

	rw /home/tetra/henry/working_3/	
	AA_Common_Types	rw common/Common_Types.sun
	BA_Layer2_Service_Primitives	rw 12/L2servp.sun
	CA_Layer2_Private	rw 12/layer2_private.sun
	L2_MS	rw 12/l2_ms.sun
	L2_MS	rw 12/l2_ms.sbt
	MobileStation	rw 12/mobilestation.sbk
	MS_DL	rw 12/ms_receive.spr
	Bad_Presiding_Block_Gen_MS_DL	rw 12/bad_presiding_block_
	Build_Broadcast1Primitive	rw 12/build_broadcast1primitiv
	Build_Broadcast2Primitive	rw 12/build_broadcast2primitiv
	SegMgmt_Clear_Blocks_Received_UR	rw 12/segmgmt_clear_blc
	SegMgmt_Get_Blocks_Expected_DD2	rw 12/set_blocks_expecte
	SDUMgmt_Clear_Blocks_Received_DL	rw 12/sdumgmt_clear_blc
	Send_DL_SDU_To_L3	rw 12/send_dl_sdu_to_l3.spd
	SDUMgmt_New	rw 12/SDUMgmt_New.spd
	Build_UR	rw 12/build_ur.spd
	MS_UL	rw 12/ms_ul.spr
	Bad_Data_Block_Gen_UL	rw 12/bad_data_block_gen_ul.spd
	Bad_Presiding_Block_Gen_MS_UL	rw 12/bad_presiding_block_
	Send_UL_Confirm_To_L3	rw 12/send_ul_confirm_to_l3_2.spd
	Push_SDU_UL	rw 12/push_sdu_ul.spd
	Build_UD1	rw 12/build_ud11.spd
	Build_UD2	rw 12/build_ud2.spd
	SegMgmt_Prepares_Next_Segment_UD2	rw 12/segmgmt_prepare_r
	UD2_Clear_Blocks_Sent	rw 12/ul_clear_blocks_sent.spd
	SegMgmt_Update_Blocks_Sent_DR2	rw 12/segmgmt_update_bloc
	Pop_SDU_UL	rw 12/pop_sdu_ul.spd

_gen_ms_dl.spd
re.spd
re.spd
ocks_received_ur.spd
ed_dd2.spd
ocks_received_dl.spd



_gen_ms_ul.spd

.
text_segment_ud2.spd
cks_sent_dr2.spd



Package AA_Common_Types

1(18)



```
/* Options, parameters and constants. */
/*
The predefined values are given in the following format:
([D], [L]/[H]),
where
D is the default value,
L is the minimum (low range) value and
H is the maximum (high range).

All the fields in the formula may or may not be given depending
on the textual description defining the values.

All the values are the ones supported by the model in respect of
the defining text (e.g. considering the range of some constants).
*/
/* Timebase values for timer duration scaling. */

SYNONYM TIMESLOT Duration = 1; /* 14.67 ms */
SYNONYM TDMA_FRAME Duration = 4*TIMESLOT; /* 56.67 ms */
SYNONYM MULTIFRAME Duration = 18*TDMA_FRAME; /* 1.02 s */
SYNONYM SECOND Duration = 18*TDMA_FRAME; /* 17.65 TDMA frames*/

SYNONYM SIG_FRAME Duration = 4*TIMESLOT; /* 56.67 ms */
```



Package AA_Common_Types

2(18)



```
/*
**
** CMCE
**
*/
/* Number of concurrent instances of the CC service */
/* Always One in this model */
SYNONYM NO_OF_CC Natural = 1;

/* Circuit mode speech Supported TRUE (TRUE/FALSE) */
SYNONYM CIRCUIT_MODE_SPEECH_SUPPORTED Boolean = TRUE;

/* Circuit mode data Supported (TRUE/FALSE) */
SYNONYM CIRCUIT_MODE_DATA_SUPPORTED Boolean = TRUE;

/* DTMF Supported (TRUE/FALSE) */
SYNONYM DTMF_SUPPORTED Boolean = TRUE; /* EXTERNAL */

/* On/Off hook signalling supported (TRUE/FALSE) */
SYNONYM ON_OFF_HOOK_SUPPORTED Boolean = TRUE; /* EXTERNAL */

/* Direct setup signalling supported (TRUE/FALSE) */
SYNONYM DIRECT_CALL_SUPPORTED Boolean = TRUE; /* EXTERNAL */

/* Call SetUp Timer value for Called MS 1-30 Sec */
SYNONYM T_301 Duration = 30*SECOND; /* External */

/* Call SetUp Timer value for Calling MS 1-60 Sec */
SYNONYM T_302 Duration = 60*SECOND; /* External */

/* Call Initiated Timer value 1-60 Sec */
SYNONYM T_303 Duration = 60*SECOND; /* External */

/* Break Resume Timer value 4-6 Sec */
SYNONYM T_306 Duration = 5*SECOND; /* External */

/* Break Restore Timer value 6-8 Sec */
SYNONYM T_307 Duration = 7*SECOND; /* External */

/* Call Disconnect Timer value MS 1-10 Sec */
SYNONYM T_308 Duration = 10*SECOND; /* External */

/* Call Length Timer value 5-NoMax Sec */
SYNONYM T_310 Duration = 900*SECOND; /* External */

/* Call Transmission Timer value 1-300 Sec */
SYNONYM T_311 Duration = 30*SECOND; /* External */
```



Package AA_Common_Types

3(18)



```
/*
**
** MM
**
*/
/* Home ITSI Number MCC,MNC,SSI */
SYNONYM HOME_ITSI TSI_Type = (. 357,975,4545 .); /* EXTERNAL */

/* TETRA Equipment Identity 60 BITS */
SYNONYM TEI TEIType = 774488; /* EXTERNAL */

/* The following values are used in Class Of MS */
/***********************/

/* Duplex supported (TRUE/FALSE) */
SYNONYM DUPLEX_SUPPORTED Boolean = TRUE; /* EXTERNAL */

SYNONYM SINGLE_MULTI_SLOT_SUPPORTED Boolean = TRUE; /* EXTERNAL */
SYNONYM CONCURRENT_MULTI_CARRIER_OPERATION_SUPPORTED Boolean = FALSE; /* EXTERNAL */
SYNONYM END_TO_END_ENCRYPTION_SUPPORTED Boolean = FALSE; /* EXTERNAL */
SYNONYM AIR_INTERFACE_ENCRYPTION_SUPPORTED Boolean = FALSE; /* EXTERNAL */
SYNONYM CLCH_NEEDED_ON_CARRIER_CHANGE_SUPPORTED Boolean = FALSE; /* EXTERNAL */
SYNONYM CONCURRENT_CHANNELS_SUPPORTED Boolean = FALSE; /* EXTERNAL */
SYNONYM MINIMUM_MODE_SUPPORTED Boolean = TRUE; /* EXTERNAL */
SYNONYM CARRIER_SPECIFIC_SIGNALLING_CHANNEL_SUPPORTED Boolean = FALSE; /* EXTERNAL */

/* TETRA Air Interface Standard Version Number 0 (0/7) */
SYNONYM TETRA_AIR_INTERFACE_STANDARD_NUMBER_0TO7 = 0; /* EXTERNAL */
/***********************/

/* Registration Timer Sec*/
SYNONYM T_351 Duration = 30*SECOND; /* External */

/*
**
** MLE
**
*/
/* Cell re-selection preparation response time (in seconds)*/
SYNONYM T_370 Duration = 5*SECOND;
```



Package AA_Common_Types

4(18)



```
/*
**
** LLC
**
*/

/* Advanced link supported (FALSE, TRUE/FALSE) */
SYNONYM ADVANCED_LINK Boolean = TRUE;
SYNONYM BS_ADVANCED_LINK Boolean = TRUE;

/* Advanced link setup needed for unacknowledged AL (TRUE, TRUE/TRUE) */
SYNONYM AL_UNACK_SETUP_NEEDED Boolean = TRUE;

/* AL-DISC PDU as a response to an AL-SETUP-PDU supported in case
   advanced link is not supported (FALSE, TRUE/FALSE) */
SYNONYM UNSUPPORTED_ADVANCED_LINK_INDICATION = TRUE;
SYNONYM BS_UNsupported_ADVANCED_LINK_INDICATION = TRUE;

/* Received duplicates suppressed in Advanced Link unacknowledged service
   (TRUE, TRUE/FALSE) */
SYNONYM AL_UNACK_SUPPRESS_RECEIVED_DUPLICATES = TRUE;

/* Received SDUs delivered in SDU-numbering order in
   Advanced Link unacknowledged service (TRUE, TRUE/FALSE) */
SYNONYM AL_UNACK_DATA_DELIVERED_IN_ORDER = TRUE;

/* Advanced link supported (TRUE, TRUE/TRUE) */
SYNONYM PRIORITY_ORDERING Boolean = TRUE;

/* Advanced link supported (TRUE, TRUE/TRUE) */
SYNONYM CANCEL_OPERATION Boolean = TRUE;

/* Advanced link supported (FALSE, FALSE/False) */
SYNONYM PRE_EMPTIVE_CANCEL Boolean = FALSE;

/* Service request queue size for all link instances (8, 1/ Platform dependent)
SYNONYM QUEUE_SIZE Natural = 8;

/* Optional FCS in Basic Link (TRUE, TRUE/TRUE) */
SYNONYM BL_FCS Boolean = TRUE;

/* Maximum number of simultaneous link instances (1, 1/1) */
SYNONYM MAX_CONCURRENT_LINKS Natural = 1;

/* Initial timer durations for LLC, Part 2, Annex A. The unit of
   timer values is next available signalling opportunity */

/* Basic Link timers */

/* Sender retry timer (4, -/-) */
SYNONYM T_251 DURATION = 4*SIG_FRAME;
```



Package AA_Common_Types

5(18)



```
/* Basic Link constants */
/* Maximum length of TL-SDU (2595, -/-) */
SYNONYM N251 Natural = 2595;

/* Maximum number of TL-SDU retransmissions for acknowledged service (1/5, 3/5 ) */
SYNONYM N252_NO_STEALING_REPEATS Natural = 1;
SYNONYM N252_WITH_STEALING_REPEATS Natural = 3;

/* Number of TL-SDU repetitions for unacknowledged service (1/5) */
SYNONYM N253 Natural = 1;

/* Advanced Link constants */

/* Advanced link number (1/4) */
SYNONYM N261 Natural = 1;

/* Maximum number of connection setup retries (1/5) */
SYNONYM N262 Natural = 3;

/* Maximum number of disconnection retries (3/5) */
SYNONYM N263 Natural = 3;

/* Maximum Number of timeslots used per TDMA frame (1/4) */
SYNONYM N264 MaxTransmissionRateType = 1;

/* Maximum length of TL-SDU (4096) */
SYNONYM N271 MaxLengthOf_TL_SDU_Type = 4096_O;

/* Window size for TL-SDU in acknowledged service (1/3) */
SYNONYM N272 Natural = 3;

/* Maximum number of TL-SDU retransmissions (0/7) */
SYNONYM N273 Natural = 3;

/* Maximum number of segment retransmissions (0/15) */
SYNONYM N274 Natural = 3;

/* Window size for TL-SDU in unacknowledged service (1/3) */
SYNONYM N281 Natural = 3;

/* Number of repetitions for unacknowledged information (0/7) */
SYNONYM N282 Natural = 1;

/* Advanced Link timers */

/* Acknowledgement waiting timer (9, -/-) */
SYNONYM T_252 DURATION = 9*SIG_FRAME;

/* Setup waiting timer (4, -/-) */
SYNONYM T_261 DURATION = 4*SIG_FRAME;

/* Disconnection waiting timer (4, -/-) */
SYNONYM T_263 DURATION = 4*SIG_FRAME;

/* Receiver not ready validity timer for the data sending entity (36, -/-) */
SYNONYM T_271 DURATION = 36*TDMA_FRAME;

/* Receiver not ready validity timer for the data receiving entity (18, -/-) */
SYNONYM T_272 DURATION = 18*TDMA_FRAME;
```



Package AA_Common_Types

6(18)



```
/*
**
** CONP
**
*/
/* CONP supported (FALSE, TRUE/FALSE) */
SYNONYM CONP_SUPPORTED Boolean = TRUE;

/* is The calling entity supporting the fastSelect (TRUE, TRUE/FALSE)*/
SYNONYM CALLING_FAST_SELECT Boolean = TRUE;

/* is The called entity supporting the fastSelect (TRUE, TRUE/FALSE)*/
SYNONYM CALLED_FAST_SELECT Boolean = TRUE;

/*
**
** S-CLNP
**
*/
/* S_CLNP supported (TRUE, TRUE/FALSE) */
SYNONYM SCLNP_SUPPORTED Boolean = TRUE;

/* SCLNP re-sends packet after MLE_CLOSE - MLE_OPEN (FALSE, TRUE/FALSE) */
SYNONYM SCLNP_RESENGS_AFTER_CLOSE Boolean = FALSE;
```



Package AA_Common_Types

7(18)



```
/*
**
** MAC
**
*/

/* Event label inactivity time-out (30, -/-) */
SYNONYM T_201 DURATION = 30*MULTIFRAME; /* External */

/* Fragmentation time-out (9, -/-) /* Downlink signalling frames */
SYNONYM T_202 DURATION = 9*SIG_FRAME; /* External */

/* Random Access time-out (5, 5/60) */
SYNONYM T_205 DURATION = 5*MULTIFRAME;

/* Reserved Access waiting time-out (18, -/-) /* Downlink signalling frames */
SYNONYM T_206 DURATION = 18*SIG_FRAME; /* External */

/* Inactivity time-out on SCCH (30, -/-) */
SYNONYM T_208 DURATION = 30*MULTIFRAME; /* External */

/* Inactivity time-out on traffic channel (18, -/-) */
SYNONYM T_209 DURATION = 18*MULTIFRAME; /* External */

/* Timer for returning to energy economy mode (18, -/-) /* TDMA frames */
SYNONYM T_210 DURATION = 18*TDMA_FRAME; /* External */

/* ACCH time-out for transmission of TCH (36, -/-) /* TDMA frames */
SYNONYM T_211 DURATION = 36*TDMA_FRAME; /* External */

/* ACCH time-out for reception of TCH (18, -/-) /* TDMA frames */
SYNONYM T_212 DURATION = 18*TDMA_FRAME; /* External */

/* DTX timer (18, -/-) /* TDMA frames */
SYNONYM T_213 DURATION = 18*TDMA_FRAME; /* External */

/* Stealing timer (6, -/-) /* uplink opportunities */
SYNONYM T_214 DURATION = 6*SIG_FRAME; /* External */

/* MAC Constants */
/* Maximum size of TM-SDU (2632, 1/2632) /* Bits */
SYNONYM N202 Natural = 2632;

/* Number of wrong AACHs to leave assigned channel (3, -/-) */
SYNONYM N208 Natural = 3; /* External */

/* Quality threshold for serving cell (4, -/-) */
SYNONYM N210 Natural = 4; /* External */

/* Number of invalid AACHs to stop transmission of TCH (3, -/-) */
SYNONYM N211 Natural = 3; /* External */

/* Number of invalid AACHs to stop reception of TCH (3, -/-) */
SYNONYM N212 Natural = 3; /* External */

/* Number of valid AACHs to allow reception of TCH (3, -/-) */
SYNONYM N213 Natural = 3; /* External */

/* Number of transmissions if stealing repeats flag is set (4, -/-) */
SYNONYM N214 Natural = 4; /* External */
```



Package AA_Common_Types

8(18)



```
/* Types for validation models (IDEAL, PRESETTABLE, SELECTABLE) */

SYNONYM IDEAL Natural = 1;
SYNONYM NON_DETERMINISTIC Natural = 2;
SYNONYM SELECTABLE Natural = 3;

SYNONYM LLC_VALIDATION_MODEL Natural = IDEAL;
SYNONYM SCLNP_VALIDATION_MODEL Natural = SELECTABLE;
```

```
/* Definition of Natural Constants */
SYNTYPE 0TO1 = /* 1 bit */
    Natural CONSTANTS 0:1
ENDSYNTYPE;

SYNTYPE 0TO3 = /* 2 bits */
    Natural CONSTANTS 0:3
ENDSYNTYPE;

SYNTYPE 0TO7 = /* 3 bits */
    Natural CONSTANTS 0:7
ENDSYNTYPE;

SYNTYPE 0TO15 = /* 4 bits */
    Natural CONSTANTS 0:15
ENDSYNTYPE;

SYNTYPE 0TO31 = /* 5 bits */
    Natural CONSTANTS 0:31
ENDSYNTYPE;

SYNTYPE 0TO63 = /* 6 bits */
    Natural CONSTANTS 0:63
ENDSYNTYPE;

SYNTYPE 0TO127 = /* 7 bits */
    Natural CONSTANTS 0:127
ENDSYNTYPE;

SYNTYPE 0TO255 =
    Natural CONSTANTS 0:255
ENDSYNTYPE;

SYNTYPE 0TO1023 = /* 10 bits */
    Natural CONSTANTS 0:1023
ENDSYNTYPE;

SYNTYPE 0TO2047 = /* 11 bits */
    Natural CONSTANTS 0:2047
ENDSYNTYPE;

SYNTYPE 0TO4095 = /* 12 bits */
    Natural CONSTANTS 0:4095
ENDSYNTYPE;
```

```
SYNTYPE 0TO16383 = /* 14 bits */
    Natural CONSTANTS 0:16383
ENDSYNTYPE;

SYNTYPE 0TO65535 = /* 16 bits */
    Natural CONSTANTS 0:65535
ENDSYNTYPE;

SYNTYPE 0TO1048575 = /* 20 bits */
    Natural CONSTANTS 0:1048575
ENDSYNTYPE;

SYNTYPE 0TO16777215 = /* 24 bits */
    Natural CONSTANTS 0:16777215
ENDSYNTYPE;
```

```
SYNONYM BITS_IN_OCTET Natural = 8;
```



Package AA_Common_Types

9(18)



```
/* Definition of TSI types from
   ETS 300 392-1 Clause 7.2 */

SYNTYPE MCC_Type = /* 10 bits */
  Natural CONSTANTS 0:999
ENDSYNTYPE;

SYNTYPE MNC_Type = /* 14 bits */
  OCTO16383
ENDSYNTYPE;

NEWTYPE TSI_ExtensionType STRUCT
  MCC MCC_Type;
  MNC MNC_Type;
ENDNEWTYPE;

NEWTYPE TSI_ExtensionType2 STRUCT
  Present Boolean;
  TSI_Extension TSI_ExtensionType;
ENDNEWTYPE;

SYNTYPE SSI_Type = /* 24 BITS */
  OCTO16777215
ENDSYNTYPE;

SYNONYM BROADCAST_ADDRESS SSI_Type =
  16777215; /* All One's */

NEWTYPE SSI_Type2 STRUCT
  Present Boolean;
  SSI SSI_Type;
ENDNEWTYPE;

NEWTYPE TSI_Type STRUCT
  MCC MCC_Type;
  MNC MNC_Type;
  SSI SSI_Type;
ENDNEWTYPE;

SYNTYPE ITSI_Type =
  TSI_Type
ENDSYNTYPE;

SYNTYPE ATSI_Type =
  TSI_Type
ENDSYNTYPE;

SYNTYPE GTSI_Type =
  TSI_Type
ENDSYNTYPE;

SYNTYPE ISSI_Type =
  SSI_Type
ENDSYNTYPE;

SYNTYPE ASSI_Type =
  SSI_Type
ENDSYNTYPE;
```

```
SYNTYPE GSSI_Type =
  SSI_Type
ENDSYNTYPE;

NEWTYPE GSSI_Type2 STRUCT
  Present Boolean;
  GSSI GSSI_Type;
ENDNEWTYPE;

SYNTYPE USSI_Type =
  SSI_Type
ENDSYNTYPE;

/* Definition of TMI types from
   ETS 300 392-1 Clause 7.3 */

SYNTYPE SMI_Type = /* 24 BITS */
  SSI_Type
ENDSYNTYPE;

NEWTYPE TMI_Type STRUCT
  MCC MCC_Type;
  MNC MNC_Type;
  SMI SMI_Type;
ENDNEWTYPE;

/* Definition of MNI types from
   ETS 300 392-1 Clause 7.6 */

NEWTYPE MNI_Type STRUCT
  MCC MCC_Type;
  MNC MNC_Type;
ENDNEWTYPE;
```



Package AA_Common_Types

10(18)



```
/* Parameter Definitions for Service Primitives, clause 11 */
NEWTYPE CommunicationTypeType LITERALS
  POINT_TO_POINT,
  POINT_TO_MULTIPOINT,
  POINT_TO_MULTIPOINT_ACK,
  BROADCAST
ENDNEWTYPE CommunicationTypeType;

NEWTYPE CommunicationTypeType2 STRUCT
  Present Boolean;
  CommunicationType CommunicationTypeType;
ENDNEWTYPE;

/* Parameter Definitions for Service Primitives, clause 16.10 */

/* Clause 16.10.11 */
SYNTYPE FrameType =
  0TO31
ENDSYNTYPE;

/* Clause 16.10.30, Length = 14 bits */
SYNTYPE LocationAreaType = 0TO16383
ENDSYNTYPE;

/* Clause 16.10.38 */
SYNTYPE MultiFrameType =
  0TO63
ENDSYNTYPE;

/* Clause 16.10.41 */
SYNTYPE ProprietaryType =
  Natural
ENDSYNTYPE;

NEWTYPE ProprietaryType3 STRUCT
  ElementID Natural;
  Length Natural;
  Proprietary ProprietaryType;
  M_Bit Boolean;
ENDNEWTYPE;

/* Clause 16.10.50 */
SYNTYPE TEIType = Natural
ENDSYNTYPE;

/* Parameter Definition for service primitives, clause 18.5 */

/* Clause 18.5.10 */
SYNTYPE MainCarrierNoType =
  0TO4095 /* 12 bit */
ENDSYNTYPE;
```



