x10,
 0x41,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x00	I, P=0, N@=0, N(S)=0
0x53	length = 20, M=1
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I20_DL, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, command to MS
0x00	I, P=0, N(R)=0, N(S)=0
0x53	length = 20, M=1
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I20_DL3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x0F,
 0x00,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,

0x2E, 0x2F, 0x3A, 0x3B
 ENDA RRA Y

/*

0xA8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, command to MS
0x00	I, P=0, N@=0, N(S)=0
0x4B	length = 18, M=1
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRA Y (I_I18_DL3, 25)

0xA 8, 0x00,
 0x00, 0x00,
 0x0F,
 0x00,
 0x4B,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F

ENDA RRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x02	I, P=0, N@=0, N(S)=1
0x11	length = 4, M=0
0x3C, ...	content
0x2B	fill bits

*/

BEGINARRA Y (I_I04_NS1_DL, 12)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x02,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B

ENDA RRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, command to MS
0x02	I, P=0, N(R)=0, N(S)=1
0x11	length = 4, M=0
0x3C, ...	content
0x2B	fill bits

*/

BEGINARRA Y (I_I04_NS1_DL3, 12)

0xB8, 0x00,
 0x00, 0x00,
 0x0F,
 0x02,
 0x11,

0x3C, 0x3D, 0x3E, 0x3F,
 0x2B
 ENDARRAY

/*

0xA8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, command to MS
0x02	I, P=0, N(R)=0, N(S)=1
0x19	length = 6, M=0
0x3A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I06_NS1_DL3, 14)

0xA8, 0x00,
 0x00, 0x00,
 0x0F,
 0x02,
 0x19, 0x3A, 0x3B,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, command to MS
0x00	I, P=0, N@=0, N(S)=0
0x41	length = 16, M=0
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_I16_S3_DL, 26)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x0F,
 0x00,
 0x41,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, command to MS
0x00	I, P=0, N@=0, N(S)=0
0x4B	length = 18, M=1
0x0A, ...	content

*/

BEGINARRAY (I_I18_S3_DL, 27)

0xA8, 0x00,
 0x10, 0x00,

0x00, 0x00,
0x0F,
0x00,
0x4B,
0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
0x2E, 0x2F
ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, response to BS
0x21	RR P=0, N@=1
0x01	length = 0, M=1
0x2B	fill bits

*/

BEGINARRAY (RR_A_S3_NR1_DL, 10)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x0D,
 0x21,
 0x01,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response to BS
0x21	RR P=0, N@=1
0x01	length = 0, M=1
0x2B	fill bits

*/

BEGINARRAY (RR_A_UL, 8)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x21,
 0x01,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, response to BS
0x21	RR P=0, N@=1
0x01	length = 0, M=1
0x2B	fill bits

*/

BEGINARRAY (RR_A_UL3, 8)

0xB8, 0x00,
 0x00, 0x00,
 0x0F,
 0x21,
 0x01,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response to BS

0x41	RR P=0, N@=2
0x01	length = 0, M=1
0x2B	fill bits

*/

BEGINARRAY (RR_A_NR2_UL, 8)

0xB8, 0x00,

0x00, 0x00,

0x03,

0x41,

0x01,

0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x0F	sapi 3, response to BS
0x41	RR P=0, N@=2
0x01	length = 0, M=1
0x2B	fill bits

*/

BEGINARRAY (RR_A_NR2_UL3, 8)

0xB8, 0x00,

0x00, 0x00,

0x0F,

0x41,

0x01,

0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x41	RR P=0, N@=2
0x01	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (RR_A_NR2_DL, 8)
    0xB8, 0x00,
    0x00, 0x00,
    0x01,
    0x41,
    0x01,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0F	sapi 3, response to BS
0x21	RR P=0, N@=1
0x01	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (RR_A_S3_UL, 10)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0F,
    0x21,
    0x01,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x0D	sapi 3, response to MS
0x41	RR P=0, N@=2
0x01	length = 0, M=1
0x2B	fill bits

*/

```
BEGINARRAY (RR_A_S3_NR2_DL, 10)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x0D,
    0x41,
    0x01,
    0x2B
ENDARRAY
```

/*

0xA8, 0x00	length in bits
------------	----------------

0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x01	address
0x03	control
0x01	length
0x2B	fill bits

*/

BEGINARRAY (EMPTY_FRAME_SACCH, 10)

0xA8, 0x00,

0x10, 0x00,

0x00, 0x00,

0x01,

0x03,

0x01,

0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	address
0x03	control
0x01	length
0x2B	fill bits

*/

BEGINARRAY (EMPTY_FRAME_DCCH, 8)

0xB8, 0x00,

0x00, 0x00,

0x01,

0x03,

0x01,

0x2B

ENDARRAY

/*

0x38, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x00	l frame, P=0
0x1D	content 7 Bytes
0x03, 0x05 ...	Setup message
0x2b	fill bits

*/

BEGINARRA Y (L_SETUP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x1D,
 0x03, 0x05, 0x04, 0x01,
 0xA0, 0x34, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0x38, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x03, 0x05 ...	Setup message

*/

BEGINARRA Y (L3_SETUP, 14)

0x38, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x03, 0x05, 0x04, 0x01,
 0xA0, 0x34, 0x3F

ENDARRA Y

BEGINARRA Y (L3_SETUP_MSG, 23)

0x80, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x03, 0x45, 0x04, 0x04,
 0x60, 0x02, 0x00, 0x81,
 0x5E, 0x03, 0x81, 0x21,
 0x43, 0x15, 0x01, 0x01

ENDARRA Y

BEGINARRA Y (I_SETUP_MSG_P0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x22, 0x41,
 0x03, 0x45, 0x04, 0x04,
 0x60, 0x02, 0x00, 0x81,
 0x5E, 0x03, 0x81, 0x21,
 0x43, 0x15, 0x01, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_SETUP_MSG_P1, 27)

0xB8, 0x00,
0x00, 0x00,
0x01, 0x52, 0x41,
0x03, 0x45, 0x04, 0x04,
0x60, 0x02, 0x00, 0x81,
0x5E, 0x03, 0x81, 0x21,
0x43, 0x15, 0x01, 0x01,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (I_HO_CMD_P0_S1, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x22, 0x53,
0x06, 0x2B, 0x98, 0xFC,
0x08, 0xB0, 0x12, 0x64,
0x07, 0x02, 0x8D, 0x7A,
0x58, 0x0B, 0x69, 0xDA,
0x08, 0xDD, 0x90, 0x63

ENDARRA Y

BEGINARRA Y (I_HO_CMD_P1_S1, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x32, 0x53,
0x06, 0x2B, 0x98, 0xFC,
0x08, 0xB0, 0x12, 0x64,
0x07, 0x02, 0x8D, 0x7A,
0x58, 0x0B, 0x69, 0xDA,
0x08, 0xDD, 0x90, 0x63

ENDARRA Y

BEGINARRA Y (I_HO_CMD_P0_S2, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x24, 0x05,
0x01, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR_HO_CMD, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x51, 0x01,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (REJ_HO_CMD, 27)

0xB8, 0x00,
0x00, 0x00,
0x03, 0x59, 0x01,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,

0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B
ENDARRAY

BEGINARRAY (L3_HO_CMD, 28)
0xA8, 0x00,
0x18, 0x00,
0x00, 0x00, 0x00,
0x06, 0x2B, 0x98, 0xFC,
0x08, 0xB0, 0x12, 0x64,
0x07, 0x02, 0x8D, 0x7A,
0x58, 0x0B, 0x69, 0xDA,
0x08, 0xDD, 0x90, 0x63,
0x01

ENDARRAY

BEGINARRAY (I_CIPH_REQ, 27)
0xB8, 0x00,
0x00, 0x00,
0x03, 0x00, 0x0D,
0x06, 0x35, 0x00, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

BEGINARRAY (RR_CIPH_REQ, 27)
0xB8, 0x00,
0x00, 0x00,
0x03, 0x21, 0x01,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

BEGINARRAY (L3_CIPH_REQ, 10)
0x18, 0x00,
0x18, 0x00,
0x00, 0x00, 0x00,
0x06, 0x35, 0x00

ENDARRAY

BEGINARRAY (L3_CIPH_RSP, 9)
0x10, 0x00,
0x18, 0x00,
0x00, 0x00, 0x00,
0x06, 0x32

ENDARRAY

BEGINARRAY (I_CIPH_RSP, 27)
0xB8, 0x00,
0x00, 0x00,
0x01, 0x20, 0x09,
0x06, 0x32, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

```
BEGINARRAY (RR_CIPH_RSP, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x01, 0x21, 0x01,
    0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B
ENDARRAY
```

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response
0x21	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINRRAY (RR_SETUP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x21,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDRRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x20	I frame, P=0
0x09	content 2 Bytes
0x83, 0x08	Call Confirm message
0x2b	fill bits

*/

BEGINRRAY (I_CALL_CONFIRM, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x20,
 0x09,
 0x83, 0x08,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDRRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x00	l frame, P=0
0x09	content 2 Bytes
0x83, 0x08	Call Confirm message
0x2b	fill bits

*/

BEGINARRAY (I_CALL_CONFIRM_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x00,
 0x09,
 0x83, 0x08,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x10	l frame, P=1
0x09	content 2 Bytes
0x83, 0x08	Call Confirm message
0x2b	fill bits

*/

BEGINARRAY (I_CALL_CONFIRM_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x10,
 0x09,
 0x83, 0x08,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x31	RR frame, P=1
0x01	no content
0x2b	fill bits

*/

BEGINARRA Y (RR_CALL_CONFIRM_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x31,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x21	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINARRA Y (RR_CALL_CONFIRM_4, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x21,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0x10, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x83, 0x08	Call Confirm message

*/

BEGINARRA Y (L3_CALL_CONFIRM, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x83, 0x08

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x22	I frame, P=0
0x21	content 8 Bytes
0x06, 0x2E,...	Assignment Command message
0x2b	fill bits

*/

BEGINARRAY (L_ASSIGN_CMD, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x22,
 0x21,
 0x06, 0x2E,
 0x0C, 0xA0, 0x1E, 0x0C,
 0x63, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0x40, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x06, 0x2E	Assignment Command message

*/

BEGINARRAY (L3_ASSIGN_CMD, 15)

0x40, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2E,
 0x0C, 0xA0, 0x1E, 0x0C,
 0x63, 0x01

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response
0x41	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINRRAY (RR_ASSIGN_CMD, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x41,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDRRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x00	I frame, P=0
0x0D	content 3 Bytes
0x06, 0x2F, ...	Assignment Failure message
0x2b	fill bits

*/

BEGINRRAY (I_ASSIGN_FAILURE, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x00,
 0x0D,
 0x06, 0x2F,
 0x00,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B

ENDRRAY

/*

0x18, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x06, 0x2F	Assignment Command message

*/

BEGINRRAY (ASSIGN_FAILURE, 10)

0x18, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2F,

0x00
ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x21	RR frame, P=0,N(R)=1
0x01	no content
0x2b	fill bits

*/

BEGINARRAY (RR_ASSIGN_FAILURE, 27)

0xB8, 0x00,
0x00, 0x00,
0x01,
0x21,
0x01,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x02	l frame, P=0
0x09	content 2 Bytes
0x83, 0x41	Alerting message
0x2b	fill bits

*/

BEGINARRA Y (I_ALERT, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x02,
 0x09,
 0x83, 0x41,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x42	l frame, P=0
0x09	content 2 Bytes
0x83, 0x41	Alerting message
0x2b	fill bits

*/

BEGINARRA Y (I_ALERT_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x42,
 0x09,
 0x83, 0x41,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRA Y

/*

0x10, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x83, 0x41	Alerting message

*/

BEGINARRAY (L3_ALERT, 9)

0x10, 0x00,
0x18, 0x00,
0x00, 0x00, 0x00,
0x83, 0x41

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x41	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINARRAY (RR_ALERT, 27)

0xB8, 0x00,
0x00, 0x00,
0x01,
0x41,
0x01,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x04	l frame, P=0
0x09	content 2 Bytes
0x83, 0x07	Connect message
0x2b	fill bits

*/

BEGINARRAY (L_CONNECT, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x04,
 0x09,
 0x83, 0x07,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

/*

0x10, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x83, 0x07	Connect message

*/

BEGINARRAY (L3_CONNECT, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x83, 0x07

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x61	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINARRA Y (RR_CONNECT, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x61,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x00	I frame, P=0
0x0D	content 3 Bytes
0x06, 0x29, ...	Assignment Complete message
0x2b	fill bits

*/

BEGINARRA Y (I_ASSIGN_CMP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x00,
 0x0D,
 0x06, 0x29, 0x00,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B

ENDARRA Y

/*

0x18, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x06, 0x29	Assignment Complete message

*/

BEGINARRAY (L3_ASSIGN_CMP, 10)

0x18, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x29, 0x00

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response
0x21	RR frame, P=0
0x01	no content
0x2b	fill bits

*/

BEGINARRAY (RR_ASSIGN_CMP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x21,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x03	UI frame, P=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (UI_FRAME_DOWNLINK, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x03,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
------------	----------------

0x00, 0x00	offset in bits
0x01	sapi 0, command
0x03	UI frame, P=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (UI_FRAME_UPLINK, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x03,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x01	sapi 0, command
0x03	UI frame, P=0
0x49	content 18 Bytes
0x2b	fill bits

*/

BEGINARRAY (MEAS_REPORT_1, 25)

0x90, 0x00,
 0x18, 0x00,
 0x00,
 0x00,
 0x00,
 0x06, 0x15, 0x01, 0x02,
 0x03, 0x04, 0x05, 0x06,
 0x07, 0x08, 0x09, 0x0A,
 0x0B, 0x0C, 0x0D, 0x0E,
 0x0F, 0x10

ENDARRAY

BEGINARRAY (MEAS_REPORT_1_L1, 27)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x01,
 0x03,
 0x49,
 0x06, 0x15, 0x01, 0x02,
 0x03, 0x04, 0x05, 0x06,
 0x07, 0x08, 0x09, 0x0A,
 0x0B, 0x0C, 0x0D, 0x0E,
 0x0F, 0x10

ENDARRAY

BEGINARRAY (MEAS_REPORT_0_L1, 27)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x01,
 0x03,

```
    0x49,
    0x06, 0x15, 0x00, 0x40,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00
ENDARRAY
BEGINARRAY (MEAS_REPORT_2, 25)
    0x90, 0x00,
    0x18, 0x00,
    0x00,
    0x00,
    0x00,
    0x06, 0x15, 0x10, 0x20,
    0x30, 0x40, 0x50, 0x60,
    0x70, 0x80, 0x90, 0xA0,
    0xB0, 0xC0, 0xD0, 0xE0,
    0xF0, 0x10
ENDARRAY
BEGINARRAY (MEAS_REPORT_2_L1, 27)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x01,
    0x03,
    0x49,
    0x06, 0x15, 0x10, 0x20,
    0x30, 0x40, 0x50, 0x60,
    0x70, 0x80, 0x90, 0xA0,
    0xB0, 0xC0, 0xD0, 0xE0,
    0xF0, 0x10
ENDARRAY
BEGINARRAY (MEAS_REPORT_INV_L1, 27)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x01,
    0x03,
    0x49,
    0x06, 0x15, 0x00, 0x40,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00
ENDARRAY
BEGINARRAY (MEAS_REPORT_NO_NCELL, 27)
    0xA8, 0x00,
    0x10, 0x00,
    0x00, 0x00,
    0x01,
    0x03,
    0x49,
    0x06, 0x15, 0x1E, 0x1E,
    0x7E, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,
    0x00, 0x00
ENDARRAY
/*
```

0xA8, 0x00	length in bits
0x10, 0x00	offset in bits
0x00, 0x00	layer 1 header
0x01	sapi 0, command
0x03	UI frame, P=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (UI_FRAME_UP_SACCH, 27)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x01,
 0x03,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B

ENDARRAY

BEGINARRAY (UI_FRAME_UP_SACCH_MEAS, 27)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x01,
 0x03,
 0x49,
 0x06, 0x15, 0x00, 0x40,
 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00

ENDARRAY

BEGINARRAY (UI_FRAME_UP_SACCH_M2, 27)

0xA8, 0x00,
 0x10, 0x00,
 0x00, 0x00,
 0x01,
 0x03,
 0x49,
 0x06, 0x15, 0x1E, 0x1E,
 0x7E, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	Command MS to BS
0x3F	SABM, P=1
0x3D	L=15, M=0
0x06, 0x27 ...	Paging Response message
0x2B, ..	fill bits

*/

BEGINARRA Y (SABM_PAGING_RESP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x3F, 0x3D,
 0x06, 0x27,
 0x00, 0x03,
 0x29, 0x78,
 0x00, 0x07,
 0x09, 0x10,
 0x10, 0x74,
 0x11, 0x94,
 0x21,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	Command MS to BS
0x73	UA, F=1
0x3D	L=15, M=0
0x06, 0x27 ...	Paging Response message
0x2B, ..	fill bits

*/

BEGINARRA Y (UA_PAGING_RESP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x73, 0x3D,
 0x06, 0x27,
 0x00, 0x03,
 0x29, 0x78,
 0x00, 0x07,
 0x09, 0x10,
 0x10, 0x74,
 0x11, 0x94,
 0x21,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	Command MS to BS
0x73	UA, F=1
0x01	L=0, M=0
0x2B, ..	fill bits

*/

BEGINARRA Y (UA_NO_PAGING_RESP, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x73, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0x78, 0x00	length in bits
0x18, 0x00	offset in bits
0x, ...	
0x06, 0x27 ...	Paging Response message

*/

BEGINARRA Y (L3_PAGING_RESP, 22)

0x78, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x27,
 0x00, 0x03,
 0x29, 0x78,
 0x00, 0x07,
 0x09, 0x10,
 0x10, 0x74,
 0x11, 0x94,
 0x21

ENDARRA Y

/*

0xC0, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	dummies for l2 header
0x0A, ...	content (24 Byte)

*/

BEGINARRA Y (LONG_MESSAGE, 31)

0xC0, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F

ENDARRA Y

BEGINARRA Y (I00, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x0F, 0x00, 0x53,

0x09,0x01,0xa6,0x01,0x00,
0x02, 0x91, 0x11, 0x00, 0x9F,
0x00, 0x0c, 0x91, 0x94, 0x98,
0x14,0x92,0x63,0x10,0x00

ENDARRAY

BEGINARRAY (I01, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x02, 0x53,
0x00,0x69,0x20,0x61,0x90,
0x43, 0x72, 0x00, 0xa0, 0x53,
0xE4, 0x53, 0x4A, 0x05, 0x35,
0x8B,0xD3,0x69,0xF0,0x58

ENDARRAY

BEGINARRAY (I02, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x04, 0x53,
0x04,0x25,0x9d,0xa0,0x21,
0x94, 0x18, 0xa4, 0x06, 0x41,
0x53, 0x6a, 0x11, 0x0a, 0x0a,
0xea,0x40,0x80,0x80,0x60

ENDARRAY

BEGINARRAY (I03, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x06, 0x53,
0x40,0x28,0x18,0x0E,0x88,
0x84, 0x62, 0xc1, 0x50, 0x34,
0x1c, 0x0f, 0x48, 0x44, 0x42,
0xA1,0x54,0x2C,0x17,0x4C

ENDARRAY

BEGINARRAY (I04, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x08, 0x53,
0x46,0xb3,0xe1,0x74,0x3c,
0x9f, 0x90, 0x68, 0x44, 0x2a,
0x99, 0x4e, 0xa8, 0x94, 0x6a,
0xc5,0x6a,0xb9,0x5e,0xb0

ENDARRAY

BEGINARRAY (I05, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x0A, 0x53,
0x98,0x6c,0x46,0xab,0xd9,
0x6e, 0xb8, 0x9c, 0x6e, 0xc7,
0xeb, 0xf9, 0x7e, 0xc0, 0xa0,
0x70,0x48,0x2c,0x1a,0x8f

ENDARRAY

BEGINARRAY (I06, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x0C, 0x53,
0xC8,0xa4,0x72,0xc9,0x6c,
0x3a, 0x9f, 0xd0, 0xa8, 0x74,

0x4a, 0xad, 0x5a, 0xaf, 0xd8,
0xac, 0x76, 0xcb, 0xed, 0x7a

ENDARRA Y

BEGINARRA Y (I07, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x0E, 0x53,
0xBF, 0xE0, 0xB0, 0x78, 0x4C,
0x2E, 0x9b, 0xCF, 0xE8, 0xB4,
0x7a, 0xCd, 0x6e, 0xBB, 0xDF,
0xF0, 0xB8, 0x7C, 0x4E, 0xAf

ENDARRA Y

BEGINARRA Y (I00_FINAL, 27)

0xB8, 0x00,
0x00, 0x00,
0x0F, 0x00, 0x25,
0xDB, 0xEF, 0xF8, 0xBC, 0x7E,
0xCF, 0xEF, 0xFB, 0xFF, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (SMS_MSG, 176)

0x48, 0x05,
0x18, 0x00,
0x00, 0x00, 0x00,
0x09, 0x01, 0xa6, 0x01, 0x00,
0x02, 0x91, 0x11, 0x00, 0x9F,
0x00, 0x0c, 0x91, 0x94, 0x98,
0x14, 0x92, 0x63, 0x10, 0x00,
0x00, 0x69, 0x20, 0x61, 0x90,
0x43, 0x72, 0x00, 0xa0, 0x53,
0xE4, 0x53, 0x4A, 0x05, 0x35,
0x8B, 0xD3, 0x69, 0xF0, 0x58,
0x04, 0x25, 0x9d, 0xa0, 0x21,
0x94, 0x18, 0xa4, 0x06, 0x41,
0x53, 0x6a, 0x11, 0x0a, 0x0a,
0xea, 0x40, 0x80, 0x80, 0x60,
0x40, 0x28, 0x18, 0x0E, 0x88,
0x84, 0x62, 0xc1, 0x50, 0x34,
0x1c, 0x0f, 0x48, 0x44, 0x42,
0xA1, 0x54, 0x2C, 0x17, 0x4C,
0x46, 0xb3, 0xe1, 0x74, 0x3c,
0x9f, 0x90, 0x68, 0x44, 0x2a,
0x99, 0x4e, 0xa8, 0x94, 0x6a,
0xc5, 0x6a, 0xb9, 0x5e, 0xb0,
0x98, 0x6c, 0x46, 0xab, 0xd9,
0x6e, 0xb8, 0x9c, 0x6e, 0xc7,
0xeb, 0xf9, 0x7e, 0xc0, 0xa0,
0x70, 0x48, 0x2c, 0x1a, 0x8f,
0xC8, 0xa4, 0x72, 0xc9, 0x6c,
0x3a, 0x9f, 0xd0, 0xa8, 0x74,
0x4a, 0xad, 0x5a, 0xaf, 0xd8,
0xac, 0x76, 0xcb, 0xed, 0x7a,
0xBF, 0xE0, 0xB0, 0x78, 0x4C,
0x2E, 0x9b, 0xCF, 0xE8, 0xB4,
0x7a, 0xCd, 0x6e, 0xBB, 0xDF,
0xF0, 0xB8, 0x7C, 0x4E, 0xAf,
0xDB, 0xEF, 0xF8, 0xBC, 0x7E,

0xCF, 0xEF, 0xFB, 0xFF
ENDARRA Y
BEGINARRA Y (RR00, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0x01, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR01, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0x21, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR02, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0x41, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR03, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0x61, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR04, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0x81, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (RR05, 27)
0xB8, 0x00,
0x00, 0x00,
0x0F, 0xA1, 0x01,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

```
BEGINARRA Y (RR06, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x0F, 0xC1, 0x01,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B
```

ENDARRA Y

```
BEGINARRA Y (RR07, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x0F, 0xE1, 0x01,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B,
    0x2B, 0x2B, 0x2B, 0x2B,0x2B
```

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	Command BS to MS
0x10	I, , P=1, N(R)=0, N(S)=0
0x11	L=4, M=0
0x03, ..	content
0x2B, ..	fill bits

*/

```
BEGINARRA Y (I_FRAME_1, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x03, 0x10, 0x11,
    0x03, 0x04, 0x06, 0x07, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	Command BS to MS
0x11	RR , P=1, N(R)=0
0x01	L=0, M=0
0x2B, ..	fill bits

*/

```
BEGINARRA Y (RR_FRAME_1, 27)
    0xB8, 0x00,
    0x00, 0x00,
    0x03, 0x11, 0x01,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	Command BS to MS
0x19	REJ, P=1, N(R)=0
0x01	L=0, M=0
0x2B, ..	fill bits

*/

BEGINARRAY (REJ_FRAME_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03, 0x19, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	Response BS to MS
0x1F	DM, F=1
0x01	L=0, M=0
0x2B, ..	fill bits

*/

BEGINARRAY (DM_FRAME, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01, 0x1F, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x00	l frame, P=0, N(S)=0, N(R)=0
0x19	content 6 Bytes
0x06, 0x2E, ...	Assignment Command message
0x2b	fill bits

*/

BEGINARRAY (I_ASSIGN_CMD_SDCCH, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x19,
 0x06, 0x2E,
 0x20, 0xA0, 0x10, 0x0C,
 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0x30, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x06, 0x2E	Assignment Command message

*/

BEGINARRAY (ASSIGN_CMD_SDCCH, 13)

0x30, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2E,
 0x20, 0xA0, 0x10, 0x0C

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x20	l frame, P=0, N(S)=0, N(R)=1
0x21	content 8 Bytes
0x06, 0x2E, ...	Assignment Command message
0x2b	fill bits

*/

BEGINARRA Y (I_ASSIGN_CMD_FACCH, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x20,
 0x21,
 0x06, 0x2E,
 0x08, 0xA0, 0x10, 0x0C,
 0x63, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command
0x22	l frame, P=0, N(S)=1, N(R)=1
0x21	content 8 Bytes
0x06, 0x2E, ...	Assignment Command message
0x2b	fill bits

*/

BEGINARRA Y (I_ASSIGN_CMD_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x22,
 0x21,
 0x06, 0x2E,
 0x08, 0xA0, 0x10, 0x0C,
 0x63, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0x40, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x06, 0x2E	Assignment Command message

*/

BEGINARRA Y (ASSIGN_CMD_FACCH, 15)

0x40, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x06, 0x2E,

0x08, 0xA0, 0x10, 0x0C, 0x63, 0x01
ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x00	l frame, P=0,N(S)=0, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x00,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x10	l frame, P=1,N(S)=0, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_0_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x10,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS (invalid)
0x10	I frame, P=1, N(S)=0, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_0_C, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x10,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	Length in bits
0x00, 0x00	Offset in bits
0x03	Sapi 0, command to MS
0x80	I frame, P=0, N(S)=0, N(R)=1
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x20,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (BS_I_0_7, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0xE0,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (BS_I_0_1_P1, 27)

0xB8, 0x00,
0x00, 0x00,
0x03,
0x30,
0x09,
0x05, 0x18,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B

ENDARRA Y

BEGINARRA Y (BS_I_0_7_P1, 27)

0xB8, 0x00,
0x00, 0x00,
0x03,
0xF0,
0x09,
0x05, 0x18,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x80	l frame, P=0,N(S)=0, N(R)=1
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_1_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x30,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x80	l frame, P=0,N(S)=0, N(R)=4
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_0_4, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x80,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x02	l frame, P=0,N(S)=1, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_1_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x02,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x22	l frame, P=0,N(S)=1, N(R)=1
0x53	content 20 Bytes, M=1
0x0A,....	message

*/

BEGINARRA Y (BS_I_1_1_M1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x22,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x24	I frame, P=0, N(S)=2, N(R)=10
0x11	content 4 Bytes, M=0
0x3C,....	message

*/

BEGINARRA Y (BS_I_2_1_M2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x24,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x04	I frame, P=0, N(S)=2, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_2_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x04,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x06	l frame, P=0,N(S)=3, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_3_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x06,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x08	l frame, P=0,N(S)=4, N(R)=0
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_4_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x08,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x2A	l frame, P=0,N(S)=5, N(R)=1
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_5_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x2A,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x4C	l frame, P=0,N(S)=6, N(R)=2
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRA Y (BS_I_6_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x4C,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x6E	I frame, P=0, N(S)=7, N(R)=3
0x09	content 2 Bytes
0x05, 0x18	Identity Request message
0x2b	fill bits

*/

BEGINARRAY (BS_I_7_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x6E,
 0x09,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x01	RR frame, P=0, N(R)=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x01,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x11	RR frame, P=1, N(R)=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_0_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x11,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x21	RR frame, P=0, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x21,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x31	RR frame, P=1, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_1_P, 27)

0xB8, 0x00,

0x00, 0x00,

0x03,

0x31,

0x01,

0x2B, 0x2B, 0x2B, 0x2B, 0x2B,

0x2B, 0x2B, 0x2B, 0x2B, 0x2B,

0x2B, 0x2B, 0x2B, 0x2B, 0x2B,

0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x31	RR frame, P=1, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_1_F, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x31,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x41	RR frame, P=0, N(R)=2
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x41,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x51	RR frame, P=1, N(R)=2
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_2_F, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x51,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x61	RR frame, P=0, N(R)=3
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (BS_RR_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x61,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0xA1	RR frame, P=0, N(R)=5
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINRRAY (BS_RR_5, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xA1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDRRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0xC1	RR frame, P=0, N(R)=6
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINRRAY (BS_RR_6, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xC1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDRRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0xE1	RR frame, P=0, N(R)=7
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (BS_RR_7, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xE1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0x10, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x05, 0x18	Identity Request message

*/

BEGINARRA Y (ID_REQUEST, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x05, 0x18

ENDARRA Y

/*

0x10, 0x00	length in bits
0x18, 0x00	offset in bits
0x00, ...	layer 2 header
0x05, 0x19	Identity Response message

*/

BEGINARRA Y (ID_RESPONSE, 9)

0x10, 0x00,
 0x18, 0x00,
 0x00, 0x00, 0x00,
 0x05, 0x19

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x20	I frame, P=0, N(S)=0, N(R)=1
0x53	content 20 Bytes, M=1
0x0A,....	message

*/

BEGINARRAY (MS_I_0_1_M1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x20,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x22	I frame, P=0, N(S)=1, N(R)=1
0x11	content 4 Bytes, M=0
0x3C,....	message

*/

BEGINARRAY (MS_I_1_1_M2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x22,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x52	I frame, P=1, N(S)=1, N(R)=2
0x11	content 4 Bytes, M=0
0x3C,....	message

*/

BEGINARRA Y (MS_I_1_2_M2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x52,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x02	I frame, P=0, N(S)=1, N(R)=0
0x53	content 20 Bytes, M=1
0x0A,....	message

*/

BEGINARRA Y (MS_I_1_0_M1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x02,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x04	I frame, P=0, N(S)=2, N(R)=0
0x11	content 4 Bytes, M=0
0x3C,....	message

*/

BEGINARRA Y (MS_I_2_0_M2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x04,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x20	I frame, P=0, N(S)=0, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_0_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x20,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x30	l frame, P=1,N(S)=0, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_0_1_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x30,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x50	l frame, P=1,N(S)=0, N(R)=2
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_0_2_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x50,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0xA0	l frame, P=0,N(S)=0, N(R)=5
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_0_5, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xA0,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x42	l frame, P=0,N(S)=1, N(R)=2
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_1_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x42,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0xC2	l frame, P=0,N(S)=1, N(R)=6
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_1_6, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xC2,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0xE4	l frame, P=0,N(S)=2, N(R)=7
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_2_7, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0xE4,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x06	l frame, P=0,N(S)=3, N(R)=0
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_3_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x06,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x28	l frame, P=0,N(S)=4, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_4_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x28,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x2A	I frame, P=0, N(S)=5, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_5_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x2A,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x2C	I frame, P=0, N(S)=6, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_6_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x2C,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command from MS
0x2E	I frame, P=0, N(S)=7, N(R)=1
0x09	content 2 Bytes
0x05, 0x19	Identity Response message
0x2b	fill bits

*/

BEGINARRA Y (MS_I_7_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x2E,
 0x09,
 0x05, 0x19,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x01	RR frame, P=0, N(R)=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_0, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x01,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x11	RR frame, F=1, N(R)=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_0_F, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x11,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x21	RR frame, P=0, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x21,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x31	RR frame, F=1, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_1_F, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x31,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x41	RR frame, P=0, N(R)=2
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x41,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x61	RR frame, P=0, N(R)=3
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x61,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x81	RR frame, P=0, N(R)=4
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_4, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x81,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0xA1	RR frame, P=0, N(R)=5
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_5, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0xA 1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0xC1	RR frame, P=0, N(R)=6
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_6, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0xC 1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0xE1	RR frame, P=0, N(R)=7
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_RR_7, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0xE1,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x53	DISC frame, P=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (DISC_FRAME, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x53,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x73	UA frame, F=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (UA_FRAME, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x73,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS (invalid)
0x3F	SABM, P=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (BS_SABM_C, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x3F,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x29	REJ frame, P=0, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_REJ_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x29,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, response from MS
0x39	REJ frame, P=1, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (MS_REJ_1_F, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x39,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x21	RR frame, P=0, N(R)=1
0x05	content 2 Bytes
0x05, 0x18	content
0x2b	fill bits

*/

BEGINARRA Y (INV_1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x21,
 0x05,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x00	sapi 0, response to MS, EA=0
0x29	REJ frame, P=0, N(R)=1
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (INV_2, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x00,
 0x29,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x3F	SABM, P=1
0x00	content 0 Bytes, EL=0
0x2b	fill bits

*/

BEGINARRA Y (INV_3, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x3F,
 0x00,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, response to MS
0x1F	DM frame, P=1
0x05	content 2 Bytes
0x05, 0x18	content
0x2b	fill bits

*/

BEGINARRA Y (INV_4, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x1F,
 0x05,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x53	DISC frame, P=1
0x02	content 0 Bytes, M=1
0x2b	fill bits

*/

BEGINARRA Y (INV_5, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x53,
 0x02,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x00	sapi 0, response to MS, EA = 0
0x63	UA frame, P=0
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRA Y (INV_6, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x00,
 0x63,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x0C	l frame, P=0, N(R)=0, N(S)=6
0x55	content 21 Bytes
0x05, 0x18	content
0x2b	fill bits

*/

BEGINARRA Y (INV_7, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x0C,
 0x55,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x0E	l frame, P=0, N(R)=0, N(S)=7
0x0B	content 2 Bytes, M=1
0x05, 0x18	content
0x2b	fill bits

*/

BEGINARRA Y (INV_8, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x0E,
 0x0B,
 0x05, 0x18,
 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRA Y

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x1D	control BEGINARRAY = xxx1 1101
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_9, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x1D,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x1B	control BEGINARRAY = xxx1 1011
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_10, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x1B,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x17	control BEGINARRAY = 0x1 0111
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_11, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x17,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x5F	control BEGINARRAY = 01x1 1111
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_12, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x5F,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x9F	control BEGINARRAY = 1x1 1111
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_13, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x9F,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x33	control BEGINARRAY = 0011 0011
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_14, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x33,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	sapi 0, command to MS
0x93	control BEGINARRAY = 1xx1 0011
0x01	content 0 Bytes
0x2b	fill bits

*/

BEGINARRAY (INV_15, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x93,
 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command
0x00	l, Nr = 0, N(S)=0, P=0
0x19	length = 6, M=0
0x0A, ...	content
0x2B	fill bits

*/

BEGINARRAY (I_DL151_UL, 24)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x00,
 0x41,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

BEGINARRAY (I_DL151_DL, 14)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x20,
 0x19,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
 0x2B

ENDARRAY

BEGINARRAY (I_DL151_ULa, 24)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x22,
 0x41,

0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

BEGINARRAY (I_DL151_DLa, 14)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x42,
 0x19,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
 0x2B

ENDARRAY

BEGINARRAY (I_DL151_ULb, 24)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x44,
 0x41,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2B

ENDARRAY

BEGINARRAY (I_DL151_DLb, 14)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x52,
 0x19,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,
 0x2B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x03	Response MS to BS
0x19	REJ, F=1, N(R)=2
0x01	L=0, M=0
0x2B, ..	fill bits

*/

BEGINARRAY (REJ_DL151_ULc, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x03,
 0x59, 0x01,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
 0x2B, 0x2B, 0x2B, 0x2B, 0x2B

ENDARRAY

BEGINARRAY (I_DL151_DLc, 14)

0xB8, 0x00,
 0x00, 0x00,
 0x03,

```
    0x44,  
    0x19,  
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,  
    0x2B  
ENDARRAY  
BEGINARRAY (I_DL151_ULd, 24)  
    0xB8, 0x00,  
    0x00, 0x00,  
    0x01,  
    0x74,  
    0x41,  
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,  
    0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,  
    0x2B  
ENDARRAY  
BEGINARRAY (I_DL151_DLd, 14)  
    0xB8, 0x00,  
    0x00, 0x00,  
    0x03,  
    0x66,  
    0x19,  
    0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F,  
    0x2B  
ENDARRAY  
/* For testcase DL152ff >>> */  
BEGINARRAY (L3_MSG_CM_SERV_REQ, 20)  
    0x68, 0x00, /* Length in bits */  
    0x18, 0x00, /* Offset in bits */  
    0x00, 0x00, 0x00, /* Free for L2 */  
    0x05, 0x24, 0x41, 0x03, 0x33, 0x19, 0x81, 0x05, 0xf4, 0x18, 0x62, 0x3b, 0xa7  
ENDARRAY  
BEGINARRAY (SABM_MSG_CM_SERV_REQ, 27)  
    0xb8, 0x00,  
    0x00, 0x00,  
    0x01, 0x3f, 0x35,  
    0x05, 0x24, 0x41, 0x03, 0x33, 0x19, 0x81, 0x05, 0xf4, 0x18, 0x62, 0x3b, 0xa7,  
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b  
ENDARRAY  
BEGINARRAY (UA_MSG_CM_SERV_REQ, 27)  
    0xb8,0x00,  
    0x00, 0x00,  
    0x01, 0x73, 0x35,  
    0x05, 0x24, 0x41, 0x03, 0x33, 0x19, 0x81, 0x05, 0xf4, 0x18, 0x62, 0x3b, 0xa7,  
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b  
ENDARRAY  
BEGINARRAY (L3_MSG_CM_SERV_ACC, 9)  
    0x10, 0x00, /* Length in bits */  
    0x18, 0x00, /* Offset in bits */  
    0x00, 0x00, 0x00, /* Free for L2 */  
    0x05, 0x21  
ENDARRAY  
BEGINARRAY (I_MSG_CM_SERV_ACC, 27)  
    0xb8, 0x00, /* Length of PL SDU */  
    0x00, 0x00, /* Offset of PL SDU */  
    0x03, 0x00, 0x09, /* L2 header */  
    0x05, 0x21, 0x2b,  
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b  
ENDARRAY
```

```
BEGINARRAY (L3_MSG_CLASSM_CHG, 17)
    0x50, 0x00, /* Length in bits */
    0x18, 0x00, /* Offset in bits */
    0x00, 0x00, 0x00, /* Free for L2 */
    0x06, 0x16, 0x03, 0x33, 0x19, 0x81, 0x20, 0x02, 0x60, 0x14
ENDARRAY
BEGINARRAY (I_MSG_CLASSM_CHG_P0, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x00, 0x29,
    0x06, 0x16, 0x03, 0x33, 0x19, 0x81, 0x20, 0x02, 0x60, 0x14,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (I_MSG_CLASSM_CHG_P1, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x30, 0x29,
    0x06, 0x16, 0x03, 0x33, 0x19, 0x81, 0x20, 0x02, 0x60, 0x14,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (RR_RSP_DL_NR1_F0, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x21, 0x01,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (RR_RSP_DL_NR2_F0, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x41, 0x01,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (RR_RSP_DL_NR3_F0, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x61, 0x01,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (REJ_RSP_DL_NR1_F1, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x39, 0x01,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (REJ_RSP_DL_NR2_F1, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x59, 0x01,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (RR_RSP_DL_NR1_F1, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x31, 0x01,
```

0x2b,
 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b, 0x2b

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x22	I, P=0, N@=1, N(S)=1
0x53	length = 20, M=1
0x0A, ...	content

*/

BEGINARRAY (I_I20_UL_NRI_NS1, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x22,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x32	I, P=1, N@=1, N(S)=1
0x53	length = 20, M=1
0x0A, ...	content

*/

BEGINARRAY (I_I20_UL_NRI_NS1_P, 27)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x32,
 0x53,
 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x1A, 0x1B,
 0x1C, 0x1D, 0x1E, 0x1F, 0x2A, 0x2B, 0x2C, 0x2D,
 0x2E, 0x2F, 0x3A, 0x3B

ENDARRAY

/*

0xB8, 0x00	length in bits
0x00, 0x00	offset in bits
0x01	sapi 0, command to BS
0x02	I, P=0, NR=1, NS=2
0x11	length = 4, M=0
0x3C, ...	content

*/

BEGINARRAY (I_I04_UL_NRI_NS2, 12)

0xB8, 0x00,
 0x00, 0x00,
 0x01,
 0x24,
 0x11,
 0x3C, 0x3D, 0x3E, 0x3F,

```
0x2B
ENDARRAY
BEGINARRAY (L3_MSG_SETUP, 24)
    0x88, 0x00, /* Length in bits */
    0x18, 0x00, /* Offset in bits */
    0x00, 0x00, 0x00, /* Free for L2 */
    0x03, /* CC PD */
    0x45, /* SETUP message type */
    0x04, 0x04, 0x60, 0x02, 0x00, 0x81, /* BC 1 */
    0x5e, 0x07, 0x81, 0x31, 0x60, 0x75, 0x01, 0x16, 0xf9 /* Called party BCD number */
ENDARRAY
BEGINARRAY (I_SETUP_UL_NRI_NS1, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x22, 0x45,
    0x03, /* CC PD */
    0x45, /* SETUP message type */
    0x04, 0x04, 0x60, 0x02, 0x00, 0x81, /* BC 1 */
    0x5e, 0x07, 0x81, 0x31, 0x60, 0x75, 0x01, 0x16, 0xf9, /* Called party BCD number */
    0x2b, 0x2b, 0x2b
ENDARRAY
BEGINARRAY (I_SETUP_UL_NRI_NS1_P, 27)
    0xb8, 0x00,
    0x00, 0x00,
    0x01, 0x32, 0x45,
    0x03, /* CC PD */
    0x45, /* SETUP message type */
    0x04, 0x04, 0x60, 0x02, 0x00, 0x81, /* BC 1 */
    0x5e, 0x07, 0x81, 0x31, 0x60, 0x75, 0x01, 0x16, 0xf9, /* Called party BCD number */
    0x2b, 0x2b, 0x2b
ENDARRAY
/* <<< For testcase DL152ff */
/* cs (Error cause) */
BYTE ERR_T200_EXPIRY 0
BYTE ERR_UNSQL_UA_RSP 2
BYTE ERR_UNSQL_DM_RSP 3
BYTE ERR_UN_DM_RSP_EST 4
BYTE ERR_SEQUENCE 6
BYTE ERR_U_FR_PARAM 7
BYTE ERR_S_FR_PARAM 8
BYTE ERR_I_FR_M_BIT 9
BYTE ERR_I_FR_LENGTH 10
BYTE ERR_FR_NOT_IMPL 11
SHORT Bitm 0x01
BYTE ENTITY 0x02
```

4 TEST CASES

4.1 Routing (internal)

4.1.1 DL000: Setup the routing and PCO view for the DL test

Description: Routings for the DL tests are set

Preamble: None

MM	RR	PL/DL
COMMAND (TAP RESET)		
COMMAND (MMI RESET)		
COMMAND (CC RESET)		
COMMAND (SS RESET)		
COMMAND (SMS RESET)		
COMMAND (MM RESET)		
COMMAND (RR RESET)		
COMMAND (DL RESET)		
COMMAND (SIM RESET)		
COMMAND (PL RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CLEAR)		
COMMAND (CC REDIRECT CLEAR)		
COMMAND (SS REDIRECT CLEAR)		
COMMAND (SMS REDIRECT CLEAR)		
COMMAND (MM REDIRECT CLEAR)		
COMMAND (RR REDIRECT CLEAR)		
COMMAND (DL REDIRECT CLEAR)		
COMMAND (SIM REDIRECT CLEAR)		
COMMAND (PL REDIRECT CLEAR)		
COMMAND (MMI REDIRECT MM NULL)		
COMMAND (MMI REDIRECT CC NULL)		
COMMAND (MMI REDIRECT SS NULL)		
COMMAND (MMI REDIRECT SMS NULL)		
COMMAND (MMI REDIRECT PL NULL)		
COMMAND (CC REDIRECT MMI NULL)		
COMMAND (CC REDIRECT MM NULL)		
COMMAND (SS REDIRECT MMI NULL)		
COMMAND (SS REDIRECT MM NULL)		
COMMAND (SMS REDIRECT MMI NULL)		
COMMAND (SMS REDIRECT MM NULL)		
COMMAND (MM REDIRECT MMI NULL)		
COMMAND (MM REDIRECT CC NULL)		
COMMAND (MM REDIRECT SS NULL)		
COMMAND (MM REDIRECT SMS NULL)		
COMMAND (MM REDIRECT SIM NULL)		
COMMAND (MM REDIRECT RR NULL)		
COMMAND (MM REDIRECT DL NULL)		
COMMAND (RR REDIRECT PL NULL)		
COMMAND (RR REDIRECT DL NULL)		
COMMAND (RR REDIRECT MM NULL)		

COMMAND (DL REDIRECT RR TAP)		
COMMAND (DL REDIRECT MM TAP)		
COMMAND (DL REDIRECT PL TAP)		
COMMAND (PL REDIRECT RR NULL)		
COMMAND (PL REDIRECT DL NULL)		
COMMAND (PL REDIRECT MMI NULL)		
COMMAND (SIM REDIRECT MM NULL)		
COMMAND (TAP REDIRECT TAP DL)		

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
------------------	------------------	--------------

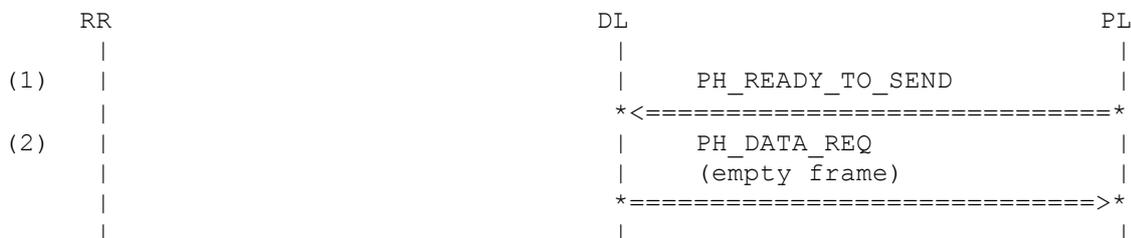
History: 04.07.97 PZ Initial

4.2 Unacknowledged Information Transfer

4.2.1 DL002: Transmit empty frame (SDCCH)

Description: DL receives a PH-READY-TO-SEND from PL. No data has been received from Layer, and DL issues a PH-DATA request primitive with an empty frame to PL.

Preamble: DL000



Parametrization

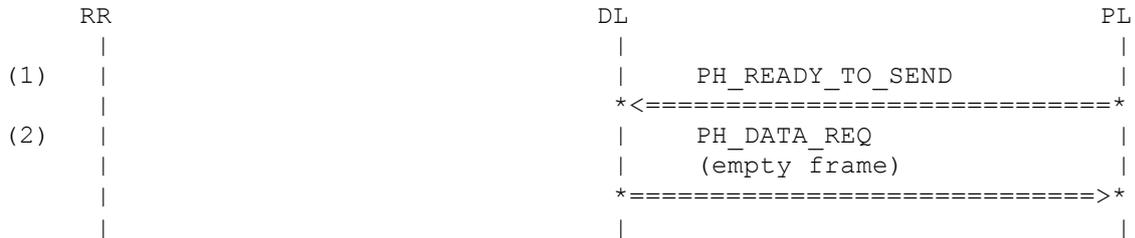
Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	EMPTY_FRAME_DCCH

History: 21.03.97 DL Initial

4.2.2 DL003: Transmit empty frame (SACCH)

Description: DL receives a PH-READY-TO-SEND from PL. No data has been received from Layer, and DL issues a PH-DATA request primitive with an empty frame to PL.

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
(2) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	sdu	EMPTY_FRAME_SACCH

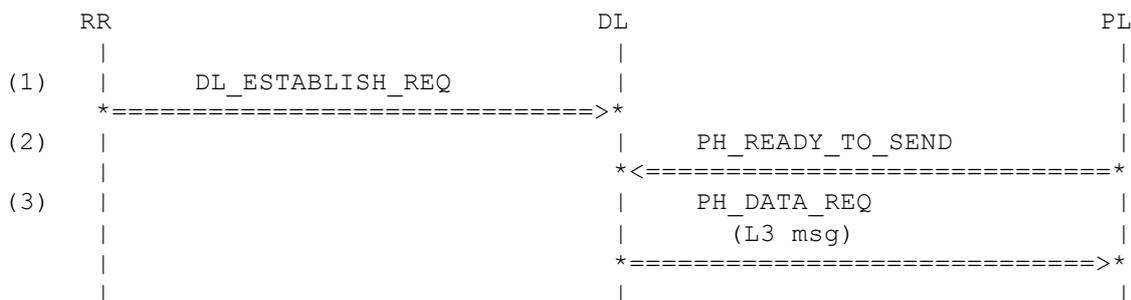
History: 21.03.97 DL Initial

4.3 Contention Resolution from Idle mode (SDCCH)

4.3.1 DL011: Initiation by Radio Resource

Description: The MS (RR) initiates establishment of a data link in order to resolve contention following access to the BS on the RACH by DL-ESTABLISHMENT request primitive containing a Layer-3 message to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then forwards the message in a SABM frame with the P-bit set to 1. (Ref. [20] 5.4.1.4)

Preamble: DL000



Parametrization

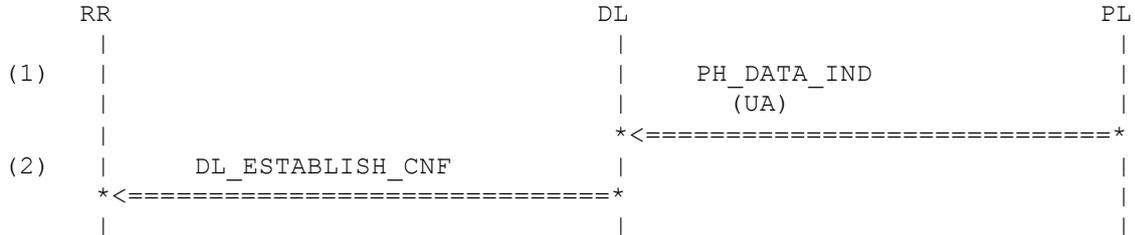
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL

History: 21.03.97 DL Initial

4.3.2 DL012: UA response received from Base Station (UA = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA corresponds to the previous SAB. DL enters the multi-frame-established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL011



Parametrization

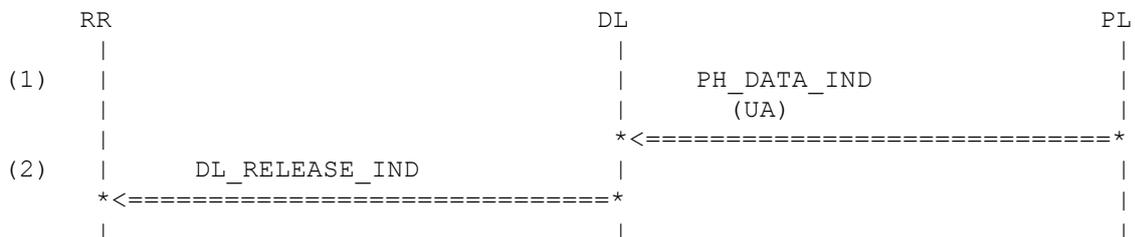
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0

History: 21.03.97 DL Initial

4.3.3 DL013: UA response received from Base Station (UA not = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA differs from the previous SABM. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL011



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH

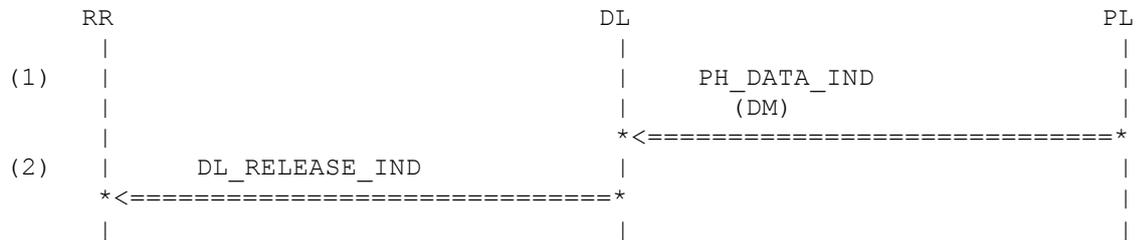
History:

21.03.97	DL	Initial
07.08.97	LE	cs added for DL_RELEASE_IND

4.3.4 DL014: DM response received from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL011



Parametrization

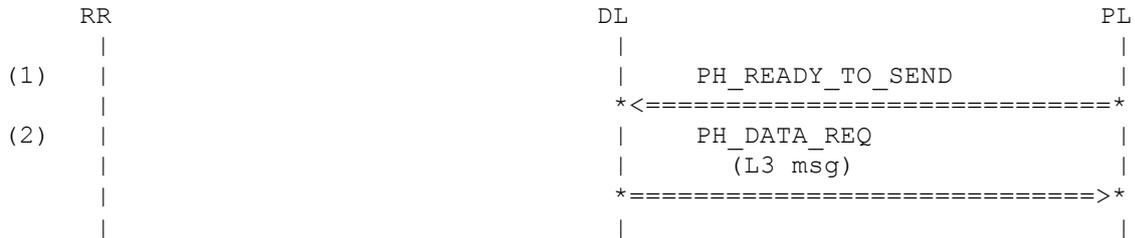
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.3.5 DL015: Expiry of T200 timer - max. number of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a PH-DATA request primitive. (Ref. [20] 5.4.1.3)

Preamble: DL011



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL

History: 21.03.97 DL Initial

4.3.6 DL016: Expiry of T200 timer - maximum number of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer 'T200 expired N200+ 1 times.“ (Ref. [20] 5.4.1.5)

Preamble: DL011

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_DATA_REQ	
		(L3 msg)	
		=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_DATA_REQ	
		(L3 msg)	
		=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_DATA_REQ	
		(L3 msg)	
		=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_DATA_REQ	
		(L3 msg)	
		=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(L3 msg)	
		=====	
(11)		PH_READY_TO_SEND	
		<=====	
(12)	DL_RELEASE_IND		
	<=====		
(13)		PH_DATA_REQ	
		(empty frame)	
		=====	

Parametrization

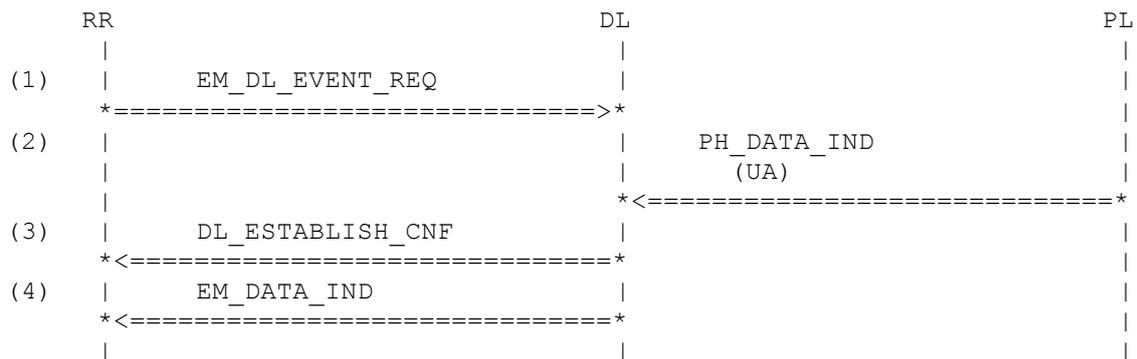
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(13) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND
 08.07.98 LE CONFIG removed

4.3.7 DL017: DL012 – Engineering mode

Description: PL sends DL a UA response in which the F bit set to 1 and the UA corresponds to the previous SAB. DL enters the multi-frame-established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL011



Parametrization

Primitive	Parameter	Value
(1) EM_DL_EVENT_REQ	bitmask_dl	Bitm
(2) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(3) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(4) EM_DATA_IND	entity	ENTITY

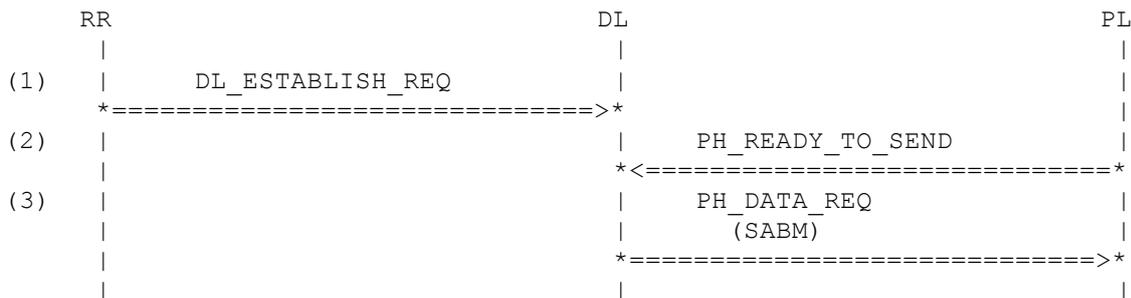
History: 23-Oct-01 OT Initial

4.4 Contention Resolution from Idle mode (FACCH Fullrate, SACCH)

4.4.1 DL021: Initiation by Radio Resource

Description: The MS (RR) initiates establishment of a data link in order to resolve contention following access to the BS on the RACH by DL-ESTABLISHMENT request primitive containing a Layer-3 message to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then forwards the message in a SABM frame with the P-bit set to 1. (Ref. [20] 5.4.1.4)

Preamble: DL000



Parametrization

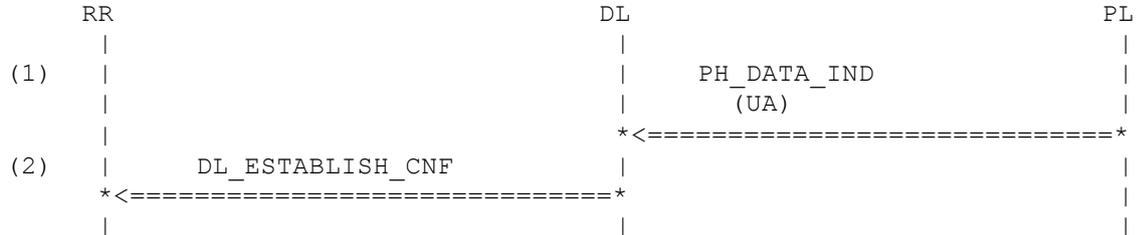
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL

History: 21.03.97 DL Initial

4.4.2 DL022: UA response received from Base Station (UA = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA corresponds to the previous SABM. DL enters the multi-frame-established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL021



Parametrization

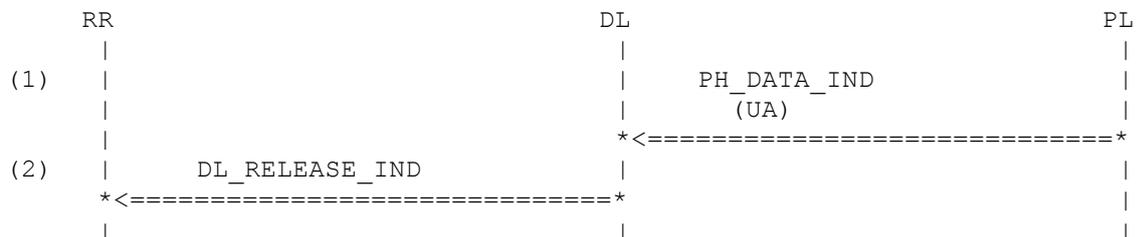
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0

History: 21.03.97 DL Initial

4.4.3 DL023: UA response received from Base Station (UA not = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA differs from the previous SABM. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL021



Parametrization

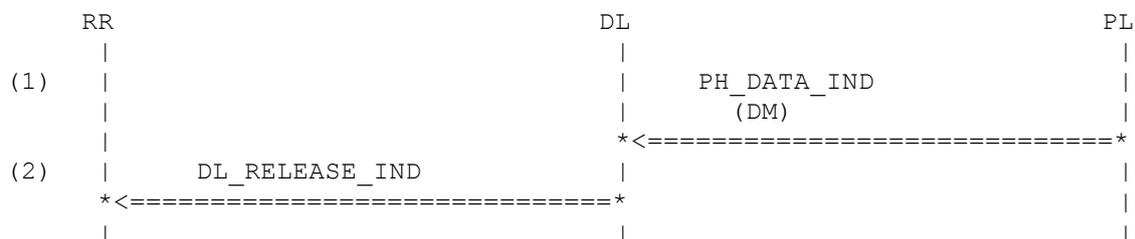
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.4.4 DL024: DM response received from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL021



Parametrization

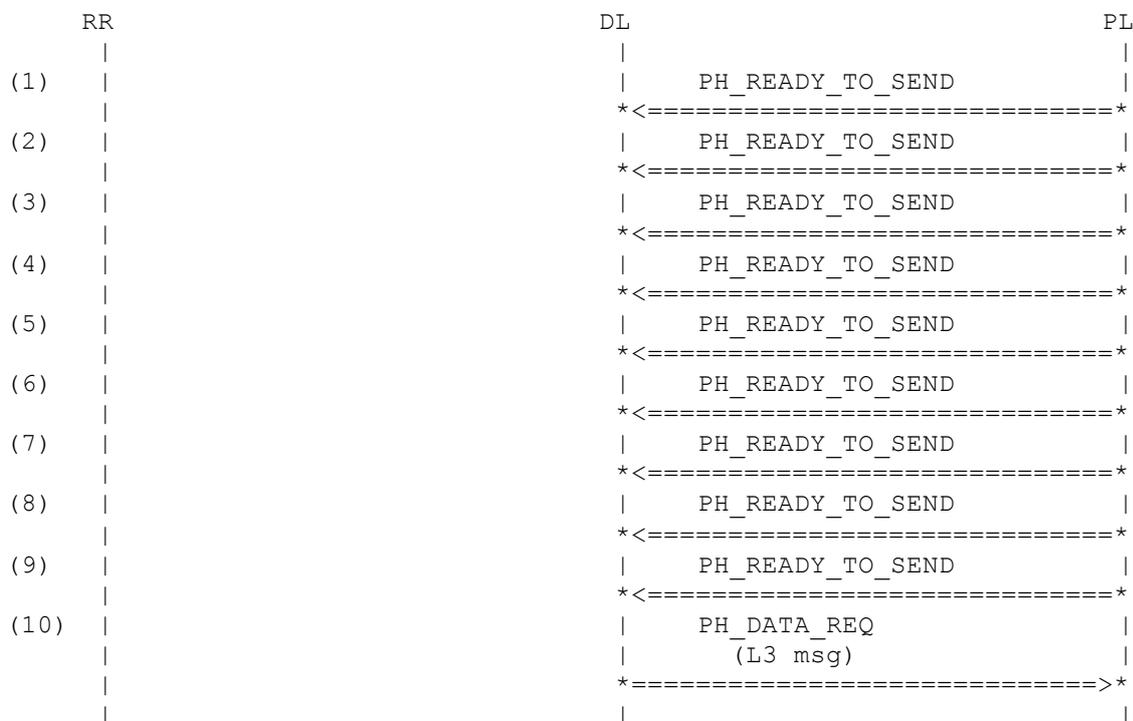
	Primitive	Parameter	Value
(1)	PH_DATA_IND	ch_type	CH_TYPE_FACCH
		dummy	NOT_USED
		sdu	DM_A_DL
(2)	DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
		sapi	SAPI_0
		cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.4.5 DL025: Expiry of T200 timer - max. number of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a PH-DATA request primitive. (Ref. [20] 5.4.1.3)

Preamble: DL021



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	CH_TYPE_FACCH
	ch_type	CH_TYPE_FACCH

dummy
sdu

NOT_USED
SABM_B_I06_UL

History: 21.03.97 DL Initial

4.4.6 DL026: Expiry of T200 timer - maximum number of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer ‘T200 expired N200+ 1 times.“ (Ref. [20] 5.4.1.5)

Preamble: DL021

RR/MM	DL	PL
(1)	PH_READY_TO_SEND	
(2)	PH_READY_TO_SEND	
(3)	PH_READY_TO_SEND	
(4)	PH_READY_TO_SEND	
(5)	PH_READY_TO_SEND	
(6)	PH_READY_TO_SEND	
(7)	PH_READY_TO_SEND	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_DATA_REQ (L3 msg)	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_READY_TO_SEND	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_READY_TO_SEND	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_DATA_REQ (L3 msg)	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_READY_TO_SEND	
(24)	PH_READY_TO_SEND	
(25)	PH_READY_TO_SEND	

```
|
|
(26) | | PH_READY_TO_SEND |
| | |
| | |
(27) | | PH_READY_TO_SEND |
| | |
| | |
(28) | | PH_READY_TO_SEND |
| | |
| | |
(29) | | PH_READY_TO_SEND |
| | |
| | |
(30) | | PH_DATA_REQ |
| | (L3 msg) |
| | |
| | |
(31) | | PH_READY_TO_SEND |
| | |
| | |
(32) | | PH_READY_TO_SEND |
| | |
| | |
(33) | | PH_READY_TO_SEND |
| | |
| | |
(34) | | PH_READY_TO_SEND |
| | |
| | |
(35) | | PH_READY_TO_SEND |
| | |
| | |
(36) | | PH_READY_TO_SEND |
| | |
| | |
(37) | | PH_READY_TO_SEND |
| | |
| | |
(38) | | PH_READY_TO_SEND |
| | |
| | |
(39) | | PH_READY_TO_SEND |
| | |
| | |
(40) | | PH_DATA_REQ |
| | (L3 msg) |
| | |
| | |
(41) | | PH_READY_TO_SEND |
| | |
| | |
(42) | | PH_READY_TO_SEND |
| | |
| | |
(43) | | PH_READY_TO_SEND |
| | |
| | |
(44) | | PH_READY_TO_SEND |
| | |
| | |
(45) | | PH_READY_TO_SEND |
| | |
| | |
(46) | | PH_READY_TO_SEND |
| | |
| | |
(47) | | PH_READY_TO_SEND |
| | |
| | |
(48) | | PH_READY_TO_SEND |
| | |
| | |
(49) | | PH_READY_TO_SEND |
| | |
| | |
(50) | | PH_DATA_REQ |
| | (L3 msg) |
| | |
| | |
(51) | | PH_READY_TO_SEND |
| | |
| | |
(52) | | PH_READY_TO_SEND |
| | |
| | |
(53) | | PH_READY_TO_SEND |
| | |
| | |
```

```

(54) | | PH_READY_TO_SEND |
| | *<===== |
(55) | | PH_READY_TO_SEND |
| | *<===== |
(56) | | PH_READY_TO_SEND |
| | *<===== |
(57) | | PH_READY_TO_SEND |
| | *<===== |
(58) | | PH_READY_TO_SEND |
| | *<===== |
(59) | | PH_READY_TO_SEND |
| | *<===== |
(60) | DL_RELEASE_IND |
| *<===== |
| | |

```

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)		PH_DATA_REQ	
		ch_type	CH_TYPE_FACCH
		dummy	NOT_USED
		sdu	SABM_B_I06_UL
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_B_I06_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_B_I06_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_B_I06_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_B_I06_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

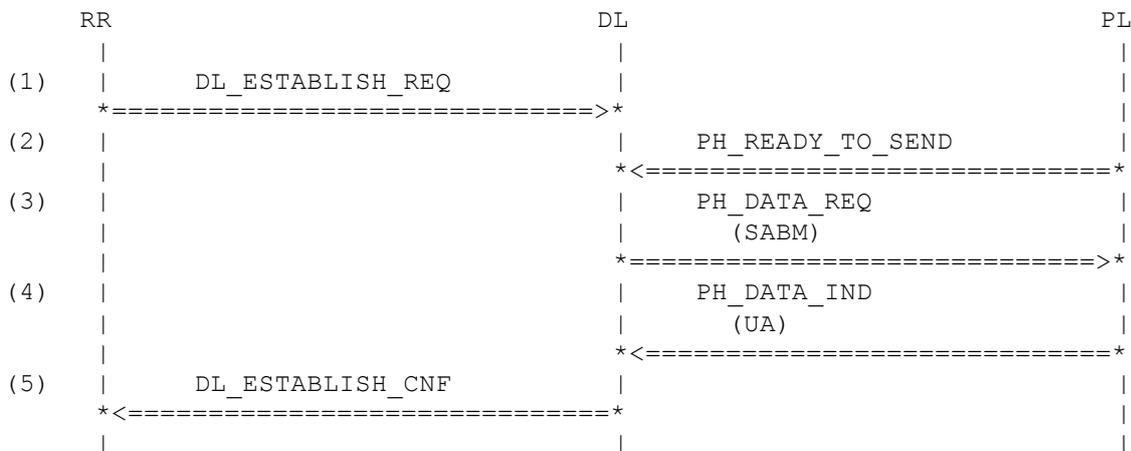
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
07.08.97 LE cs added for DL_RELEASE_IND
08.07.98 LE CONFIG removed

4.4.7 DL027: Initiation and establishment of multiple frame state (SACCH, SAPI 3)

Description: RR establishes a layer 2 connection on SACCH for SAPI 3.

Preamble: DL022



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SA_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UA_A_S3_SA_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3

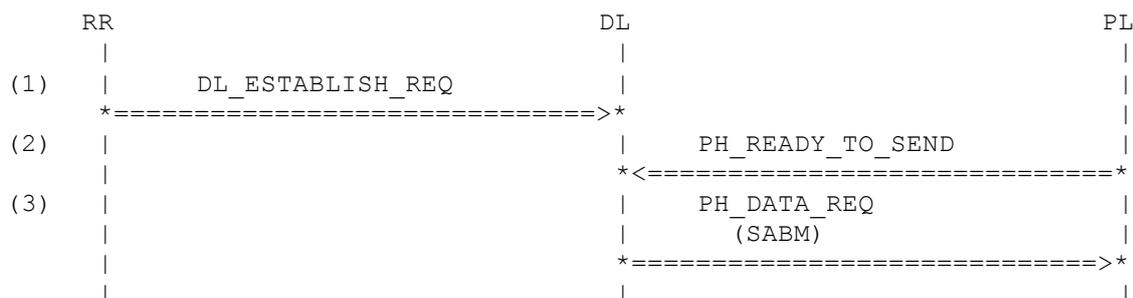
History: 21.03.97 DL Initial

4.5 Contention Resolution from Idle mode (FACCH Halfrate, SACCH)

4.5.1 DL421: Initiation by Radio Resource

Description: The MS (RR) initiates establishment of a data link in order to resolve contention following access to the BS on the RACH by DL-ESTABLISHMENT request primitive containing a Layer-3 message to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then forwards the message in a SABM frame with the P-bit set to 1. (Ref. [20] 5.4.1.4)

Preamble: DL000



Parametrization

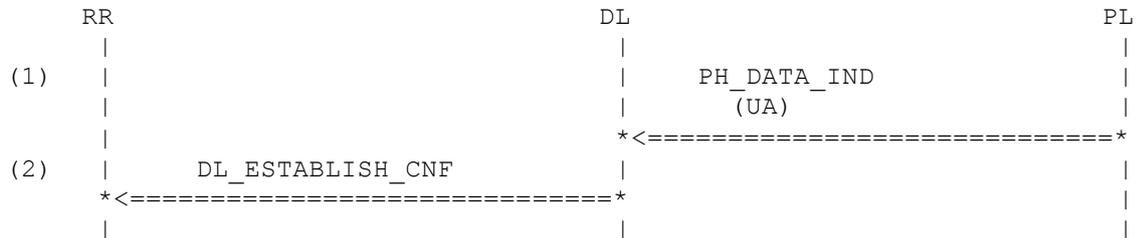
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_UL

History: 08.07.98 LE Initial

4.5.2 DL422: UA response received from Base Station (UA = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA corresponds to the previous SABM. DL enters the multi-frame-established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL421



Parametrization

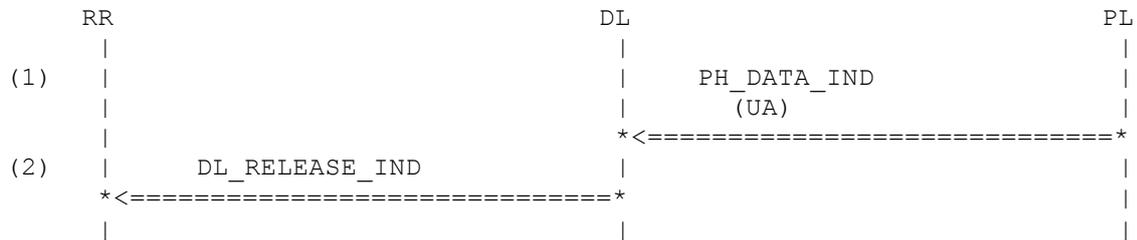
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0

History: 08.07.98 LE Initial

4.5.3 DL423: UA response received from Base Station (UA not = SABM)

Description: PL sends DL a UA response in which the F bit set to 1 and the UA differs from the previous SABM. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL421



Parametrization

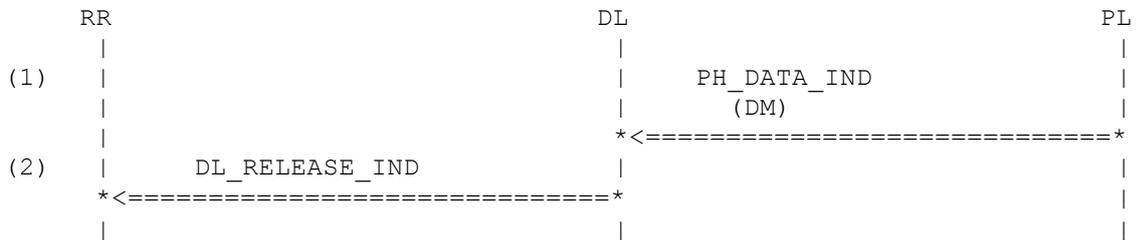
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH

History: 08.07.98 LE Initial

4.5.4 DL424: DM response received from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL421



Parametrization

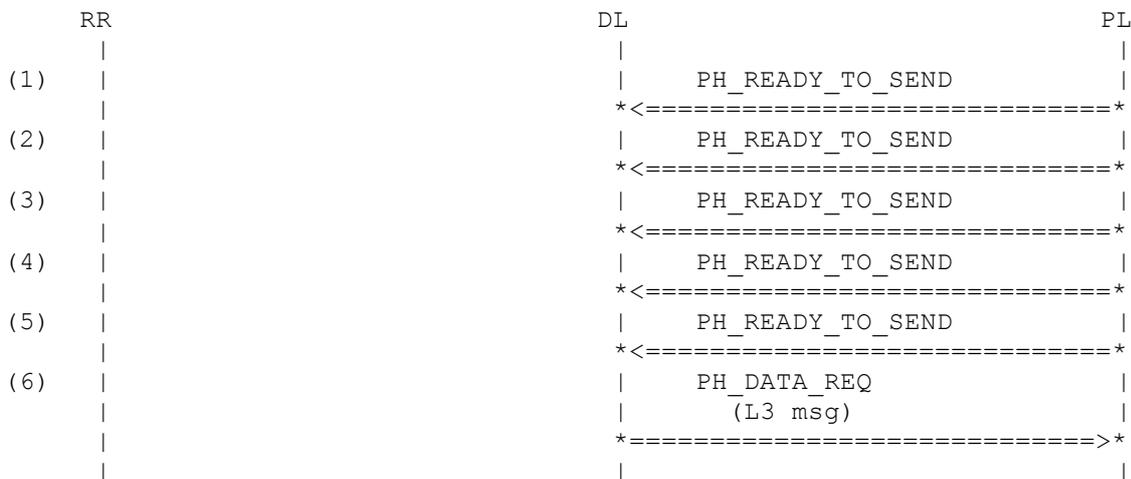
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.5.5 DL425: Expiry of T200 timer - max. number of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a PH-DATA request primitive. (Ref. [20] 5.4.1.3)

Preamble: DL421



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_UL

History: 08.07.98 DL Initial

4.5.6 DL426: Expiry of T200 timer - maximum number of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer ‘T200 expired N200+ 1 times.“ (Ref. [20] 5.4.1.5)

Preamble: DL421

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(L3 msg)	
		*=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(L3 msg)	
		*=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	


```

(54) | | PH_READY_TO_SEND |
| | *<=====*
(55) | | PH_READY_TO_SEND |
| | *<=====*
(56) | | PH_READY_TO_SEND |
| | *<=====*
(57) | | PH_READY_TO_SEND |
| | *<=====*
(58) | | PH_READY_TO_SEND |
| | *<=====*
(59) | | PH_READY_TO_SEND |
| | *<=====*
(60) | DL_RELEASE_IND |
| *<=====*
| |

```

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)		PH_DATA_REQ	
		ch_type	CH_TYPE_FACCH_HR
		dummy	NOT_USED
		sdu	SABM_B_I06_UL
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR

(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_B_I06_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_B_I06_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_B_I06_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_B_I06_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

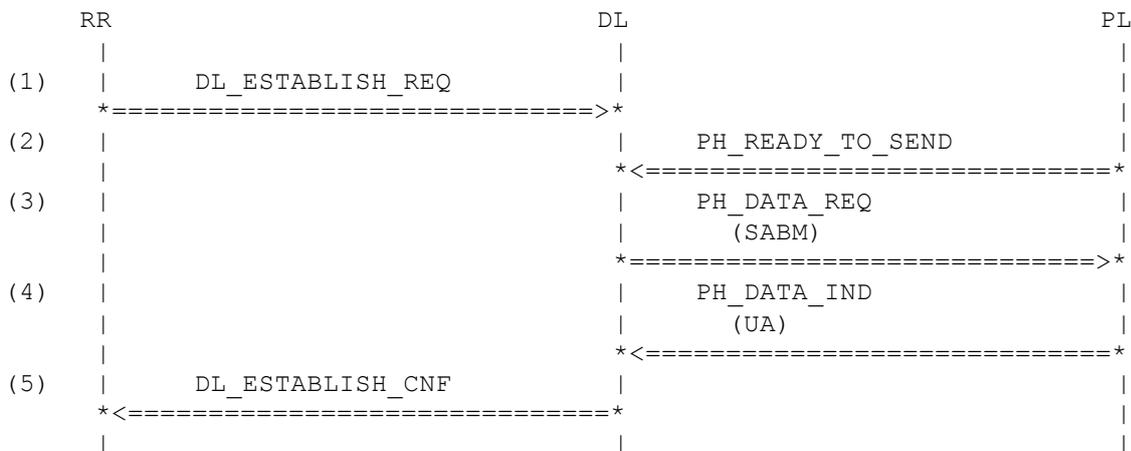
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.5.7 DL427: Initiation and establishment of multiple frame state (SACCH, SAPI 3)

Description: RR establishes a layer 2 connection on SACCH for SAPI 3.