Package AA_Common_Types

11(18)



```
/* Clause 18.5.11 */
NEWTYPE MainCarrierNoExtType STRUCT
    FrequencyBand 0TO15;
    Offset 0TO3;
    DuplexSpacing 0TO7;
    ReverseOperation 0TO1;
ENDNEWTYPE MainCarrierNoExtType;

NEWTYPE MainCarrierNoExtType2 STRUCT
    Present Boolean;
    MainCarrierNoExt MainCarrierNoExtType;
ENDNEWTYPE MainCarrierNoExtType2;

/* Clause 18.5.3 */
SYNTYPE CellIdentifierType =
    0TO31 /* 5 bit */;
ENDSYNTYPE;

/* Parameter definitions for service primitives, clause 20.2.4 */

/* Clause 20.2.4.1 */
NEWTYPE AddrTypeType
    LITERALS
        T_ISSI, T_ASSI, T_USSI, T_SMI, T_GSSI;
ENDNEWTYPE;

/* Clause 20.2.4.2 */
NEWTYPE ChanType STRUCT
    MainCarrierNo MainCarrierNoType;
    MainCarrierNoExt MainCarrierNoExtType2;
ENDNEWTYPE ChanType;

SYNTYPE ChanListIndex =
    Natural CONSTANTS 0:5
ENDSYNTYPE ChanListIndex;

/* Clause 20.2.4.2 */
NEWTYPE ChanListType
    ARRAY(ChanListIndex, ChanType);
ENDNEWTYPE;

NEWTYPE ChannelArrayType STRUCT
    Length Natural;
    ChanList ChanListType;
ENDNEWTYPE;

/* Clause 20.2.4.3 */
SYNTYPE ChannelChangeAcceptedType =
    Boolean
ENDSYNTYPE;

/* Clause 20.2.4.4 */
SYNTYPE ChannelChangeResponseRequiredType =
    Boolean
ENDSYNTYPE;

/* Clause 20.2.4.5 and 16.10.8 */
NEWTYPE DistributionOn18thFrameType
    LITERALS
        TimeSlot1,
        TimeSlot2,
        TimeSlot3,
        TimeSlot4;
ENDNEWTYPE;
```



Package AA_Common_Types

12(18)



```
/* Clause 20.2.4.6 */
NEWTYPE EncryptionType
LITERALS
    ENCRYPTED,
    CLEAR;
ENDNEWTYPE;

NEWTYPE EncryptionType2 STRUCT
    Present Boolean;
    Encryption EncryptionType;
ENDNEWTYPE;

/* Clause 20.2.4.7 */
SYNTYPE Endpoint_ID_Type =
    Natural
ENDSYNTYPE;

/* Clause 20.2.4.8 */
NEWTYPE EnergySavingModeType
LITERALS
    NO_EG, EG1, EG2,
    EG3, EG4, EG5, EG6, EG7;
ENDNEWTYPE;

NEWTYPE EnergySavingModeType2 STRUCT
    Present Boolean;
    EnergySavingMode
        EnergySavingModeType;
ENDNEWTYPE;

/* Clause 20.2.4.9 */
NEWTYPE EnergyEconomyStartpointType STRUCT
    Frame FrameType;
    MultiFrame MultiFrameType;
ENDNEWTYPE;

/* Clause 20.2.4.10 */
SYNTYPE FCS_FlagType =
    Boolean
ENDSYNTYPE;

/* Clause 20.2.4.11 */
NEWTYPE GroupCallReleaseType
LITERALS
    STAY,
    LEAVE;
ENDNEWTYPE;

/* Type for link identification in handle */
NEWTYPE Link_ID_Type
LITERALS
    NULL,
    ACK,
    UNACK,
    L2;
ENDNEWTYPE;
```



```
/* Clause 20.2.4.18 */
NEWTYPE MainAddrType STRUCT
    Addr SSI_Type;
ENDNEWTYPE;

/* Clause 20.2.4.19 */
NEWTYPE MLE_ActivityType
LITERALS
    MLE_ACTIVE,
    MLE_NOT_ACTIVE;
ENDNEWTYPE;

/* Clause 20.2.4.20 */
SYNTYPE NewEndpoint_ID_Type =
    Endpoint_ID_Type
ENDSYNTYPE;

/* Clause 20.2.4.21 */
SYNTYPE NumberOfTimeslotsType =
    Natural
ENDSYNTYPE;

/* Clause 20.2.4.22 */
NEWTYPE U_PlaneFlagType
LITERALS
    U_PLANE_ON,
    U_PLANE_OFF;
ENDNEWTYPE;

/* Clause 20.2.4.22 */
NEWTYPE TX_GrantFlagType
LITERALS
    TX_GRANTED,
    TX_CEASED;
ENDNEWTYPE;

/* Clause 20.2.4.22 */
NEWTYPE DuplexFlagType
LITERALS
    SIMPLEX,
    DUPLEX;
ENDNEWTYPE;

NEWTYPE DuplexFlagType2 STRUCT
    Present Boolean;
    DuplexFlag DuplexFlagType;
ENDNEWTYPE;

/* Clause 20.2.4.22 */
SYNTYPE InterleavingDepthType =
    Natural
ENDSYNTYPE;
```



Package AA_Common_Types

13(18)

```
/* Clause 20.2.4.22 */
NEWTYPE CircuitModeType
LITERALS
TCH_S,
TCH_7_2,
TCH4.8_N1,
TCH4.8_N4,
TCH4.8_N8,
TCH2.4_N1,
TCH2.4_N4,
TCH2.4_N8;
ENDNEWTYPE CircuitModeType;

/* Clause 20.2.4.22 */
NEWTYPE OperatingModeType STRUCT
U_PlaneFlag U_PlaneFlagType;
TX_GrantFlag TX_GrantFlagType;
DuplexFlag DuplexFlagType;
TypeOfCircuit CircuitModeType;
Encryption EncryptionType;
Endpoint_ID Endpoint_ID_Type;
ENDNEWTYPE;

/* Clause 20.2.4.23 */
SYNTYPE PathLossC1Type =
Natural
ENDSYNTYPE;

/* Clause 20.2.4.23 */
SYNTYPE PathLossC2Type =
Natural
ENDSYNTYPE;

/* Clause 20.2.4.23 */
NEWTYPE PathLossType
STRUCT
C1 PathLossC1Type;
C2 PathLossC2Type;
ENDNEWTYPE;

/* Clause 20.2.4.24 */
SYNTYPE PDU_PriorityType =
Natural
CONSTANTS 0:7
ENDSYNTYPE;

SYNONYM MIN_PDU_PRIORITY
PDU_PriorityType = 0;

SYNONYM MAX_PDU_PRIORITY
PDU_PriorityType = 7;

SYNONYM EMERGENCY_PRIORITY
PDU_PriorityType = MAX_PDU_PRIORITY;

/* Clause 20.2.4.25 */
SYNTYPE QualityIndicationType =
Natural
ENDSYNTYPE;
```

```
/* Clause 20.2.4.26 */
SYNTYPE MaxTransmissionRateType =
Natural
CONSTANTS 1:4
/* The number of timeslots used */
ENDSYNTYPE;

/* Clause 20.2.4.26 */
NEWTYPE MeanTransmissionRateType
LITERALS
NetworkDependent,
1_32,
1_16,
1_8,
1_4,
1_2,
Maximum;
/* The notation 1_32 = 1/32 */
ENDNEWTYPE;

/* Clause 20.2.4.26 */
NEWTYPE MaxLengthOf_TL_SDU_Type
LITERALS
32_O,
64_O,
128_O,
256_O,
512_O,
1024_O,
2048_O,
4096_O;
/* Notation 32_O = 32 octets */
OPERATORS
ORDERING
ENDNEWTYPE;

/* Clause 20.2.4.26 */
SYNTYPE TL_SDU_WindowSizeType =
Natural
CONSTANTS 1:3
/* Maximum window size is LLC
parameter N.272 */
ENDSYNTYPE;

/* Clause 20.2.4.26 */
NEWTYPE ThroughputType STRUCT
MaxTransmissionRate
MaxTransmissionRateType;
MeanTransmissionRate
MeanTransmissionRateType;
MaxLengthOf_TL_SDU
MaxLengthOf_TL_SDU_Type;
TL_SDU_WindowSize
TL_SDU_WindowSizeType;
ENDNEWTYPE;

/* Clause 20.2.4.26 */
SYNTYPE Max_TL_SDU_ReTransmissionsType =
Natural
CONSTANTS 0:7
ENDSYNTYPE;
```



Package AA_Common_Types

14(18)



```
/* Clause 20.2.4.26 */
SYNTYPE MaxSegmentReTransmissionsType =
    Natural
        CONSTANTS 0:15
ENDSYNTYPE;

/* Clause 20.2.4.26 */
NEWTYPE TransferFailureProbabilityType STRUCT
    Max_TL_SDU_ReTransmissions Max_TL_SDU_ReTransmissionsType;
    MaxSegmentReTransmissions MaxSegmentReTransmissionsType;
ENDNEWTYPE;

/* Clause 20.2.4.26 */
NEWTYPE QoS_Type STRUCT
    Throughput ThroughputType;
    TransferFailureProbability TransferFailureProbabilityType;
ENDNEWTYPE;

/* Clause 20.2.4.27 */
NEWTYPE ReportType
    LITERALS
        HANDLE,
        SERVICE_DEFINITION,
        SERVICE_CHANGE,
        RESET_REPORT,
        SETUP_FAILURE,
        SERVICE_TEMPORARILY_NOT_SUPPORTED,
        SERVICE_NOT_SUPPORTED,
        REJECTION,
        CLOSE,
        INCOMING_DISCONNECTION,
        DISCONNECTION_FAILURE,
        LOCAL_DISCONNECTION,
        FIRST_COMPLETE_TRANSMISSION,
        SUCCESSFUL_TRANSFER,
        FAILED_TRANSFER,
        ABORTED_SDU_NOT_COMPLETELY_SENT,
        ABORTED_SDU_SENT_AT_LEAST_ONCE,
        LAYER_TWO_TRANSMISSION_ACTIVITIES_CONTINUING,
        FIRST_COMPLETE_TRANSMISSION_BY_RANDOM_ACCESS,
        SUCCESSFUL_COMPLETE_TRANSMISSION_BY_RANDOM_ACCESS,
        COMPLETE_TRANSMISSION_BY_STEALING_OR_BY_RESERVED_ACCESS,
        RANDOM_ACCESS_FAILURE,
        FRAGMENTATION_FAILURE;
    ENDNEWTYPE;

/* Derived definition from Clause xxx, proposed by PT 93 */
NEWTYPE TLC_ReportType
    LITERALS
        USAGE_MARKER_MISMATCH,
        DOWNLINK_FAILURE,
        UPLINK_FAILURE,
        MAXIMUM_PATH_DELAY_EXCEEDED;
    ENDNEWTYPE;
```



Package AA_Common_Types

15(18)



```
/* Clause 20.2.4.28 */
NEWTYPE ScanningMeasurementMethodType
LITERALS
    FOREGROUND,
    BACKGROUND,
    INTERRUPTING;
ENDNEWTYPE;

/* Clause 20.2.4.30 */
SYNTYPE ScramblingCodeType =
    MNI_Type;
ENDSYNTYPE;

/* Clause 20.2.4.31 */
NEWTYPE SetupReportType
LITERALS
    SUCCESS,
    SERVICE_DEFINITION,
    SERVICE_CHANGE,
    RESET_REPORT;
ENDNEWTYPE;

NEWTYPE DisconnectionReportType
LITERALS
    SUCCESS,
    CLOSE,
    REJECT,
    SERVICE_NOT_SUPPORTED,
    SERVICE_TEMPORARILY_UNAVAILABLE;
ENDNEWTYPE;

NEWTYPE LinkServiceType
LITERALS
    UNACKNOWLEDGED,
    ACKNOWLEDGED;
ENDNEWTYPE;

/* Clause 20.2.4.35, 18.5.22 */
SYNTYPE SubscriberClassType =
    OT065535;
ENDSYNTYPE;

/* Clause 20.2.4.28 */
NEWTYPE SubscriberClassType2 STRUCT
    Present Boolean;
    SubscriberClass SubscriberClassType;
ENDNEWTYPE;

SYNONYM ALL_SUBSCRIBERCLASS_ALLOWED
SubscriberClassType = 65535;

/* Clause 20.2.4.36 */
SYNTYPE ThresholdLevelType =
    Natural;
ENDSYNTYPE;

/* Clause 20.2.4.37 */
SYNTYPE ThresholdValuesType =
    Natural;
ENDSYNTYPE;

/* Clause 20.2.4.38 */
SYNTYPE TL_SDU_IndexType =
    Natural
    CONSTANTS 1:110 /* N.271 */
ENDSYNTYPE;

NEWTYPE TL_SDU_ArrayType
    ARRAY(TL_SDU_IndexType, Natural);
ENDNEWTYPE;

NEWTYPE TL_SDU_Type
STRUCT
    A TL_SDU_ArrayType;
/*#ADT(W(B))
#BODY
#endif XREADANDWRITEF
extern char * yWri_(#(tl_sdu_type))( void * Value)
{
static char xCharTmp[30];
sprintf(xCharTmp,
    "(. TL-SDU : %d %d %d %d .) ",
    (*(#(tl_sdu_type) *)Value).a.A[0],
    (*(#(tl_sdu_type) *)Value).a.A[1],
    (*(#(tl_sdu_type) *)Value).a.A[2],
    (*(#(tl_sdu_type) *)Value).a.A[3],
    (*(#(tl_sdu_type) *)Value).a.A[4]);
return xCharTmp;
}
#endif
*/
ENDNEWTYPE;

/* Clause 20.2.4.39 */
SYNTYPE TL_SDU_LengthType =
    Natural /* N.271 */;
ENDSYNTYPE;

NEWTYPE ValidAddressType STRUCT
    AddrKind AddrTypeType;
    Addr SSI_Type;
ENDNEWTYPE;
```



Package AA_Common_Types

16(18)



```
/* Clause 20.2.4.42 */
SYNTYPE ValidAddressesIndexType =
    Natural CONSTANTS 0:7
ENDSYNTYPE;

NEWTYPE ValidAddrList
    ARRAY(ValidAddressesIndexType,
        ValidAddressType);
ENDNEWTYPE;

/* Clause 20.2.4.42 */
NEWTYPE ValidAddressesType STRUCT
    Length Natural;
    Element ValidAddrList;
ENDNEWTYPE ValidAddressesType;

/* Primitives at the TLB/TMB-SAP,
clause 19.2.3.3 */

/* Clause 19.2.3.3.1
NEWTYPE TLB_Broadcast1RequestType STRUCT
    Chan ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1
NEWTYPE TLB_Broadcast1IndicationType STRUCT
    Chan ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1
NEWTYPE TLB_Broadcast2RequestType STRUCT
    Chan ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1
NEWTYPE TLB_Broadcast2IndicationType STRUCT
    Chan ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;
*/
```

All of the Layer 2 Primitives
on the next 3 pages
have been moved to the
BA_Layer2_Service_Primitives
package.



Package AA_Common_Types

17(18)



```
/* Primitives at the TLC/TMC-SAP, clause 19.2.3.4 */
/* Clause xxx, proposed by PT 93
NEWTYPE TLC_ConfigureRequestType STRUCT
    ThresholdValues_present Boolean;
    ThresholdValues ThresholdValuesType;
    EnergySavingMode_present Boolean;
    EnergySavingMode EnergySavingModeType;
    MLE_Activity_present Boolean;
    MLE_Activity MLE_ActivityType;
    ValidAddresses_present Boolean;
    ValidAddresses ValidAddressesType;
ENDNEWTYPE;

/* Clause xxx, proposed by PT 93
NEWTYPE TLC_ConfigureConfirmType STRUCT
    ThresholdValues_present Boolean;
    ThresholdValues ThresholdValuesType;
    EnergySavingMode_present Boolean;
    EnergySavingMode EnergySavingModeType;
    ValidAddresses_present Boolean;
    ValidAddresses ValidAddressesType;
ENDNEWTYPE;

/* Clause 19.2.3.4.2
NEWTYPE TLC_ServingIndicationType STRUCT
    Chan ChanType;
    PathlossC1 PathLossC1Type;
    QualityIndication_present Boolean;
    QualityIndication QualityIndicationType;
ENDNEWTYPE;

/* Clause 19.2.3.4.3
NEWTYPE TLC_MonitorIndicationType STRUCT
    Chan ChanType;
    PathlossC2 PathLossC2Type;
    QualityIndication_present Boolean;
    QualityIndication QualityIndicationType;
ENDNEWTYPE;

*/
```



Package AA_Common_Types

18(18)



```
/* Clause 19.2.3.4.3
NEWTYPE TLC_MonitorRequestType STRUCT
    Chan ChanType;
    ChannelArray_present Boolean;
    ChannelArray ChannelArrayType;
ENDNEWTYPE;

/* Clause xxx, proposed by PT 93
NEWTYPE TLC_ReportIndicationType STRUCT
    Handle_present Boolean;
    Handle Endpoint_ID_Type;
    Report TLC_ReportType;
ENDNEWTYPE;

/* Clause 19.2.3.4.1
NEWTYPE TLC_ScanRequestType STRUCT
    Chan ChanType;
    ScanningMeasurementMethod
        ScanningMeasurementMethodType;
    ThresholdLevel_present Boolean;
    ThresholdLevel ThresholdLevelType;
ENDNEWTYPE;

/* Clause 19.2.3.4.1
NEWTYPE TLC_ScanConfirmType STRUCT
    Chan ChanType;
    ScanningMeasurementMethod
        ScanningMeasurementMethodType;

    PathlossC1 PathLossC1Type;
    Report ReportType;
ENDNEWTYPE;

/* Clause 20.3.5.4.7
NEWTYPE TLC_ScanReportIndicationType STRUCT
    Chan ChanType;
    PathlossC1 PathLossC1Type;
    Report_present Boolean;
    Report ReportType;
ENDNEWTYPE;

/* Clause 20.3.5.4.8
NEWTYPE TLC_SelectRequestType STRUCT
    Chan ChanType;
    ThresholdLevel_present Boolean;
    ThresholdLevel ThresholdLevelType;
    MainCarrierNo_present Boolean;
    MainCarrierNo MainCarrierNoType;
ENDNEWTYPE;

*/
```

```
/* Clause 20.3.5.4.8
NEWTYPE TLC_SelectConfirmType STRUCT
    Chan ChanType;
    ThresholdLevel ThresholdLevelType;
    MainCarrierNo_present Boolean;
    MainCarrierNo MainCarrierNoType;
    Report_present Boolean;
    Report ReportType;
ENDNEWTYPE;

NEWTYPE TLC_AddListRequestType STRUCT
    AddrList ValidAddressesType;
ENDNEWTYPE;

*/
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

1 (7)

```
/* TLA-SAP Service Primitives */
```

```
NEWTYPE TransferReportType
LITERALS
    TRANSFER_REPORT_OK,
    TRANSFER_REPORT_BS_DL_NO_RX_UR_DD1,
    TRANSFER_REPORT_BS_DL_NO_RX_UR_DD2,
    TRANSFER_REPORT_BS_UL_NO_RX_UD2_DR1,
    TRANSFER_REPORT_BS_UL_NO_RX_UD2_DR2,
    TRANSFER_REPORT_MS_DL_NO_RX_DD2,
    TRANSFER_REPORT_MS_UL_NO_RX_DR1,
    TRANSFER_REPORT_MS_UL_NO_RX_DR2,
    TRANSFER_REPORT_L2_SDU_QUEUE_FULL,
    TRANSFER_REPORT_CANCEL_FAILED_UNKNOWN_ENDPOINT,
    TRANSFER_REPORT_CANCEL_PERFORMED;
ENDNEWTYPE;
```

```
/* Clause 19.2.3.2.2 + PT93 recommendation */
NEWTYPE TLA_DataRequestType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
    ScramblingCode ScramblingCodeType;
    SubscriberClass SubscriberClassType;
    FCS_Flag FCS_FlagType;

    ProtocolReq Layer2ProtocolRequestType;
    MaxNumAccessAttempts AccessAttemptType;
ENDNEWTYPE;
```

```
/* Clause 19.2.3.2.2 + PT93 recommendation */
NEWTYPE TLA_DataIndicationType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
    FCS_Flag FCS_FlagType;

    Report TransferReportType;
    NumAccessAttempts AccessAttemptType;
ENDNEWTYPE;
```

```
/* Clause 19.2.3.2.2 */
NEWTYPE TLA_DataConfirmType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    Report TransferReportType;
ENDNEWTYPE;
```

```
NEWTYPE Layer2ProtocolRequestType
LITERALS
    PROTOCOL_REQ_ACK_ALL_BLOCKS_RETRANS,
    PROTOCOL_REQ_ACK_NO_BLOCKS_RETRANS,
    PROTOCOL_REQ_UNACK;
ENDNEWTYPE;
```

```
SYNTYPE AccessAttemptType =
    0TO15
ENDSYNTYPE;
```

```
/* Clause ???, this SP is not specified in the ETS, however it
may be necessary to handle the MLE_Cancel_Req */
NEWTYPE TLA_CancelRequestType STRUCT
    EndpointID Endpoint_ID_Type;
ENDNEWTYPE;
```

```

/* Clause 19.2.3.2.2 + PT93 recommendation */
NEWTYPE TLA_DataRequestType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
    ScramblingCode ScramblingCodeType;
    SubscriberClass SubscriberClassType;
    FCS_Flag FCS_FlagType;

    ProtocolReq Layer2ProtocolRequestType;
    MaxNumAccessAttempts AccessAttemptType;
ENDNEWTYPE;

/* Clause 19.2.3.2.2 + PT93 recommendation */
NEWTYPE TLA_DataIndicationType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
    FCS_Flag FCS_FlagType;

    Report TransferReportType;
    NumAccessAttempts AccessAttemptType;
ENDNEWTYPE;

/* Clause 19.2.3.2 */
NEWTYPE TLA_DataConfirmType STRUCT
    AddrType AddrTypeType;
    MainAddr MainAddrType;
    EndpointID Endpoint_ID_Type;
    Report TransferReportType;
ENDNEWTYPE;

```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

2(7)



```
/* Clause 19.2.3.2.3 */
NEWTYPE TLA_UnitdataRequestType STRUCT
  AddrType AddrTypeType;
  MainAddr MainAddrType;
  EndpointID_Present Boolean;
  EndpointID Endpoint_ID_Type;
  TL_SDU TL_SDU_Type;
  TL_SDU_Length TL_SDU_LengthType;
  PDU_Priority PDU_PriorityType;
  ScramblingCode ScramblingCodeType;
  SubscriberClass SubscriberClassType;
  FCS_Flag FCS_FlagType;
ENDNEWTYPE;

/* Clause 19.2.3.2.3 */
NEWTYPE TLA_UnitdataIndicationType STRUCT
  AddrType AddrTypeType;
  MainAddr MainAddrType;
  EndpointID Endpoint_ID_Type;
  TL_SDU TL_SDU_Type;
  TL_SDU_Length TL_SDU_LengthType;
  PDU_Priority PDU_PriorityType;
  FCS_Flag FCS_FlagType;
ENDNEWTYPE;

/* Clause 19.2.3.2.3 */
NEWTYPE TLA_UnitdataConfirmType STRUCT
  AddrType AddrTypeType;
  MainAddr MainAddrType;
  EndpointID Endpoint_ID_Type;
  Report TransferReportType;
ENDNEWTYPE;
```

```
/* External synonyms for Selectable options, which are used if
   ValidationOption = 2
*/
SYNONYM EXT_CANCEL_POSSIBLE Boolean= TRUE;
SYNONYM EXT_TRANSMISSION_SUCCESS Boolean=TRUE;
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

3(7)



```
/* Primitives at the TLB/TMB-SAP,
 clause 19.2.3.3 */

/* Clause 19.2.3.3.1 */
NEWTYPE TLB_Broadcast1RequestType STRUCT
    Chan   ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1 */
NEWTYPE TLB_Broadcast1IndicationType STRUCT
    Chan   ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;

/* Proposed by PT93 */
NEWTYPE TLB_Broadcast1ConfirmType STRUCT
    Report TransferReportType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1 */
NEWTYPE TLB_Broadcast2RequestType STRUCT
    Chan   ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
    PDU_Priority PDU_PriorityType;
ENDNEWTYPE;

/* Clause 19.2.3.3.1 */
NEWTYPE TLB_Broadcast2IndicationType STRUCT
    Chan   ChanType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;

/* Proposed by PT93 */
NEWTYPE TLB_Broadcast2ConfirmType STRUCT
    Report TransferReportType;
ENDNEWTYPE;
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

4(7)



```
/* Primitives at the TLC/TMC-SAP, clause 19.2.3.4 */

/* Clause xxx, proposed by PT 93 */
NEWTYPE TLC_ConfigureRequestType STRUCT
    ThresholdValues_present Boolean;
    ThresholdValues ThresholdValuesType;
    EnergySavingMode_present Boolean;
    EnergySavingMode EnergySavingModeType;
    MLE_Activity_present Boolean;
    MLE_Activity MLE_ActivityType;
    ValidAddresses_present Boolean;
    ValidAddresses ValidAddressesType;
ENDNEWTYPE;

/* Clause xxx, proposed by PT 93 */
NEWTYPE TLC_ConfigureConfirmType STRUCT
    ThresholdValues_present Boolean;
    ThresholdValues ThresholdValuesType;
    EnergySavingMode_present Boolean;
    EnergySavingMode EnergySavingModeType;
    ValidAddresses_present Boolean;
    ValidAddresses ValidAddressesType;
ENDNEWTYPE;

/* Clause 19.2.3.4.2 */
NEWTYPE TLC_ServingIndicationType STRUCT
    Chan ChanType;
    PathlossC1 PathLossC1Type;
    QualityIndication_present Boolean;
    QualityIndication QualityIndicationType;
ENDNEWTYPE;

/* Clause 19.2.3.4.3 */
NEWTYPE TLC_MonitorIndicationType STRUCT
    Chan ChanType;
    PathlossC2 PathLossC2Type;
    QualityIndication_present Boolean;
    QualityIndication QualityIndicationType;
ENDNEWTYPE;
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

5 (7)



```
/* Clause 19.2.3.4.3 */
NEWTYPE TLC_MonitorRequestType STRUCT
    Chan ChanType;
    ChannelArray_present Boolean;
    ChannelArray ChannelArrayType;
ENDNEWTYPE;

/* Clause xxx, proposed by PT 93 */
NEWTYPE TLC_ReportIndicationType STRUCT
    Handle_present Boolean;
    Handle Endpoint_ID_Type;
    Report TLC_ReportType;
ENDNEWTYPE;

/* Clause 19.2.3.4.1 */
NEWTYPE TLC_ScanRequestType STRUCT
    Chan ChanType;
    ScanningMeasurementMethod
        ScanningMeasurementMethodType;
    ThresholdLevel_present Boolean;
    ThresholdLevel ThresholdLevelType;
ENDNEWTYPE;

/* Clause 19.2.3.4.1 */
NEWTYPE TLC_ScanConfirmType STRUCT
    Chan ChanType;
    ScanningMeasurementMethod
        ScanningMeasurementMethodType;
    /* replaced by Signal quality ? */
    PathlossC1 PathLossC1Type;
    Report ReportType; /* ? */
ENDNEWTYPE;

/* Clause 20.3.5.4.7 */
NEWTYPE TLC_ScanReportIndicationType STRUCT
    Chan ChanType;
    PathlossC1 PathLossC1Type;
    Report_present Boolean;
    Report ReportType; /* ? */
ENDNEWTYPE;

/* Clause 20.3.5.4.8 */
NEWTYPE TLC_SelectRequestType STRUCT
    Chan ChanType;
    ThresholdLevel_present Boolean;
    ThresholdLevel ThresholdLevelType;
    MainCarrierNo_present Boolean;
    MainCarrierNo MainCarrierNoType;
ENDNEWTYPE;
```

```
/* Clause 20.3.5.4.8 */
NEWTYPE TLC_SelectConfirmType STRUCT
    Chan ChanType;
    ThresholdLevel ThresholdLevelType;
    MainCarrierNo_present Boolean;
    MainCarrierNo MainCarrierNoType;
    Report_present Boolean;
    Report ReportType;
ENDNEWTYPE;

NEWTYPE TLC_AddListRequestType STRUCT
    AddrList ValidAddressesType;
ENDNEWTYPE;
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

6(7)

```
/* Service primitive definitions for MLE-LLC, clause 20.3.4 */
SIGNAL
  TLA_DATA_request(TLA_DataRequestType),
  TLA_DATA_indication(TLA_DataIndicationType),
  TLA_DATA_confirm(TLA_DataConfirmType),
  TLA_UNITDATA_request(TLA_UnitdataRequestType),
  TLA_UNITDATA_confirm(TLA_UnitdataConfirmType),
  TLA_UNITDATA_indication(TLA_UnitdataIndicationType),
  TLA_CANCEL_request(TLA_CancelRequestType),

  TLB_BROADCAST1_request(TLB_Broadcast1RequestType),
  TLB_BROADCAST2_request(TLB_Broadcast2RequestType),
  TLB_BROADCAST1_indication(TLB_Broadcast1IndicationType),
  TLB_BROADCAST2_indication(TLB_Broadcast2IndicationType),
  TLB_BROADCAST1_confirm(TLB_Broadcast1ConfirmType),
  TLB_BROADCAST2_confirm(TLB_Broadcast2ConfirmType),

  TLC_CONFIGURE_request(TLC_ConfigureRequestType),
  TLC_CONFIGURE_confirm(TLC_ConfigureConfirmType),
  TLC_SERVING_indication(TLC_ServingIndicationType),
  TLC_MONITOR_request(TLC_MonitorRequestType),
  TLC_MONITOR_indication(TLC_MonitorIndicationType),
  TLC_REPORT_indication(TLC_ReportIndicationType),
  TLC_SCAN_request(TLC_ScanRequestType),
  TLC_SCAN_confirm(TLC_ScanConfirmType),
  TLC_SCAN_REPORT_indication(TLC_ScanReportIndicationType),
  TLC_SELECT_request(TLC_SelectRequestType),
  TLC_SELECT_confirm(TLC_SelectConfirmType);

SIGNALLIST TLA_DataRequests =
  TLA_DATA_request,
  TLA_CANCEL_request;

SIGNALLIST TLAU_DataRequests =
  TLA_UNITDATA_request,
  TLA_CANCEL_request;
```

```
USE AA_Common_Types;
```

Package BA_Layer2_Service_Primitives

7(7)



```
SIGNALLIST TLA_DataIndications =
    TLA_DATA_indication,
    TLA_DATA_confirm;

SIGNALLIST TLAU_DataIndications =
    TLA_UNITDATA_indication,
    TLA_UNITDATA_confirm;

SIGNALLIST TLA_DataRequests =
    (TLAA_DataRequests),
    (TLAU_DataRequests);

SIGNALLIST TLA_DataIndications =
    (TLAA_DataIndications),
    (TLAU_DataIndications);

SIGNALLIST TLA_Requests =
    (TLA_DataRequests),
    TLA_Cancel_request;

SIGNALLIST TLA_Indications =
    (TLA_DataIndications);
```

```
SIGNALLIST TLB_Requests =
    TLB_BROADCAST1_request,
    TLB_BROADCAST2_request;

SIGNALLIST TLB_Indications =
    TLB_BROADCAST1_indication,
    TLB_BROADCAST2_indication;

SIGNALLIST TLB_Confirms =
    TLB_BROADCAST1_confirm,
    TLB_BROADCAST2_confirm;
/*
SIGNALLIST TLC_Requests =
    TLC_CONFIGURE_request,
    TLC_MONITOR_request,
    TLC_SCAN_request,
    TLC_SELECT_request;
*/
SIGNALLIST TLC_Indications =
    TLC_CONFIGURE_confirm,
    TLC_SERVING_indication,
    TLC_MONITOR_indication,
    TLC_REPORT_indication,
    TLC_SCAN_confirm,
    TLC_SELECT_confirm;
```

```
USE AA_Common_Types;
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

1(6)



```
NEWTYPE BLOCK_ARRAY
    Array(Integer,Boolean)
ENDNEWTYPE BLOCK_ARRAY;
```

```
NEWTYPE BLOCK_NUMBER_ARRAY
    Array(Integer,Integer)
ENDNEWTYPE BLOCK_NUMBER_ARRAY;
```

```
NEWTYPE BAD_PRESIDING_BLOCK_MGMT STRUCT
    Enable Boolean;
    Pattern_BS BLOCK_ARRAY;
    Pattern_MS BLOCK_ARRAY;
    PatternSize Integer;
    Index Integer;
ENDNEWTYPE;
```

```
NEWTYPE BAD_DATA_BLOCK_MGMT STRUCT
    Enable Boolean;
    MaxVal Integer;
    Blk BLOCK_NUMBER_ARRAY;
ENDNEWTYPE;
```

```
/* SDU Array Management */
NEWTYPE SDU_ARRAY_MGMT STRUCT
    NumPending Integer;
    QueueSize Integer;
    TrailingIndex Integer;
    LeadingIndex Integer;
ENDNEWTYPE;
```

```
/* SLEEP PRIMITIVES */
NEWTYPE TLC_SleepRequestType STRUCT
    /* Supported fields */
    EnableSleep Boolean;
ENDNEWTYPE;
```

```
/* TEST SETUP PRIMITIVES */
NEWTYPE TLC_TestSetupRequestType STRUCT
    /* Supported fields */
    PresBlkMgmt BAD_PRESIDING_BLOCK_MGMT;
    DataBlkMgmt BAD_DATA_BLOCK_MGMT;
ENDNEWTYPE;
```

```
USE AA_Common_Types;
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

2(6)



```
NEWTYPE PDUTypeType
LITERALS
  PDU_TYPE_SYSINFO1,
  PDU_TYPE_SYSINFO2,
  PDU_TYPE_DOWNLINK_DATA1,
  PDU_TYPE_DOWNLINK_DATA2,
  PDU_TYPE_WAKEUP,
  PDU_TYPE_ACCESS_ANNOUNCE,
  PDU_TYPE_DOWNLINK_RESP1,
  PDU_TYPE_DOWNLINK_RESP2,
  PDU_TYPE_UPLINK_DATA1,
  PDU_TYPE_UPLINK_DATA2,
  PDU_TYPE_UPLINK_RESP;
ENDNEWTYPE;
```

```
NEWTYPE L2ProtocolType
LITERALS
  L2_PROTOCOL_TYPE_ACKNOWLEDGED,
  L2_PROTOCOL_TYPE_UNACKNOWLEDGED;
ENDNEWTYPE;
```

```
SYNTYPE BurstLengthType = /* 6 bits */
  0TO63
ENDSYNTYPE;

SYNTYPE SDUPriorityType = /* 3 bits */
  0TO7
ENDSYNTYPE;

SYNTYPE RelativeBlockNumberType = /* 8 bits */
  0TO255
ENDSYNTYPE;

SYNTYPE CarrierNumberType = /* 10 bits */
  0TO1023
ENDSYNTYPE;

SYNTYPE TrafficLabelType = /* 10 bits */
  0TO1023
ENDSYNTYPE;

SYNTYPE BroadcastLabelType = /* 2 bits */
  0TO3
ENDSYNTYPE;

SYNTYPE PDUNumberType = /* 3 bits */
  0TO7
ENDSYNTYPE;

SYNTYPE SegmentSizeType = /* 6 bits */
  0TO63
ENDSYNTYPE;

SYNTYPE BlocksReservedType = /* 8 bits */
  0TO255
ENDSYNTYPE;
```

```
USE AA_Common_Types;
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

3(6)



```
/* MCCH */
NEWTYPE SYSINFO_1_PDU STRUCT
  PDUType PDUTypeType;
  TL_SDU TL_SDU_Type;
  TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;

NEWTYPE SYSINFO_2_PDU STRUCT
  PDUType PDUTypeType;
  BurstLength BurstLengthType;
  BRLabel BroadcastLabelType;
  TL_SDU TL_SDU_Type;
  TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;
```

```
/* ACCH */
NEWTYPE ACCESS_ANNOUNCE_PDU STRUCT
/* Supported */
  PDUType PDUTypeType;
  WindowMFNStart Integer;
  WindowFNStart Integer;
  WindowSize Integer;
  Priority1 SDUPriorityType;
  Priority2 SDUPriorityType;
  SubWindow1Start RelativeBlockNumber;
  SubWindow1End RelativeBlockNumber;
  SubWindow2Start RelativeBlockNumber;
  SubWindow2End RelativeBlockNumber;
  RACHCarrierNumber CarrierNumberType;
ENDNEWTYPE;

NEWTYPE WAKEUP_PDU STRUCT
/* New */
  NextWU_MultiframeNumber Integer;
  NextWU_FrameNumber Integer;
/* Supported */
  PDUType PDUTypeType;
ENDNEWTYPE;

NEWTYPE DOWNLINK_RESP_1_PDU STRUCT
/* New */
  UD2MultiframeNumber Integer;
  UD2FrameNumber Integer;
/* Supported */
  MainAddr MainAddrType;
  AddrType AddrTypeType;
  PDUType PDUTypeType;
  UTLabel TrafficLabelType;
  NumberBlocksReserved BlocksReserv;
  UplinkCarrierNumber CarrierNumber;
  DownlinkCarrierNumber CarrierNumber;
ENDNEWTYPE;
```

```
/* ACCH */
NEWTYPE DOWNLINK_DATA_1_PDU STRUCT
/* New */
  ProtocolSel L2ProtocolType;
  URMultiframeNumber Integer;
  URFrameNumber Integer;
  NumAccessAttempts AccessAttemptType;
/* Supported */
  PDUType PDUTypeType;
  DTLLabel TrafficLabelType;
  MainAddr MainAddrType;
  AddrType AddrTypeType;
  PDU_Priority PDU_PriorityType;
  UplinkCarrierNumber CarrierNumberType;
  DownlinkCarrierNumber CarrierNumberType;
ENDNEWTYPE;
```

```
/* DTCH */
NEWTYPE DOWNLINK_DATA_2_PDU STRUCT
/* New */
  CloseSDU Boolean;
  NewSegment Boolean;
  URMultiframeNumber Integer;
  URFrameNumber Integer;
/* Supported */
  PDUType PDUTypeType;
  BurstLength BurstLengthType;
  DTLLabel TrafficLabelType;
  PDUNumber PDUNumberType;
  SegmentSize SegmentSizeType;
  BlockFlags BLOCK_ARRAY;
ENDNEWTYPE;

NEWTYPE DOWNLINK_RESP_2_PDU STRUCT
/* New */
  UD2MultiframeNumber Integer;
  UD2FrameNumber Integer;
/* Supported */
  PDUType PDUTypeType;
  UTLabel TrafficLabelType;
  PDUNumber PDUNumberType;
  BlockFlags BLOCK_ARRAY;
  NumberBlocksReserved BlocksReserv;
ENDNEWTYPE;
```

```
/* MCCH */
NEWTYPE SYSINFO_1_PDU STRUCT
    PDUType PDUTypeType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;

NEWTYPE SYSINFO_2_PDU STRUCT
    PDUType PDUTypeType;
    BurstLength BurstLengthType;
    BRLLabel BroadcastLabelType;
    TL_SDU TL_SDU_Type;
    TL_SDU_Length TL_SDU_LengthType;
ENDNEWTYPE;
```

```

/* ACCH */
NEWTYPE ACCESS_ANNOUNCE_PDU STRUCT
/* Supported */
    PDUType PDUTypeType;
    WindowMFNStart Integer;
    WindowFNStart Integer;
    WindowSize Integer;
    Priority1 SDUPriorityType;
    Priority2 SDUPriorityType;
    SubWindow1Start RelativeBlockNumberType;
    SubWindow1End RelativeBlockNumberType;
    SubWindow2Start RelativeBlockNumberType;
    SubWindow2End RelativeBlockNumberType;
    RACHCarrierNumber CarrierNumberType;
ENDNEWTYPE;

NEWTYPE WAKEUP_PDU STRUCT
/* New */
    NextWU_MultiframeNumber Integer;
    NextWU_FrameNumber Integer;
/* Supported */
    PDUType PDUTypeType;
ENDNEWTYPE;

NEWTYPE DOWNLINK_RESP_1_PDU STRUCT
/* New */
    UD2MultiframeNumber Integer;
    UD2FrameNumber Integer;
/* Supported */
    MainAddr MainAddrType;
    AddrType AddrTypeType;
    PDUType PDUTypeType;
    UTLabel TrafficLabelType;
    NumberBlocksReserved BlocksReservedType;
    UplinkCarrierNumber CarrierNumberType;
    DownlinkCarrierNumber CarrierNumberType;
ENDNEWTYPE;

```

```
/* DTCH */
NEWTYPE DOWNLINK_DATA_2_PDU STRUCT
/* New */
    CloseSDU Boolean;
    NewSegment Boolean;
    URMultiframeNumber Integer;
    URFframeNumber Integer;
/* Supported */
    PDUType PDUTypeType;
    BurstLength BurstLengthType;
    DTLlabel TrafficLabelType;
    PDUNumber PDUNumberType;
    SegmentSize SegmentSizeType;
    BlockFlags BLOCK_ARRAY;
ENDNEWTYPE;

NEWTYPE DOWNLINK_RESP_2_PDU STRUCT
/* New */
    UD2MultiframeNumber Integer;
    UD2FrameNumber Integer;
/* Supported */
    PDUType PDUTypeType;
    UTLabel TrafficLabelType;
    PDUNumber PDUNumberType;
    BlockFlags BLOCK_ARRAY;
    NumberBlocksReserved BlocksReservedType;
ENDNEWTYPE;
```

```
USE AA_Common_Types;  
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

4(6)

Uplink PDUs

```
/* RACH */  
NEWTYPE UPLINK_DATA_1_PDU STRUCT  
/* New */  
    ProtocolSel L2ProtocolType;  
    CompensateReqForBadBlocks Boolean;  
    NumAccessAttempts AccessAttemptType;  
/* Supported */  
    PDUType PDUTypeType;  
    MainAddr MainAddrType;  
    AddrType AddrTypeType;  
    PDU_Priority PDU_PriorityType;  
    NumberDataBlocksRequested BlocksReservedType;  
ENDNEWTYPE;
```

```
/* UTCI */  
NEWTYPE UPLINK_DATA_2_PDU STRUCT  
/* New */  
    CloseSDU Boolean;  
    NewSegment Boolean;  
    SegmentSize SegmentSizeType;  
/* Supported */  
    PDUType PDUTypeType;  
    BurstLength BurstLengthType;  
    UTLabel TrafficLabelType;  
    PDUNumber PDUNumberType;  
    NumberDataBlocksRequested BlocksReservedType;  
    BlockFlags BLOCK_ARRAY;  
ENDNEWTYPE;
```

```
/* UTCI */  
NEWTYPE UPLINK_RESP_PDU STRUCT  
/* Supported */  
    PDUType PDUTypeType;  
    DTLabel TrafficLabelType;  
    PDUNumber PDUNumberType;  
    BlockFlags BLOCK_ARRAY;  
ENDNEWTYPE;
```

```
USE AA_Common_Types;
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

5(6)



```
***** Base-only constants *****
SYNONYM EVENT_LABEL_DT TrafficLabelType = 127;
SYNONYM EVENT_LABEL_UT TrafficLabelType = 33;

SYNONYM CARRIER_NUM_MCCH CarrierNumberType = 0;
SYNONYM CARRIER_NUM_ACCH CarrierNumberType = 1;
SYNONYM CARRIER_NUM_DTCH CarrierNumberType = 2;
SYNONYM CARRIER_NUM_UTCH CarrierNumberType = 3;
SYNONYM CARRIER_NUM_RACH CarrierNumberType = 4;

SYNONYM RACH_WINDOW_SIZE Integer = 20;

***** Mobile-only constants *****
SYNONYM SHORT_SUBSCRIBER_IDENTITY SSI_Type = 34;
/* max number of AA's between MS access attempts */
SYNONYM MAX_RETRY_WINDOW_COUNT Integer = 4;

***** Common constants *****
SYNONYM MULTIFRAMES_PER_SUPERFRAME Integer = 4096;
SYNONYM FRAMES_PER_MULTIFRAME Integer = 8;

SYNONYM MAX_DATA_BLOCKS_PER_BURST Integer = 3; /* spec: 40 */
SYNONYM MAX_DATA_BLOCKS_PER_SEG Integer = 7; /* spec: 40 */

SYNONYM NUM_FRAMES_PER_TEST Duration = 500;
SYNONYM NUM_FRAMES_PER_PDU_TEST Duration = 3000;

SYNONYM SDU_PRIORITY SDUPriorityType = 4;

SYNONYM SDU_QUEUE_SIZE Integer = 3;

SYNONYM MAX_RETRIES Integer = 2; /* N253 */
SYNONYM DR1_INTERVAL Duration = 72; /* T255 */
SYNONYM DL_INTERVAL Duration = 24; /* T256 */
SYNONYM DL_INTERVAL_2 Duration = 48;
SYNONYM DL_INTERVAL_3 Duration = 72;
```

```
USE AA_Common_Types;
USE BA_Layer2_Service_Primitives;
```