Preamble: DL422



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SA_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UA_A_S3_SA_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3

History: 08.07.98 LE Initial

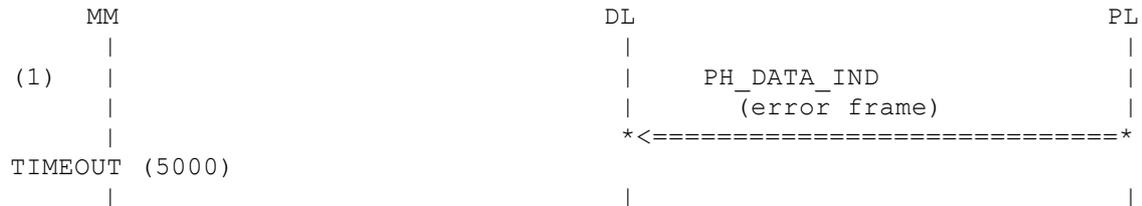
4.6 Error Conditions

4.6.1 DL031: Error frame received (SDCCH)

Description: An error frame is received by DL in the multiple frame establishment state as part of PH-DATA indication primitive, causing DL to ignore the frame and sent nothing to MM.

Preamble: DL012

Variants: <A>...<G>



Parametrization

Primitive	Parameter	Value	
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH	
	dummy	NOT_USED	
	<A>	sdu	ERR_UN SOL_UA_DL
		sdu	ERR_UN SOL_DM_DL
	<C>	sdu	ERR_U_PARAM_DL
	<D>	sdu	ERR_S_PARAM_DL
	<E>	sdu	ERR_I_M_BIT_DL
	<F>	sdu	ERR_I_FR_LEN
<G>	sdu	ERR_I_FR_NOT_IMP	

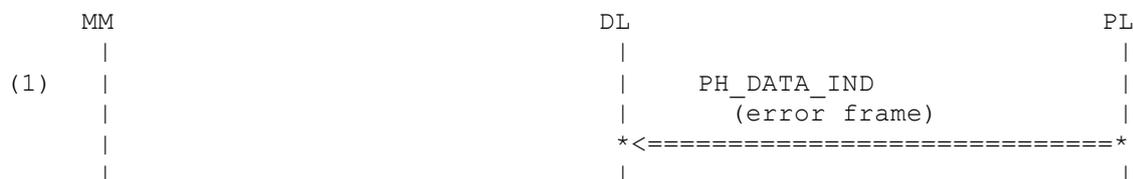
History: 21.03.97 DL Initial

4.6.2 DL032: Error frame received (FACCH Fullrate)

Description: An error frame is received by DL in the multiple frame establishment state as part of PH-DATA indication primitive, causing DL to issue a MDL-ERROR indication to MM with the error cause set according to the frame received. (Ref. [20] 5.4.2.2)

Preamble: DL022

Variants: <A>...<G>



Parametrization

Primitive	Parameter	Value	
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH	
	dummy	NOT_USED	
	<A>	sdu	ERR_UNSQL_UA_DL
		sdu	ERR_UNSQL_DM_DL
	<C>	sdu	ERR_U_PARAM_DL
	<D>	sdu	ERR_S_PARAM_DL
	<E>	sdu	ERR_I_M_BIT_DL
	<F>	sdu	ERR_I_FR_LEN
<G>	sdu	ERR_I_FR_NOT_IMP	

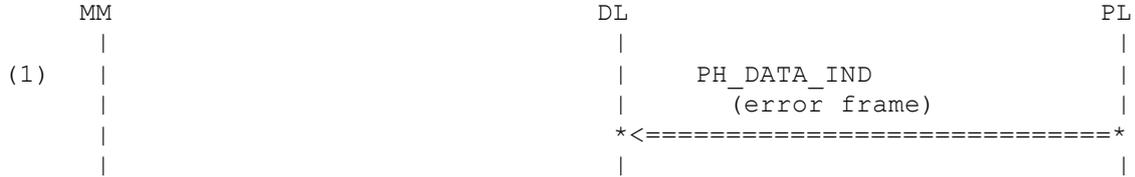
History: 08.07.98 LE Initial

4.6.3 DL033: Error frame received (FACCH Halfrate)

Description: An error frame is received by DL in the multiple frame establishment state as part of PH-DATA indication primitive, causing DL to issue a MDL-ERROR indication to MM with the error cause set according to the frame received. (Ref. [20] 5.4.2.2)

Preamble: DL422

Variants: <A>...<G>



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
<A>	sdu	ERR_UNSQL_UA_DL
	sdu	ERR_UNSQL_DM_DL
<C>	sdu	ERR_U_PARAM_DL
<D>	sdu	ERR_S_PARAM_DL
<E>	sdu	ERR_I_M_BIT_DL
<F>	sdu	ERR_I_FR_LEN
<G>	sdu	ERR_I_FR_NOT_IMP

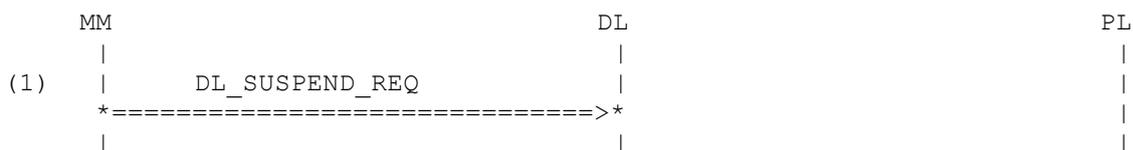
History: 08.07.98 LE Initial

4.7 Suspension and Resumption of Multiframe Operation (SAPI 0, SDCCH)

4.7.1 DL041: Suspend multiframe operation

Description: Layer 3 (RR) requests suspension of multiple frame operation by issuing a DL-SUPSEND request primitive. DL then suspends transmission of any Layer-3 messages in the send buffer, resets the buffer and issues a DL-SUSPEND confirmation primitive to RR. (Ref. [20] 5.4.3.2)

Preamble: DL012



Parametrization

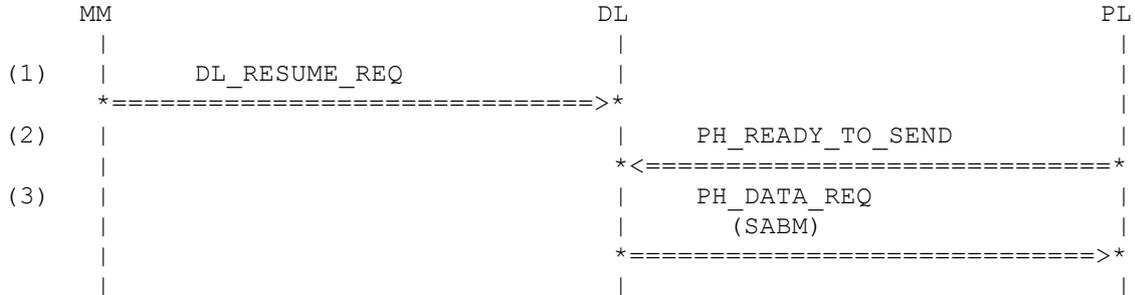
Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0

History: 21.03.97 DL Initial

4.7.2 DL042: Initiate resumption of multiframe operation

Description: Layer 3 (RR) requests resumption of multiple frame operation by issuing a DL-RESUME request primitive containing a Layer-3 message. DL awaits receipt of a PH-READY-TO-SEND primitive and then issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3)

Preamble: DL041



Parametrization

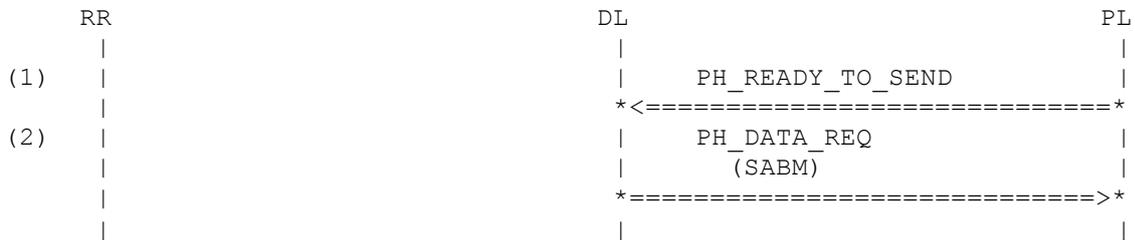
Primitive	Parameter	Value
(1) DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial

4.7.3 DL043: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.4.3)

Preamble: DL042



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial
 08.07.98 LE CONFIG removed

4.7.4 DL044: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer 'T200 expired N200+ 1 times.“ (Ref. [20] 5.4.4.3)

Preamble: DL042

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_DATA_REQ	
		(SABM)	
		=====>	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_DATA_REQ	
		(SABM)	
		=====>	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_DATA_REQ	
		(SABM)	
		=====>	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_DATA_REQ	
		(SABM)	
		=====>	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)	DL_RELEASE_IND		
	<=====		
(13)		PH_DATA_REQ	
		(empty frame)	
		=====>	

Parametrization

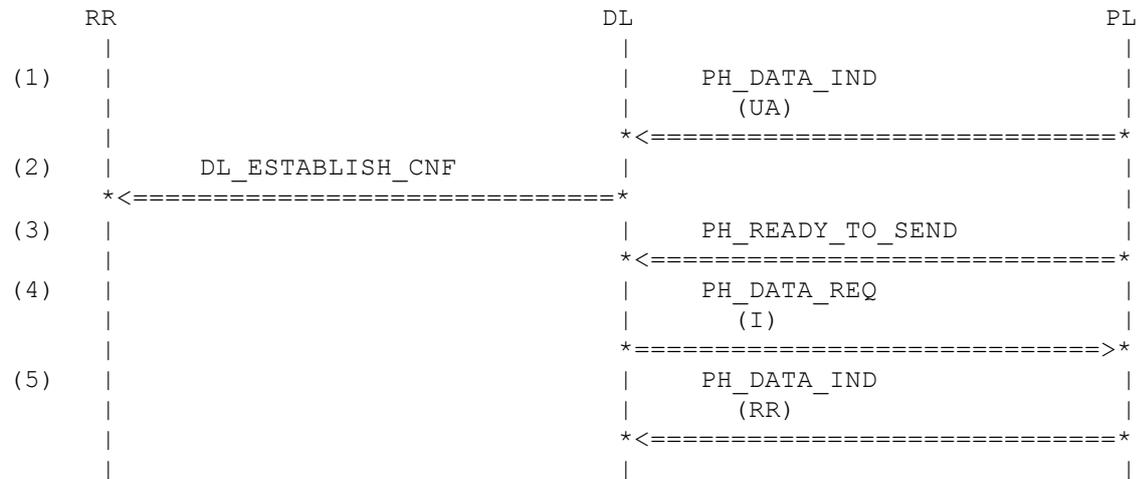
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(12)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(13)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND
 08.07.98 LE CONFIG removed

4.7.5 DL045: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the multiframe established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL042



Parametrization

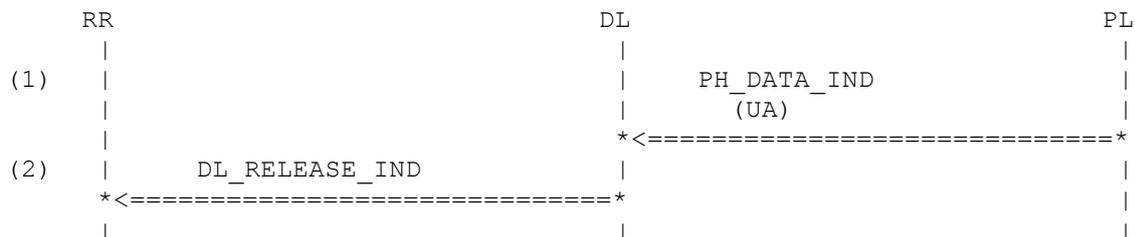
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL

History: 21.03.97 DL Initial

4.7.6 DL046: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL042



Parametrization

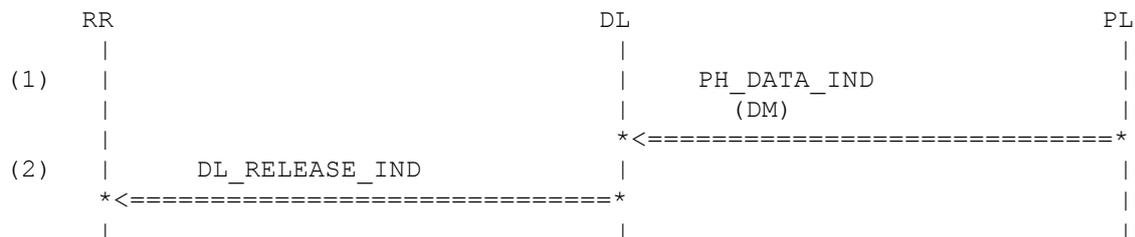
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.7.7 DL047: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL042



Parametrization

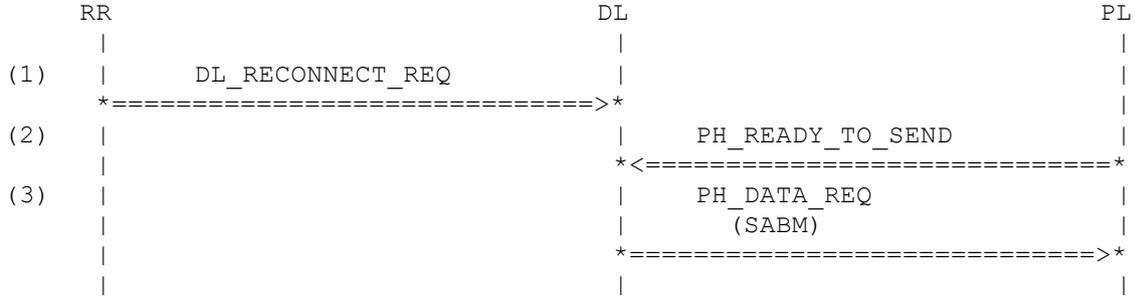
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.7.8 DL051: Initiate reconnection of multiframe operation

Description: Layer 3 (RR) requests reconnection by issuing a DL-RECONNECT request primitive containing a Layer-3 message (Assignment or Handover Failure). DL awaits receipt of a PH-READY-TO-SEND primitive and then issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL047



Parametrization

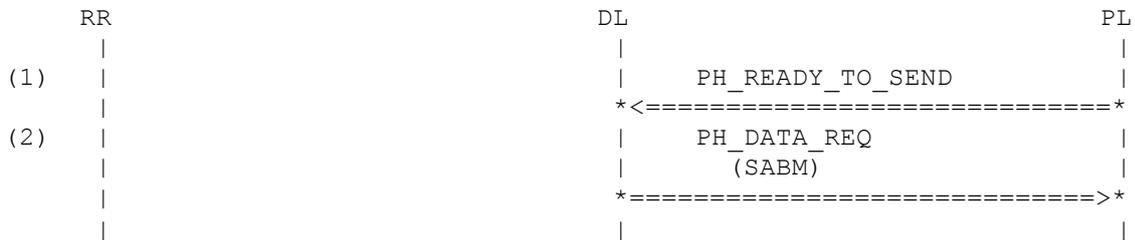
Primitive	Parameter	Value
(1) DL_RECONNECT_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial

4.7.9 DL052: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL051



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial
 08.07.98 LE CONFIG removed

4.7.10 DL053: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer 'T200 expired N200+ 1 times.“ (Ref. [20] 5.4.3.3.2) DL051

Preamble: DL051

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_DATA_REQ	
		(SABM)	
		=====>	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_DATA_REQ	
		(SABM)	
		=====>	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_DATA_REQ	
		(SABM)	
		=====>	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_DATA_REQ	
		(SABM)	
		=====>	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)	DL_RELEASE_IND		
	<=====		
(13)		PH_DATA_REQ	
		(empty frame)	
		=====>	

Parametrization

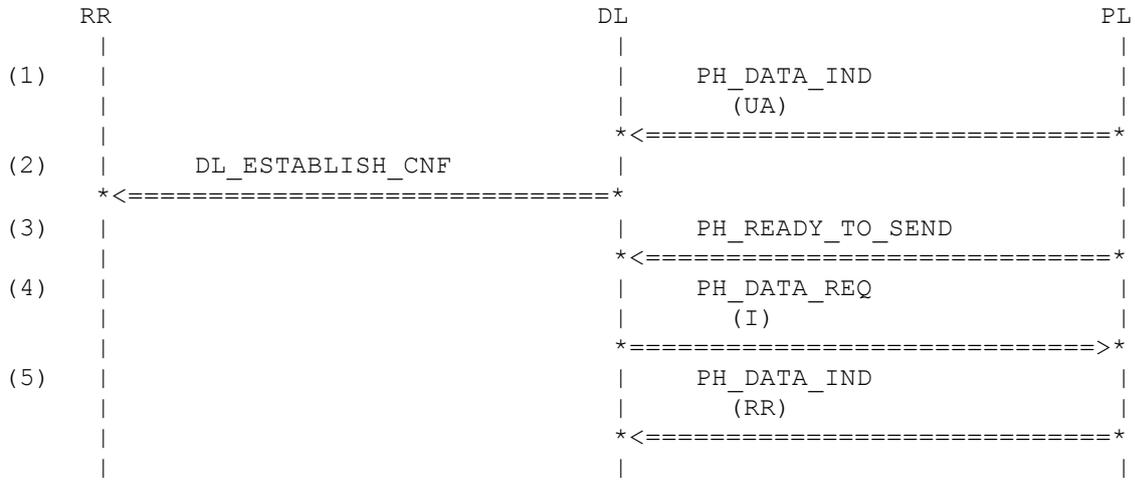
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(14)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND
 08.07.98 LE CONFIG removed

4.7.11 DL054: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the multiframe established state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL051



Parametrization

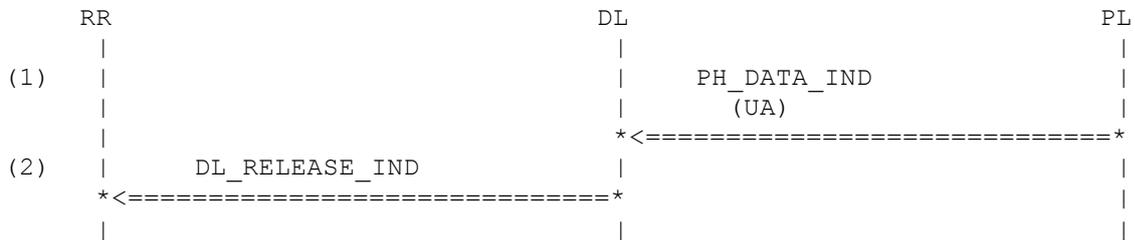
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL

History: 21.03.97 DL Initial

4.7.12 DL055: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL051



Parametrization

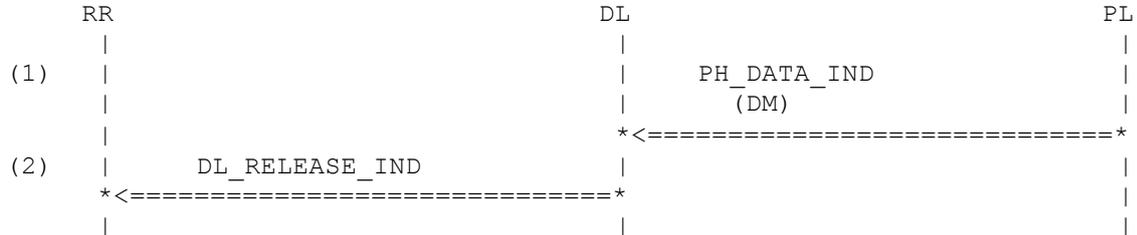
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.7.13 DL056: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL051



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

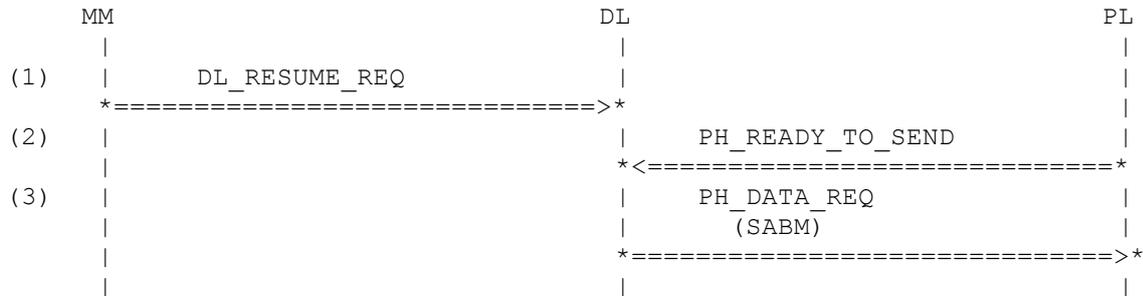
History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.8 Resumption of Multiframe Operation (SAPI 0, FACCH Fullrate)

4.8.1 DL061: Initiate resumption of multiframe operation

Description: Layer 3 (RR) requests resumption of multiple frame operation by issuing a DL-RESUME request primitive containing a Layer-3 message (Assignment or Handover Complete). DL issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3)

Preamble: DL041



Parametrization

Primitive	Parameter	Value
(1) DL_RESUME_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial

4.8.2 DL062: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit= 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.4.3)

Preamble: DL061

RR	DL	PL
(1)	PH_READY_TO_SEND	
	<=====	
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ	
	(SABM)	
	<=====>	

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ ch_type	CH_TYPE_FACCH

dummy
sdu

NOT_USED
SABM_A_UL

History: 21.03.97 DL Initial

4.8.3 DL063: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.4.3)

Preamble: DL061

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		*<=====	
(2)		PH_READY_TO_SEND	
		*<=====	
(3)		PH_READY_TO_SEND	
		*<=====	
(4)		PH_READY_TO_SEND	
		*<=====	
(5)		PH_READY_TO_SEND	
		*<=====	
(6)		PH_READY_TO_SEND	
		*<=====	
(7)		PH_READY_TO_SEND	
		*<=====	
(8)		PH_READY_TO_SEND	
		*<=====	
(9)		PH_READY_TO_SEND	
		*<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		*=====>	
(11)		PH_READY_TO_SEND	
		*<=====	
(12)		PH_READY_TO_SEND	
		*<=====	
(13)		PH_READY_TO_SEND	
		*<=====	
(14)		PH_READY_TO_SEND	
		*<=====	
(15)		PH_READY_TO_SEND	
		*<=====	
(16)		PH_READY_TO_SEND	
		*<=====	
(17)		PH_READY_TO_SEND	
		*<=====	
(18)		PH_READY_TO_SEND	
		*<=====	
(19)		PH_READY_TO_SEND	
		*<=====	
(20)		PH_DATA_REQ	
		(SABM)	
		*=====>	
(21)		PH_READY_TO_SEND	
		*<=====	
(22)		PH_READY_TO_SEND	
		*<=====	
(23)		PH_READY_TO_SEND	
		*<=====	
(24)		PH_READY_TO_SEND	
		*<=====	
(25)		PH_READY_TO_SEND	
		*<=====	

```
(26) | | PH_READY_TO_SEND |
      | | *<=====*
(27) | | PH_READY_TO_SEND |
      | | *<=====*
(28) | | PH_READY_TO_SEND |
      | | *<=====*
(29) | | PH_READY_TO_SEND |
      | | *<=====*
(30) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>*
(31) | | PH_READY_TO_SEND |
      | | *<=====*
(32) | | PH_READY_TO_SEND |
      | | *<=====*
(33) | | PH_READY_TO_SEND |
      | | *<=====*
(34) | | PH_READY_TO_SEND |
      | | *<=====*
(35) | | PH_READY_TO_SEND |
      | | *<=====*
(36) | | PH_READY_TO_SEND |
      | | *<=====*
(37) | | PH_READY_TO_SEND |
      | | *<=====*
(38) | | PH_READY_TO_SEND |
      | | *<=====*
(39) | | PH_READY_TO_SEND |
      | | *<=====*
(40) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>*
(41) | | PH_READY_TO_SEND |
      | | *<=====*
(42) | | PH_READY_TO_SEND |
      | | *<=====*
(43) | | PH_READY_TO_SEND |
      | | *<=====*
(44) | | PH_READY_TO_SEND |
      | | *<=====*
(45) | | PH_READY_TO_SEND |
      | | *<=====*
(46) | | PH_READY_TO_SEND |
      | | *<=====*
(47) | | PH_READY_TO_SEND |
      | | *<=====*
(48) | | PH_READY_TO_SEND |
      | | *<=====*
(49) | | PH_READY_TO_SEND |
      | | *<=====*
(50) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>*
(51) | | PH_READY_TO_SEND |
      | | *<=====*
(52) | | PH_READY_TO_SEND |
      | | *<=====*
(53) | | PH_READY_TO_SEND |
      | | *<=====*
(54) | | PH_READY_TO_SEND |
```


(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

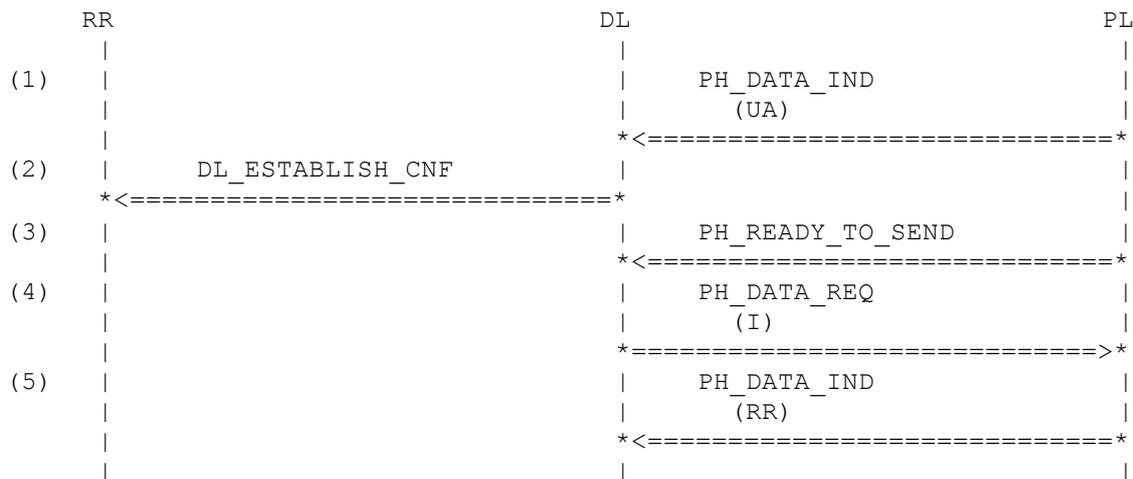
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
07.08.97 LE cs added for DL_RELEASE_IND
08.07.98 LE CONFIG removed

4.8.4 DL064: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the multiframe established state and issues a DL-RESUME confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL061



Parametrization

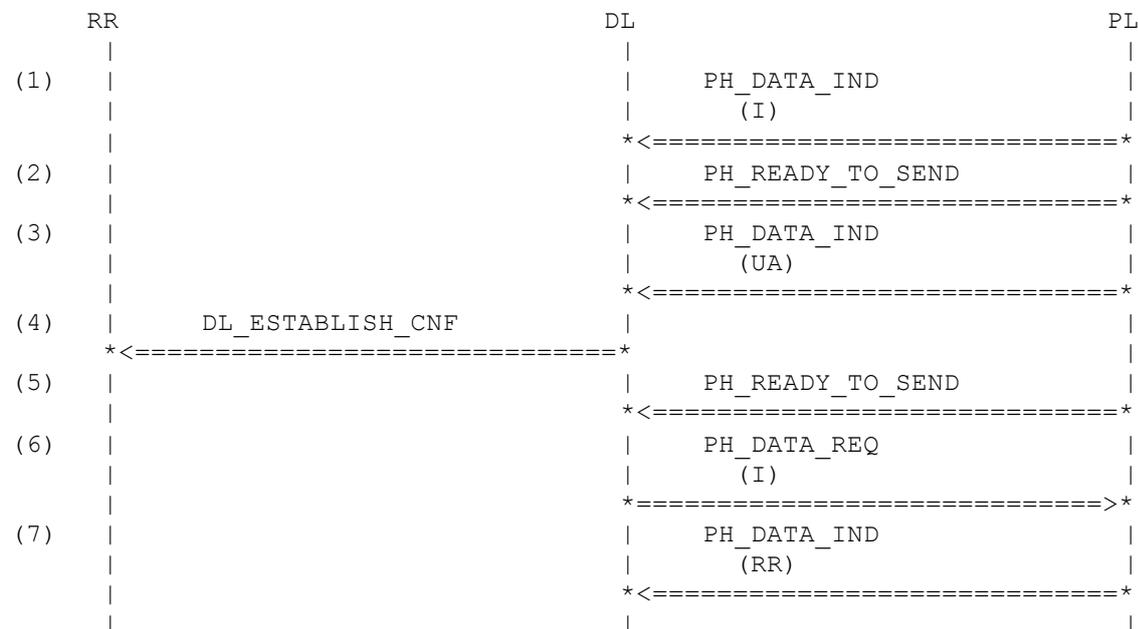
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL

History: 21.03.97 DL Initial

4.8.5 DL264: UA response from Base Station - information field empty (II)

Description: Before the MS receives the expected UA response, the retransmitted Handover command will be received. An UA response with the F bit set to 1 and an empty information field. DL enters the multiframe established state and issues a DL-RESUME confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL061



Parametrization

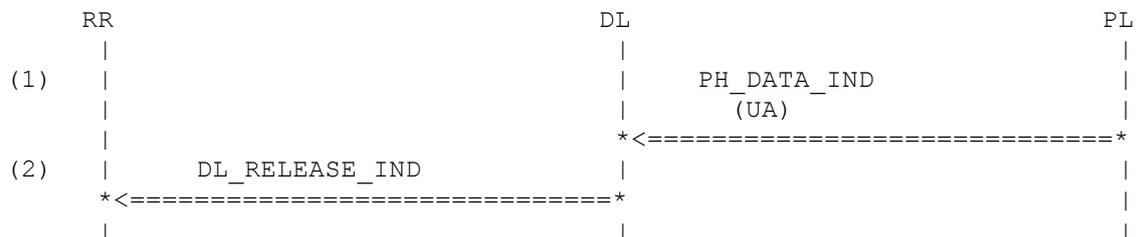
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_HO_RETRANSMIT
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_A_DL
(4) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_I06_UL
(7) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	RR_A_NR1_DL

History: 21.03.97 DL Initial

4.8.6 DL065: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL061



Parametrization

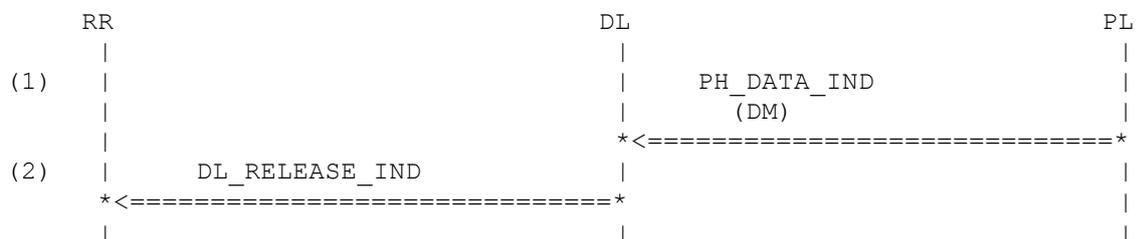
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.8.7 DL066: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL061



Parametrization

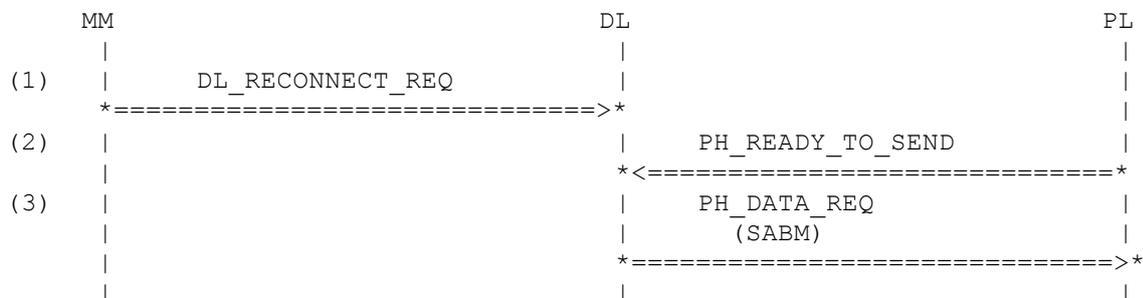
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.8.8 DL071: Initiate reconnection of multiframe operation

Description: Layer 3 (RR) requests reconnection by issuing a DL-RECONNECT request primitive containing a Layer-3 message (Assignment or Handover Failure). DL then issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL066



Parametrization

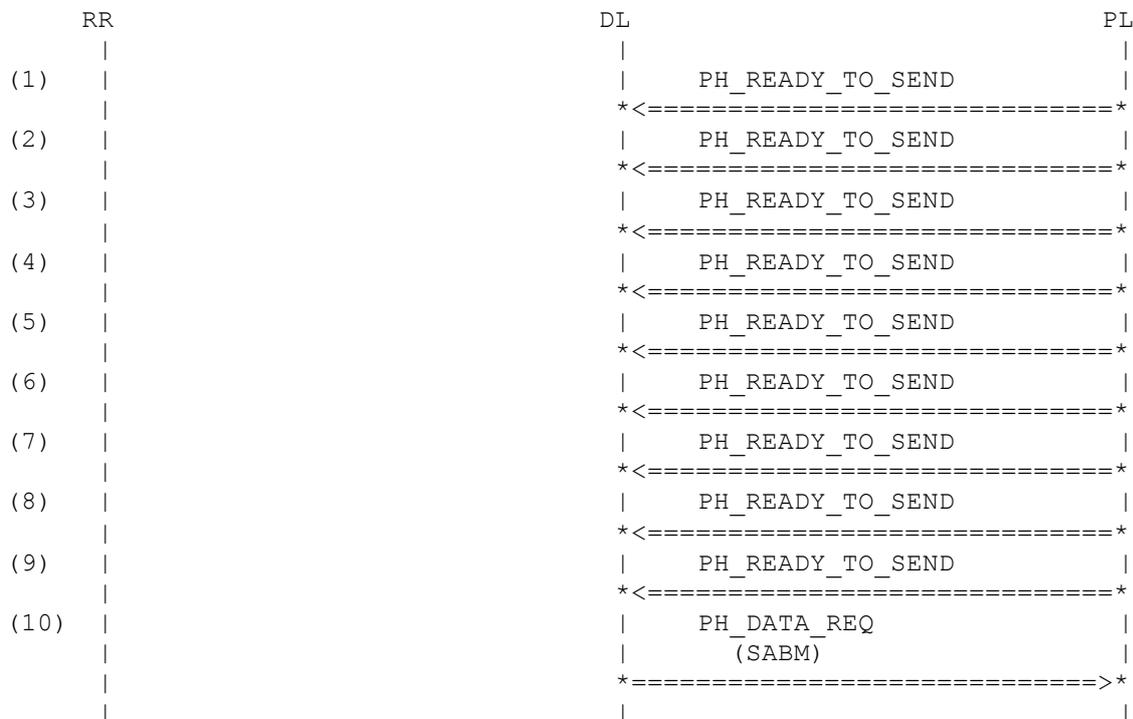
Primitive	Parameter	Value
(1) DL_RECONNECT_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 21.03.97 DL Initial

4.8.9 DL072: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit= 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL071



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ ch_type	CH_TYPE_FACCH

dummy
sdu

NOT_USED
SABM_A_UL

History: 21.03.97 DL Initial

4.8.10 DL073: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL071

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(SABM)	
		=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	
		<=====	

```
(26) | | PH_READY_TO_SEND |
      | | *<===== |
(27) | | PH_READY_TO_SEND |
      | | *<===== |
(28) | | PH_READY_TO_SEND |
      | | *<===== |
(29) | | PH_READY_TO_SEND |
      | | *<===== |
(30) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> |
(31) | | PH_READY_TO_SEND |
      | | *<===== |
(32) | | PH_READY_TO_SEND |
      | | *<===== |
(33) | | PH_READY_TO_SEND |
      | | *<===== |
(34) | | PH_READY_TO_SEND |
      | | *<===== |
(35) | | PH_READY_TO_SEND |
      | | *<===== |
(36) | | PH_READY_TO_SEND |
      | | *<===== |
(37) | | PH_READY_TO_SEND |
      | | *<===== |
(38) | | PH_READY_TO_SEND |
      | | *<===== |
(39) | | PH_READY_TO_SEND |
      | | *<===== |
(40) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> |
(41) | | PH_READY_TO_SEND |
      | | *<===== |
(42) | | PH_READY_TO_SEND |
      | | *<===== |
(43) | | PH_READY_TO_SEND |
      | | *<===== |
(44) | | PH_READY_TO_SEND |
      | | *<===== |
(45) | | PH_READY_TO_SEND |
      | | *<===== |
(46) | | PH_READY_TO_SEND |
      | | *<===== |
(47) | | PH_READY_TO_SEND |
      | | *<===== |
(48) | | PH_READY_TO_SEND |
      | | *<===== |
(49) | | PH_READY_TO_SEND |
      | | *<===== |
(50) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> |
(51) | | PH_READY_TO_SEND |
      | | *<===== |
(52) | | PH_READY_TO_SEND |
      | | *<===== |
(53) | | PH_READY_TO_SEND |
      | | *<===== |
(54) | | PH_READY_TO_SEND |
```

```

(55) |                                     *<=====
      |                                     | PH_READY_TO_SEND |
      |                                     *<=====
(56) |                                     | PH_READY_TO_SEND |
      |                                     *<=====
(57) |                                     | PH_READY_TO_SEND |
      |                                     *<=====
(58) |                                     | PH_READY_TO_SEND |
      |                                     *<=====
(59) |                                     | PH_READY_TO_SEND |
      |                                     *<=====
(60) | DL_RELEASE_IND |
      | *<=====
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_A_UL
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH

(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

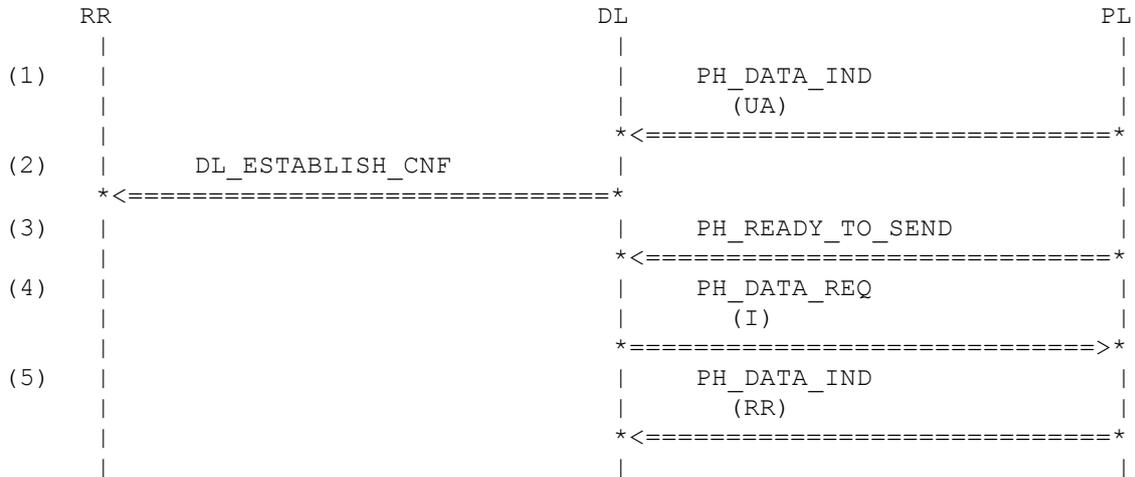
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
07.08.97 LE cs added for DL_RELEASE_IND
08.07.98 LE CONFIG removed

4.8.11 DL074: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the send sstate and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL071



Parametrization

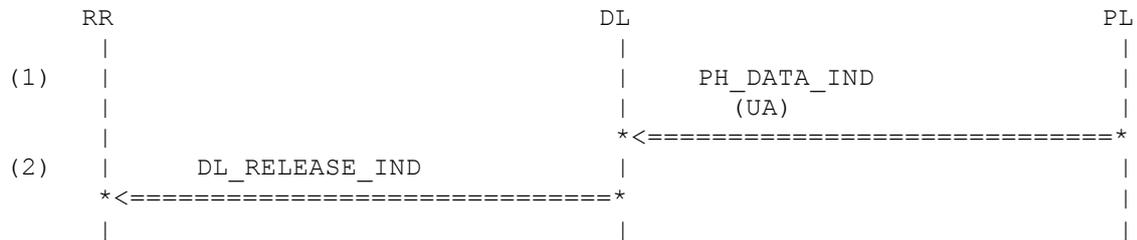
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL

History: 21.03.97 DL Initial

4.8.12 DL075: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set. DL remains in the idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL071



Parametrization

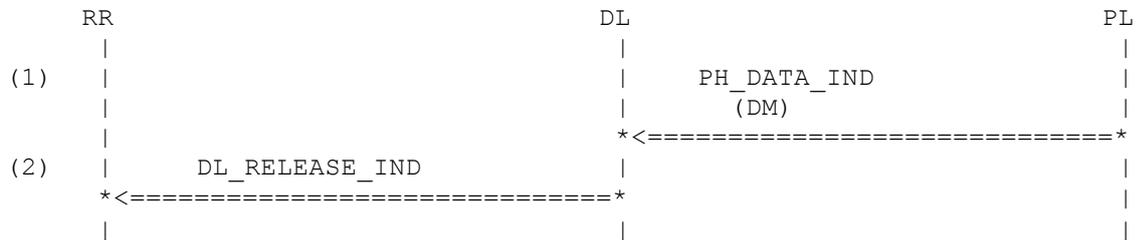
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.8.13 DL076: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL071



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

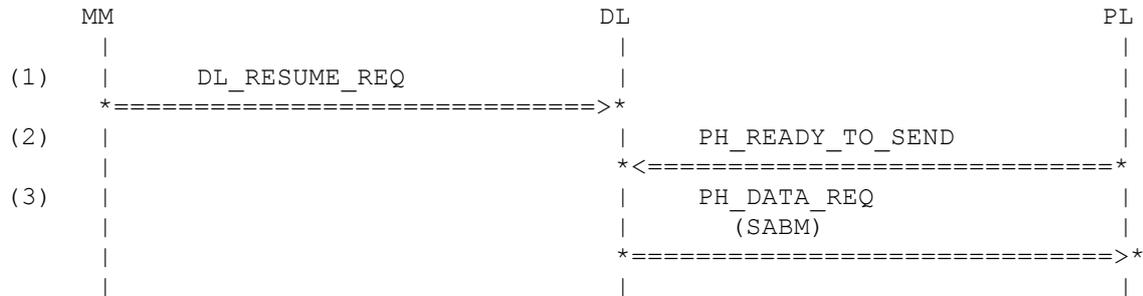
History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.9 Resumption of Multiframe Operation (SAPI 0, FACCH Halfrate)

4.9.1 DL461: Initiate resumption of multiframe operation

Description: Layer 3 (RR) requests resumption of multiple frame operation by issuing a DL-RESUME request primitive containing a Layer-3 message (Assignment or Handover Complete). DL issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3)

Preamble: DL041



Parametrization

Primitive	Parameter	Value
(1) DL_RESUME_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 08.07.98 LE Initial

4.9.2 DL462: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit= 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.4.3)

Preamble: DL461

RR	DL	PL
(1)	PH_READY_TO_SEND	
	<=====	
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ	
	(SABM)	
	=====>	

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR

dummy
sdu

NOT_USED
SABM_A_UL

History: 08.07.98 DL Initial

4.9.3 DL463: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.4.3)

Preamble: DL461

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(SABM)	
		=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	
		<=====	

```
(26) | | PH_READY_TO_SEND |
      | | *<===== *
(27) | | PH_READY_TO_SEND |
      | | *<===== *
(28) | | PH_READY_TO_SEND |
      | | *<===== *
(29) | | PH_READY_TO_SEND |
      | | *<===== *
(30) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> *
(31) | | PH_READY_TO_SEND |
      | | *<===== *
(32) | | PH_READY_TO_SEND |
      | | *<===== *
(33) | | PH_READY_TO_SEND |
      | | *<===== *
(34) | | PH_READY_TO_SEND |
      | | *<===== *
(35) | | PH_READY_TO_SEND |
      | | *<===== *
(36) | | PH_READY_TO_SEND |
      | | *<===== *
(37) | | PH_READY_TO_SEND |
      | | *<===== *
(38) | | PH_READY_TO_SEND |
      | | *<===== *
(39) | | PH_READY_TO_SEND |
      | | *<===== *
(40) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> *
(41) | | PH_READY_TO_SEND |
      | | *<===== *
(42) | | PH_READY_TO_SEND |
      | | *<===== *
(43) | | PH_READY_TO_SEND |
      | | *<===== *
(44) | | PH_READY_TO_SEND |
      | | *<===== *
(45) | | PH_READY_TO_SEND |
      | | *<===== *
(46) | | PH_READY_TO_SEND |
      | | *<===== *
(47) | | PH_READY_TO_SEND |
      | | *<===== *
(48) | | PH_READY_TO_SEND |
      | | *<===== *
(49) | | PH_READY_TO_SEND |
      | | *<===== *
(50) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====> *
(51) | | PH_READY_TO_SEND |
      | | *<===== *
(52) | | PH_READY_TO_SEND |
      | | *<===== *
(53) | | PH_READY_TO_SEND |
      | | *<===== *
(54) | | PH_READY_TO_SEND |
```

```

(55) | | | *<=====
      | | | | PH_READY_TO_SEND |
      | | | *<=====
(56) | | | | PH_READY_TO_SEND |
      | | | *<=====
(57) | | | | PH_READY_TO_SEND |
      | | | *<=====
(58) | | | | PH_READY_TO_SEND |
      | | | *<=====
(59) | | | | PH_READY_TO_SEND |
      | | | *<=====
(60) | | DL_RELEASE_IND |
      | *<=====
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_A_UL
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(13)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(14)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(16)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR

(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

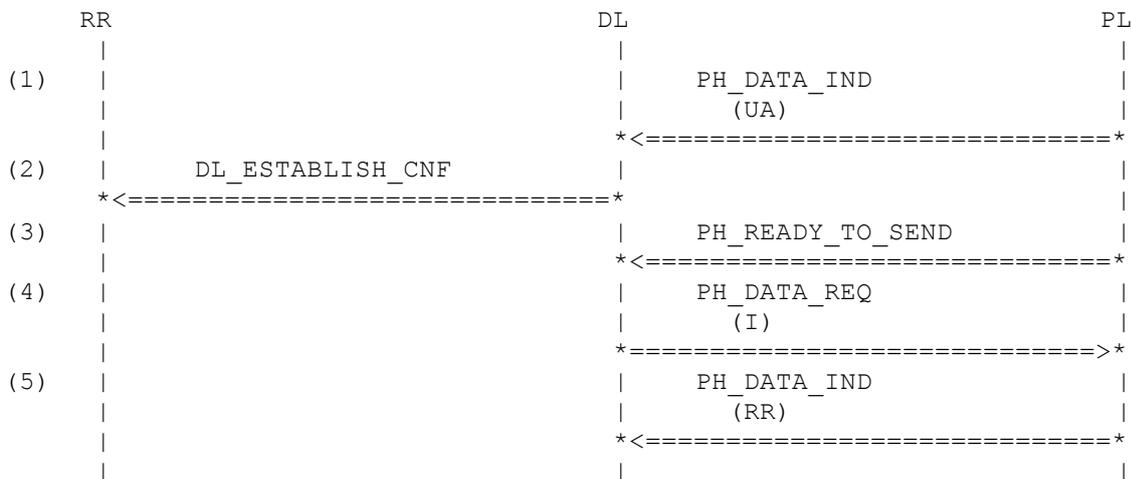
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.9.4 DL464: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the multiframe established state and issues a DL-RESUME confirmation primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL461



Parametrization

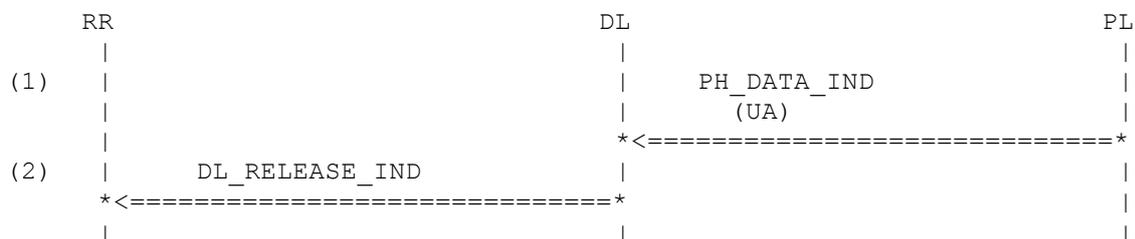
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED RR_A_NR1_DL

History: 08.07.98 DL Initial

4.9.5 DL465: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)

Preamble: DL461



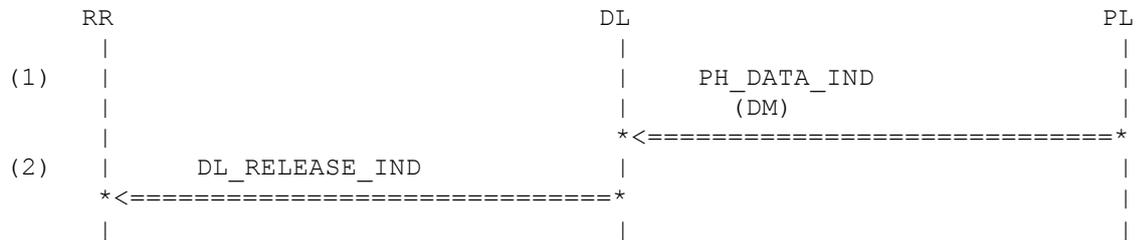
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.9.6 DL466: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the suspension state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.1.2)
Preamble: DL461



Parametrization

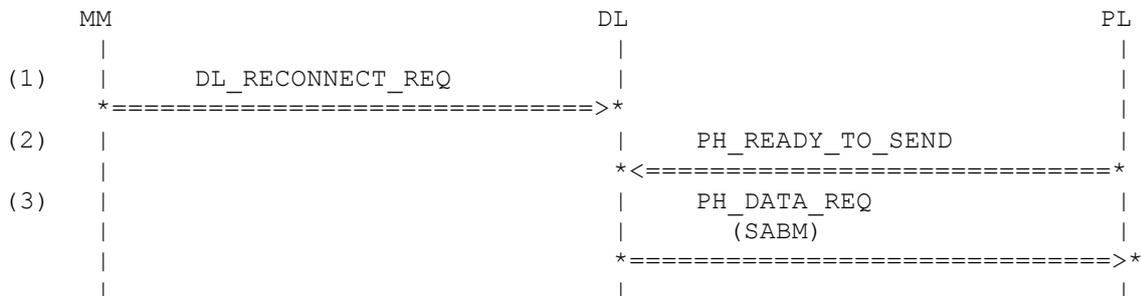
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.9.7 DL471: Initiate reconnection of multiframe operation

Description: Layer 3 (RR) requests reconnection by issuing a DL-RECONNECT request primitive containing a Layer-3 message (Assignment or Handover Failure). DL then issues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL466



Parametrization

Primitive	Parameter	Value
(1) DL_RECONNECT_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_A_UL

History: 08.07.98 LE Initial

4.9.8 DL472: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL471

RR	DL	PL
(1)	PH_READY_TO_SEND	
	<=====	
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ	
	(SABM)	
	=====>	

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR

dummy
sdu

NOT_USED
SABM_A_UL

History: 08.07.98 LE Initial

4.9.9 DL473: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL471

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(SABM)	
		=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	
		<=====	

```
(26) | | PH_READY_TO_SEND |
      | | *<===== *
(27) | | PH_READY_TO_SEND |
      | | *<===== *
(28) | | PH_READY_TO_SEND |
      | | *<===== *
(29) | | PH_READY_TO_SEND |
      | | *<===== *
(30) | | PH_DATA_REQ |
      | | (SABM) |
      | | *===== > *
(31) | | PH_READY_TO_SEND |
      | | *<===== *
(32) | | PH_READY_TO_SEND |
      | | *<===== *
(33) | | PH_READY_TO_SEND |
      | | *<===== *
(34) | | PH_READY_TO_SEND |
      | | *<===== *
(35) | | PH_READY_TO_SEND |
      | | *<===== *
(36) | | PH_READY_TO_SEND |
      | | *<===== *
(37) | | PH_READY_TO_SEND |
      | | *<===== *
(38) | | PH_READY_TO_SEND |
      | | *<===== *
(39) | | PH_READY_TO_SEND |
      | | *<===== *
(40) | | PH_DATA_REQ |
      | | (SABM) |
      | | *===== > *
(41) | | PH_READY_TO_SEND |
      | | *<===== *
(42) | | PH_READY_TO_SEND |
      | | *<===== *
(43) | | PH_READY_TO_SEND |
      | | *<===== *
(44) | | PH_READY_TO_SEND |
      | | *<===== *
(45) | | PH_READY_TO_SEND |
      | | *<===== *
(46) | | PH_READY_TO_SEND |
      | | *<===== *
(47) | | PH_READY_TO_SEND |
      | | *<===== *
(48) | | PH_READY_TO_SEND |
      | | *<===== *
(49) | | PH_READY_TO_SEND |
      | | *<===== *
(50) | | PH_DATA_REQ |
      | | (SABM) |
      | | *===== > *
(51) | | PH_READY_TO_SEND |
      | | *<===== *
(52) | | PH_READY_TO_SEND |
      | | *<===== *
(53) | | PH_READY_TO_SEND |
      | | *<===== *
(54) | | PH_READY_TO_SEND |
```


(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

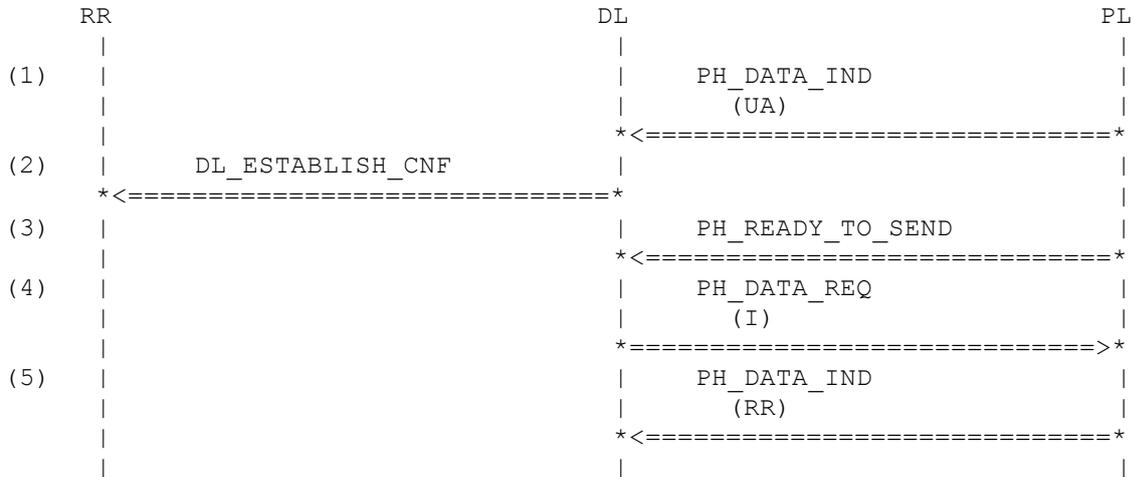
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.9.10 DL474: UA response from Base Station - information field empty

Description: A UA response with the F bit set to 1 and an empty information field. DL enters the send state and issues a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL471



Parametrization

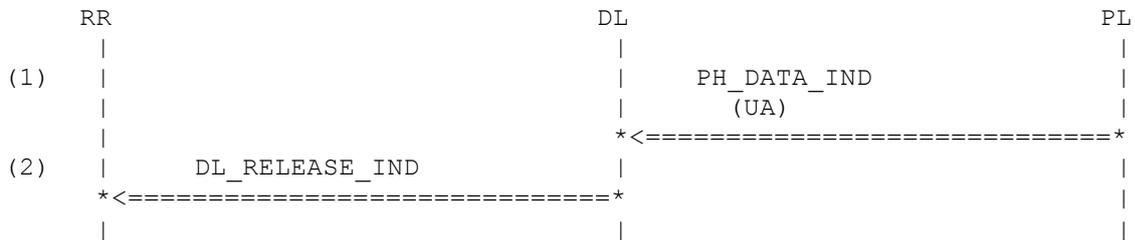
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED UA_A_DL
(2) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED I_I06_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED RR_A_NR1_DL

History: 08.07.98 LE Initial

4.9.11 DL475: UA response from Base Station - information field set

Description: A UA response with the F bit set to 1 and the information field set. DL remains in the idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL471



Parametrization

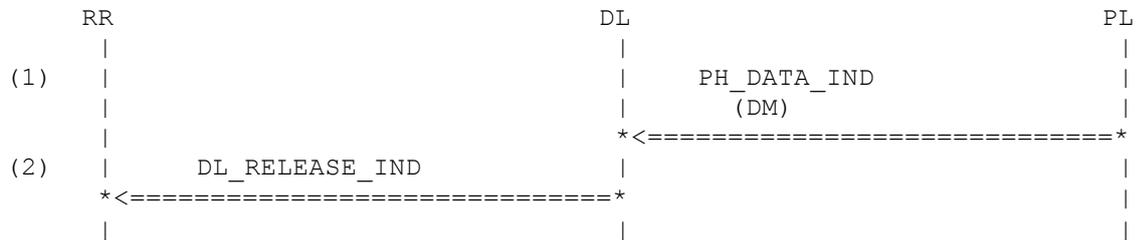
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_B_I06_2_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.9.12 DL476: DM response from Base Station - establishment rejected

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL remains in the idle state and issues a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL471



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

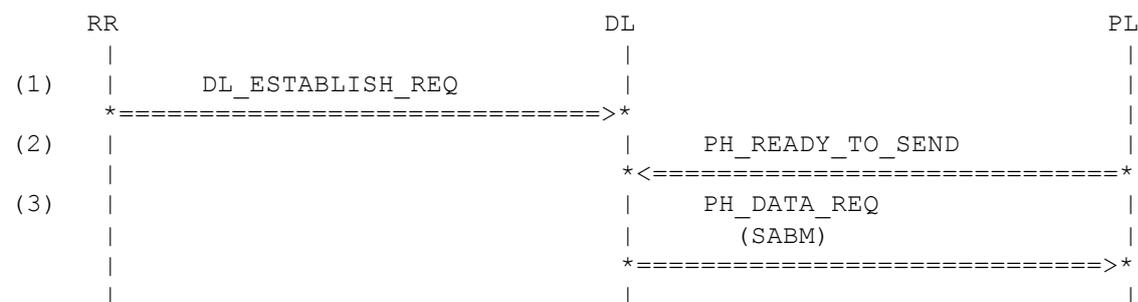
History: 08.07.98 LE Initial

4.10 Normal Establishment (SAPI 3)

4.10.1 DL081: Initiation by Radio Resource

Description: Once a SAPI 0 connection has been established, the MS (RR) initiates normal establishment (SAPI 3) by sending a DL-ESTABLISHMENT request primitive containing no Layer-3 message to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then forwards the request in a SABM frame with the P-bit set to 1.

Preamble: DL045



Parametrization

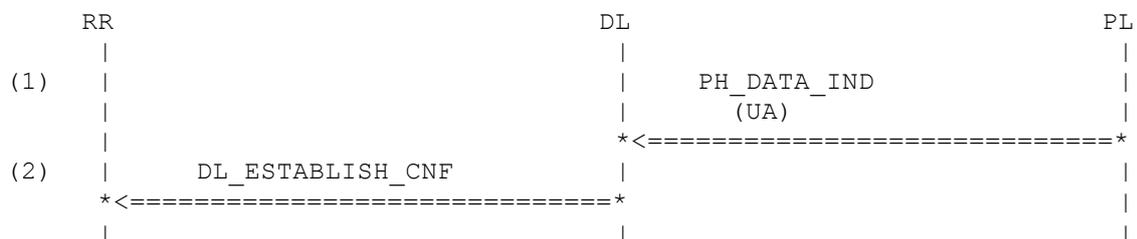
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_S3_UL

History: 21.03.97 DL Initial

4.10.2 DL082: UA response received from Base Station (information field empty)

Description: PL sends DL a UA response in which the F bit set to 1 and the information field is empty. DL enters the multi-frame-established state and issues a DL-ESTABLISH confirmation primitive to RR.

Preamble: DL081



Parametrization

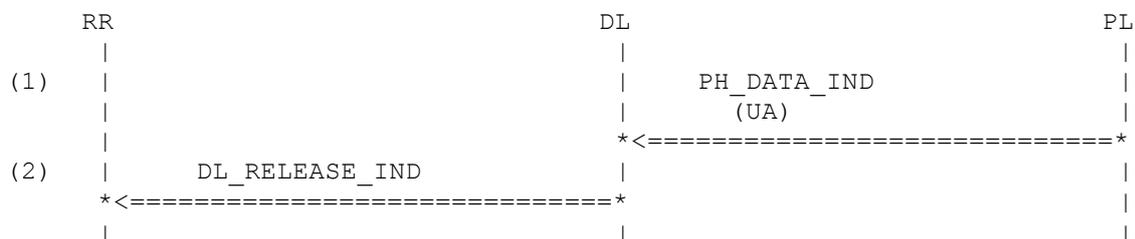
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_A_S3_SD_DL
(2) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_3

History: 21.03.97 DL Initial

4.10.3 DL083: UA response received from Base Station (information field not empty)

Description: PL sends DL a UA response in which the F bit is set to 1 and the information field is not empty. DL remains in the Idle state and issues a DL-RELEASE indication primitive to RR.

Preamble: DL081



Parametrization

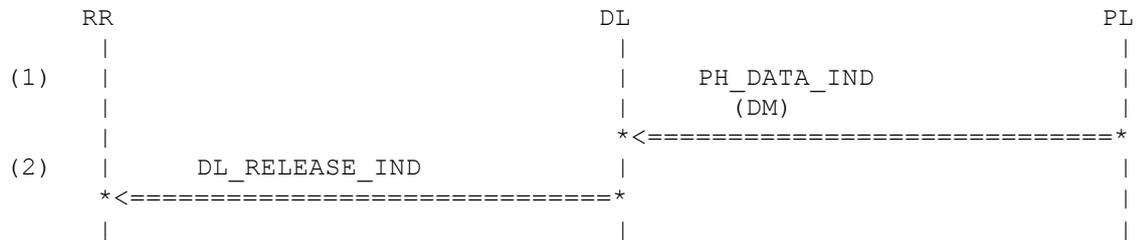
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_S3_I04_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_3
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.10.4 DL084: DM response received from Base Station -

Description: A DM response with the F bit set to 1 to indicate that the peer entity has rejected establishment. DL enters the Idle state and issues a DL-RELEASE indication primitive to RR.

Preamble: DL081



Parametrization

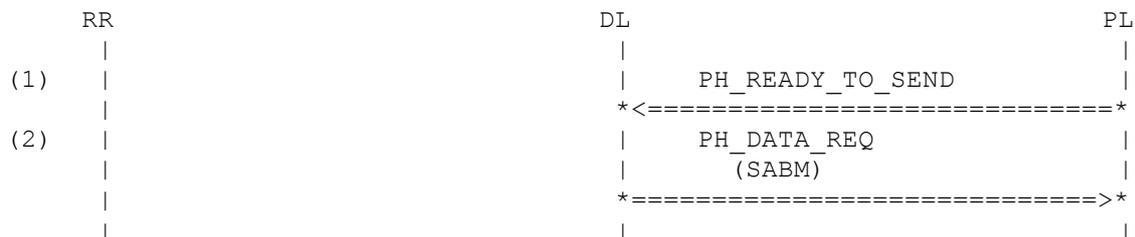
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_S3_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_3
	cs	NOT_PRESENT_8BIT

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.10.5 DL085: Expiry of T200 timer - maximum number of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a PH-DATA request primitive.

Preamble: DL081



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_S3_UL

History: 21.03.97 DL Initial

4.10.6 DL086: Expiry of T200 timer - maximum number of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer 'T200 expired N200+ 1 times.“

Preamble: DL081

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_DATA_REQ	
		(SABM)	
		=====>	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_DATA_REQ	
		(SABM)	
		=====>	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_DATA_REQ	
		(SABM)	
		=====>	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_DATA_REQ	
		(SABM)	
		=====>	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(SABM)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)	DL_RELEASE_IND		
	<=====		
(13)		PH_DATA_REQ	
		(empty frame)	
		=====>	

Parametrization

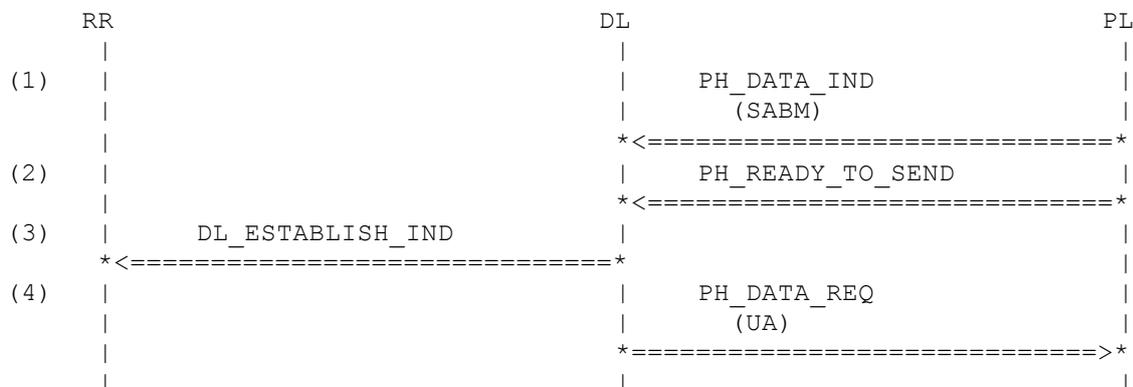
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_S3_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_S3_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_S3_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_S3_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_S3_UL
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(12)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_SDCCH SAPI_3 NOT_PRESENT_8BIT
(13)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND
 24.09.98 VK remove CONFIG

4.10.7 DL087: Initiation by the Base Station - MS able to accept establishment

Description: The SAPI 3 link has been established by the base station. A PH-DATA indication primitive is sent to DL. DL is capable of accepting the establishment, waits for PH-READY-TO-SEND primitive from PL and then forwards the response in a UA frame with the P-bit set to 1.

Preamble: DL012



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SD_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) DL_ESTABLISH_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_A_S3_UL

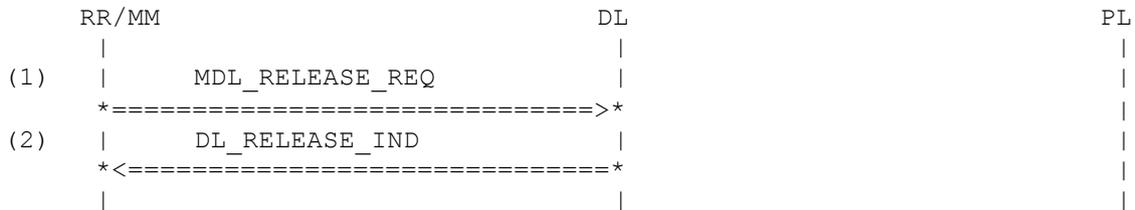
History: 21.03.97 DL Initial

4.11 Release (SDCCH)

4.11.1 DL091: Local end release

Description: DL receives a MDL-RELEASE request primitive from Layer 3 (MM). DL then ISSUES A DL-RELEASE indication primitive to RR and enters the idle state. (Ref. 5.4.4.4)

Preamble: DL012



Parametrization

Primitive	Parameter	Value
(1) MDL_RELEASE_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

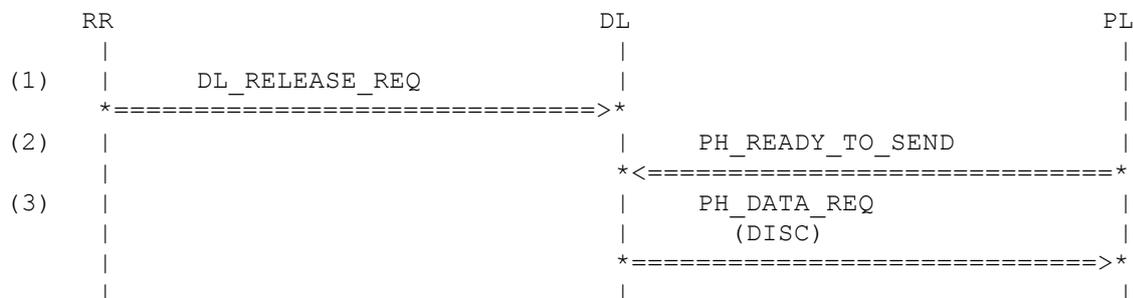
History: 21.03.97 DL Initial

07.08.97 LE cs added for DL_RELEASE_IND

4.11.2 DL092: Normal release initiated by MS (SDCCH)

Description: DL receives a DL-RELEASE request primitive from Layer 3 (MM). DL awaits receipt of PH-READY-TO-SEND primitive from PL and then issues a DISC command (P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. 5.4.4.2)

Preamble: DL012



Parametrization

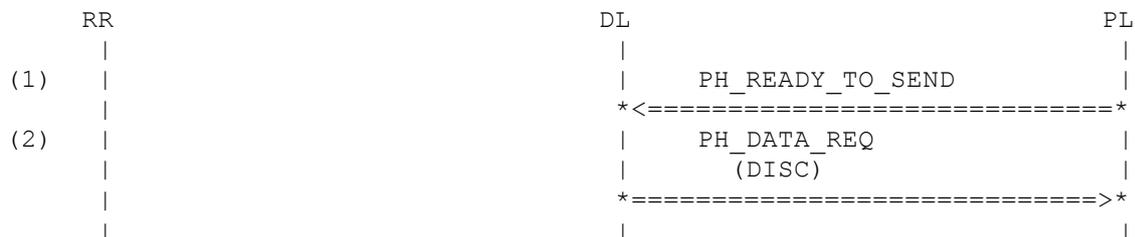
Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_UL

History: 21.03.97 DL Initial

4.11.3 DL093: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL awaits a PH-READY-TO-SEND primitive before reissuing a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL092



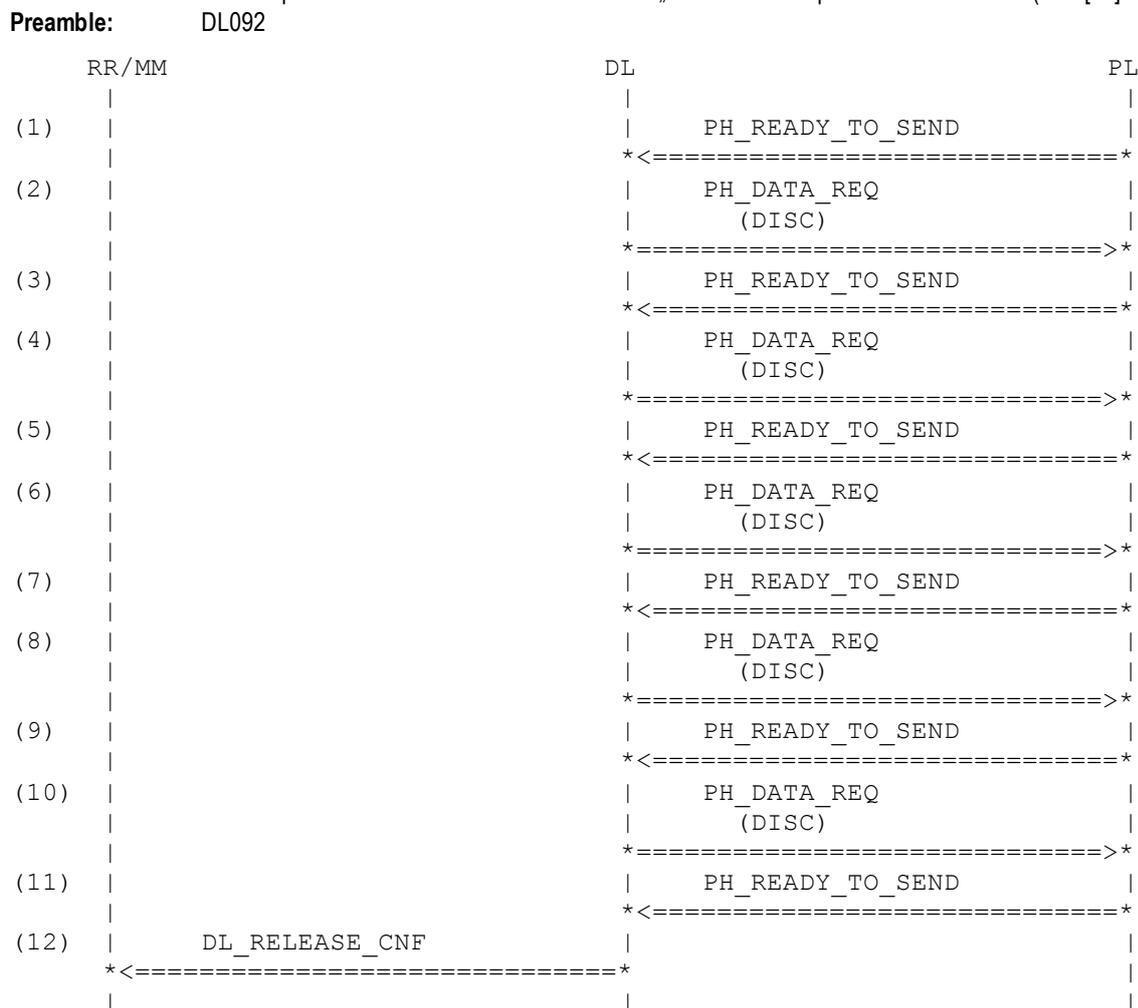
Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sdu	DISC_A_UL

History: 21.03.97 DL Initial

4.11.4 DL094: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer ‘T200 expired N200+ 1 times.“ (Ref. [20] 5.4.3.3.2) DL092



Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(2) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH

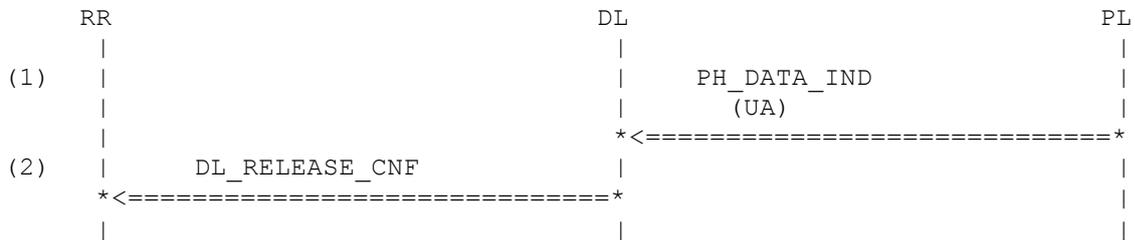
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) DL_RELEASE_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0

History: 21.03.97 DL Initial
24.09.98 VK CONFIG removed

4.11.5 DL095: Response from base station

Description: DL receives a UA response from the base station (F bit = 0) and then issues a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL092



Parametrization

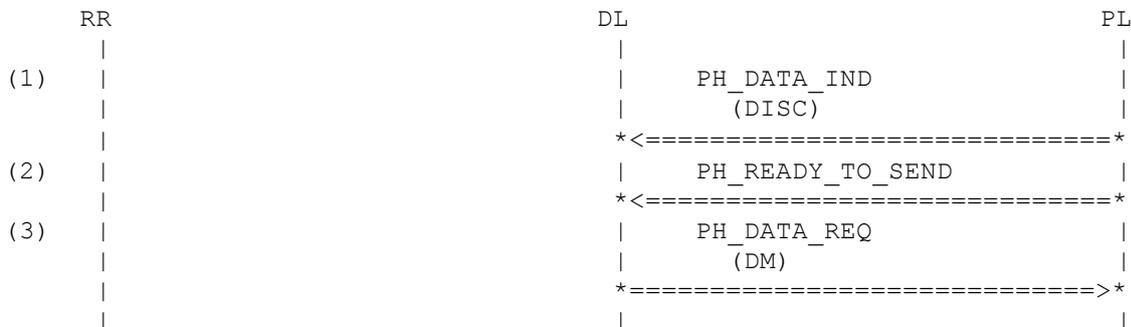
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_RELEASE_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0

History: 21.03.97 DL Initial

4.11.6 DL096: Release initiation by base station - MS in idle state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive and then issues a DM response in the form of a PH-DATA request primitive. (Ref. [20] 5.4.4.2)

Preamble: DL095



Parametrization

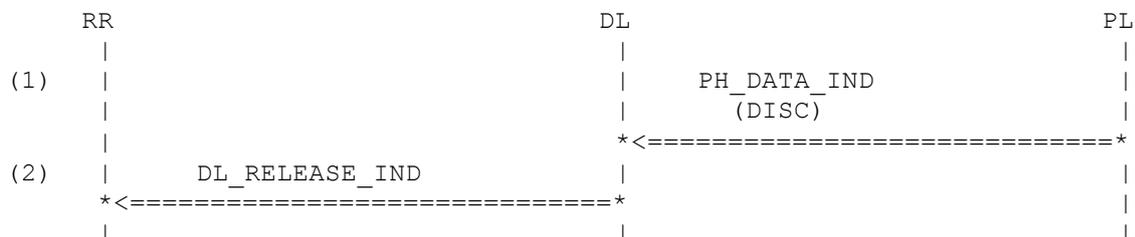
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_UL

History: 21.03.97 DL Initial

4.11.7 DL097: BS initiates release - MS in multiple frame established state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive, issues a UA response in the form of a PH-DATA request primitive and sends a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.4.2)

Preamble: DL042



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

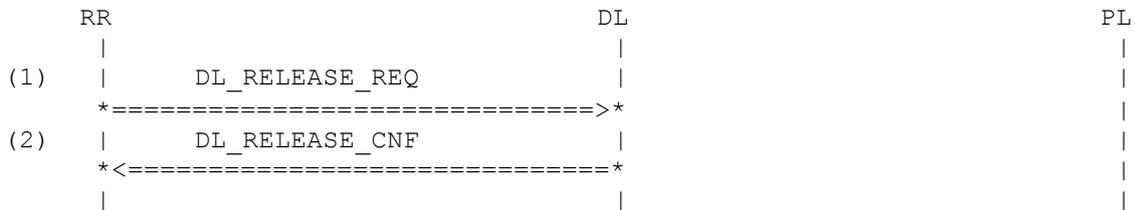
History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.12 Release (FACCH fullrate)

4.12.1 DL101: Normal release initiated by MS (idle state)

Description: DL receives a DL-RELEASE request primitive from Layer 3 (MM). DL then issues a DISC command (P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. 5.4.4.2)

Preamble: DL000



Parametrization

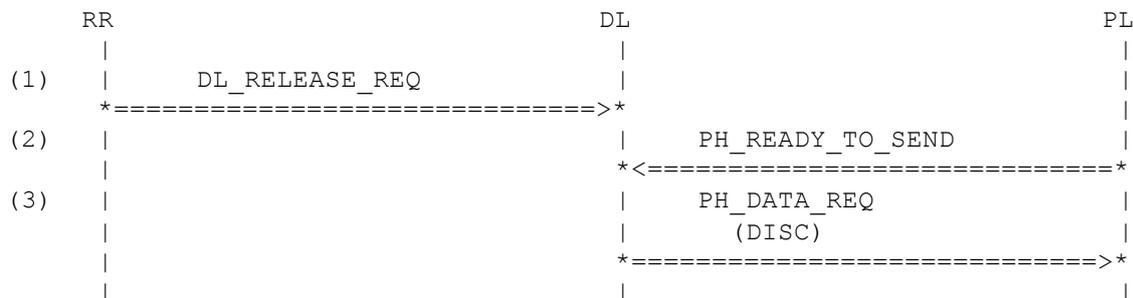
Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type sapi	CH_TYPE_FACCH SAPI_0
(2) DL_RELEASE_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0

History: 21.03.97 DL Initial

4.12.2 DL102: Normal release initiated by MS (multiframe established state)

Description: DL receives a DL-RELEASE request primitive from Layer 3 (MM). DL then issues a DISC command (P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. 5.4.4.2)

Preamble: DL064



Parametrization

Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_UL

History: 21.03.97 DL Initial

4.12.3 DL103: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit= 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL102

RR	DL	PL
(1)	PH_READY_TO_SEND	
	<=====	
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ	
	(DISC)	
	=====>	

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ ch_type	CH_TYPE_FACCH

dummy
sdu

NOT_USED
DISC_A_UL

History: 21.03.97 DL Initial

4.12.4 DL104: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer ‘T200 expired N200+ 1 times.“ (Ref. [20] 5.4.3.3.2) DL102

Preamble:

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(DISC)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(DISC)	
		=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	

```
|
|
(26) | | PH_READY_TO_SEND |
| |
| |
(27) | | PH_READY_TO_SEND |
| |
| |
(28) | | PH_READY_TO_SEND |
| |
| |
(29) | | PH_READY_TO_SEND |
| |
| |
(30) | | PH_DATA_REQ |
| | (DISC) |
| |
| |
(31) | | PH_READY_TO_SEND |
| |
| |
(32) | | PH_READY_TO_SEND |
| |
| |
(33) | | PH_READY_TO_SEND |
| |
| |
(34) | | PH_READY_TO_SEND |
| |
| |
(35) | | PH_READY_TO_SEND |
| |
| |
(36) | | PH_READY_TO_SEND |
| |
| |
(37) | | PH_READY_TO_SEND |
| |
| |
(38) | | PH_READY_TO_SEND |
| |
| |
(39) | | PH_READY_TO_SEND |
| |
| |
(40) | | PH_DATA_REQ |
| | (DISC) |
| |
| |
(41) | | PH_READY_TO_SEND |
| |
| |
(42) | | PH_READY_TO_SEND |
| |
| |
(43) | | PH_READY_TO_SEND |
| |
| |
(44) | | PH_READY_TO_SEND |
| |
| |
(45) | | PH_READY_TO_SEND |
| |
| |
(46) | | PH_READY_TO_SEND |
| |
| |
(47) | | PH_READY_TO_SEND |
| |
| |
(48) | | PH_READY_TO_SEND |
| |
| |
(49) | | PH_READY_TO_SEND |
| |
| |
(50) | | PH_DATA_REQ |
| | (DISC) |
| |
| |
(51) | | PH_READY_TO_SEND |
| |
| |
(52) | | PH_READY_TO_SEND |
| |
| |
(53) | | PH_READY_TO_SEND |
| |
| |
```

```

(54) | | PH_READY_TO_SEND |
| | *<===== |
(55) | | PH_READY_TO_SEND |
| | *<===== |
(56) | | PH_READY_TO_SEND |
| | *<===== |
(57) | | PH_READY_TO_SEND |
| | *<===== |
(58) | | PH_READY_TO_SEND |
| | *<===== |
(59) | | PH_READY_TO_SEND |
| | *<===== |
(60) | DL_RELEASE_CNF |
| *<===== |
| | |

```

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)		PH_DATA_REQ	
		ch_type	CH_TYPE_FACCH
		dummy	NOT_USED
		sdu	DISC_A_UL
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DISC_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DISC_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DISC_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DISC_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

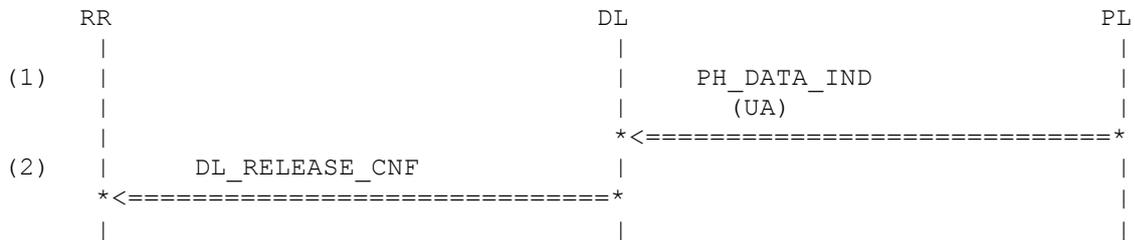
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	DL_RELEASE_CNF ch_type sapi	CH_TYPE_FACCH SAPI_0

History: 21.03.97 DL Initial
08.07.98 LE CONFIG removed

4.12.5 DL105: Response from base station

Description: DL receives a UA response from the base station (F bit = 0) and then issues a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL102



Parametrization

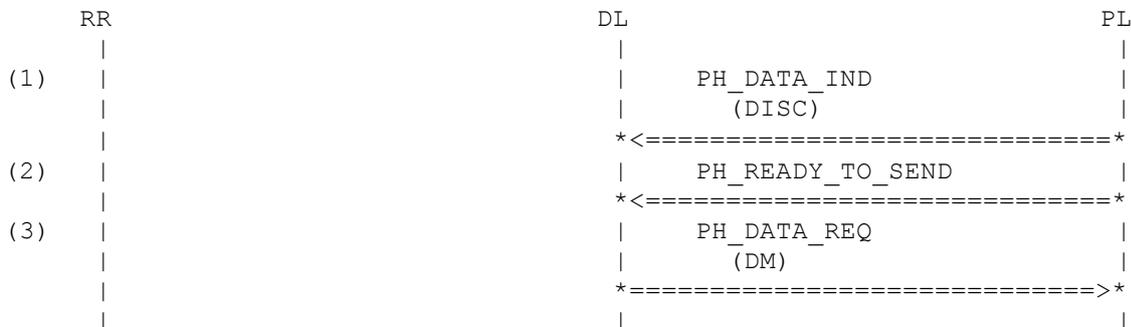
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_RELEASE_CNF	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0

History: 21.03.97 DL Initial

4.12.6 DL106: Release initiation by base station - MS in idle state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive and then issues a DM response in the form of a PH-DATA request primitive. (Ref. [20] 5.4.4.2)

Preamble: DL105



Parametrization

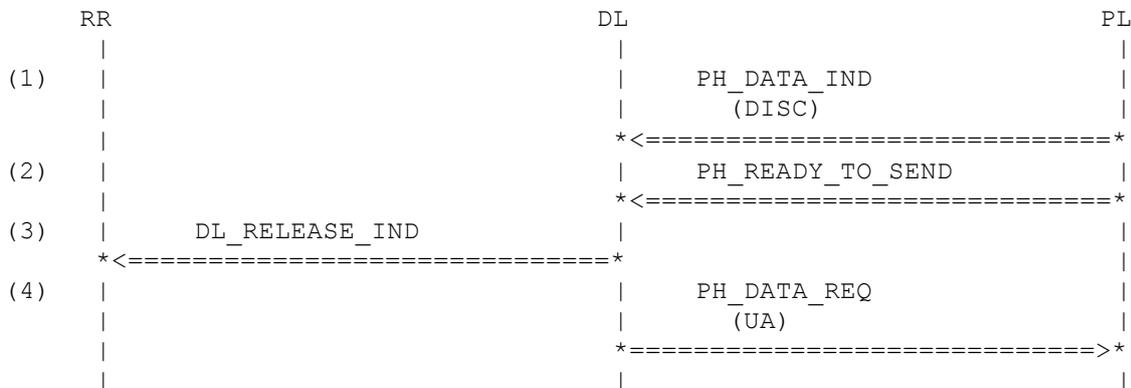
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_A_UL

History: 21.03.97 DL Initial

4.12.7 DL107: BS initiates release - MS in multiple frame established state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive, issues a UA response in the form of a PH-DATA request primitive and sends a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.4.2)

Preamble: DL064



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_A_UL

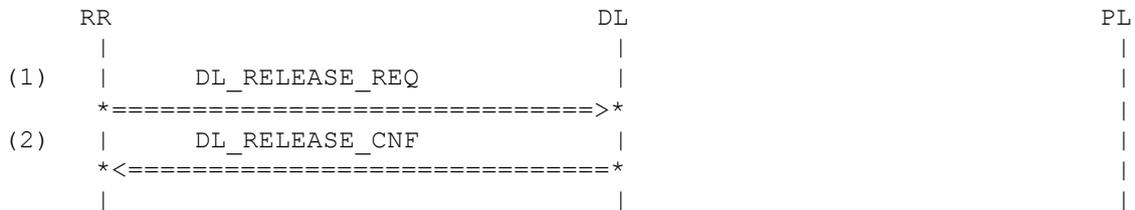
History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.13 Release (FACCH Halfrate)

4.13.1 DL501: Normal release initiated by MS (idle state)

Description: DL receives a DL-RELEASE request primitive from Layer 3 (MM). DL then issues a DISC command (P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. 5.4.4.2)

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0
(2) DL_RELEASE_CNF	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0

History: 08.07.98 LE Initial

4.13.3 DL503: Expiry of T200 timer - max. no. of retransmissions not reached

Description: Following expiry of the T200 timer, DL reissues a SABM command (with P bit = 1, length = 0) as part of a PH-DATA request primitive. (Ref. [20] 5.4.3.3.2)

Preamble: DL502

RR	DL	PL
(1)	PH_READY_TO_SEND	
	<=====	
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ	
	(DISC)	
	<=====>*	

Parametrization

Primitive	Parameter	Value
(1) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR

dummy
sdu

NOT_USED
DISC_A_UL

History: 08.07.98 LE Initial

4.13.4 DL504: Expiry of T200 timer - max. no. of retransmissions reached

Description: Following expiry of the T200 timer and the maximum number of transmissions having been reached, DL enters the idle state and issues a DL-RELEASE indication primitive to RR followed by a MDL-ERROR indication primitive to MM with error cause set to „Timer ‘T200 expired N200+ 1 times.“ (Ref. [20] 5.4.3.3.2) DL502

Preamble:

	RR/MM	DL	PL
(1)		PH_READY_TO_SEND	
		<=====	
(2)		PH_READY_TO_SEND	
		<=====	
(3)		PH_READY_TO_SEND	
		<=====	
(4)		PH_READY_TO_SEND	
		<=====	
(5)		PH_READY_TO_SEND	
		<=====	
(6)		PH_READY_TO_SEND	
		<=====	
(7)		PH_READY_TO_SEND	
		<=====	
(8)		PH_READY_TO_SEND	
		<=====	
(9)		PH_READY_TO_SEND	
		<=====	
(10)		PH_DATA_REQ	
		(DISC)	
		=====>	
(11)		PH_READY_TO_SEND	
		<=====	
(12)		PH_READY_TO_SEND	
		<=====	
(13)		PH_READY_TO_SEND	
		<=====	
(14)		PH_READY_TO_SEND	
		<=====	
(15)		PH_READY_TO_SEND	
		<=====	
(16)		PH_READY_TO_SEND	
		<=====	
(17)		PH_READY_TO_SEND	
		<=====	
(18)		PH_READY_TO_SEND	
		<=====	
(19)		PH_READY_TO_SEND	
		<=====	
(20)		PH_DATA_REQ	
		(DISC)	
		=====>	
(21)		PH_READY_TO_SEND	
		<=====	
(22)		PH_READY_TO_SEND	
		<=====	
(23)		PH_READY_TO_SEND	
		<=====	
(24)		PH_READY_TO_SEND	
		<=====	
(25)		PH_READY_TO_SEND	

```
(26) | | *<=====*
| | | PH_READY_TO_SEND |
| | | *<=====*
(27) | | | PH_READY_TO_SEND |
| | | *<=====*
(28) | | | PH_READY_TO_SEND |
| | | *<=====*
(29) | | | PH_READY_TO_SEND |
| | | *<=====*
(30) | | | PH_DATA_REQ |
| | | (DISC) |
| | | *=====>*
(31) | | | PH_READY_TO_SEND |
| | | *<=====*
(32) | | | PH_READY_TO_SEND |
| | | *<=====*
(33) | | | PH_READY_TO_SEND |
| | | *<=====*
(34) | | | PH_READY_TO_SEND |
| | | *<=====*
(35) | | | PH_READY_TO_SEND |
| | | *<=====*
(36) | | | PH_READY_TO_SEND |
| | | *<=====*
(37) | | | PH_READY_TO_SEND |
| | | *<=====*
(38) | | | PH_READY_TO_SEND |
| | | *<=====*
(39) | | | PH_READY_TO_SEND |
| | | *<=====*
(40) | | | PH_DATA_REQ |
| | | (DISC) |
| | | *=====>*
(41) | | | PH_READY_TO_SEND |
| | | *<=====*
(42) | | | PH_READY_TO_SEND |
| | | *<=====*
(43) | | | PH_READY_TO_SEND |
| | | *<=====*
(44) | | | PH_READY_TO_SEND |
| | | *<=====*
(45) | | | PH_READY_TO_SEND |
| | | *<=====*
(46) | | | PH_READY_TO_SEND |
| | | *<=====*
(47) | | | PH_READY_TO_SEND |
| | | *<=====*
(48) | | | PH_READY_TO_SEND |
| | | *<=====*
(49) | | | PH_READY_TO_SEND |
| | | *<=====*
(50) | | | PH_DATA_REQ |
| | | (DISC) |
| | | *=====>*
(51) | | | PH_READY_TO_SEND |
| | | *<=====*
(52) | | | PH_READY_TO_SEND |
| | | *<=====*
(53) | | | PH_READY_TO_SEND |
| | | *<=====*
```

```

(54) | | PH_READY_TO_SEND |
| | *<=====*
(55) | | PH_READY_TO_SEND |
| | *<=====*
(56) | | PH_READY_TO_SEND |
| | *<=====*
(57) | | PH_READY_TO_SEND |
| | *<=====*
(58) | | PH_READY_TO_SEND |
| | *<=====*
(59) | | PH_READY_TO_SEND |
| | *<=====*
(60) | DL_RELEASE_CNF |
| *<=====*
| |

```

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)		PH_DATA_REQ	
		ch_type	CH_TYPE_FACCH_HR
		dummy	NOT_USED
		sdu	DISC_A_UL
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR

(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED DISC_A_UL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED DISC_A_UL
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED DISC_A_UL
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED DISC_A_UL
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

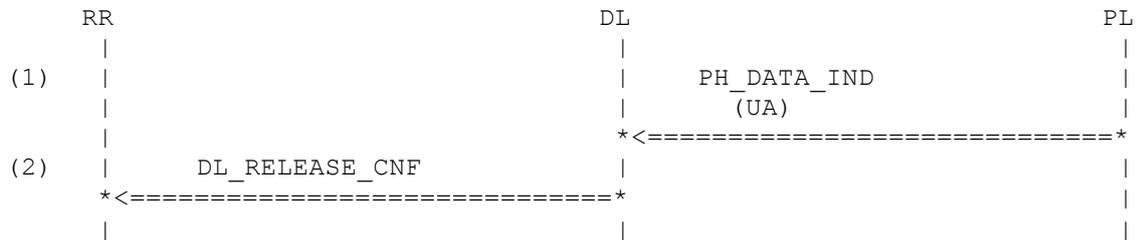
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	DL_RELEASE_CNF ch_type sapi	CH_TYPE_FACCH_HR SAPI_0

History: 21.03.97 DL Initial

4.13.5 DL505: Response from base station

Description: DL receives a UA response from the base station (F bit = 0) and then issues a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.3.3.2)

Preamble: DL502



Parametrization

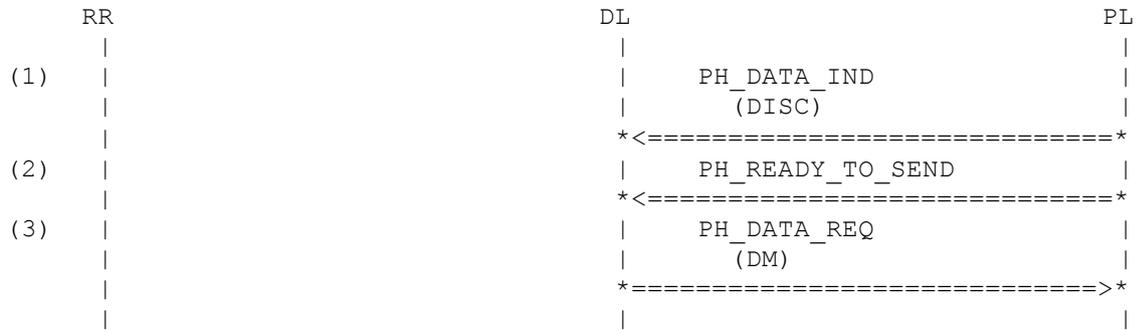
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_B_I06_1_DL
(2) DL_RELEASE_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0

History: 08.07.98 LE Initial

4.13.6 DL506: Release initiation by base station - MS in idle state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive and then issues a DM response in the form of a PH-DATA request primitive. (Ref. [20] 5.4.4.2)

Preamble: DL505



Parametrization

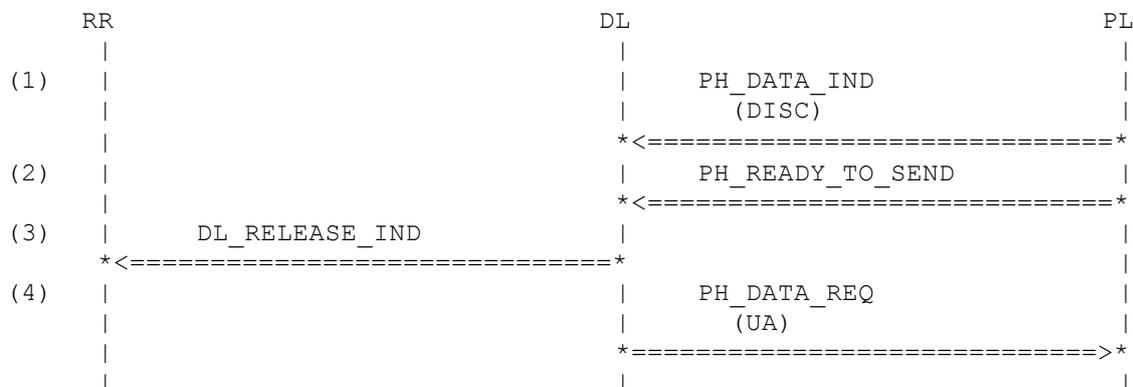
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_UL

History: 08.07.98 LE Initial

4.13.7 DL507: BS initiates release - MS in multiple frame established state

Description: DL receives a DISC command from the base station (F bit = 0) while in the idle state. It awaits a PH-READY-TO-SEND primitive, issues a UA response in the form of a PH-DATA request primitive and sends a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.4.2)

Preamble: DL464



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_A_UL

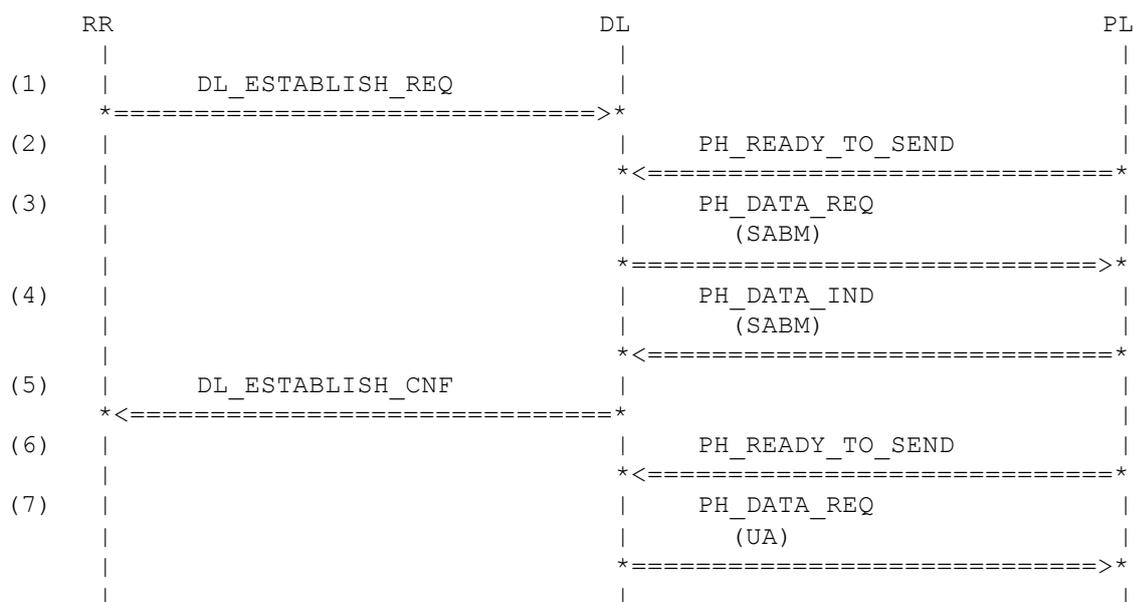
History: 08.07.98 LE Initial

4.14 Collision of Unnumbered Commands and Responses

4.14.1 DL111: Identical commands - establish (SDCCH)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing the same unnumbered SABM command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

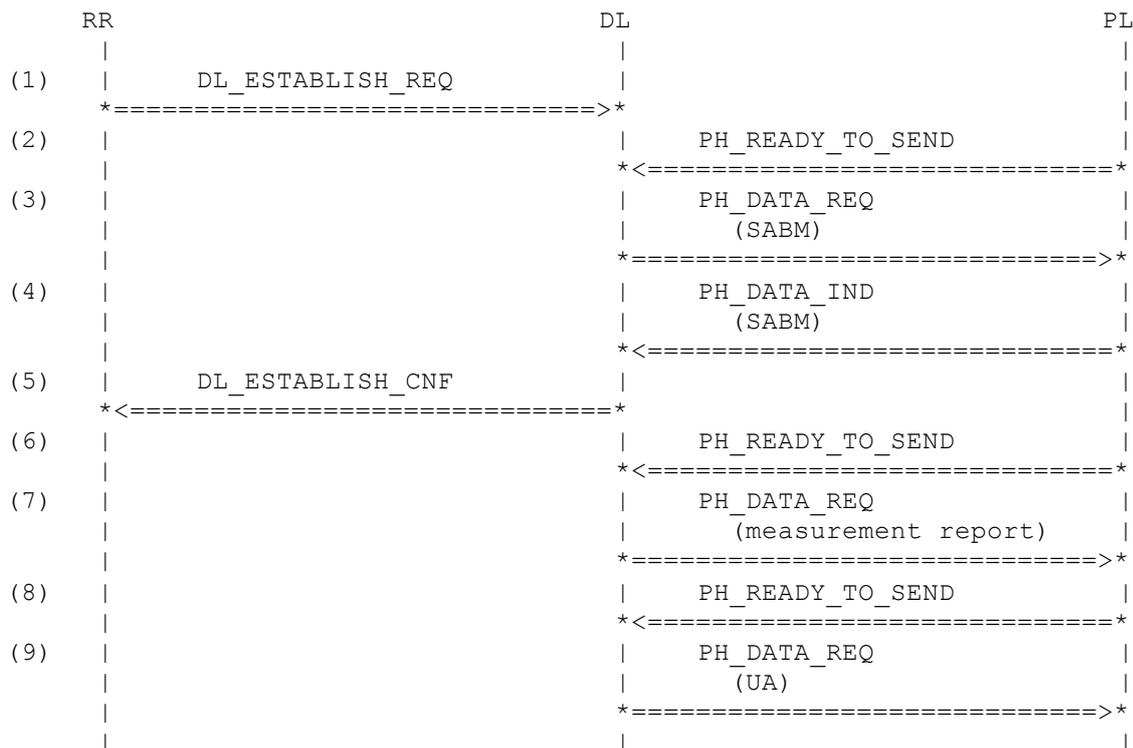
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_B_I06_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_A_UL

History: 21.03.97 DL Initial

4.14.2 DL112: Identical commands - establish (SACCH)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing the same unnumbered SABM command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

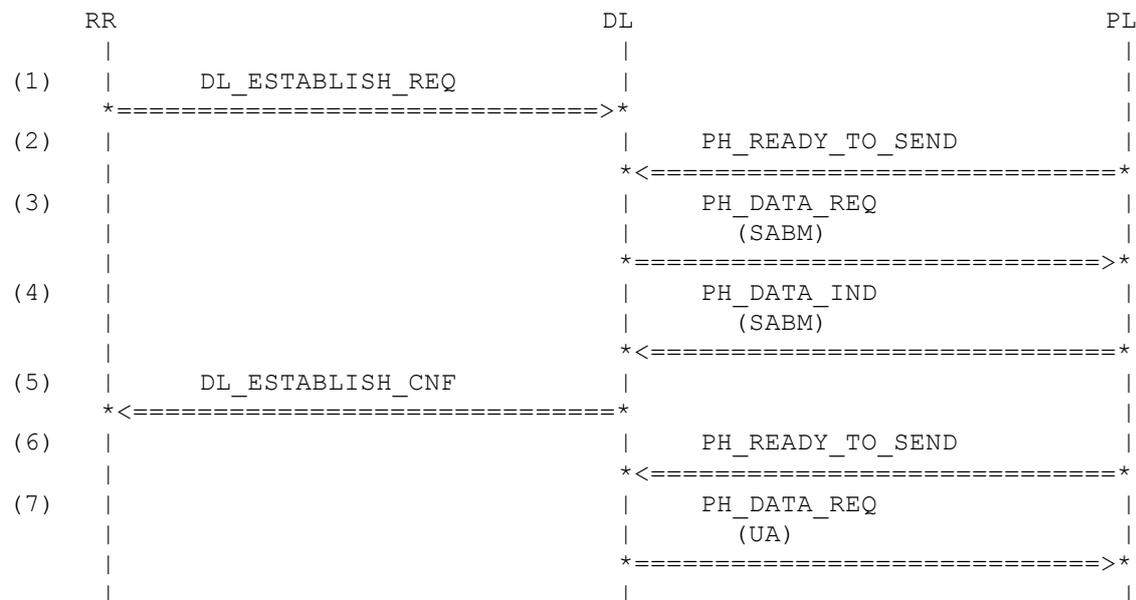
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SA_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SA_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UI_FRAME_UP_SACCH_M2
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(9) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UA_A_S3_SA_UL

History: 21.03.97 DL Initial

4.14.3 DL113: Identical commands - establish (FACCH Fullrate)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing the same unnumbered SABM command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

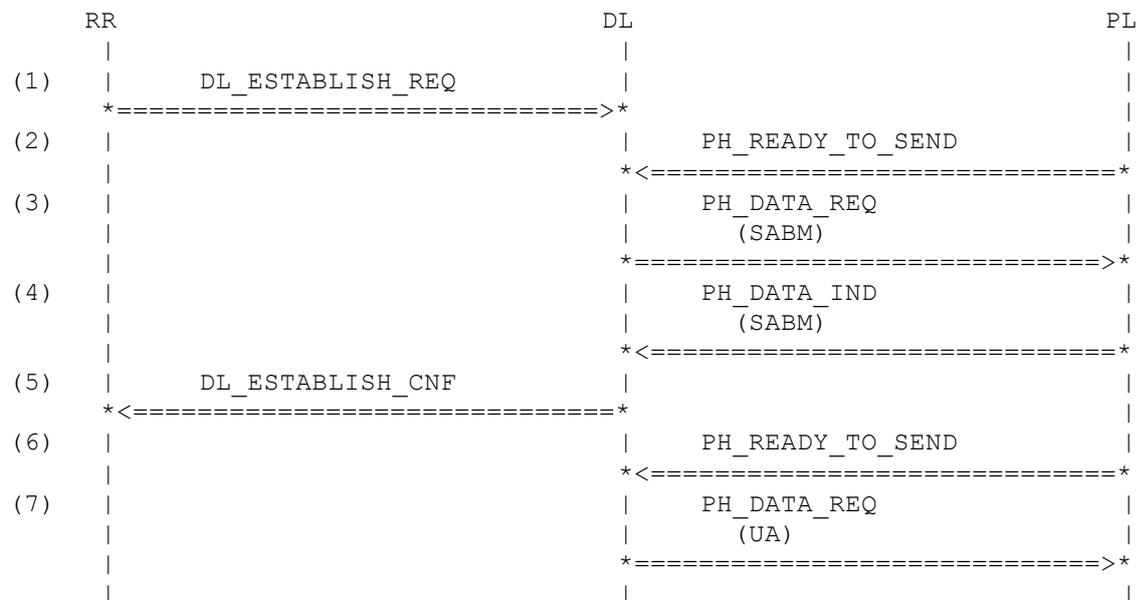
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_B_I06_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_A_UL

History: 21.03.97 DL Initial

4.14.4 DL513: Identical commands - establish (FACCH Halfrate)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing the same unnumbered SABM command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

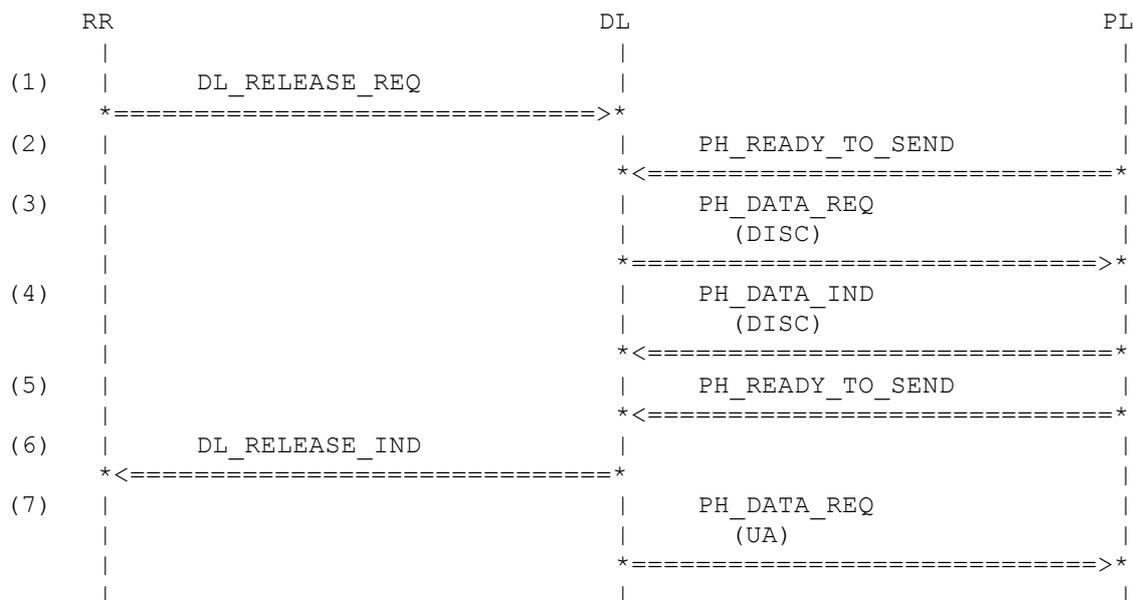
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_DL
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_A_UL

History: 08.07.98 LE Initial

4.14.5 DL114: Identical commands - release (SDCCH)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the unnumbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing the same unnumbered DISC command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL012



Parametrization

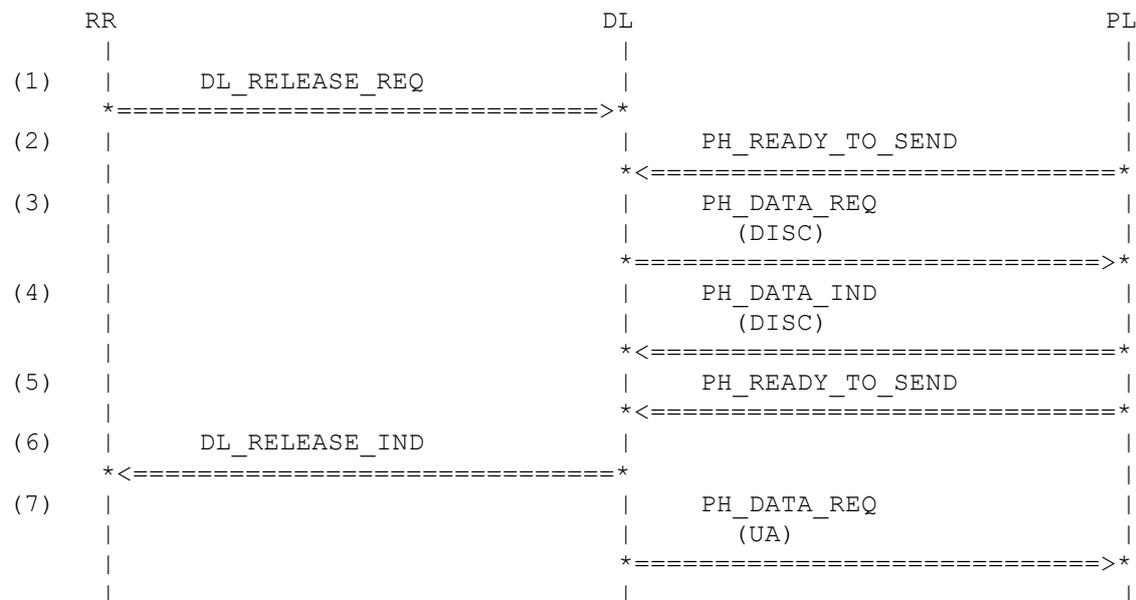
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_RELEASE_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_UL

History: 21.03.97 DL Initial

4.14.6 DL115: Identical commands - release (FACCH Fullrate)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing the same unnumbered DISC command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

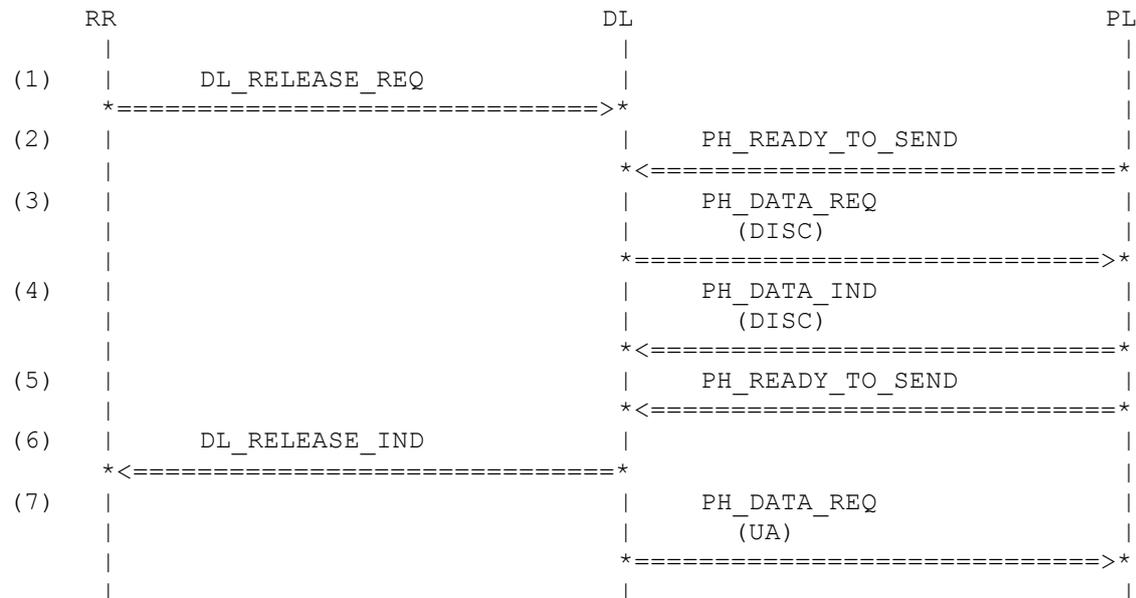
Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_A_UL

History: 21.03.97 DL Initial

4.14.7 DL515: Identical commands - release (FACCH Halfrate)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing the same unnumbered DISC command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports successful conclusion by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

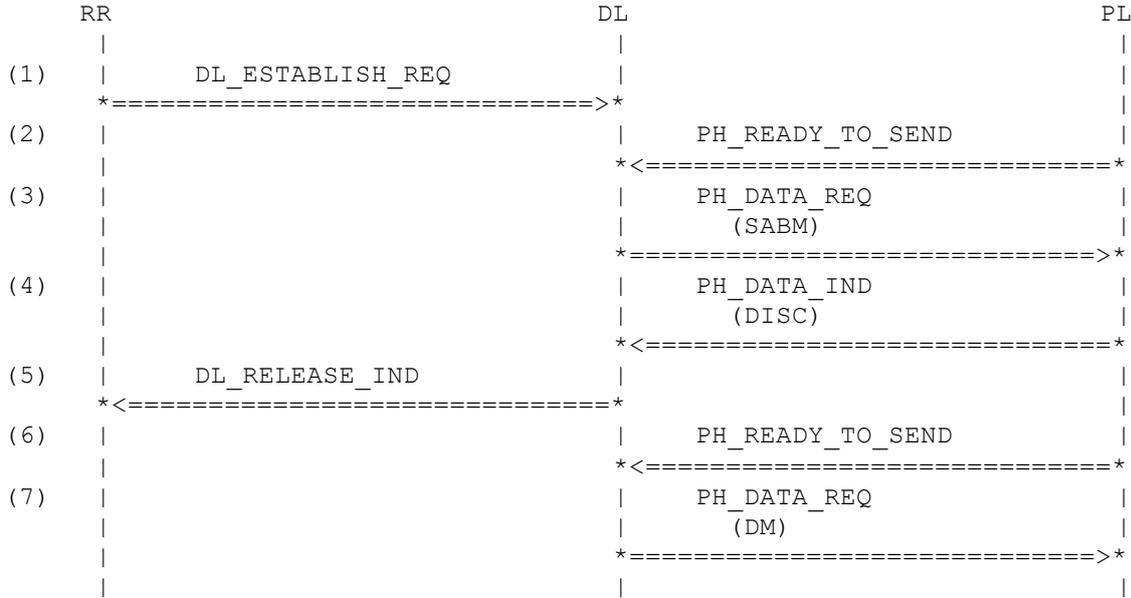
Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_A_UL

History: 08.07.98 LE Initial

4.14.8 DL116: Different commands - establish (SDCCH)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the un-numbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing a SABM command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a DM response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

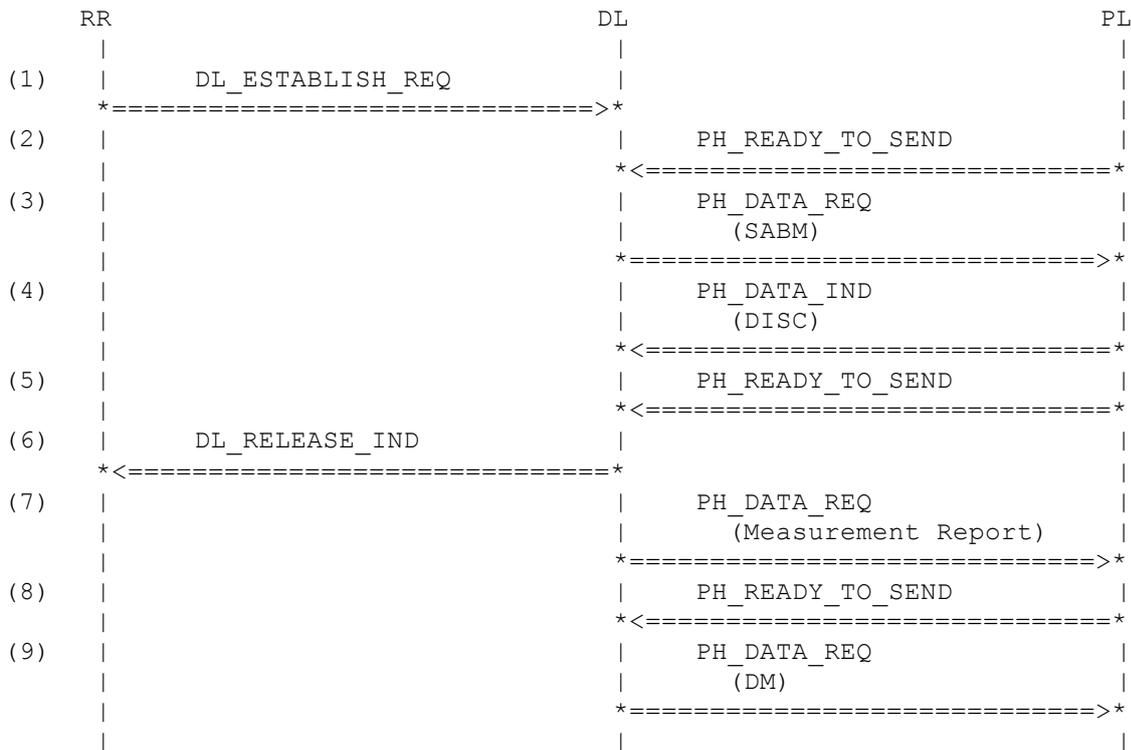
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(5) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_A_UL

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.14.9 DL117: Different commands - establish (SACCH)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the un-numbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing a SABM command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a DM response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-RELEASE indication primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L0_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	SABM_A_S3_SA_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	DISC_A_S3_UL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(6) DL_RELEASE_IND	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	cs	NOT_PRESENT_8BIT
(7) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UI_FRAME_UP_SACCH_M2
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(9) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	DM_A_S3_SA_DL

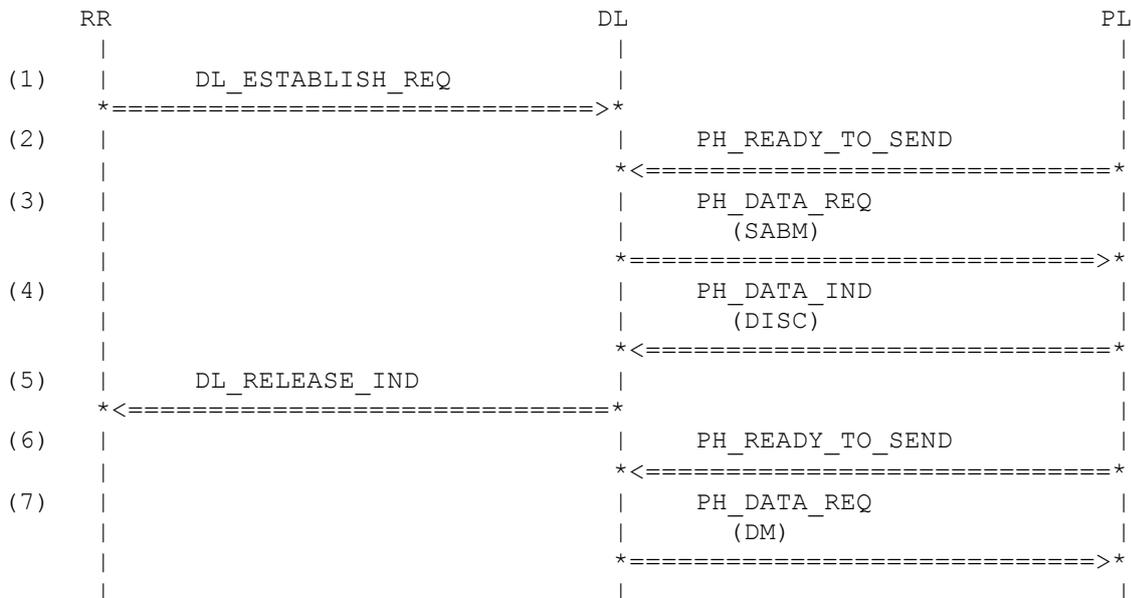
History: 21.03.97 DL Initial

07.08.97 LE cs added for DL_RELEASE_IND

4.14.10 DL118: Different commands - establish (FACCH Fullrate)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing a different unnumbered SABM command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

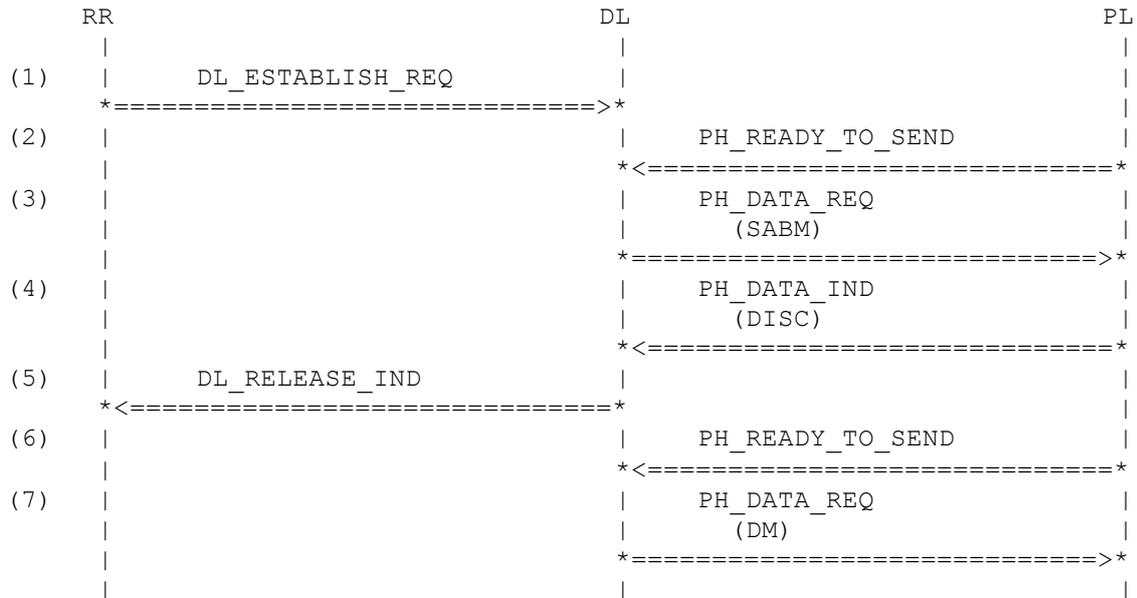
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_DL
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_A_UL

History: 21.03.97 DL Initial
 07.08.97 LE cs added for DL_RELEASE_IND

4.14.11 DL518: Different commands - establish (FACCH Halfrate)

Description: RR starts an established request by sending a DL-ESTABLISH request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a SABM frame. On receiving a PH-DATA indication primitive containing a different unnumbered SABM command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-ESTABLISH confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL000



Parametrization

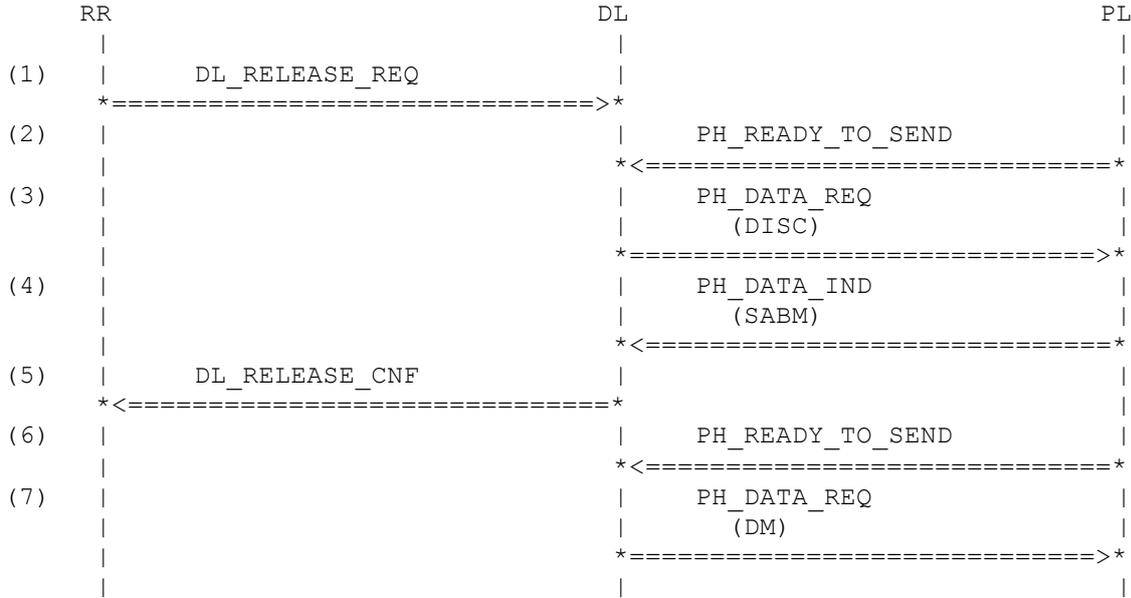
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_DL
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_UL

History: 08.07.98 LE Initial

4.14.12 DL119: Different commands - release (SDCCH)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL awaits a PH-READY-TO-SEND primitive from PL and then issues a PH-DATA request primitive containing the un-numbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing a SABM command from the base station, DL awaits a PH-READY-TO-SEND primitive and then issues PH-DATA request primitive with a DM response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1) DL012

Preamble:



Parametrization

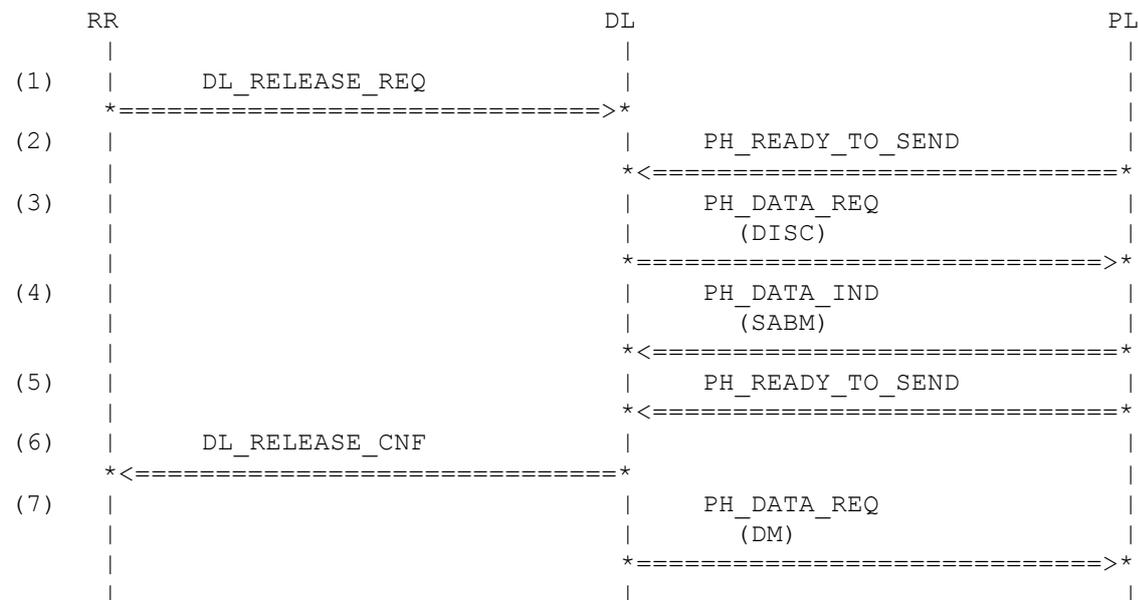
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_RELEASE_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_DL
(5) DL_RELEASE_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DM_A_UL

History: 21.03.97 DL Initial

4.14.13 DL120: Different commands - release (FACCH Fullrate)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing a different unnumbered DISC command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

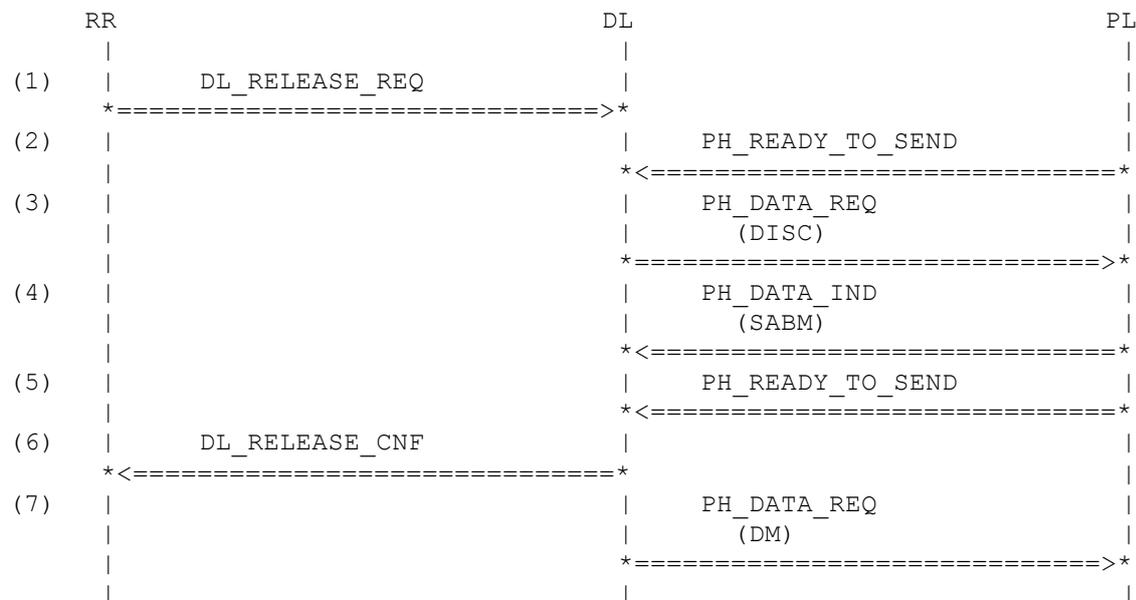
Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_B_I06_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) DL_RELEASE_CNF	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_A_UL

History: 21.03.97 DL Initial

4.14.14 DL520: Different commands - release (FACCH Halfrate)

Description: RR starts a disconnect request by sending a DL-RELEASE request primitive to DL. DL then issues a PH-DATA request primitive containing the unnumbered command as part of a DISC frame. On receiving a PH-DATA indication primitive containing a different unnumbered DISC command from the base station, DL then issues PH-DATA request primitive with a UA response (F bit set to the same value of the P bit in the previous command). DL then reports failure by sending a DL-RELEASE confirmation primitive to RR. (Ref. [20] 5.4.6.1)

Preamble: DL022



Parametrization

Primitive	Parameter	Value
(1) DL_RELEASE_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_A_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_B_I06_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) DL_RELEASE_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_A_UL

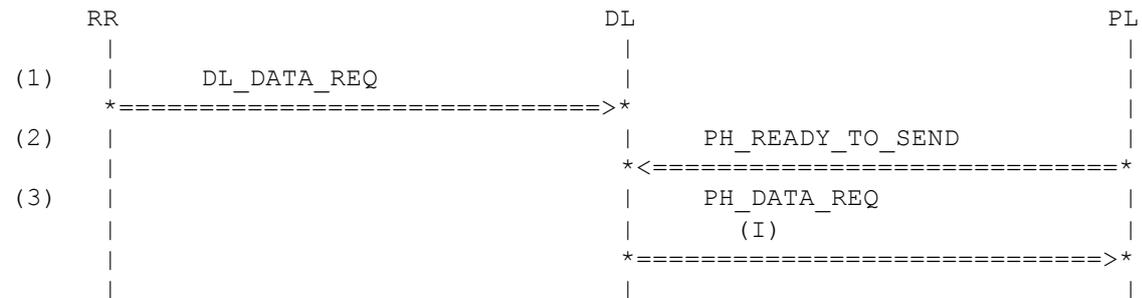
History: 08.07.98 LE Initial

4.15 Acknowledged Information Transfer

4.15.1 DL131: Transmit single I frame (SDCCH)

Description: RR sends a Layer-3 message to DL. The data to be sent is less than N201 bytes. DL awaits a PH-READY-TO-SEND primitive and then and then sends an I frame (M bit set to 0) to PL as part of a PH-DATA request primitive. (Ref. [20] 5.5.1)

Preamble: DL012



Parametrization

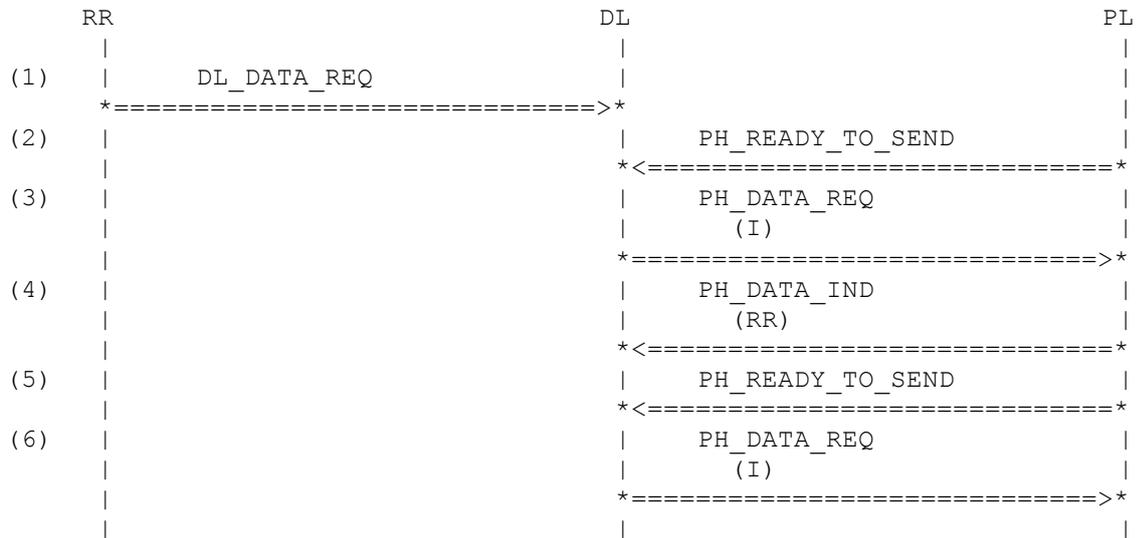
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L16_1_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_16_UL

History: 21.03.97 DL Initial

4.15.2 DL132: Transmit several I frames (SDCCH)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL awaits a PH-READY-TO-SEND primitive and then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL then awaits a RR frame (as part of a PH-DATA indication primitive) and a PH-READY-TO-SEND primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is sent. (Ref. [20] 5.5.1)

Preamble: DL012



Parametrization

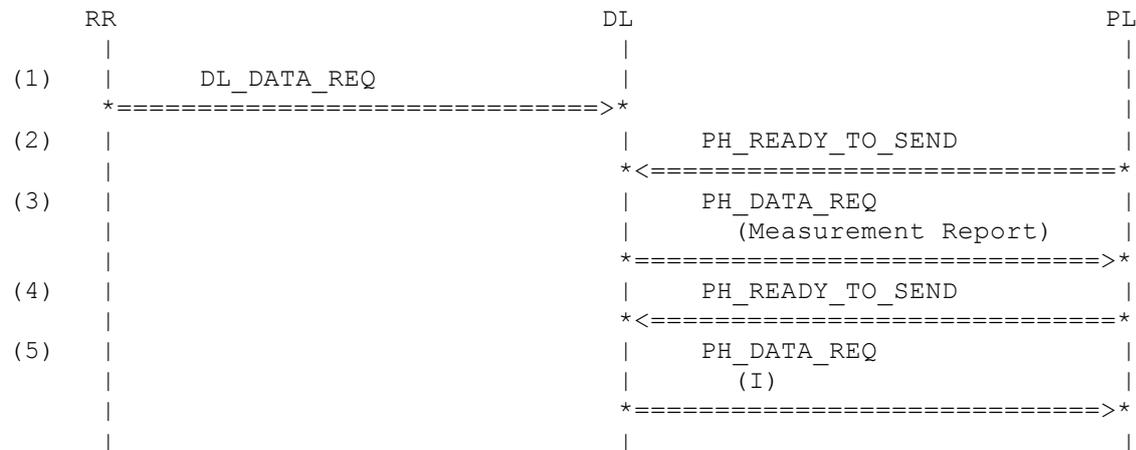
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I04_UL

History: 21.03.97 DL Initial

4.15.3 DL133: Transmit single I frame (SACCH)

Description: RR sends a Layer-3 message to DL. The data to be sent is less than N201 bytes. DL awaits a PH-READY-TO-SEND primitive and then and then sends an I frame (M bit set to 0) to PL as part of a PH-DATA request primitive. (Ref. [20] 5.5.1)

Preamble: DL027



Parametrization

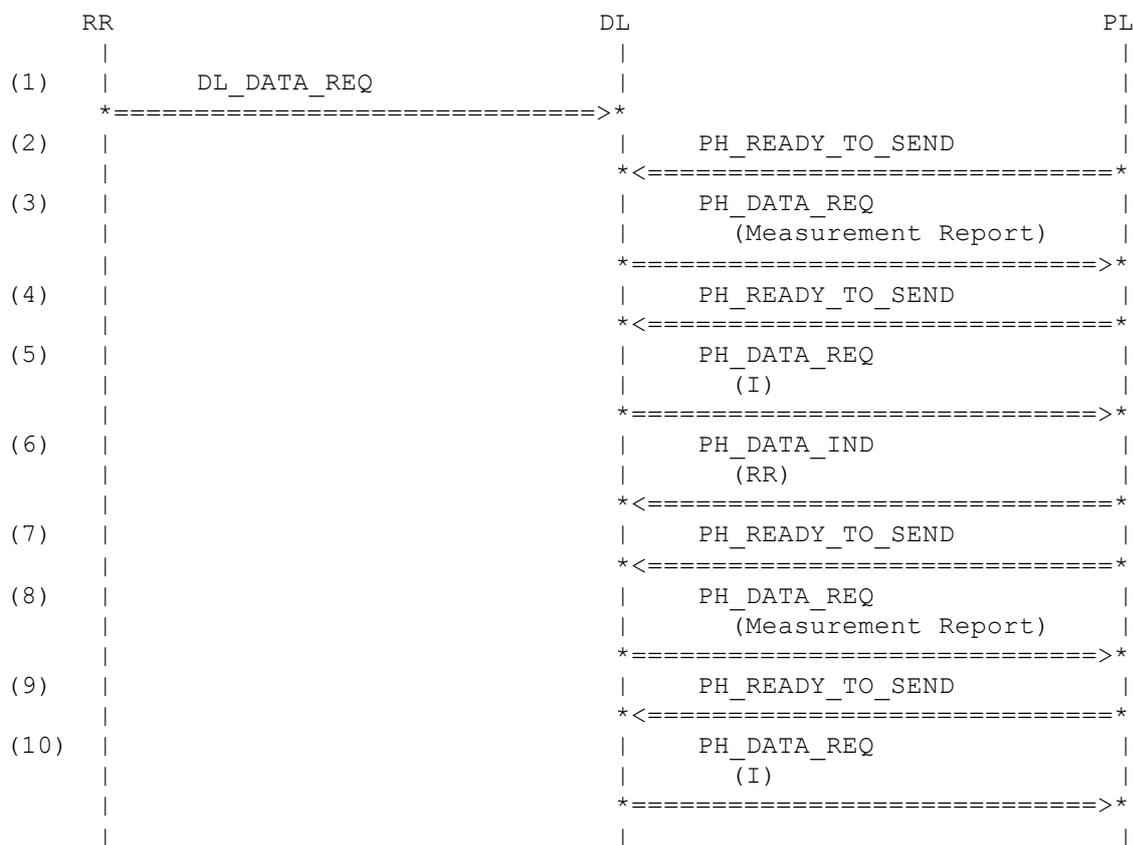
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L16_1_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UI_FRAME_UP_SACCH_M2
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(5) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	I_16_S3_UL

History: 21.03.97 DL Initial
 08.07.98 LE UI frame added

4.15.4 DL134: Transmit several I frames (SACCH)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL awaits a PH-READY-TO-SEND primitive and then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL then awaits a RR frame (as part of a PH-DATA indication primitive) and a PH-READY-TO-SEND primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is sent. (Ref. [20] 5.5.1)

Preamble: DL027



Parametrization

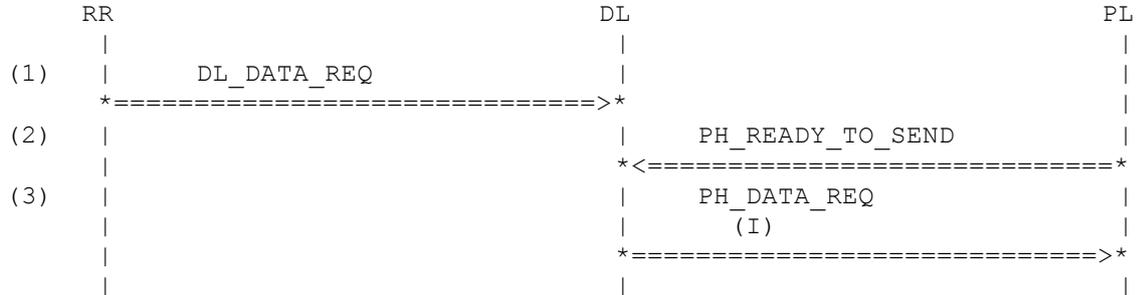
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_3 L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED UI_FRAME_UP_SACCH_M2
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(5) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED I_118_S3_UL
(6) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED RR_A_S3_NR1_DL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED UI_FRAME_UP_SACCH_M2
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED I_106_S3_UL

History: 21.03.97 DL Initial
 08.07.98 LE UI frame added

4.15.5 DL135: Transmit single I frame (FACCH Fullrate)

Description: RR sends a Layer-3 message to DL. The data to be is less N201 bytes and can be sent as a single I frame. DL then sends the I frame (M bit set to 0) to PL as part of a PH-DATA request primitive. (Ref. [20] 5.5.1)

Preamble: DL022



Parametrization

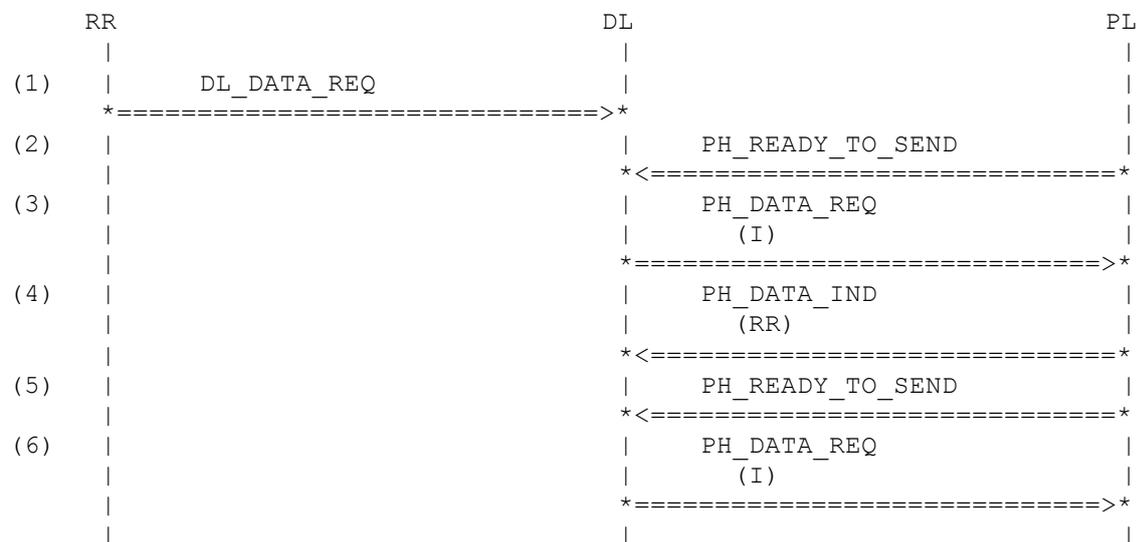
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L16_1_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_16_UL

History: 21.03.97 DL Initial

4.15.6 DL136: Transmit several I frames (FACCH Fullrate)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL then awaits a RR frame as part of a PH-DATA indication primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is sent. (Ref. [20] 5.5.1)

Preamble: DL022



Parametrization

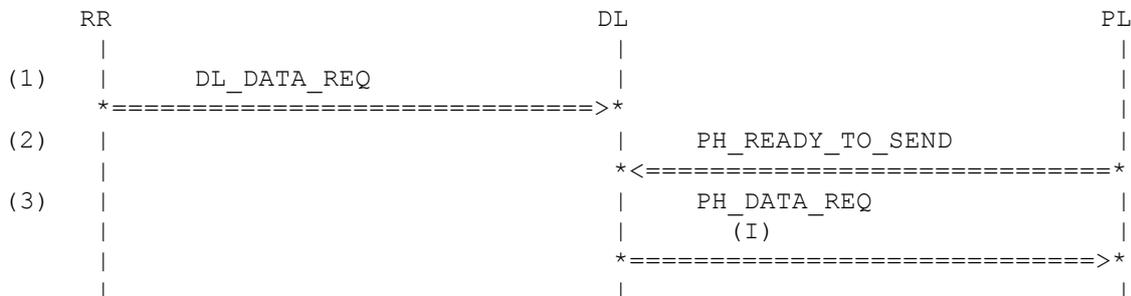
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_I20_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	RR_A_NR1_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_I04_UL

History: 21.03.97 DL Initial

4.15.7 DL535: Transmit single I frame (FACCH Halfrate)

Description: RR sends a Layer-3 message to DL. The data to be is less N201 bytes and can be sent as a single I frame. DL then sends the I frame (M bit set to 0) to PL as part of a PH-DATA request primitive. (Ref. [20] 5.5.1)

Preamble: DL422



Parametrization

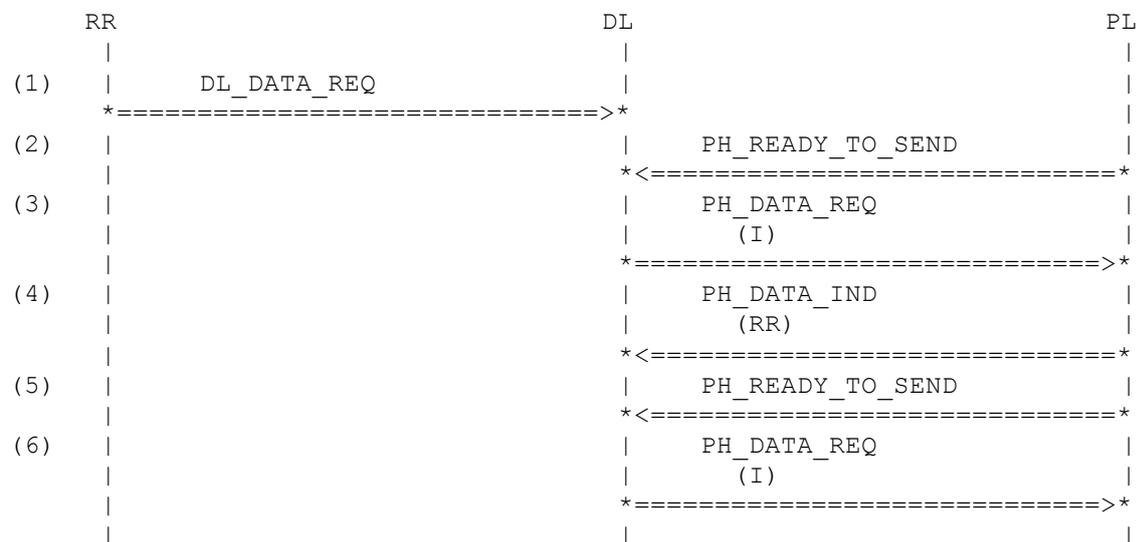
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L16_1_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	I_16_UL

History: 08.07.98 LE Initial

4.15.8 DL536: Transmit several I frames (FACCH Halfrate)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL then awaits a RR frame as part of a PH-DATA indication primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is sent. (Ref. [20] 5.5.1)

Preamble: DL422



Parametrization

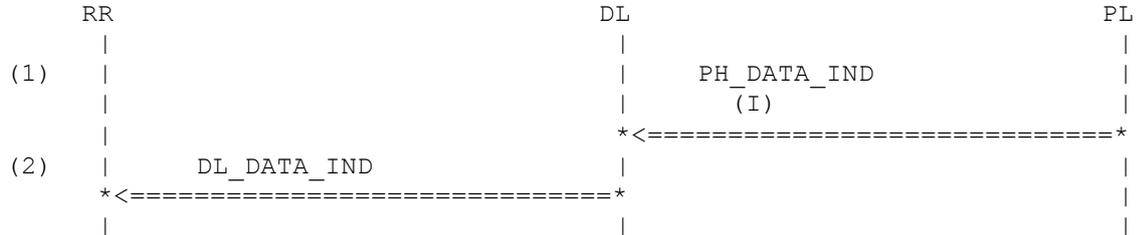
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	I_I20_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	RR_A_NR1_DL
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	I_I04_UL

History: 08.07.98 LE Initial

4.15.9 DL137: Receive single I frame (SDCCH)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received is less than N201 bytes (M bit set to 0). DL then sends L3 message to RR as part of a DL-DATA indication primitive. (Ref. [20] 5.5.2)

Preamble: DL012



Parametrization

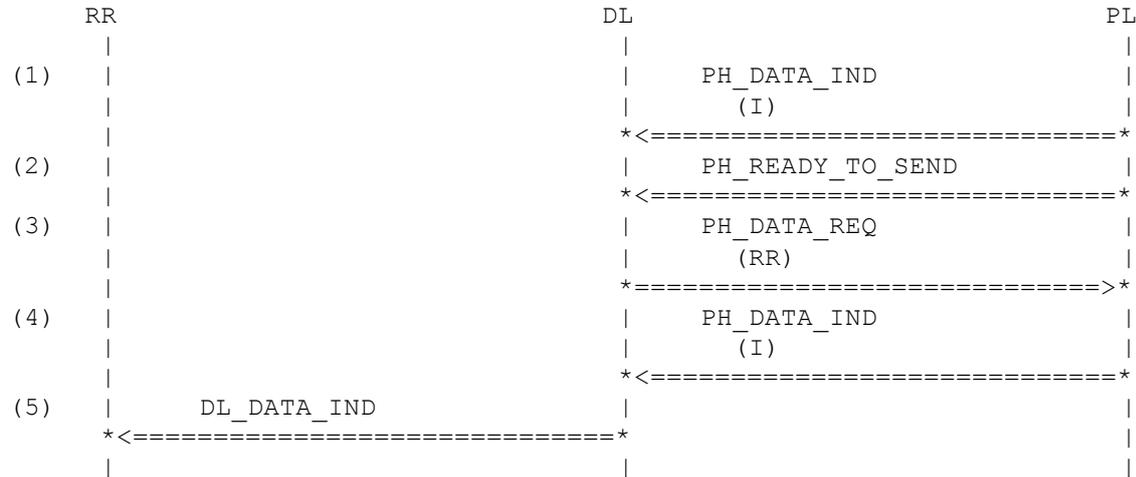
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_16_DL
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L16_2_UL

History: 21.03.97 DL Initial

4.15.10 DL138: Receive several I frames (SDCCH)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received exceeds N201 bytes. DL awaits a PH-READY-TO-SEND primitive and then sends a RR frame to PL as part of a PH-DATA request primitive. DL then awaits a RR frame (as part of a PH-DATA indication primitive) and a PH-READY-TO-SEND primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is received. DL then issues a DL-DATA indication primitive to RR. (Ref. [20] 5.5.2) DL012

Preamble:



Parametrization

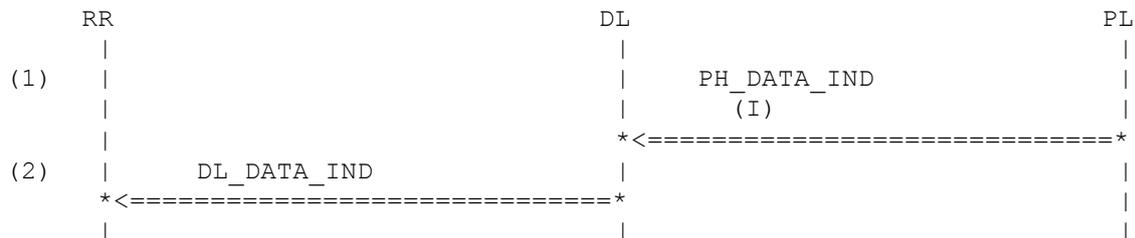
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_I20_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	RR_A_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_I04_NS1_DL
(5) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L24_DL

History: 21.03.97 DL Initial

4.15.11 DL139: Receive single I frame (SACCH)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received is less than N201 bytes (M bit set to 0). DL then sends L3 message to RR as part of a DL-DATA indication primitive. (Ref. [20] 5.5.2)

Preamble: DL027



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	I_I16_S3_DL
(2) DL_DATA_IND	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L16_2_UL

History: 21.03.97 DL Initial

sapi
sdu

SAPI_3
L3_MSG_L24_DL

History: 21.03.97 DL Initial
08.07.98 LE UI frame added

4.15.13 DL141: Receive single I frame (FACCH Fullrate)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received is less than N201 bytes (M bit set to 0). DL then sends L3 message to RR as part of a DL-DATA indication primitive. (Ref. [20] 5.5.2)

Preamble: DL022



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_116_DL
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L16_2_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	RR_A_UL

History: 21.03.97 DL Initial

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_I20_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_UL
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_I04_NS1_DL
(5) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_MSG_L24_DL
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR2_UL

History: 21.03.97 DL Initial

4.15.15 DL541: Receive single I frame (FACCH Halfrate)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received is less than N201 bytes (M bit set to 0). DL then sends L3 message to RR as part of a DL-DATA indication primitive. (Ref. [20] 5.5.2)