Package CA_Layer2_Private

6 (6)

```

-----+
-----+ SIGNAL
-----+ /* System Timing */
-----+ FrameSync(Integer, Integer),
-----+ FrameSyncDL(Integer, Integer),
-----+ FrameSyncUL(Integer, Integer),

-----+ /* Broadcast PDUs - Downlink */
-----+ PBLK_SysInfo1(SYSINFO_1_PDU, CarrierNumberType),
-----+ PBLK_SysInfo2(SYSINFO_2_PDU, CarrierNumberType),
-----+ PBLK_AccessAnnounce(ACCESS_ANNOUNCE_PDU, CarrierNumberType),
-----+ PBLK_DownlinkData1(DOWNLINK_DATA_1_PDU, Boolean, CarrierNumberType),

-----+ /* Broadcast PDUs - Uplink */
-----+ PBLK_UplinkData1(UPLINK_DATA_1_PDU, Boolean, CarrierNumberType),

-----+ /* Base PDUs */
-----+ PBLK_DownlinkData2(DOWNLINK_DATA_2_PDU, Boolean, CarrierNumberType),

-----+ /* Base ACK PDUs */
-----+ PBLK_DownlinkResp1(DOWNLINK_RESP_1_PDU, Boolean, CarrierNumberType),
-----+ PBLK_DownlinkResp2(DOWNLINK_RESP_2_PDU, Boolean, CarrierNumberType),
-----+ PBLK_WakeUp(WAKEUP_PDU, CarrierNumberType),

-----+ /* Mobile PDUs */
-----+ PBLK_UplinkData2(UPLINK_DATA_2_PDU, Boolean, CarrierNumberType),

-----+ /* Mobile ACK PDUs */
-----+ PBLK_UplinkResp(UPLINK_RESP_PDU, Boolean, CarrierNumberType),

-----+ /* Generic Data Block */
-----+ FBLK_DataBlock(Integer, Boolean, CarrierNumberType),

-----+ /* MLE Primitives - Testing Only */
-----+ TLC_SLEEP_Request(TLC_SleepRequestType),
-----+ TLC_TESTSETUP_Request_DL(TLC_TestSetupRequestType),
-----+ TLC_TESTSETUP_Request_UL(TLC_TestSetupRequestType);

```

```
/*#INCLUDE 'random.pr' */
/*#INCLUDE 'file.pr' */
```

```
SIGNALLIST FrameSyncs =
FrameSync,
FrameSyncUL,
FrameSyncDL;
```

```

SIGNALLIST BroadcastPDUs =
PBLK_SysInfo1,
PBLK_SysInfo2,
PBLK_AccessAnnounce,
PBLK_WakeUp;

SIGNALLIST BasePDUs =
PBLK_DownlinkData1,
PBLK_DownlinkData2,
FBLK_DataBlock;

SIGNALLIST BaseAckPDUs =
PBLK_DownlinkResp1,
PBLK_DownlinkResp2;

SIGNALLIST MobilePDUs =
PBLK_UplinkData1,
PBLK_UplinkData2,
FBLK_DataBlock;

SIGNALLIST MobileAckPDUs =
PBLK_UplinkResp;

```

```
SIGNALLIST TLC_Requests =
TLC_SLEEP_Request,
TLC_TESTSETUP_Request_DL,
TLC_TESTSETUP_Request_UL,
TLC_CONFIGURE_request,
TLC_MONITOR_request,
TLC_SCAN_request,
TLC_SELECT_request;
```

```
SIGNALLIST TLC_Requests_DL =
TLC_TESTSETUP_Request_DL,
TLC_SLEEP_Request,
TLC_CONFIGURE_request,
TLC_MONITOR_request,
TLC_SCAN_request,
TLC_SELECT_request;
```

```

SIGNALLIST FrameSyncsMS =
FrameSyncUL,
FrameSyncDL;

SIGNALLIST TLA_Requests2 =
TLA_DATA_Request,
TLA_UNITDATA_Request,
TLA_CANCEL_Request;

SIGNALLIST TLA_Responses =
TLA_DATA_Indication,
TLA_UNITDATA_Indication,
TLA_DATA_Confirm,
TLA_UNITDATA_Confirm;

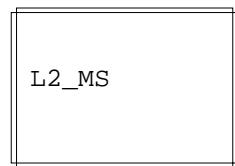
```

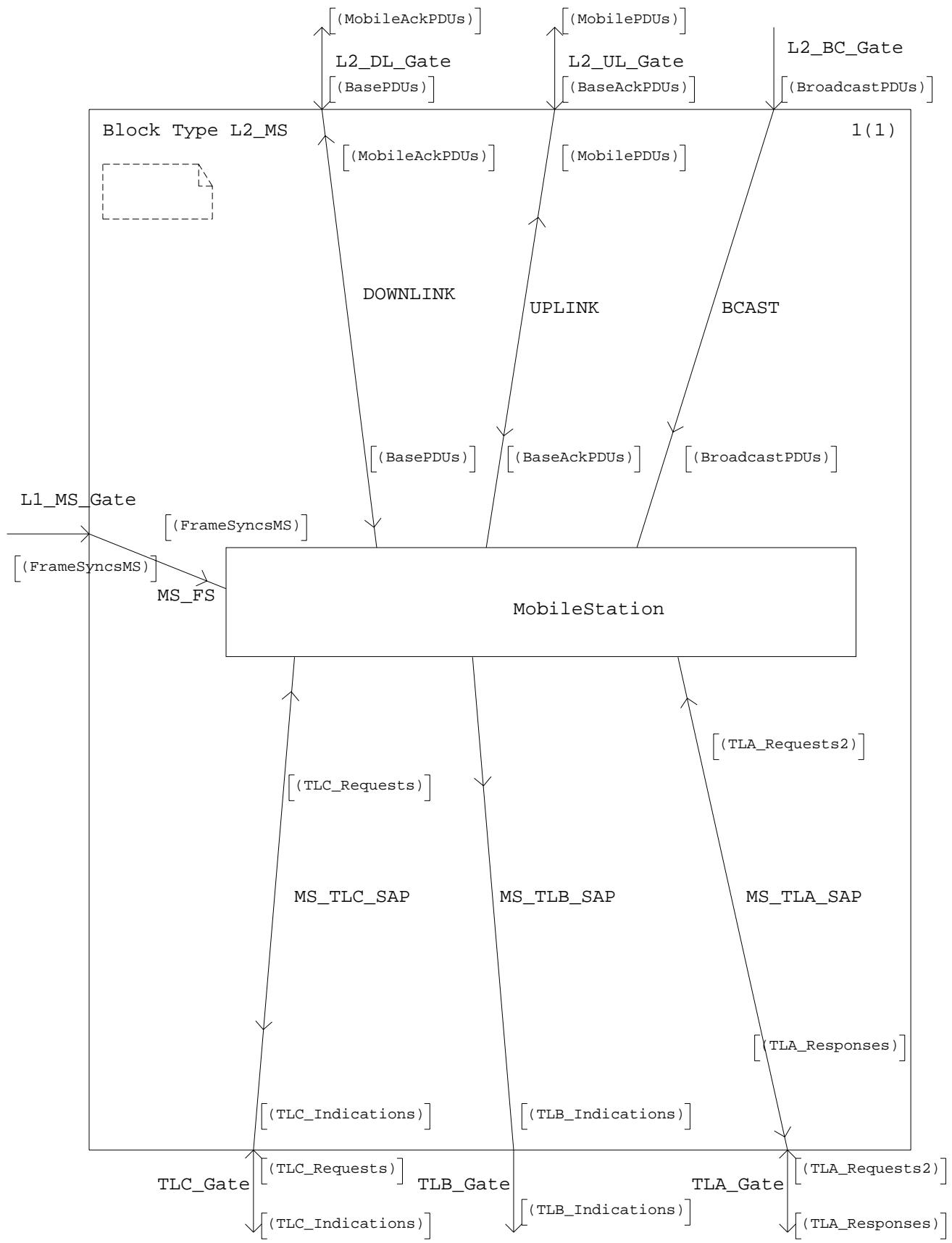
```
SIGNALLIST TLC_Requests_BS =
TLC_SLEEP_Request,
TLC_CONFIGURE_Request,
TLC_MONITOR_Request,
TLC_SCAN_Request,
TLC_SELECT_Request;
```

```
USE AA_Common_Types;  
USE BA_Layer2_Service_Primitives;  
USE CA_Layer2_Private;
```

Package L2_MS

1(1)



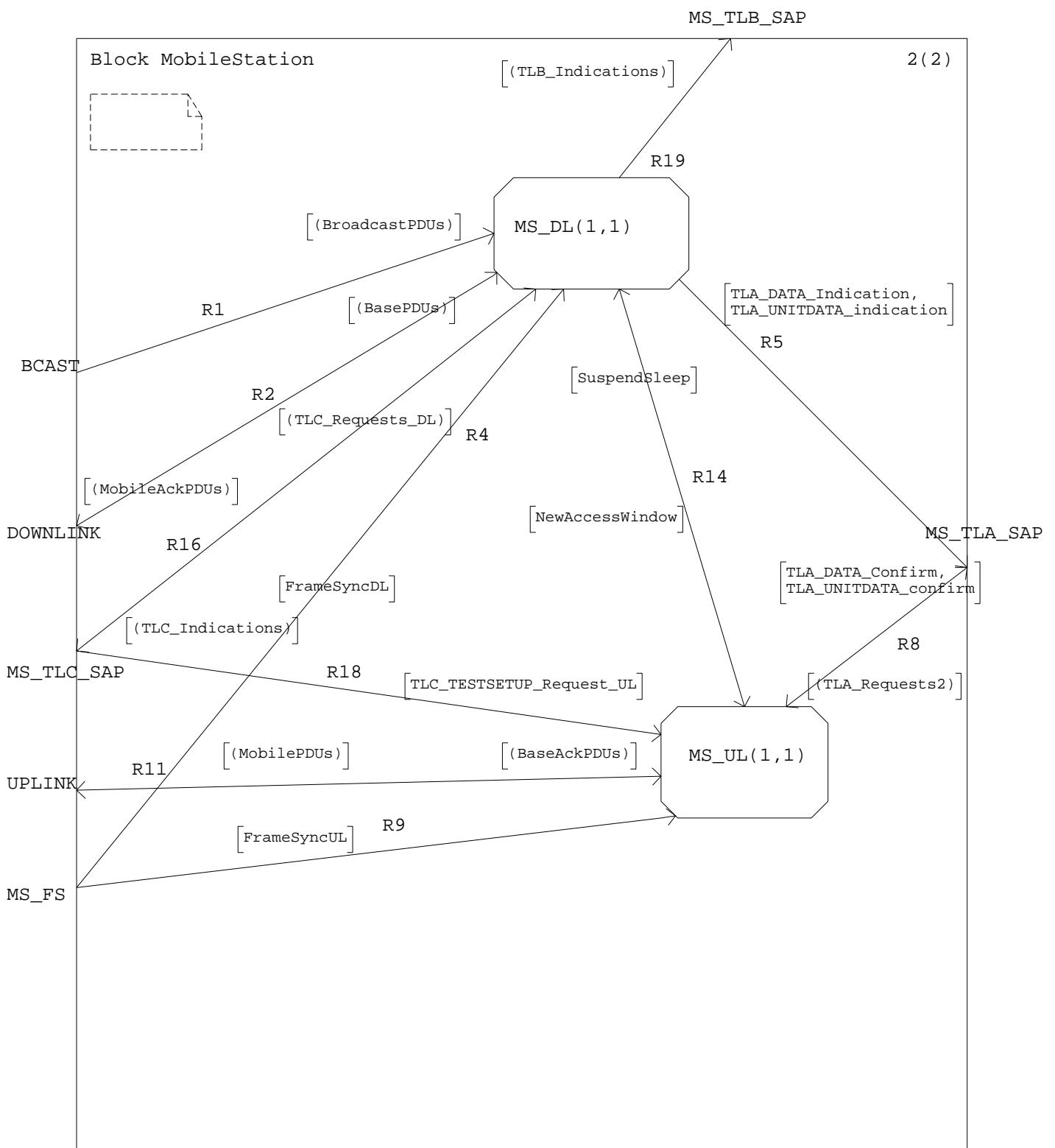


Block MobileStation

1(2)



```
SIGNAL  
NewAccesswindow(ACCESS_ANNOUNCE_PDU),  
SuspendSleep(Boolean);
```



Process MS_DL

1(15)



```
/* SDU Management */
NEWTYPE SDU_MGMT STRUCT
    ReturnToACCH Boolean;
    DTLlabel TrafficLabelType;
    ProtocolSel L2ProtocolType;
    MainAddr MainAddrType;
    AddrType AddrTypeType;
    PDU_Priority PDU_PriorityType;
    NumAccessAttempts AccessAttemptType;
    NumSDUDataBlocksRxOK Integer;
    TL_SDU TL_SDU_Type;
    BlockReceived BLOCK_ARRAY;
    TL_SDU_Length Integer;
    SegStart Integer;
    NextPDUNumber PDUNumberType;
ENDNEWTYPE;

/* Segment Management */
NEWTYPE SEG_MGMT STRUCT
    NumSegDataBlocksRxOK Integer;
    BlockExpected BLOCK_NUMBER_ARRAY;
ENDNEWTYPE;

/* Burst Management */
NEWTYPE BURST_MGMT STRUCT
    BurstInProg Boolean;
    NumBurstDataBlocks Integer;
    NumBurstDataBlocksRx Integer;
    TxRequestUR Boolean;
    URMultiframeNumber Integer;
    URFrameNumber Integer;
ENDNEWTYPE;
```

```
Timer DL_Timeout,Sleep_Timeout;
```

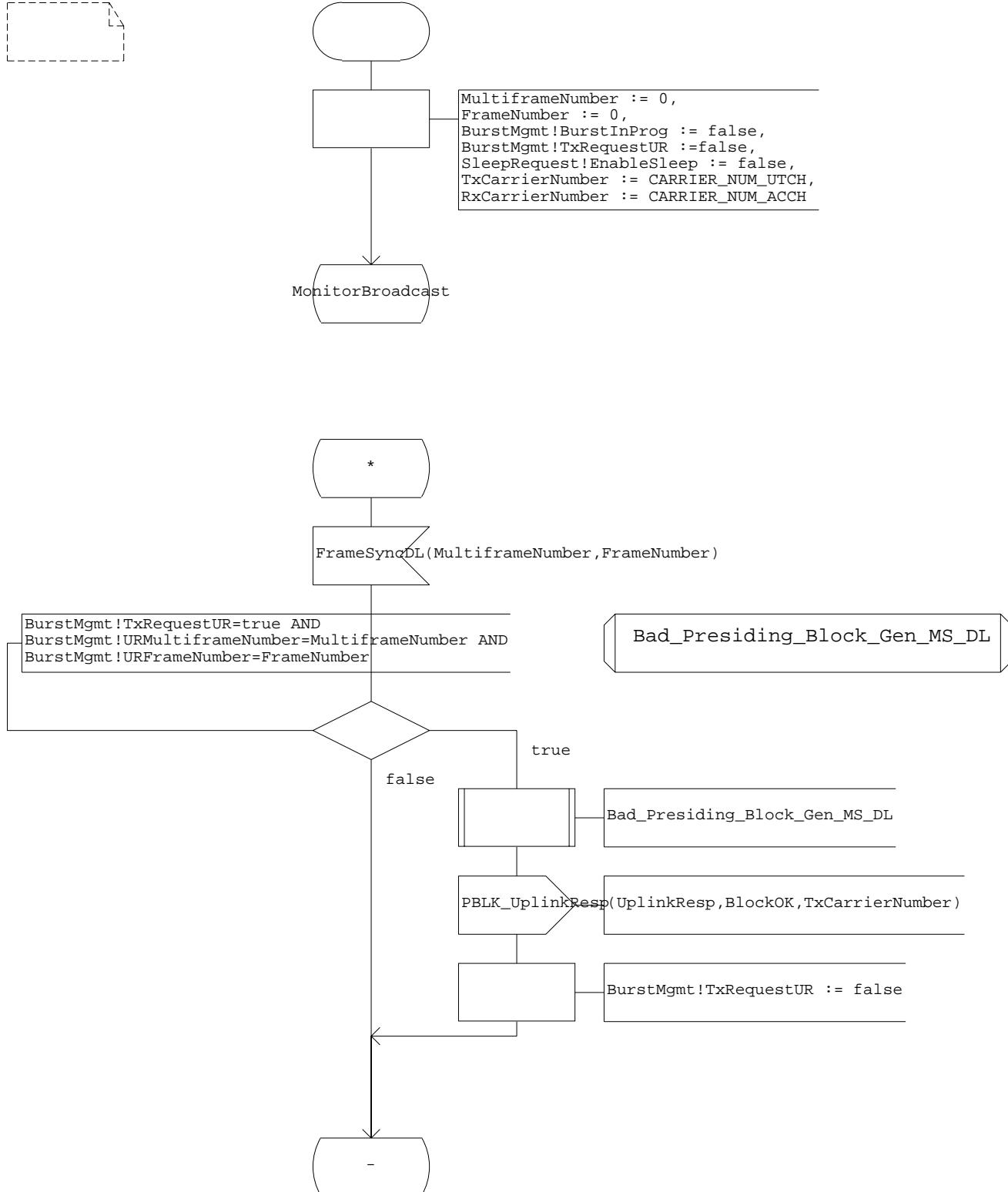
```
DCL
MultiframeNumber Integer,
FrameNumber Integer,
SDUMgmt SDU_MGMT,
SegMgmt SEG_MGMT,
BurstMgmt BURST_MGMT;
```

```
DCL
Sysinfo1Indication TLB_Broadcast1IndicationType,
Sysinfo2Indication TLB_Broadcast2IndicationType,
SysInfo1 SYSINFO_1_PDU,
SysInfo2 SYSINFO_2_PDU,
AccessAnnounce ACCESS_ANNOUNCE_PDU,
WakeUp WAKEUP_PDU,
DownlinkData1 DOWNLINK_DATA_1_PDU,
DownlinkData2 DOWNLINK_DATA_2_PDU,
UplinkResp UPLINK_RESP_PDU;
```

```
DCL
BlockNumber Integer,
BlockData Integer,
TestSetup TLC_TestSetupRequestType,
BlockOK Boolean,
CarrierNumber CarrierNumberType,
TxCarrierNumber CarrierNumberType,
RxCarrierNumber CarrierNumberType,
SleepTime Duration,
SuspendSleep Boolean,
SleepRequest TLC_SleepRequestType;
```

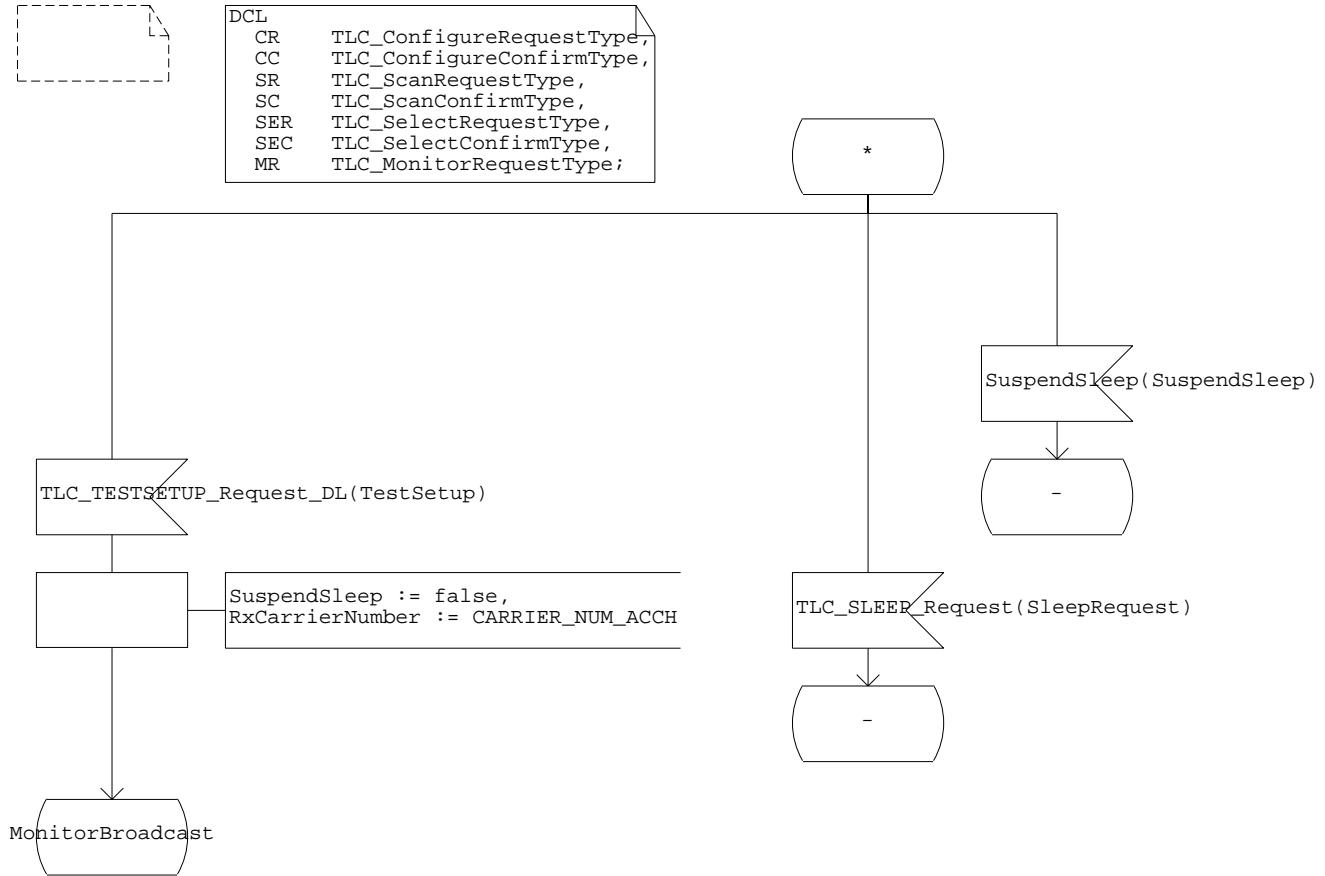
Process MS_DL

2 (15)