Preamble: DL422



Parametrization

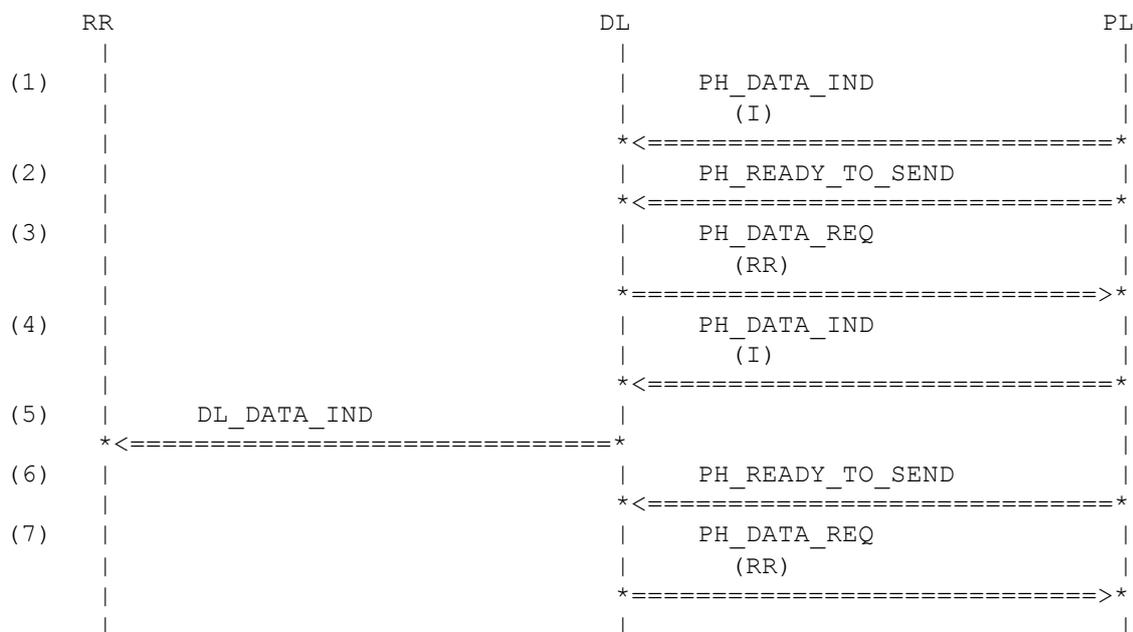
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	I_116_DL
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L16_2_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	RR_A_UL

History: 08.07.98 LE Initial

4.15.16 DL542: Receive several I frames (FACCH Halfrate)

Description: An I frame is received as part of a PH-DATA indication primitive. The data to be received exceeds N201 bytes. DL awaits a PH-READY-TO-SEND primitive and then sends a RR frame to PL as part of a PH-DATA request primitive. DL then awaits a RR frame (as part of a PH-DATA indication primitive) and a PH-READY-TO-SEND primitive before sending the next frame. The process continues until an I frame with M bit set to 0 is received. DL then issues a DL-DATA indication primitive to RR. (Ref. [20] 5.5.2)

Preamble: DL422



Parametrization

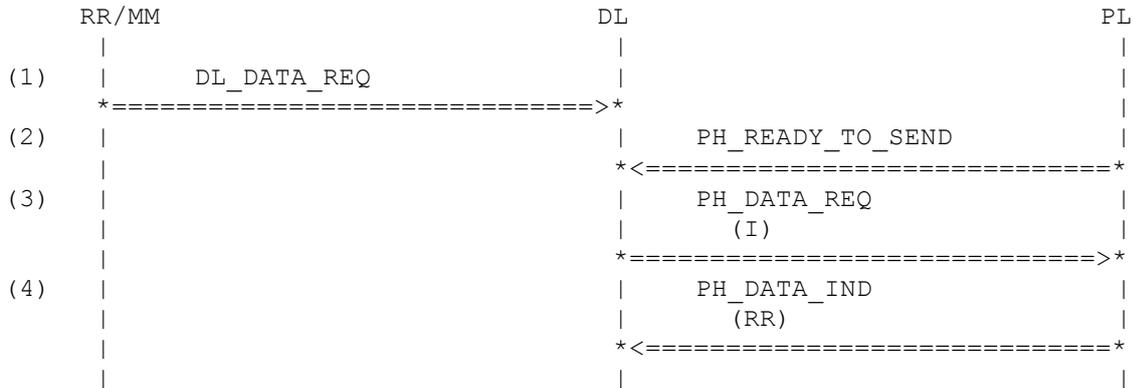
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED I_I20_DL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED RR_A_UL
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED I_I04_NS1_DL
(5) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 L3_MSG_L24_DL
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED RR_A_NR2_UL

History: 08.07.98 LE Initial

4.15.17 DL143: N(R) sequence error occurs during frame transmission (SDCCH)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL awaits a PH-READY-TO-SEND primitive and then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL subsequently receives a RR frame (as part of a PH-DATA indication primitive) which does not contain the expected N(R) sequence. Further transmission is suspended and a MDL-ERROR indication primitive with error cause set to „sequence error: perform abnormal release“ is sent to MM. (Ref. [20] 5.7.4)

Preamble: DL012



Parametrization

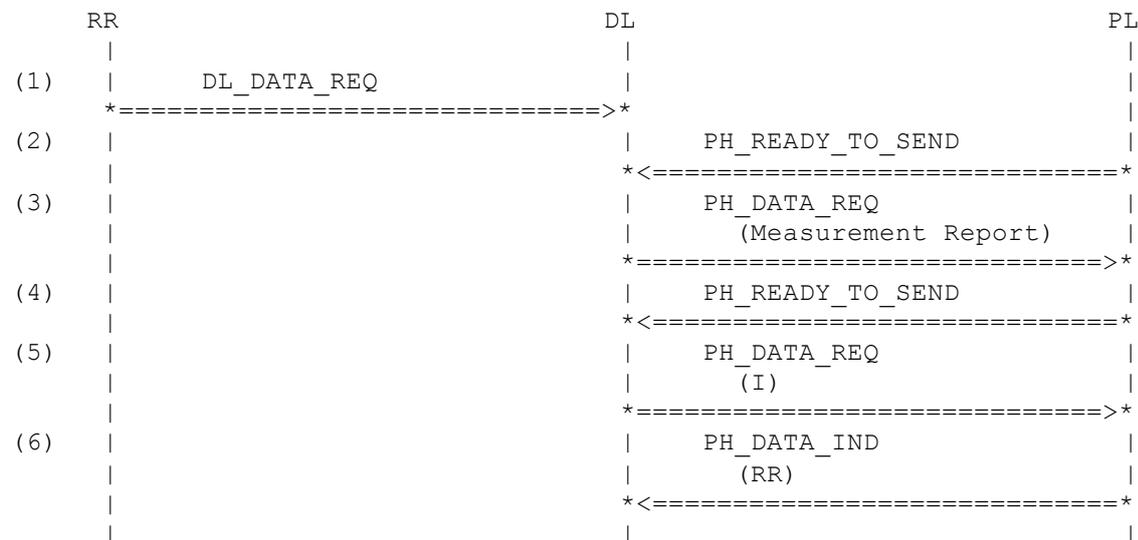
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_I20_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	RR_A_NR2_DL

History: 21.03.97 DL Initial

4.15.18 DL144: N(R) sequence error occurs during frame transmission (SACCH)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL awaits a PH-READY-TO-SEND primitive and then sets the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL subsequently receives a RR frame (as part of a PH-DATA indication primitive) which does not contain the expected N(R) sequence. Further transmission is suspended and a MDL-ERROR indication primitive with error cause set to „sequence error: perform abnormal release“ is sent to MM. (Ref. [20] 5.7.4)

Preamble: DL027



Parametrization

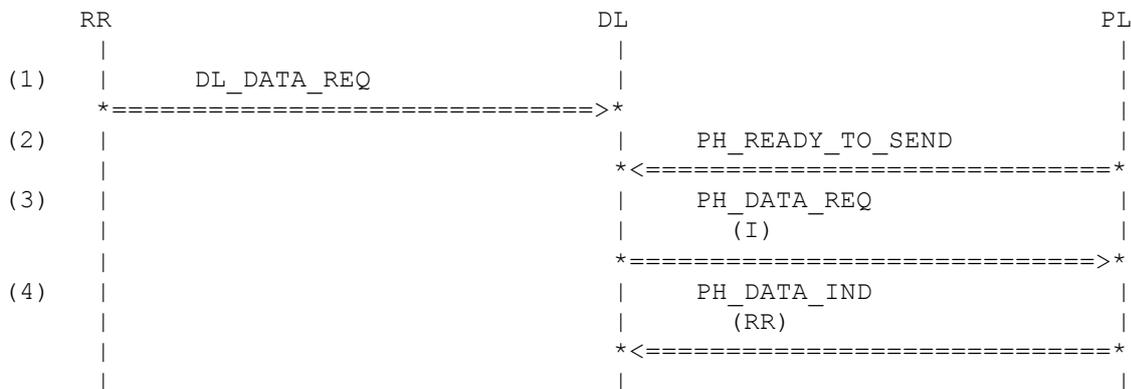
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_3
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	UI_FRAME_UP_SACCH_M2
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(5) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	I_18_S3_UL
(6) PH_DATA_IND	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	RR_A_S3_NR2_DL

History: 21.03.97 DL Initial

4.15.19 DL145: N(R) sequence error occurs during frame transmission (FACCH Fullrate)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL sends the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL subsequently receives a RR frame (as part of a PH-DATA indication primitive) which does not contain the expected N(R) sequence. Further transmission is suspended and a MDL-ERROR indication primitive with error cause set to „sequence error: perform abnormal release“ is sent to MM. (Ref. [20] 5.7.4)

Preamble: DL022



Parametrization

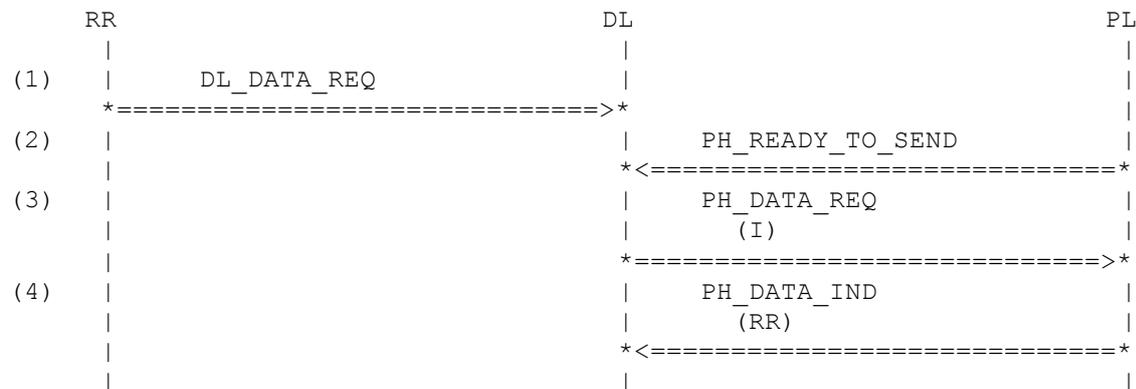
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	I_I20_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	RR_A_NR2_DL

History: 21.03.97 DL Initial

4.15.20 DL545: N(R) sequence error occurs during frame transm. (FACCH Halfrate)

Description: RR sends a Layer-3 message to DL. The data to be sent exceeds N201 bytes and is accordingly segmented by DL. DL sends the first I frame (M bit set to 1) to PL as part of a PH-DATA request primitive. DL subsequently receives a RR frame (as part of a PH-DATA indication primitive) which does not contain the expected N(R) sequence. Further transmission is suspended and a MDL-ERROR indication primitive with error cause set to „sequence error: perform abnormal release“ is sent to MM. (Ref. [20] 5.7.4)

Preamble: DL422



Parametrization

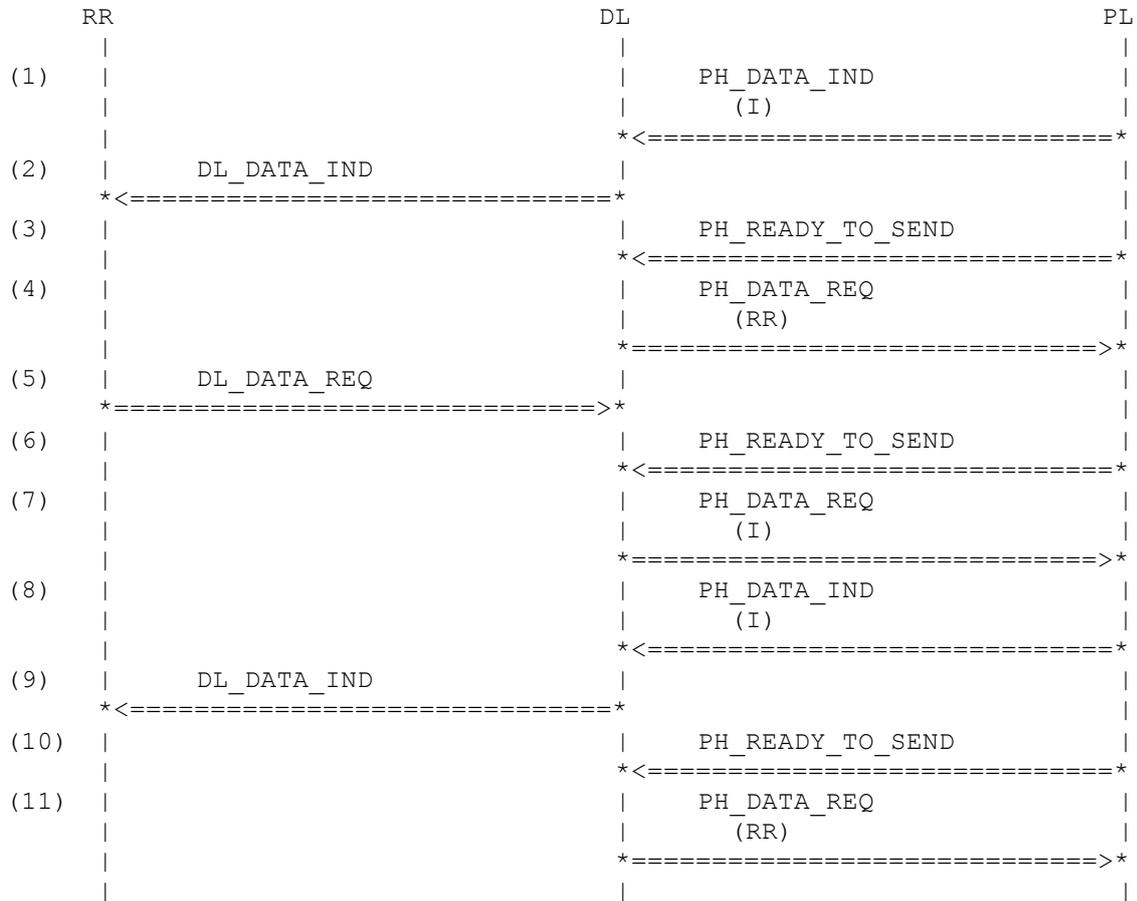
Primitive	Parameter	Value
(1) DL_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_MSG_L24_UL
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	I_I20_UL
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	RR_A_NR2_DL

History: 08.07.98 LE Initial

4.15.21 DL146: Transmit I frame, acknowledged by I frame (SDCCH)

Description: RR sends a Layer-3 message to DL. The data is sent by an I frame to the infrastructure. The message is acknowledged by the network with an I command containing the next message. The message is forwarded to the upper layer and confirmed with an RR response.

Preamble: DL012



Parametrization

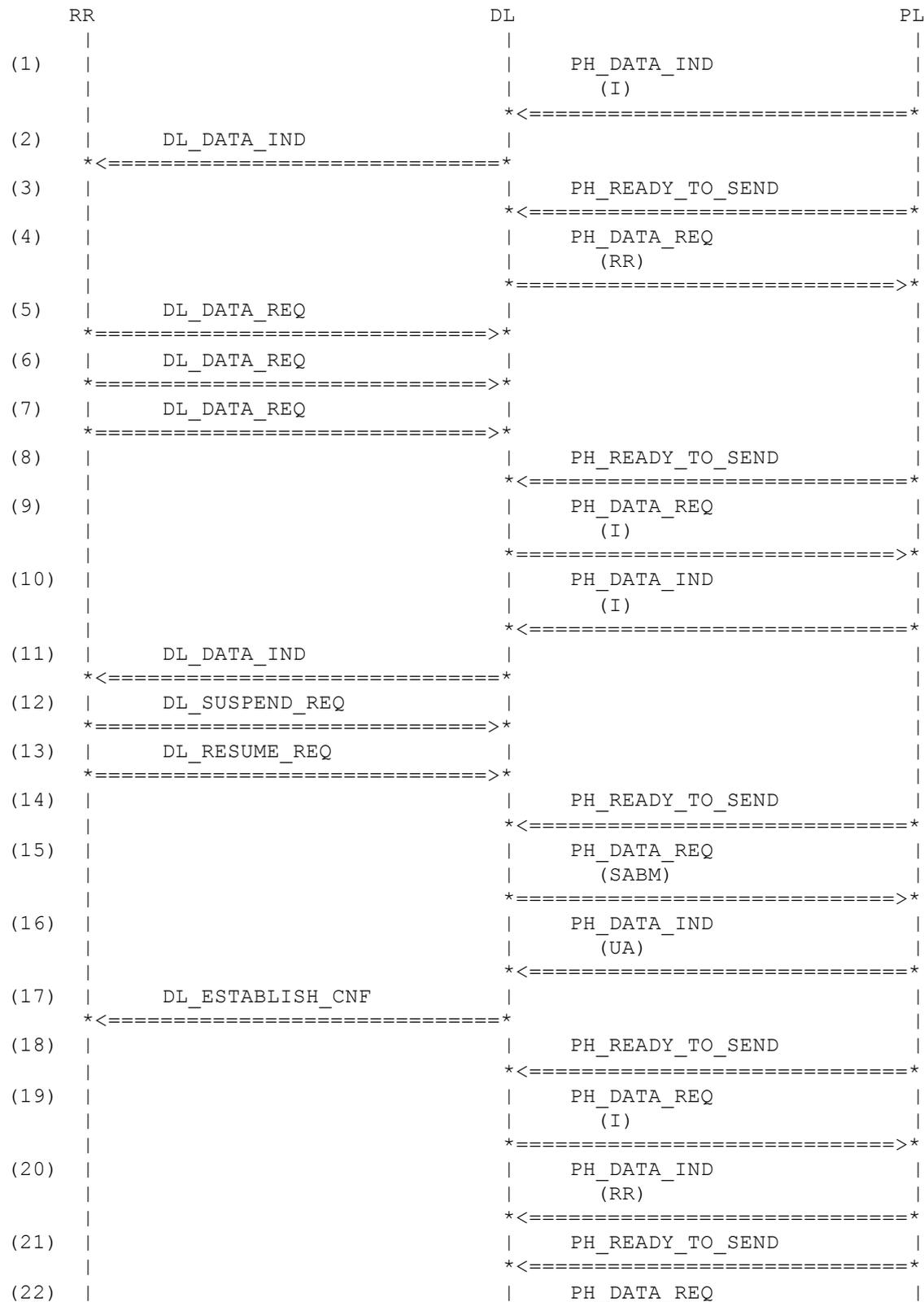
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_SETUP
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_SETUP
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CALL_CONFIRM
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD
(9) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMD
(10) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(11) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_CMD

History: 04.12.97 LE Initial

4.15.22 DL147: Transmit I frames , acknowledged by I frame (SDCCH)

Description: RR sends a three Layer-3 message to DL. The first data is sent by an I frame to the infrastructure. The message is acknowledged by the network with an I command containing the assignment command. Then the switch to the facch is done. After resumption the two other messages has to be send.

Preamble: DL012



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_SETUP
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_SETUP
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CALL_CONFIRM
(6) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ALERT
(7) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CONNECT
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM
(10) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD
(11) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMD
(12) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(13) DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

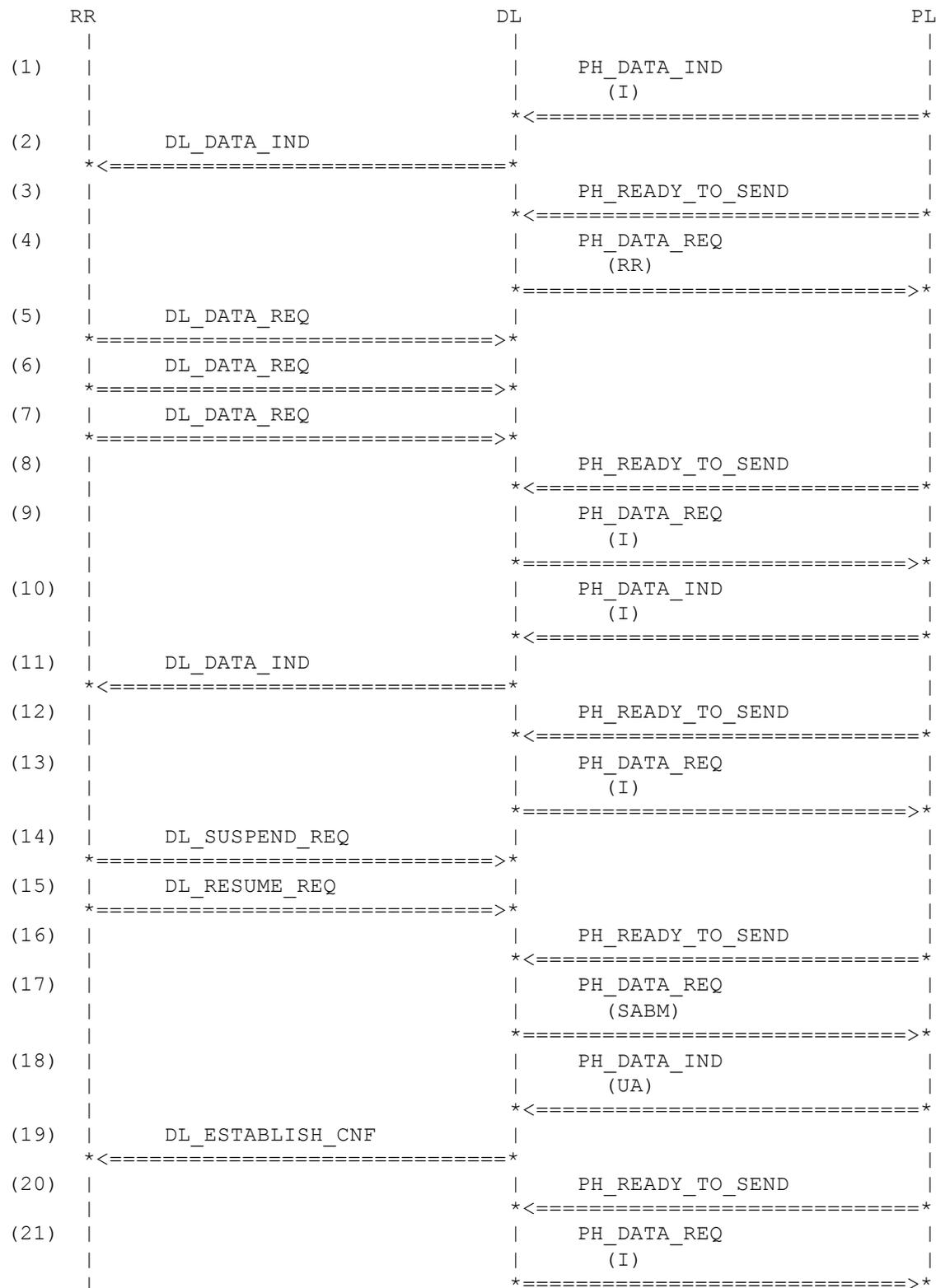
(15) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(16) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(17) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(18) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(19) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(20) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP
(21) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(22) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ALERT
(23) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ALERT
(24) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(25) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_CONNECT
(26) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_CONNECT

History: 05.12.97 LE Initial

4.15.23 DL148: Transmit I frames , retransmission on new channel

Description: RR sends a three Layer-3 message to DL. The first data is sent by an I frame to the infrastructure. The message is acknowledged by the network with an I command containing the assignment command. The message is acknowledged with an I frame on SDCCH. Then the switch to the facch is done. After re-sumption the two other messages has to be send.

Preamble: DL012



```
(22) | | PH_DATA_IND | |  
      | | (RR) | |  
      | | *<===== | |  
(23) | | PH_READY_TO_SEND | |  
      | | *<===== | |  
(24) | | PH_DATA_REQ | |  
      | | (I) | |  
      | | *===== | |  
(25) | | PH_DATA_IND | |  
      | | (RR) | |  
      | | *<===== | |  
(26) | | PH_READY_TO_SEND | |  
      | | *<===== | |  
(27) | | PH_DATA_REQ | |  
      | | (I) | |  
      | | *===== | |  
(28) | | PH_DATA_IND | |  
      | | (RR) | |  
      | | *<===== | |  
      | | | |
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_SETUP
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_SETUP
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CALL_CONFIRM
(6) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ALERT
(7) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CONNECT
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM
(10)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD
(11)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMD
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(13)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ALERT_2
(14)	DL_SUSPEND_REQ ch_type sapi	CH_TYPE_SDCCH SAPI_0

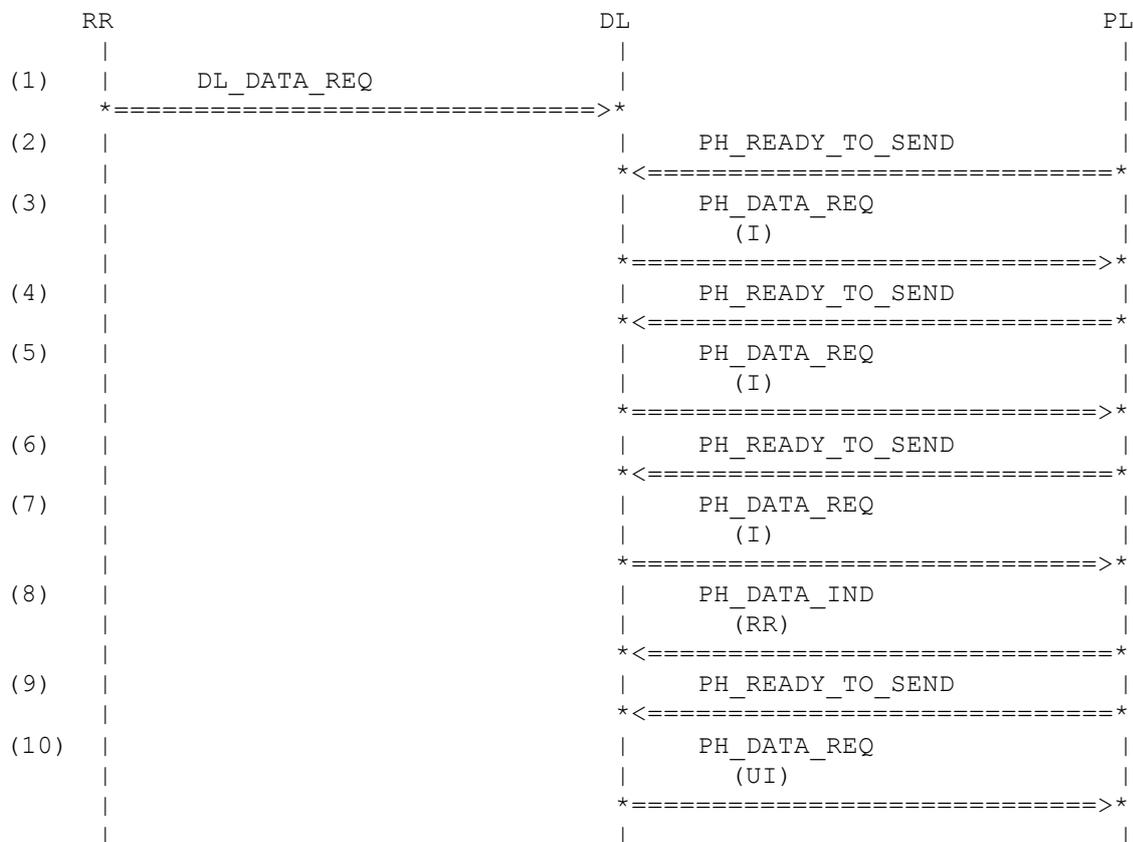
(15)	DL_RESUME_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(17)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(18)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(19)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH SAPI_0
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(21)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(22)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ALERT
(25)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ALERT
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_CONNECT
(28)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_CONNECT

History: 05.12.97 LE Initial

4.15.24 DL149: Retransmission of I frames , Timer Recovery

Description: RR sends a layer 3 message to DL. The transmission of I frames is not acknowledged by the infrastructure. After time-out of T200 DL retransmit the I -frame and enters Timer Recovery state. After time-out of T200 the message is retransmitted.

Preamble: DL012



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CALL_CONFIRM
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM_2
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM_3
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CALL_CONFIRM_3
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_CALL_CONFIRM_3
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.15.25 DL150: Resumption with stored messages

Description: A connection is established on SDCCH. A setup message receives. The upper layer answer with a call confirm message followed by an alerting message. The call confirm message is send to the network and confirmed. Before sending the alerting message the assignment command receives. The SDCCH connection is released locally. The resumption on FACCH is expected. After successfull establishment the assignment command is send and confirmed by the network. DL shall send then the stored alerting message.

Preamble: DL000

RR	DL	PL
(1)		
DL_ESTABLISH_REQ		
(Paging response)		
=====>		
(2)	PH_READY_TO_SEND	
	<=====	
(3)	PH_DATA_REQ	
	(SABM, Paging Response)	
	=====>	
(4)	PH_DATA_IND	
	(UA, Paging Response)	
	<=====	
(5)		
DL_ESTABLISH_CNF		
<=====		
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_DATA_REQ	
	(UI)	
	=====>	
(8)	PH_DATA_IND	
	(I, Setup)	
	<=====	
(9)		
DL_DATA_IND		
(Setup Command)		
<=====		
(10)		
DL_DATA_REQ		
(Call Confirm)		
=====>		
(11)		
DL_DATA_REQ		
(Alerting)		
=====>		
(12)	PH_READY_TO_SEND	
	<=====	
(13)	PH_DATA_REQ	
	(I, Call Confirm)	
	=====>	
(14)	PH_DATA_IND	
	(RR)	
	<=====	

```

(15) | | PH_DATA_IND | |
      | | (I, Assignment Command) | |
      | | *<===== | |
(16) | | DL_DATA_IND | |
      | | (Assignment Command) | |
      | | *<===== | |
(17) | | DL_SUSPEND_REQ | |
      | | *===== | |
(18) | | DL_RESUME_REQ | |
      | | (Assignment Complete) | |
      | | *===== | |
(19) | | PH_READY_TO_SEND | |
      | | *<===== | |
(20) | | PH_DATA_REQ | |
      | | (SABM) | |
      | | *===== | |
(21) | | PH_DATA_IND | |
      | | (UA) | |
      | | *<===== | |
(22) | | DL_ESTABLISH_CNF | |
      | | *<===== | |
(23) | | PH_READY_TO_SEND | |
      | | *<===== | |
(24) | | PH_DATA_REQ | |
      | | (I, Assignment Complete) | |
      | | *===== | |
(25) | | PH_DATA_IND | |
      | | (RR) | |
      | | *<===== | |
(26) | | PH_READY_TO_SEND | |
      | | *<===== | |
(27) | | PH_DATA_REQ | |
      | | (I, Alerting) | |
      | | *===== | |
(28) | | PH_DATA_IND | |
      | | (RR) | |
      | | *<===== | |
      | | | |
    
```

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_UPLINK
(8) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_SETUP
(9) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_SETUP
(10) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_CALL_CONFIRM
(11) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_ALERT
(12) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
	ch_type	CH_TYPE_SDCCH
(13) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_CALL_CONFIRM
(14) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	RR_CALL_CONFIRM_4

(15) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_2
(16) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_FACCH
(17) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(18) DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(19) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(20) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(21) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(22) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(23) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(24) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(25) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP
(26) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(27) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ALERT
(28) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ALERT

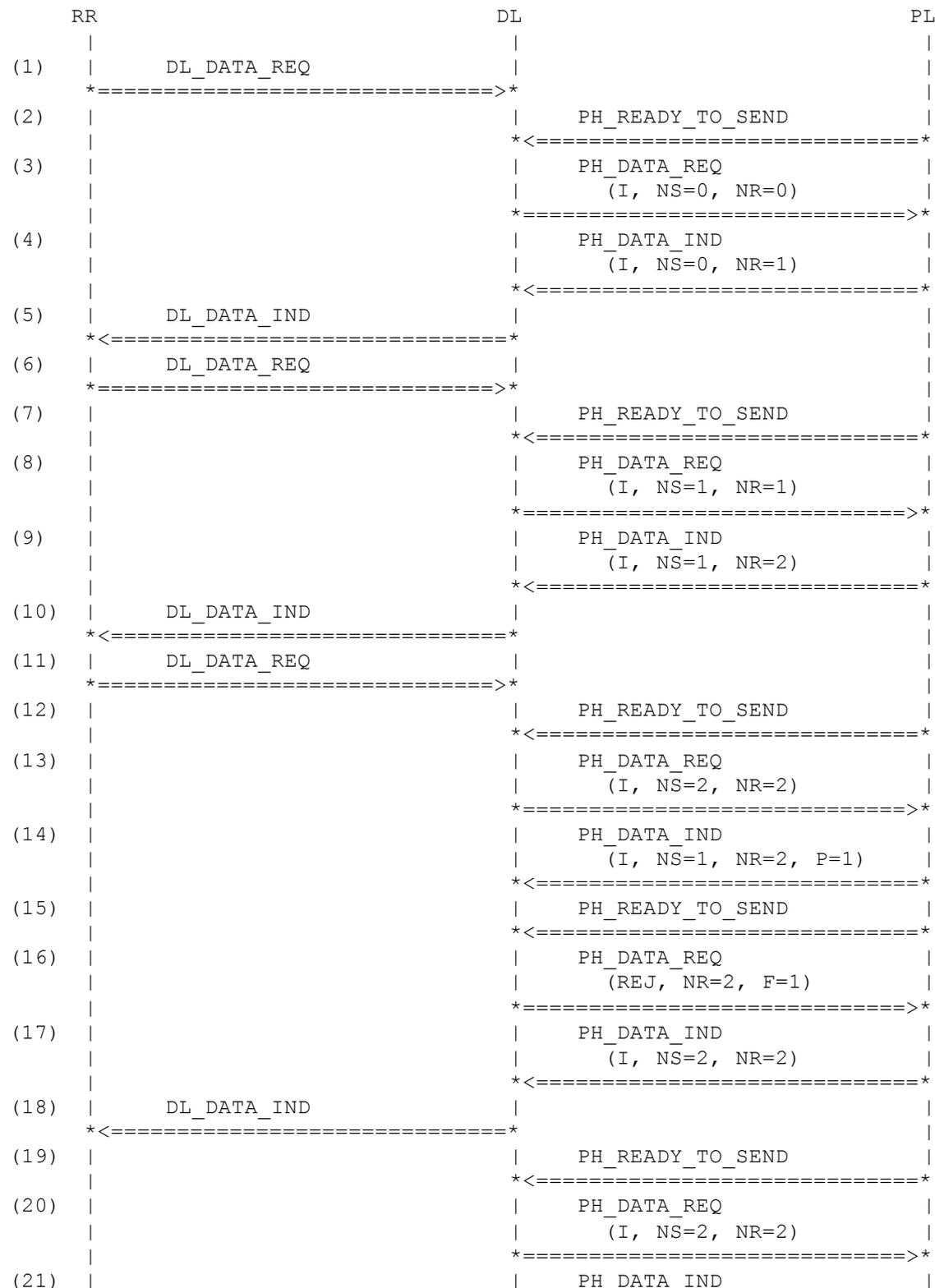
History:

22.01.98 LE Initial

4.15.26 DL151: RR response frame loss (MS to SS), SDCCH

Description: To test the layer 2 recovery mechanism in the event of RR frame loss.
 Message (9) is not arriving at the network side. The network resends its old I-frame which has not been acked by the MS. This leads to a reject supervisory frame (12) (MS is receiving the wrong seq. nr) send to the network(also T200 is stopped and restarted) which is also not received by network.

Preamble: DL200



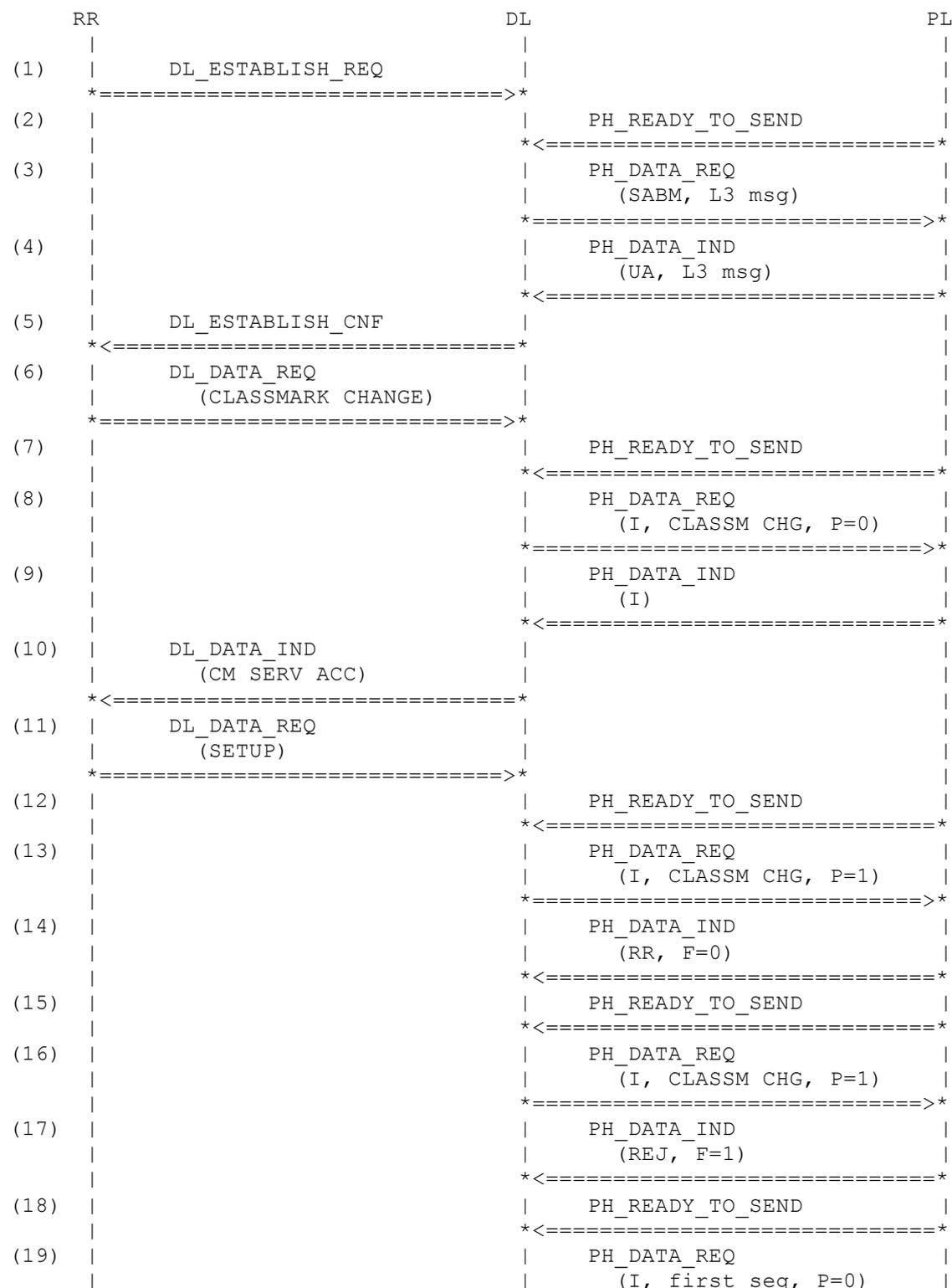
(13) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_DL151_ULb
(14) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_DL151_DLb
(15) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(16) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_DL151_ULc
(17) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_DL151_DLc
(18) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 NOT_USED
(19) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(20) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_DL151_ULd
(21) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_DL151_DLd
(22) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 NOT_USED

History: 02.06.2000 MPA Initial

4.15.27 DL152: Customer reported problem from China

Description: The testcase shall reproduce some strange behaviour seen in China. The special about this are two things. First, immediately after entering multiple frame established state, the mobile sends an I-Frame containing a RR layer 3 frame (classmark change). This leads to a crossing of I-Frames with the network. Second, the network doesn't acknowledge the received I-frame at the first possible opportunity by its own DL-I-frame. Network implementation is quite ok, at least for phase 1.

Preamble: DL000



PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CM_SERV_ACC
DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_ACC
DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_L24_UL
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F0
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR1_F1
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F1
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1_P
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR2_F0

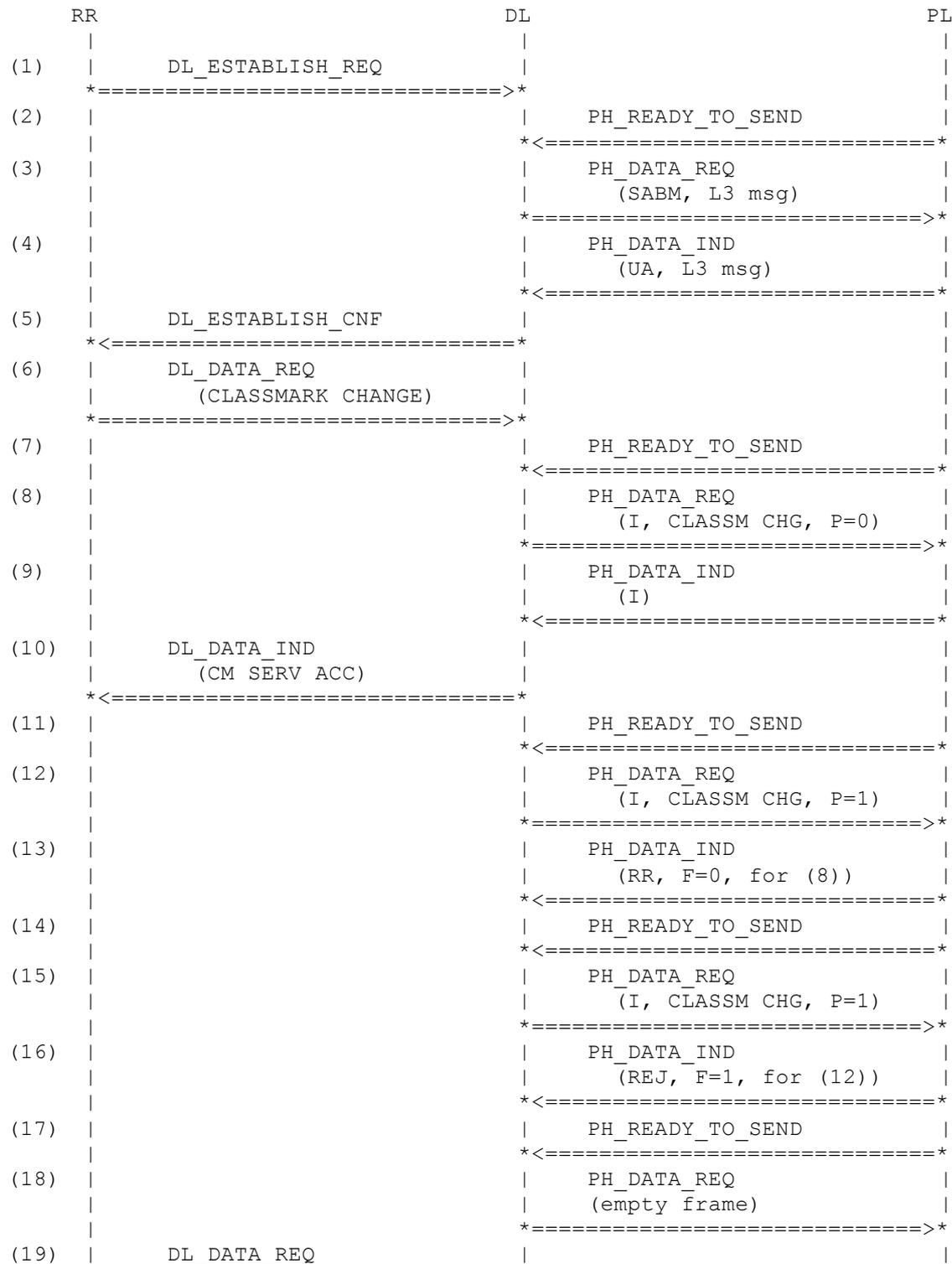
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1_P
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR2_F1
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I04_UL_NR1_NS2

History: 31.10.00 HM Initial

4.15.28 DL153: Customer reported problem from China, delayed setup

Description: The testcase shall reproduce some strange behaviour seen in China. The special about this are two things. First, immediately after entering multiple frame established state, the mobile sends an I-Frame containing a RR layer 3 frame (classmark change). This leads to a crossing of I-Frames with the network. Second, the network doesn't acknowledge the received I-frame at the first possible opportunity by its own DL-I-frame. Network implementation is quite ok, at least for phase 1. The difference between this test-case and DL152 is that the setup is delayed for a while to allow for layer 3 processing delays.

Preamble: DL000



```

        |          (SETUP)          |
        | *=====>*              |
(20)   |                          | PH_READY_TO_SEND |
        | *<=====*              |
(21)   |                          | PH_DATA_REQ      |
        |                          | (I, first seg, P=0) |
        | *=====>*              |
(22)   |                          | PH_DATA_IND      |
        |                          | (RR, F=1, for (16)) |
        | *<=====*              |
(23)   |                          | PH_READY_TO_SEND |
        | *<=====*              |
(24)   |                          | PH_DATA_REQ      |
        |                          | (I, first seg, P=1) |
        | *=====>*              |
(25)   |                          | PH_DATA_IND      |
        |                          | (RR, F=0, for (21)) |
        | *<=====*              |
(26)   |                          | PH_READY_TO_SEND |
        | *<=====*              |
(27)   |                          | PH_DATA_REQ      |
        |                          | (I, first seg, P=1) |
        | *=====>*              |
(28)   |                          | PH_DATA_IND      |
        |                          | (REJ, F=1, for (24)) |
        | *<=====*              |
(29)   |                          | PH_READY_TO_SEND |
        | *<=====*              |
(30)   |                          | PH_DATA_REQ      |
        |                          | (I, last seg, P=0) |
        | *=====>*              |
    
```

Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CM_SERV_REQ
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_MSG_CM_SERV_REQ
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_MSG_CM_SERV_REQ
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(6) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CLASSM_CHG

(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P0
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CM_SERV_ACC
(10)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_ACC
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
(13)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F0
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
(16)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR1_F1
(17)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(18)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH
(19)	DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_L24_UL
(20)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(21)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1

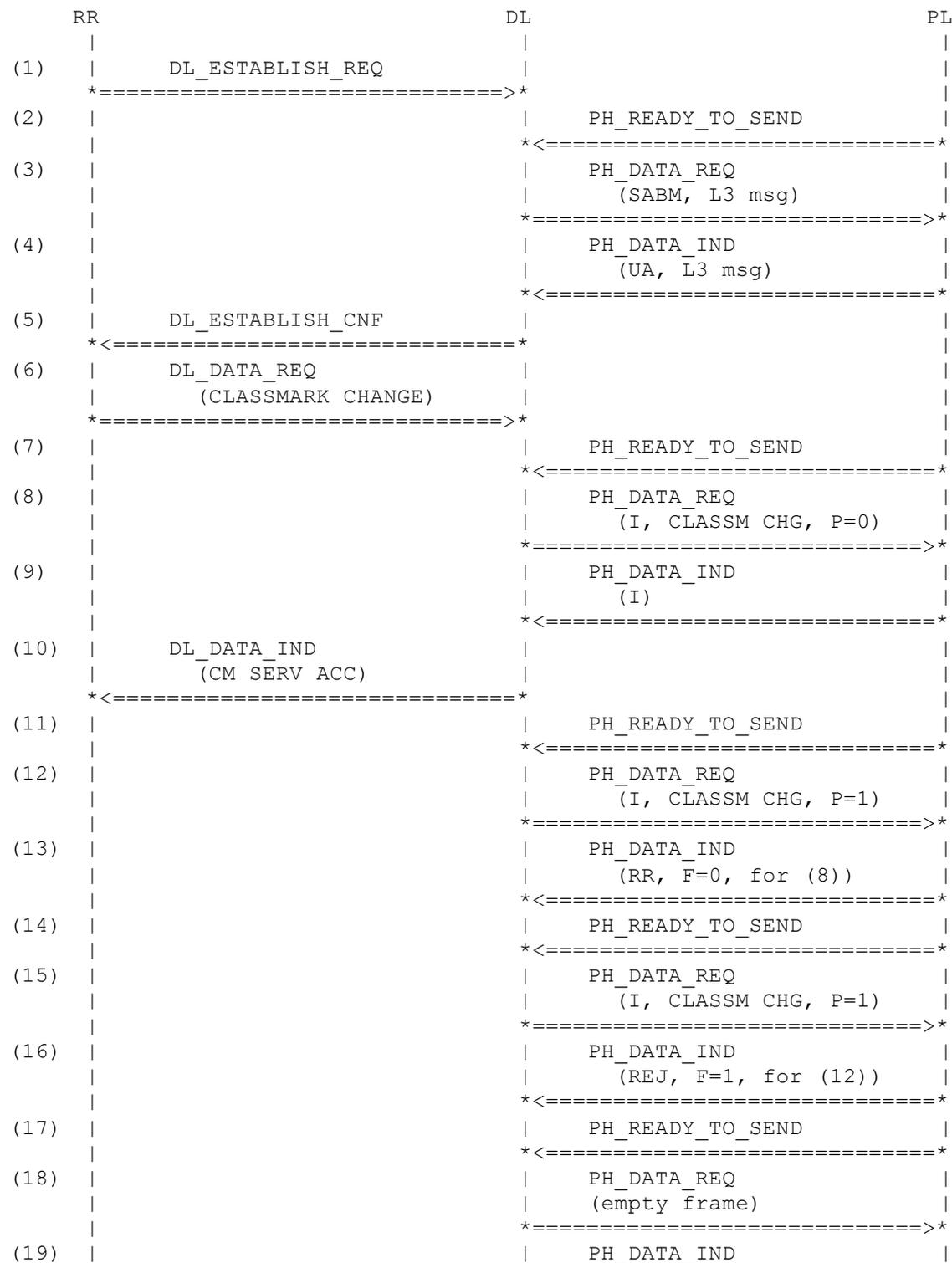
(22)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F1
(23)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(24)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1_P
(25)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR2_F0
(26)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(27)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1_P
(28)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR2_F1
(29)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(30)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I04_UL_NR1_NS2

History: 01.11.00 HM Initial

4.15.29 DL154: Customer reported problem from China, variation I after step 18

Description: The testcase shall reproduce some strange behaviour seen in China. The special about this are two things. First, immediately after entering multiple frame established state, the mobile sends an I-Frame containing a RR layer 3 frame (classmark change). This leads to a crossing of I-Frames with the network. Second, the network doesn't acknowledge the received I-frame at the first possible opportunity by its own DL-I-frame. Network implementation is quite ok, at least for phase 1. The difference between this test-case and DL153 is that after step 18 a still outstanding RR with P=1 is received before the setup is sent DL000

Preamble:



```

|                                     | (RR, F=1, for (15)) |
|                                     | *<=====|
(20) | DL_DATA_REQ                       |                       |
      | (SETUP)                          |                       |
| *=====>|
(21) |                                     | PH_READY_TO_SEND    |
|                                     | *<=====|
(22) |                                     | PH_DATA_REQ         |
      |                                     | (I, first seg, P=0) |
|                                     | *=====>|
(23) |                                     | PH_DATA_IND         |
      |                                     | (RR, F=0, for (22)) |
|                                     | *<=====|
(24) |                                     | PH_READY_TO_SEND    |
|                                     | *<=====|
(25) |                                     | PH_DATA_REQ         |
      |                                     | (I, last seg, P=0)  |
|                                     | *=====>|
(26) |                                     | PH_DATA_IND         |
      |                                     | (RR, F=0, for (24)) |
|                                     | *<=====|
(27) |                                     | PH_READY_TO_SEND    |
|                                     | *<=====|
(28) |                                     | PH_DATA_REQ         |
      |                                     | (empty frame)       |
|                                     | *=====>|
|                                     |                       |

```

Parametrization

Primitive	Parameter	Value
DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CM_SERV_REQ
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_MSG_CM_SERV_REQ
PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_MSG_CM_SERV_REQ
DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CLASSM_CHG
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

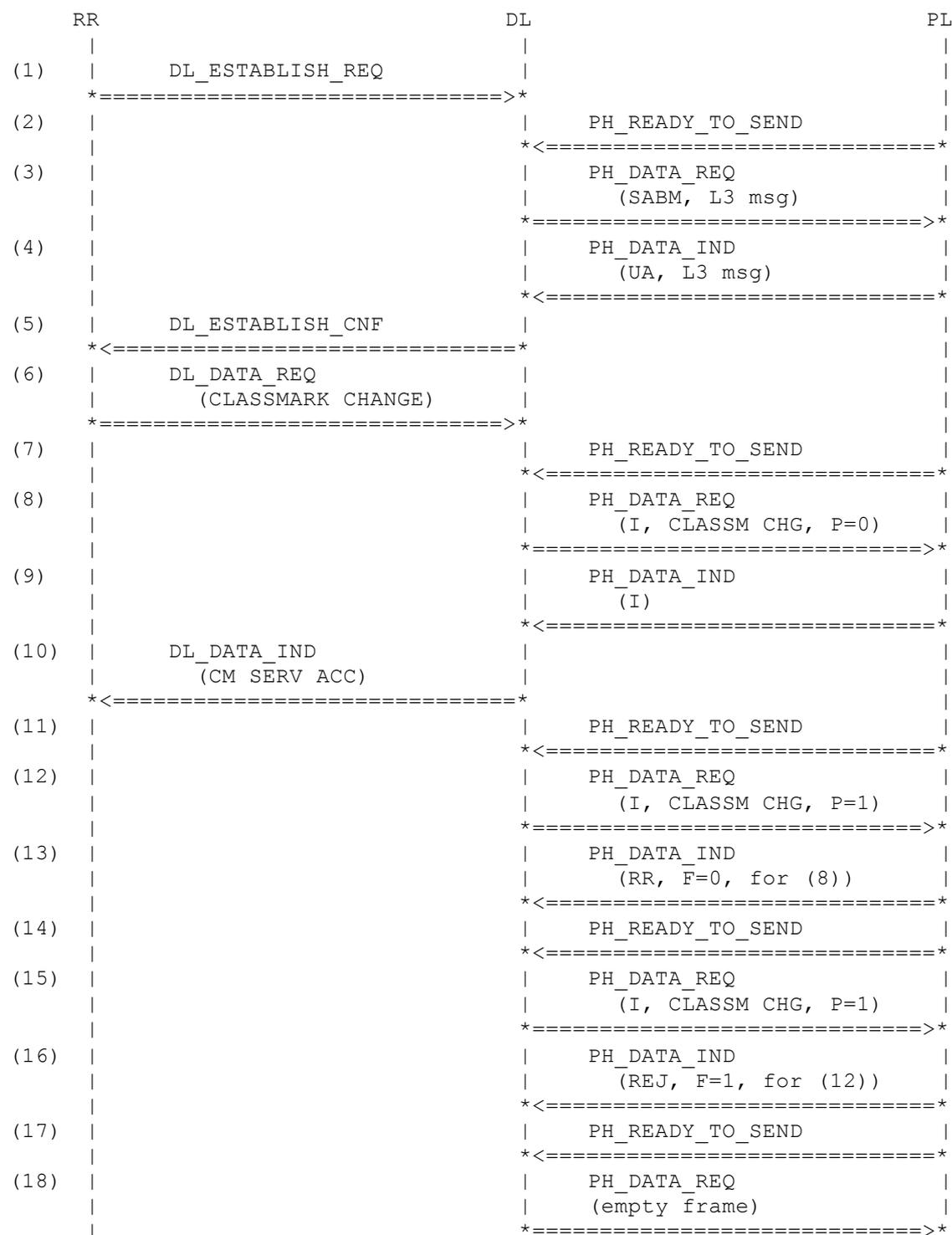
	dummy sdu	NOT_USED I_MSG_CLASSM_CHG_P0
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CM_SERV_ACC
DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_ACC
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F0
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR1_F1
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F1
DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_L24_UL
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_UL_NR1_NS1
PH_DATA_IND	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED RR_RSP_DL_NR2_F0
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I04_UL_NR1_NS2
PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR3_F0
PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 01.11.00 HM Initial

4.15.30 DL155: Customer reported problem from China, variation II after step 18

Description: The testcase shall reproduce some strange behaviour seen in China. The special about this are two things. First, immediately after entering multiple frame established state, the mobile sends an I-Frame containing a RR layer 3 frame (classmark change). This leads to a crossing of I-Frames with the network. Second, the network doesn't acknowledge the received I-frame at the first possible opportunity by its own DL-I-frame. Network implementation is quite ok, at least for phase 1. The difference between this testcase and DL154 is that after step 18 a still outstanding RR with P=1 is received and then no DL_DATA_REQ primitive is received, so that at the next opportunity an empty frame will be sent.
Preamble: DL000



```

(19) | | PH_DATA_IND | |
      | | (RR, F=1, for (15)) | |
      | | *<===== | |
(20) | | PH_READY_TO_SEND | |
      | | *<===== | |
(21) | | PH_DATA_REQ | |
      | | (empty frame) | |
      | | *===== > | |
      | | | |
    
```

Parametrization

Primitive	Parameter	Value
(31) DL_ESTABLISH_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_REQ
(32) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(33) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_MSG_CM_SERV_REQ
(34) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_MSG_CM_SERV_REQ
(35) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(36) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CLASSM_CHG
(37) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(38) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P0
(39) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CM_SERV_ACC
(40) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_ACC
(41) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(42) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1

(4 3) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F0
(4 4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4 5) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
(4 6) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR1_F1
(4 7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4 8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH
(4 9) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F1
(5 0) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5 1) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

History: 01.11.00 HM Initial