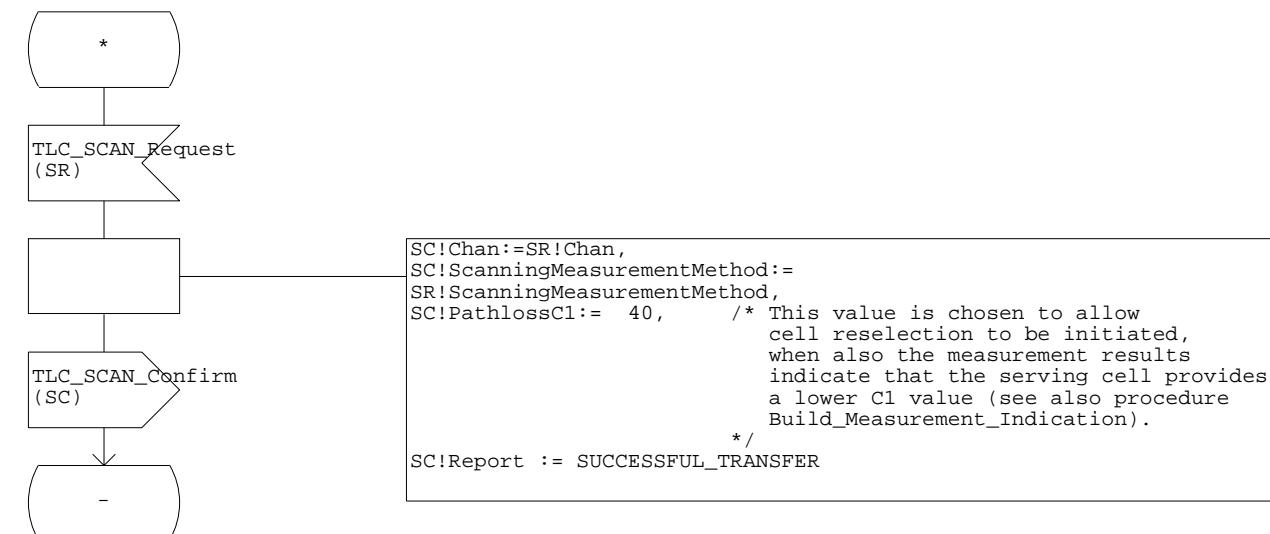
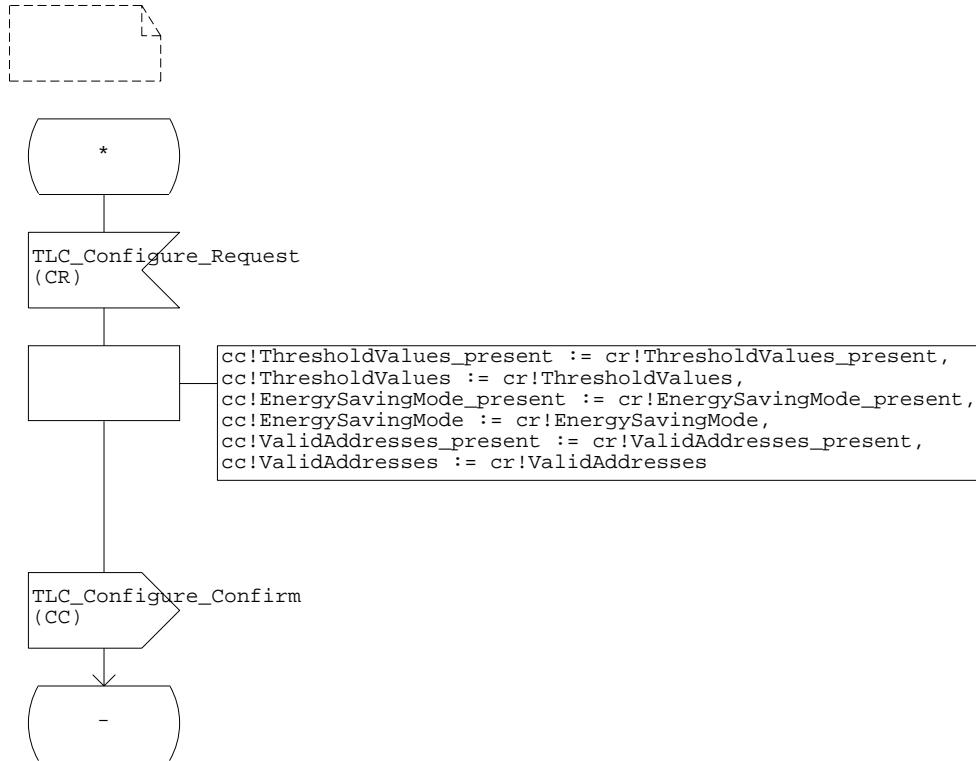
Process MS_DL

3 (15)



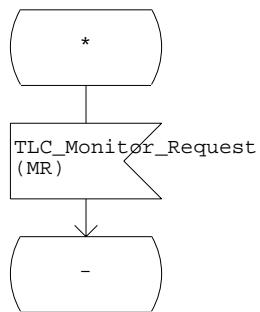
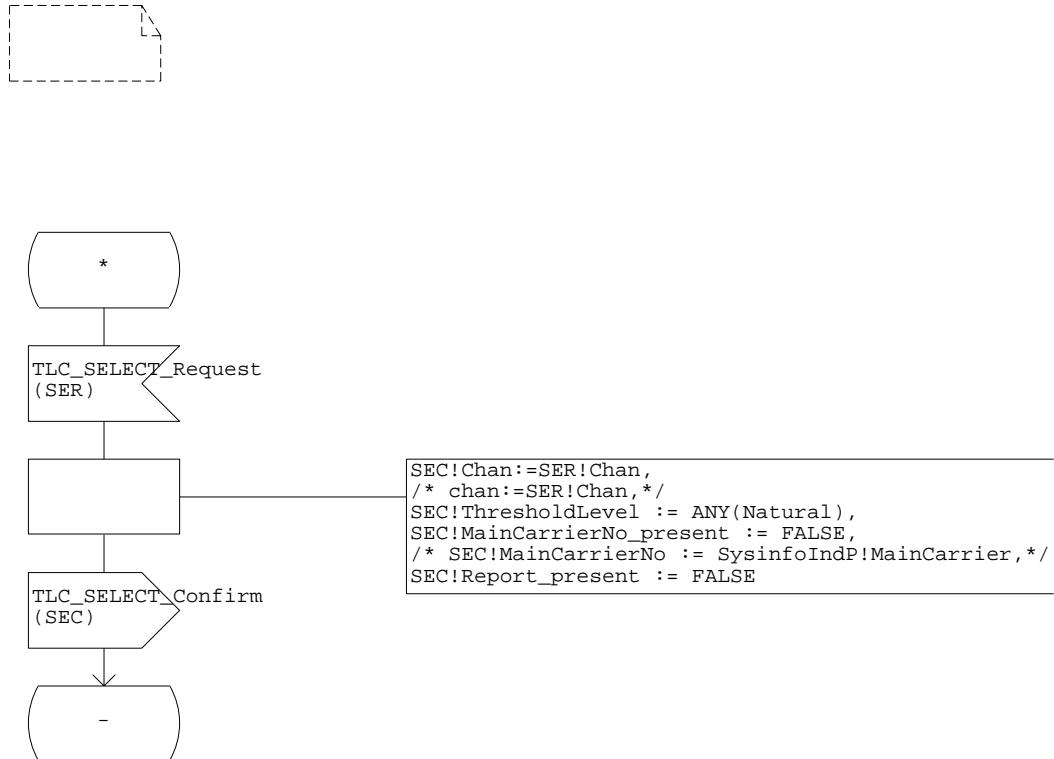
Process MS_DL

4(15)



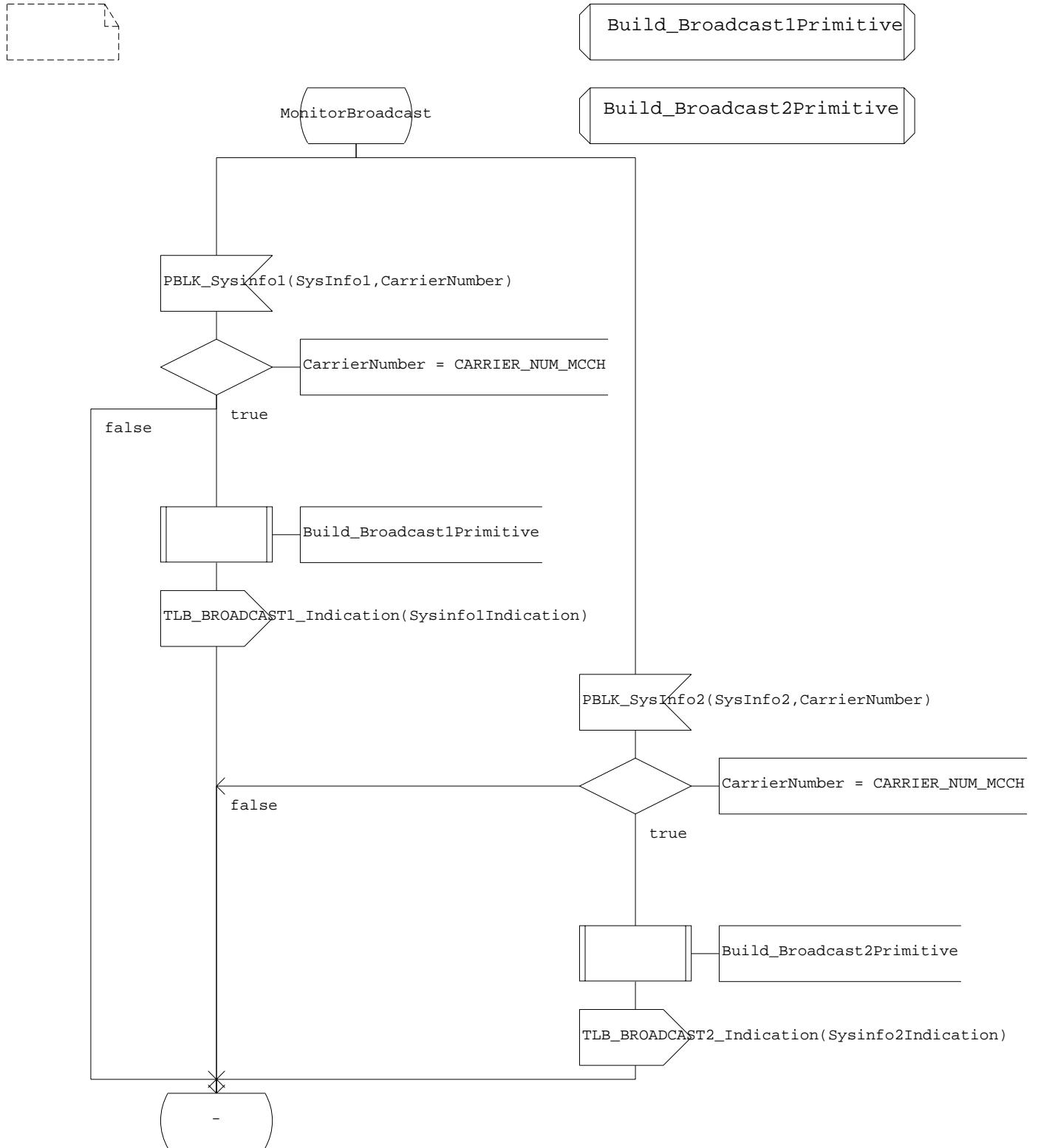
Process MS_DL

5 (15)



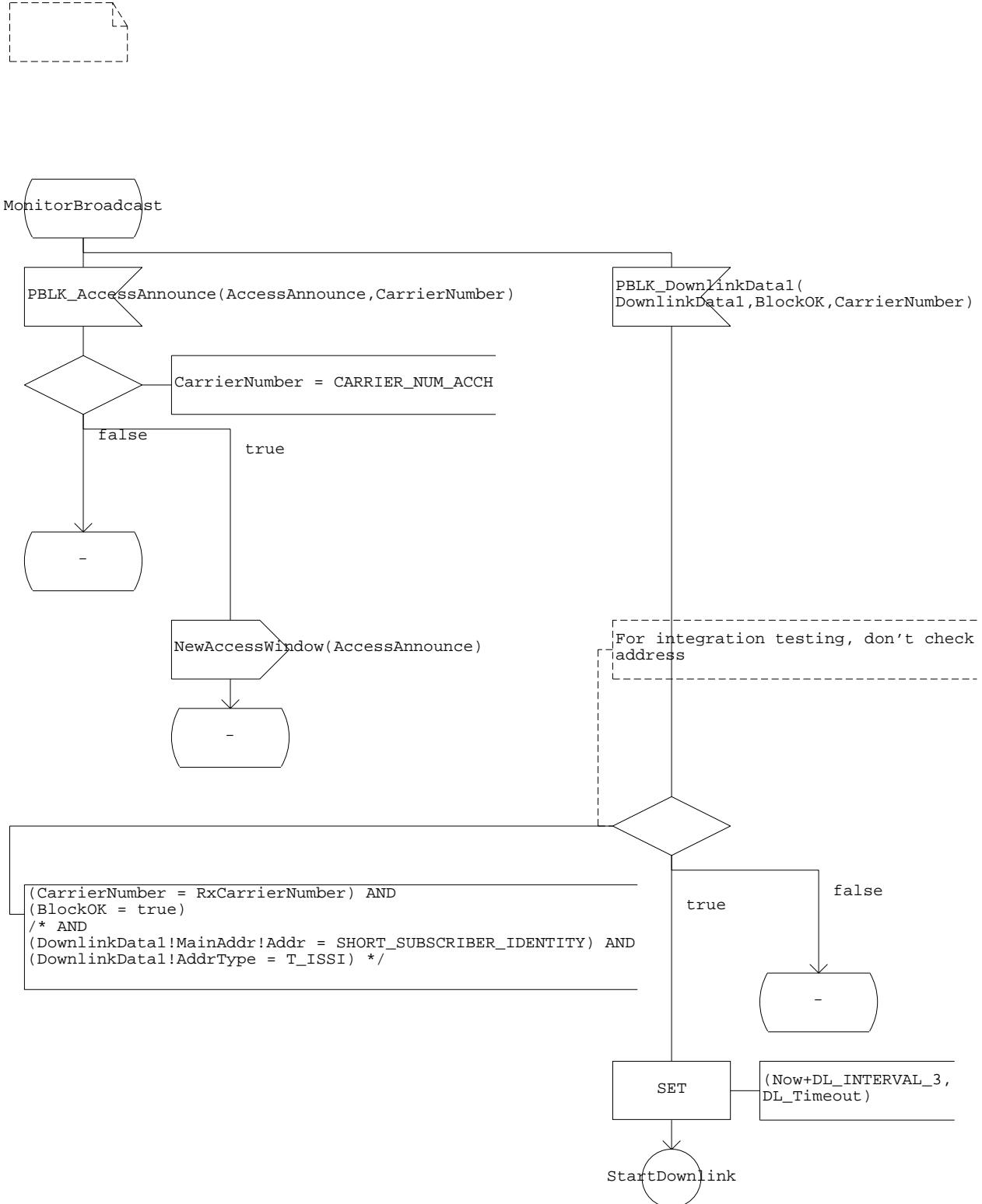
Process MS_DL

6 (15)



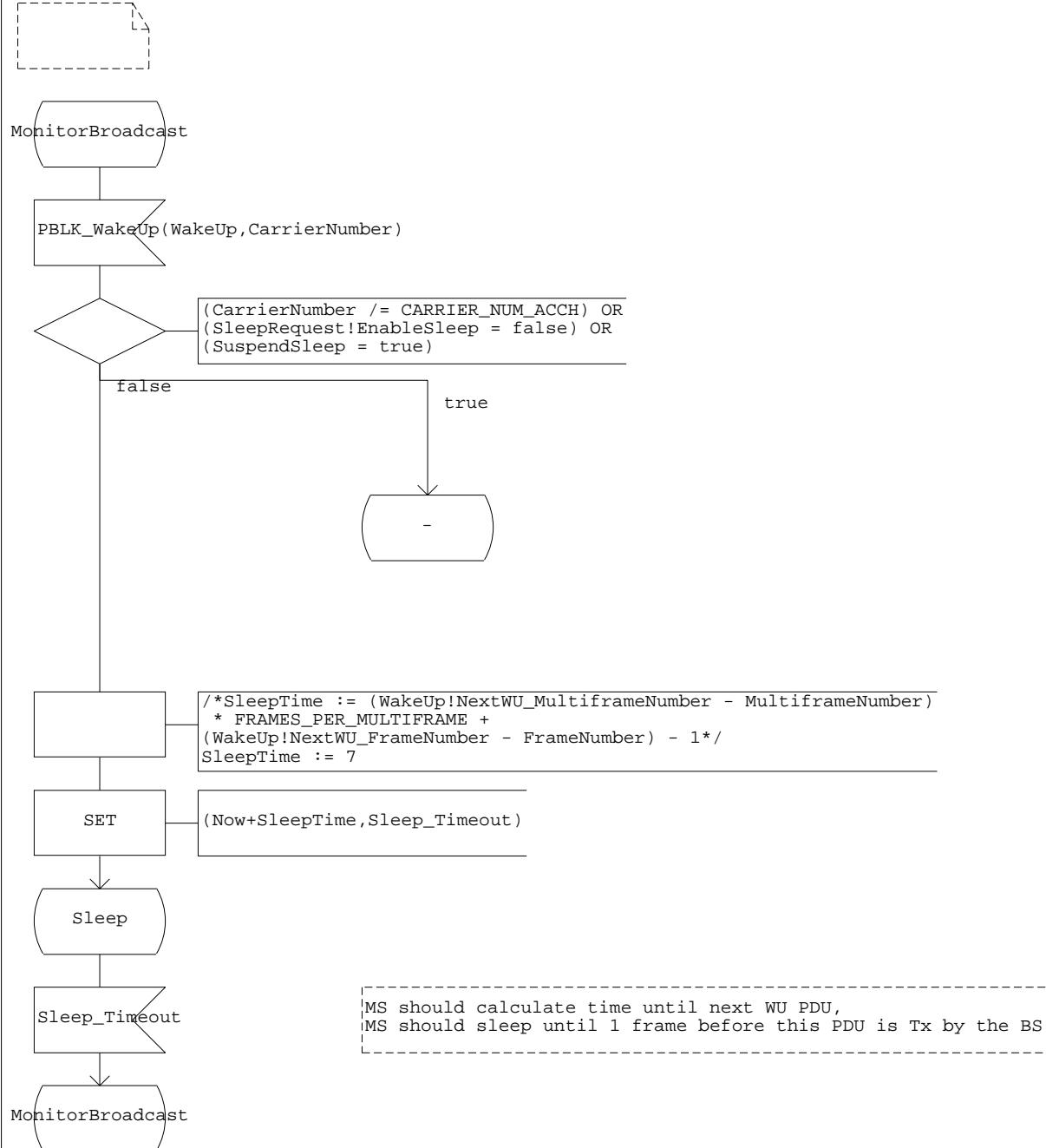
Process MS_DL

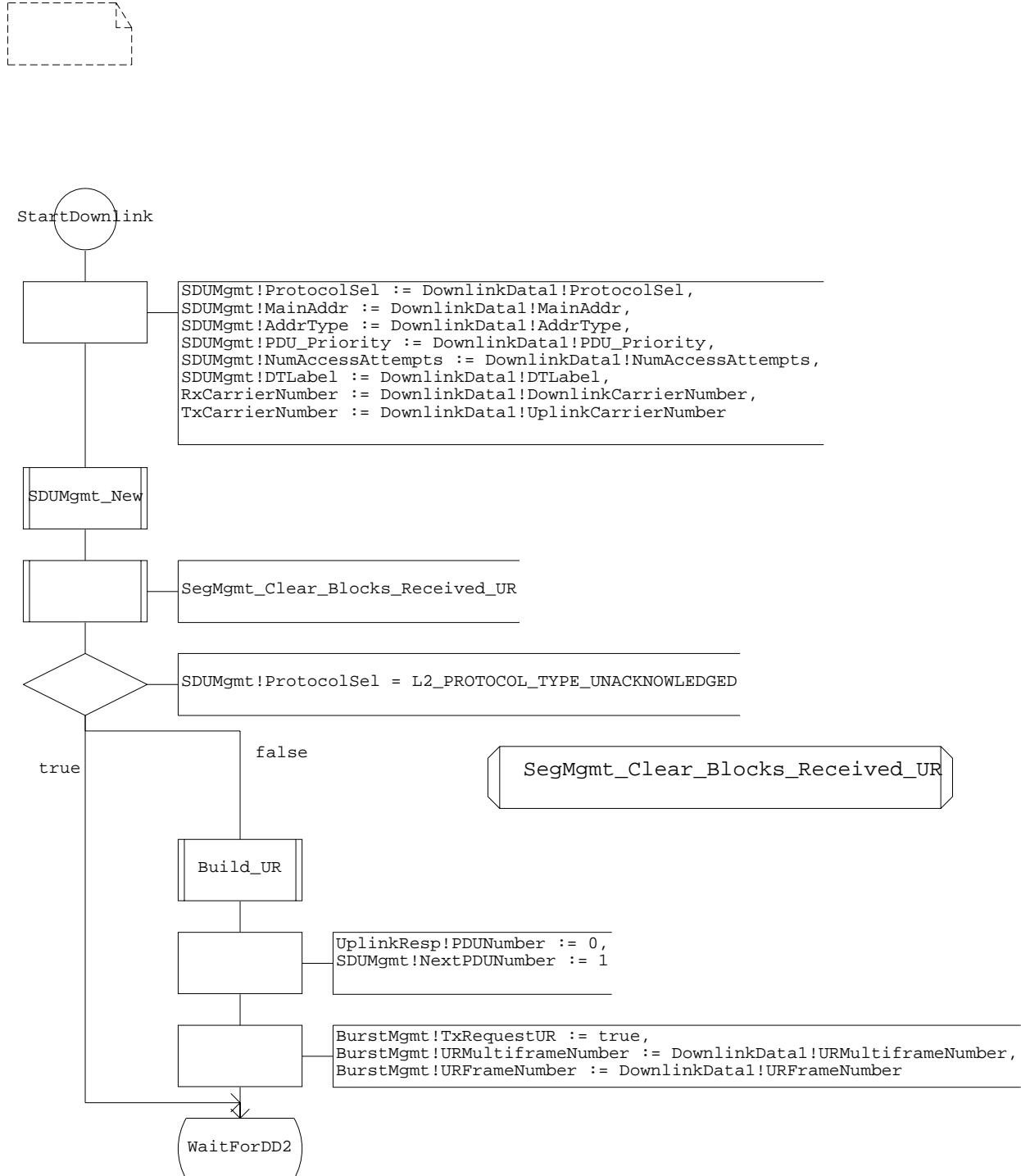
7 (15)