```

(19) | | PH_DATA_REQ |
| | (I, SETUP, P=0) |
| | *=====>* |
(20) | | PH_DATA_IND |
| | (RR, F=1, for (16)) |
| | *<=====* |
(21) | | PH_READY_TO_SEND |
| | *<=====* |
(22) | | PH_DATA_REQ |
| | (I, SETUP, P=1) |
| | *=====>* |
(23) | | PH_DATA_IND |
| | (RR, F=0, for (19)) |
| | *<=====* |
(24) | | PH_READY_TO_SEND |
| | *<=====* |
(25) | | PH_DATA_REQ |
| | (I, SETUP, P=1) |
| | *=====>* |
(26) | | PH_DATA_IND |
| | (REJ, F=1, for (22)) |
| | *<=====* |
(27) | | PH_READY_TO_SEND |
| | *<=====* |
(28) | | PH_DATA_REQ |
| | (empty frame) | |
| | *=====>* |
| | | |

```

Parametrization

Primitive	Parameter	Value
(52) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CM_SERV_REQ
(53) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(54) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_MSG_CM_SERV_REQ
(55) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_MSG_CM_SERV_REQ
(56) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(57) DL_DATA_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CLASSM_CHG
(58) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(59) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED I_MSG_CLASSM_CHG_P0
(60) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CM_SERV_ACC
(61) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_CM_SERV_ACC
(62) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_SETUP
(63) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(64) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
(65) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F0
(66) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(67) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_CLASSM_CHG_P1
(68) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR1_F1
(69) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(70) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP_UL_NR1_NS1
(71) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR1_F1
(72) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(73) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP_UL_NR1_NS1_P

(74) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_RSP_DL_NR2_F0
(75) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(76) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP_UL_NR1_NS1_P
(77) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED REJ_RSP_DL_NR2_F1
(78) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(79) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH

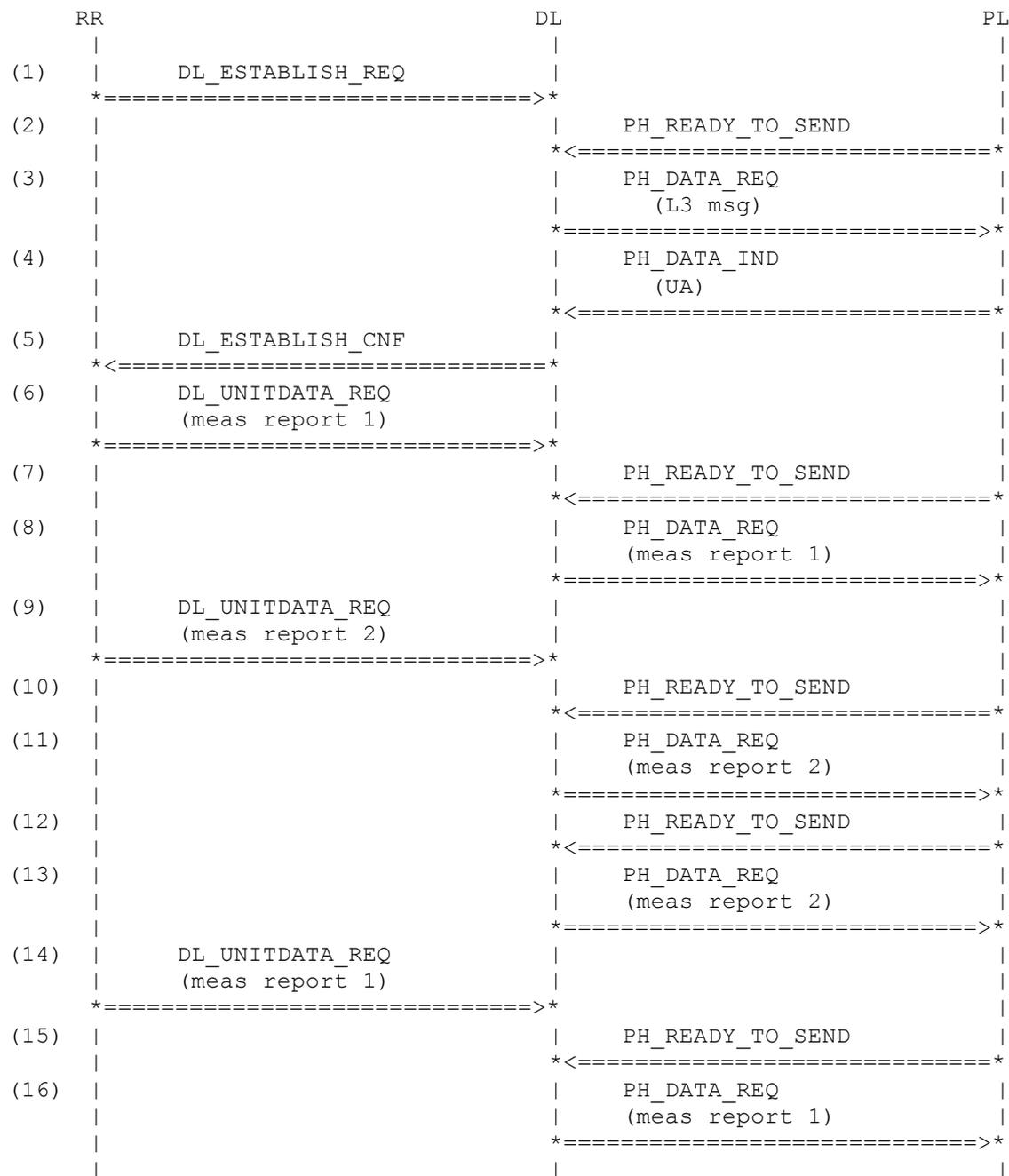
History: 02.11.00 HM Initial

4.16 Measurement Reporting

4.16.1 DL160: Repetition of last Measurement Report (main link = SDCCH)

Description: A sapi 0 link is established for SDCCH. RR forwards measurement reports to DL. Layer 1 requests the measurement reports for SACCH. If RR has coded no new measurement reports in time, the last received measurement report from RR is repeated.

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH

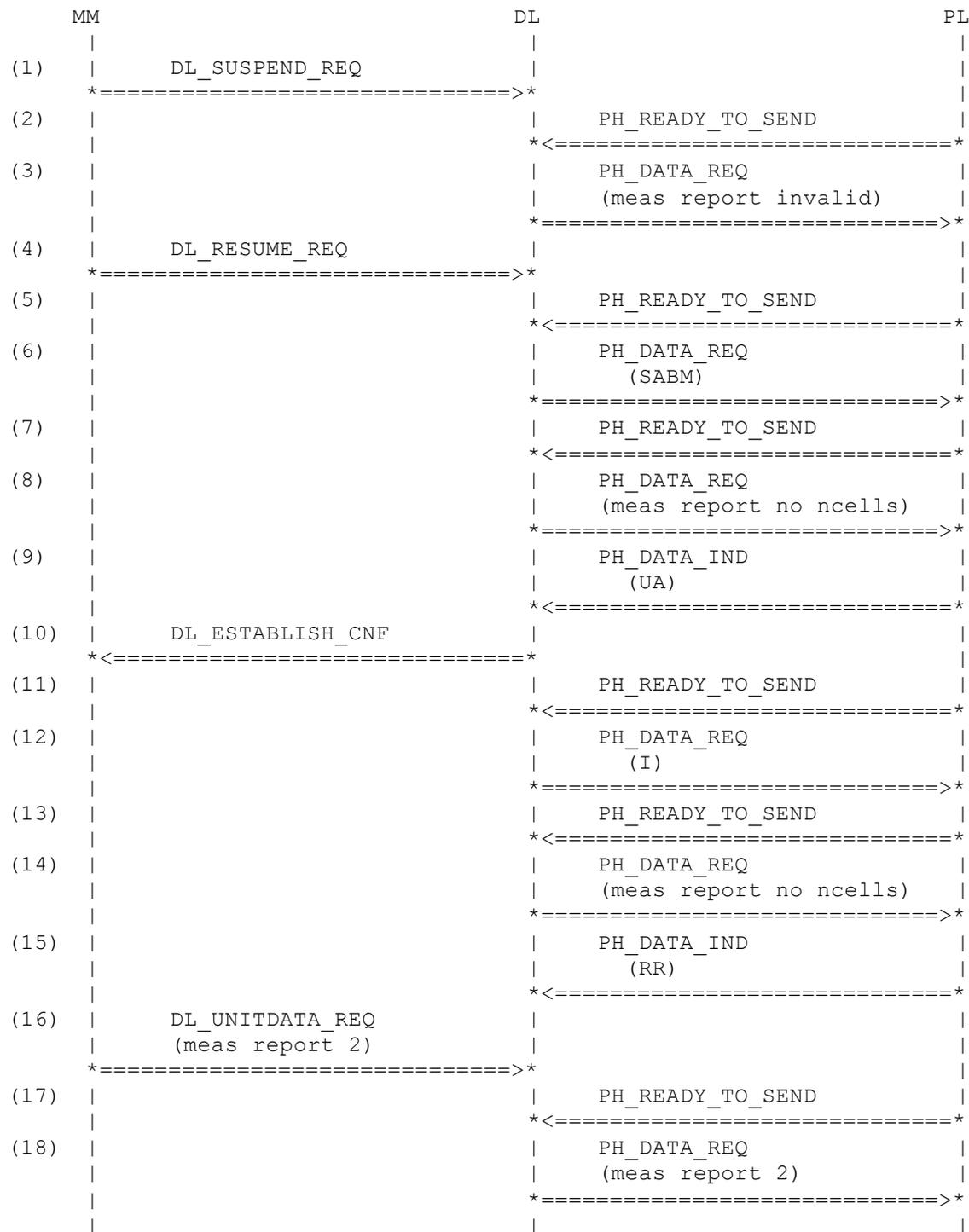
	sapi sdu	SAPI_0 L3_MSG_L06_UL
(2)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(3)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_B_I06_UL
(4)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_B_I06_1_DL
(5)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(6)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_1
(7)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(8)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(9)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(10)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(11)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1
(12)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(13)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1
(14)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_1
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(16)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1

History: 21.03.97 DL Initial

4.16.2 DL161: Suspend and Resume Multiframe operation (SDCCH, Variant I)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after three SACCH uplink frames requested by layer 1.

Preamble: DL160



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0

(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(4)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_L06_UL
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(10)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED L106_UL
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(15)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL
(16)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(17)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH

(18)

PH_DATA_REQ
ch_type
dummy
sdu

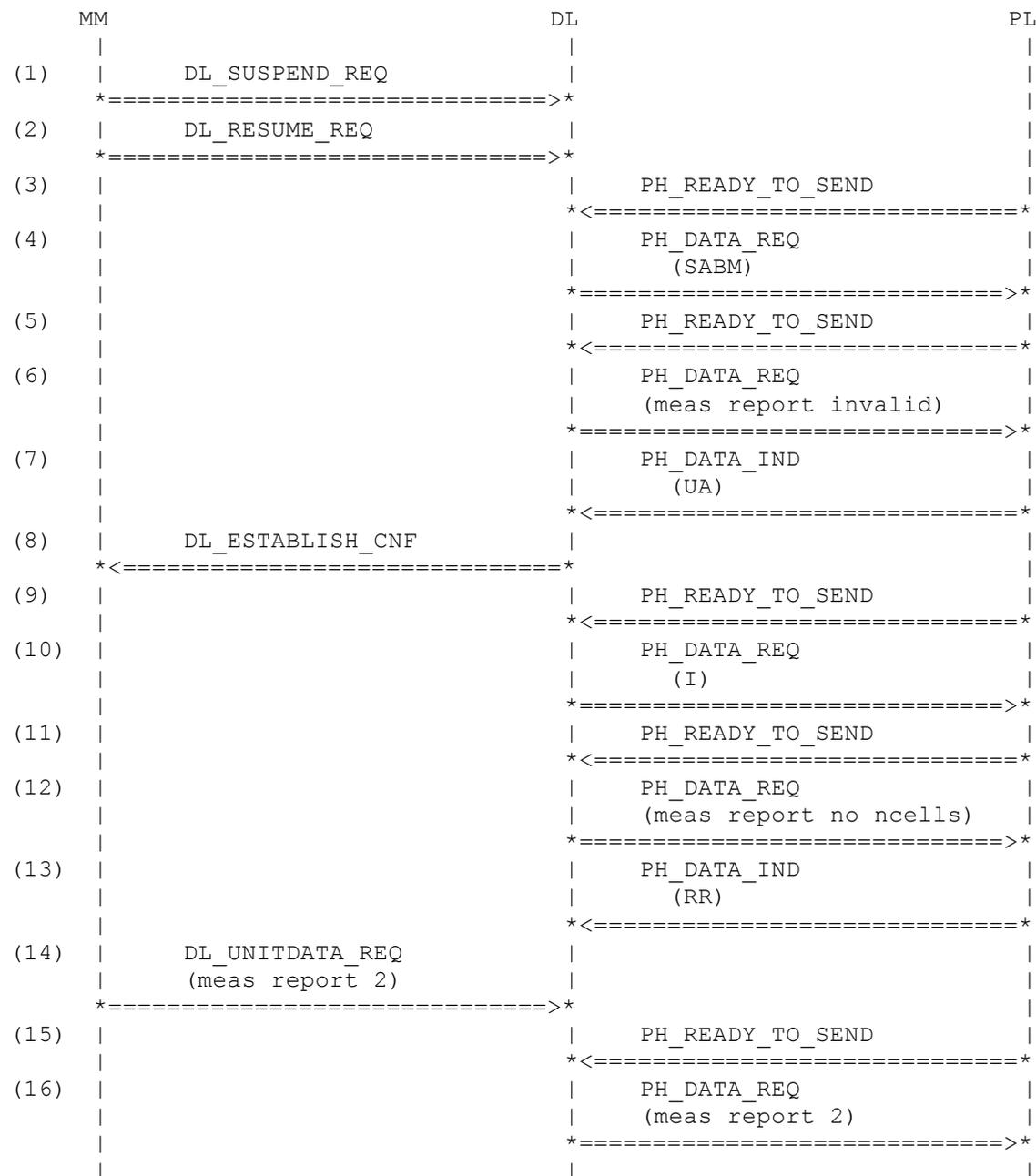
CH_TYPE_SACCH
NOT_USED
MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.3 DL162: Suspend and Resume Multiframe operation (SDCCH, Variant II)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after two SACCH uplink frames requested by layer 1. There is no request between suspension and resumption.

Preamble: DL160



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(2) DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH

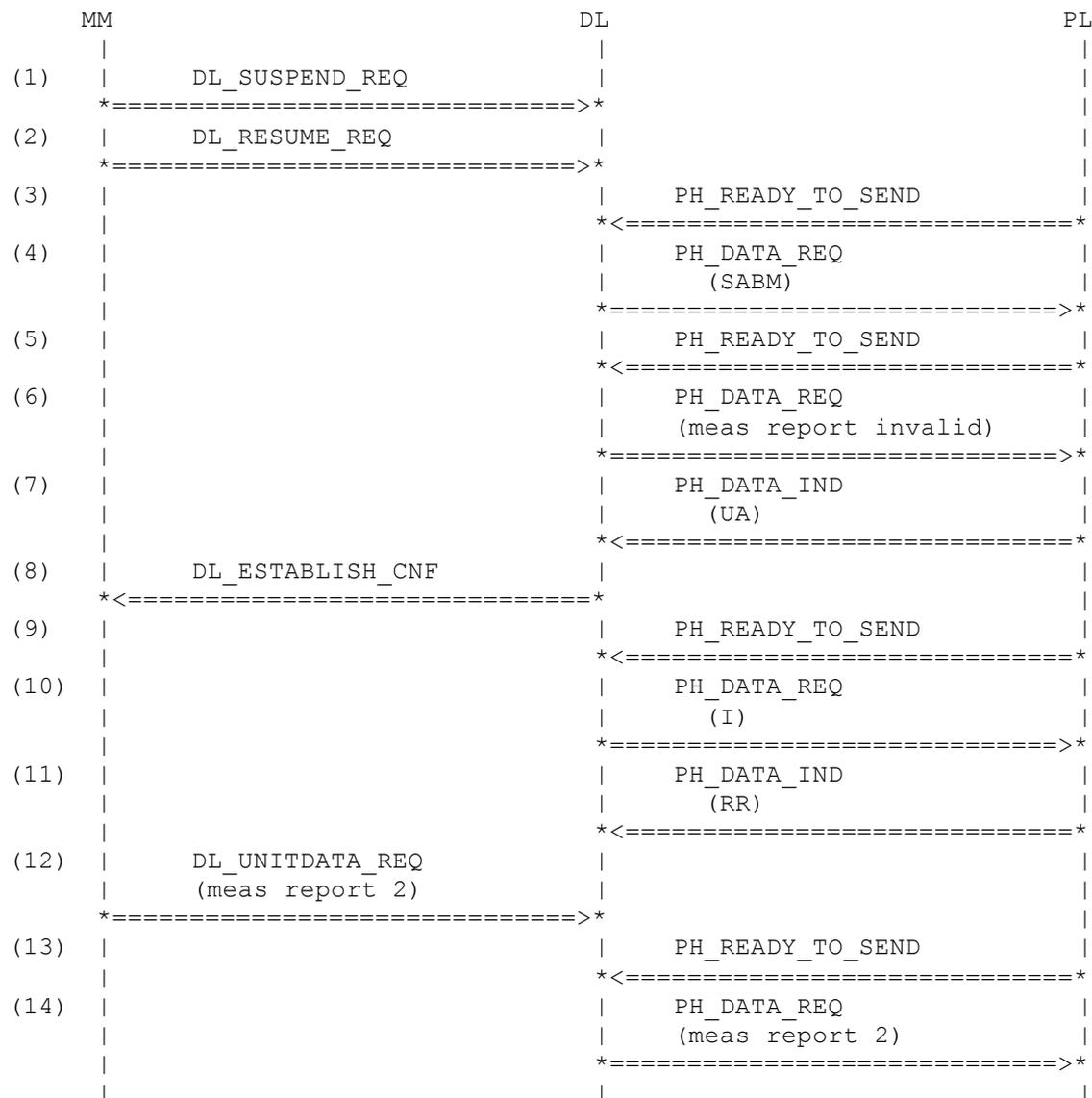
	sapi sdu	SAPI_0 L3_MSG_L06_UL
(3)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(4)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(5)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(6)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(7)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(8)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(9)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED L_I06_UL
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(12)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(13)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL
(14)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(16)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.4 DL163: Suspend and Resume Multiframe operation (SDCCH, Variant III)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after one SACCH uplink frames requested by layer 1. There is no request between suspension and resumption.

Preamble: DL160



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(2) DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH

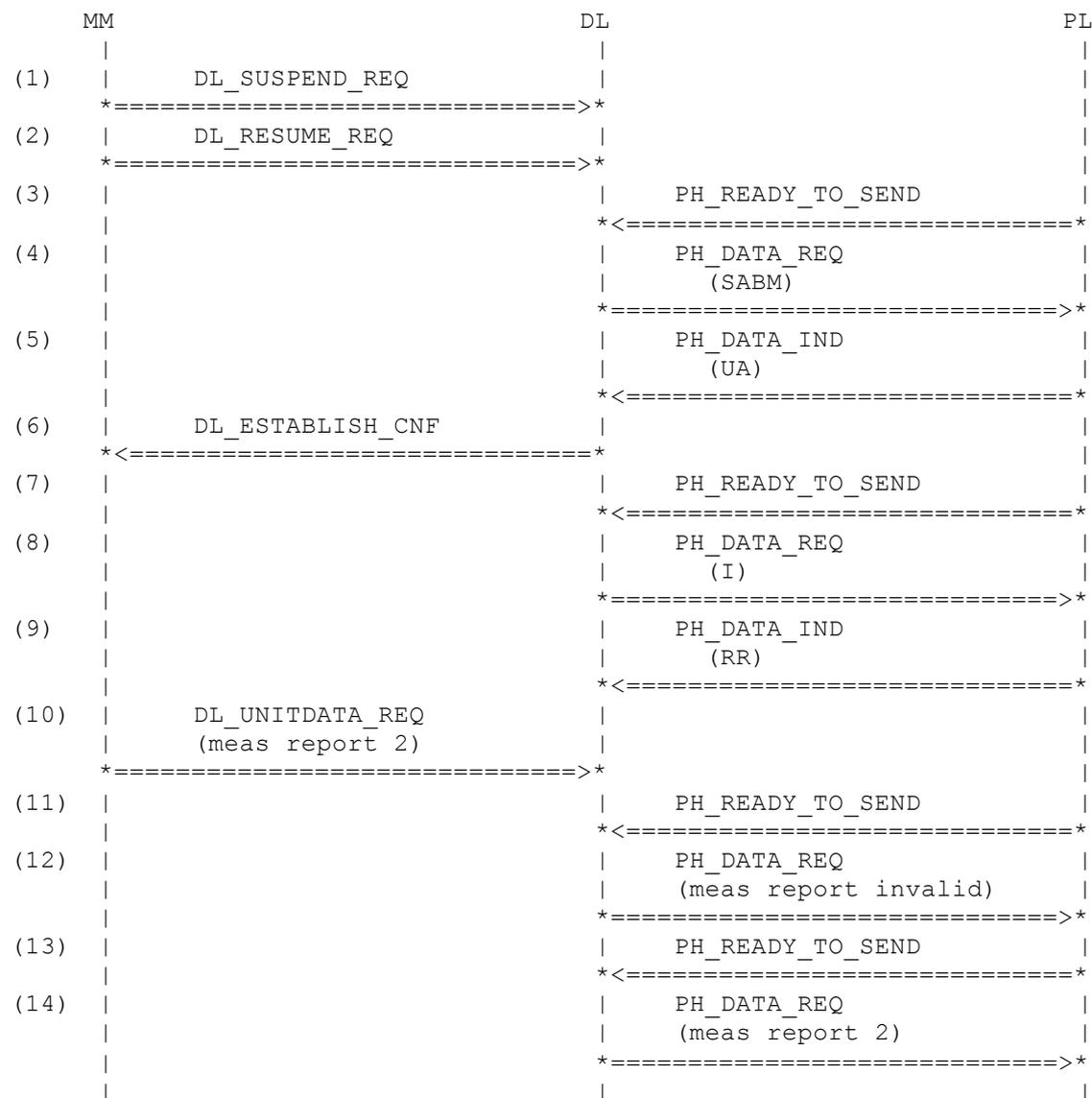
(4)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(7)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(8)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_106_UL
(11)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL
(12)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.5 DL164: Suspend and Resume Multiframe operation (SDCCH, Variant IV)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL before request of layer 1.

Preamble: DL160



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(2) DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

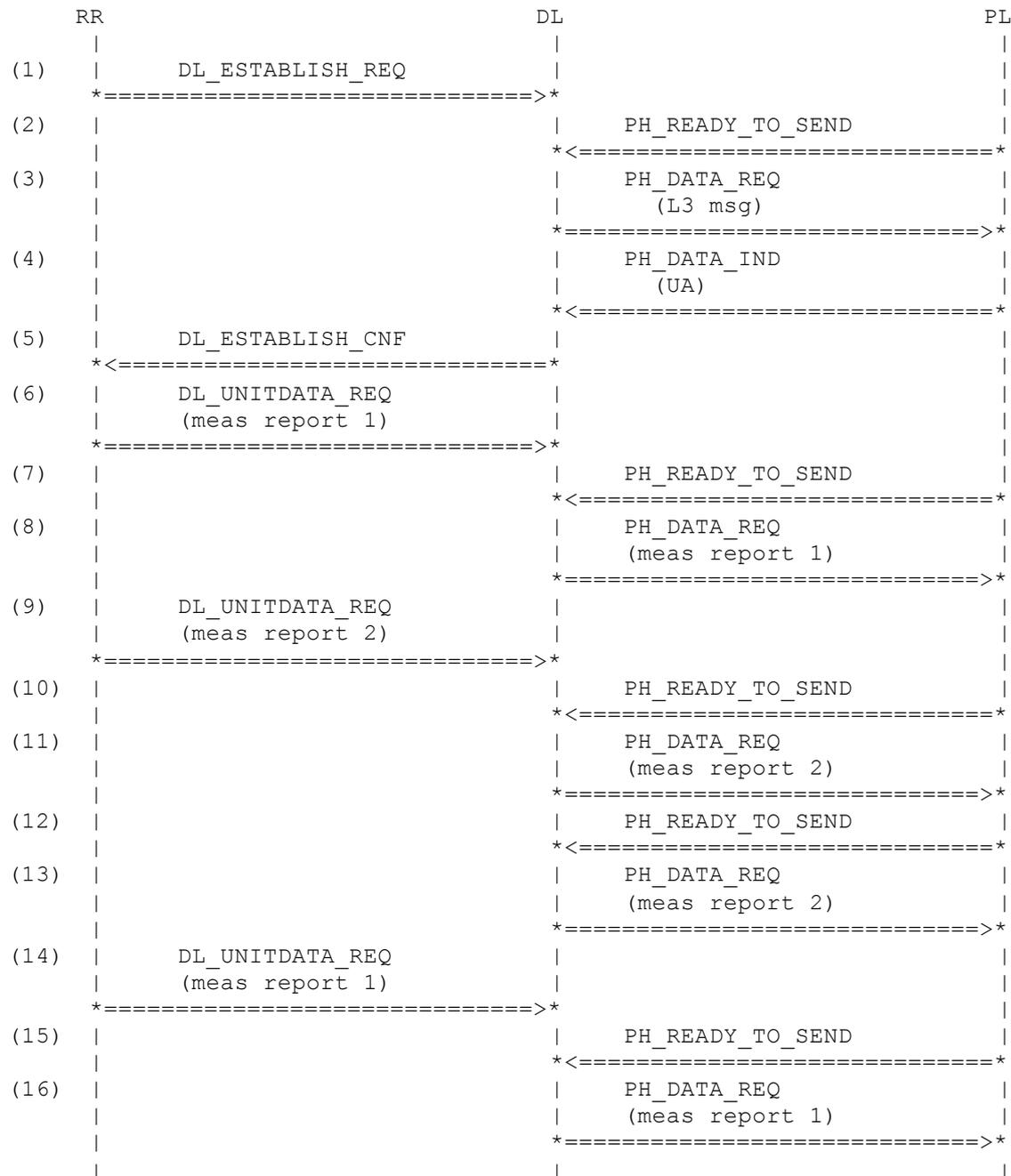
		dummy sdu	NOT_USED SABM_A_UL
(5)	PH_DATA_IND		
		ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(6)	DL_ESTABLISH_CNF		
		ch_type sapi	CH_TYPE_SDCCH SAPI_0
(7)	PH_READY_TO_SEND		
		ch_type	CH_TYPE_SDCCH
(8)	PH_DATA_REQ		
		ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_106_UL
(9)	PH_DATA_IND		
		ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR1_DL
(10)		DL_UNITDATA_REQ	
		ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(11)		PH_READY_TO_SEND	
		ch_type	CH_TYPE_SACCH
(12)		PH_DATA_REQ	
		ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(13)		PH_READY_TO_SEND	
		ch_type	CH_TYPE_SACCH
(14)		PH_DATA_REQ	
		ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.6 DL170: Repetition of last Measurement Report (main link = FACCH)

Description: A sapi 0 link is established for FACCH. RR forwards measurement reports to DL. Layer 1 requests the measurement reports for SACCH. If RR has coded no new measurement reports in time, the last received measurement report from RR is repeated.

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL

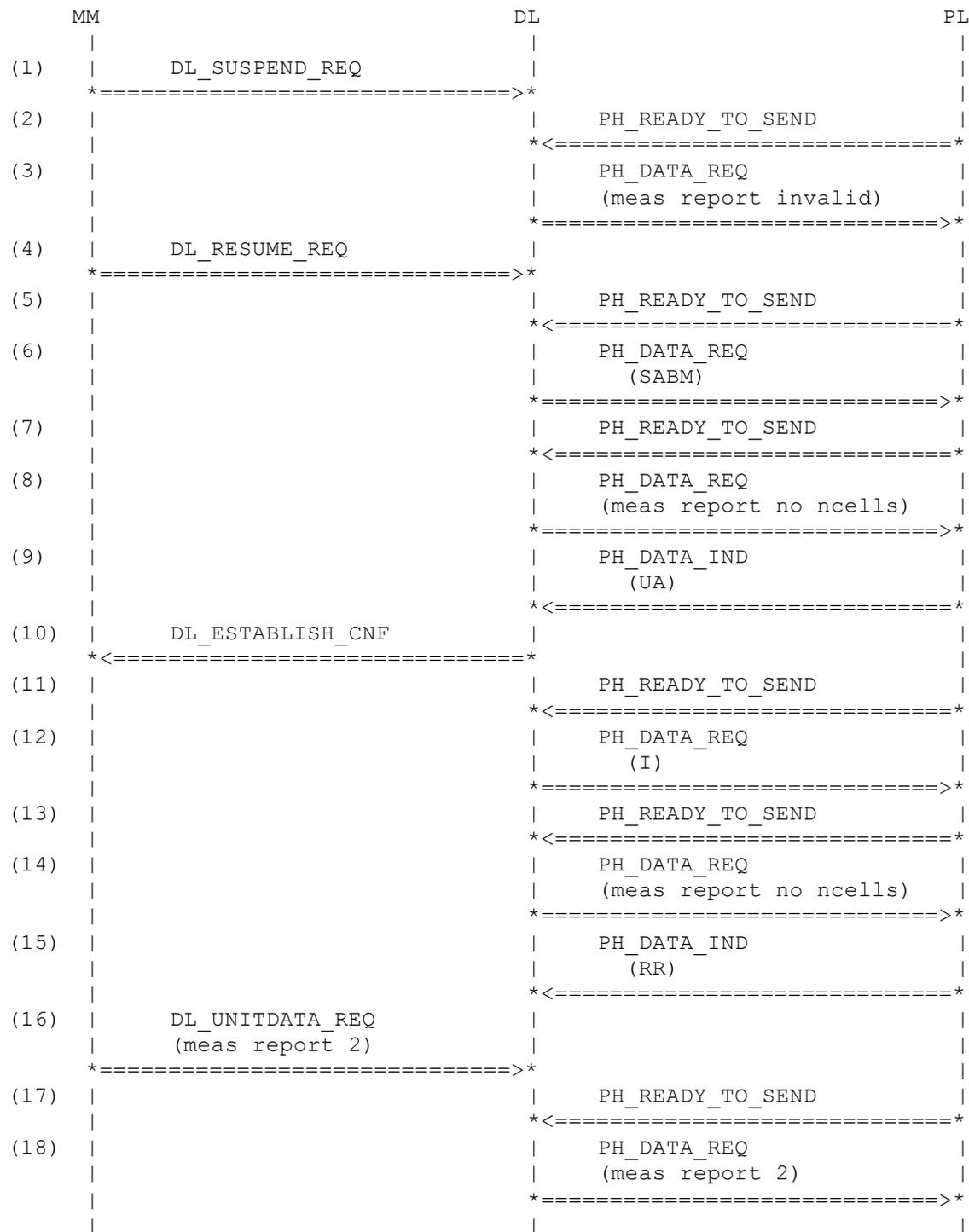
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_B_I06_UL
(4)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_B_I06_1_DL
(5)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(6)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_1
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(9)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(11)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(13)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1
(14)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_1
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(16)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1

History: 21.03.97 DL Initial

4.16.7 DL171: Suspend and Resume Multiframe operation (FACCH, Variant I)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after three SACCH uplink frames requested by layer 1.

Preamble: DL170



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_FACCH SAPI_0

(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(4)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_MSG_L06_UL
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(10)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED L106_UL
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(15)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL
(16)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(17)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH

(18)

PH_DATA_REQ
ch_type
dummy
sdu

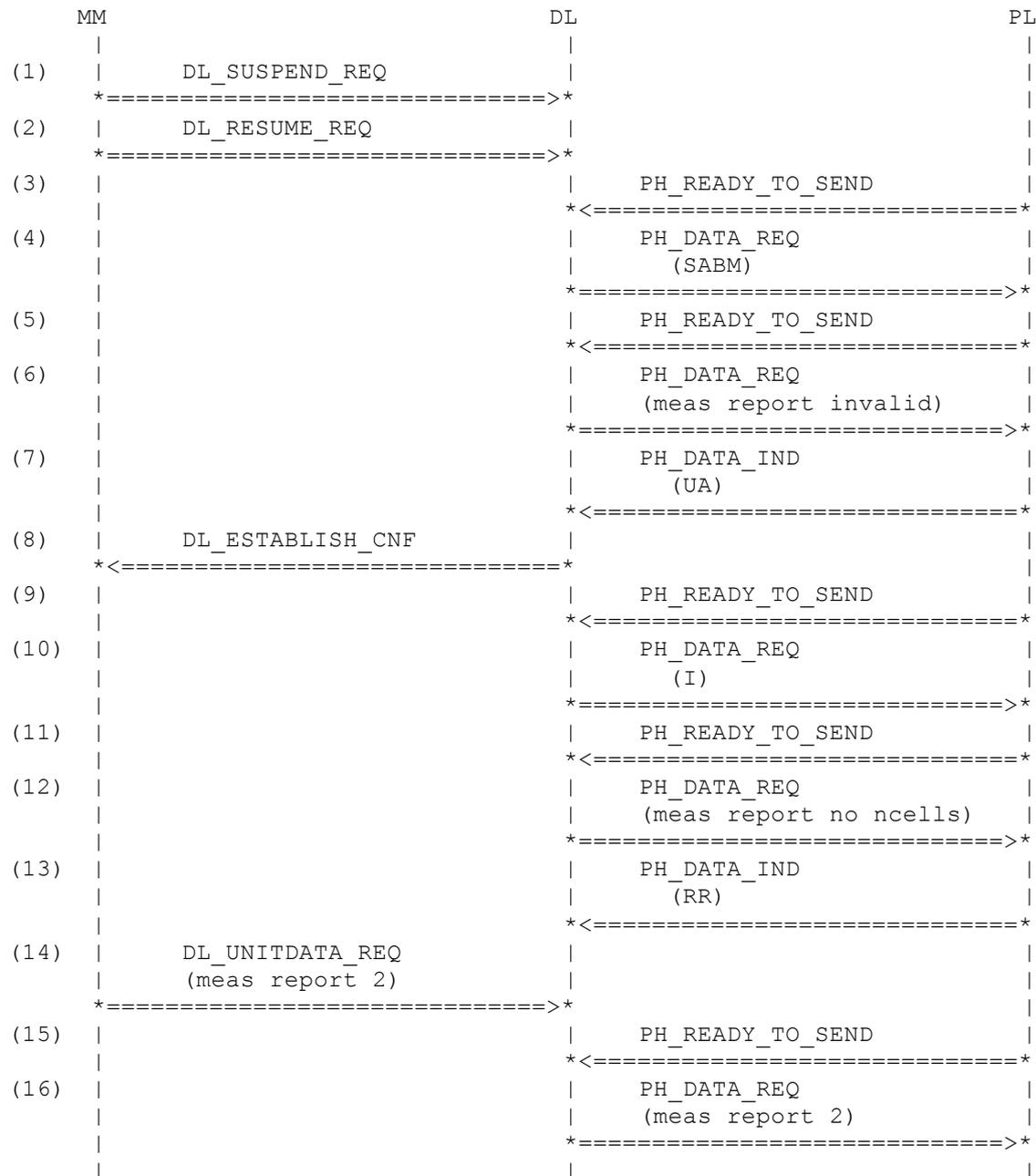
CH_TYPE_SACCH
NOT_USED
MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.8 DL172: Suspend and Resume Multiframe operation (SDCCH, Variant II)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after two SACCH uplink frames requested by layer 1. There is no request between suspension and resumption.

Preamble: DL170



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(2) DL_RESUME_REQ	ch_type	CH_TYPE_FACCH

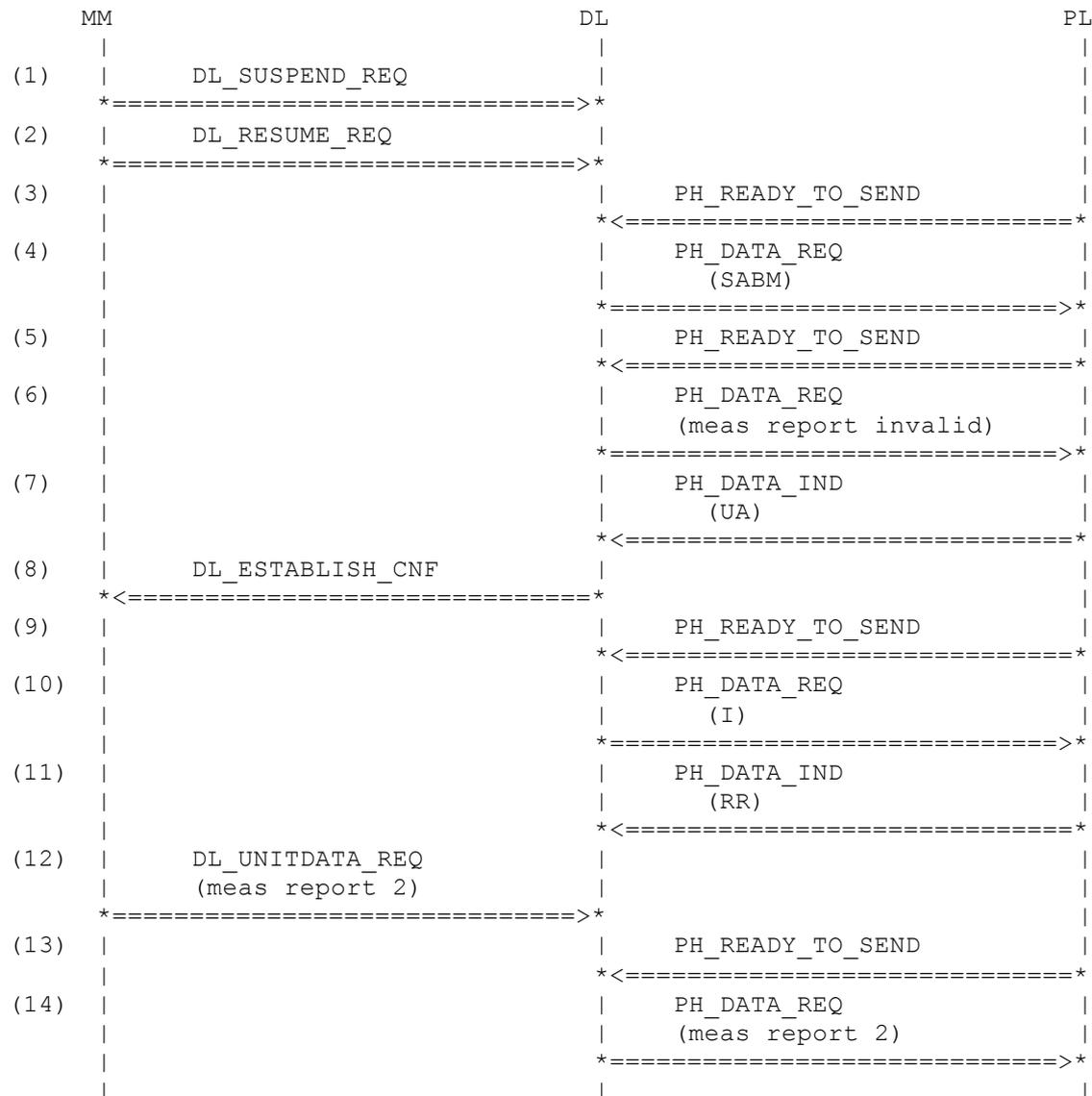
	sapi sdu	SAPI_0 L3_MSG_L06_UL
(3)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(4)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(5)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(6)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(7)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(8)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_FACCH SAPI_0
(9)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED L_I06_UL
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(12)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_NO_NCELL
(13)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL
(14)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(16)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.9 DL173: Suspend and Resume Multiframe operation (FACCH, Variant III)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL after one SACCH uplink frames requested by layer 1. There is no request between suspension and resumption.

Preamble: DL170



Parametrization

	Primitive	Parameter	Value
(1)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_FACCH SAPI_0
(2)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_MSG_L06_UL
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

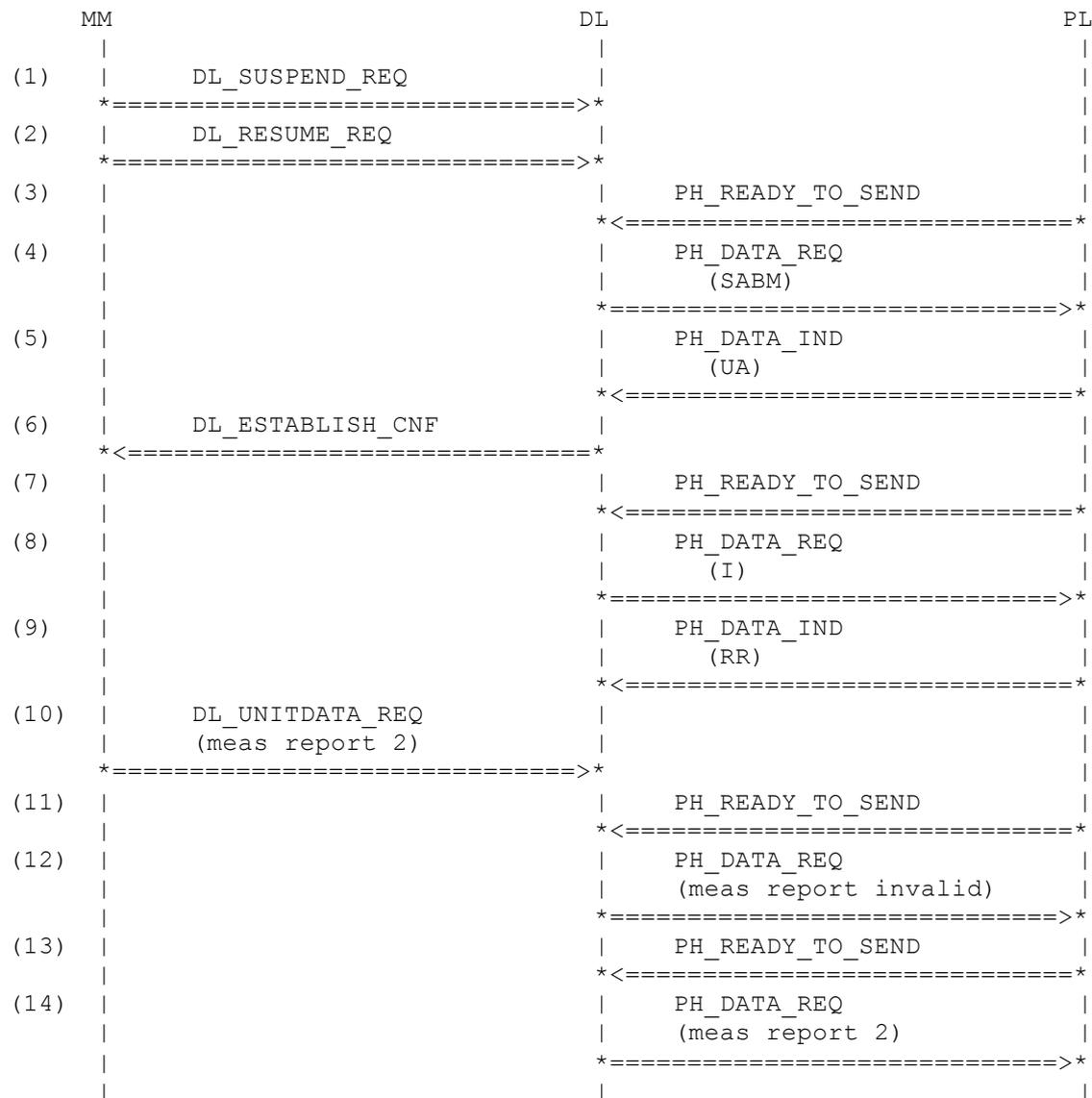
(4)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(7)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(8)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_106_UL
(11)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL
(12)	DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

History: 21.03.97 DL Initial

4.16.10 DL174: Suspend and Resume Multiframe operation (FACCH, Variant IV)

Description: The SAPI 0 link is suspended and resumed. In this variant the first measurement report for the new channel is forwarded to DL before request of layer 1.

Preamble: DL170



Parametrization

Primitive	Parameter	Value
(1) DL_SUSPEND_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
(2) DL_RESUME_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_MSG_L06_UL
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH

	dummy sdu	NOT_USED SABM_A_UL
(5)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(6)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_FACCH SAPI_0
(7)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(8)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_106_UL
(9)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_A_NR1_DL
(10)	DL_UNITDATA_REQ	
	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_2
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(12)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_INV_L1
(13)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_2_L1

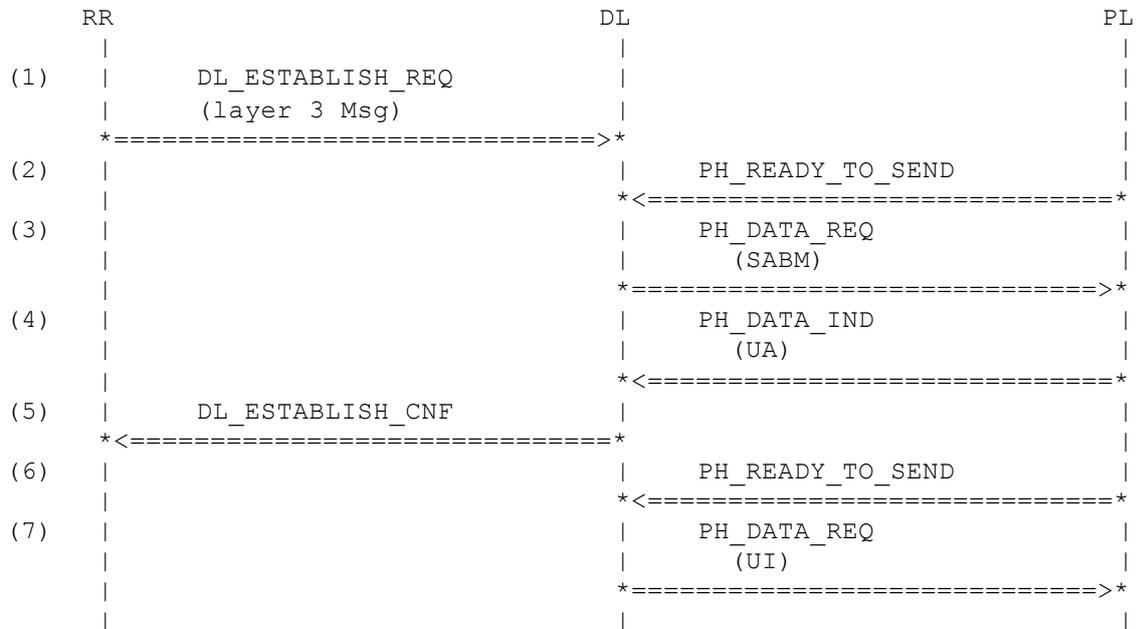
History: 21.03.97 DL Initial

4.17 GSM 11.10 Test Cases

4.17.1 DL200: Normal Initialization (25.2.1.1.1), SDCCH

Description: To test the normal establishment of multiple frame operation between the SS and the MS when contention resolution is required.

Preamble: DL000



Parametrization

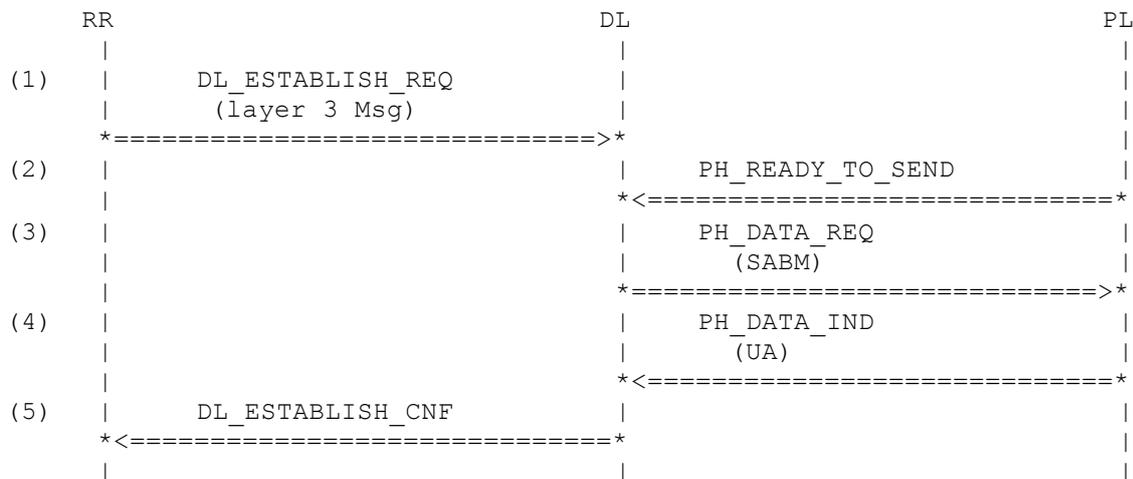
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.2 DL201: Normal Initialization (25.2.1.1.1), FACCH Fullrate

Description: To test the normal establishment of multiple frame operation between the SS and the MS when contention resolution is required.

Preamble: DL000



Parametrization

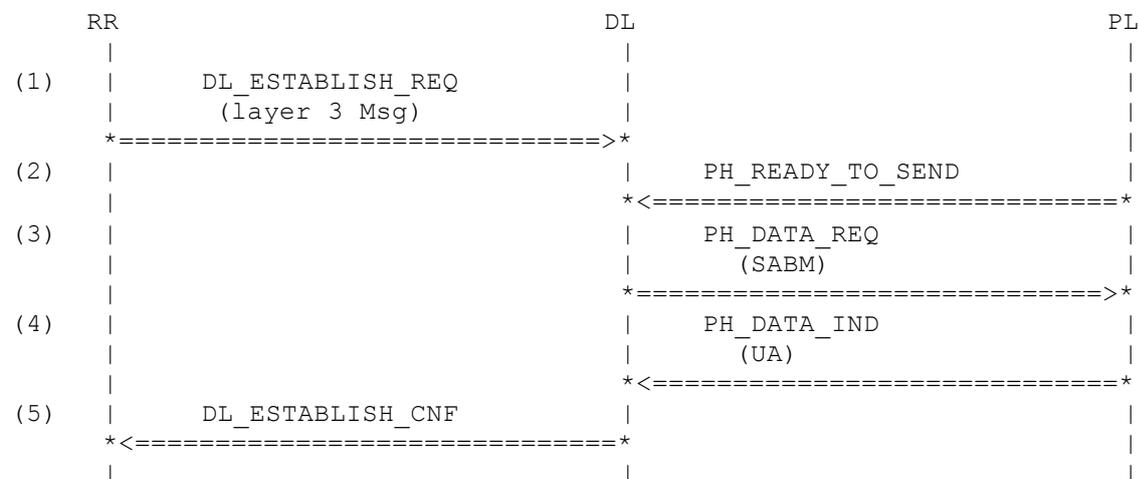
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0

History: 05.12.97 LE Initial

4.17.3 DL601: Normal Initialization (25.2.1.1.1), FACCH Halfrate

Description: To test the normal establishment of multiple frame operation between the SS and the MS when contention resolution is required.

Preamble: DL000



Parametrization

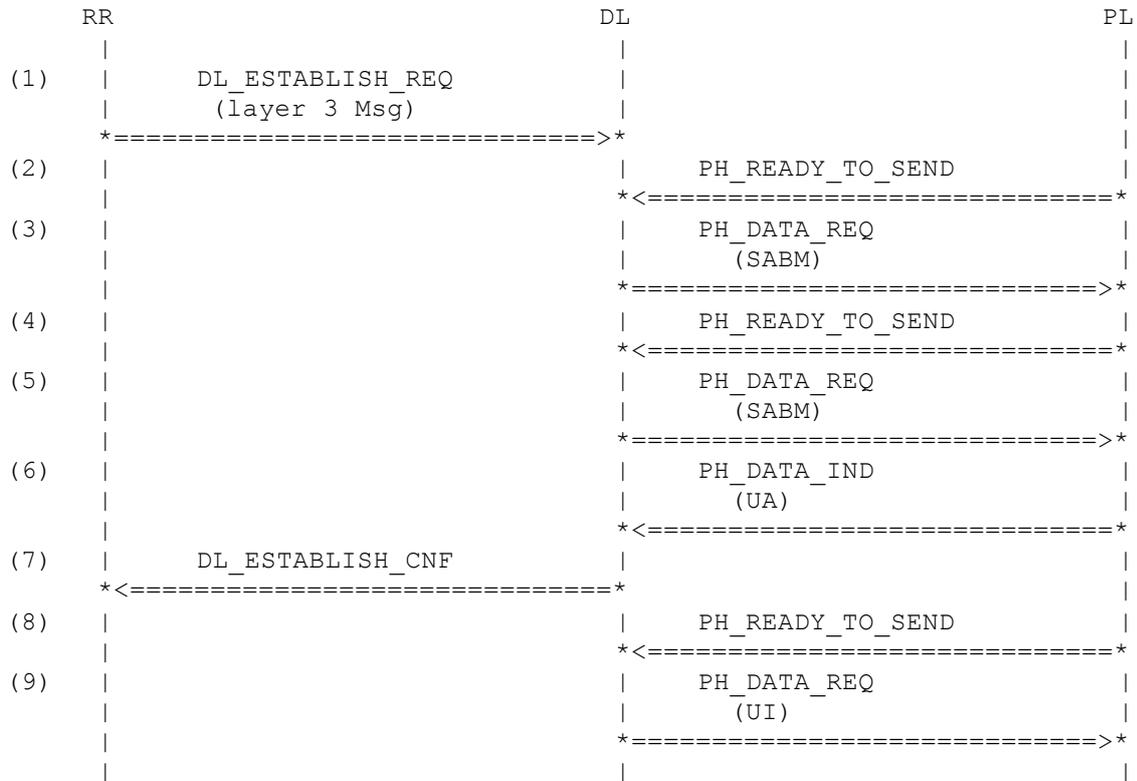
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(5) DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0

History: 08.07.98 LE Initial

4.17.4 DL202: Loss of UA frame (25.2.1.1.2.1), SDCCH

Description: To test the MS response to the loss of a Layer 2 UA frame during initialization.

Preamble: DL000



Parametrization

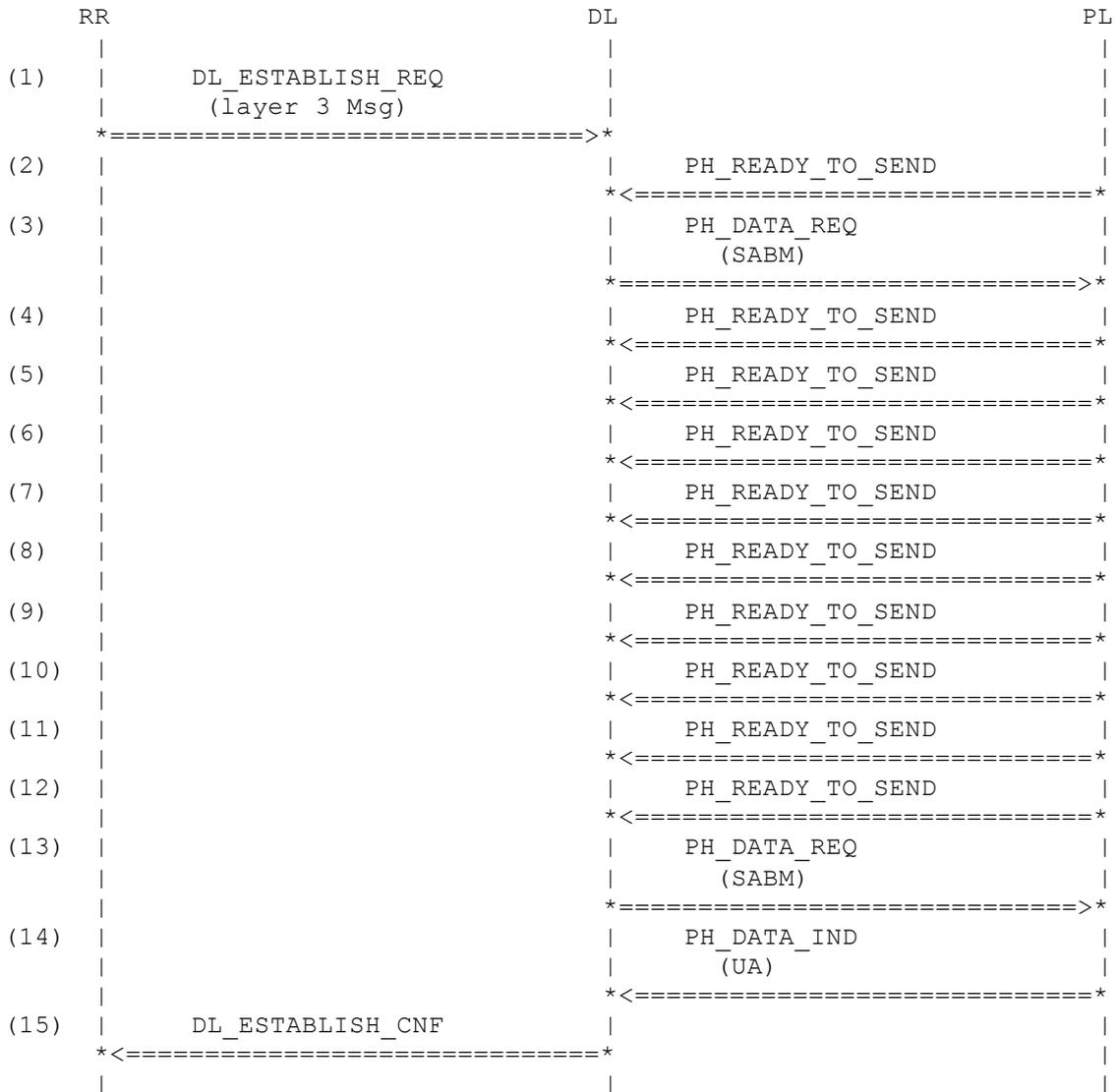
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(6) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(7) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
(8) PH_READY_TO_SEND	sapi	SAPI_0
	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.5 DL203: Loss of UA frame (25.2.1.1.2.1), FACCH Fullrate

Description: To test the MS response to the loss of a Layer 2 UA frame during initialization.

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy		NOT_USED
	sdu		SABM_PAGING_RESP
(14)	PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy		NOT_USED
	sdu		UA_PAGING_RESP
(15)	DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH
	sapi		SAPI_0

History: 05.12.97 LE Initial

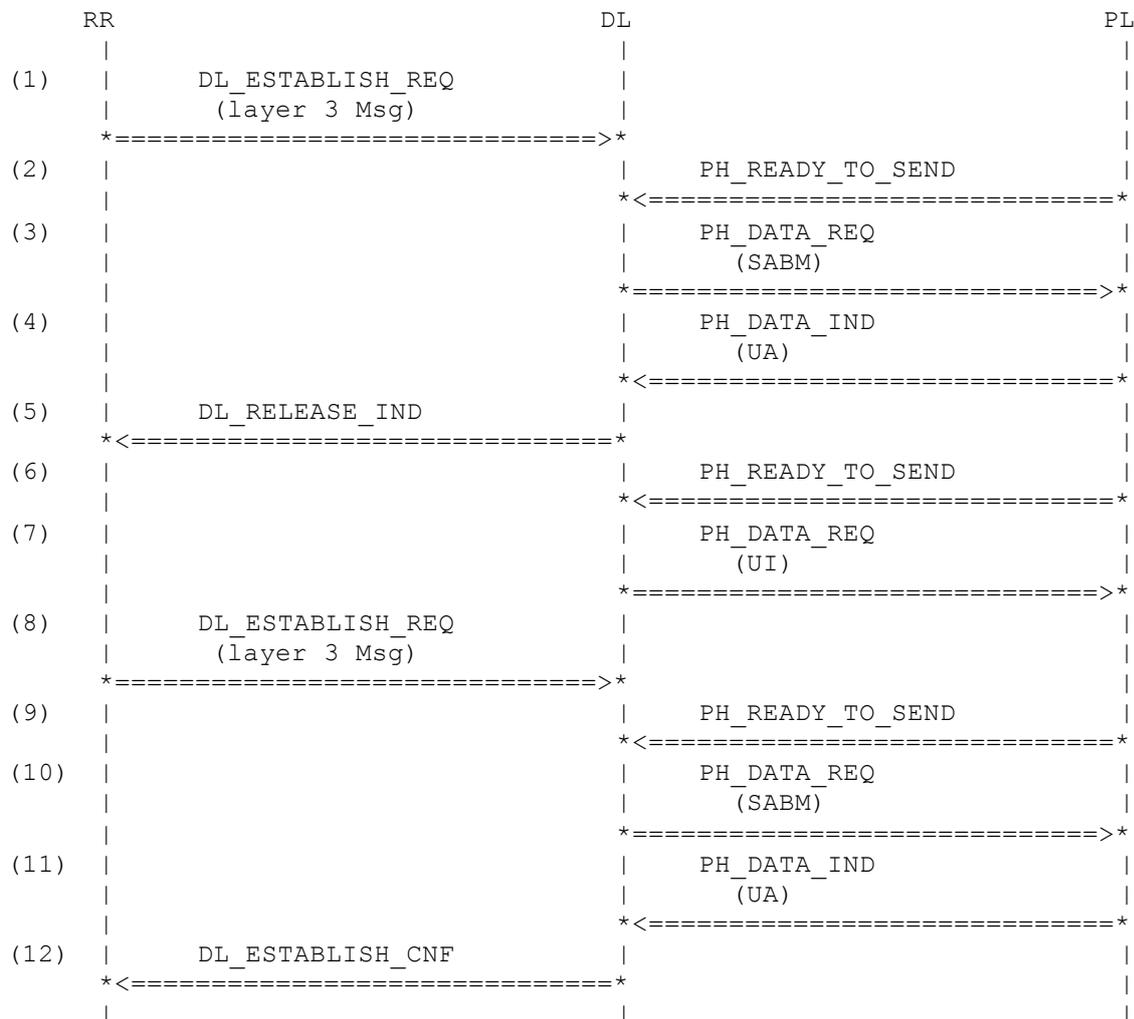
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(13)	PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy		NOT_USED
	sdu		SABM_PAGING_RESP
(14)	PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy		NOT_USED
	sdu		UA_PAGING_RESP
(15)	DL_ESTABLISH_CNF	ch_type	CH_TYPE_FACCH_HR
	sapi		SAPI_0

History: 08.07.98 LE Initial

4.17.7 DL204: UA frame with different information field (25.2.1.1.2.2), SDCCH

Description: To test that the MS will leave the channel and return to the idle state when multiple frame establishment fails because a UA frame with a different information field is received in response to the SABM frame.

Preamble: DL000



Parametrization

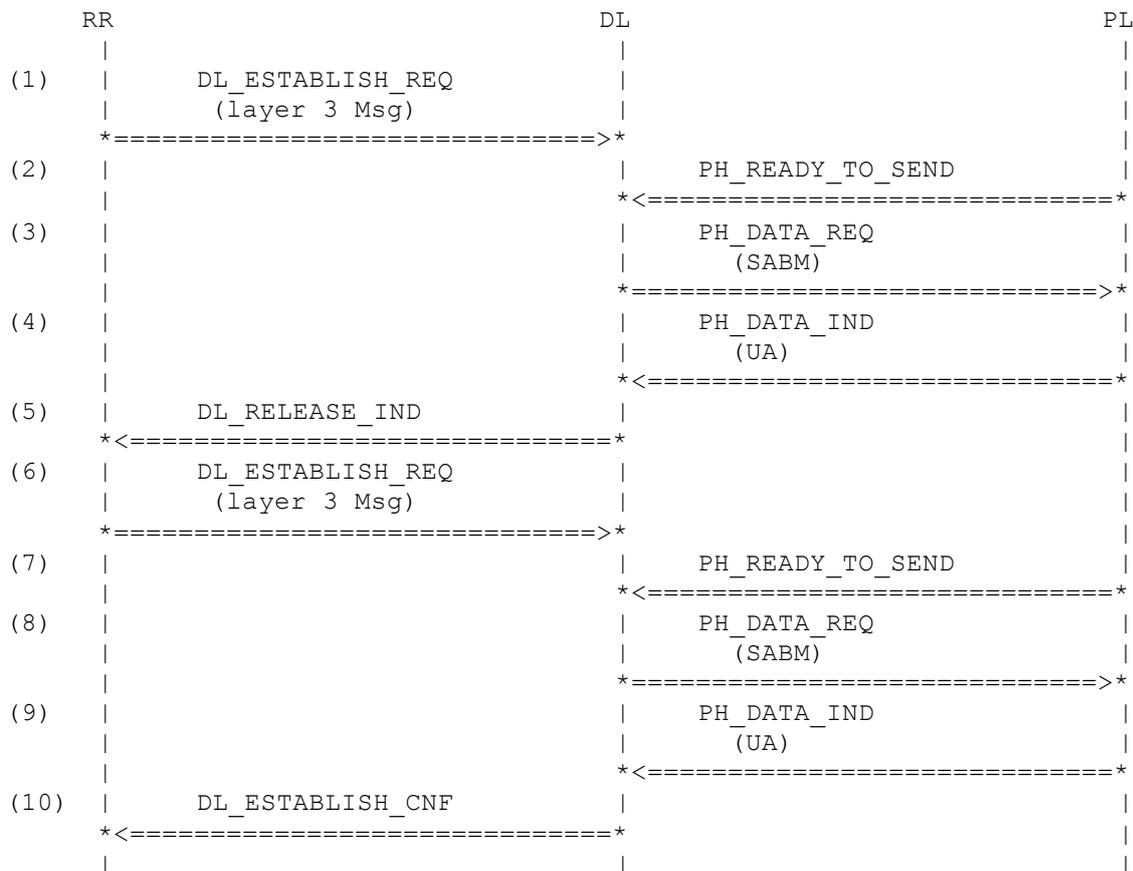
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_NO_PAGING_RESP
(5) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_UPLINK
(8) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(11) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(12) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0

History: 05.12.97 LE Initial

4.17.8 DL205: UA frame with different info field (25.2.1.1.2.2), FACCH Fullrate

Description: To test that the MS will leave the channel and return to the idle state when multiple frame establishment fails because a UA frame with a different information field is received in response to the SABM frame.

Preamble: DL000



Parametrization

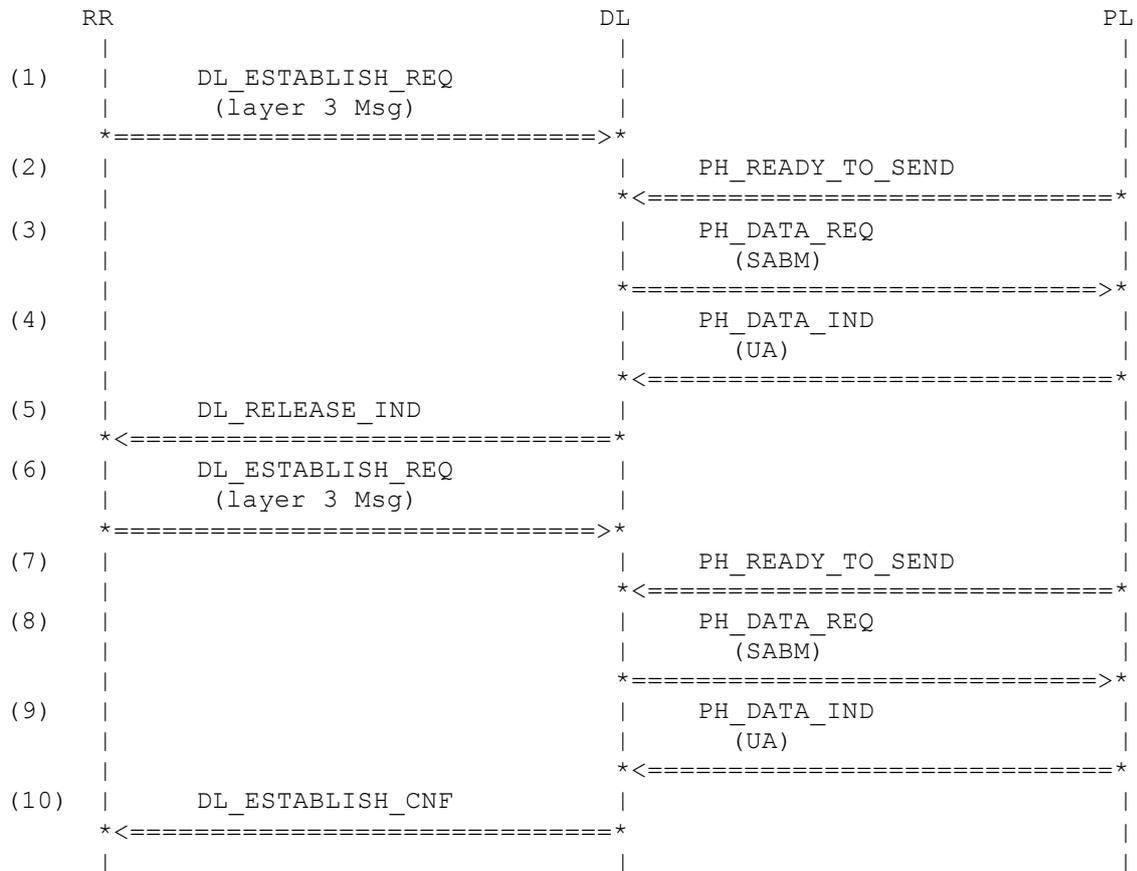
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_NO_PAGING_RESP
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH
(6) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(9) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(10)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_FACCH SAPI_0

History: 05.12.97 LE Initial

4.17.9 DL605: UA frame with different info field (25.2.1.1.2.2), FACCH Halfrate

Description: To test that the MS will leave the channel and return to the idle state when multiple frame establishment fails because a UA frame with a different information field is received in response to the SABM frame.

Preamble: DL000



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_NO_PAGING_RESP
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	INFO_FIELD_MISMATCH
(6) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(9) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(10)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0

History: 08.07.98 LE Initial

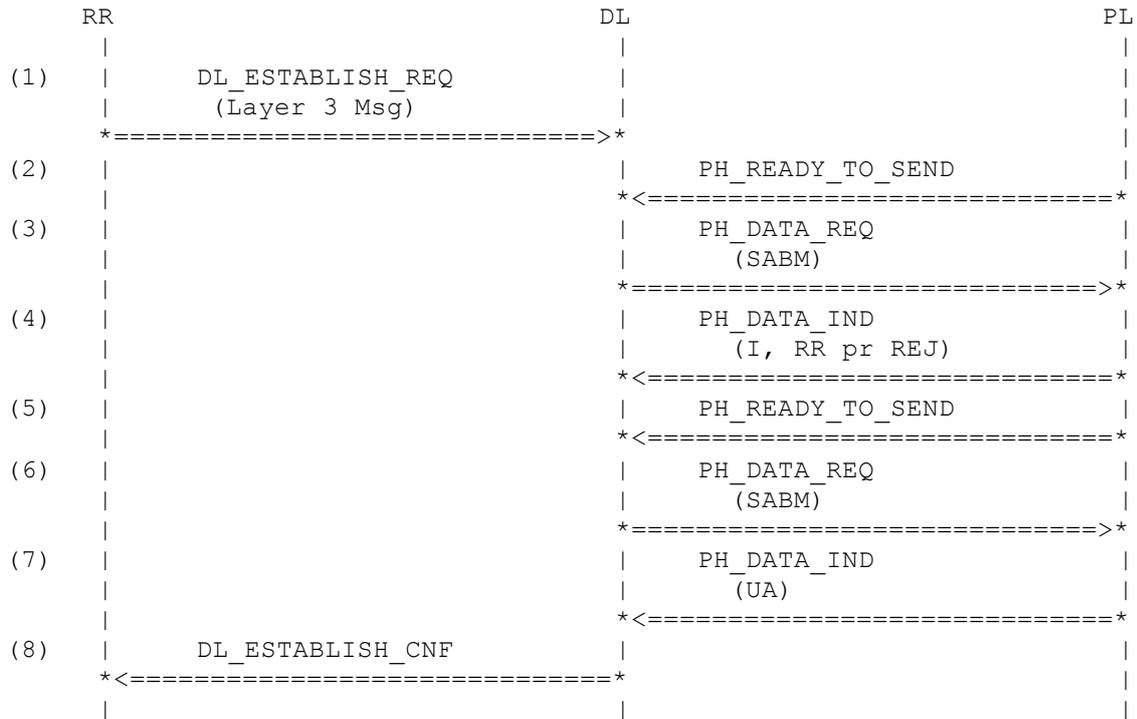
4.17.10 DL206: Wrong frames in response to SABM (25.2.1.1.2.2), SDCCH

Description: To test that the MS will ignore receipt of frames other than a UA when received in response to the SABM frame.

- A: I frame
- B: RR frame
- C: REJ frame

Preamble: DL000

Variants: <A> ... <C>



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	<A> sdu	I_FRAME_1
	 sdu	RR_FRAME_1
<C> sdu	REJ_FRAME_1	
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(7) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(8) DL_ESTABLISH_CNF	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0

History: 05.12.97 LE Initial

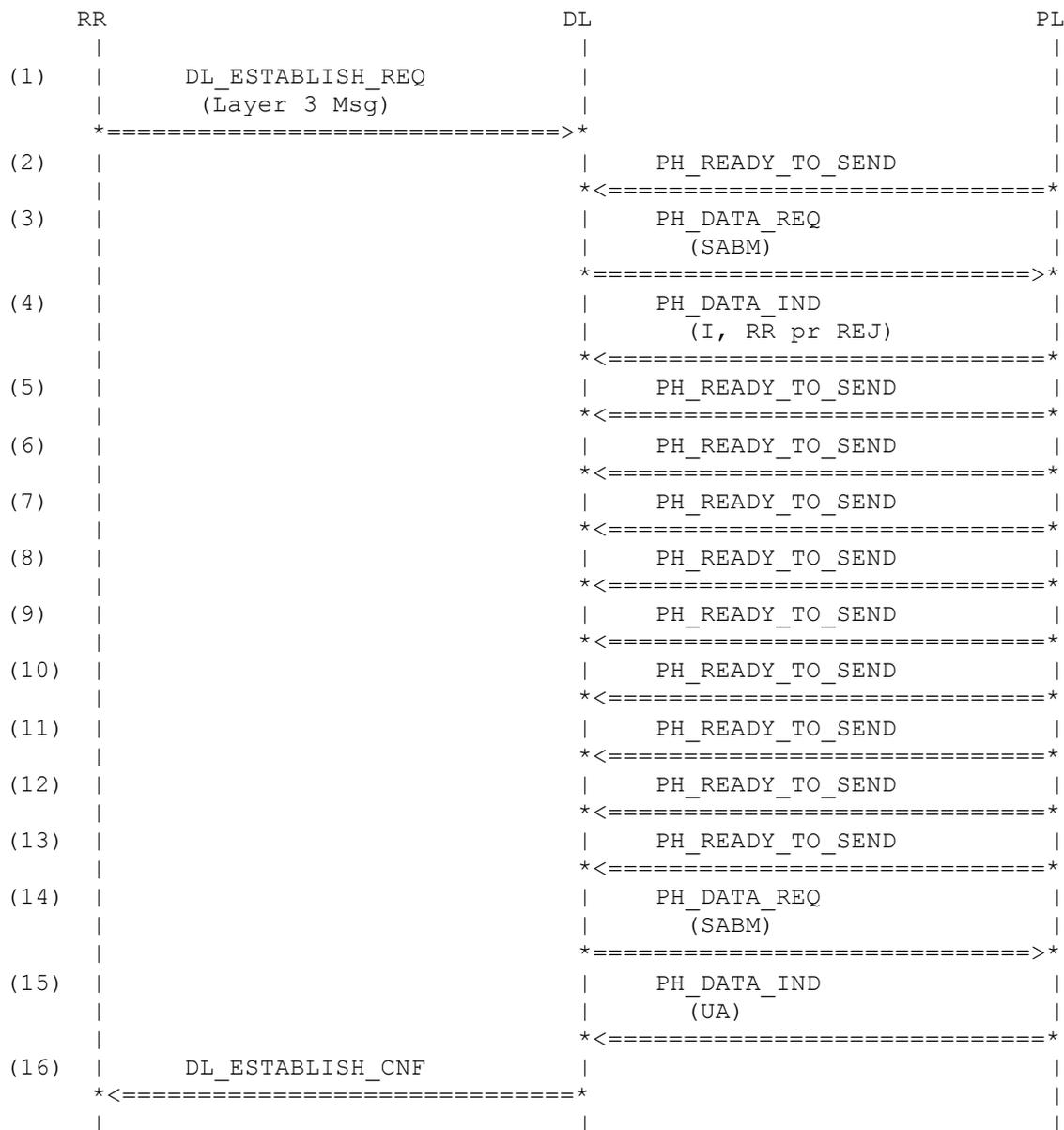
4.17.11 DL207: Wrong frames in response to SABM (25.2.1.1.2.2), FACCH Fullrate

Description: To test that the MS will ignore receipt of frames other than a UA when received in response to the SABM frame.

- A: I frame
- B: RR frame
- C: REJ frame

Preamble: DL000

Variants: <A> ... <C>



Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	<A> sdu	I_FRAME_1
	 sdu	RR_FRAME_1
	<C> sdu	REJ_FRAME_1
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(14)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy sdu	NOT_USED SABM_PAGING_RESP
(15)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH
	dummy sdu	NOT_USED UA_PAGING_RESP
(16)	DL_ESTABLISH_CNF	
	ch_type sapi	CH_TYPE_FACCH SAPI_0

History: 05.12.97 LE Initial

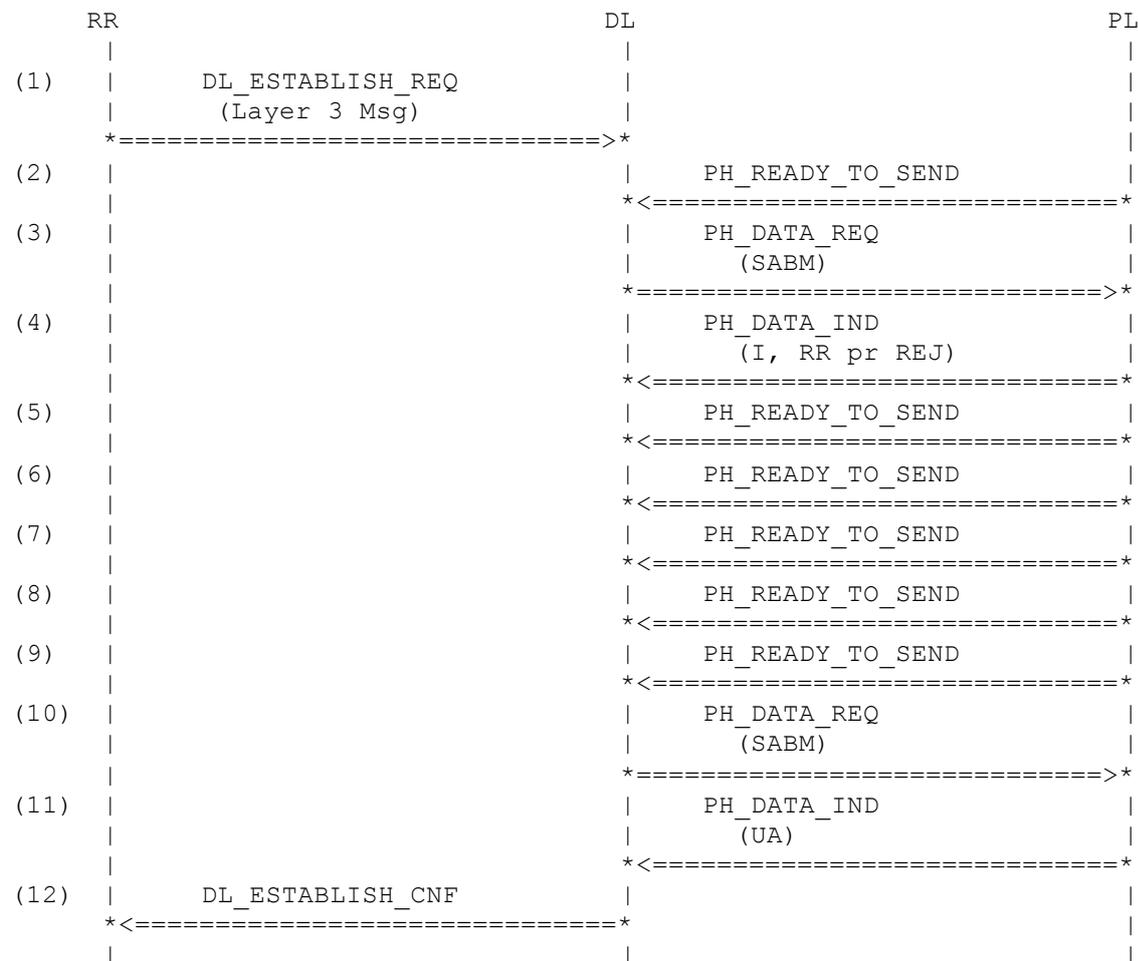
4.17.12 DL607: Wrong frames in response to SABM (25.2.1.1.2.2), FACCH Halfrate

Description: To test that the MS will ignore receipt of frames other than a UA when received in response to the SABM frame.

- A: I frame
- B: RR frame
- C: REJ frame

Preamble: DL000

Variants: <A> ... <C>



Parametrization

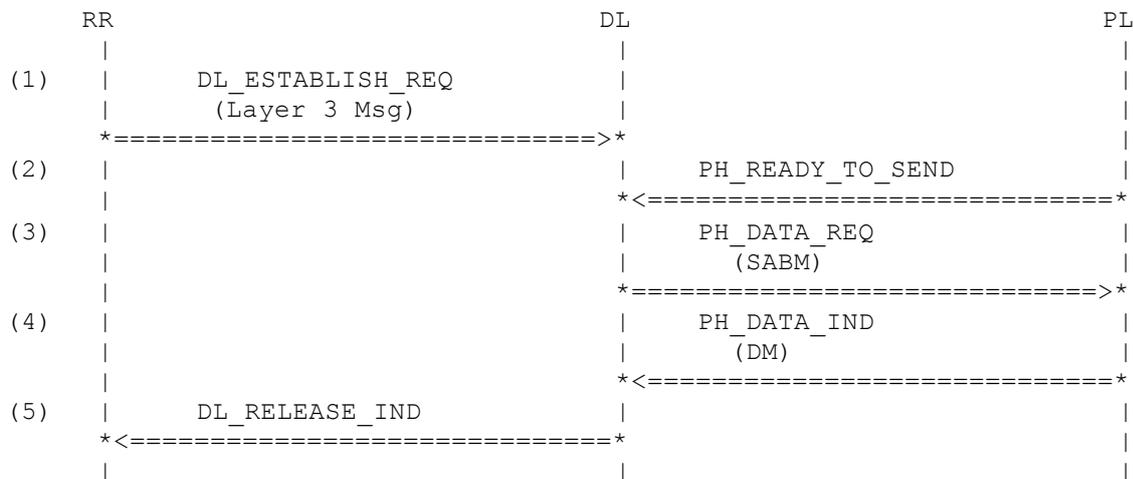
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	<A> sdu	I_FRAME_1
	 sdu	RR_FRAME_1
	<C> sdu	REJ_FRAME_1
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
(11)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
(12)	DL_ESTABLISH_CNF	
	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0

History: 08.07.98 LE Initial

4.17.13 DL208: Initialisation denial (25.2.1.1.3), SDCCH

Description: To test that the MS takes appropriate action if the network side indicates that it can not enter the multiple frame established state.

Preamble: DL000



Parametrization

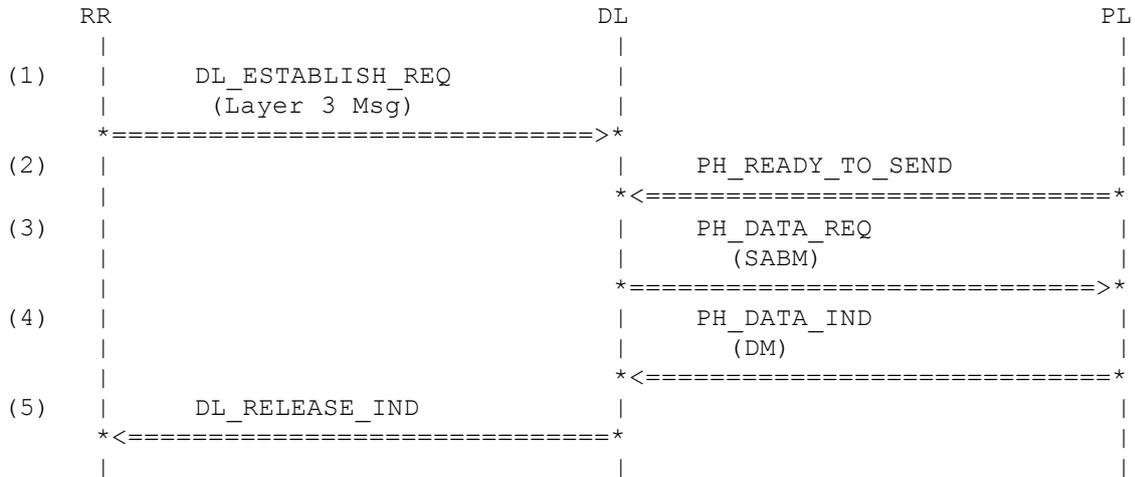
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DM_FRAME
(5) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 05.12.97 LE Initial

4.17.14 DL209: Initialisation denial (25.2.1.1.3), FACCH Fullrate

Description: To test that the MS takes appropriate action if the network side indicates that it can not enter the multiple frame established state.

Preamble: DL000



Parametrization

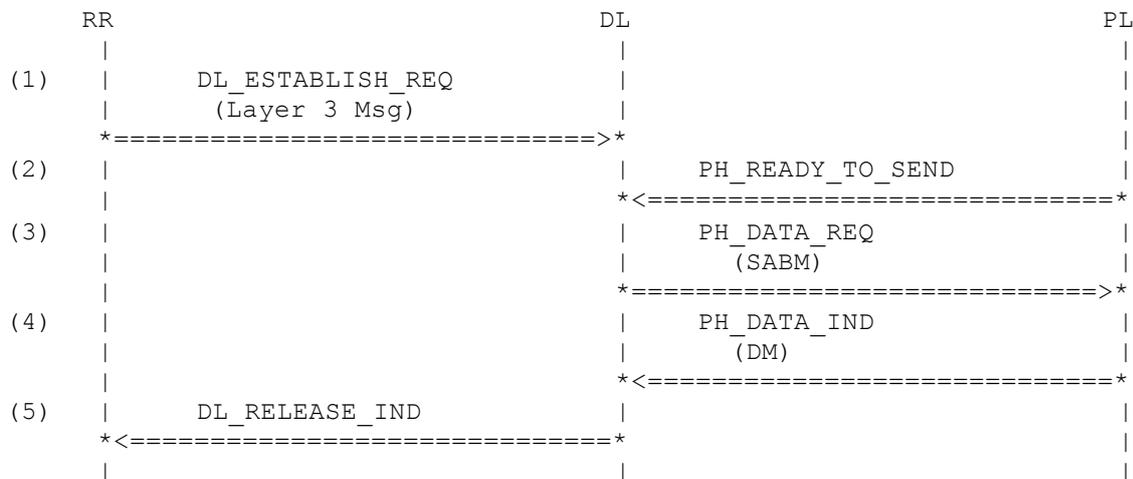
Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DM_FRAME
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 05.12.97 LE Initial

4.17.15 DL609: Initialisation denial (25.2.1.1.3), FACCH halfrate

Description: To test that the MS takes appropriate action if the network side indicates that it can not enter the multiple frame established state.

Preamble: DL000



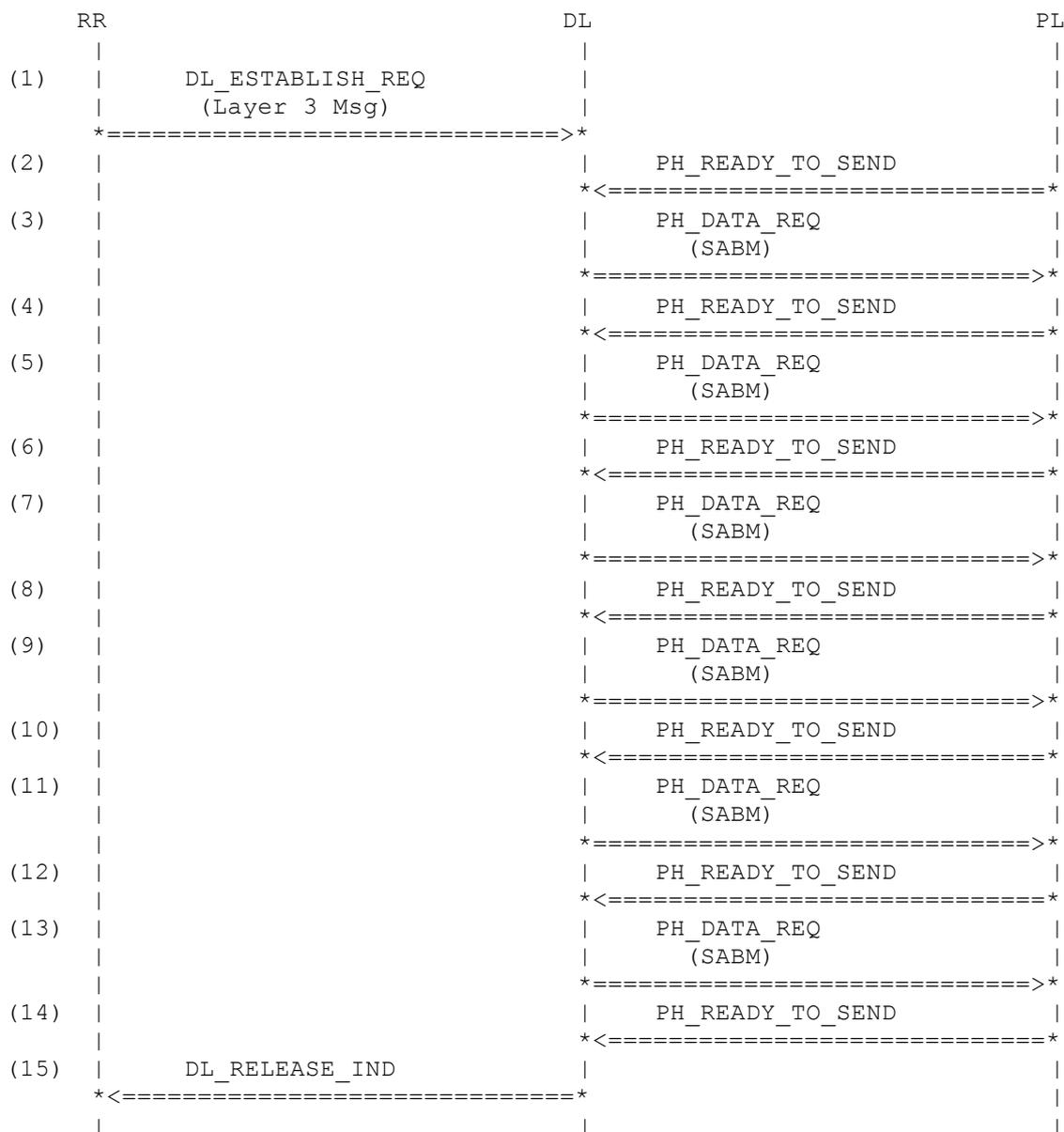
Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DM_FRAME
(5) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.17.16 DL210: Total Initialisation failure (25.2.1.1.4), SDCCH

Description: To test the MS response to the lack of the system to respond to requests to initialize the data link.
Preamble: DL000



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(10) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(11) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(12) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(13) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 05.12.97 LE Initial

4.17.17 DL211: Total Initialisation failure (25.2.1.1.4), FACCH Fullrate

Description: To test the MS response to the lack of the system to respond to requests to initialize the data link.
Preamble: DL000

RR	DL	PL
(1)	DL_ESTABLISH_REQ (Layer 3 Msg)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (SABM)	
(4)	PH_READY_TO_SEND	
(5)	PH_READY_TO_SEND	
(6)	PH_READY_TO_SEND	
(7)	PH_READY_TO_SEND	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_READY_TO_SEND	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_DATA_REQ (SABM)	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_READY_TO_SEND	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_READY_TO_SEND	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_DATA_REQ (SABM)	
(24)	PH_READY_TO_SEND	
(25)	PH_READY_TO_SEND	

```
|
|
(26) | | PH_READY_TO_SEND |
| | |
| | |
(27) | | PH_READY_TO_SEND |
| | |
| | |
(28) | | PH_READY_TO_SEND |
| | |
| | |
(29) | | PH_READY_TO_SEND |
| | |
| | |
(30) | | PH_READY_TO_SEND |
| | |
| | |
(31) | | PH_READY_TO_SEND |
| | |
| | |
(32) | | PH_READY_TO_SEND |
| | |
| | |
(33) | | PH_DATA_REQ |
| | (SABM) |
| | |
| | |
(34) | | PH_READY_TO_SEND |
| | |
| | |
(35) | | PH_READY_TO_SEND |
| | |
| | |
(36) | | PH_READY_TO_SEND |
| | |
| | |
(37) | | PH_READY_TO_SEND |
| | |
| | |
(38) | | PH_READY_TO_SEND |
| | |
| | |
(39) | | PH_READY_TO_SEND |
| | |
| | |
(40) | | PH_READY_TO_SEND |
| | |
| | |
(41) | | PH_READY_TO_SEND |
| | |
| | |
(42) | | PH_READY_TO_SEND |
| | |
| | |
(43) | | PH_DATA_REQ |
| | (SABM) |
| | |
| | |
(44) | | PH_READY_TO_SEND |
| | |
| | |
(45) | | PH_READY_TO_SEND |
| | |
| | |
(46) | | PH_READY_TO_SEND |
| | |
| | |
(47) | | PH_READY_TO_SEND |
| | |
| | |
(48) | | PH_READY_TO_SEND |
| | |
| | |
(49) | | PH_READY_TO_SEND |
| | |
| | |
(50) | | PH_READY_TO_SEND |
| | |
| | |
(51) | | PH_READY_TO_SEND |
| | |
| | |
(52) | | PH_READY_TO_SEND |
| | |
| | |
(53) | | PH_DATA_REQ |
| | (SABM) |
| | |
| | |
```

```
(54) | | PH_READY_TO_SEND |
      | | *<=====*
(55) | | PH_READY_TO_SEND |
      | | *<=====*
(56) | | PH_READY_TO_SEND |
      | | *<=====*
(57) | | PH_READY_TO_SEND |
      | | *<=====*
(58) | | PH_READY_TO_SEND |
      | | *<=====*
(59) | | PH_READY_TO_SEND |
      | | *<=====*
(60) | | PH_READY_TO_SEND |
      | | *<=====*
(61) | | PH_READY_TO_SEND |
      | | *<=====*
(62) | | PH_READY_TO_SEND |
      | | *<=====*
(63) | DL_RELEASE_IND |
      | *<=====*
      | |
```

Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(13)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(14)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(17)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH

(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_PAGING_RESP
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_PAGING_RESP
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_PAGING_RESP
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_PAGING_RESP
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(63)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT

History: 05.12.97 LE Initial

4.17.18 DL611: Total Initialisation failure (25.2.1.1.4), FACCH Halfrate

Description: To test the MS response to the lack of the system to respond to requests to initialise the data link.
Preamble: DL000

RR	DL	PL
(1)	DL_ESTABLISH_REQ (Layer 3 Msg)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (SABM)	
(4)	PH_READY_TO_SEND	
(5)	PH_READY_TO_SEND	
(6)	PH_READY_TO_SEND	
(7)	PH_READY_TO_SEND	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_READY_TO_SEND	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_DATA_REQ (SABM)	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_READY_TO_SEND	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_READY_TO_SEND	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_DATA_REQ (SABM)	
(24)	PH_READY_TO_SEND	
(25)	PH_READY_TO_SEND	

```

(26) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(27) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(28) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(29) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(30) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(31) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(32) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(33) | | PH_DATA_REQ |
| | (SABM) |
| | | | |
* =====> *
(34) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(35) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(36) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(37) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(38) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(39) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(40) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(41) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(42) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(43) | | PH_DATA_REQ |
| | (SABM) |
| | | | |
* =====> *
(44) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(45) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(46) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(47) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(48) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(49) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(50) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(51) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(52) | | PH_READY_TO_SEND |
| | | | |
* <===== *
(53) | | PH_DATA_REQ |
| | (SABM) |
| | | | |
* =====> *
    
```

```
(54) | | PH_READY_TO_SEND |
      | | *<=====*
(55) | | PH_READY_TO_SEND |
      | | *<=====*
(56) | | PH_READY_TO_SEND |
      | | *<=====*
(57) | | PH_READY_TO_SEND |
      | | *<=====*
(58) | | PH_READY_TO_SEND |
      | | *<=====*
(59) | | PH_READY_TO_SEND |
      | | *<=====*
(60) | | PH_READY_TO_SEND |
      | | *<=====*
(61) | | PH_READY_TO_SEND |
      | | *<=====*
(62) | | PH_READY_TO_SEND |
      | | *<=====*
(63) | DL_RELEASE_IND |
      | *<=====*
      | |
```

Parametrization

Primitive	Parameter	Value
(1) DL_ESTABLISH_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(13)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(14)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(15)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(16)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(17)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR

(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_PAGING_RESP
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_PAGING_RESP
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_PAGING_RESP
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_PAGING_RESP
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(63)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT

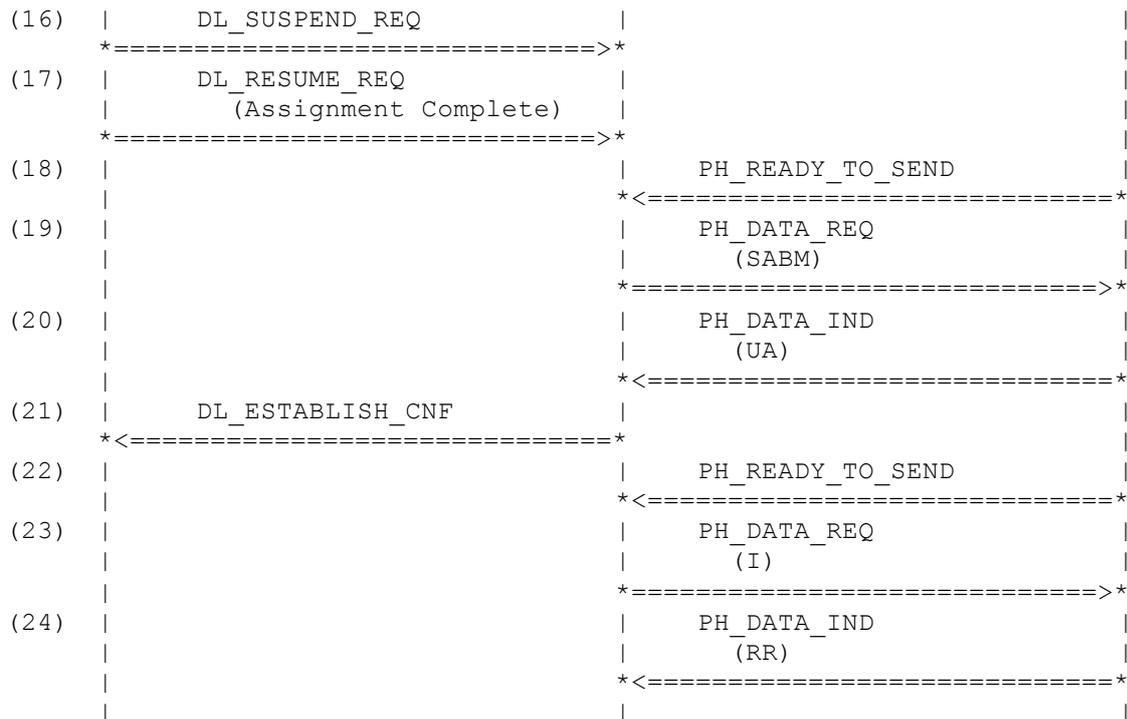
History: 08.07.98 LE Initial

4.17.19 DL212: Initialisation without Contention Resolution (25.2.1.2.1)

Description: This procedure is used after a data link has been established with contention resolution and a new data link is established on a new channel, e.g. handover, dedicated channel assignment

Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (I, Assignment Command)	
	<=====	
(2)	DL_DATA_IND (Assignment Command)	
	<=====	
(3)	DL_SUSPEND_REQ	
	=====>	
(4)	DL_RESUME_REQ (Assignment Complete)	
	=====>	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_DATA_REQ (SABM)	
	=====>	
(7)	PH_DATA_IND (UA)	
	<=====	
(8)	DL_ESTABLISH_CNF	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_DATA_REQ (I)	
	=====>	
(11)	PH_DATA_IND (RR)	
	<=====	
(12)	PH_READY_TO_SEND	
	<=====	
(13)	PH_DATA_REQ (UI)	
	=====>	
(14)	PH_DATA_IND (I, Assignment Command)	
	<=====	
(15)	DL_DATA_IND (Assignment Command)	
	<=====	



Parametrization

	Primitive	Parameter	Value
(1)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_SDCCH
(2)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_SDCCH
(3)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(4)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMP
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL

(7)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(8)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMP
(11)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_CMP
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(13)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(14)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_FACCH
(15)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_FACCH
(16)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(17)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(18)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(19)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(20)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(21)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0

(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(24)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP

History: 05.12.97 LE Initial

4.17.20 DL213: Initialisation Failure (25.2.1.2.2)

Description: To test the MS response to the loss of a layer 2 UA frame during initialisation.
Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (I, Assignment Command)	
	<=====	
(2)	DL_DATA_IND (Assignment Command)	
	<=====	
(3)	DL_SUSPEND_REQ	
	=====>	
(4)	DL_RESUME_REQ (Assignment Complete)	
	=====>	
(5)	PH_READY_TO_SEND	
	<=====	
(6)	PH_DATA_REQ (SABM)	
	=====>	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_DATA_REQ (SABM)	
	=====>	
(9)	PH_DATA_IND (UA)	
	<=====	
(10)	DL_ESTABLISH_CNF	
	<=====	
(11)	PH_READY_TO_SEND	
	<=====	
(12)	PH_DATA_REQ (I)	
	=====>	
(13)	PH_DATA_IND (RR)	
	<=====	
(14)	PH_READY_TO_SEND	
	<=====	
(15)	PH_DATA_REQ (UI)	
	=====>	
(16)	PH_DATA_IND (I, Assignment Command)	
	<=====	
(17)	DL_DATA_IND (Assignment Command)	
	<=====	

```

(18) |          DL_SUSPEND_REQ          |          |
      | *=====>*                    |          |
(19) |          DL_RESUME_REQ          |          |
      |          (Assignment Complete) |          |
      | *=====>*                    |          |
(20) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(21) |          |          PH_DATA_REQ      |          |
      |          |          (SABM)        |          |
      |          | *=====>*          |          |
(22) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(23) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(24) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(25) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(26) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(27) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(28) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(29) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(30) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(31) |          |          PH_DATA_REQ      |          |
      |          |          (SABM)        |          |
      |          | *=====>*          |          |
(32) |          |          PH_DATA_IND      |          |
      |          |          (UA)          |          |
      |          | *<=====*          |          |
(33) |          DL_ESTABLISH_CNF      |          |
      | *<=====*                    |          |
(34) |          |          PH_READY_TO_SEND |          |
      |          | *<=====*          |          |
(35) |          |          PH_DATA_REQ      |          |
      |          |          (I)          |          |
      |          | *=====>*          |          |
(36) |          |          PH_DATA_IND      |          |
      |          |          (RR)          |          |
      |          | *<=====*          |          |
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_ASSIGN_CMD_SDCCH
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	ASSIGN_CMD_SDCCH
(3) DL_SUSPEND_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0

(4)	DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH
		sapi	SAPI_0
		sdu	L3_ASSIGN_CMP
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH

(6)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(10)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMP
(13)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_CMP
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(16)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_FACCH
(17)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_FACCH

(18)	DL_SUSPEND_REQ ch_type sapi	CH_TYPE_SDCCH SAPI_0
(19)	DL_RESUME_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(21)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(31)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(32)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(33)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH SAPI_0
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED L_ASSIGN_CMP

(36)

PH_DATA_IND
ch_type
dummy
sdu

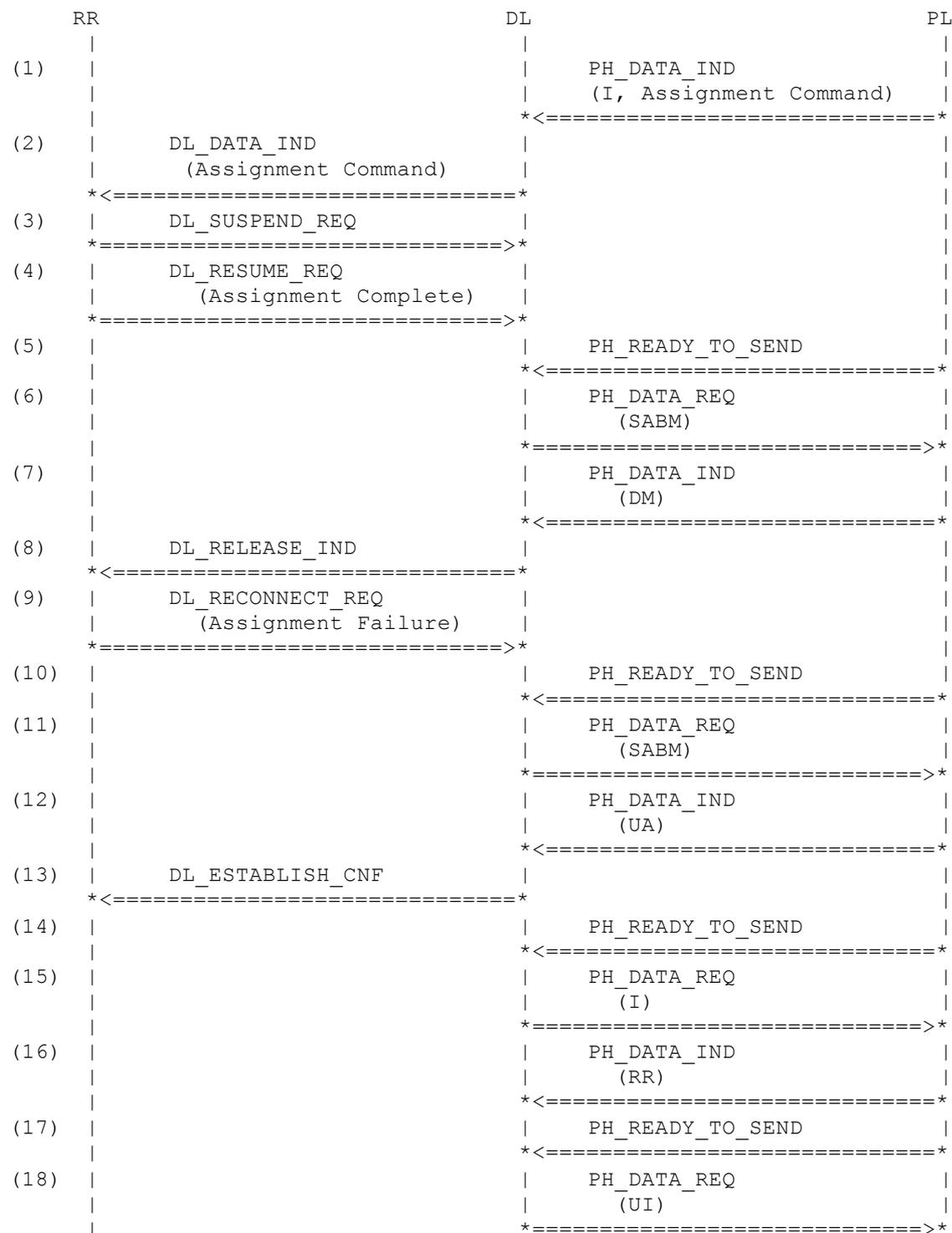
CH_TYPE_FACCH
NOT_USED
RR_ASSIGN_CMP

History: 05.12.97 LE Initial

4.17.21 DL214: Initialisation Denial (25.2.1.2.3)

Description: To test that the MS takes appropriate action if the data link can not be initialised if the network side indicates that the layer 3 process is busy.

Preamble: DL200



```

(19) | | PH_DATA_IND | |
    | | (I, Assignment Command) | |
    | | *<===== | |
(20) | | DL_DATA_IND | |
    | | (Assignment Command) | |
    | | *<===== | |
(21) | | DL_SUSPEND_REQ | |
    | | *===== | |
(22) | | DL_RESUME_REQ | |
    | | (Assignment Complete) | |
    | | *===== | |
(23) | | PH_READY_TO_SEND | |
    | | *<===== | |
(24) | | PH_DATA_REQ | |
    | | (SABM) | |
    | | *===== | |
(25) | | PH_DATA_IND | |
    | | (DM) | |
    | | *<===== | |
(26) | | DL_RELEASE_IND | |
    | | *<===== | |
(27) | | DL_RECONNECT_REQ | |
    | | (Assignment Failure) | |
    | | *===== | |
(28) | | PH_READY_TO_SEND | |
    | | *<===== | |
(29) | | PH_DATA_REQ | |
    | | (SABM) | |
    | | *===== | |
(30) | | PH_DATA_IND | |
    | | (UA) | |
    | | *<===== | |
(31) | | DL_ESTABLISH_CNF | |
    | | *<===== | |
(32) | | PH_READY_TO_SEND | |
    | | *<===== | |
(33) | | PH_DATA_REQ | |
    | | (I) | |
    | | *===== | |
(34) | | PH_DATA_IND | |
    | | (RR) | |
    | | *<===== | |
(35) | | PH_READY_TO_SEND | |
    | | *<===== | |
(36) | | PH_DATA_REQ | |
    | | (UI) | |
    | | *===== | |
    | | | |
    
```

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_SDCCH
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_SDCCH
(3) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(4) DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMP
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(7) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DM_FRAME
(8) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(9) DL_RECONNECT_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_FAILURE
(10) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH

(11) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(12) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(13) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_FAILURE
(16) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_FAILURE
(17) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(18) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(19) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_FACCH
(20) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_FACCH
(21) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0

(22)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(23)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(24)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(25)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DM_FRAME
(26)	DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT
(27)	DL_RECONNECT_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_FAILURE
(28)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(29)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(30)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(31)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(32)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(33)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED L_ASSIGN_FAILURE

(34) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_FAILURE
(35) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(36) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.22 DL215: Total Initialisation Failure (25.2.1.2.4)

Description: To test the MS response to the lack of the system to respond to requests to initialise the data link.
Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (I, Assignment Command)	
(2)	DL_DATA_IND (Assignment Command)	
(3)	DL_SUSPEND_REQ	
(4)	DL_RESUME_REQ (Assignment Complete)	
(5)	PH_READY_TO_SEND	
(6)	PH_DATA_REQ (SABM)	
(7)	PH_READY_TO_SEND	
(8)	PH_DATA_REQ (SABM)	
(9)	PH_READY_TO_SEND	
(10)	PH_DATA_REQ (SABM)	
(11)	PH_READY_TO_SEND	
(12)	PH_DATA_REQ (SABM)	
(13)	PH_READY_TO_SEND	
(14)	PH_DATA_REQ (SABM)	
(15)	PH_READY_TO_SEND	
(16)	PH_DATA_REQ (SABM)	
(17)	PH_READY_TO_SEND	
(18)	DL_RELEASE_IND	
(19)	PH_DATA_REQ (Empty Frame)	
(20)	DL_RECONNECT_REQ (Assignment Failure)	
(21)	PH_READY_TO_SEND	

```

(22) | | PH_DATA_REQ | |
      | | (SABM) | |
      | | *=====| |
(23) | | PH_DATA_IND | |
      | | (UA) | |
      | | *<=====| |
(24) | DL_ESTABLISH_CNF | |
      | *<=====| |
(25) | | PH_READY_TO_SEND | |
      | | *<=====| |
(26) | | PH_DATA_REQ | |
      | | (I) | |
      | | *=====| |
(27) | | PH_DATA_IND | |
      | | (RR) | |
      | | *<=====| |
(28) | | PH_READY_TO_SEND | |
      | | *<=====| |
(29) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====| |
(30) | | PH_DATA_IND | |
      | | (I, Assignment Command) | |
      | | *<=====| |
(31) | DL_DATA_IND | |
      | | (Assignment Command) | |
      | | *<=====| |
(32) | DL_SUSPEND_REQ | |
      | | *=====| |
(33) | DL_RESUME_REQ | |
      | | (Assignment Complete) | |
      | | *=====| |
(34) | | PH_READY_TO_SEND | |
      | | *<=====| |
(35) | | PH_DATA_REQ | |
      | | (SABM) | |
      | | *=====| |
(36) | | PH_READY_TO_SEND | |
      | | *<=====| |
(37) | | PH_READY_TO_SEND | |
      | | *<=====| |
(38) | | PH_READY_TO_SEND | |
      | | *<=====| |
(39) | | PH_READY_TO_SEND | |
      | | *<=====| |
(40) | | PH_READY_TO_SEND | |
      | | *<=====| |
(41) | | PH_READY_TO_SEND | |
      | | *<=====| |
(42) | | PH_READY_TO_SEND | |
      | | *<=====| |
(43) | | PH_READY_TO_SEND | |
      | | *<=====| |
(44) | | PH_READY_TO_SEND | |
      | | *<=====| |
(45) | | PH_DATA_REQ | |
      | | (SABM) | |
      | | *=====| |
(46) | | PH_READY_TO_SEND | |
      | | *<=====| |
    
```

```
(47) | | PH_READY_TO_SEND |
      | | *<=====*
(48) | | PH_READY_TO_SEND |
      | | *<=====*
(49) | | PH_READY_TO_SEND |
      | | *<=====*
(50) | | PH_READY_TO_SEND |
      | | *<=====*
(51) | | PH_READY_TO_SEND |
      | | *<=====*
(52) | | PH_READY_TO_SEND |
      | | *<=====*
(53) | | PH_READY_TO_SEND |
      | | *<=====*
(54) | | PH_READY_TO_SEND |
      | | *<=====*
(55) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>*
(56) | | PH_READY_TO_SEND |
      | | *<=====*
(57) | | PH_READY_TO_SEND |
      | | *<=====*
(58) | | PH_READY_TO_SEND |
      | | *<=====*
(59) | | PH_READY_TO_SEND |
      | | *<=====*
(60) | | PH_READY_TO_SEND |
      | | *<=====*
(61) | | PH_READY_TO_SEND |
      | | *<=====*
(62) | | PH_READY_TO_SEND |
      | | *<=====*
(63) | | PH_READY_TO_SEND |
      | | *<=====*
(64) | | PH_READY_TO_SEND |
      | | *<=====*
(65) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>*
(66) | | PH_READY_TO_SEND |
      | | *<=====*
(67) | | PH_READY_TO_SEND |
      | | *<=====*
(68) | | PH_READY_TO_SEND |
      | | *<=====*
(69) | | PH_READY_TO_SEND |
      | | *<=====*
(70) | | PH_READY_TO_SEND |
      | | *<=====*
(71) | | PH_READY_TO_SEND |
      | | *<=====*
(72) | | PH_READY_TO_SEND |
      | | *<=====*
(73) | | PH_READY_TO_SEND |
      | | *<=====*
(74) | | PH_READY_TO_SEND |
      | | *<=====*
(75) | | PH_DATA_REQ |
      | | (SABM) |
```

```

(76) | | PH_READY_TO_SEND |
| | | |
*=====*>
(77) | | PH_READY_TO_SEND |
| | | |
*<=====*
```

(78) | | PH_READY_TO_SEND |

(79) | | PH_READY_TO_SEND |

(80) | | PH_READY_TO_SEND |

(81) | | PH_READY_TO_SEND |

(82) | | PH_READY_TO_SEND |

(83) | | PH_READY_TO_SEND |

(84) | | PH_READY_TO_SEND |

(85) | | PH_DATA_REQ |

| | (SABM) |

=====>

(86) | | PH_READY_TO_SEND |

(87) | | PH_READY_TO_SEND |

(88) | | PH_READY_TO_SEND |

(89) | | PH_READY_TO_SEND |

(90) | | PH_READY_TO_SEND |

(91) | | PH_READY_TO_SEND |

(92) | | PH_READY_TO_SEND |

(93) | | PH_READY_TO_SEND |

(94) | | PH_READY_TO_SEND |

(95) | | DL_RELEASE_IND |

<=====

(96) | | DL_RECONNECT_REQ |

| | (Assignment Failure) |

=====>

(97) | | PH_READY_TO_SEND |

(98) | | PH_DATA_REQ |

| | (SABM) |

=====>

(99) | | PH_DATA_IND |

| | (UA) |

<=====

(100) | | DL_ESTABLISH_CNF |

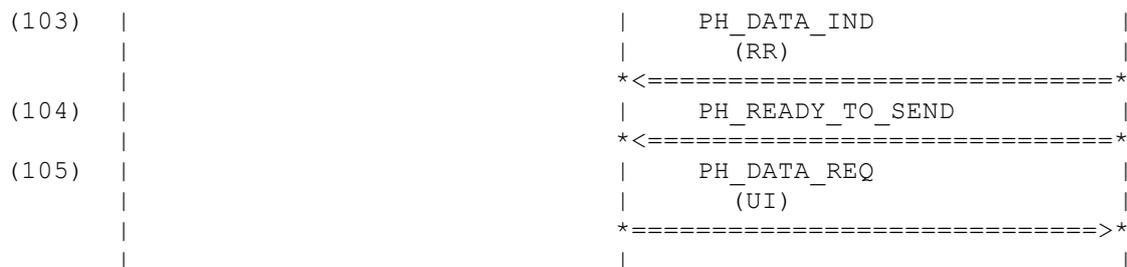
<=====

(101) | | PH_READY_TO_SEND |

(102) | | PH_DATA_REQ |

| | (I) |

=====>



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_ASSIGN_CMD_SDCCH
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	ASSIGN_CMD_SDCCH
(3) DL_SUSPEND_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(4) DL_RESUME_REQ	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_ASSIGN_CMP
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10)	PH_DATA_REQ	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL
(11)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(12)	PH_DATA_REQ	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_A_UL
(13)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH

(14)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(17)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(18)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT

(19)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH
(20)	DL_RECONNECT_REQ ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_FAILURE
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(22)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(23)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(24)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_SDCCH SAPI_0
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(26)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_FAILURE
(27)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_FAILURE
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(29)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(30)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMD_FACCH
(31)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ASSIGN_CMD_FACCH
(32)	DL_SUSPEND_REQ ch_type sapi	CH_TYPE_SDCCH SAPI_0
(33)	DL_RESUME_REQ ch_type	CH_TYPE_FACCH

	sapi	SAPI_0
	sdu	L3_ASSIGN_CMP
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(63)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(64)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(65)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(67)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(68)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(69)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(70)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(71)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(72)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(73)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(74)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(75)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(76)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(77)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(78)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(79)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(80)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(81)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(82)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(83)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(84)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(85)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(86)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(87)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(88)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(89)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(90)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(91)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(92)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(93)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(94)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(95)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT

(96)

DL_RECONNECT_REQ	
ch_type	CH_TYPE_SDCCH
sapi	SAPI_0
sdu	ASSIGN_FAILURE

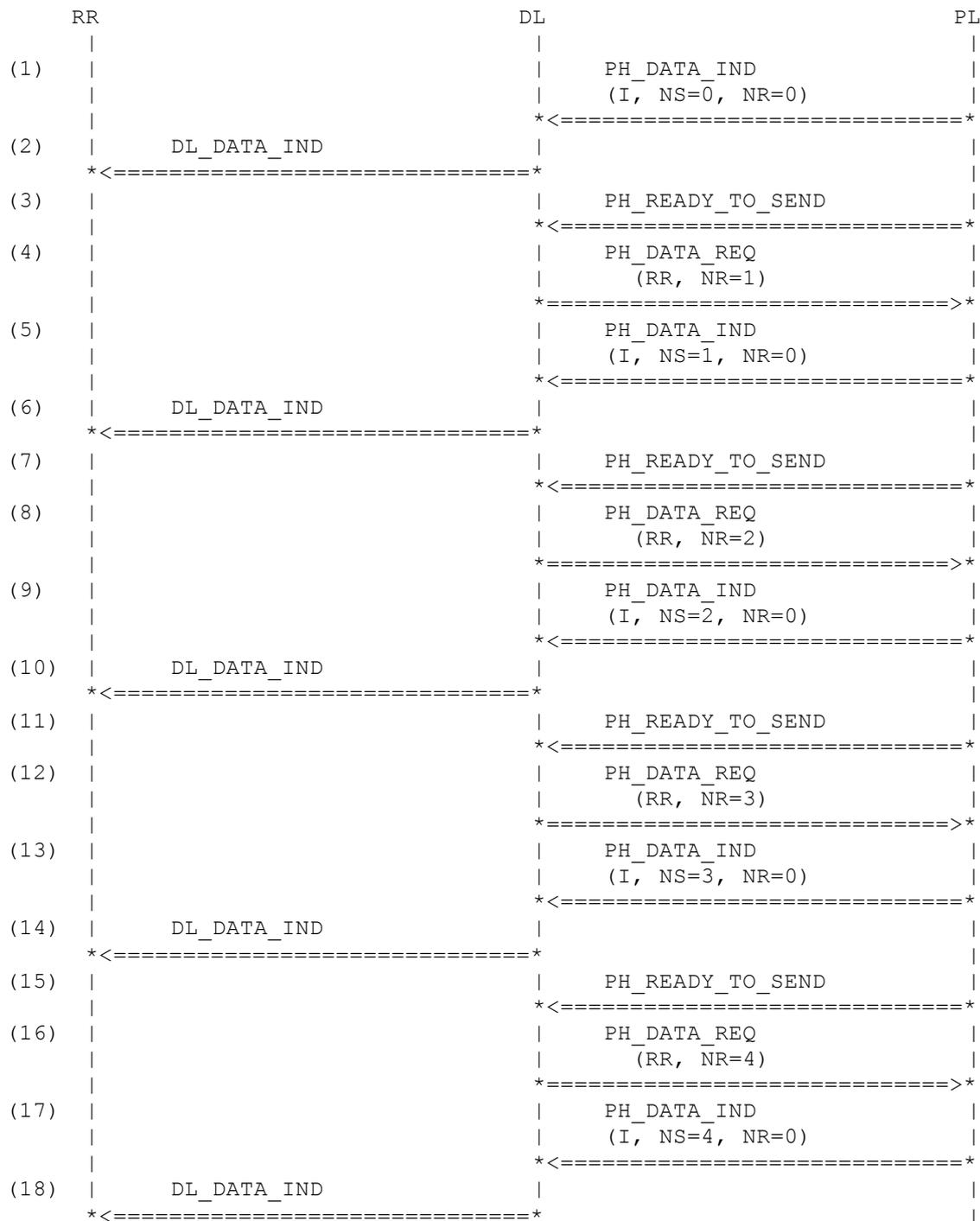
(97)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(98)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(99)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(100)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_SDCCH SAPI_0
(101)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(102)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_FAILURE
(103)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_FAILURE
(104)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(105)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.23 DL216: Normal Information Transfer (25.2.2.1), SDCCH

Description: To test the operation of layer 2 sequence numbering. Since there are 8 sequence numbers the test cycles through 9 information transfers.

Preamble: DL200



```

(19) | DL_DATA_REQ | |
      | *=====>* | |
(20) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(21) | | PH_DATA_REQ | |
      | | (I, NS=0, NR=5) | |
      | | *=====>* | |
(22) | | PH_DATA_IND | |
      | | (I, NS=5, NR=1) | |
      | | *<=====* | |
(23) | DL_DATA_IND | |
      | *<=====* | |
(24) | DL_DATA_REQ | |
      | *=====>* | |
(25) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(26) | | PH_DATA_REQ | |
      | | (I, NS=1, NR=6) | |
      | | *=====>* | |
(27) | | PH_DATA_IND | |
      | | (I, NS=6, NR=2) | |
      | | *<=====* | |
(28) | DL_DATA_IND | |
      | *<=====* | |
(29) | DL_DATA_REQ | |
      | *=====>* | |
(30) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(31) | | PH_DATA_REQ | |
      | | (I, NS=2, NR=7) | |
      | | *=====>* | |
(32) | | PH_DATA_IND | |
      | | (I, NS=7, NR=3) | |
      | | *<=====* | |
(33) | DL_DATA_IND | |
      | *<=====* | |
(34) | DL_DATA_REQ | |
      | *=====>* | |
(35) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(36) | | PH_DATA_REQ | |
      | | (I, NS=3, NR=0) | |
      | | *=====>* | |
(37) | | PH_DATA_IND | |
      | | (I, NS=0, NR=4) | |
      | | *<=====* | |
(38) | DL_DATA_IND | |
      | *<=====* | |
(39) | DL_DATA_REQ | |
      | *=====>* | |
(40) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(41) | | PH_DATA_REQ | |
      | | (I, NS=4, NR=1) | |
      | | *=====>* | |
(42) | | PH_DATA_IND | |
      | | (RR, NR=5) | |
      | | *<=====* | |
    
```

```

(43) | DL_DATA_REQ |
      | *=====>* |
(44) | | PH_READY_TO_SEND |
      | | *<=====* |
(45) | | PH_DATA_REQ |
      | | (I, NS=5, NR=1) |
      | | *=====>* |
(46) | | PH_DATA_IND |
      | | (RR, NR=6) |
      | | *<=====* |
(47) | DL_DATA_REQ |
      | *=====>* |
(48) | | PH_READY_TO_SEND |
      | | *<=====* |
(49) | | PH_DATA_REQ |
      | | (I, NS=6, NR=1) |
      | | *=====>* |
(50) | | PH_DATA_IND |
      | | (RR, NR=7) |
      | | *<=====* |
(51) | DL_DATA_REQ |
      | *=====>* |
(52) | | PH_READY_TO_SEND |
      | | *<=====* |
(53) | | PH_DATA_REQ |
      | | (I, NS=7, NR=1) |
      | | *=====>* |
(54) | | PH_DATA_IND |
      | | (RR, NR=0) |
      | | *<=====* |
(55) | DL_DATA_REQ |
      | *=====>* |
(56) | | PH_READY_TO_SEND |
      | | *<=====* |
(57) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1) |
      | | *=====>* |
(58) | | PH_DATA_IND |
      | | (RR, NR=1) |
      | | *<=====* |
      | | |
    
```

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_1_0
(6) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(7) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(8) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_2
(9) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_2_0
(10) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_3
(13) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_3_0
(14) DL_DATA_IND	ch_type	CH_TYPE_SDCCH

	sapi sdu	SAPI_0 ID_REQUEST
(15) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(16) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_4
(17) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_4_0
(18) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(19) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(20) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(21) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_5
(22) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_5_1
(23) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(24) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(25) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(26) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_1_6
(27) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_6_2
(28) DL_DATA_IND	ch_type	CH_TYPE_SDCCH

	sapi sdu	SAPI_0 ID_REQUEST
(29) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(30) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(31) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_1_2_7
(32) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_1_7_3
(33) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(34) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(35) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(36) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_1_3_0
(37) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_1_0_4
(38) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(39) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(40) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(41) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_1_4_1
(42) PH_DATA_IND	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED BS_RR_5
(4 3) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(4 4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4 5) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_5_1
(4 6) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_6
(4 7) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(4 8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4 9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_6_1
(5 0) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_7
(5 1) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(5 2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5 3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_7_1
(5 4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0
(5 5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(5 6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH

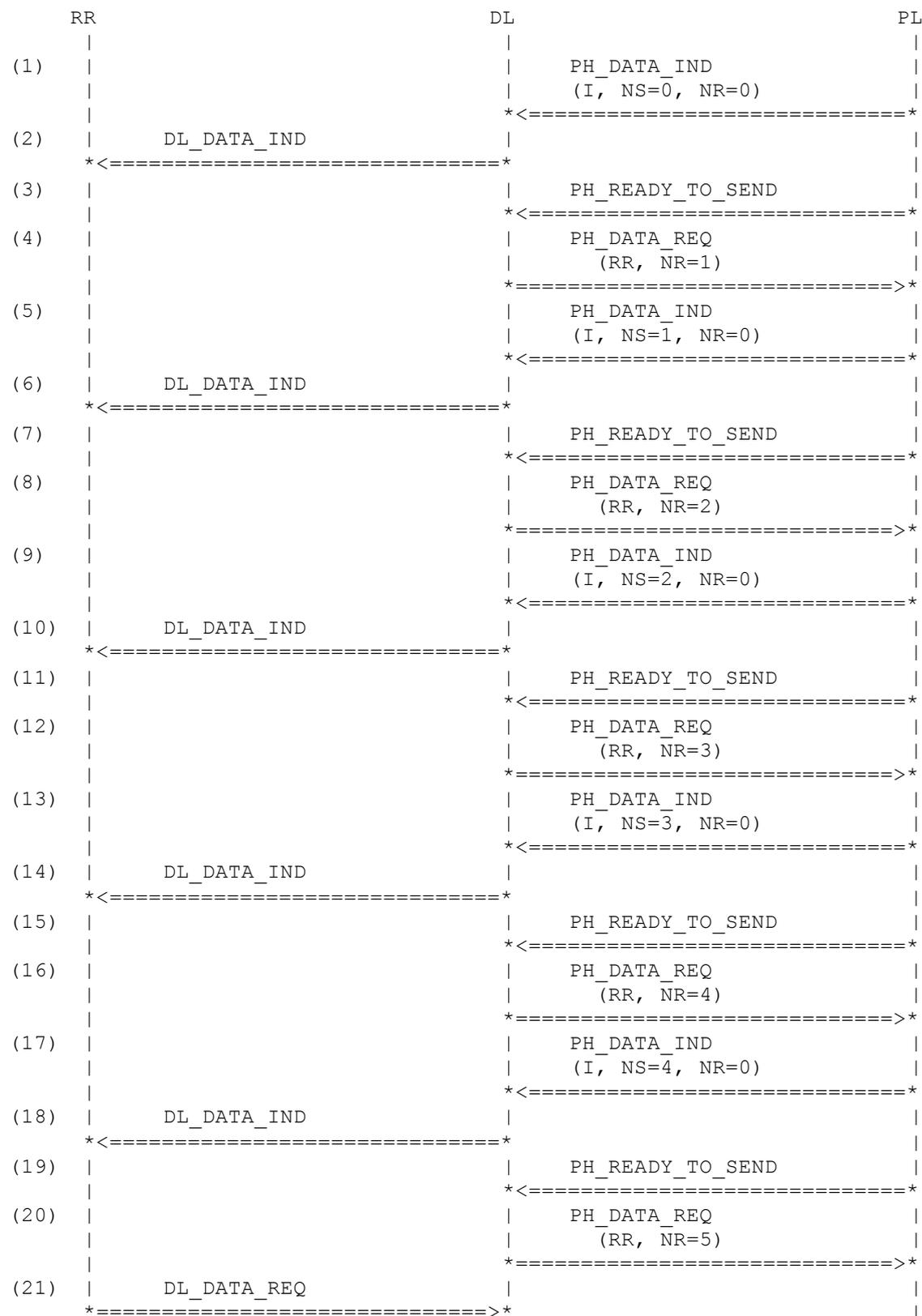
(57) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1
(58) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_1

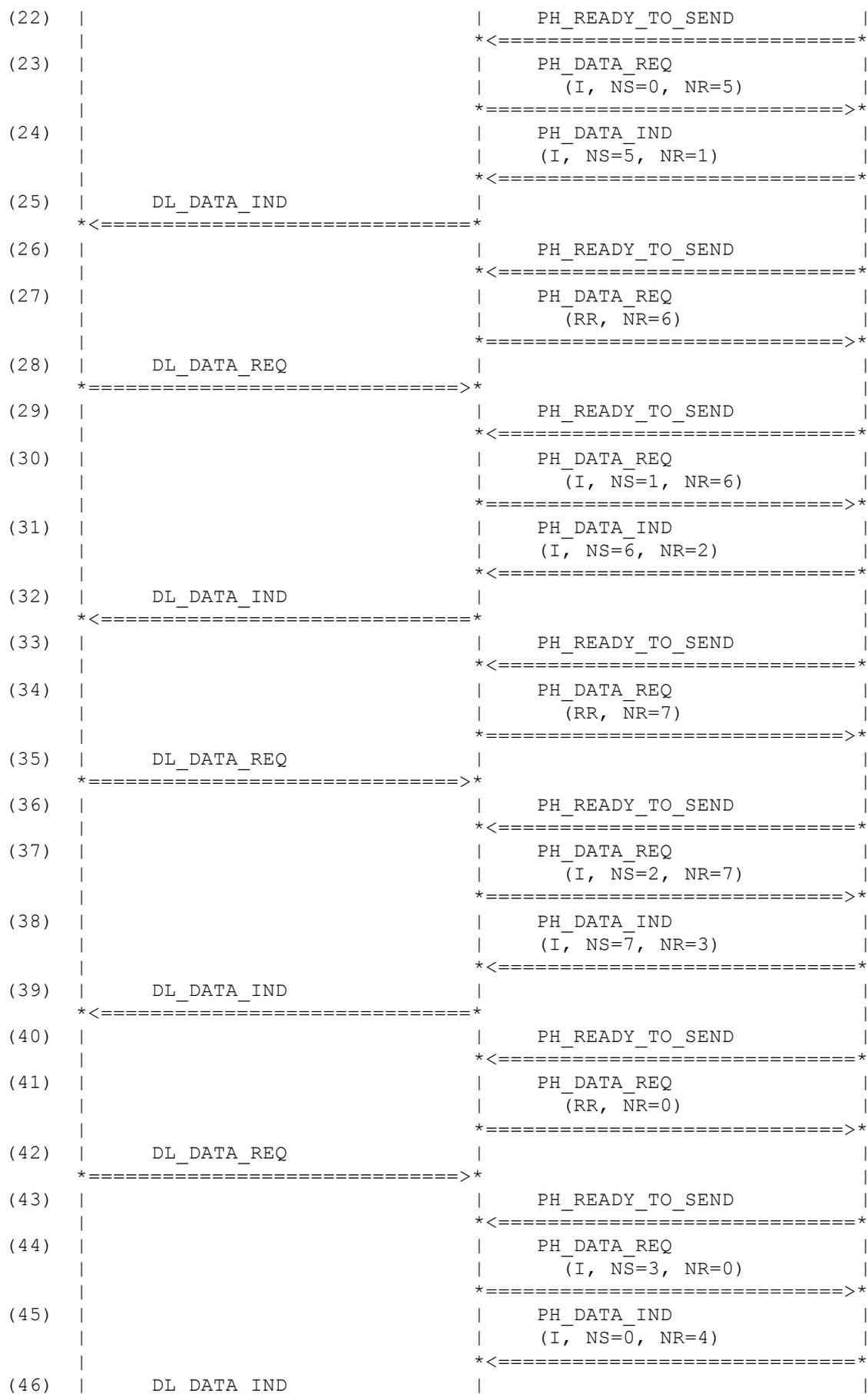
History: 05.12.97 LE Initial

4.17.24 DL217: Normal Information Transfer (25.2.2.1), FACCH Fullrate

Description: To test the operation of layer 2 sequence numbering. Since there are 8 sequence numbers the test cycles through 9 information transfers.

Preamble: DL201





```

(47) | *<===== |
      | | PH_READY_TO_SEND |
      | | *<===== |
(48) | | PH_DATA_REQ |
      | | (RR, NR=1) |
      | | *<===== |
(49) | DL_DATA_REQ |
      | *===== |
(50) | | PH_READY_TO_SEND |
      | | *<===== |
(51) | | PH_DATA_REQ |
      | | (I, NS=4, NR=1) |
      | | *<===== |
(52) | | PH_DATA_IND |
      | | (RR, NR=5) |
      | | *<===== |
(53) | DL_DATA_REQ |
      | *===== |
(54) | | PH_READY_TO_SEND |
      | | *<===== |
(55) | | PH_DATA_REQ |
      | | (I, NS=5, NR=1) |
      | | *<===== |
(56) | | PH_DATA_IND |
      | | (RR, NR=6) |
      | | *<===== |
(57) | DL_DATA_REQ |
      | *===== |
(58) | | PH_READY_TO_SEND |
      | | *<===== |
(59) | | PH_DATA_REQ |
      | | (I, NS=6, NR=1) |
      | | *<===== |
(60) | | PH_DATA_IND |
      | | (RR, NR=7) |
      | | *<===== |
(61) | DL_DATA_REQ |
      | *===== |
(62) | | PH_READY_TO_SEND |
      | | *<===== |
(63) | | PH_DATA_REQ |
      | | (I, NS=7, NR=1) |
      | | *<===== |
(64) | | PH_DATA_IND |
      | | (RR, NR=0) |
      | | *<===== |
(65) | DL_DATA_REQ |
      | *===== |
(66) | | PH_READY_TO_SEND |
      | | *<===== |
(67) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1) |
      | | *<===== |
(68) | | PH_DATA_IND |
      | | (RR, NR=1) |
      | | *<===== |
      | |
    
```

Parametrization

Primitive	Parameter	Value
-----------	-----------	-------

(1)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_0
(2)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_1
(5)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_1_0
(6)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_2
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_2_0
(10)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_3
(13)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_3_0
(14)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST

(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_4
(17)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_4_0
(18)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_5
(21)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_5
(24)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_5_1
(25)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_6
(28)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_1_6
(31)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_6_2
(32)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_7
(35)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_2_7
(38)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_7_3
(39)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(41)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0
(42)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_DATA_REQ ch_type	CH_TYPE_FACCH

	dummy	NOT_USED
	sdu	MS_I_3_0
(45)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_I_0_4
(46)	DL_DATA_IND	
	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(47)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(48)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_RR_1
(49)	DL_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_RESPONSE
(50)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(51)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_I_4_1
(52)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_RR_5
(53)	DL_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_RESPONSE
(54)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH
(55)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_I_5_1
(56)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_RR_6
(57)	DL_DATA_REQ	
	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_RESPONSE
(58)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH

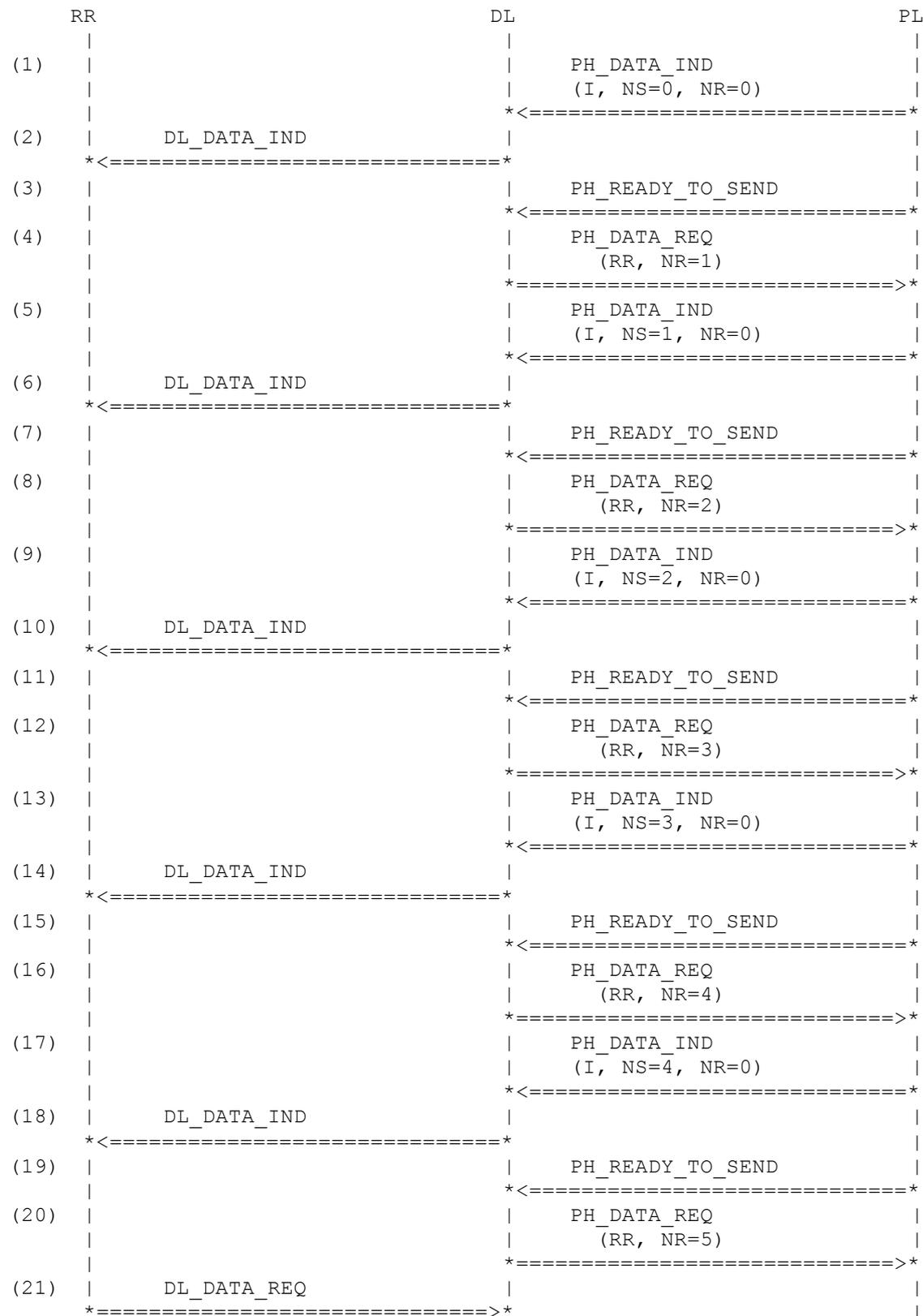
(59)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_6_1
(60)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_7
(61)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(63)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_7_1
(64)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0
(65)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(67)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1
(68)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_1

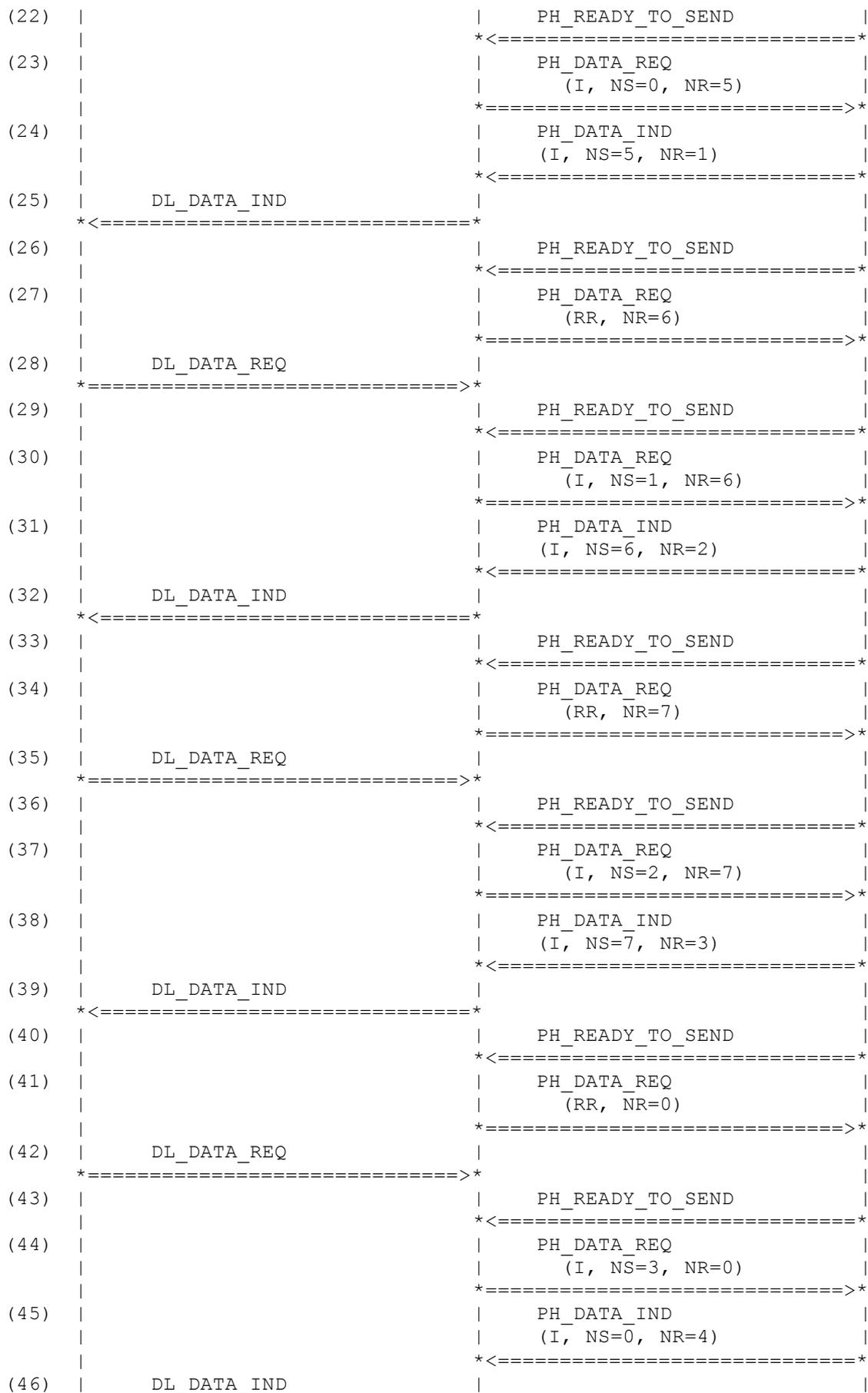
History: 05.12.97 LE Initial

4.17.25 DL617: Normal Information Transfer (25.2.2.1), FACCH Halfrate

Description: To test the operation of layer 2 sequence numbering. Since there are 8 sequence numbers the test cycles through 9 information transfers.

Preamble: DL601





```

* <=====
(47) | | PH_READY_TO_SEND |
| | * <=====
(48) | | PH_DATA_REQ |
| | (RR, NR=1) |
| | * =====> *
(49) | DL_DATA_REQ |
| | * =====> *
(50) | | PH_READY_TO_SEND |
| | * <=====
(51) | | PH_DATA_REQ |
| | (I, NS=4, NR=1) |
| | * =====> *
(52) | | PH_DATA_IND |
| | (RR, NR=5) |
| | * <=====
(53) | DL_DATA_REQ |
| | * =====> *
(54) | | PH_READY_TO_SEND |
| | * <=====
(55) | | PH_DATA_REQ |
| | (I, NS=5, NR=1) |
| | * =====> *
(56) | | PH_DATA_IND |
| | (RR, NR=6) |
| | * <=====
(57) | DL_DATA_REQ |
| | * =====> *
(58) | | PH_READY_TO_SEND |
| | * <=====
(59) | | PH_DATA_REQ |
| | (I, NS=6, NR=1) |
| | * =====> *
(60) | | PH_DATA_IND |
| | (RR, NR=7) |
| | * <=====
(61) | DL_DATA_REQ |
| | * =====> *
(62) | | PH_READY_TO_SEND |
| | * <=====
(63) | | PH_DATA_REQ |
| | (I, NS=7, NR=1) |
| | * =====> *
(64) | | PH_DATA_IND |
| | (RR, NR=0) |
| | * <=====
(65) | DL_DATA_REQ |
| | * =====> *
(66) | | PH_READY_TO_SEND |
| | * <=====
(67) | | PH_DATA_REQ |
| | (I, NS=0, NR=1) |
| | * =====> *
(68) | | PH_DATA_IND |
| | (RR, NR=1) |
| | * <=====
| |

```

Parametrization

Primitive	Parameter	Value
-----------	-----------	-------

(1)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0
(2)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1
(5)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_1_0
(6)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_2
(9)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_2_0
(10)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(12)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_3
(13)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_3_0
(14)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST

(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_4
(17)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_4_0
(18)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_5
(21)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_5
(24)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_5_1
(25)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_6
(28)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_1_6
(31)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_6_2
(32)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_7
(35)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_2_7
(38)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_7_3
(39)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(41)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0
(42)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_DATA_REQ ch_type	CH_TYPE_FACCH_HR

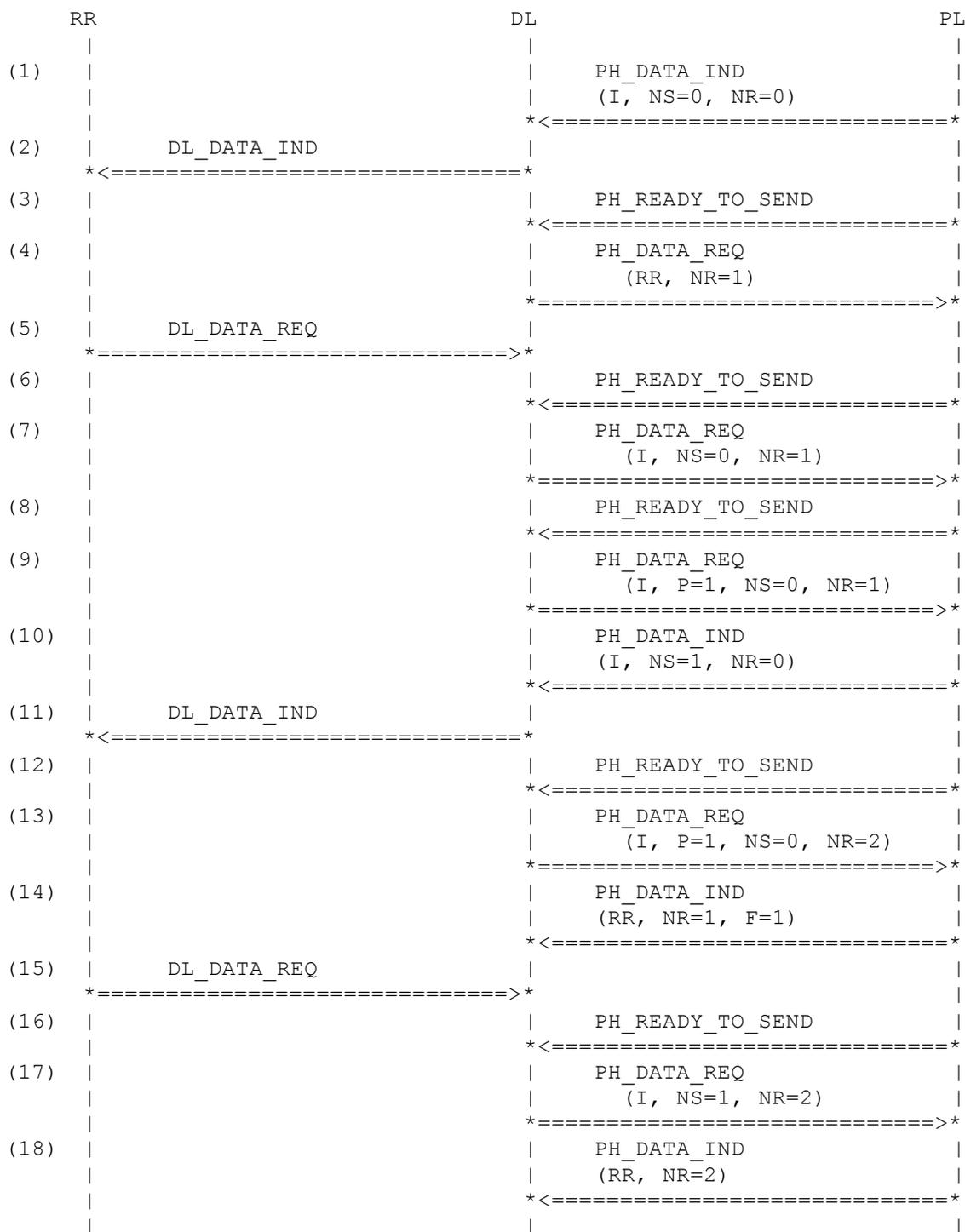
	dummy sdu	NOT_USED MS_I_3_0
(45)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_4
(46)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1
(49)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(51)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_4_1
(52)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_5
(53)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_5_1
(56)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_6
(57)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(59)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_6_1
(60)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_7
(61)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(63)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_7_1
(64)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0
(65)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(67)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1
(68)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_1

History: 08.07.98 LE Initial

4.17.26 DL218: Receipt of I frame in Timer Recovery (25.2.2.2), SDCCH

Description: To test that the MS is able to respond to I frames whilst in the timer recovery state.
Preamble: DL200



Parametrization

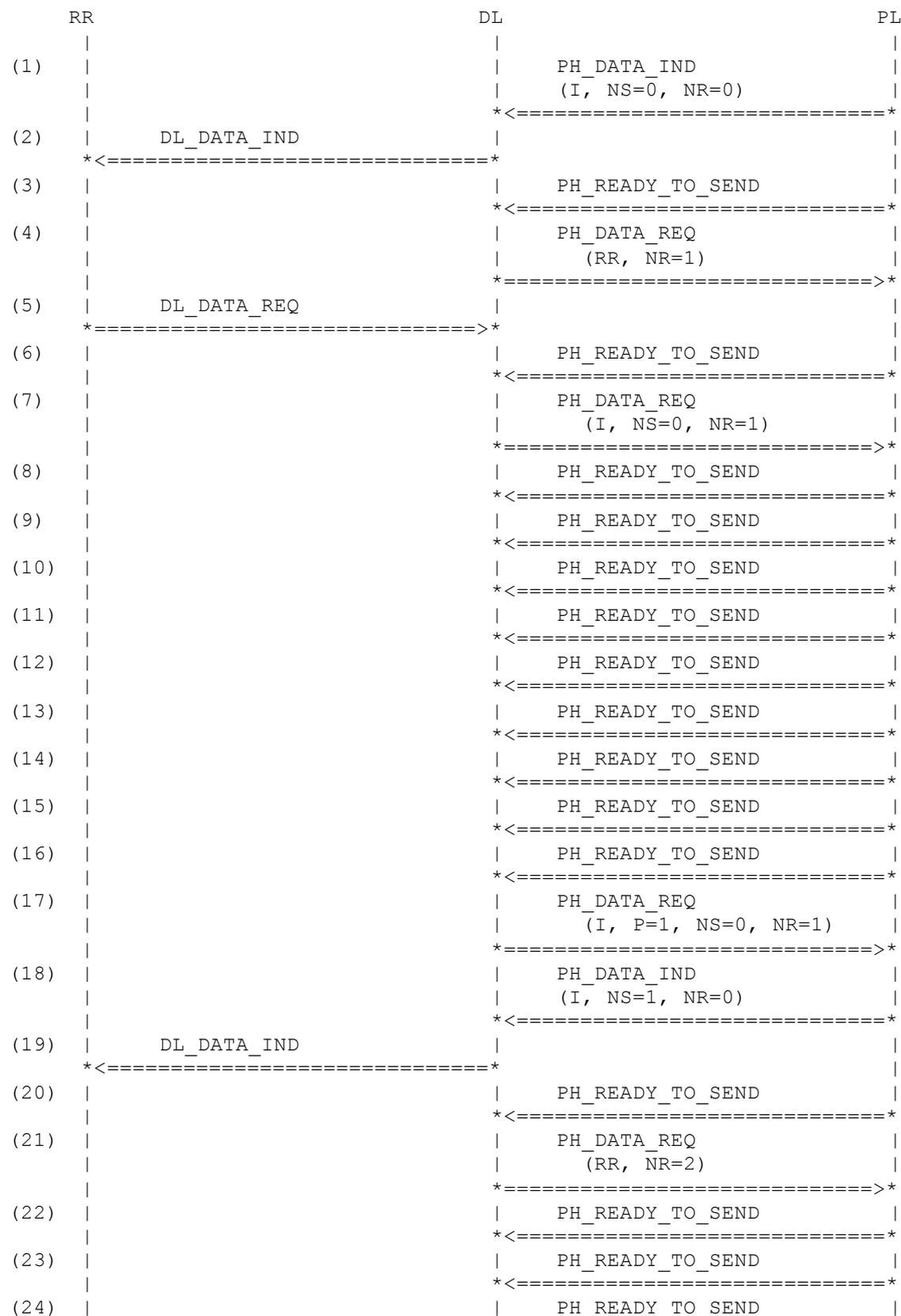
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(10) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_1_0
(11) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(12) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(13) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_2_P
(14) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_1_F

(15) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(16) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(17) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_1_2
(18) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_2_F

History: 05.12.97 LE Initial

4.17.27 DL219: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Fullrate

Description: To test that the MS is able to respond to I frames whilst in the timer recovery state.
Preamble: DL201



```

(25) | | | *<=====
      | | | | PH_READY_TO_SEND |
      | | | *<=====
(26) | | | | PH_READY_TO_SEND |
      | | | *<=====
(27) | | | | PH_READY_TO_SEND |
      | | | *<=====
(28) | | | | PH_READY_TO_SEND |
      | | | *<=====
(29) | | | | PH_READY_TO_SEND |
      | | | *<=====
(30) | | | | PH_DATA_REQ |
      | | | | (I, P=1, NS=0, NR=2) |
      | | | *=====>*
(31) | | | | PH_DATA_IND |
      | | | | (RR, NR=1, F=1) |
      | | | *<=====
(32) | | DL_DATA_REQ |
      | *=====>*
(33) | | | | PH_READY_TO_SEND |
      | | | *<=====
(34) | | | | PH_DATA_REQ |
      | | | | (I, NS=1, NR=2) |
      | | | *=====>*
(35) | | | | PH_DATA_IND |
      | | | | (RR, NR=2) |
      | | | *<=====
      | | | |
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_RR_1
(5) DL_DATA_REQ	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_I_0_1

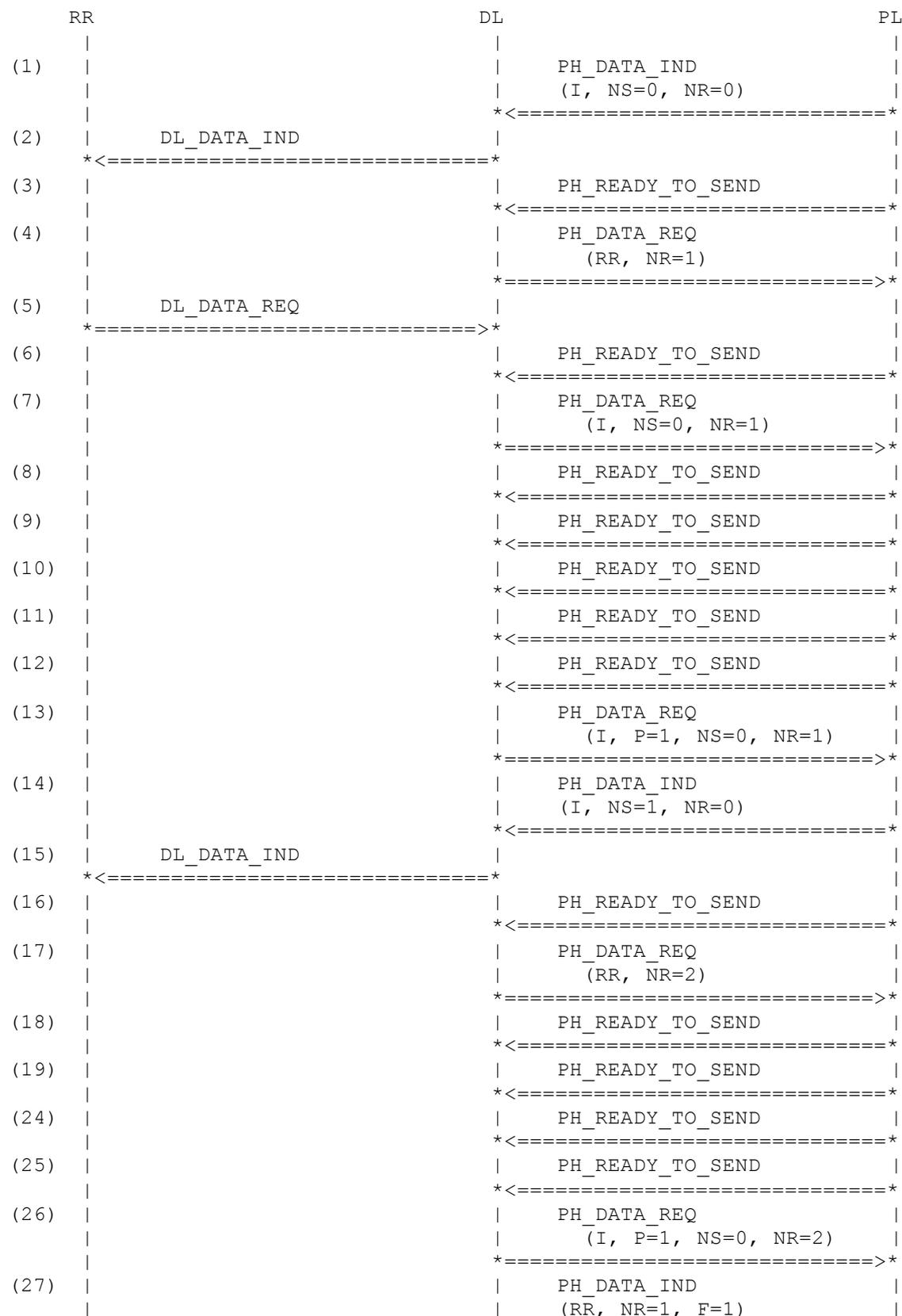
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(17)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(18)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_1_0
(19)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(20)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(21)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_2
(22)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_2_P
(31)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_1_F
(32)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_1_2
(35)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_2

History: 05.12.97 LE Initial

4.17.28 DL619: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Halfrate

Description: To test that the MS is able to respond to I frames whilst in the timer recovery state.
Preamble: DL601



```
|
|
(28) | DL_DATA_REQ |
| *=====> * |
(29) | | PH_READY_TO_SEND |
| | *===== * |
(30) | | PH_DATA_REQ |
| | (I, NS=1, NR=2) |
| | *===== * |
(31) | | PH_DATA_IND |
| | (RR, NR=2) |
| | *===== * |
| | |
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(13)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(14)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_1_0
(15)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(17)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_2
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_2_P
(23)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_1_F
(24)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_1_2
(27)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_2

History: 08.07.98 LE Initial

4.17.29 DL620: Receipt of I frame in Timer Recovery (25.2.2.2), FACCH Halfrate (II)

Description: To test that the MS is able to respond to I frames whilst in the timer recovery state.

Preamble: DL601

RR	DL	PL
(1)	PH_DATA_IND (I, NS=0, NR=0)	
	<=====	
(2)	DL_DATA_IND	
	<=====	
(3)	DL_DATA_REQ	
	<=====	
(4)	PH_READY_TO_SEND	
	<=====	
(5)	PH_DATA_REQ (I, NS=0, NR=1)	
	<=====	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_DATA_IND (I, NS=0, NR=0, P=1)	
	<=====	
(10)	PH_READY_TO_SEND	
	<=====	
(11)	PH_DATA_REQ (REJ, NR=1)	
	<=====	
(12)	PH_READY_TO_SEND	
	<=====	

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0	
(2) DL_DATA_IND ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST	
(3) DL_DATA_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE	
(4) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	
(5) PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1	
(6) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	
(7) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	
(8) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	
(9) PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0_P	
(10) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	
(11) PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_REJ_1_F	
(12) PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR	

History: 08.07.98 LE Initial

4.17.30 DL220: Segmentation and Concatenation (25.2.2.3), SDCCH

Description: To test the proper use of segmentation and concatenation, suspend and resume.
Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (I, NS=0, NR=0)	
(2)	DL_DATA_IND	
(3)	PH_READY_TO_SEND	
(4)	PH_DATA_REQ (RR, NR=1)	
(5)	DL_DATA_REQ	
(6)	PH_READY_TO_SEND	
(7)	PH_DATA_REQ (I, NS=0, NR=1, M=1)	
(8)	PH_DATA_IND (RR, NR=1)	
(9)	PH_READY_TO_SEND	
(10)	PH_DATA_REQ (I, NS=1, NR=1)	
(11)	PH_DATA_IND (I, NS=1, NR=1, M=1)	
(12)	PH_READY_TO_SEND	
(13)	PH_DATA_REQ (I, P=1, NS=1, NR=2)	
(14)	PH_DATA_IND (I, NS=2, NR=1)	
(15)	DL_DATA_IND	
(16)	DL_SUSPEND_REQ	
(17)	DL_RESUME_REQ	
(18)	PH_READY_TO_SEND	
(19)	PH_DATA_REQ (SABM)	
(20)	PH_DATA_IND (UA)	
(21)	DL_ESTABLISH_CNF	
(22)	PH_READY_TO_SEND	

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 LONG_MESSAGE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_M1
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_1
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_1_1_M2
(11) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_1_1_M1
(12) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(13) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_1_2_M2
(14) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_2_1_M2

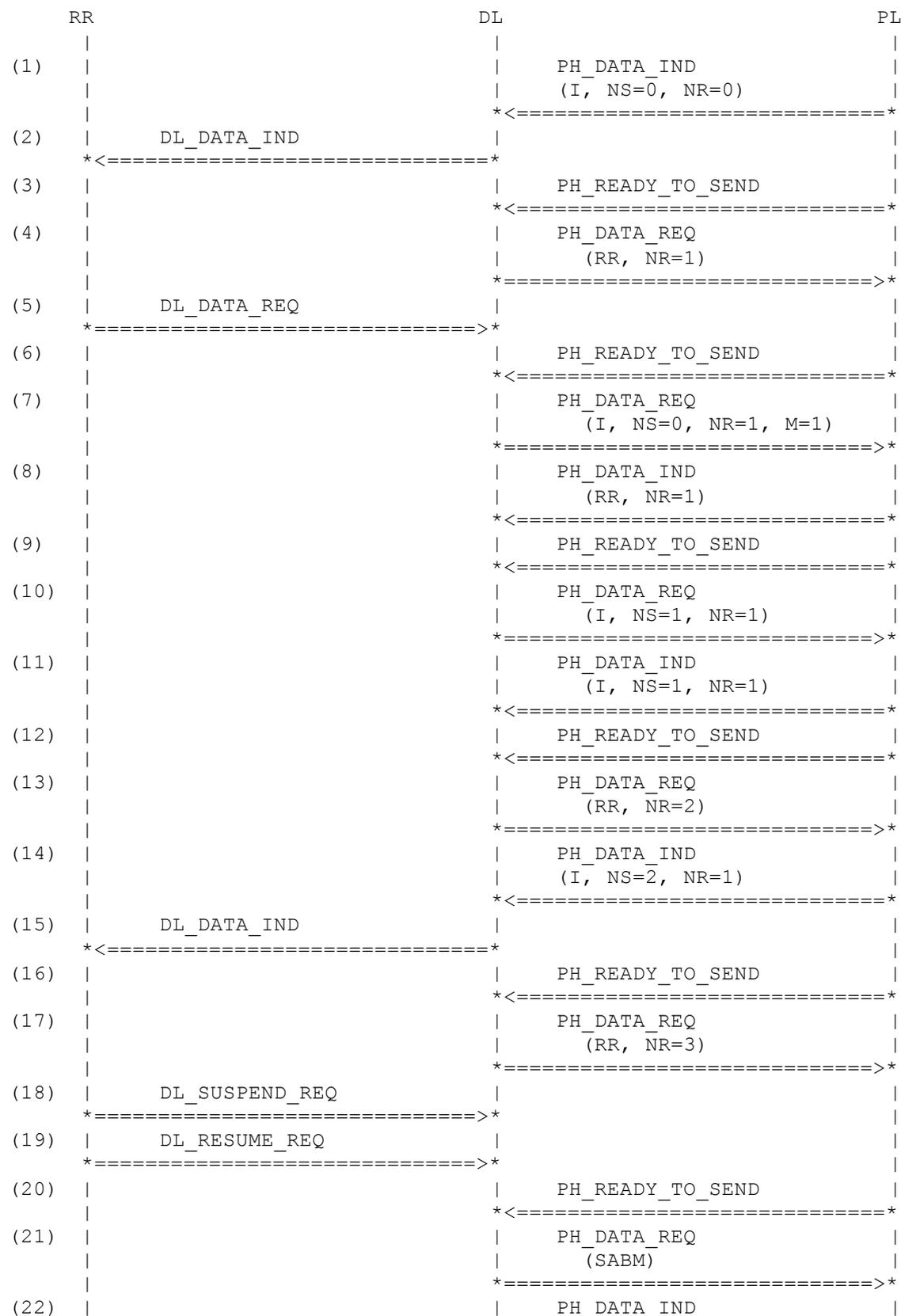
(15)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 LONG_MESSAGE
(16)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(17)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMP
(18)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(19)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_A_UL
(20)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_A_DL
(21)	DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(22)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(23)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_ASSIGN_CMP
(24)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_ASSIGN_CMP
(25)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(26)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_1_0_M1
(27)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_2
(28)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(29)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_2_0_M2

(30) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_3
(31) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(32) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.31 DL221: Segmentation and Concatenation (25.2.2.3), FACCH Fullrate

Description: To test the proper use of segmentation and concatenation, suspend and resume.
Preamble: DL201



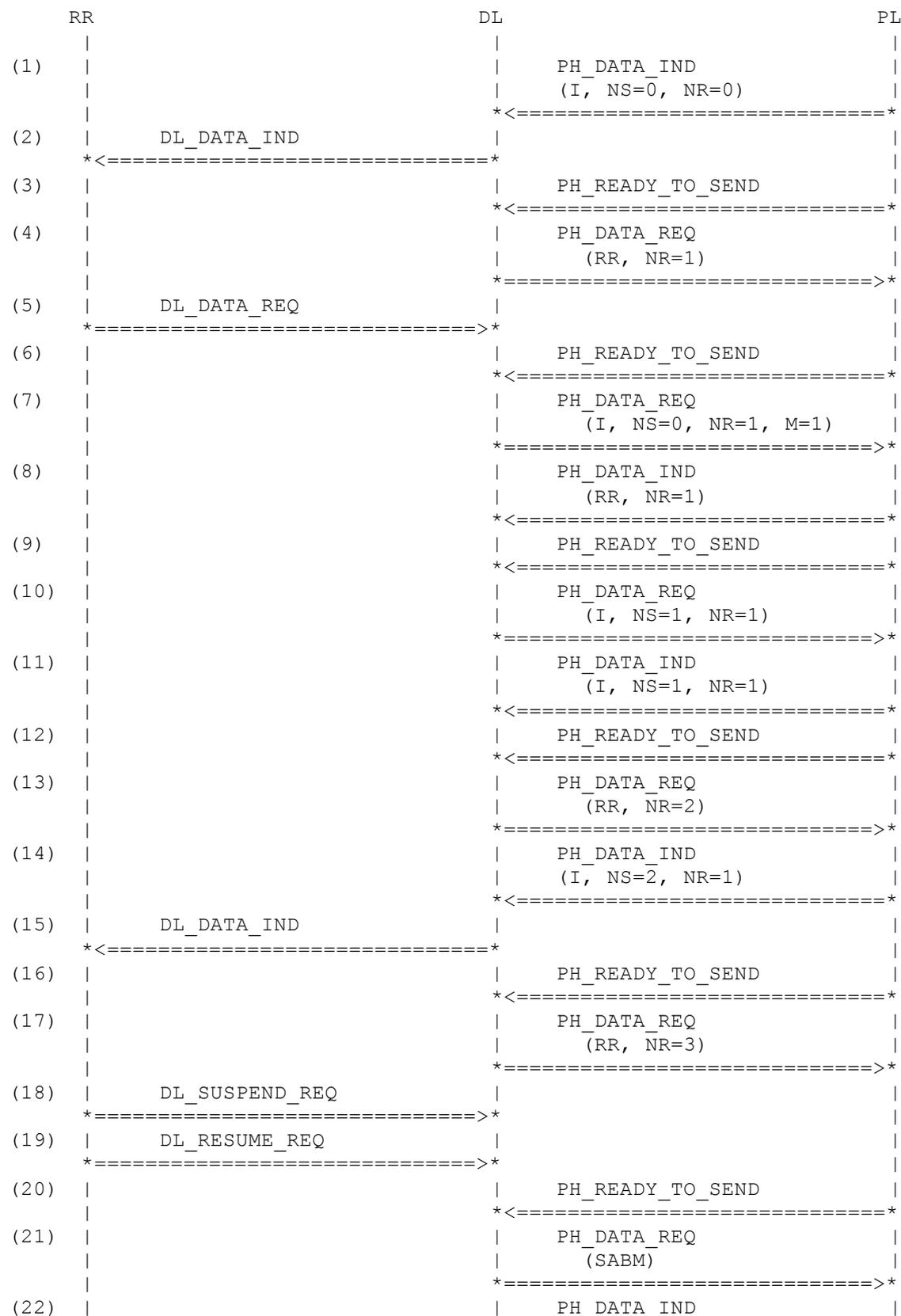
(8)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH SAPI_0 BS_RR_1
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_1_1_M2
(11)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_1_1_M1
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_2
(14)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_2_1_M2
(15)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 LONG_MESSAGE
(16)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(17)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_3
(18)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_FACCH SAPI_0
(19)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(20)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(21)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(22)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL

(23)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH SAPI_0
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(26)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_1_0_M1
(29)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_2
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(31)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_2_0_M2
(32)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_3

History: 05.12.97 LE Initial

4.17.32 DL621: Segmentation and Concatenation (25.2.2.3), FACCH Halfrate

Description: To test the proper use of segmentation and concatenation, suspend and resume.
Preamble: DL601



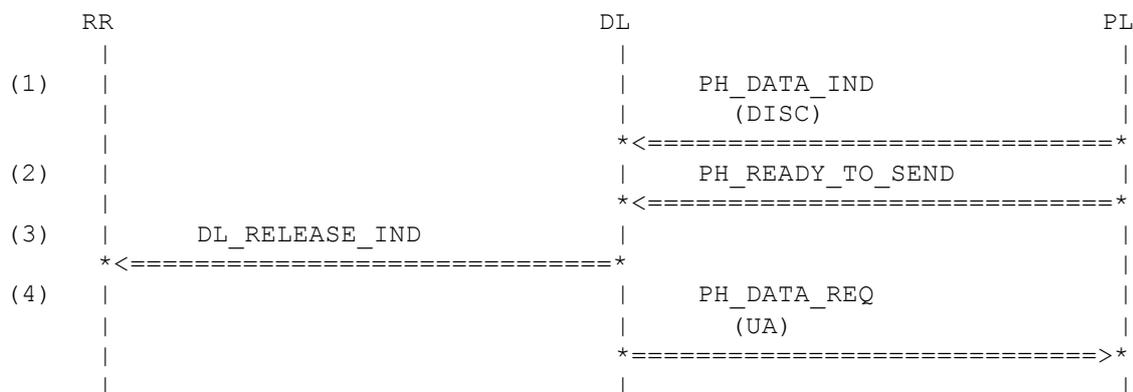
(8)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR SAPI_0 BS_RR_1
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_1_1_M2
(11)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_1_1_M1
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(13)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_2
(14)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_2_1_M2
(15)	DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 LONG_MESSAGE
(16)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(17)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_3
(18)	DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_FACCH_HR SAPI_0
(19)	DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 L3_ASSIGN_CMP
(20)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(21)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_A_UL
(22)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED UA_A_DL

(23)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH_HR SAPI_0
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED I_ASSIGN_CMP
(26)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED RR_ASSIGN_CMP
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_1_0_M1
(29)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_2
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(31)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_2_0_M2
(32)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_3

History: 08.07.98 LE Initial

4.17.33 DL222: Normal Layer 2 Disconnection (25.2.3), SDCCH

Description: To test the normal data link disconnection sequences.
Preamble: DL200



Parametrization

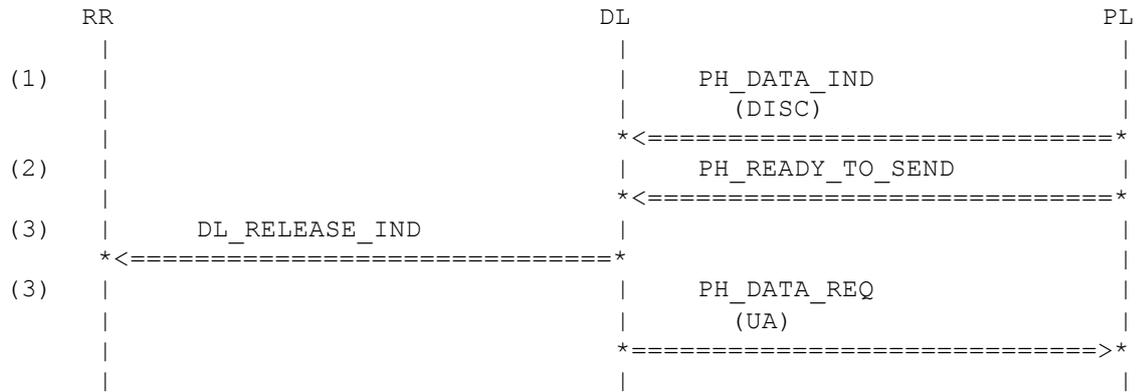
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_FRAME
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_FRAME

History: 05.12.97 LE Initial
 09.07.98 LE First UA then RELEASE_IND

4.17.34 DL223: Normal Layer 2 Disconnection (25.2.3), FACCH Fullrate

Description: To test the normal data link disconnection sequences.