Process MS_DL

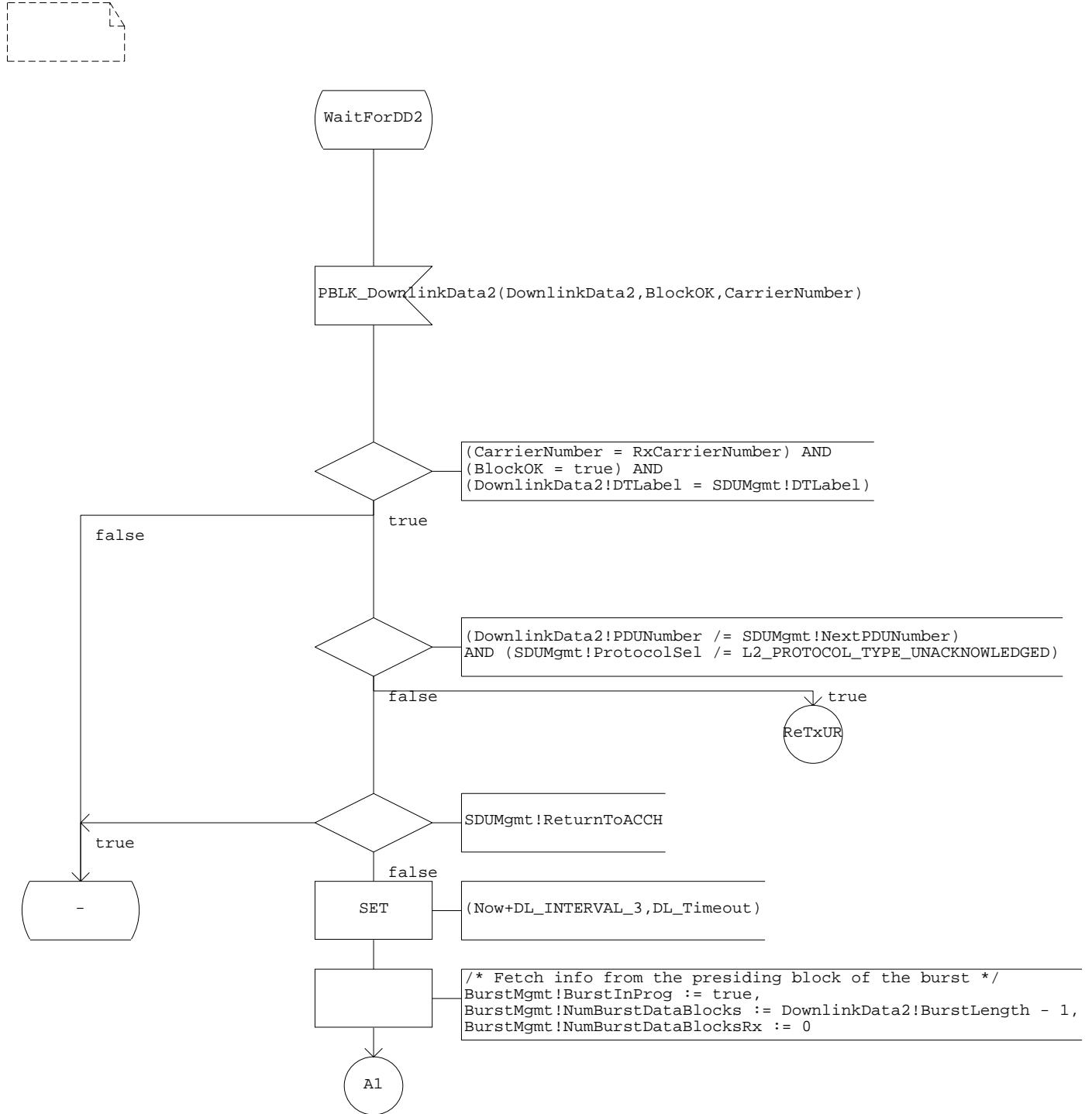
8(15)

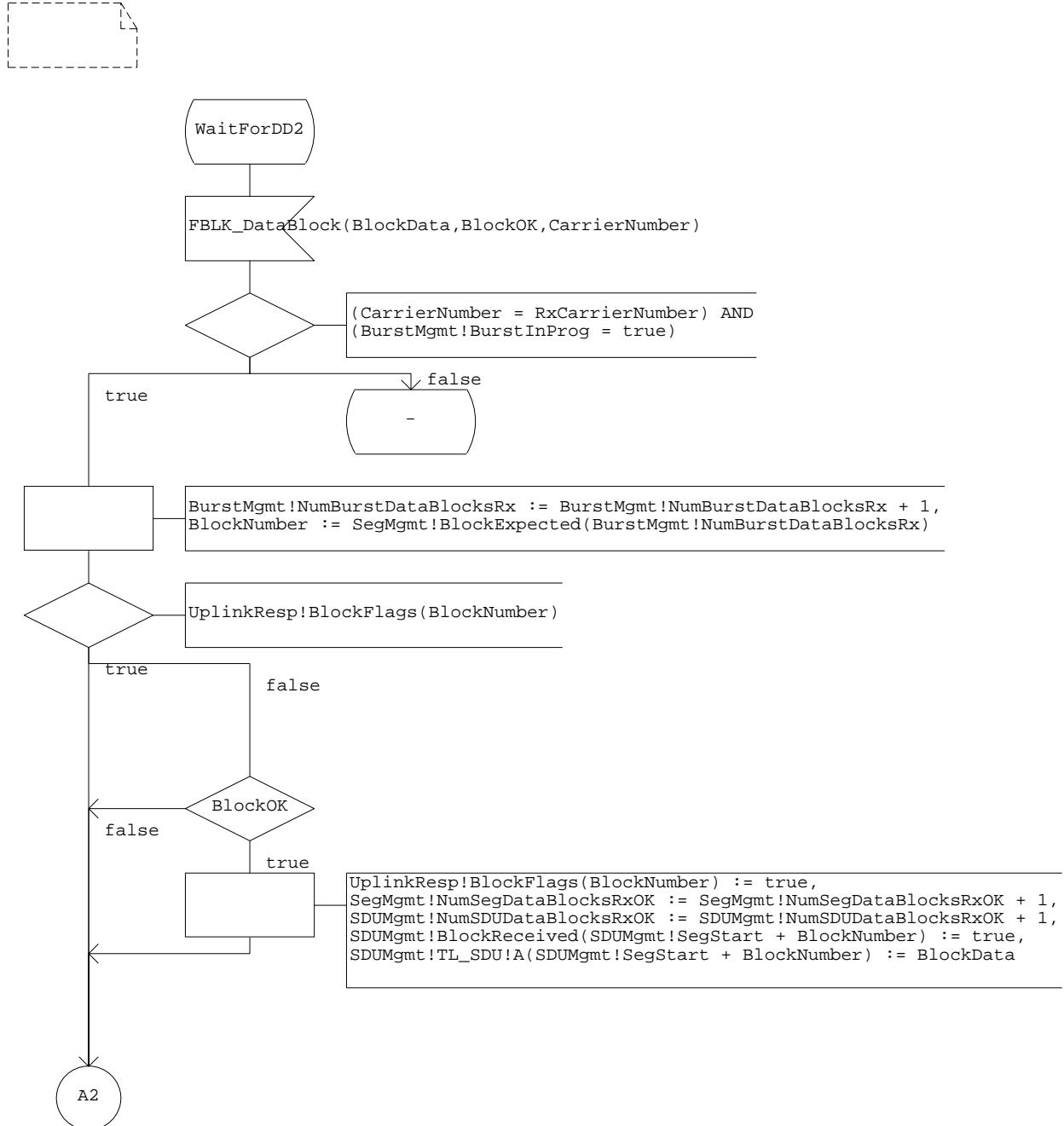


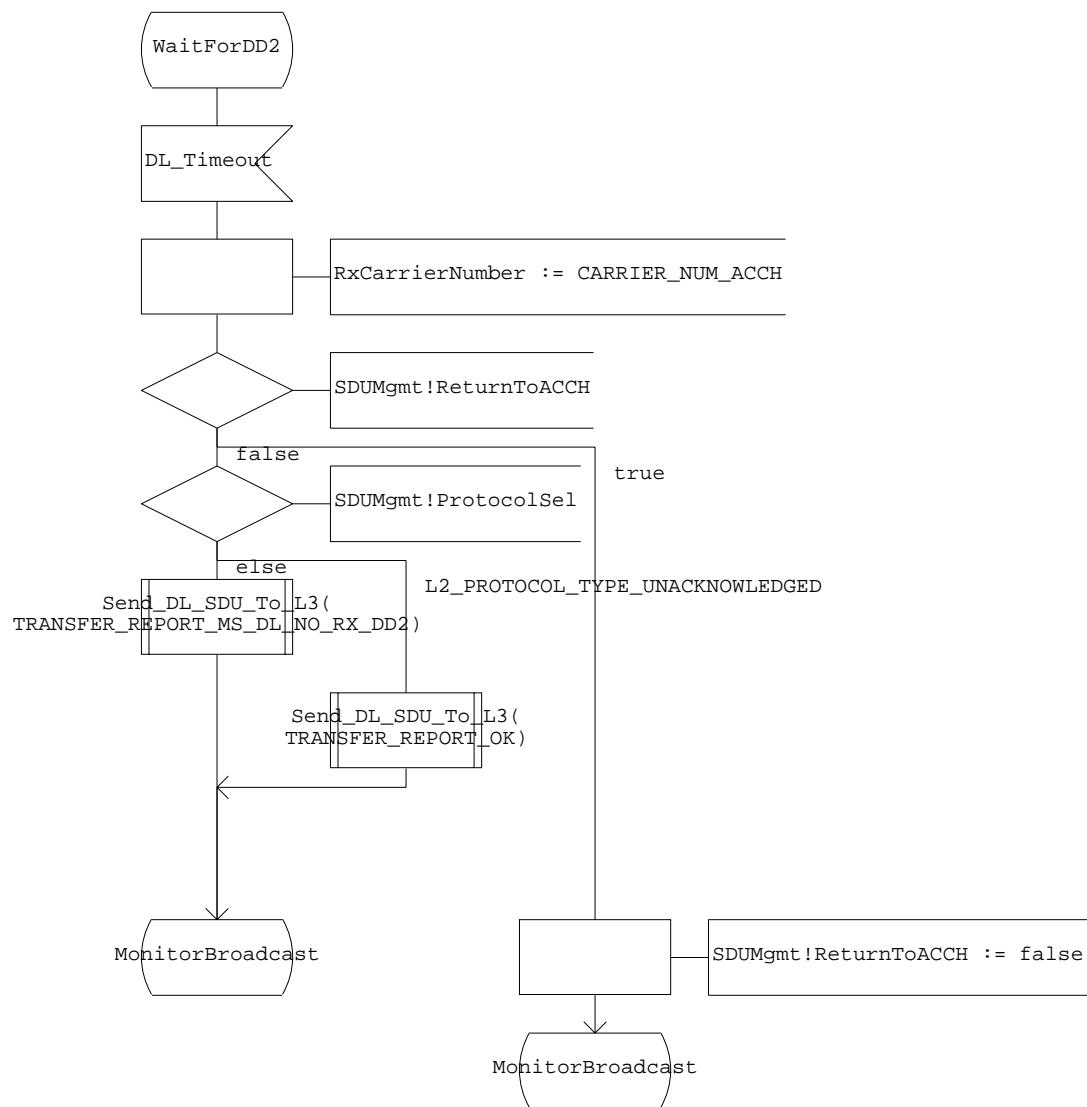


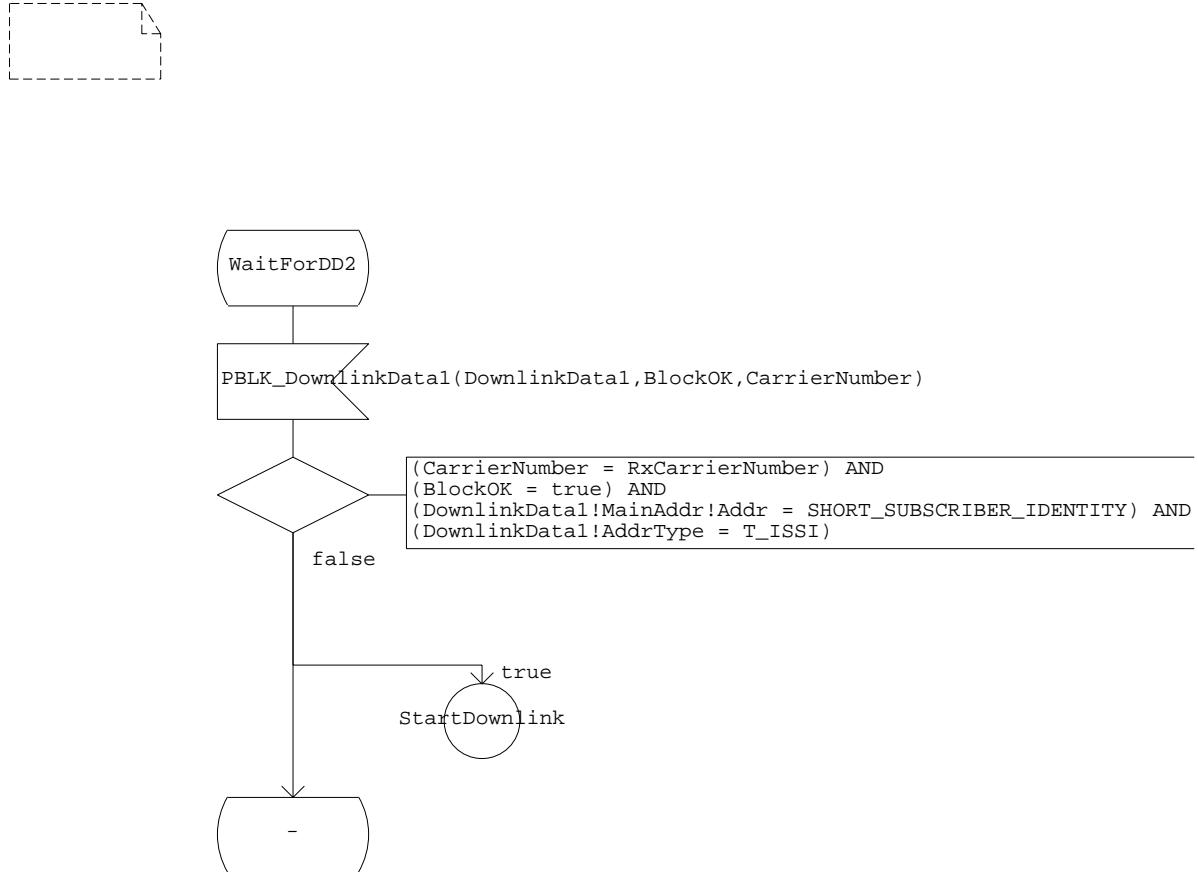
Process MS_DL

10(15)



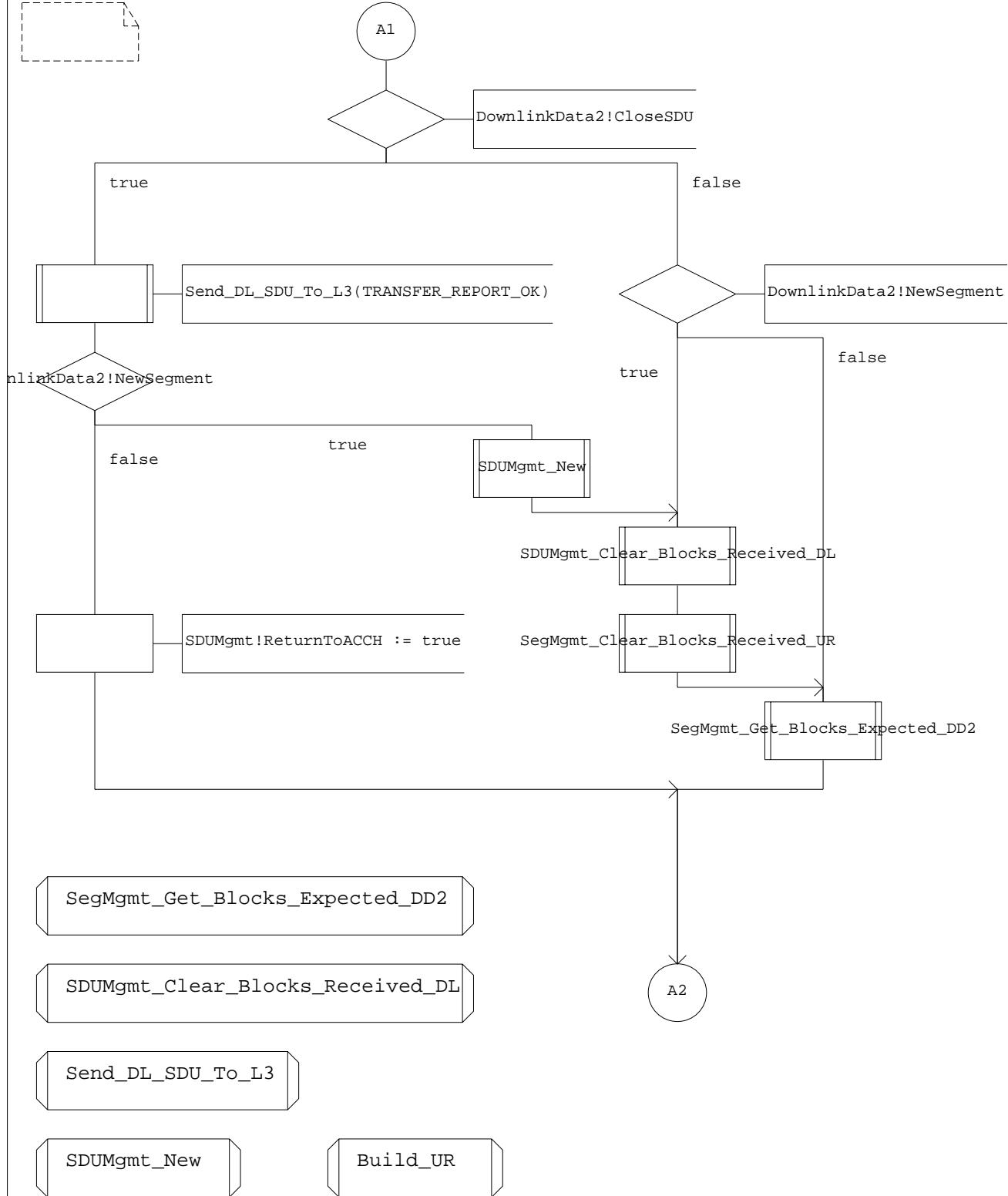


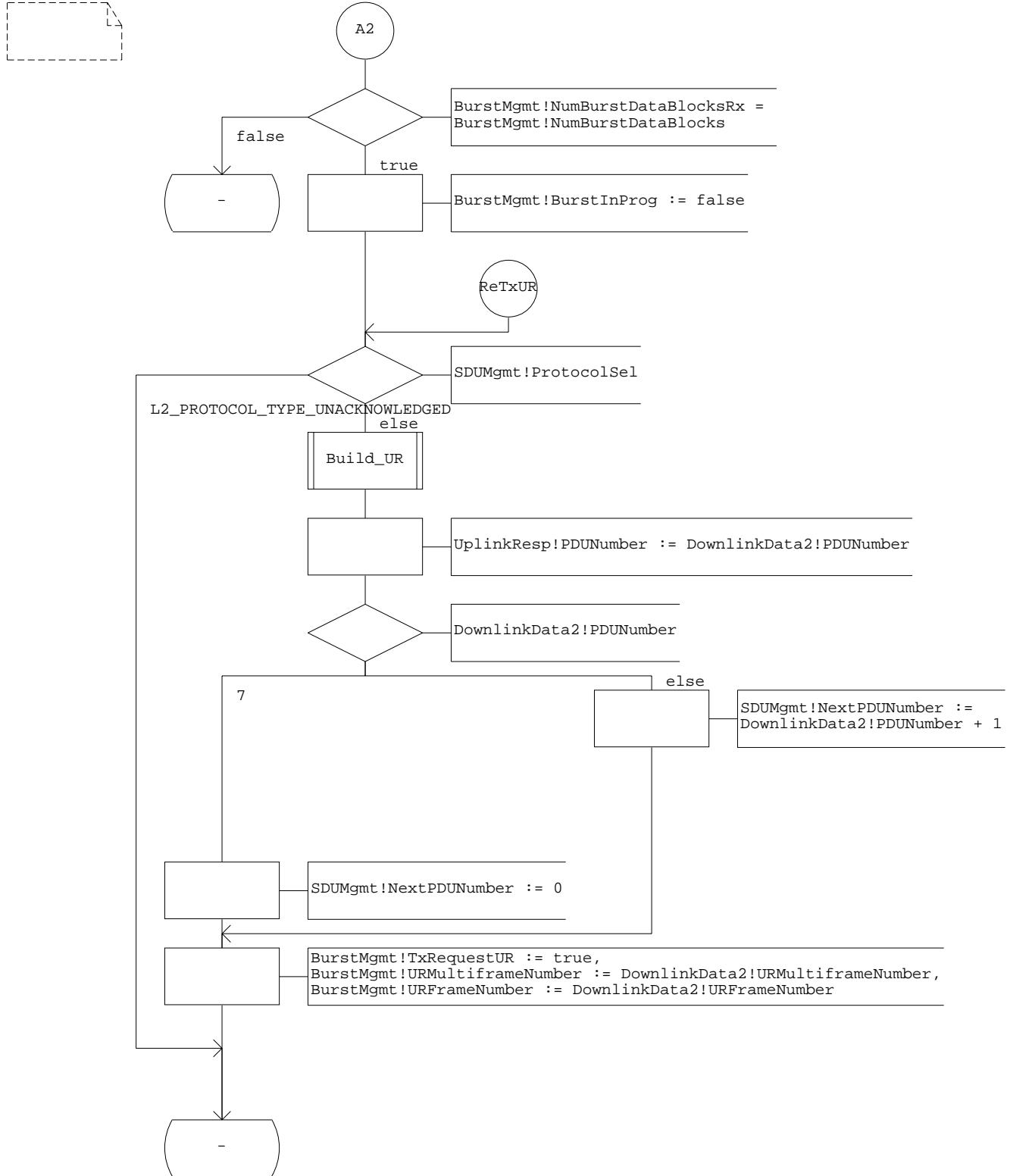




Process MS_DL

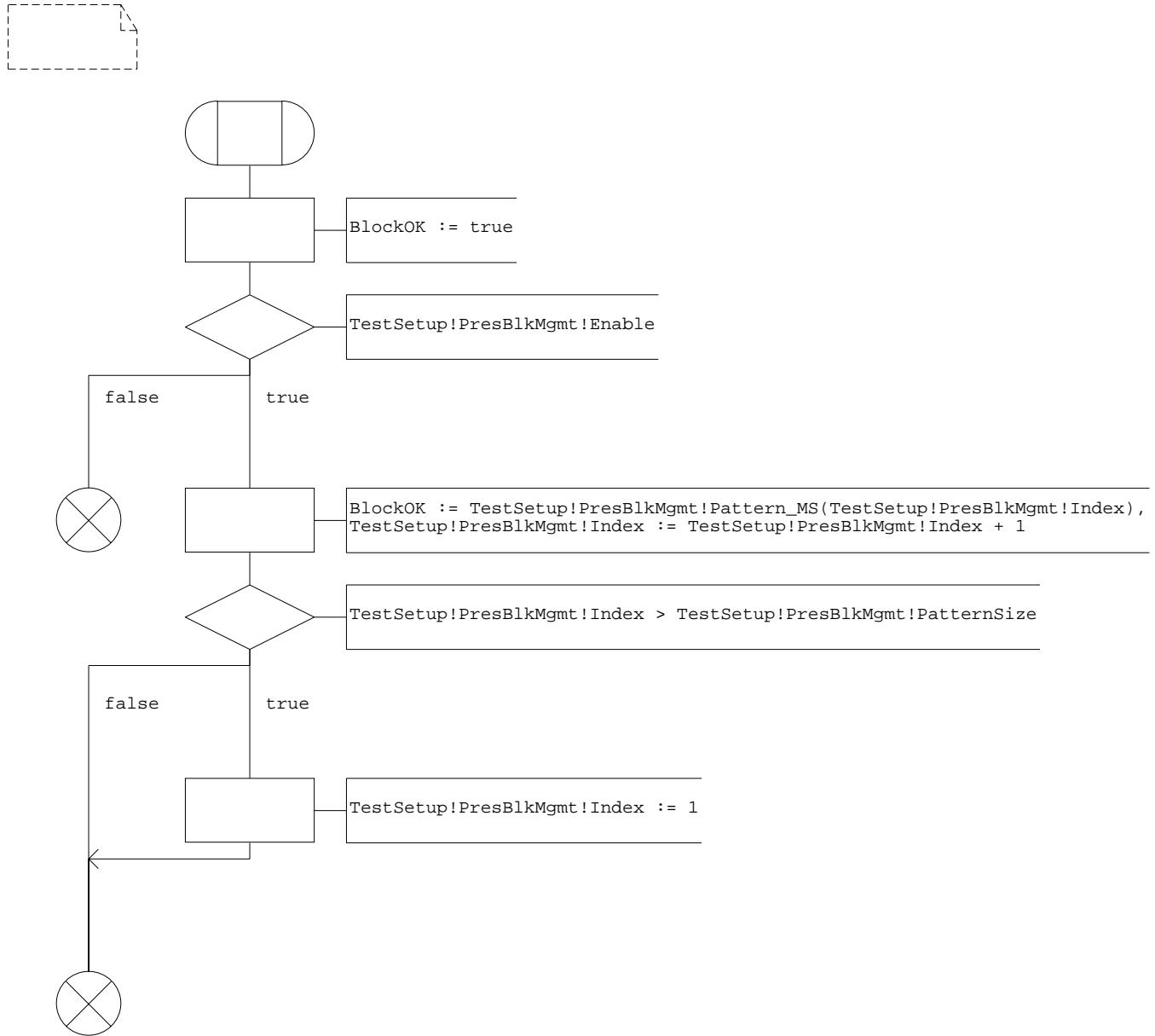
14 (15)





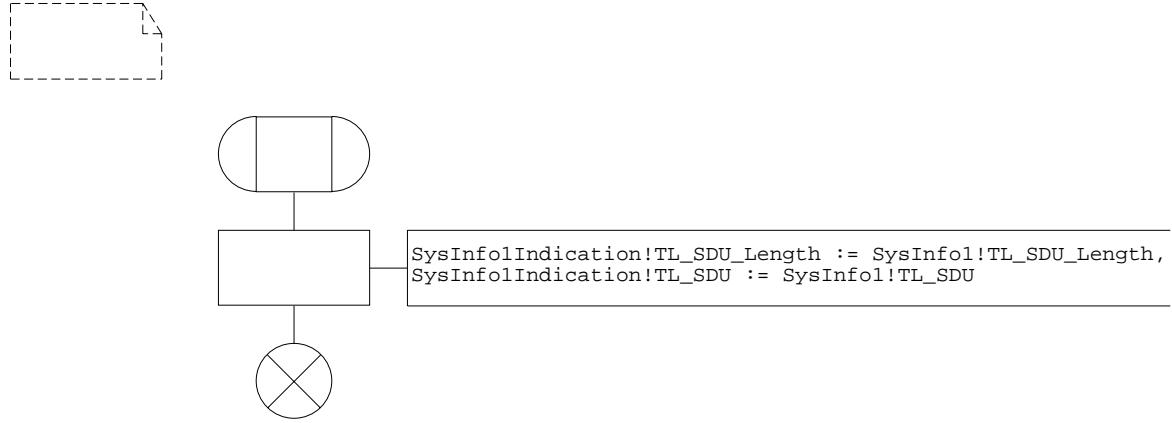
Procedure Bad_Presiding_Block_Gen_MS_DL

1(1)



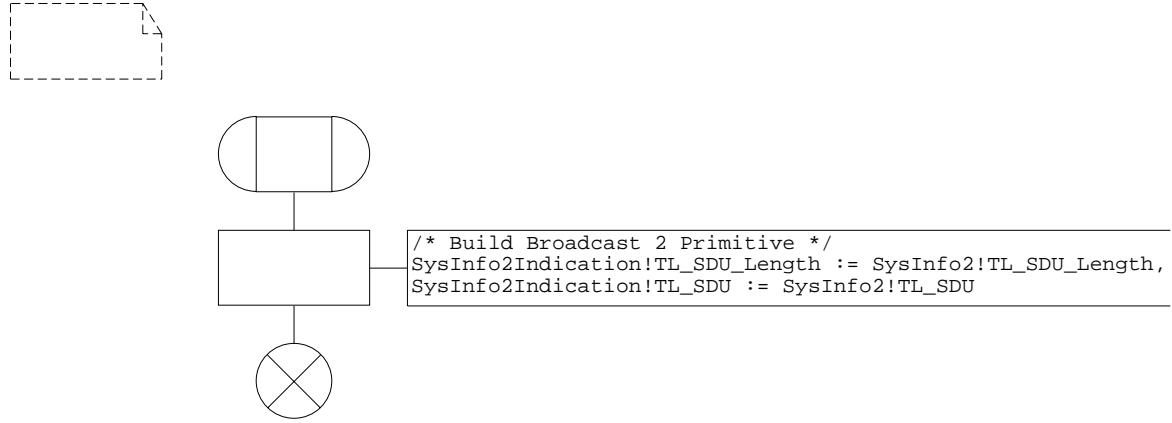
Procedure Build_Broadcast1Primitive

1(1)



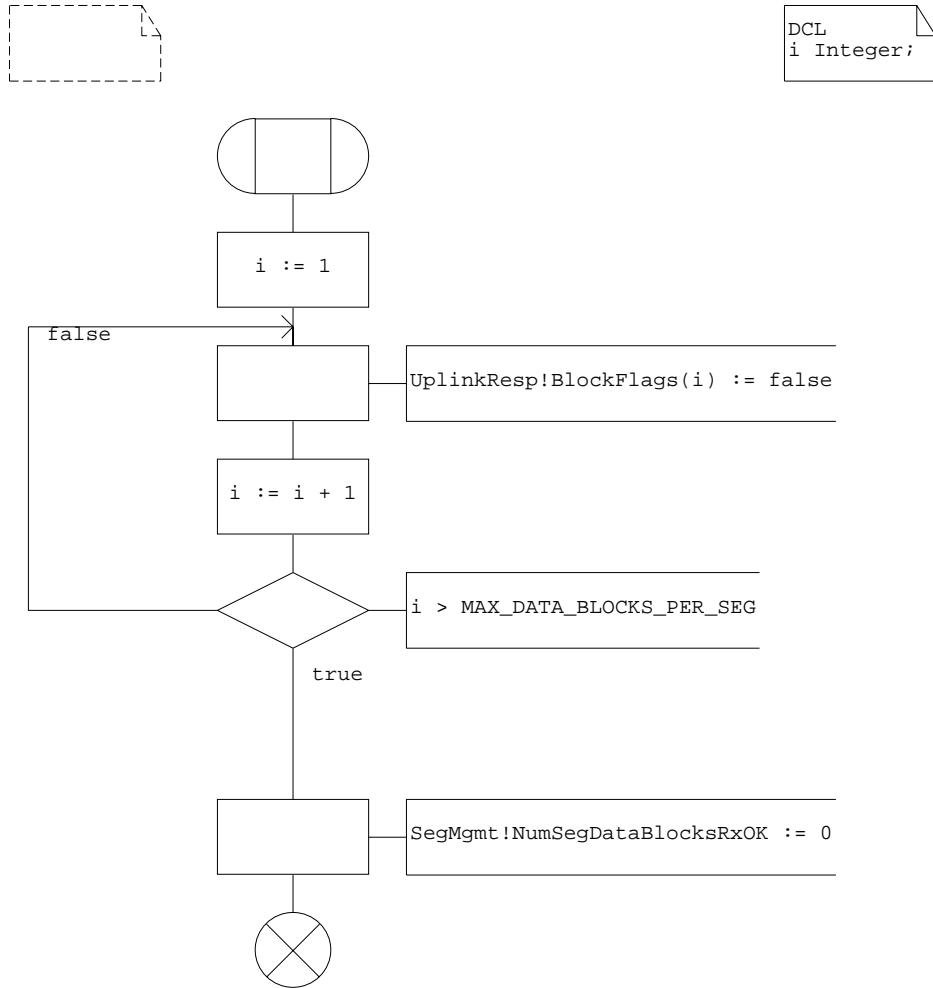
Procedure Build_Broadcast2Primitive

1(1)



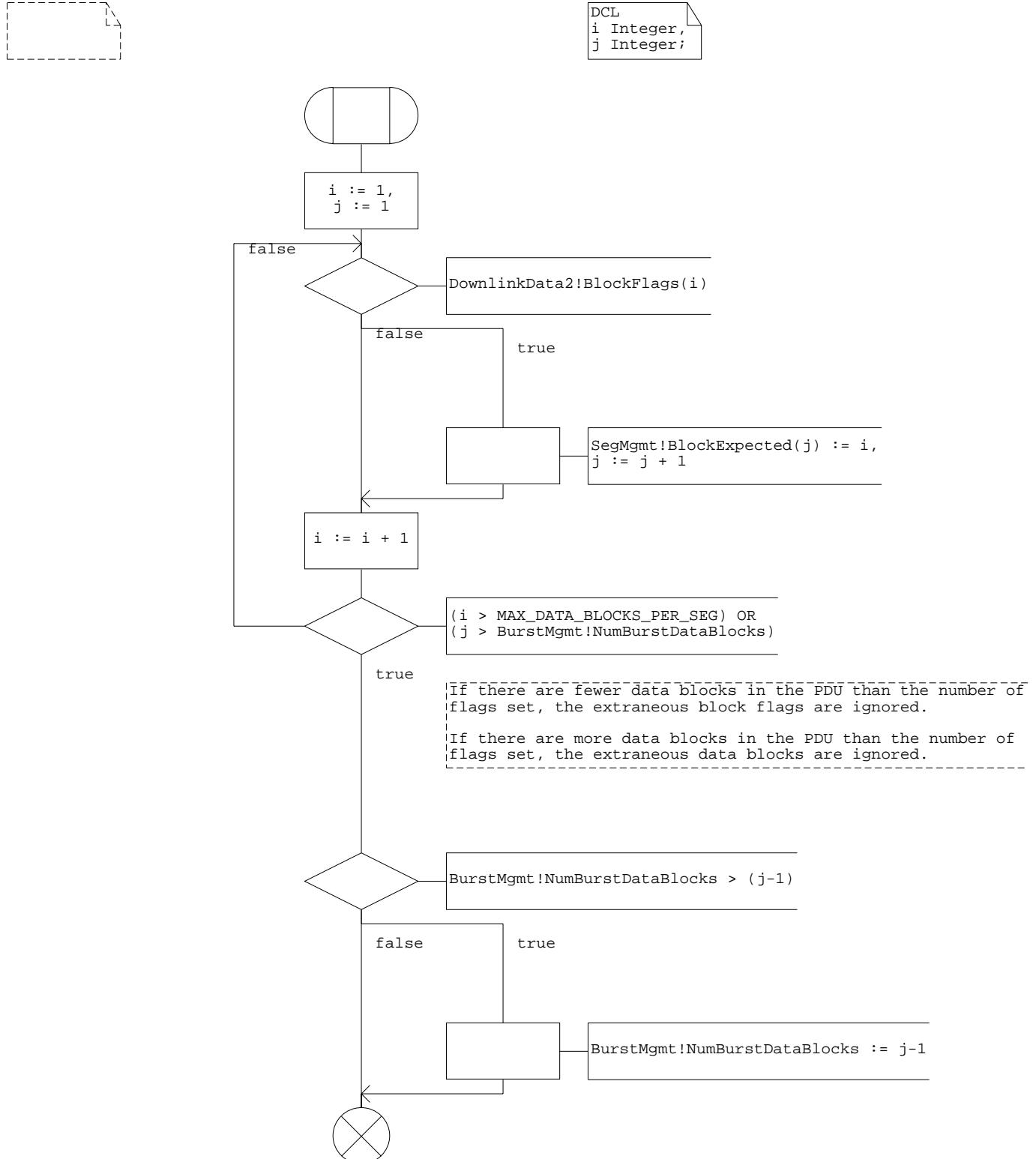
Procedure SegMgmt_Clear_Blocks_Received_UR

1(1)



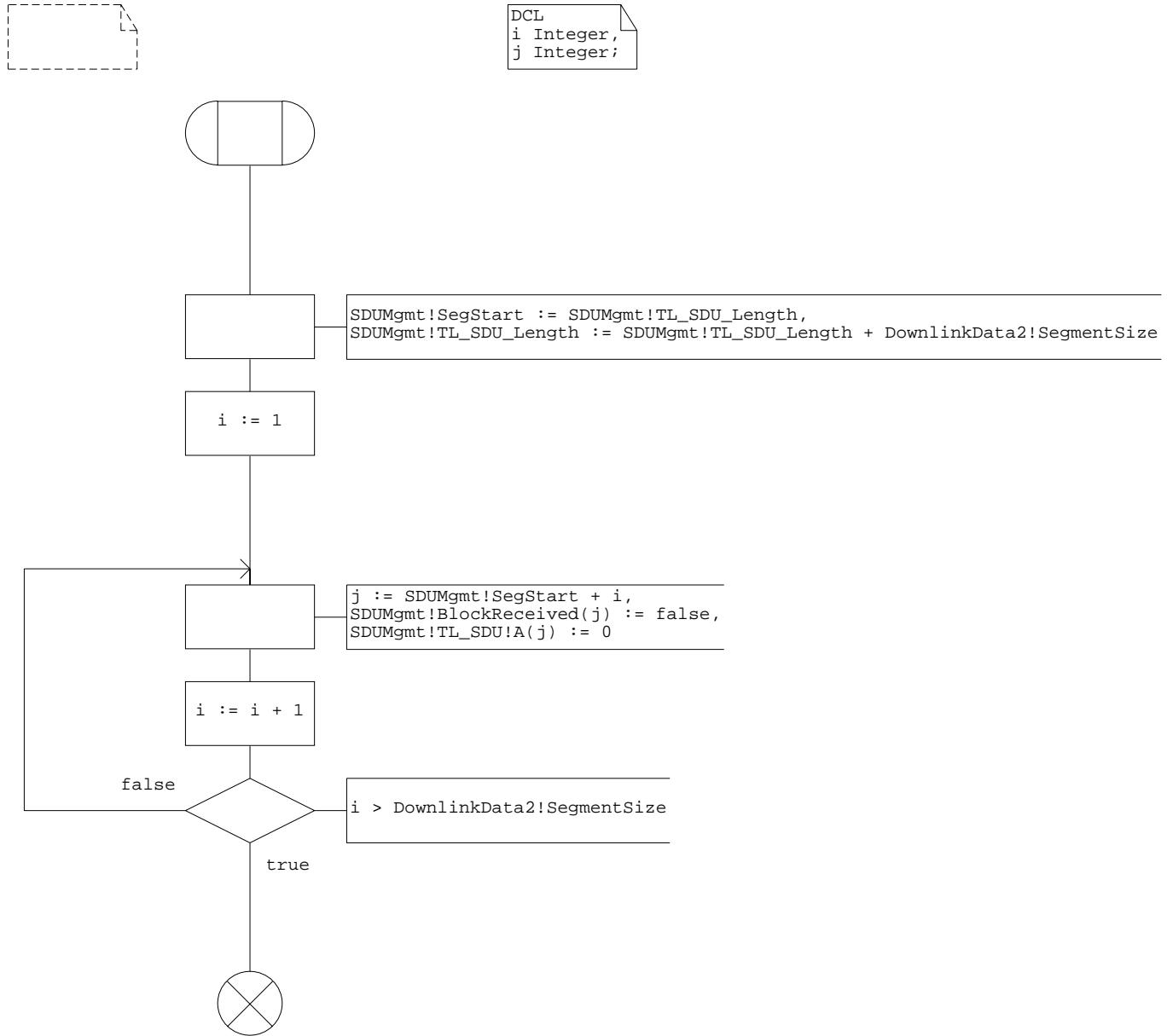
Procedure SegMgmt_Get_Blocks_Expected_DD2

1 (1)



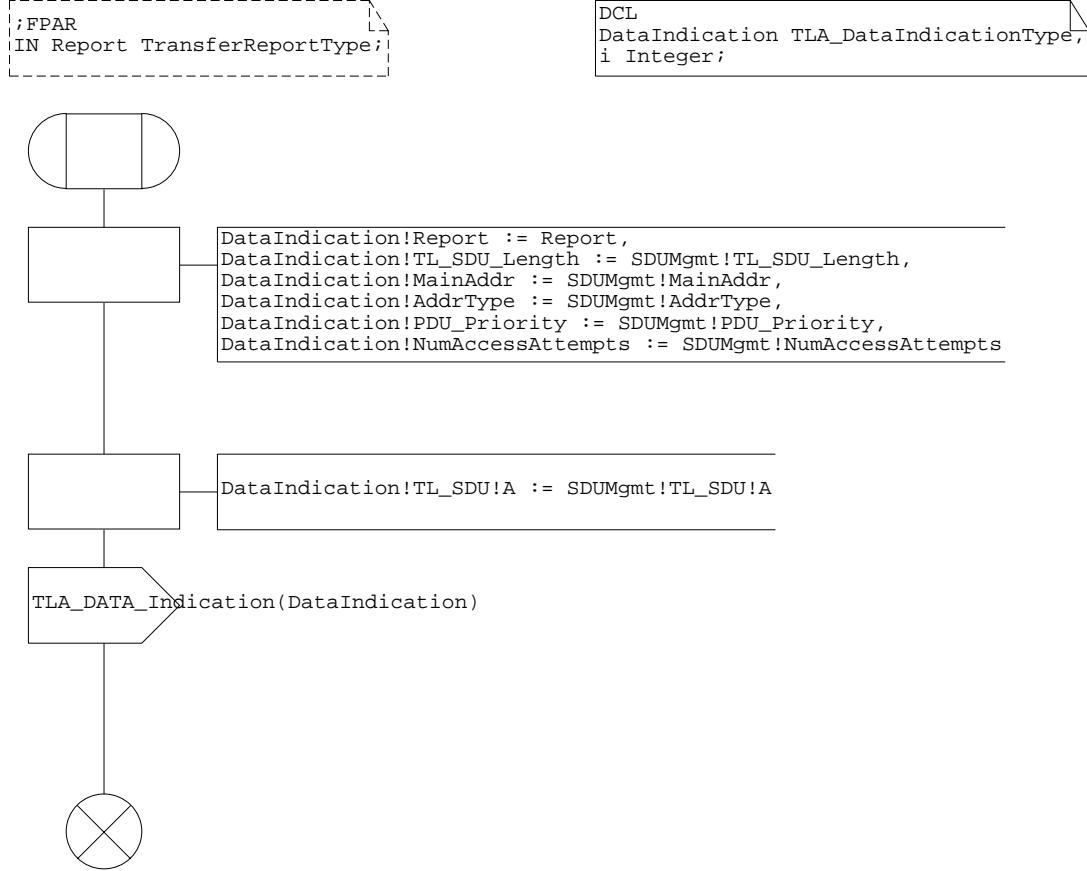
Procedure SDUMgmt_Clear_Blocks_Received_DL

1(1)



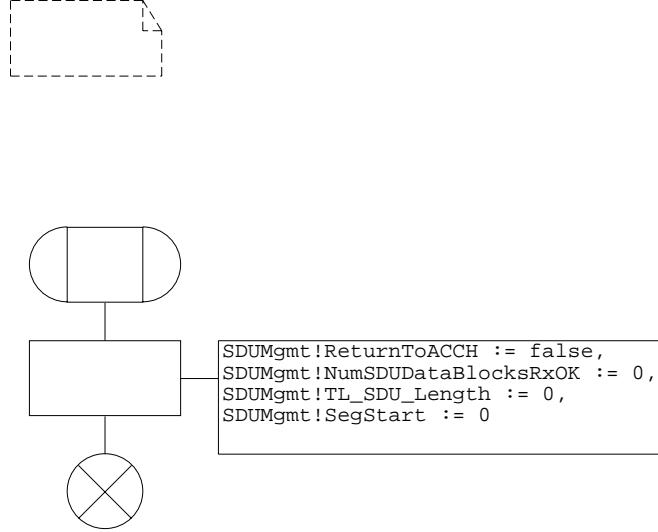
Procedure Send_DL_SDU_To_L3

1(1)



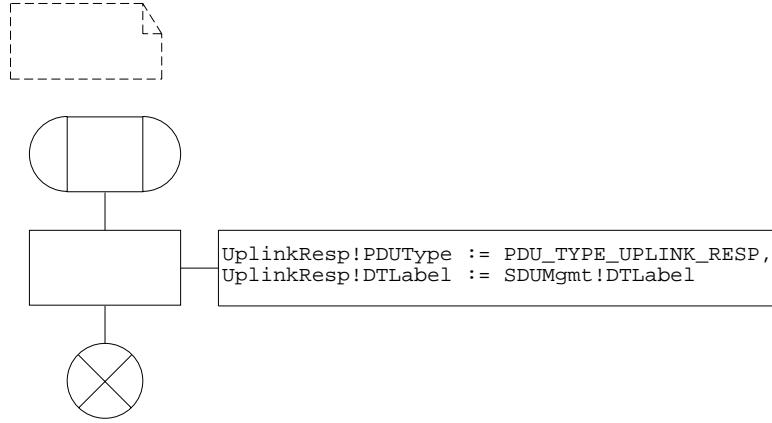
Procedure SDUMgmt_New

1(1)