Preamble: DL201



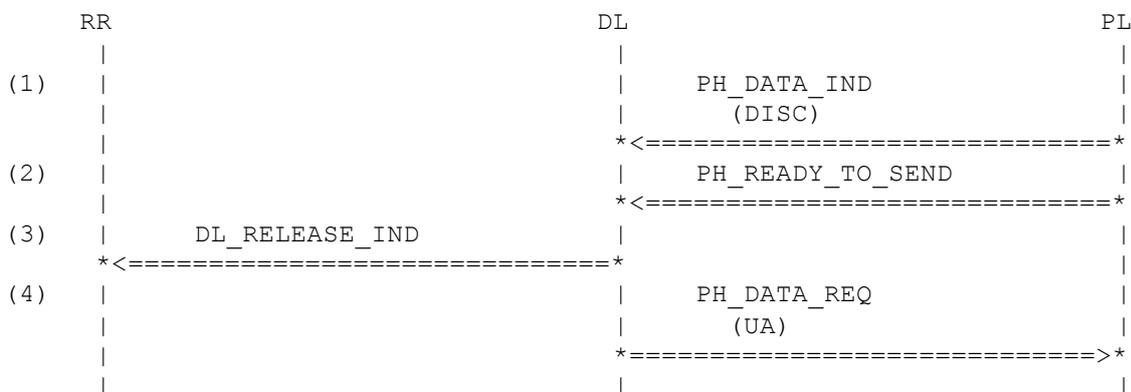
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	DISC_FRAME
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	UA_FRAME

History: 05.12.97 LE Initial

4.17.35 DL623: Normal Layer 2 Disconnection (25.2.3), FACCH Halfrate

Description: To test the normal data link disconnection sequences.
Preamble: DL601



Parametrization

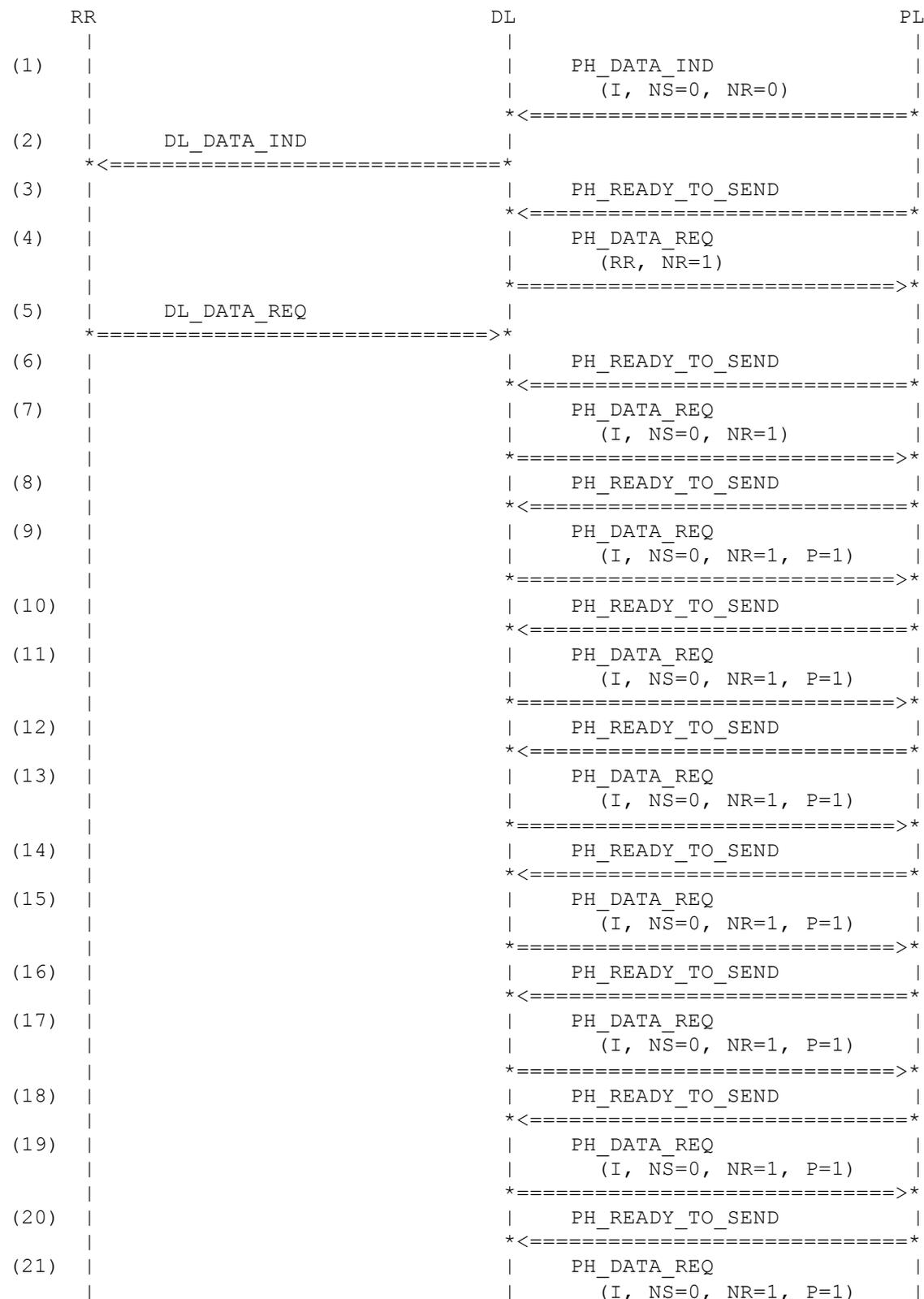
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	DISC_FRAME
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	UA_FRAME

History: 08.07.98 LE Initial

4.17.36 DL224: I Frame Loss (MS to SS), (25.2.4.1), SDCCH

Description: To test that the MS repeats an I frame N200 times with T200 between two I frames and that the MS releases the layer 2 link after N200 repetitions of the I frame in the case when no answer to the I frame is received.

Preamble: DL200



```
(22) | | *=====>*
| | | PH_READY_TO_SEND |
| | *<=====>*
(23) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(24) | | | PH_READY_TO_SEND |
| | *<=====>*
(25) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(26) | | | PH_READY_TO_SEND |
| | *<=====>*
(27) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(28) | | | PH_READY_TO_SEND |
| | *<=====>*
(29) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(30) | | | PH_READY_TO_SEND |
| | *<=====>*
(31) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(32) | | | PH_READY_TO_SEND |
| | *<=====>*
(33) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(34) | | | PH_READY_TO_SEND |
| | *<=====>*
(35) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(36) | | | PH_READY_TO_SEND |
| | *<=====>*
(37) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(38) | | | PH_READY_TO_SEND |
| | *<=====>*
(39) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(40) | | | PH_READY_TO_SEND |
| | *<=====>*
(41) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(42) | | | PH_READY_TO_SEND |
| | *<=====>*
(43) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | *=====>*
(44) | | | PH_READY_TO_SEND |
| | *<=====>*
(45) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
```


(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1
(5)	DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(11)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(13)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(16)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(17)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(18)	PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(19)	PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED MS_I_0_1_P
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(21)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(23)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(25)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(29)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(31)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(33)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(35)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P

(36)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(37)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(39)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(41)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(43)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(45)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(47)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(49)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(51)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH

(53)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1_P
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(55)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_SDCCH SAPI_0 NOT_PRESENT_8BIT
(56)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED EMPTY_FRAME_DCCH
(57)	DL_ESTABLISH_REQ ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_PAGING_RESP
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(59)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_PAGING_RESP
(60)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UA_PAGING_RESP
(61)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_SDCCH SAPI_0
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(63)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.37 DL225: I Frame Loss (MS to SS), (25.2.4.1), FACCH Fullrate (I)

Description: To test that the MS repeats an I frame N200 times with T200 between two I frames and that the MS releases the layer 2 link after N200 repetitions of the I frame in the case when no answer to the I frame is received.

Preamble: DL201

RR	DL	PL
(1)	PH_DATA_IND (I, NS=0, NR=0)	
	<=====	
(2)	DL_DATA_IND	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_DATA_REQ (RR, NR=1)	
	=====>	
(5)	DL_DATA_REQ	
	=====>	
(6)	PH_READY_TO_SEND	
	<=====	
(7)	PH_DATA_REQ (I, NS=0, NR=1)	
	=====>	
(8)	PH_READY_TO_SEND	
	<=====	
(9)	PH_READY_TO_SEND	
	<=====	
(10)	PH_READY_TO_SEND	
	<=====	
(11)	PH_READY_TO_SEND	
	<=====	
(12)	PH_READY_TO_SEND	
	<=====	
(13)	PH_READY_TO_SEND	
	<=====	
(14)	PH_READY_TO_SEND	
	<=====	
(15)	PH_READY_TO_SEND	
	<=====	
(16)	PH_READY_TO_SEND	
	<=====	
(17)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
	=====>	
(18)	PH_READY_TO_SEND	
	<=====	
(19)	PH_READY_TO_SEND	
	<=====	
(20)	PH_READY_TO_SEND	
	<=====	
(21)	PH_READY_TO_SEND	
	<=====	
(22)	PH_READY_TO_SEND	
	<=====	
(23)	PH_READY_TO_SEND	
	<=====	
(24)	PH_READY_TO_SEND	

```

(25) | | *<=====
      | | | PH_READY_TO_SEND |
      | | *<=====
(26) | | | PH_READY_TO_SEND |
      | | *<=====
(27) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(28) | | | PH_READY_TO_SEND |
      | | *<=====
(29) | | | PH_READY_TO_SEND |
      | | *<=====
(30) | | | PH_READY_TO_SEND |
      | | *<=====
(31) | | | PH_READY_TO_SEND |
      | | *<=====
(32) | | | PH_READY_TO_SEND |
      | | *<=====
(33) | | | PH_READY_TO_SEND |
      | | *<=====
(34) | | | PH_READY_TO_SEND |
      | | *<=====
(34) | | | PH_READY_TO_SEND |
      | | *<=====
(36) | | | PH_READY_TO_SEND |
      | | *<=====
(37) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(38) | | | PH_READY_TO_SEND |
      | | *<=====
(39) | | | PH_READY_TO_SEND |
      | | *<=====
(40) | | | PH_READY_TO_SEND |
      | | *<=====
(41) | | | PH_READY_TO_SEND |
      | | *<=====
(42) | | | PH_READY_TO_SEND |
      | | *<=====
(43) | | | PH_READY_TO_SEND |
      | | *<=====
(44) | | | PH_READY_TO_SEND |
      | | *<=====
(45) | | | PH_READY_TO_SEND |
      | | *<=====
(46) | | | PH_READY_TO_SEND |
      | | *<=====
(47) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(48) | | | PH_READY_TO_SEND |
      | | *<=====
(49) | | | PH_READY_TO_SEND |
      | | *<=====
(50) | | | PH_READY_TO_SEND |
      | | *<=====
(51) | | | PH_READY_TO_SEND |
      | | *<=====
(52) | | | PH_READY_TO_SEND |
      | | *<=====
    
```

```
(53) | | PH_READY_TO_SEND |
      | | *<===== |
(54) | | PH_READY_TO_SEND |
      | | *<===== |
(55) | | PH_READY_TO_SEND |
      | | *<===== |
(56) | | PH_READY_TO_SEND |
      | | *<===== |
(57) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(58) | | PH_READY_TO_SEND |
      | | *<===== |
(59) | | PH_READY_TO_SEND |
      | | *<===== |
(60) | | PH_READY_TO_SEND |
      | | *<===== |
(61) | | PH_READY_TO_SEND |
      | | *<===== |
(62) | | PH_READY_TO_SEND |
      | | *<===== |
(63) | | PH_READY_TO_SEND |
      | | *<===== |
(64) | | PH_READY_TO_SEND |
      | | *<===== |
(65) | | PH_READY_TO_SEND |
      | | *<===== |
(66) | | PH_READY_TO_SEND |
      | | *<===== |
(67) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(68) | | PH_READY_TO_SEND |
      | | *<===== |
(69) | | PH_READY_TO_SEND |
      | | *<===== |
(70) | | PH_READY_TO_SEND |
      | | *<===== |
(71) | | PH_READY_TO_SEND |
      | | *<===== |
(72) | | PH_READY_TO_SEND |
      | | *<===== |
(73) | | PH_READY_TO_SEND |
      | | *<===== |
(74) | | PH_READY_TO_SEND |
      | | *<===== |
(75) | | PH_READY_TO_SEND |
      | | *<===== |
(76) | | PH_READY_TO_SEND |
      | | *<===== |
(77) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(78) | | PH_READY_TO_SEND |
      | | *<===== |
(79) | | PH_READY_TO_SEND |
      | | *<===== |
(80) | | PH_READY_TO_SEND |
      | | *<===== |
(81) | | PH_READY_TO_SEND |
```

```

(82) | | *<=====
      | | | PH_READY_TO_SEND |
      | | *<=====
(83) | | | PH_READY_TO_SEND |
      | | *<=====
(84) | | | PH_READY_TO_SEND |
      | | *<=====
(85) | | | PH_READY_TO_SEND |
      | | *<=====
(86) | | | PH_READY_TO_SEND |
      | | *<=====
(87) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(88) | | | PH_READY_TO_SEND |
      | | *<=====
(89) | | | PH_READY_TO_SEND |
      | | *<=====
(90) | | | PH_READY_TO_SEND |
      | | *<=====
(91) | | | PH_READY_TO_SEND |
      | | *<=====
(92) | | | PH_READY_TO_SEND |
      | | *<=====
(93) | | | PH_READY_TO_SEND |
      | | *<=====
(94) | | | PH_READY_TO_SEND |
      | | *<=====
(95) | | | PH_READY_TO_SEND |
      | | *<=====
(96) | | | PH_READY_TO_SEND |
      | | *<=====
(97) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(98) | | | PH_READY_TO_SEND |
      | | *<=====
(99) | | | PH_READY_TO_SEND |
      | | *<=====
(100) | | | PH_READY_TO_SEND |
      | | *<=====
(101) | | | PH_READY_TO_SEND |
      | | *<=====
(102) | | | PH_READY_TO_SEND |
      | | *<=====
(103) | | | PH_READY_TO_SEND |
      | | *<=====
(104) | | | PH_READY_TO_SEND |
      | | *<=====
(105) | | | PH_READY_TO_SEND |
      | | *<=====
(106) | | | PH_READY_TO_SEND |
      | | *<=====
(107) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(108) | | | PH_READY_TO_SEND |
      | | *<=====
(109) | | | PH_READY_TO_SEND |
      | | *<=====
    
```

```

(110) | | PH_READY_TO_SEND |
      | | *<=====*
(111) | | PH_READY_TO_SEND |
      | | *<=====*
(112) | | PH_READY_TO_SEND |
      | | *<=====*
(113) | | PH_READY_TO_SEND |
      | | *<=====*
(114) | | PH_READY_TO_SEND |
      | | *<=====*
(115) | | PH_READY_TO_SEND |
      | | *<=====*
(116) | | PH_READY_TO_SEND |
      | | *<=====*
(117) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(118) | | PH_READY_TO_SEND |
      | | *<=====*
(119) | | PH_READY_TO_SEND |
      | | *<=====*
(120) | | PH_READY_TO_SEND |
      | | *<=====*
(121) | | PH_READY_TO_SEND |
      | | *<=====*
(122) | | PH_READY_TO_SEND |
      | | *<=====*
(123) | | PH_READY_TO_SEND |
      | | *<=====*
(124) | | PH_READY_TO_SEND |
      | | *<=====*
(125) | | PH_READY_TO_SEND |
      | | *<=====*
(126) | | PH_READY_TO_SEND |
      | | *<=====*
(127) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(128) | | PH_READY_TO_SEND |
      | | *<=====*
(129) | | PH_READY_TO_SEND |
      | | *<=====*
(130) | | PH_READY_TO_SEND |
      | | *<=====*
(131) | | PH_READY_TO_SEND |
      | | *<=====*
(132) | | PH_READY_TO_SEND |
      | | *<=====*
(133) | | PH_READY_TO_SEND |
      | | *<=====*
(134) | | PH_READY_TO_SEND |
      | | *<=====*
(135) | | PH_READY_TO_SEND |
      | | *<=====*
(136) | | PH_READY_TO_SEND |
      | | *<=====*
(137) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(138) | | PH_READY_TO_SEND |
    
```

```

(139) | | *<=====
| | | PH_READY_TO_SEND |
| | | *<=====
(140) | | | PH_READY_TO_SEND |
| | | *<=====
(141) | | | PH_READY_TO_SEND |
| | | *<=====
(142) | | | PH_READY_TO_SEND |
| | | *<=====
(143) | | | PH_READY_TO_SEND |
| | | *<=====
(144) | | | PH_READY_TO_SEND |
| | | *<=====
(145) | | | PH_READY_TO_SEND |
| | | *<=====
(146) | | | PH_READY_TO_SEND |
| | | *<=====
(147) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | | *=====>
(148) | | | PH_READY_TO_SEND |
| | | *<=====
(149) | | | PH_READY_TO_SEND |
| | | *<=====
(150) | | | PH_READY_TO_SEND |
| | | *<=====
(151) | | | PH_READY_TO_SEND |
| | | *<=====
(152) | | | PH_READY_TO_SEND |
| | | *<=====
(153) | | | PH_READY_TO_SEND |
| | | *<=====
(154) | | | PH_READY_TO_SEND |
| | | *<=====
(155) | | | PH_READY_TO_SEND |
| | | *<=====
(156) | | | PH_READY_TO_SEND |
| | | *<=====
(157) | | | PH_DATA_REQ |
| | | (I, NS=0, NR=1, P=1) |
| | | *=====>
(158) | | | PH_READY_TO_SEND |
| | | *<=====
(159) | | | PH_READY_TO_SEND |
| | | *<=====
(160) | | | PH_READY_TO_SEND |
| | | *<=====
(161) | | | PH_READY_TO_SEND |
| | | *<=====
(162) | | | PH_READY_TO_SEND |
| | | *<=====
(163) | | | PH_READY_TO_SEND |
| | | *<=====
(164) | | | PH_READY_TO_SEND |
| | | *<=====
(165) | | | PH_READY_TO_SEND |
| | | *<=====
(166) | | | PH_READY_TO_SEND |
| | | *<=====
(167) | | | PH_DATA_REQ |
    
```

```
|      (I, NS=0, NR=1, P=1)      |
|=====*>
(168) | PH_READY_TO_SEND              |
|=====*<=====
(169) | PH_READY_TO_SEND              |
|=====*<=====
(170) | PH_READY_TO_SEND              |
|=====*<=====
(171) | PH_READY_TO_SEND              |
|=====*<=====
(172) | PH_READY_TO_SEND              |
|=====*<=====
(173) | PH_READY_TO_SEND              |
|=====*<=====
(174) | PH_READY_TO_SEND              |
|=====*<=====
(175) | PH_READY_TO_SEND              |
|=====*<=====
(176) | PH_READY_TO_SEND              |
|=====*<=====
(177) | PH_DATA_REQ                   |
|      (I, NS=0, NR=1, P=1)      |
|=====*>
(178) | PH_READY_TO_SEND              |
|=====*<=====
(179) | PH_READY_TO_SEND              |
|=====*<=====
(180) | PH_READY_TO_SEND              |
|=====*<=====
(181) | PH_READY_TO_SEND              |
|=====*<=====
(182) | PH_READY_TO_SEND              |
|=====*<=====
(183) | PH_READY_TO_SEND              |
|=====*<=====
(184) | PH_READY_TO_SEND              |
|=====*<=====
(185) | PH_READY_TO_SEND              |
|=====*<=====
(186) | PH_READY_TO_SEND              |
|=====*<=====
(187) | PH_DATA_REQ                   |
|      (I, NS=0, NR=1, P=1)      |
|=====*>
(188) | PH_READY_TO_SEND              |
|=====*<=====
(189) | PH_READY_TO_SEND              |
|=====*<=====
(190) | PH_READY_TO_SEND              |
|=====*<=====
(191) | PH_READY_TO_SEND              |
|=====*<=====
(192) | PH_READY_TO_SEND              |
|=====*<=====
(193) | PH_READY_TO_SEND              |
|=====*<=====
(194) | PH_READY_TO_SEND              |
|=====*<=====
(195) | PH_READY_TO_SEND              |
|=====*<=====
```

```

(196) | | PH_READY_TO_SEND |
      | | *<===== |
(197) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
      | | |
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(17)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(63)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(64)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(65)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(67)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(68)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(69)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(70)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(71)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(72)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(73)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(74)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(75)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(76)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(77)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(78)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(79)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(80)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(81)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(82)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(83)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(84)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(85)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(86)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(87)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(88)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(89)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(90)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(91)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(92)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(93)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(94)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(95)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(96)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(97)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(98)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(99)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(100)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(101)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(102)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(103)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(104)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(105)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(106)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(107)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(108)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(109)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(110)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(111)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(112)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(113)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(114)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(115)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(116)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(117)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(118)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(119)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(120)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(121)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(122)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(123)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(124)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(125)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(126)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(127)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(128)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(129)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(130)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(131)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(132)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(133)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(134)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(135)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(136)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(137)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(138)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(139)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(140)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(141)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(142)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(143)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(144)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(145)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(146)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(147)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(148)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(149)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(150)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(151)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(152)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(153)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(154)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(155)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(156)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(157)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(158)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(159)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(160)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(161)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(162)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(163)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(164)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(165)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(166)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(167)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(168)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(169)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(170)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(171)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(172)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(173)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(174)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(175)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(176)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(177)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(178)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(179)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(180)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(181)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(182)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(183)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(184)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(185)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(186)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(187)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(188)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(189)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(190)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(191)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(192)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(193)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(194)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(195)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(196)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(197)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P

History: 05.12.97 LE Initial

4.17.38 DL250: I Frame Loss (MS to SS), (25.2.4.1), FACCH Fullrate (II)

Description: To test that the MS repeats an I frame N200 times with T200 between two I frames and that the MS releases the layer 2 link after N200 repetitions of the I frame in the case when no answer to the I frame is received.

Preamble: DL225

RR	DL	PL
(1)	PH_READY_TO_SEND	
(2)	PH_READY_TO_SEND	
(3)	PH_READY_TO_SEND	
(4)	PH_READY_TO_SEND	
(5)	PH_READY_TO_SEND	
(6)	PH_READY_TO_SEND	
(7)	PH_READY_TO_SEND	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_READY_TO_SEND	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_READY_TO_SEND	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_READY_TO_SEND	
(24)	PH_READY_TO_SEND	
(25)	PH_READY_TO_SEND	

```
(26) | *<=====*  
      | | PH_READY_TO_SEND |  
      | *<=====*  
(27) | | PH_READY_TO_SEND |  
      | *<=====*  
(28) | | PH_READY_TO_SEND |  
      | *<=====*  
(29) | | PH_READY_TO_SEND |  
      | *<=====*  
(30) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(31) | | PH_READY_TO_SEND |  
      | *<=====*  
(32) | | PH_READY_TO_SEND |  
      | *<=====*  
(33) | | PH_READY_TO_SEND |  
      | *<=====*  
(34) | | PH_READY_TO_SEND |  
      | *<=====*  
(35) | | PH_READY_TO_SEND |  
      | *<=====*  
(36) | | PH_READY_TO_SEND |  
      | *<=====*  
(37) | | PH_READY_TO_SEND |  
      | *<=====*  
(38) | | PH_READY_TO_SEND |  
      | *<=====*  
(39) | | PH_READY_TO_SEND |  
      | *<=====*  
(40) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(41) | | PH_READY_TO_SEND |  
      | *<=====*  
(42) | | PH_READY_TO_SEND |  
      | *<=====*  
(43) | | PH_READY_TO_SEND |  
      | *<=====*  
(44) | | PH_READY_TO_SEND |  
      | *<=====*  
(45) | | PH_READY_TO_SEND |  
      | *<=====*  
(46) | | PH_READY_TO_SEND |  
      | *<=====*  
(47) | | PH_READY_TO_SEND |  
      | *<=====*  
(48) | | PH_READY_TO_SEND |  
      | *<=====*  
(49) | | PH_READY_TO_SEND |  
      | *<=====*  
(50) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(51) | | PH_READY_TO_SEND |  
      | *<=====*  
(52) | | PH_READY_TO_SEND |  
      | *<=====*  
(53) | | PH_READY_TO_SEND |  
      | *<=====*
```

```
(54) | | PH_READY_TO_SEND |
      | | *<===== *
(55) | | PH_READY_TO_SEND |
      | | *<===== *
(56) | | PH_READY_TO_SEND |
      | | *<===== *
(57) | | PH_READY_TO_SEND |
      | | *<===== *
(58) | | PH_READY_TO_SEND |
      | | *<===== *
(59) | | PH_READY_TO_SEND |
      | | *<===== *
(60) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> *
(61) | | PH_READY_TO_SEND |
      | | *<===== *
(62) | | PH_READY_TO_SEND |
      | | *<===== *
(63) | | PH_READY_TO_SEND |
      | | *<===== *
(64) | | PH_READY_TO_SEND |
      | | *<===== *
(65) | | PH_READY_TO_SEND |
      | | *<===== *
(66) | | PH_READY_TO_SEND |
      | | *<===== *
(67) | | PH_READY_TO_SEND |
      | | *<===== *
(68) | | PH_READY_TO_SEND |
      | | *<===== *
(69) | | PH_READY_TO_SEND |
      | | *<===== *
(70) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> *
(71) | | PH_READY_TO_SEND |
      | | *<===== *
(72) | | PH_READY_TO_SEND |
      | | *<===== *
(73) | | PH_READY_TO_SEND |
      | | *<===== *
(74) | | PH_READY_TO_SEND |
      | | *<===== *
(75) | | PH_READY_TO_SEND |
      | | *<===== *
(76) | | PH_READY_TO_SEND |
      | | *<===== *
(77) | | PH_READY_TO_SEND |
      | | *<===== *
(78) | | PH_READY_TO_SEND |
      | | *<===== *
(79) | | PH_READY_TO_SEND |
      | | *<===== *
(80) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> *
(81) | | PH_READY_TO_SEND |
      | | *<===== *
(82) | | PH_READY_TO_SEND |
```

```
(83) | | *<=====*  
| | | PH_READY_TO_SEND |  
| | *<=====*  
(84) | | | PH_READY_TO_SEND |  
| | *<=====*  
(85) | | | PH_READY_TO_SEND |  
| | *<=====*  
(86) | | | PH_READY_TO_SEND |  
| | *<=====*  
(87) | | | PH_READY_TO_SEND |  
| | *<=====*  
(88) | | | PH_READY_TO_SEND |  
| | *<=====*  
(89) | | | PH_READY_TO_SEND |  
| | *<=====*  
(90) | | | PH_DATA_REQ |  
| | | (I, NS=0, NR=1, P=1) |  
| | *=====>*  
(91) | | | PH_READY_TO_SEND |  
| | *<=====*  
(92) | | | PH_READY_TO_SEND |  
| | *<=====*  
(93) | | | PH_READY_TO_SEND |  
| | *<=====*  
(94) | | | PH_READY_TO_SEND |  
| | *<=====*  
(95) | | | PH_READY_TO_SEND |  
| | *<=====*  
(96) | | | PH_READY_TO_SEND |  
| | *<=====*  
(97) | | | PH_READY_TO_SEND |  
| | *<=====*  
(98) | | | PH_READY_TO_SEND |  
| | *<=====*  
(99) | | | PH_READY_TO_SEND |  
| | *<=====*  
(100) | | | PH_DATA_REQ |  
| | | (I, NS=0, NR=1, P=1) |  
| | *=====>*  
(101) | | | PH_READY_TO_SEND |  
| | *<=====*  
(102) | | | PH_READY_TO_SEND |  
| | *<=====*  
(103) | | | PH_READY_TO_SEND |  
| | *<=====*  
(104) | | | PH_READY_TO_SEND |  
| | *<=====*  
(105) | | | PH_READY_TO_SEND |  
| | *<=====*  
(106) | | | PH_READY_TO_SEND |  
| | *<=====*  
(107) | | | PH_READY_TO_SEND |  
| | *<=====*  
(108) | | | PH_READY_TO_SEND |  
| | *<=====*  
(109) | | | PH_READY_TO_SEND |  
| | *<=====*  
(110) | | | PH_DATA_REQ |  
| | | (I, NS=0, NR=1, P=1) |  
| | *=====>*
```

```
(111) | | PH_READY_TO_SEND |
| | *<=====*
(112) | | PH_READY_TO_SEND |
| | *<=====*
(113) | | PH_READY_TO_SEND |
| | *<=====*
(114) | | PH_READY_TO_SEND |
| | *<=====*
(115) | | PH_READY_TO_SEND |
| | *<=====*
(116) | | PH_READY_TO_SEND |
| | *<=====*
(117) | | PH_READY_TO_SEND |
| | *<=====*
(118) | | PH_READY_TO_SEND |
| | *<=====*
(119) | | PH_READY_TO_SEND |
| | *<=====*
(120) | | PH_DATA_REQ |
| | (I, NS=0, NR=1, P=1) |
| | *=====>*
(121) | | PH_READY_TO_SEND |
| | *<=====*
(122) | | PH_READY_TO_SEND |
| | *<=====*
(123) | | PH_READY_TO_SEND |
| | *<=====*
(124) | | PH_READY_TO_SEND |
| | *<=====*
(125) | | PH_READY_TO_SEND |
| | *<=====*
(126) | | PH_READY_TO_SEND |
| | *<=====*
(127) | | PH_READY_TO_SEND |
| | *<=====*
(128) | | PH_READY_TO_SEND |
| | *<=====*
(129) | | PH_READY_TO_SEND |
| | *<=====*
(130) | | PH_DATA_REQ |
| | (I, NS=0, NR=1, P=1) |
| | *=====>*
(131) | | PH_READY_TO_SEND |
| | *<=====*
(132) | | PH_READY_TO_SEND |
| | *<=====*
(133) | | PH_READY_TO_SEND |
| | *<=====*
(134) | | PH_READY_TO_SEND |
| | *<=====*
(135) | | PH_READY_TO_SEND |
| | *<=====*
(136) | | PH_READY_TO_SEND |
| | *<=====*
(137) | | PH_READY_TO_SEND |
| | *<=====*
(138) | | PH_READY_TO_SEND |
| | *<=====*
(139) | | PH_READY_TO_SEND |
| | *<=====*
```

```

(140) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>* |
(141) | | PH_READY_TO_SEND |
      | | *<=====* |
(142) | | PH_READY_TO_SEND |
      | | *<=====* |
(143) | | PH_READY_TO_SEND |
      | | *<=====* |
(144) | | PH_READY_TO_SEND |
      | | *<=====* |
(145) | | PH_READY_TO_SEND |
      | | *<=====* |
(146) | | PH_READY_TO_SEND |
      | | *<=====* |
(147) | | PH_READY_TO_SEND |
      | | *<=====* |
(148) | | PH_READY_TO_SEND |
      | | *<=====* |
(149) | | PH_READY_TO_SEND |
      | | *<=====* |
(150) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>* |
(151) | | PH_READY_TO_SEND |
      | | *<=====* |
(152) | | PH_READY_TO_SEND |
      | | *<=====* |
(153) | | PH_READY_TO_SEND |
      | | *<=====* |
(154) | | PH_READY_TO_SEND |
      | | *<=====* |
(155) | | PH_READY_TO_SEND |
      | | *<=====* |
(156) | | PH_READY_TO_SEND |
      | | *<=====* |
(157) | | PH_READY_TO_SEND |
      | | *<=====* |
(158) | | PH_READY_TO_SEND |
      | | *<=====* |
(159) | | PH_READY_TO_SEND |
      | | *<=====* |
(160) | | DL_RELEASE_IND |
      | | *<=====* |
(161) | | DL_ESTABLISH_REQ |
      | | (layer 3 Msg) |
      | | *=====>* |
(162) | | PH_READY_TO_SEND |
      | | *<=====* |
(163) | | PH_DATA_REQ |
      | | (SABM) |
      | | *=====>* |
(164) | | PH_DATA_IND |
      | | (UA) |
      | | *<=====* |
(165) | | DL_ESTABLISH_CNF |
      | | *<=====* |
    
```

Parametrization

Primitive	Parameter	Value
-----------	-----------	-------

(1)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(2)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(3)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(6)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(9)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ	ch_type	CH_TYPE_FACCH
		dummy	NOT_USED
		sdu	MS_I_0_1_P
(11)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(17)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(18)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(19)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ	ch_type	CH_TYPE_FACCH
		dummy	NOT_USED
		sdu	MS_I_0_1_P
(21)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(63)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(64)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(65)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(67)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(68)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(69)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(70)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(71)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(72)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(73)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(74)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(75)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(76)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(77)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(78)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(79)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(80)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(81)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(82)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(83)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(84)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(85)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(86)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(87)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(88)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(89)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(90)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(91)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(92)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(93)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(94)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(95)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(96)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(97)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(98)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(99)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(100)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(101)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(102)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(103)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(104)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(105)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(106)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(107)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(108)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(109)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(110)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(111)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(112)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(113)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(114)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(115)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(116)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(117)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(118)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(119)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(120)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(121)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(122)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(123)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(124)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(125)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(126)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(127)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(128)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(129)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(130)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(131)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(132)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(133)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(134)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(135)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(136)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(137)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(138)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(139)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(140)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(141)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(142)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(143)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(144)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(145)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(146)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(147)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(148)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(149)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(150)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1_P
(151)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(152)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(153)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(154)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(155)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(156)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(157)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(158)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(159)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(160)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT
(161)	DL_ESTABLISH_REQ ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_PAGING_RESP
(162)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(163)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_PAGING_RESP
(164)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_PAGING_RESP
(165)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH SAPI_0

History: 05.12.97 LE Initial

4.17.39 DL625: I Frame Loss (MS to SS), (25.2.4.1), FACCH Halfrate (I)

Description: To test that the MS repeats an I frame N200 times with T200 between two I frames and that the MS releases the layer 2 link after N200 repetitions of the I frame in the case when no answer to the I frame is received.

Preamble: DL601

RR	DL	PL
(1)	PH_DATA_IND (I, NS=0, NR=0)	
(2)	DL_DATA_IND	
(3)	PH_READY_TO_SEND	
(4)	PH_DATA_REQ (RR, NR=1)	
(5)	DL_DATA_REQ	
(6)	PH_READY_TO_SEND	
(7)	PH_DATA_REQ (I, NS=0, NR=1)	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_READY_TO_SEND	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_READY_TO_SEND	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_READY_TO_SEND	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_READY_TO_SEND	
(24)	PH_READY_TO_SEND	

```
(25) | | *<=====*
```

```
| | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(26) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(27) | | | PH_DATA_REQ |
```

```
| | | (I, NS=0, NR=1, P=1) |
```

```
| | | *=====>*
```

```
(28) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(29) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(30) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(31) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(32) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(33) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(34) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(35) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(36) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(37) | | | PH_DATA_REQ |
```

```
| | | (I, NS=0, NR=1, P=1) |
```

```
| | | *=====>*
```

```
(38) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(39) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(40) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(41) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(42) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(43) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(44) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(45) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(46) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(47) | | | PH_DATA_REQ |
```

```
| | | (I, NS=0, NR=1, P=1) |
```

```
| | | *=====>*
```

```
(48) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(49) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(50) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(51) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(52) | | | PH_READY_TO_SEND |
```

```
| | | *<=====*
```

```
(53) | | PH_READY_TO_SEND |
      | | *<=====*
(54) | | PH_READY_TO_SEND |
      | | *<=====*
(55) | | PH_READY_TO_SEND |
      | | *<=====*
(56) | | PH_READY_TO_SEND |
      | | *<=====*
(57) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(58) | | PH_READY_TO_SEND |
      | | *<=====*
(59) | | PH_READY_TO_SEND |
      | | *<=====*
(60) | | PH_READY_TO_SEND |
      | | *<=====*
(61) | | PH_READY_TO_SEND |
      | | *<=====*
(62) | | PH_READY_TO_SEND |
      | | *<=====*
(63) | | PH_READY_TO_SEND |
      | | *<=====*
(64) | | PH_READY_TO_SEND |
      | | *<=====*
(65) | | PH_READY_TO_SEND |
      | | *<=====*
(66) | | PH_READY_TO_SEND |
      | | *<=====*
(67) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(68) | | PH_READY_TO_SEND |
      | | *<=====*
(69) | | PH_READY_TO_SEND |
      | | *<=====*
(70) | | PH_READY_TO_SEND |
      | | *<=====*
(71) | | PH_READY_TO_SEND |
      | | *<=====*
(72) | | PH_READY_TO_SEND |
      | | *<=====*
(73) | | PH_READY_TO_SEND |
      | | *<=====*
(74) | | PH_READY_TO_SEND |
      | | *<=====*
(75) | | PH_READY_TO_SEND |
      | | *<=====*
(76) | | PH_READY_TO_SEND |
      | | *<=====*
(77) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====>*
(78) | | PH_READY_TO_SEND |
      | | *<=====*
(79) | | PH_READY_TO_SEND |
      | | *<=====*
(80) | | PH_READY_TO_SEND |
      | | *<=====*
(81) | | PH_READY_TO_SEND |
```

```
(82) | *<=====*  
      | | PH_READY_TO_SEND |  
      | *<=====*  
(83) | | PH_READY_TO_SEND |  
      | *<=====*  
(84) | | PH_READY_TO_SEND |  
      | *<=====*  
(85) | | PH_READY_TO_SEND |  
      | *<=====*  
(86) | | PH_READY_TO_SEND |  
      | *<=====*  
(87) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(88) | | PH_READY_TO_SEND |  
      | *<=====*  
(89) | | PH_READY_TO_SEND |  
      | *<=====*  
(90) | | PH_READY_TO_SEND |  
      | *<=====*  
(91) | | PH_READY_TO_SEND |  
      | *<=====*  
(92) | | PH_READY_TO_SEND |  
      | *<=====*  
(93) | | PH_READY_TO_SEND |  
      | *<=====*  
(94) | | PH_READY_TO_SEND |  
      | *<=====*  
(95) | | PH_READY_TO_SEND |  
      | *<=====*  
(96) | | PH_READY_TO_SEND |  
      | *<=====*  
(97) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(98) | | PH_READY_TO_SEND |  
      | *<=====*  
(99) | | PH_READY_TO_SEND |  
      | *<=====*  
(100) | | PH_READY_TO_SEND |  
      | *<=====*  
(101) | | PH_READY_TO_SEND |  
      | *<=====*  
(102) | | PH_READY_TO_SEND |  
      | *<=====*  
(103) | | PH_READY_TO_SEND |  
      | *<=====*  
(104) | | PH_READY_TO_SEND |  
      | *<=====*  
(105) | | PH_READY_TO_SEND |  
      | *<=====*  
(106) | | PH_READY_TO_SEND |  
      | *<=====*  
(107) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(108) | | PH_READY_TO_SEND |  
      | *<=====*  
(109) | | PH_READY_TO_SEND |  
      | *<=====*
```

```

(110) | | PH_READY_TO_SEND |
| | *<=====|
(111) | | PH_READY_TO_SEND |
| | *<=====|
(112) | | PH_READY_TO_SEND |
| | *<=====|
(113) | | PH_READY_TO_SEND |
| | *<=====|
(114) | | PH_READY_TO_SEND |
| | *<=====|
(115) | | PH_READY_TO_SEND |
| | *<=====|
(116) | | PH_READY_TO_SEND |
| | *<=====|
(117) | | PH_DATA_REQ |
| | (I, NS=0, NR=1, P=1) |
| | *=====>|
(118) | | PH_READY_TO_SEND |
| | *<=====|
(119) | | PH_READY_TO_SEND |
| | *<=====|
(120) | | PH_READY_TO_SEND |
| | *<=====|
(121) | | PH_READY_TO_SEND |
| | *<=====|
(122) | | PH_READY_TO_SEND |
| | *<=====|
(123) | | PH_READY_TO_SEND |
| | *<=====|
(124) | | PH_READY_TO_SEND |
| | *<=====|
(125) | | PH_READY_TO_SEND |
| | *<=====|
(126) | | PH_READY_TO_SEND |
| | *<=====|
(127) | | PH_DATA_REQ |
| | (I, NS=0, NR=1, P=1) |
| | *=====>|
(128) | | PH_READY_TO_SEND |
| | *<=====|
(129) | | PH_READY_TO_SEND |
| | *<=====|
(130) | | PH_READY_TO_SEND |
| | *<=====|
(131) | | PH_READY_TO_SEND |
| | *<=====|
(132) | | PH_READY_TO_SEND |
| | *<=====|
(133) | | PH_READY_TO_SEND |
| | *<=====|
(134) | | PH_READY_TO_SEND |
| | *<=====|
(135) | | PH_READY_TO_SEND |
| | *<=====|
(136) | | PH_READY_TO_SEND |
| | *<=====|
(137) | | PH_DATA_REQ |
| | (I, NS=0, NR=1, P=1) |
| | *=====>|
(138) | | PH_READY_TO_SEND |
    
```

```
(139) | *<=====*  
      | | PH_READY_TO_SEND |  
      | *<=====*  
(140) | | PH_READY_TO_SEND |  
      | *<=====*  
(141) | | PH_READY_TO_SEND |  
      | *<=====*  
(142) | | PH_READY_TO_SEND |  
      | *<=====*  
(143) | | PH_READY_TO_SEND |  
      | *<=====*  
(144) | | PH_READY_TO_SEND |  
      | *<=====*  
(145) | | PH_READY_TO_SEND |  
      | *<=====*  
(146) | | PH_READY_TO_SEND |  
      | *<=====*  
(147) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(148) | | PH_READY_TO_SEND |  
      | *<=====*  
(149) | | PH_READY_TO_SEND |  
      | *<=====*  
(150) | | PH_READY_TO_SEND |  
      | *<=====*  
(151) | | PH_READY_TO_SEND |  
      | *<=====*  
(152) | | PH_READY_TO_SEND |  
      | *<=====*  
(153) | | PH_READY_TO_SEND |  
      | *<=====*  
(154) | | PH_READY_TO_SEND |  
      | *<=====*  
(155) | | PH_READY_TO_SEND |  
      | *<=====*  
(156) | | PH_READY_TO_SEND |  
      | *<=====*  
(157) | | PH_DATA_REQ |  
      | | (I, NS=0, NR=1, P=1) |  
      | *=====>*  
(158) | | PH_READY_TO_SEND |  
      | *<=====*  
(159) | | PH_READY_TO_SEND |  
      | *<=====*  
(160) | | PH_READY_TO_SEND |  
      | *<=====*  
(161) | | PH_READY_TO_SEND |  
      | *<=====*  
(162) | | PH_READY_TO_SEND |  
      | *<=====*  
(163) | | PH_READY_TO_SEND |  
      | *<=====*  
(164) | | PH_READY_TO_SEND |  
      | *<=====*  
(165) | | PH_READY_TO_SEND |  
      | *<=====*  
(166) | | PH_READY_TO_SEND |  
      | *<=====*  
(167) | | PH_DATA_REQ |
```

```

|          (I, NS=0, NR=1, P=1)          |
|=====>*
(168) | PH_READY_TO_SEND                      |
|=====*<
(169) | PH_READY_TO_SEND                      |
|=====*<
(170) | PH_READY_TO_SEND                      |
|=====*<
(171) | PH_READY_TO_SEND                      |
|=====*<
(172) | PH_READY_TO_SEND                      |
|=====*<
(173) | PH_READY_TO_SEND                      |
|=====*<
(174) | PH_READY_TO_SEND                      |
|=====*<
(175) | PH_READY_TO_SEND                      |
|=====*<
(176) | PH_READY_TO_SEND                      |
|=====*<
(177) | PH_DATA_REQ                            |
|          (I, NS=0, NR=1, P=1)          |
|=====>*
(178) | PH_READY_TO_SEND                      |
|=====*<
(179) | PH_READY_TO_SEND                      |
|=====*<
(180) | PH_READY_TO_SEND                      |
|=====*<
(181) | PH_READY_TO_SEND                      |
|=====*<
(182) | PH_READY_TO_SEND                      |
|=====*<
(183) | PH_READY_TO_SEND                      |
|=====*<
(184) | PH_READY_TO_SEND                      |
|=====*<
(185) | PH_READY_TO_SEND                      |
|=====*<
(186) | PH_READY_TO_SEND                      |
|=====*<
(187) | PH_DATA_REQ                            |
|          (I, NS=0, NR=1, P=1)          |
|=====>*
(188) | PH_READY_TO_SEND                      |
|=====*<
(189) | PH_READY_TO_SEND                      |
|=====*<
(190) | PH_READY_TO_SEND                      |
|=====*<
(191) | PH_READY_TO_SEND                      |
|=====*<
(192) | PH_READY_TO_SEND                      |
|=====*<
(193) | PH_READY_TO_SEND                      |
|=====*<
(194) | PH_READY_TO_SEND                      |
|=====*<
(195) | PH_READY_TO_SEND                      |
|=====*<

```

```

(196) | | PH_READY_TO_SEND |
      | | *<===== |
(197) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
      | | |
    
```

Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_1
(5) DL_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_I_0_1
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(10)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(11)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(12)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(13)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(14)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(15)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR
(16)	ch_type	CH_TYPE_FACCH_HR
	PH_READY_TO_SEND	CH_TYPE_FACCH_HR

(17)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(63)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(64)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(65)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(67)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(68)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(69)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(70)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(71)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(72)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(73)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(74)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(75)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(76)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(77)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(78)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(79)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(80)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(81)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(82)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(83)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(84)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(85)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(86)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(87)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(88)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(89)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(90)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(91)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(92)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(93)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(94)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(95)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(96)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(97)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(98)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(99)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(100)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(101)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(102)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(103)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(104)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(105)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(106)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(107)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(108)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(109)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(110)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(111)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(112)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(113)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(114)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(115)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(116)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(117)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(118)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(119)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(120)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(121)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(122)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(123)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(124)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(125)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(126)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(127)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(128)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(129)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(130)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(131)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(132)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(133)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(134)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(135)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(136)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(137)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(138)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(139)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(140)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(141)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(142)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(143)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(144)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(145)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(146)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(147)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(148)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(149)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(150)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(151)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(152)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(153)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(154)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(155)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(156)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(157)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(158)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(159)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(160)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(161)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(162)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(163)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(164)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(165)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(166)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(167)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(168)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(169)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(170)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(171)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(172)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(173)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(174)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(175)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(176)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(177)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(178)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(179)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(180)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(181)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(182)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(183)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(184)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(185)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(186)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(187)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(188)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(189)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(190)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(191)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(192)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(193)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(194)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(195)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(196)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(197)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P

History: 08.07.98 LE Initial

4.17.40 DL650: I Frame Loss (MS to SS), (25.2.4.1), FACCH Halfrate (II)

Description: To test that the MS repeats an I frame N200 times with T200 between two I frames and that the MS releases the layer 2 link after N200 repetitions of the I frame in the case when no answer to the I frame is received.

Preamble: DL625

RR	DL	PL
(1)	PH_READY_TO_SEND	
(2)	PH_READY_TO_SEND	
(3)	PH_READY_TO_SEND	
(4)	PH_READY_TO_SEND	
(5)	PH_READY_TO_SEND	
(6)	PH_READY_TO_SEND	
(7)	PH_READY_TO_SEND	
(8)	PH_READY_TO_SEND	
(9)	PH_READY_TO_SEND	
(10)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
(11)	PH_READY_TO_SEND	
(12)	PH_READY_TO_SEND	
(13)	PH_READY_TO_SEND	
(14)	PH_READY_TO_SEND	
(15)	PH_READY_TO_SEND	
(16)	PH_READY_TO_SEND	
(17)	PH_READY_TO_SEND	
(18)	PH_READY_TO_SEND	
(19)	PH_READY_TO_SEND	
(20)	PH_DATA_REQ (I, NS=0, NR=1, P=1)	
(21)	PH_READY_TO_SEND	
(22)	PH_READY_TO_SEND	
(23)	PH_READY_TO_SEND	
(24)	PH_READY_TO_SEND	
(25)	PH_READY_TO_SEND	

```
(26) | *<=====*  
| PH_READY_TO_SEND |  
| *<=====*  
(27) | | PH_READY_TO_SEND |  
| *<=====*  
(28) | | PH_READY_TO_SEND |  
| *<=====*  
(29) | | PH_READY_TO_SEND |  
| *<=====*  
(30) | | PH_DATA_REQ |  
| (I, NS=0, NR=1, P=1) |  
| *=====>*  
(31) | | PH_READY_TO_SEND |  
| *<=====*  
(32) | | PH_READY_TO_SEND |  
| *<=====*  
(33) | | PH_READY_TO_SEND |  
| *<=====*  
(34) | | PH_READY_TO_SEND |  
| *<=====*  
(35) | | PH_READY_TO_SEND |  
| *<=====*  
(36) | | PH_READY_TO_SEND |  
| *<=====*  
(37) | | PH_READY_TO_SEND |  
| *<=====*  
(38) | | PH_READY_TO_SEND |  
| *<=====*  
(39) | | PH_READY_TO_SEND |  
| *<=====*  
(40) | | PH_DATA_REQ |  
| (I, NS=0, NR=1, P=1) |  
| *=====>*  
(41) | | PH_READY_TO_SEND |  
| *<=====*  
(42) | | PH_READY_TO_SEND |  
| *<=====*  
(43) | | PH_READY_TO_SEND |  
| *<=====*  
(44) | | PH_READY_TO_SEND |  
| *<=====*  
(45) | | PH_READY_TO_SEND |  
| *<=====*  
(46) | | PH_READY_TO_SEND |  
| *<=====*  
(47) | | PH_READY_TO_SEND |  
| *<=====*  
(48) | | PH_READY_TO_SEND |  
| *<=====*  
(49) | | PH_READY_TO_SEND |  
| *<=====*  
(50) | | PH_DATA_REQ |  
| (I, NS=0, NR=1, P=1) |  
| *=====>*  
(51) | | PH_READY_TO_SEND |  
| *<=====*  
(52) | | PH_READY_TO_SEND |  
| *<=====*  
(53) | | PH_READY_TO_SEND |  
| *<=====*
```

```

(54) | | PH_READY_TO_SEND |
      | | *<===== |
(55) | | PH_READY_TO_SEND |
      | | *<===== |
(56) | | PH_READY_TO_SEND |
      | | *<===== |
(57) | | PH_READY_TO_SEND |
      | | *<===== |
(58) | | PH_READY_TO_SEND |
      | | *<===== |
(59) | | PH_READY_TO_SEND |
      | | *<===== |
(60) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(61) | | PH_READY_TO_SEND |
      | | *<===== |
(62) | | PH_READY_TO_SEND |
      | | *<===== |
(63) | | PH_READY_TO_SEND |
      | | *<===== |
(64) | | PH_READY_TO_SEND |
      | | *<===== |
(65) | | PH_READY_TO_SEND |
      | | *<===== |
(66) | | PH_READY_TO_SEND |
      | | *<===== |
(67) | | PH_READY_TO_SEND |
      | | *<===== |
(68) | | PH_READY_TO_SEND |
      | | *<===== |
(69) | | PH_READY_TO_SEND |
      | | *<===== |
(70) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(71) | | PH_READY_TO_SEND |
      | | *<===== |
(72) | | PH_READY_TO_SEND |
      | | *<===== |
(73) | | PH_READY_TO_SEND |
      | | *<===== |
(74) | | PH_READY_TO_SEND |
      | | *<===== |
(75) | | PH_READY_TO_SEND |
      | | *<===== |
(76) | | PH_READY_TO_SEND |
      | | *<===== |
(77) | | PH_READY_TO_SEND |
      | | *<===== |
(78) | | PH_READY_TO_SEND |
      | | *<===== |
(79) | | PH_READY_TO_SEND |
      | | *<===== |
(80) | | PH_DATA_REQ |
      | | (I, NS=0, NR=1, P=1) |
      | | *=====> |
(81) | | PH_READY_TO_SEND |
      | | *<===== |
(82) | | PH_READY_TO_SEND |
    
```

```

(83) | | *<=====
      | | | PH_READY_TO_SEND |
      | | *<=====
(84) | | | PH_READY_TO_SEND |
      | | *<=====
(85) | | | PH_READY_TO_SEND |
      | | *<=====
(86) | | | PH_READY_TO_SEND |
      | | *<=====
(87) | | | PH_READY_TO_SEND |
      | | *<=====
(88) | | | PH_READY_TO_SEND |
      | | *<=====
(89) | | | PH_READY_TO_SEND |
      | | *<=====
(90) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(91) | | | PH_READY_TO_SEND |
      | | *<=====
(92) | | | PH_READY_TO_SEND |
      | | *<=====
(93) | | | PH_READY_TO_SEND |
      | | *<=====
(94) | | | PH_READY_TO_SEND |
      | | *<=====
(95) | | | PH_READY_TO_SEND |
      | | *<=====
(96) | | | PH_READY_TO_SEND |
      | | *<=====
(97) | | | PH_READY_TO_SEND |
      | | *<=====
(98) | | | PH_READY_TO_SEND |
      | | *<=====
(99) | | | PH_READY_TO_SEND |
      | | *<=====
(100) | | | PH_DATA_REQ |
      | | | (I, NS=0, NR=1, P=1) |
      | | *=====>
(101) | | | PH_READY_TO_SEND |
      | | *<=====
(102) | | | PH_READY_TO_SEND |
      | | *<=====
(103) | | | PH_READY_TO_SEND |
      | | *<=====
(104) | | | PH_READY_TO_SEND |
      | | *<=====
(105) | | | PH_READY_TO_SEND |
      | | *<=====
(106) | | | PH_READY_TO_SEND |
      | | *<=====
(107) | | | PH_READY_TO_SEND |
      | | *<=====
(108) | | | PH_READY_TO_SEND |
      | | *<=====
(109) | | | PH_READY_TO_SEND |
      | | *<=====
(110) | | DL_RELEASE_IND |
      | | *<=====
(111) | | DL_ESTABLISH_REQ |
    
```


(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(22)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(25)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(26)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(29)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(30)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(33)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(34)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(37)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(38)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(41)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(42)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(44)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(45)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(46)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(48)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(49)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(50)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(52)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(53)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(54)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(56)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(57)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(58)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(61)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(62)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(63)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(64)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(65)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(66)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(67)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(68)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(69)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(70)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(71)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(72)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(73)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(74)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(75)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(76)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(77)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(78)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(79)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR

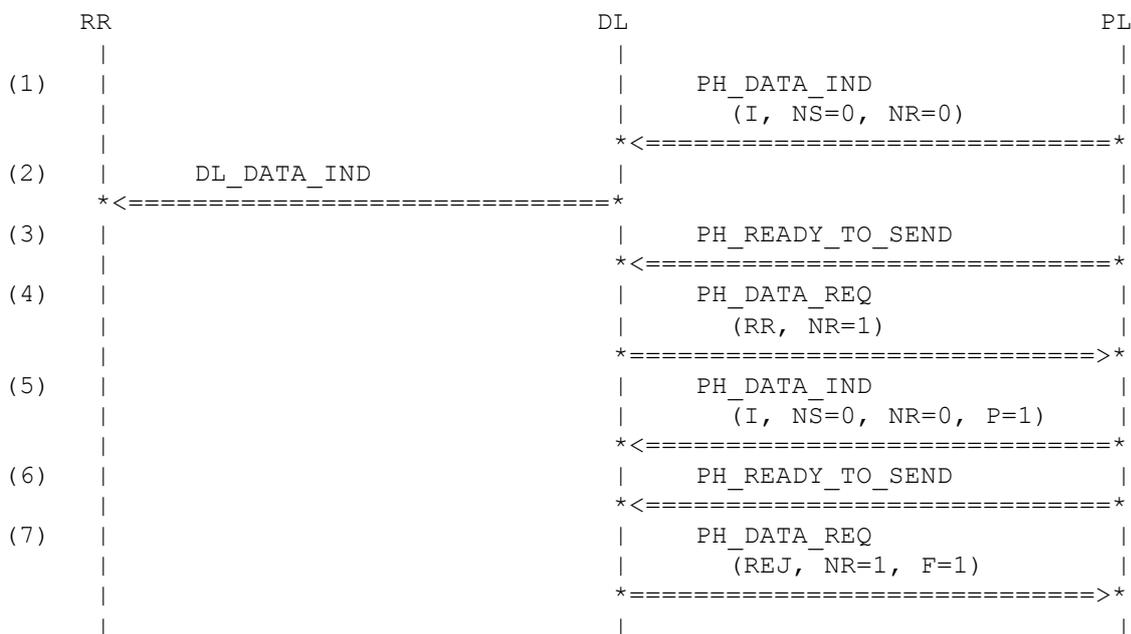
(80)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(81)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(82)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(83)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(84)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(85)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(86)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(87)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(88)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(89)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(90)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P
(91)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(92)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(93)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(94)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(95)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(96)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(97)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(98)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(99)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(100)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1_P

(101)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(102)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(103)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(104)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(105)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(106)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(107)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(108)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(109)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(110)	DL_RELEASE_IND ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT
(111)	DL_ESTABLISH_REQ ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 L3_PAGING_RESP
(112)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(113)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED SABM_PAGING_RESP
(114)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED UA_PAGING_RESP
(115)	DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_FACCH_HR SAPI_0

History: 08.07.98 LE Initial

4.17.41 DL226: RR response frame loss (MS to SS), (25.2.4.3), SDCCH

Description: To test the layer 2 recovery mechanism in the event of RR frame loss.
Preamble: DL200



Parametrization

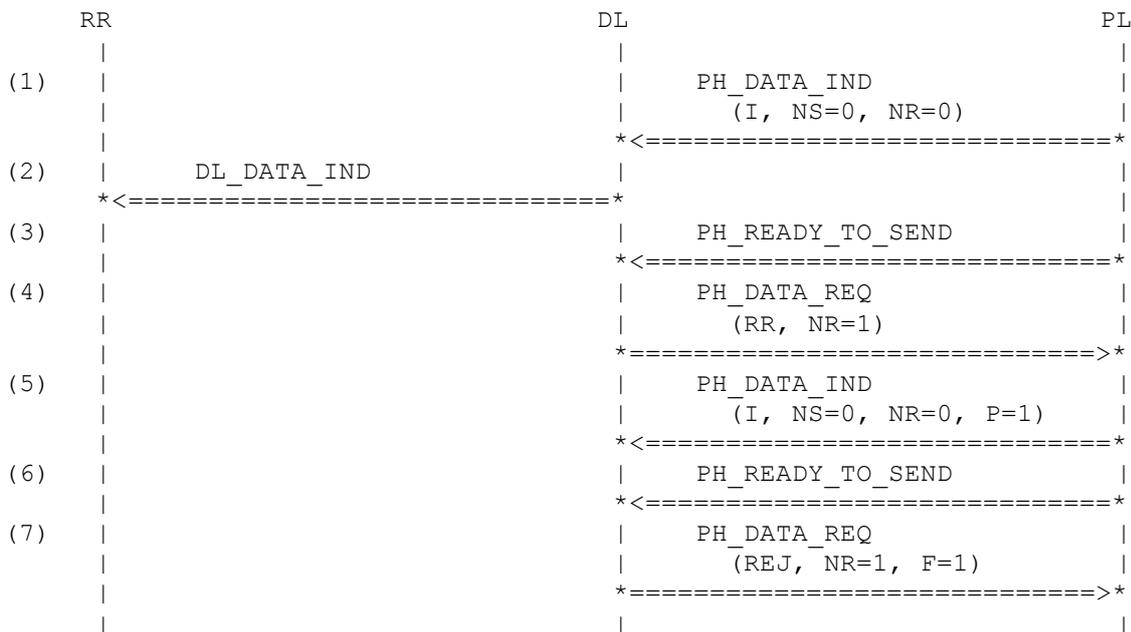
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	MS_RR_1
(5) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	BS_I_0_0_P
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	MS_REJ_1_F

History: 05.12.97 LE Initial

4.17.42 DL227: RR response frame loss (MS to SS), (25.2.4.3), FACCH Fullrate

Description: To test the layer 2 recovery mechanism in the event of RR frame loss.

Preamble: DL201



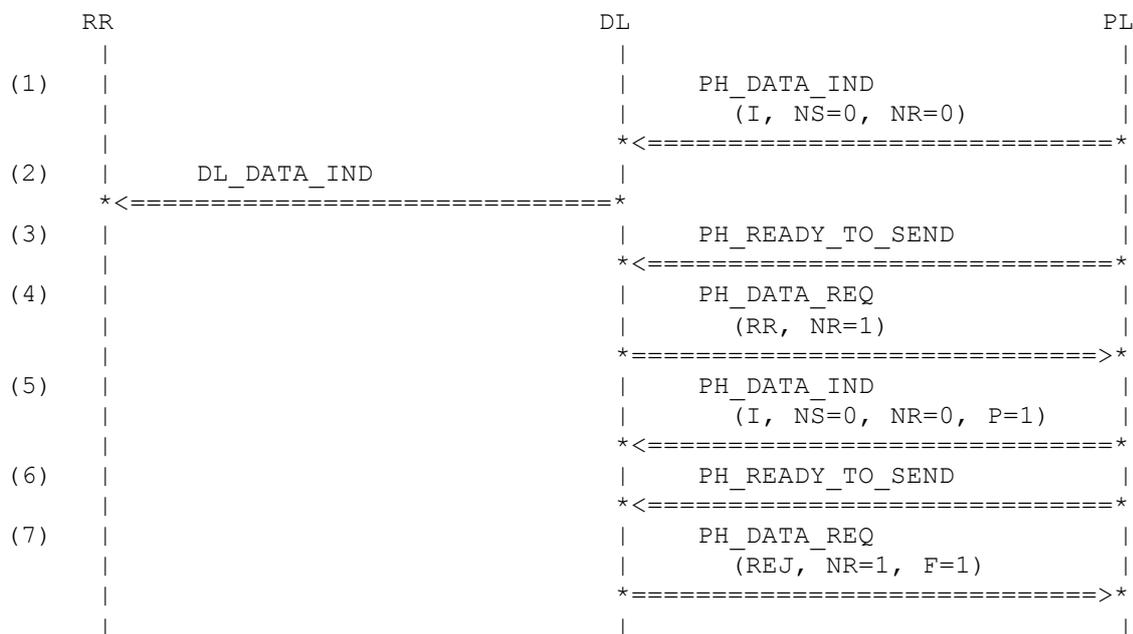
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_1
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_0_P
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_REJ_1_F

History: 05.12.97 LE Initial

4.17.43 DL627: RR response frame loss (MS to SS), (25.2.4.3), FACCH Halfrate

Description: To test the layer 2 recovery mechanism in the event of RR frame loss.
Preamble: DL601



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0_P
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_REJ_1_F

History: 08.07.98 LE Initial

4.17.44 DL228: I frame with C bit set to zero (25.2.5.1), SDCCH

Description: To test that the MS will take no action when it receives an I frame with C bit set to zero (response).
Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (I, NS=0, NR=0, C=0)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (UI)	
(4)	PH_DATA_IND (RR, P=1, C=1)	
(5)	PH_READY_TO_SEND	
(6)	PH_DATA_REQ (RR, F=1, R=1)	

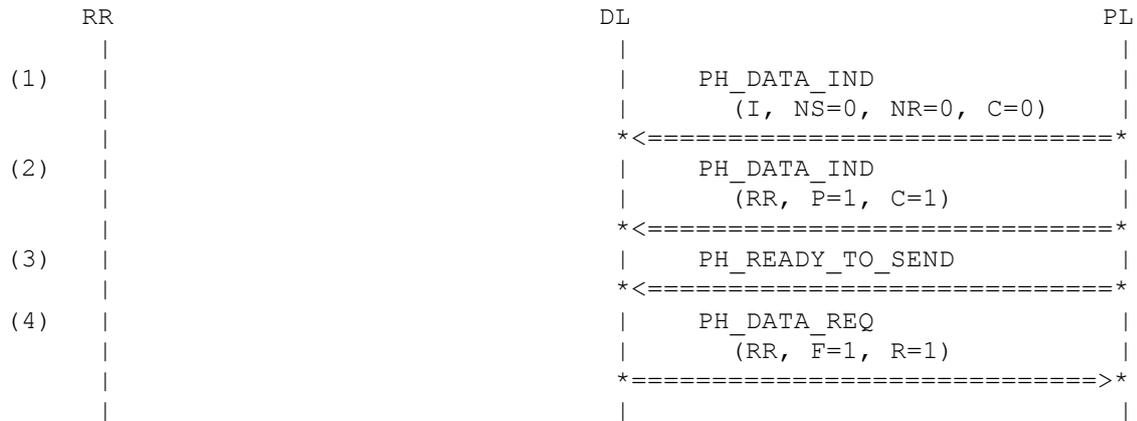
Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_0_C
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F

History: 05.12.97 LE Initial

4.17.45 DL229: I frame with C bit set to zero (25.2.5.1), FACCH Fullrate

Description: To test that the MS will take no action when it receives an I frame with C bit set to zero (response).
Preamble: DL201



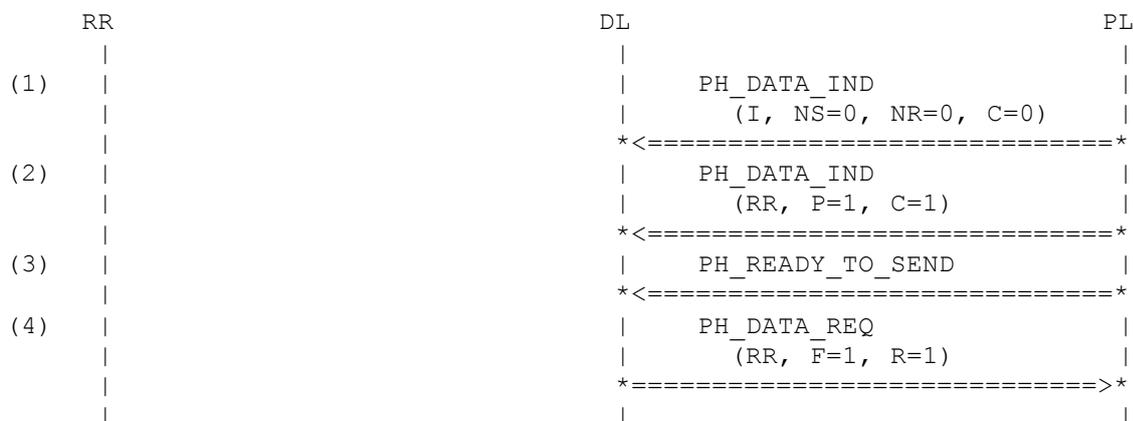
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_I_0_0_C
(2) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_RR_0_P
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_RR_0_F

History: 05.12.97 LE Initial

4.17.46 DL629: I frame with C bit set to zero (25.2.5.1), FACCH Halfrate

Description: To test that the MS will take no action when it receives an I frame with C bit set to zero (response).
Preamble: DL601



Parametrization

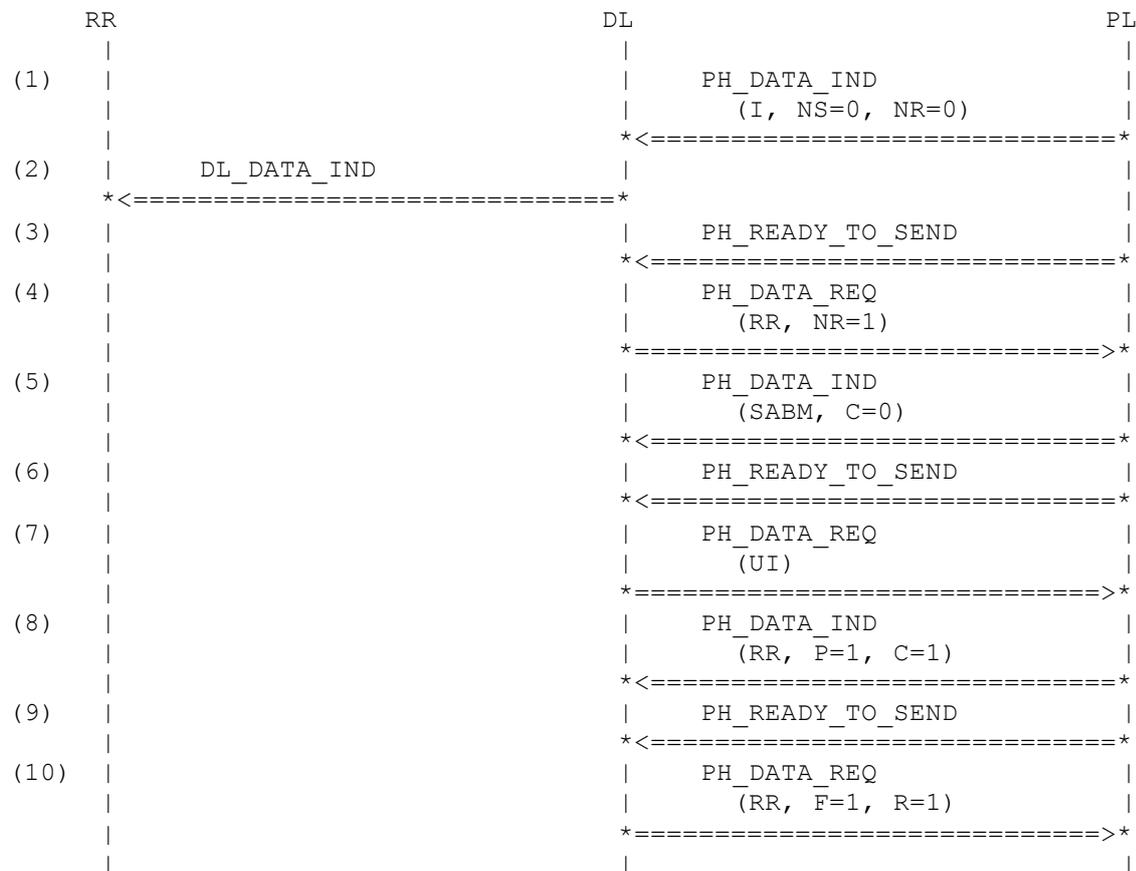
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_I_0_0_C
(2) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_RR_0_P
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_0_F

History: 08.07.98 LE Initial

4.17.47 DL230: SABM Frame with C bit set to zero (25.2.5.2), SDCCH

Description: To test that the MS will take no action when it receives an SABM frame with the C bit set to zero (response).

Preamble: DL200



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	MS_RR_1
(5) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	BS_SABM_C

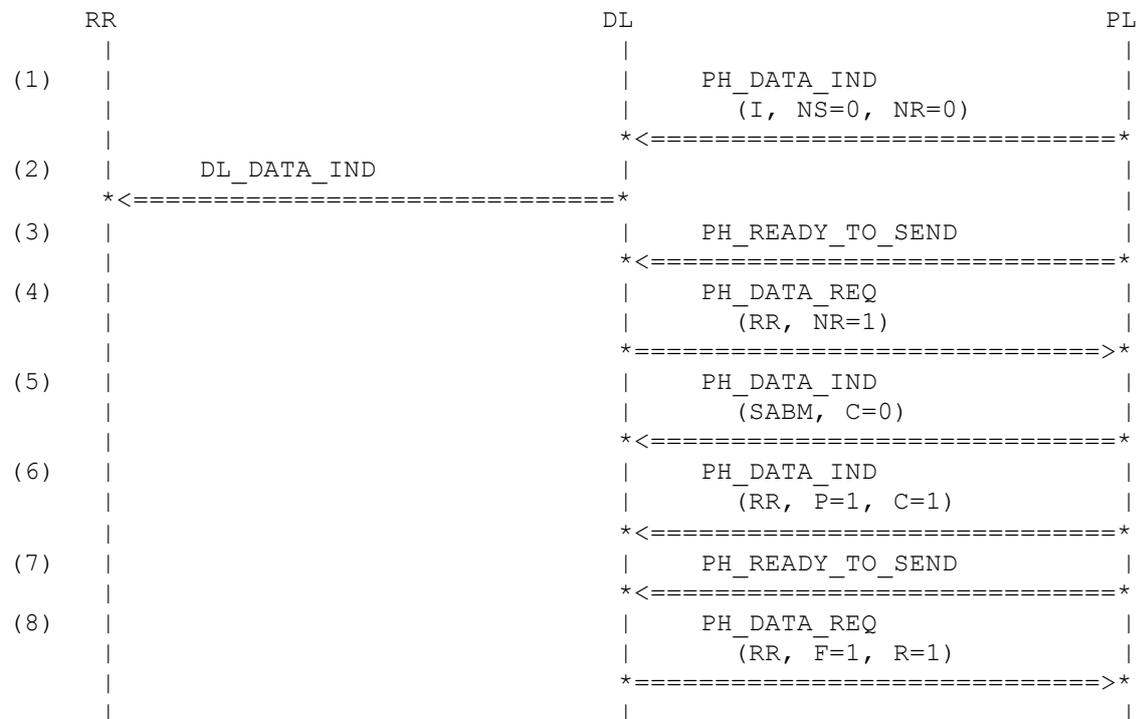
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_1_P
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1_F

History: 05.12.97 LE Initial

4.17.48 DL231: SABM Frame with C bit set to zero (25.2.5.2), FACCH Fullrate

Description: To test that the MS will take no action when it receives an SABM frame with the C bit set to zero (response).

Preamble: DL201



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	MS_RR_1
(5) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_SABM_C
(6) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_RR_1_P

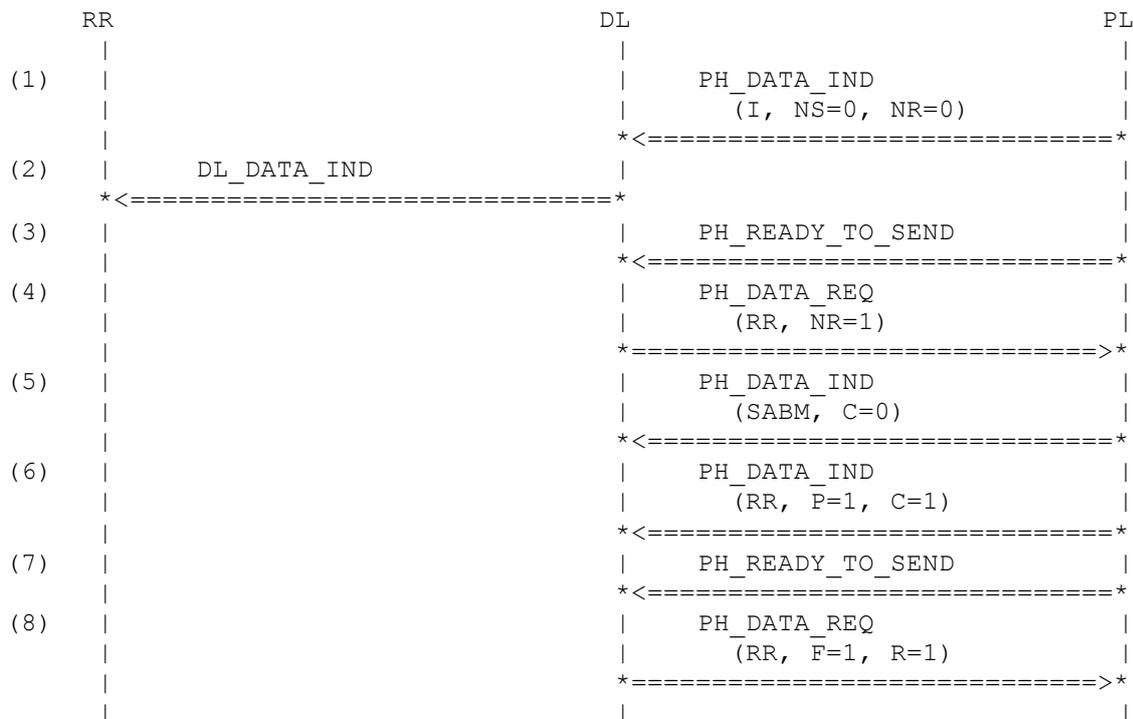
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_1_F

History: 05.12.97 LE Initial

4.17.49 DL631: SABM Frame with C bit set to zero (25.2.5.2), FACCH Halfrate

Description: To test that the MS will take no action when it receives an SABM frame with the C bit set to zero (response).

Preamble: DL601



Parametrization

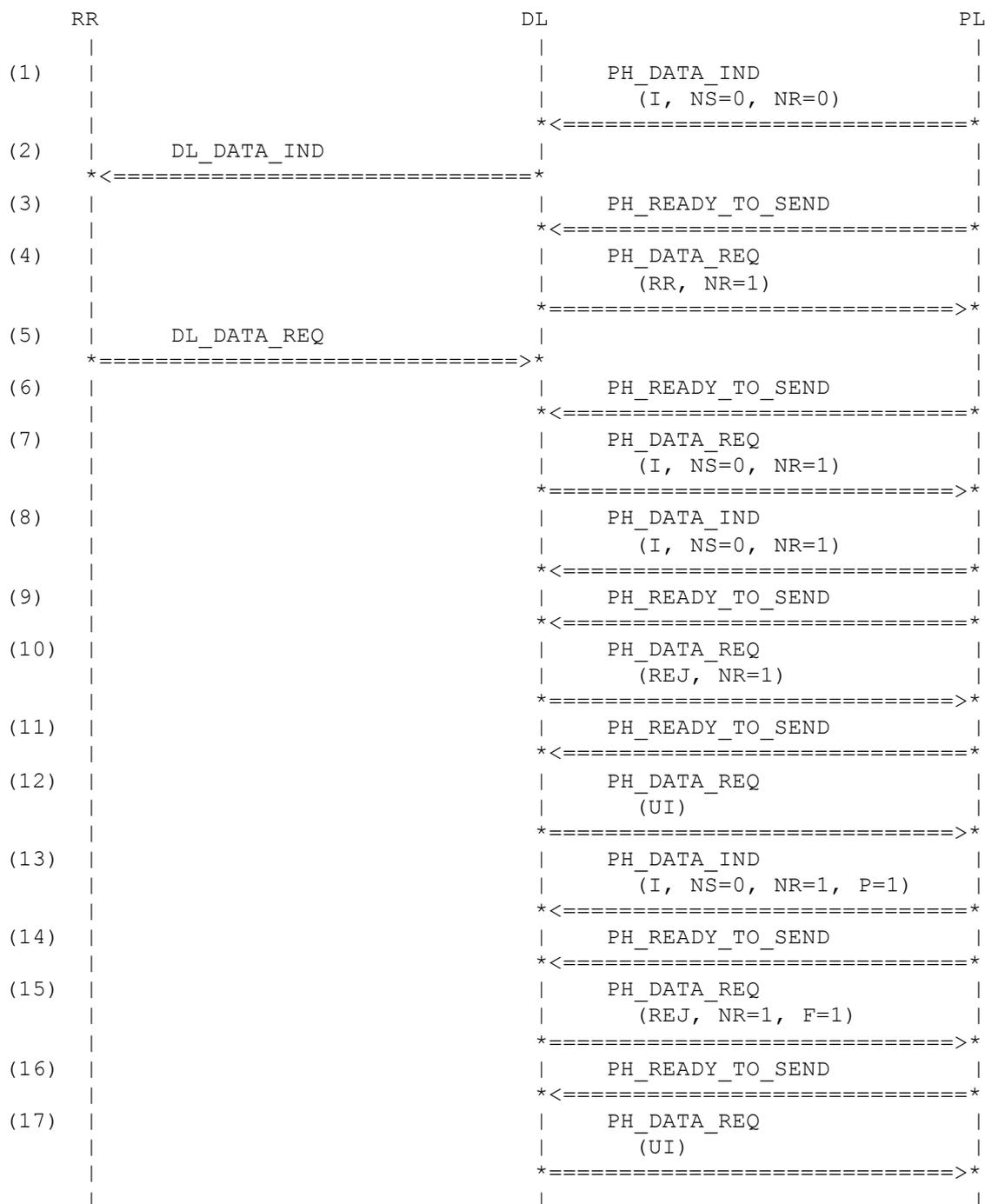
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_I_0_0
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_1

(5)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_SABM_C
(6)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_1_P
(7)	PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(8)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1_F

History: 08.07.98 LE Initial

4.17.50 DL232: N(S) Sequence Error (25.2.6.1), SDCCH

Description: To test that the MS will ignore the contents of the I field of an out-of-sequence I frame from the SS.
Preamble: DL200



Parametrization

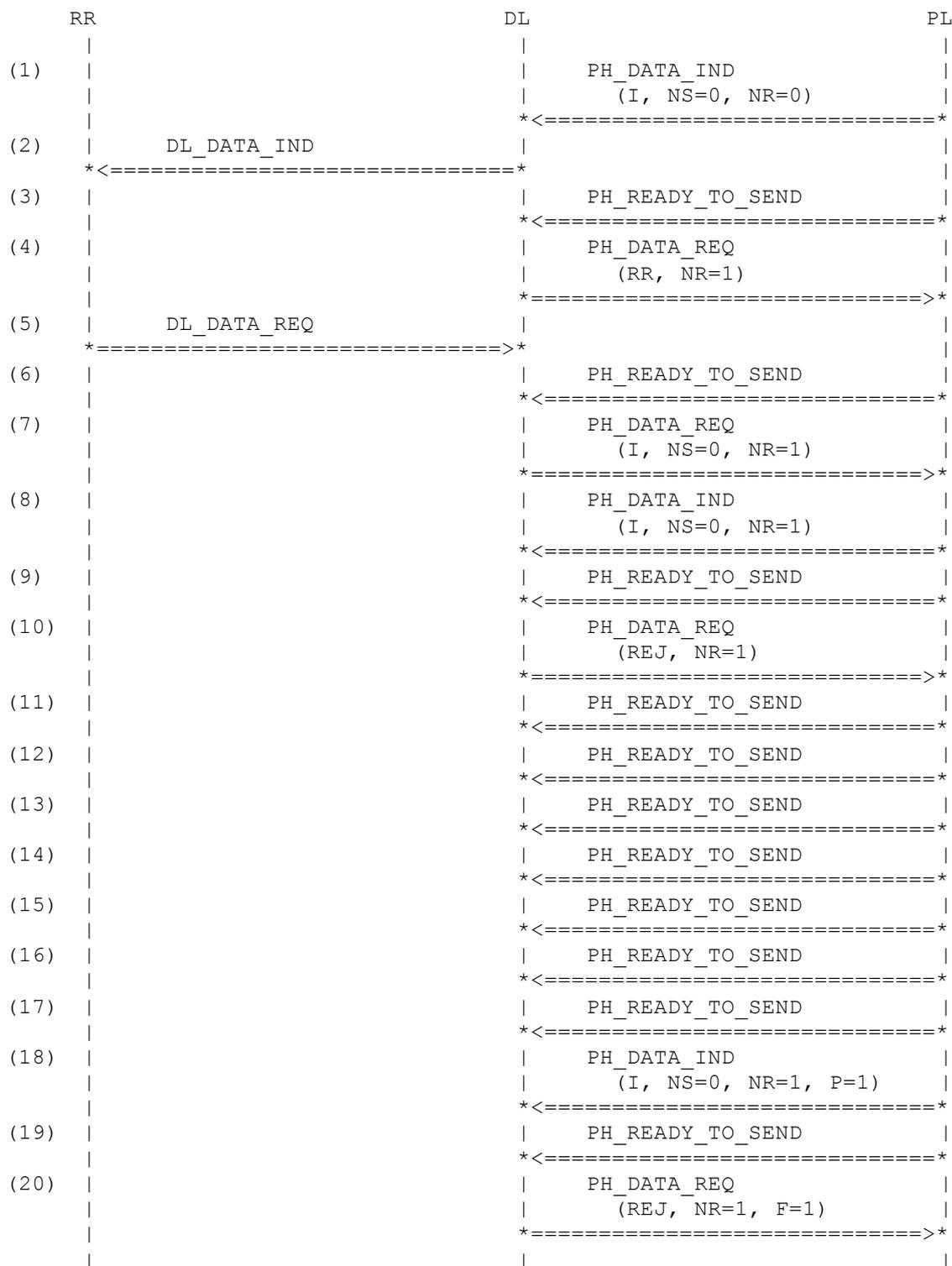
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_I_0_1
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_1
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(10) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_REJ_1
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(13) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_I_0_1_P
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED MS_REJ_1_F
(16) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(17) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

History: 05.12.97 LE Initial

4.17.51 DL233: N(S) Sequence Error (25.2.6.1), FACCH Fullrate

Description: To test that the MS will ignore the contents of the I field of an out-of-sequence I frame from the SS.
Preamble: DL201



Parametrization

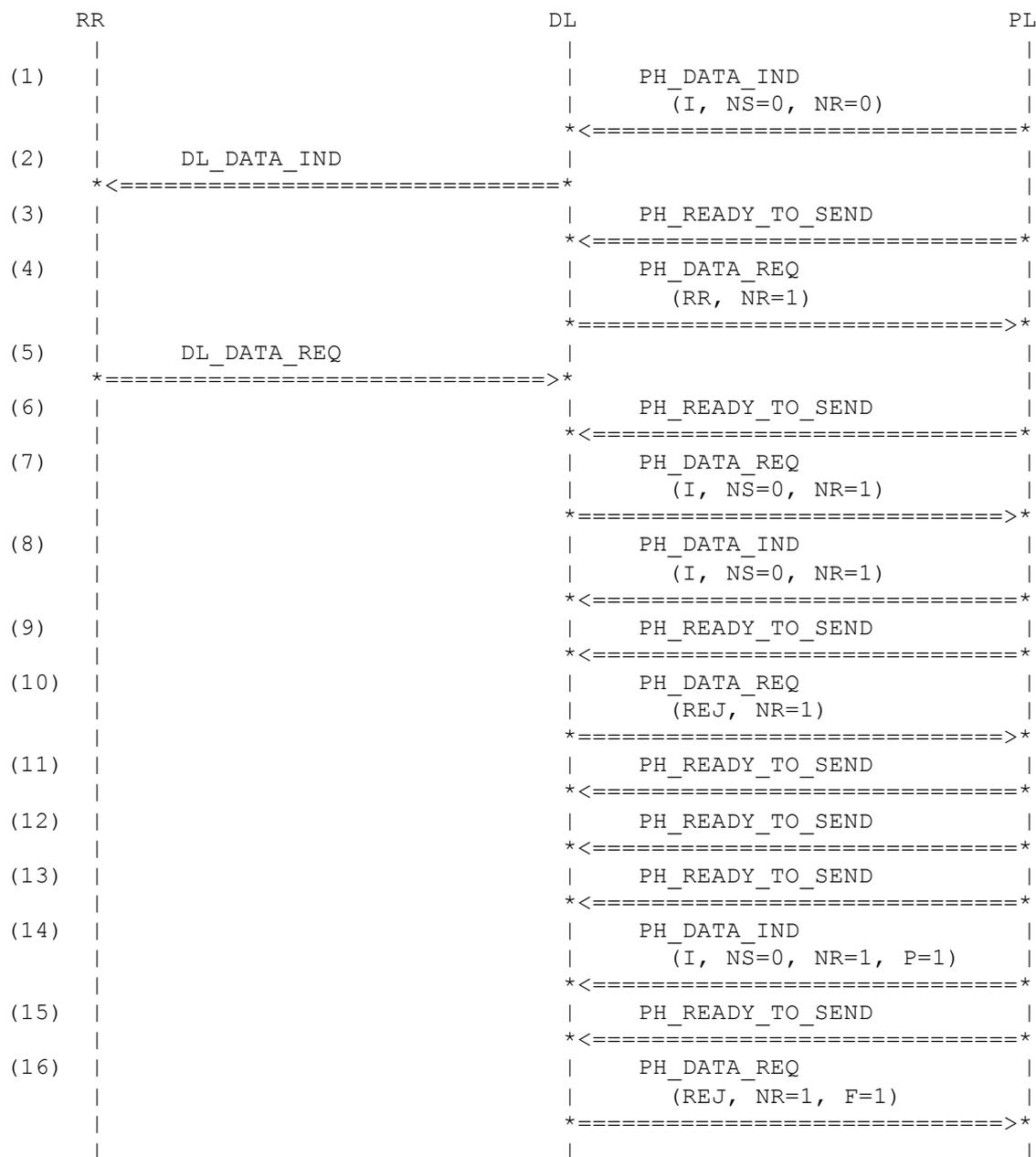
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_I_0_1
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_1
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(10)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_REJ_1
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(13)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(14)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(16)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(17)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(18)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_1_P
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_REJ_1_F

History: 05.12.97 LE Initial

4.17.52 DL633: N(S) Sequence Error (25.2.6.1), FACCH Halfrate

Description: To test that the MS will ignore the contents of the I field of an out-of-sequence I frame from the SS.
Preamble: DL201



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_0
(2) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_REQUEST
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_1
(5) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_FACCH_HR SAPI_0 ID_RESPONSE
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_I_0_1
(8) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_1
(9) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(10)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_REJ_1
(11)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(13)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(14)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_1_P
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(16)	PH_DATA_REQ ch_type	CH_TYPE_FACCH_HR

dummy
sdu

NOT_USED
MS_REJ_1_F

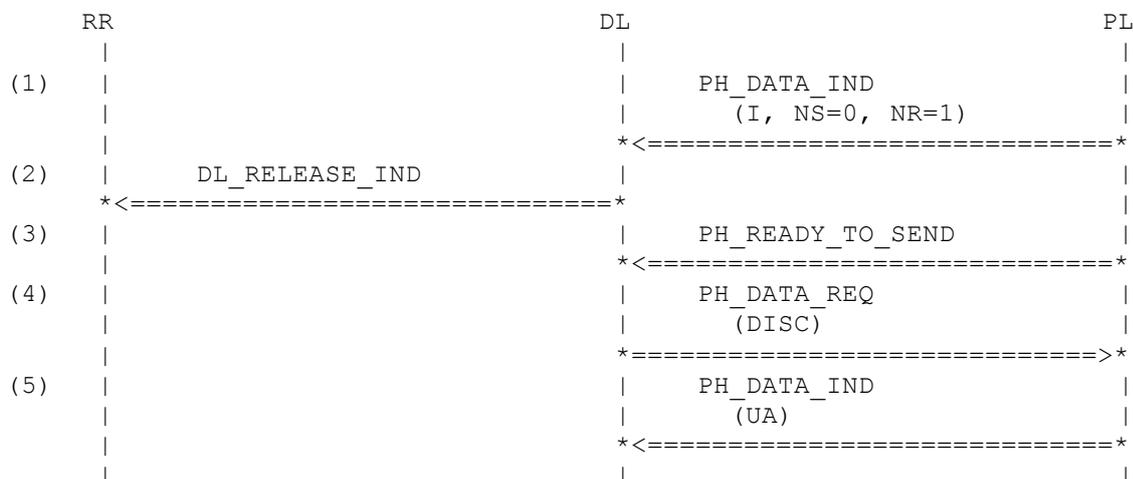
History: 08.07.98 LE Initial

4.17.53 DL234: N(R) Sequence Error (25.2.6.2), SDCCH

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it. A is the original version according to GSM 11.10. B is the Anite implementation. The P-Bit is set to zero, so the incoming message shall be discarded.

Preamble: DL200

Variants: <A>...



Parametrization

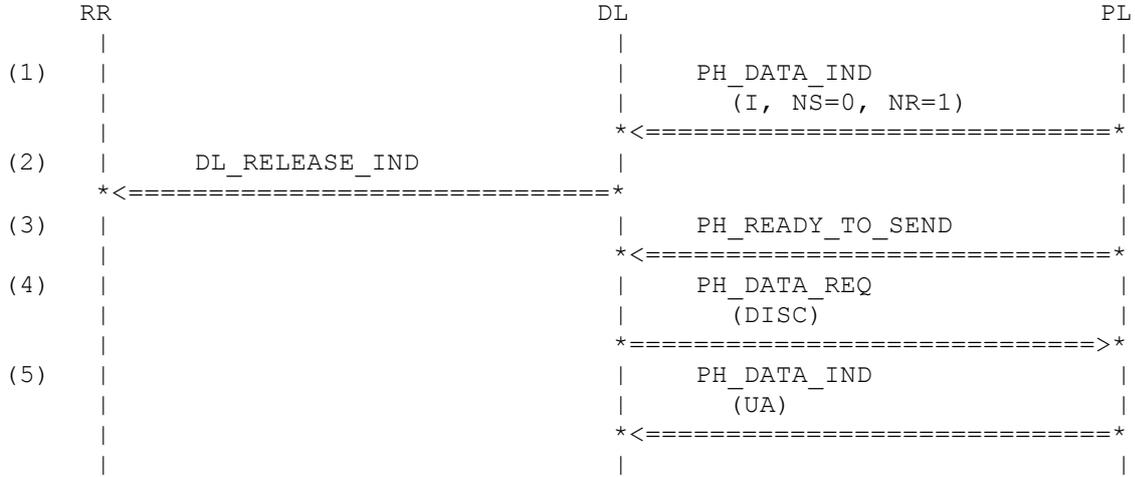
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
<A>	sdu	BS_I_0_1
	sdu	BS_I_0_7
(2) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(4) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(5) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_A_DL

History: 05.12.97 LE Initial

4.17.54 DL235: N(R) Sequence Error (25.2.6.2), FACCH Fullrate

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it. The P-Bit is set to zero, so the incoming message shall be discarded.

Preamble: DL201



Parametrization

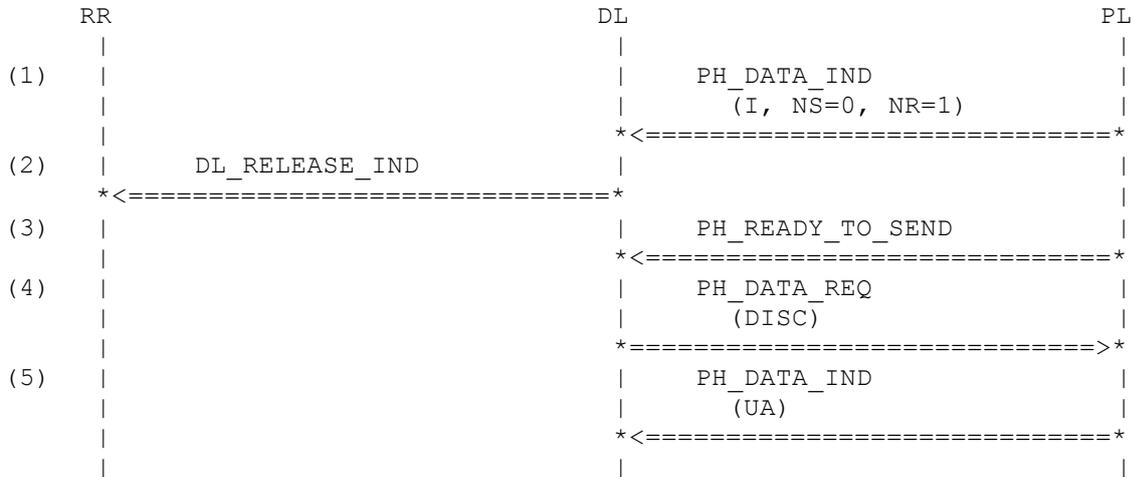
Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_I_0_1
(2) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_FACCH SAPI_0 NOT_PRESENT_8BIT
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED DISC_A_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL

History: 05.12.97 LE Initial

4.17.55 DL635: N(R) Sequence Error (25.2.6.2), FACCH Halfrate

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it. The P-Bit is set to zero, so the incoming message shall be discarded.

Preamble: DL201



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_I_0_1
(2) DL_RELEASE_IND	ch_type sapi cs	CH_TYPE_FACCH_HR SAPI_0 NOT_PRESENT_8BIT
(3) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH_HR
(4) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED DISC_A_UL
(5) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED UA_A_DL

History: 08.07.98 LE Initial

4.17.56 DL236: Test on receipt of invalid frames (25.2.7), SDCCH

Description: To test that the MS will ignore invalid frames.
Preamble: DL200

RR	DL	PL
(1)	PH_DATA_IND (RR, NR=1)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (UI)	
(4)	PH_DATA_IND (RR, NR=0, P=1)	
(5)	PH_READY_TO_SEND	
(6)	PH_DATA_REQ (RR, NR=0, F=1)	
(7)	PH_DATA_IND (REJ, NR=1)	
(8)	PH_READY_TO_SEND	
(9)	PH_DATA_REQ (UI)	
(10)	PH_DATA_IND (RR, NR=0, P=1)	
(11)	PH_READY_TO_SEND	
(12)	PH_DATA_REQ (RR, NR=0, F=1)	
(13)	PH_DATA_IND (SABM, EL=0)	
(14)	PH_READY_TO_SEND	
(15)	PH_DATA_REQ (UI)	
(16)	PH_DATA_IND (RR, NR=0, P=1)	
(17)	PH_READY_TO_SEND	
(18)	PH_DATA_REQ (RR, NR=0, F=1)	
(19)	PH_DATA_IND (DM, L>0)	
(20)	PH_READY_TO_SEND	

```

(21) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(22) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(23) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(24) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(25) | | PH_DATA_IND | |
      | | (DISC, M=1) | |
      | | *<=====* | |
(26) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(27) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(28) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(29) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(30) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(31) | | PH_DATA_IND | |
      | | (UA, EA=0) | |
      | | *<=====* | |
(32) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(33) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(34) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(35) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(36) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(37) | | PH_DATA_IND | |
      | | (I, M=0, L>N201) | |
      | | *<=====* | |
(38) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(39) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(40) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(41) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(42) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(43) | | PH_DATA_IND | |
    
```


Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_1
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(7) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_2
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(10) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P

(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(13) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_3
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(16) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(17) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(18) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(19) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_4
(20) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(21) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK

(22) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(23) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(24) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(25) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_5
(26) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(27) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(28) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(29) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(30) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(31) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_6
(32) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(33) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(34) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(35) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(36) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F

(37) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_7
(38) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(39) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(40) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(41) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(42) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(43) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_8
(44) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(45) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(46) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(47) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(48) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F

History: 05.12.97 LE Initial

4.17.57 DL237: Test on receipt of invalid frames II(25.2.7), SDCCH

Description: To test that the MS will ignore invalid frames.
Preamble: DL236

RR	DL	PL
(1)	PH_DATA_IND (xxx1 1101)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (UI)	
(4)	PH_DATA_IND (RR, NR=0, P=1)	
(5)	PH_READY_TO_SEND	
(6)	PH_DATA_REQ (RR, NR=0, F=1)	
(7)	PH_DATA_IND (xxx1 1011)	
(8)	PH_READY_TO_SEND	
(9)	PH_DATA_REQ (UI)	
(10)	PH_DATA_IND (RR, NR=0, P=1)	
(11)	PH_READY_TO_SEND	
(12)	PH_DATA_REQ (RR, NR=0, F=1)	
(13)	PH_DATA_IND (xxx1 0111)	
(14)	PH_READY_TO_SEND	
(15)	PH_DATA_REQ (UI)	
(16)	PH_DATA_IND (RR, NR=0, P=1)	
(17)	PH_READY_TO_SEND	
(18)	PH_DATA_REQ (RR, NR=0, F=1)	
(19)	PH_DATA_IND (01x1 1111)	
(20)	PH_READY_TO_SEND	

```

(21) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(22) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(23) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(24) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(25) | | PH_DATA_IND | |
      | | (1x1 1111) | |
      | | *<=====* | |
(26) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(27) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(28) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(29) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(30) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(31) | | PH_DATA_IND | |
      | | (0011 0011) | |
      | | *<=====* | |
(32) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(33) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(34) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(35) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(36) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
(37) | | PH_DATA_IND | |
      | | (1xx1 0011) | |
      | | *<=====* | |
(38) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(39) | | PH_DATA_REQ | |
      | | (UI) | |
      | | *=====>* | |
(40) | | PH_DATA_IND | |
      | | (RR, NR=0, P=1) | |
      | | *<=====* | |
(41) | | PH_READY_TO_SEND | |
      | | *<=====* | |
(42) | | PH_DATA_REQ | |
      | | (RR, NR=0, F=1) | |
      | | *=====>* | |
    
```

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_9
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(7) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_10
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(10) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(13) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_11
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED UI_FRAME_UPLINK
(16) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(17) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(18) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(19) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_12
(20) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(21) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(22) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(23) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(24) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(25) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_13
(26) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(27) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(28) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(29) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(30) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED MS_RR_0_F
(31) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_14
(32) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(33) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(34) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(35) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(36) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F
(37) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED INV_15
(38) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(39) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED UI_FRAME_UPLINK
(40) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED BS_RR_0_P
(41) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(42) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED MS_RR_0_F

History: 05.12.97 LE Initial

4.17.58 DL238: Test on receipt of invalid frames (25.2.7), FACCH Fullrate

Description: To test that the MS will ignore invalid frames.
Preamble: DL201

RR	DL	PL
(1)	PH_DATA_IND (RR, NR=1)	
	<=====	
TIMEOUT (3000)		
(2)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(5)	PH_DATA_IND (REJ, NR=1)	
	<=====	
TIMEOUT (3000)		
(6)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(9)	PH_DATA_IND (SABM, EL=0)	
	<=====	
TIMEOUT (3000)		
(10)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(11)	PH_READY_TO_SEND	
	<=====	
(12)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(13)	PH_DATA_IND (DM, L>0)	
	<=====	
TIMEOUT (3000)		
(14)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(15)	PH_READY_TO_SEND	
	<=====	
(16)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(17)	PH_DATA_IND (DISC, M=1)	
	<=====	
TIMEOUT (3000)		
(18)	PH_DATA_IND	

```

|                                     | (RR, NR=0, P=1) |
| *<=====|
(19) | PH_READY_TO_SEND |
| *<=====|
(20) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(21) | PH_DATA_IND |
| (UA, EA=0) |
| *<=====|
TIMEOUT (3000)
(22) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(23) | PH_READY_TO_SEND |
| *<=====|
(24) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(25) | PH_DATA_IND |
| (I, M=0, L>N201) |
| *<=====|
TIMEOUT (3000)
(26) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(27) | PH_READY_TO_SEND |
| *<=====|
(28) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(29) | PH_DATA_IND |
| (I, M=1, L<N201) |
| *<=====|
TIMEOUT (3000)
(30) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(31) | PH_READY_TO_SEND |
| *<=====|
(32) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(33) | PH_DATA_IND |
| (xxx1 1101) |
| *<=====|
TIMEOUT (3000)
(34) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(35) | PH_READY_TO_SEND |
| *<=====|
(36) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(37) | PH_DATA_IND |
| (xxx1 1011) |
| *<=====|
TIMEOUT (3000)
(38) | PH_DATA_IND |

```

```

|                                     | (RR, NR=0, P=1) |
| *<=====|
(39) | PH_READY_TO_SEND |
| *<=====|
(40) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(41) | PH_DATA_IND |
| (xxx1 0111) |
| *<=====|
TIMEOUT (3000)
(42) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(43) | PH_READY_TO_SEND |
| *<=====|
(44) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(45) | PH_DATA_IND |
| (01x1 1111) |
| *<=====|
TIMEOUT (3000)
(46) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(47) | PH_READY_TO_SEND |
| *<=====|
(48) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(49) | PH_DATA_IND |
| (1x1 1111) |
| *<=====|
TIMEOUT (3000)
(50) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(51) | PH_READY_TO_SEND |
| *<=====|
(52) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(53) | PH_DATA_IND |
| (0011 0011) |
| *<=====|
TIMEOUT (3000)
(54) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(55) | PH_READY_TO_SEND |
| *<=====|
(56) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(57) | PH_DATA_IND |
| (1xx1 0011) |
| *<=====|
TIMEOUT (3000)
(58) | PH_DATA_IND |

```


(13)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_4
(14)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(17)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_5
(18)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(21)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_6
(22)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(24)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(25)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_7
(26)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P

(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(28)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(29)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_8
(30)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(32)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(33)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_9
(34)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(36)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(37)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_10
(38)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(41)	PH_DATA_IND ch_type	CH_TYPE_FACCH

	dummy sdu	NOT_USED INV_11
(42)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(43)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(44)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(45)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_12
(46)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(47)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(48)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(49)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_13
(50)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(51)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(52)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(53)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_14
(54)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(55)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH

(56)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F
(57)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED INV_15
(58)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH NOT_USED BS_RR_0_P
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH
(60)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH NOT_USED MS_RR_0_F

History: 05.12.97 LE Initial

4.17.59 DL638: Test on receipt of invalid frames (25.2.7), FACCH Halfrate

Description: To test that the MS will ignore invalid frames.
Preamble: DL601

RR	DL	PL
(1)	PH_DATA_IND (RR, NR=1)	
	<=====	
TIMEOUT (3000)		
(2)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(3)	PH_READY_TO_SEND	
	<=====	
(4)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(5)	PH_DATA_IND (REJ, NR=1)	
	<=====	
TIMEOUT (3000)		
(6)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(7)	PH_READY_TO_SEND	
	<=====	
(8)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(9)	PH_DATA_IND (SABM, EL=0)	
	<=====	
TIMEOUT (3000)		
(10)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(11)	PH_READY_TO_SEND	
	<=====	
(12)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(13)	PH_DATA_IND (DM, L>0)	
	<=====	
TIMEOUT (3000)		
(14)	PH_DATA_IND (RR, NR=0, P=1)	
	<=====	
(15)	PH_READY_TO_SEND	
	<=====	
(16)	PH_DATA_REQ (RR, NR=0, F=1)	
	=====>	
(17)	PH_DATA_IND (DISC, M=1)	
	<=====	
TIMEOUT (3000)		
(18)	PH_DATA_IND	

```

|                                     | (RR, NR=0, P=1) |
| *<=====|
(19) | PH_READY_TO_SEND |
| *<=====|
(20) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(21) | PH_DATA_IND |
| (UA, EA=0) |
| *<=====|
TIMEOUT (3000)
(22) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(23) | PH_READY_TO_SEND |
| *<=====|
(24) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(25) | PH_DATA_IND |
| (I, M=0, L>N201) |
| *<=====|
TIMEOUT (3000)
(26) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(27) | PH_READY_TO_SEND |
| *<=====|
(28) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(29) | PH_DATA_IND |
| (I, M=1, L<N201) |
| *<=====|
TIMEOUT (3000)
(30) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(31) | PH_READY_TO_SEND |
| *<=====|
(32) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(33) | PH_DATA_IND |
| (xxx1 1101) |
| *<=====|
TIMEOUT (3000)
(34) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(35) | PH_READY_TO_SEND |
| *<=====|
(36) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(37) | PH_DATA_IND |
| (xxx1 1011) |
| *<=====|
TIMEOUT (3000)
(38) | PH_DATA_IND |

```

```

|                                     | (RR, NR=0, P=1) |
| *<=====|
(39) | PH_READY_TO_SEND |
| *<=====|
(40) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(41) | PH_DATA_IND |
| (xxx1 0111) |
| *<=====|
TIMEOUT (3000)
(42) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(43) | PH_READY_TO_SEND |
| *<=====|
(44) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(45) | PH_DATA_IND |
| (01x1 1111) |
| *<=====|
TIMEOUT (3000)
(46) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(47) | PH_READY_TO_SEND |
| *<=====|
(48) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(49) | PH_DATA_IND |
| (1x1 1111) |
| *<=====|
TIMEOUT (3000)
(50) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(51) | PH_READY_TO_SEND |
| *<=====|
(52) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(53) | PH_DATA_IND |
| (0011 0011) |
| *<=====|
TIMEOUT (3000)
(54) | PH_DATA_IND |
| (RR, NR=0, P=1) |
| *<=====|
(55) | PH_READY_TO_SEND |
| *<=====|
(56) | PH_DATA_REQ |
| (RR, NR=0, F=1) |
| *=====>|
(57) | PH_DATA_IND |
| (1xx1 0011) |
| *<=====|
TIMEOUT (3000)
(58) | PH_DATA_IND |

```


(13)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_4
(14)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(17)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_5
(18)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(19)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(20)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(21)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_6
(22)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(23)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(24)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(25)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_7
(26)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P

(27)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(28)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(29)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_8
(30)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(31)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(32)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(33)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_9
(34)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(35)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(36)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(37)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_10
(38)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(39)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(40)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(41)	PH_DATA_IND ch_type	CH_TYPE_FACCH_HR

	dummy	NOT_USED
	sdu	INV_11
(42)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_RR_0_P
(43)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(44)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_0_F
(45)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	INV_12
(46)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_RR_0_P
(47)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(48)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_0_F
(49)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	INV_13
(50)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_RR_0_P
(51)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR
(52)	PH_DATA_REQ	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	MS_RR_0_F
(53)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	INV_14
(54)	PH_DATA_IND	
	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_RR_0_P
(55)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_FACCH_HR

(56)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F
(57)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED INV_15
(58)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED BS_RR_0_P
(59)	PH_READY_TO_SEND ch_type	CH_TYPE_FACCH_HR
(60)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_FACCH_HR NOT_USED MS_RR_0_F

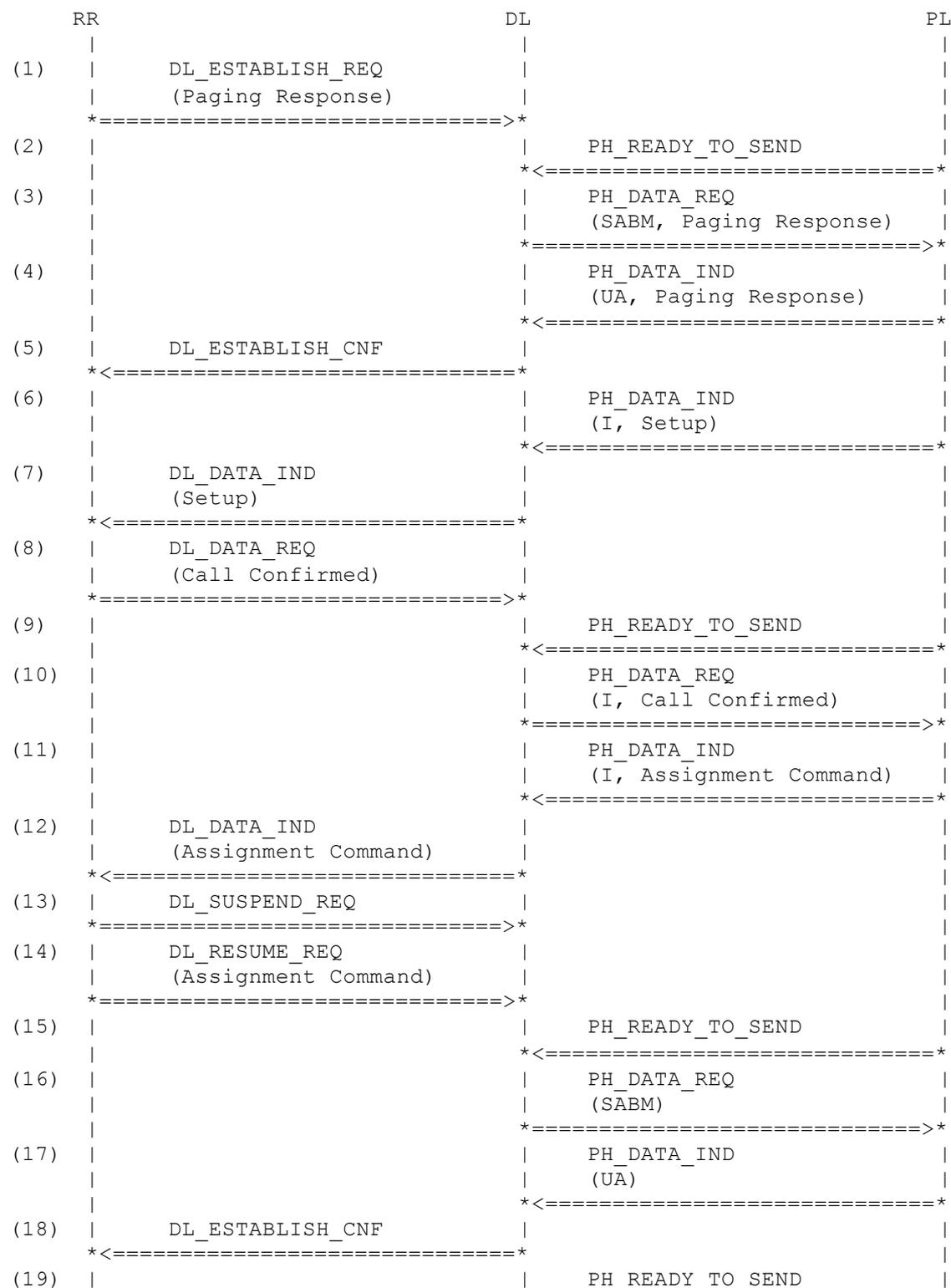
History: 08.07.98 LE Initial

4.18 Additional Multilayer Tests

4.18.1 DL300: Mobile Terminated Call

Description: To test a typical mobile terminated call situation. The connection is established as fast as possible. Confirmations of I frames is done with I frames.

Preamble: DL000



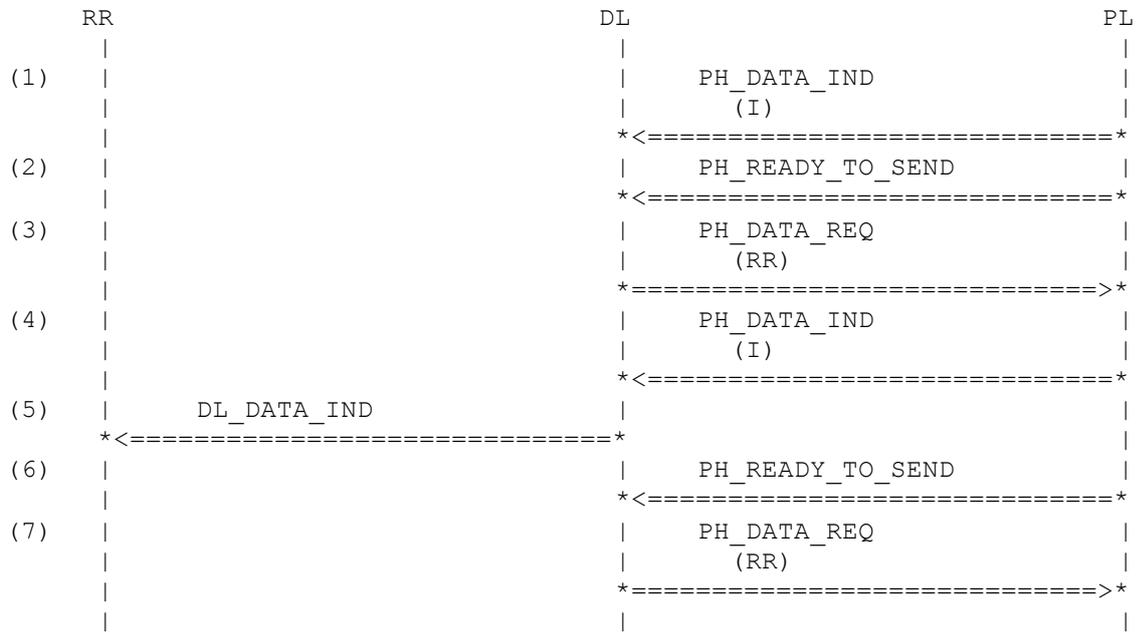
(12) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_ASSIGN_CMD
(13) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(14) DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_ASSIGN_CMP
(15) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(16) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_A_UL
(17) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_A_DL
(18) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(19) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(20) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_ASSIGN_CMP
(21) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_ASSIGN_CMP

History: 05.12.97 LE Initial

4.18.2 DL301: Mobile Terminated SMS (SDCCH)

Description: To test a typical mobile terminated SMS situation. The connection is established and the short message receives in more than one segments.

Preamble: DL087



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I20_DL3
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_UL3
(4) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_I04_NS1_DL3
(5) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_3 L3_MSG_L24_DL
(6) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(7) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_A_NR2_UL3

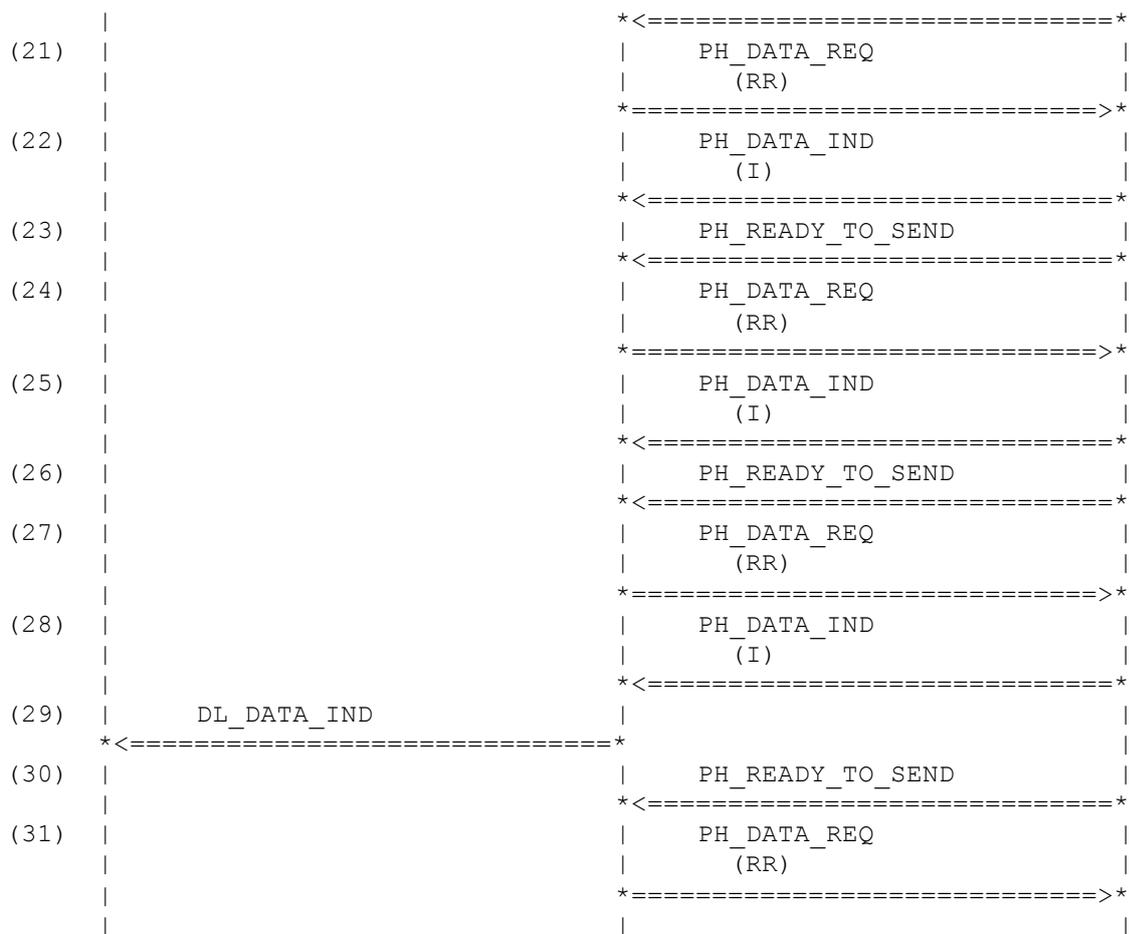
History: 05.12.97 LE Initial

4.18.3 DL302: Mobile Terminated SMS (SDCCH), very long message

Description: To test a typical mobile terminated SMS situation. The connection is established and the short message receives in more than one segments.

Preamble: DL087

RR	DL	PL
(1)	PH_DATA_IND (UI)	
(2)	PH_READY_TO_SEND	
(3)	PH_DATA_REQ (UI)	
(4)	PH_DATA_IND (I)	
(5)	PH_READY_TO_SEND	
(6)	PH_DATA_REQ (RR)	
(7)	PH_DATA_IND (I)	
(8)	PH_READY_TO_SEND	
(9)	PH_DATA_REQ (RR)	
(10)	PH_DATA_IND (I)	
(11)	PH_READY_TO_SEND	
(12)	PH_DATA_REQ (RR)	
(13)	PH_DATA_IND (I)	
(14)	PH_READY_TO_SEND	
(15)	PH_DATA_REQ (RR)	
(16)	PH_DATA_IND (I)	
(17)	PH_READY_TO_SEND	
(18)	PH_DATA_REQ (RR)	
(19)	PH_DATA_IND (I)	
(20)	PH_READY_TO_SEND	



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_DOWNLINK
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UI_FRAME_UPLINK
(4) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	100
(5) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(6) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	RR01
(7) PH_DATA_IND	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED I01
(8) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(9) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR02
(10) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I02
(11) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(12) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR03
(13) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I03
(14) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(15) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR04
(16) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I04
(17) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(18) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR05
(19) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I05
(20) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(21) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR06
(22) PH_DATA_IND	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED I06
(23) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(24) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR07
(25) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I07
(26) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(27) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR00
(28) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I00_FINAL
(29) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_3 SMS_MSG
(30) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(31) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR01

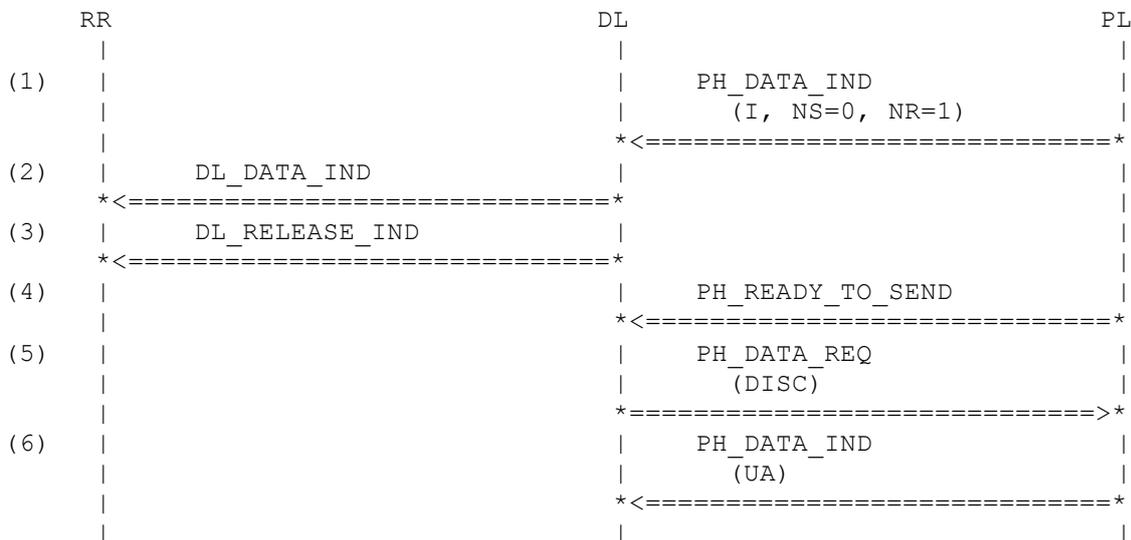
History: 05.12.97 LE Initial

4.18.4 DL303: N(R) Sequence Error (25.2.6.2), SDCCH

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it. A is the original version according GSM 11.10. B is the Anite implementation for N(R). The P-Bit is set, so the incoming message must be forwarded.

Preamble: DL200

Variants: <A>...



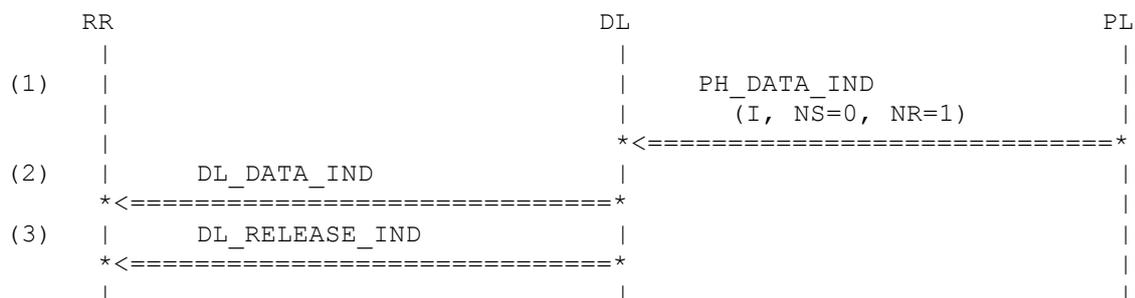
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
<A>	sdu	BS_I_0_1_P1
	sdu	BS_I_0_7_P1
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) DL_RELEASE_IND	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT
(4) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(5) PH_DATA_REQ	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DISC_A_UL
(6) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_A_DL

History: 05.12.97 LE Initial

4.18.5 DL304: N(R) Sequence Error (25.2.6.2), FACCH Fullrate

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it
Preamble: DL201



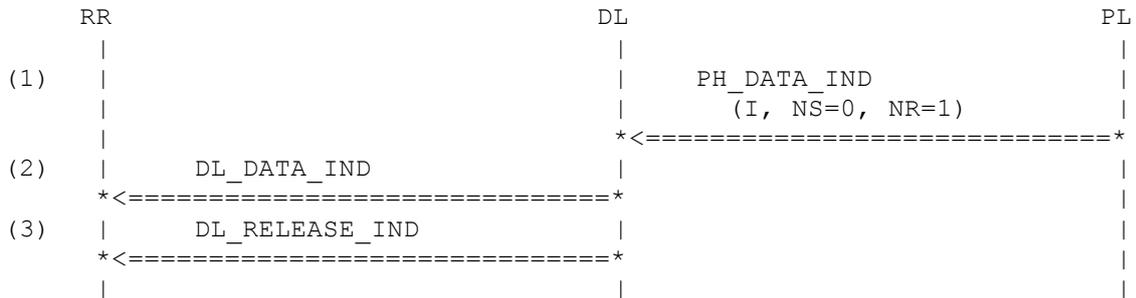
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH
	dummy	NOT_USED
	sdu	BS_I_0_1_P1
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 05.12.97 LE Initial

4.18.6 DL305: N(R) Sequence Error (25.2.6.2), FACCH Halfrate

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it
Preamble: DL201



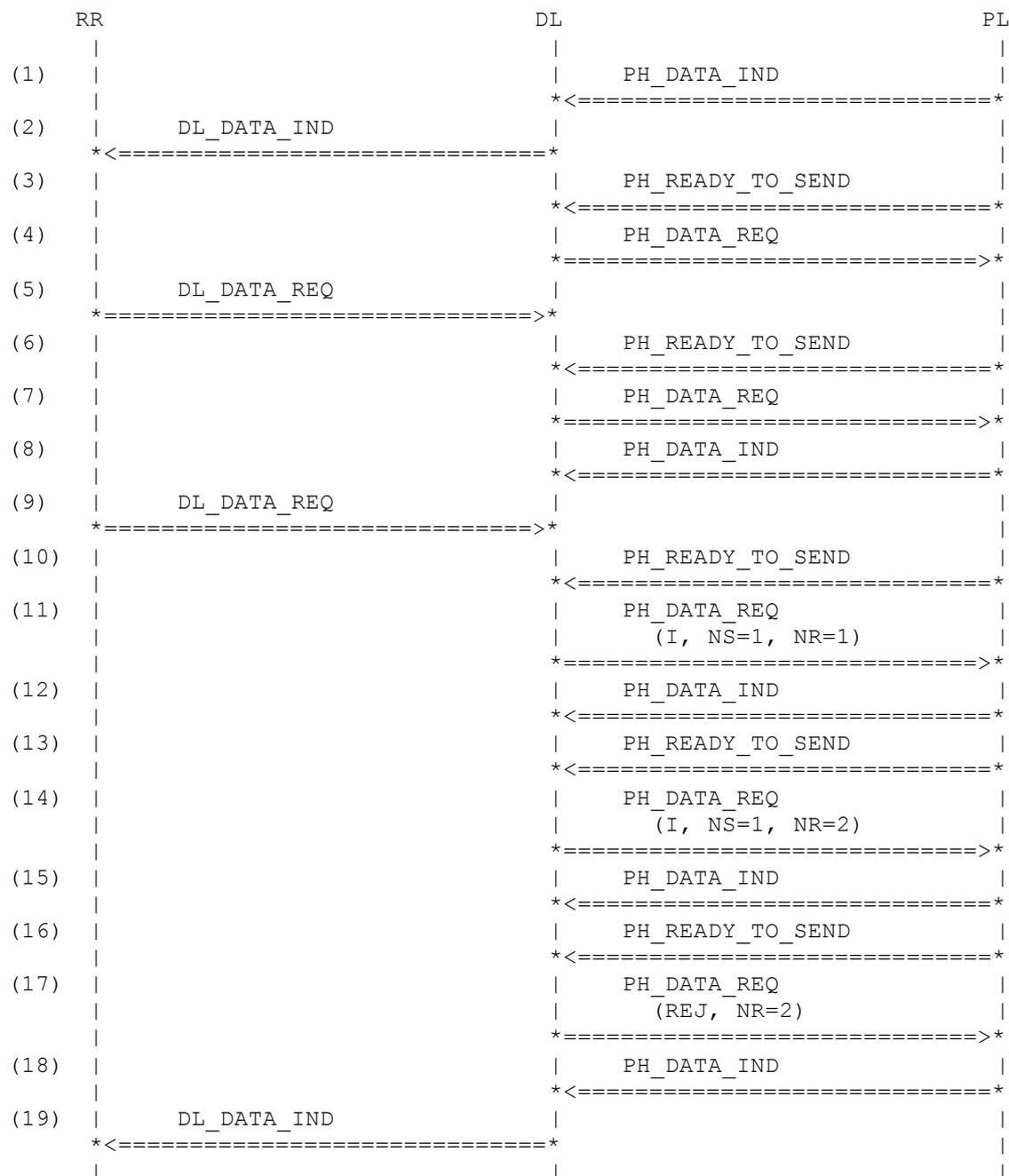
Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	dummy	NOT_USED
	sdu	BS_I_0_1_P1
(2) DL_DATA_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	sdu	ID_REQUEST
(3) DL_RELEASE_IND	ch_type	CH_TYPE_FACCH_HR
	sapi	SAPI_0
	cs	NOT_PRESENT_8BIT

History: 08.07.98 LE Initial

4.18.7 DL306: Collision Problem Establishment and incoming Segments

Description: To test that the MS will detect a N(R) sequence error and react in the proper way to it
Preamble: DL012



Parametrization

Primitive	Parameter	Value
(1) PH_DATA_IND	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	I_CIPH_REQ
(2) DL_DATA_IND	ch_type	CH_TYPE_SDCCH

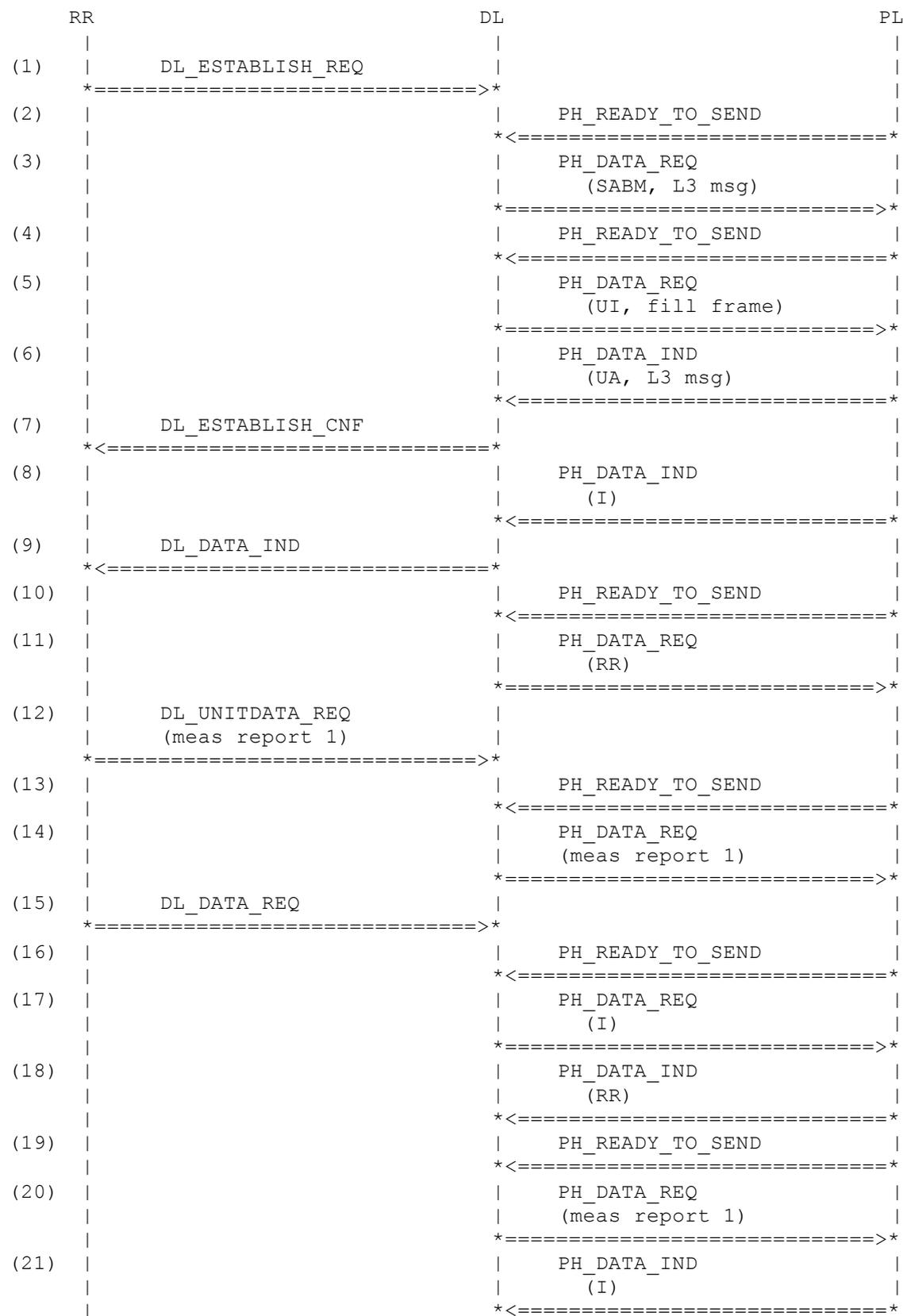
	sapi sdu	SAPI_0 L3_CIPH_REQ
(3)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(4)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_CIPH_REQ
(5)	DL_DATA_REQ	
	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_CIPH_RSP
(6)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(7)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_CIPH_RSP
(8)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_CIPH_RSP
(9)	DL_DATA_REQ	
	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_SETUP_MSG
(10)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(11)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP_MSG_P0
(12)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_HO_CMD_P0_S1
(13)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(14)	PH_DATA_REQ	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_SETUP_MSG_P1
(15)	PH_DATA_IND	
	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_HO_CMD_P1_S1
(16)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(17)	PH_DATA_REQ	
	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED REJ_HO_CMD
(18)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_HO_CMD_P0_S2
(19)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_HO_CMD

History: 08.07.98 LE Initial

4.18.8 DL307: SMS-Testcase 34.2.1 on the Anite (Part 1)

Description: The testcase shall reproduce some strange behaviour during testcase 34.2.1 on the Anite test system.
Preamble: DL000



```

(22) | DL_DATA_IND |
      | *<===== |
(23) | DL_DATA_REQ |
      | *=====> * |
(24) | | PH_READY_TO_SEND |
      | *<===== |
(25) | | PH_DATA_REQ |
      | | (I) |
      | *=====> * |
(26) | | PH_DATA_IND |
      | | (RR) |
      | *<===== |
(27) | | PH_READY_TO_SEND |
      | *<===== |
(28) | | PH_DATA_REQ |
      | | (meas report 1) |
      | *=====> * |
(29) | | PH_DATA_IND |
      | | (I) |
      | *<===== |
(30) | DL_DATA_IND |
      | *<===== |
(31) | | PH_READY_TO_SEND |
      | *<===== |
(32) | | PH_DATA_REQ |
      | | (RR) |
      | *=====> * |
(33) | DL_DATA_REQ |
      | *=====> * |
(34) | | PH_READY_TO_SEND |
      | *<===== |
(35) | | PH_DATA_REQ |
      | | (I) |
      | *=====> * |
(36) | | PH_READY_TO_SEND |
      | *<===== |
(37) | | PH_DATA_REQ |
      | | (meas report 1) |
      | *=====> * |
(38) | | PH_DATA_IND |
      | | (RR) |
      | *<===== |
(39) | DL_DATA_REQ |
      | *=====> * |
(40) | | PH_READY_TO_SEND |
      | *<===== |
(41) | | PH_DATA_REQ |
      | | (I) |
      | *=====> * |
(42) | | PH_DATA_IND |
      | | (RR) |
      | *<===== |
(43) | | PH_READY_TO_SEND |
      | *<===== |
(44) | | PH_DATA_REQ |
      | | (meas report 1) |
      | *=====> * |
(45) | | PH_READY_TO_SEND |
      | *<===== |
(46) | | PH_DATA_REQ |
    
```

```

(47) |                                     | (meas report 1) |
      |                                     | *=====>*
      | PH_READY_TO_SEND                 |
      | *<=====*
(48) | PH_DATA_REQ                           |
      | (meas report 1)                   |
      | *=====>*
(49) | PH_READY_TO_SEND                       |
      | *<=====*
(50) | PH_DATA_REQ                           |
      | (meas report 1)                   |
      | *=====>*
(51) | PH_READY_TO_SEND                       |
      | *<=====*
(52) | PH_DATA_REQ                           |
      | (meas report 1)                   |
      | *=====>*
(53) | DL_DATA_REQ                           |
      | *=====>*
(54) | PH_READY_TO_SEND                       |
      | *<=====*
(55) | PH_DATA_REQ                           |
      | (I)                                 |
      | *=====>*
(56) | PH_DATA_IND                           |
      | (RR)                                 |
      | *<=====*
(57) | PH_READY_TO_SEND                       |
      | *<=====*
(58) | PH_DATA_REQ                           |
      | (meas report 1)                   |
      | *=====>*
(59) | PH_DATA_IND                           |
      | (I)                                 |
      | *<=====*
(60) | DL_DATA_IND                           |
      | *<=====*
(61) | DL_SUSPEND_REQ                       |
      | *=====>*
(62) | DL_RESUME_REQ                         |
      | *=====>*
(63) | PH_READY_TO_SEND                       |
      | *<=====*
(64) | PH_DATA_REQ                           |
      | (SABM)                             |
      | *=====>*
(65) | PH_DATA_IND                           |
      | (UA)                                 |
      | *<=====*
(66) | DL_ESTABLISH_CNF                     |
      | *<=====*
(67) | PH_READY_TO_SEND                       |
      | *<=====*
(68) | PH_DATA_REQ                           |
      | (I)                                 |
      | *=====>*
(69) | PH_DATA_IND                           |
      | (RR)                                 |
      | *<=====*
(70) | PH_DATA_IND                           |
    
```


	dummy sdu	NOT_USED RR_MSG_2
(12) DL_UNITDATA_REQ	ch_type sapi sdu	CH_TYPE_SACCH SAPI_0 MEAS_REPORT_1
(13) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(15) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_3
(16) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(17) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_3
(18) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_3
(19) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(20) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(21) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_4
(22) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_4
(23) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_5
(24) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(25) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5
(26) PH_DATA_IND	ch_type	CH_TYPE_SDCCH

	dummy sdu	NOT_USED RR_MSG_5
(27) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(28) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(29) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_6
(30) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_6
(31) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(32) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_6
(33) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_7
(34) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(35) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_7
(36) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(37) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(38) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_7
(39) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_8
(40) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(41) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_8

(42) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_8
(43) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(44) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(45) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(46) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(47) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(48) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(49) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(50) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(51) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(52) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(53) DL_DATA_REQ	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_9
(54) PH_READY_TO_SEND	ch_type	CH_TYPE_SDCCH
(55) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_9
(56) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_9
(57) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH

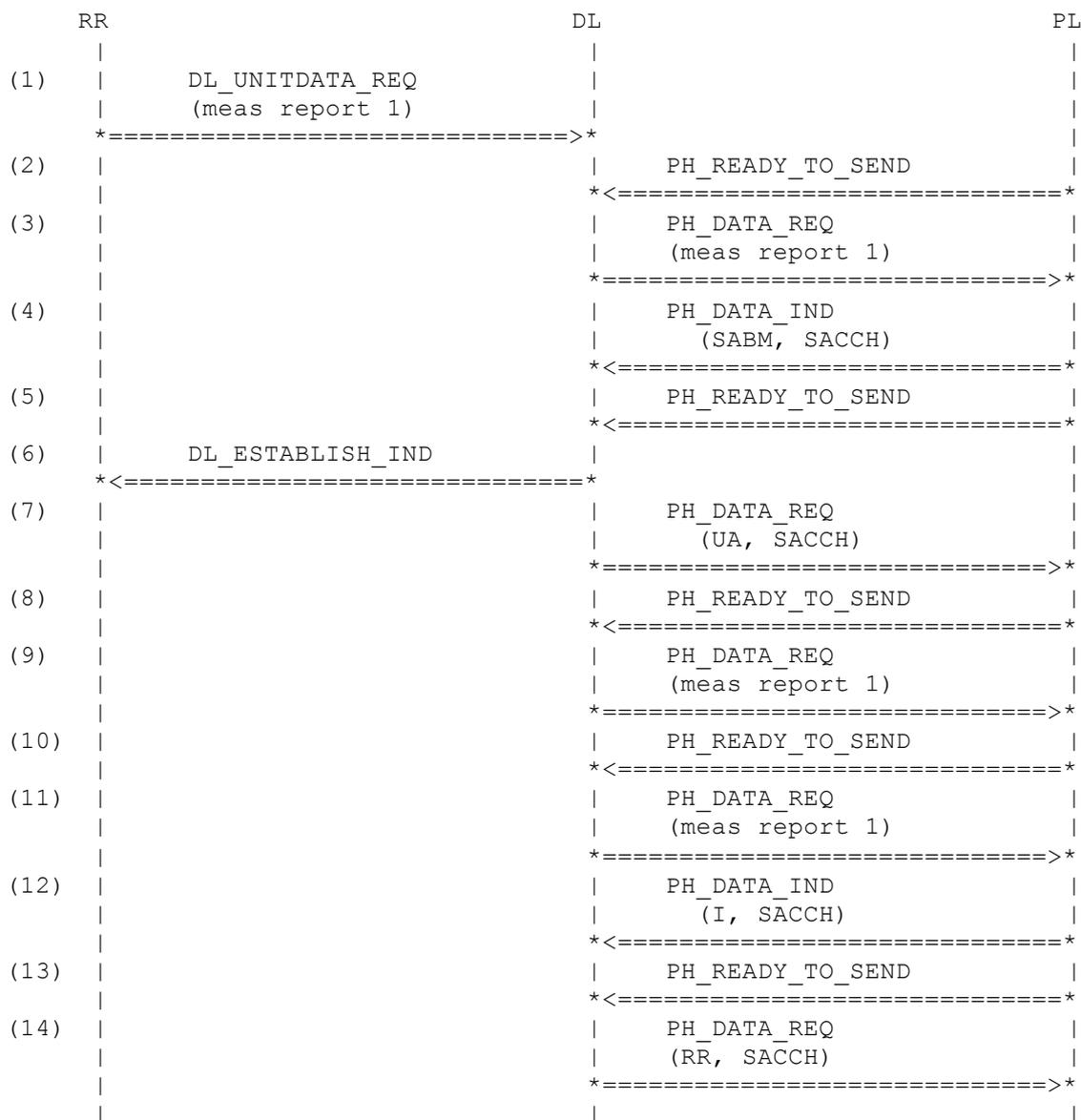
(58) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(59) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_10
(60) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_10
(61) DL_SUSPEND_REQ	ch_type sapi	CH_TYPE_SDCCH SAPI_0
(62) DL_RESUME_REQ	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_MSG_11
(63) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(64) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED SABM_MSG_11
(65) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED UA_MSG_11
(66) DL_ESTABLISH_CNF	ch_type sapi	CH_TYPE_FACCH SAPI_0
(67) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH
(68) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_MSG_11
(69) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_MSG_11
(70) PH_DATA_IND	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED I_MSG_12
(71) DL_DATA_IND	ch_type sapi sdu	CH_TYPE_FACCH SAPI_0 L3_MSG_12
(72) PH_READY_TO_SEND	ch_type	CH_TYPE_FACCH

(73) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_FACCH NOT_USED RR_MSG_12
(74) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(75) PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_0_L1

History: 08.07.98 LE Initial

4.18.9 DL308: SMS-Testcase 34.2.1 on the Anite (Part 2)

Description: The testcase shall reproduce some strange behaviour during testcase 34.2.1 on the Anite test system.
Preamble: DL307



Parametrization

Primitive	Parameter	Value
(1) DL_UNITDATA_REQ	ch_type	CH_TYPE_SACCH
	sapi	SAPI_0
	sdu	MEAS_REPORT_1
(2) PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(3) PH_DATA_REQ	ch_type	CH_TYPE_SACCH
	dummy	NOT_USED
	sdu	MEAS_REPORT_1_L1

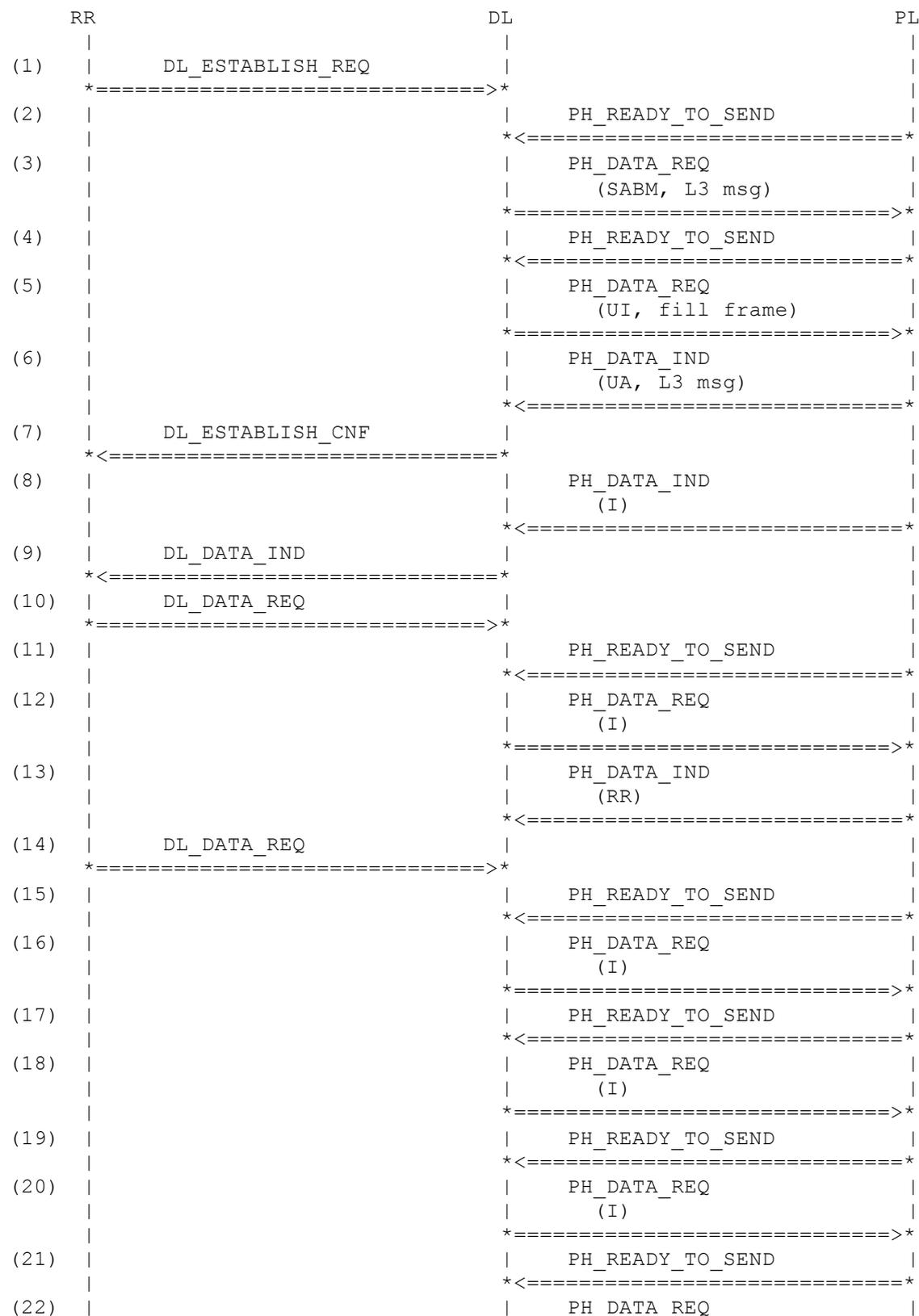
(4)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED SABM_MSG_13
(5)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(6)	DL_ESTABLISH_IND	ch_type sapi sdu	CH_TYPE_SACCH SAPI_3 L3_MSG_L0_UL
(7)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED UA_MSG_13
(8)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(9)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(10)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(11)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED MEAS_REPORT_1_L1
(12)	PH_DATA_IND	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED I_MSG_14
(13)	PH_READY_TO_SEND	ch_type	CH_TYPE_SACCH
(14)	PH_DATA_REQ	ch_type dummy sdu	CH_TYPE_SACCH NOT_USED RR_MSG_14

History: 08.07.98 LE Initial

4.18.10 DL309: Handover Testcase on the Anite

Description: The testcase shall reproduce some strange behaviour during testcase 26.6.5.2.3 on the Anite test system for a customer mobile (behaviour on SDCCH/8).

Preamble: DL000



```

|                                     | (I) |
|                                     |=====>*
(23) | PH_READY_TO_SEND |
|                                     |=====*
(24) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(25) | PH_READY_TO_SEND |
|                                     |=====*
(26) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(27) | PH_READY_TO_SEND |
|                                     |=====*
(28) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(29) | PH_READY_TO_SEND |
|                                     |=====*
(30) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(31) | PH_DATA_IND |
|                                     | (I, 1. segment) |
|                                     |=====*
(32) | PH_READY_TO_SEND |
|                                     |=====*
(33) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(34) | PH_DATA_IND |
|                                     | (I, 2. segment) |
|                                     |=====*
(35) | DL_DATA_IND |
|=====*
(36) | DL_SUSPEND_REQ |
|=====>*
(37) | DL_RESUME_REQ |
|=====>*
(38) | PH_READY_TO_SEND |
|=====*
(39) | PH_DATA_REQ |
|                                     | (SABM) |
|                                     |=====>*
(40) | PH_DATA_IND |
|                                     | (UA) |
|                                     |=====*
(41) | DL_ESTABLISH_CNF |
|=====*
(42) | PH_READY_TO_SEND |
|=====*
(43) | PH_DATA_REQ |
|                                     | (I) |
|                                     |=====>*
(44) | PH_DATA_IND |
|                                     | (RR) |
|                                     |=====*
(45) | PH_READY_TO_SEND |
|=====*
(46) | PH_DATA_REQ |
|

```


sapi	SAPI_0
sdu	L3_MSG_2
DL_DATA_REQ	
ch_type	CH_TYPE_SDCCH
sapi	SAPI_0
sdu	L3_MSG_3
PH_READY_TO_SEND	
ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	I_MSG_3B
PH_DATA_IND	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	RR_MSG_3
DL_DATA_REQ	
ch_type	CH_TYPE_SDCCH
sapi	SAPI_0
sdu	L3_MSG_5
PH_READY_TO_SEND	
ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	I_MSG_5A
PH_READY_TO_SEND	
ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	I_MSG_5B
PH_READY_TO_SEND	
ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	I_MSG_5B
PH_READY_TO_SEND	
ch_type	CH_TYPE_SDCCH
PH_DATA_REQ	
ch_type	CH_TYPE_SDCCH
dummy	NOT_USED
sdu	I_MSG_5B

PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5B
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5B
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5B
PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_HO_1
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5C
PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_HO_2
DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_HO
DL_SUSPEND_REQ ch_type sapi	CH_TYPE_SDCCH SAPI_0
DL_RESUME_REQ ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_11
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED SABM_MSG_11
PH_DATA_IND ch_type	CH_TYPE_SDCCH

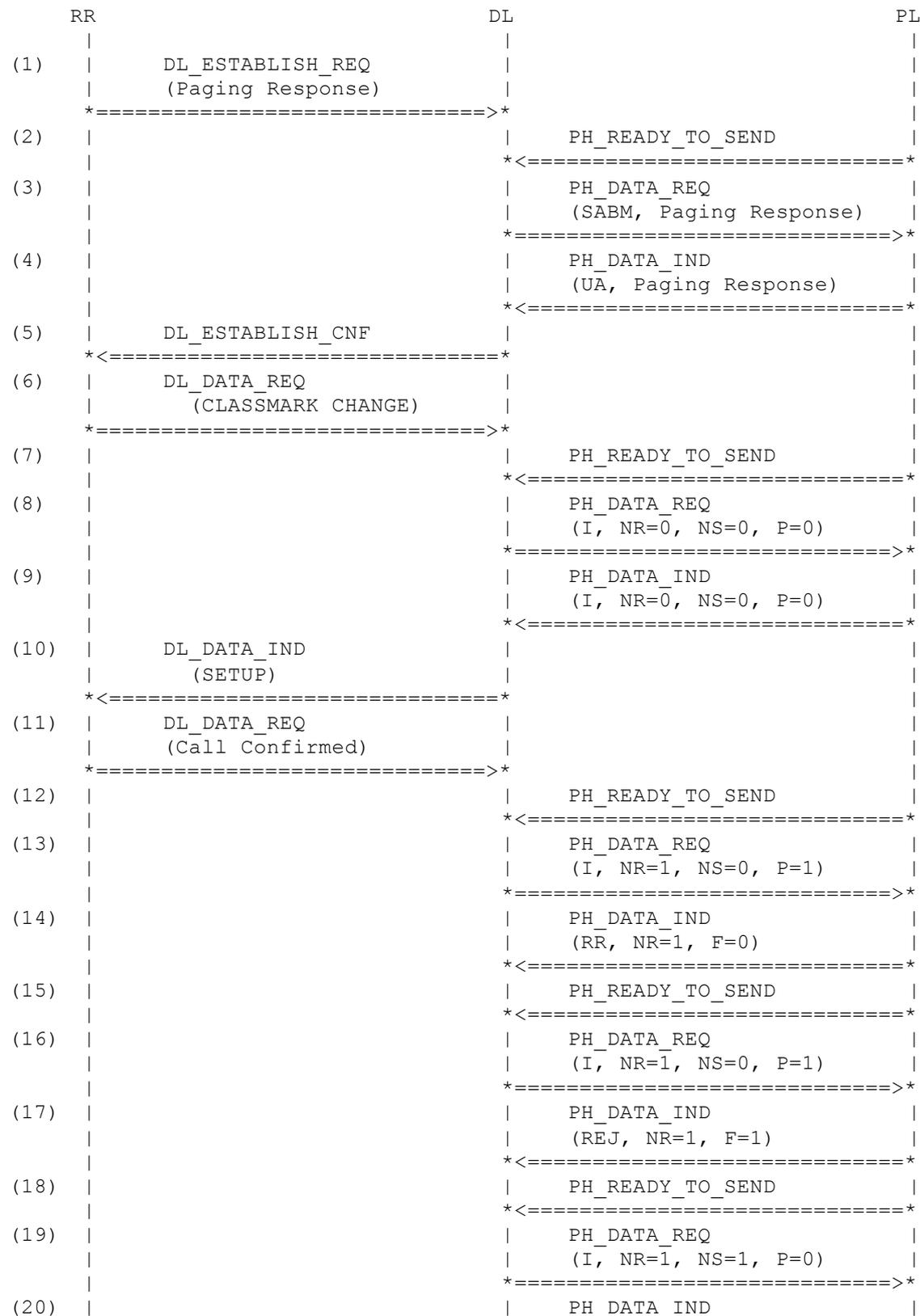
dummy sdu	NOT_USED UA_MSG_11
DL_ESTABLISH_CNF ch_type sapi	CH_TYPE_SDCCH SAPI_0
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_11
PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_11
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_5D
PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_5D
PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED I_MSG_12B
DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 L3_MSG_12
DL_RELEASE_REQ ch_type sapi	CH_TYPE_SDCCH SAPI_0
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED RR_MSG_12B
PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DISC_A_UL

History: 08.07.98 LE Initial

4.18.11 DL311: Mobile Terminated Call (slow network)

Description: The testcase shall reproduce some strange behaviour on Mobilstar, Belgium with timer recovery condition clearing during a mobile terminated call situation.

Preamble: DL000



```

|                                     | (RR, NR=1, P=1) |
| *<=====|
(21) | PH_READY_TO_SEND |
| *<=====|
(22) | PH_DATA_REQ |
| (I, NR=1, NS=1, P=1) |
| *=====>|
(23) | PH_DATA_IND |
| (I, NR=2, NS=1, P=0) |
| *<=====|
(24) | PH_READY_TO_SEND |
| *<=====|
(25) | PH_DATA_REQ |
| (I, NR=2, NS=1, P=1) |
| *=====>|
(26) | PH_DATA_IND |
| (REJ, NR=2, P=1) |
| *<=====|
|                                     |
    
```

Parametrization

Primitive	Parameter	Value
(1)	DL_ESTABLISH_REQ	
	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_PAGING_RESP
(2)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(3)	PH_DATA_REQ	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	SABM_PAGING_RESP
(4)	PH_DATA_IND	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	UA_PAGING_RESP
(5)	DL_ESTABLISH_CNF	
	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
(6)	DL_DATA_REQ	
	ch_type	CH_TYPE_SDCCH
	sapi	SAPI_0
	sdu	L3_MSG_CLASSM_CHG
(7)	PH_READY_TO_SEND	
	ch_type	CH_TYPE_SDCCH
(8)	PH_DATA_REQ	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DL311_I_CMD_0_0_UL
(9)	PH_DATA_IND	
	ch_type	CH_TYPE_SDCCH
	dummy	NOT_USED
	sdu	DL311_I_CMD_0_0_DL

(10)	DL_DATA_IND ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 DL311_L3_SETUP
(11)	DL_DATA_REQ ch_type sapi sdu	CH_TYPE_SDCCH SAPI_0 DL311_L3_CALL_CONF
(12)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(13)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_1_0_P_UL
(14)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_RR_RSP_1_DL
(15)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(16)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_1_0_P_UL
(17)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_REJ_RSP_1_F_DL
(18)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(19)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_1_1_UL
(20)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_RR_RSP_1_F_DL
(21)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH
(22)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_1_1_P_UL
(23)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_2_1_DL
(24)	PH_READY_TO_SEND ch_type	CH_TYPE_SDCCH

(25)	PH_DATA_REQ ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_I_CMD_2_1_P_UL
(26)	PH_DATA_IND ch_type dummy sdu	CH_TYPE_SDCCH NOT_USED DL311_REJ_RSP_2_F_DL

History: 29.01.01 msb Initial

Appendices

A. Acronyms

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

B. Glossary

International Mobile Telecommunication 2000 (IMT-2000/ITU-2000) Formerly referred to as FPLMTS (Future Public Land-Mobile Telephone System), this is the ITU's specification/family of standards for 3G. This initiative provides a global infrastructure through both satellite and terrestrial systems, for fixed and mobile phone users. The family of standards is a framework comprising a mix/blend of systems providing global roaming. <URL: <http://www.imt-2000.org/>>