Procedure Build_UR

1(1)





```

/* Chain Management */
NEWTYPE CHAIN_MGMT STRUCT
    ULabel TrafficLabelType;
    PDUNumber PDUNumberType;
    NextToLastUD2 Boolean;
    LastUD2 Boolean;
    CurEndpointID Endpoint_ID_Type;
    CurProtocolReq Layer2ProtocolRequestType;
ENDNEWTYPE;

/* SDU Management */
NEWTYPE SDU_MGMT STRUCT
    /* given parameters */
    EndpointID Endpoint_ID_Type;
    NumSDUDataBlocks Integer; /* total number of blocks in t
    TL_SDU TL_SDU_Type;
    MainAddr MainAddrType;
    AddrType AddrTypeType;
    PDU_Priority PDU_PriorityType;
    ProtocolReq Layer2ProtocolRequestType;
    MaxNumAccessAttempts AccessAttemptType;
    /* working parameters */
    AccessWindowCountDown Integer;
    NumSDUDataBlocksLeft Integer; /* number of blocks sent (a
    SegStart Integer;
    NumAccessAttempts AccessAttemptType;
ENDNEWTYPE;

/* Segment Management */
NEWTYPE SEG_MGMT STRUCT
    NumSegDataBlocks Integer;
    NumSegDataBlocksLeft Integer;
    BlockAck BLOCK_ARRAY;
    BlockExpected BLOCK_NUMBER_ARRAY;
ENDNEWTYPE;

/* PDU Burst Management */
NEWTYPE BURST_MGMT STRUCT
    TxRequestUD1 Boolean;
    TxRequestUD2 Boolean;
    UDMultiframeNumber Integer;
    UDFrameNumber Integer;
    UD2TxInProg Boolean;
    NumBurstBlocks Integer; /* total number of blocks in thi
    NumBurstBlocksTx Integer; /* running counter */
ENDNEWTYPE;

```

```

DCL
MultiframeNumber Integer,
FrameNumber Integer,

ChainMgmt CHAIN_MGMT,
SDUArrMgmt SDU_ARRAY_MGMT,
SDUMgmt SDU_MGMT_ARRAY,
SeqMgmt SEG_MGMT,
BurstMgmt BURST_MGMT,

TestSetup TLC_TestSetupRequestType,
BlockOK Boolean,
BlockData Integer,
CarrierNumber CarrierNumberType,
TxCarrierNumber CarrierNumberType,
RxCarrierNumber CarrierNumberType,
UD1MultiframeNumber Integer,
UD1FrameNumber Integer,
i Integer,
ti Integer,
k Integer;

```

```

DCL
AccessAnnounce ACCESS_ANNOUNCE_PDU,
UplinkData1 UPLINK_DATA_1_PDU,
DownlinkResp1 DOWNLINK_RESP_1_PDU,
UplinkData2 UPLINK_DATA_2_PDU,
DownlinkResp2 DOWNLINK_RESP_2_PDU;

```

```

DCL
DataRequest TLA_DataRequestType,
DataConfirm TLA_DataConfirmType,
DataCancel TLA_CancelRequestType;

```

```

DCL
RandomAccessSeq RandomControl;

```

```

NEWTYPE SDU_MGMT_ARRAY
    Array(Integer,SDU_MGMT)
ENDNEWTYPE SDU_MGMT_ARRAY;

```

```

Timer
DL_Timeout,
BS_Cancel;

```

```

/* Chain Management */
NEWTYPE CHAIN_MGMT STRUCT
    UTLabel TrafficLabelType;
    PDUNumber PDUNumberType;
    NextToLastUD2 Boolean;
    LastUD2 Boolean;
    CurEndpointID Endpoint_ID_Type;
    CurProtocolReq Layer2ProtocolRequestType;
ENDNEWTYPE;

/* SDU Management */
NEWTYPE SDU_MGMT STRUCT
    /* given parameters */
    EndpointID Endpoint_ID_Type;
    NumSDUDataBlocks Integer; /* total number of blocks in the SDU */
    TL_SDU TL_SDU_Type;
    MainAddr MainAddrType;
    AddrType AddrTypeType;
    PDU_Priority PDU_PriorityType;
    ProtocolReq Layer2ProtocolRequestType;
    MaxNumAccessAttempts AccessAttemptType;
    /* working parameters */
    AccessWindowCountDown Integer;
    NumSDUDataBlocksLeft Integer; /* number of blocks sent (and ACK'd) */
    SegStart Integer;
    NumAccessAttempts AccessAttemptType;
ENDNEWTYPE;

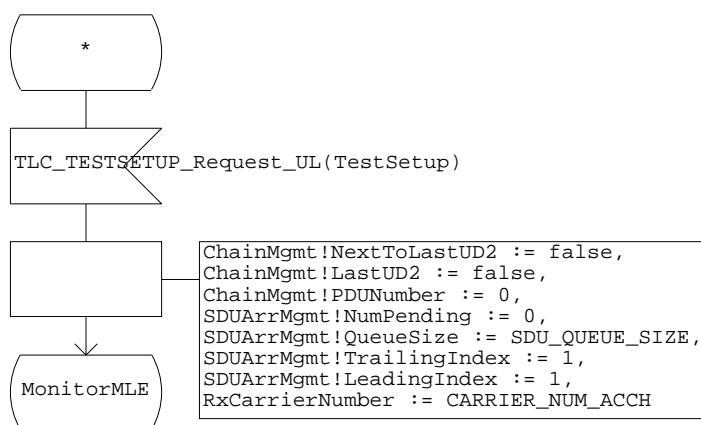
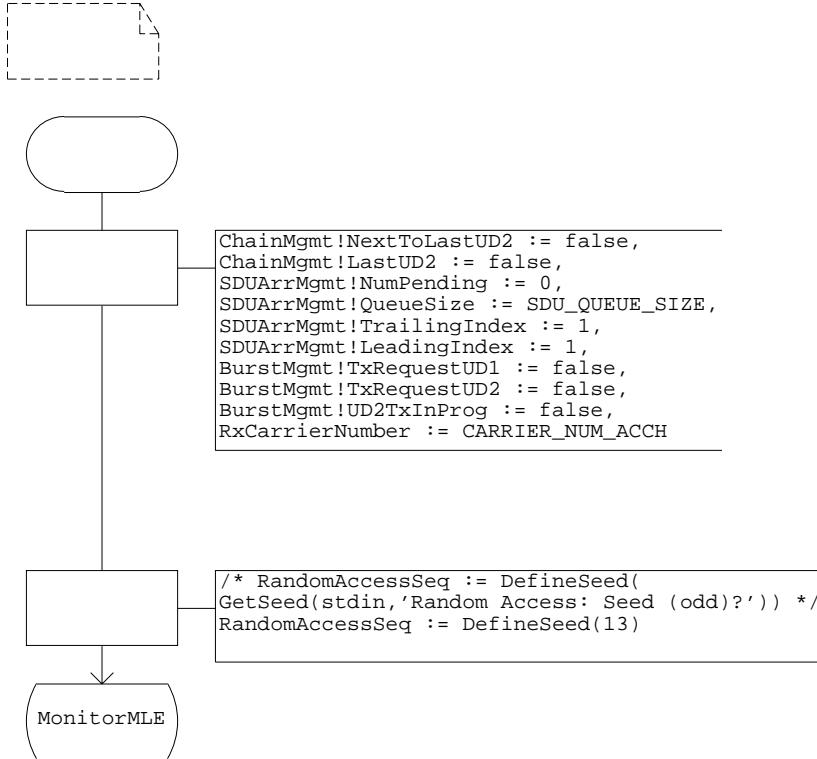
/* Segment Management */
NEWTYPE SEG_MGMT STRUCT
    NumSegDataBlocks Integer;
    NumSegDataBlocksLeft Integer;
    BlockAck BLOCK_ARRAY;
    BlockExpected BLOCK_NUMBER_ARRAY;
ENDNEWTYPE;

/* PDU Burst Management */
NEWTYPE BURST_MGMT STRUCT
    TxRequestUD1 Boolean;
    TxRequestUD2 Boolean;
    UDMultiframeNumber Integer;
    UDFrameNumber Integer;
    UD2TxInProg Boolean;
    NumBurstBlocks Integer; /* total number of blocks in this burst */
    NumBurstBlocksTx Integer; /* running counter */
ENDNEWTYPE;

```

Process MS_UL

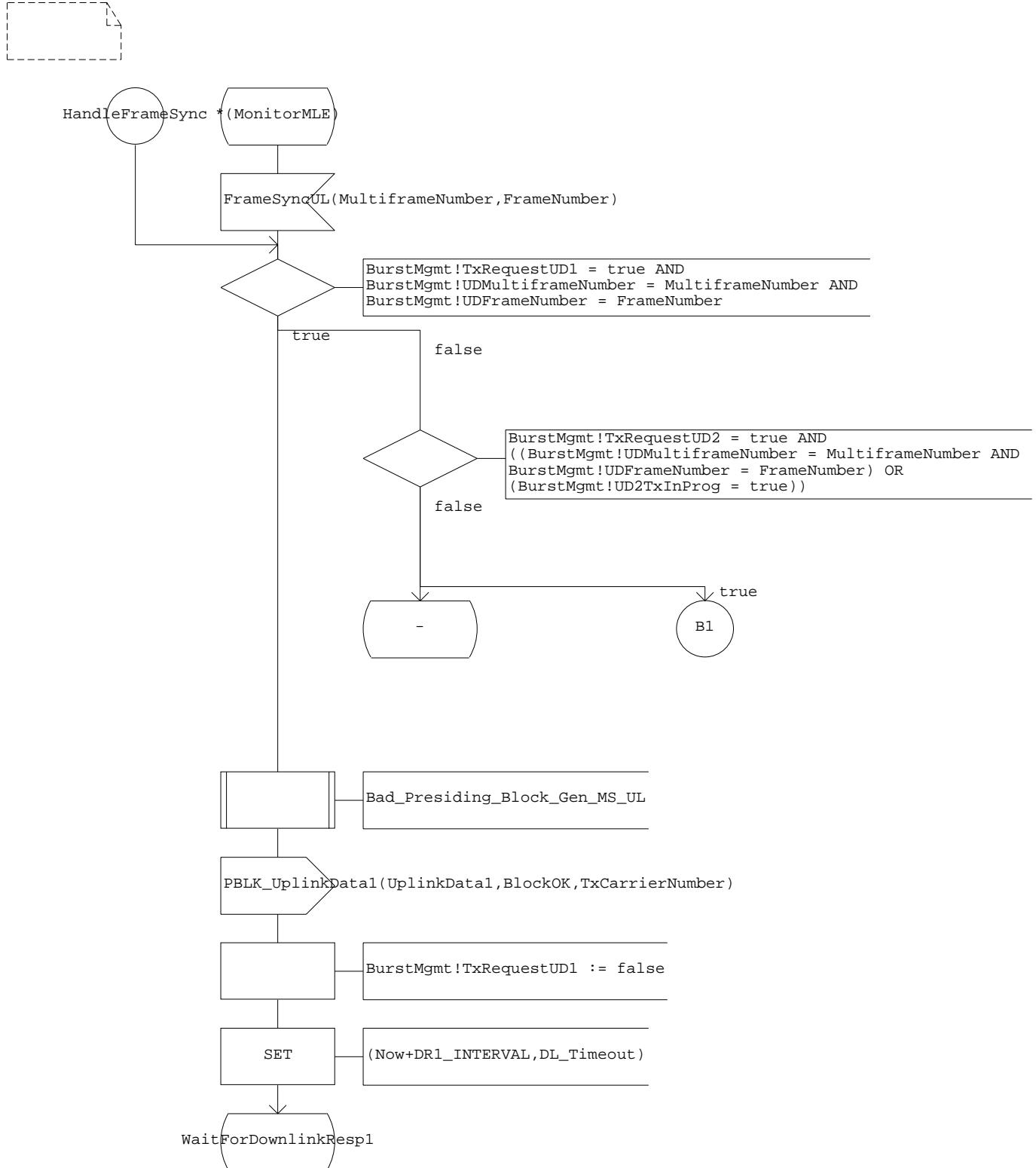
2(15)

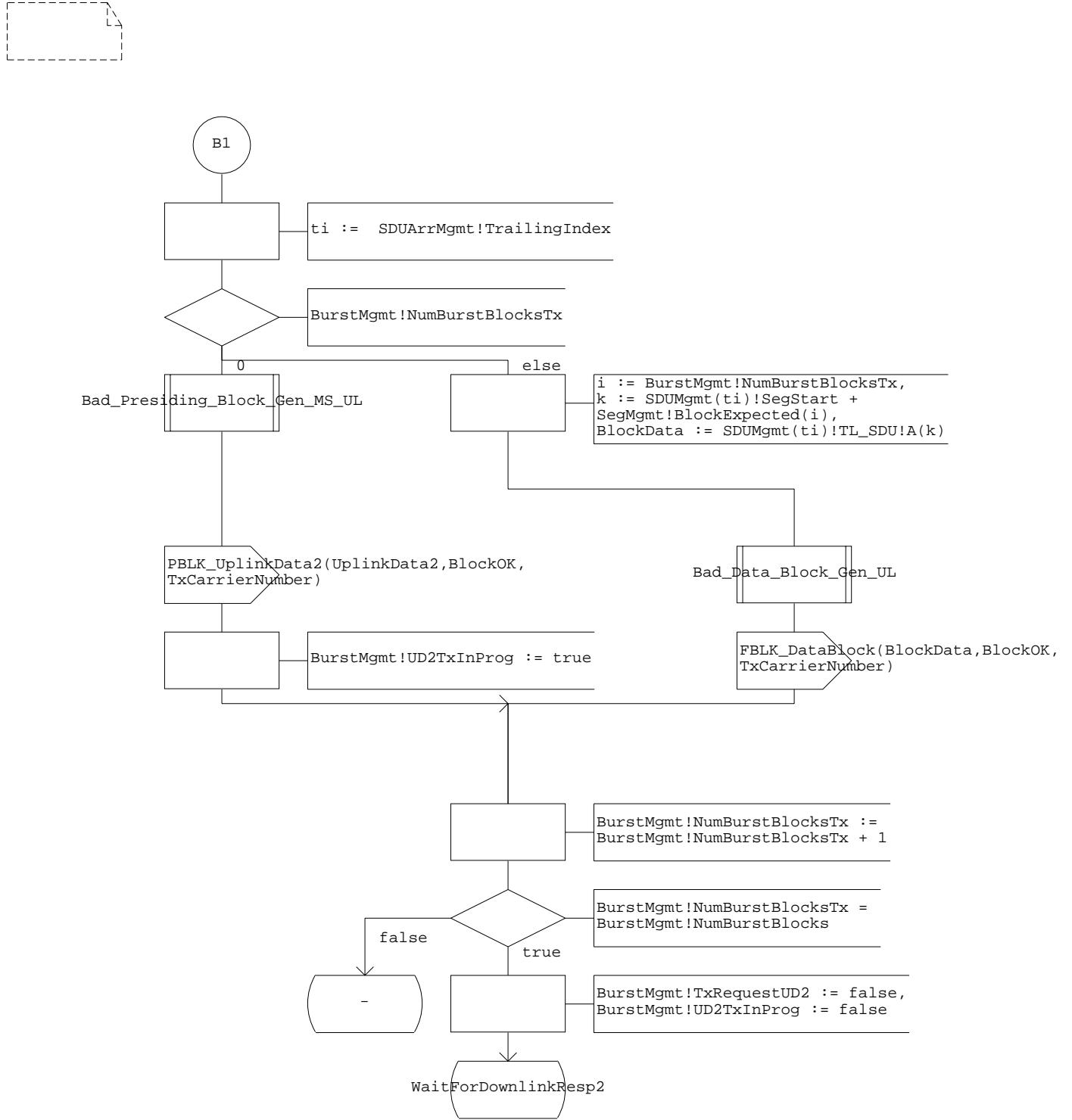


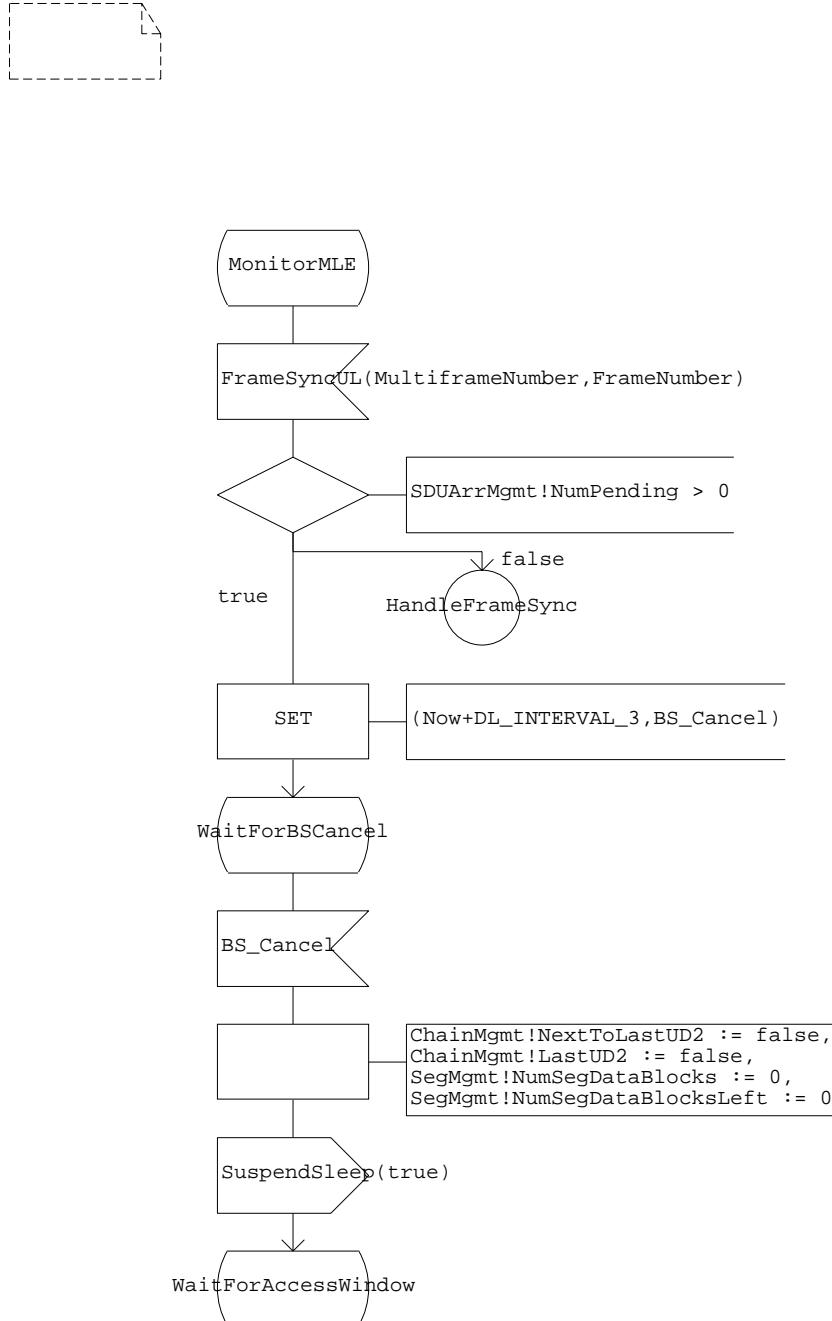
Bad_Data_Block_Gen_UL

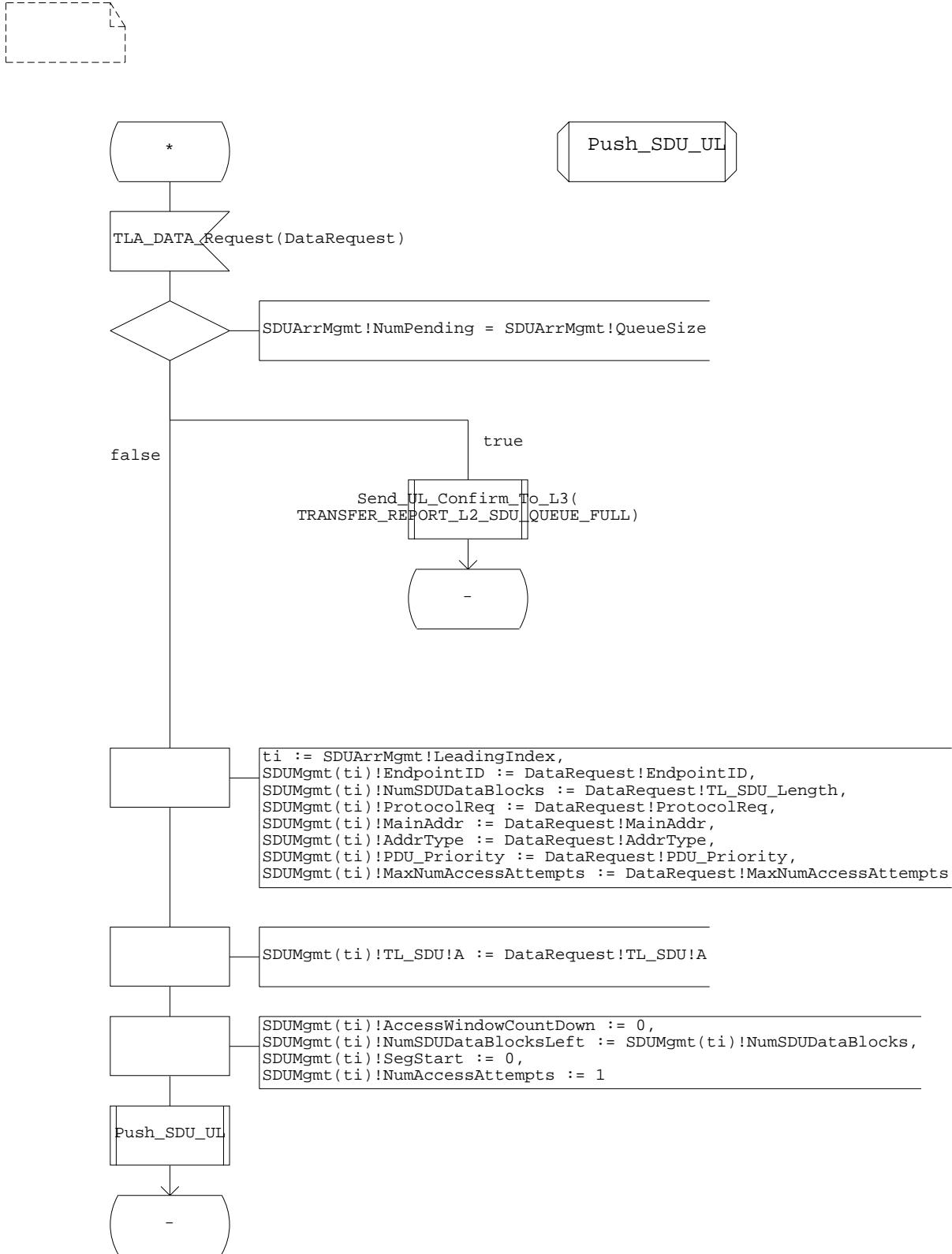
Bad_Presiding_Block_Gen_MS_UL

Send_UL_Confirm_To_L3



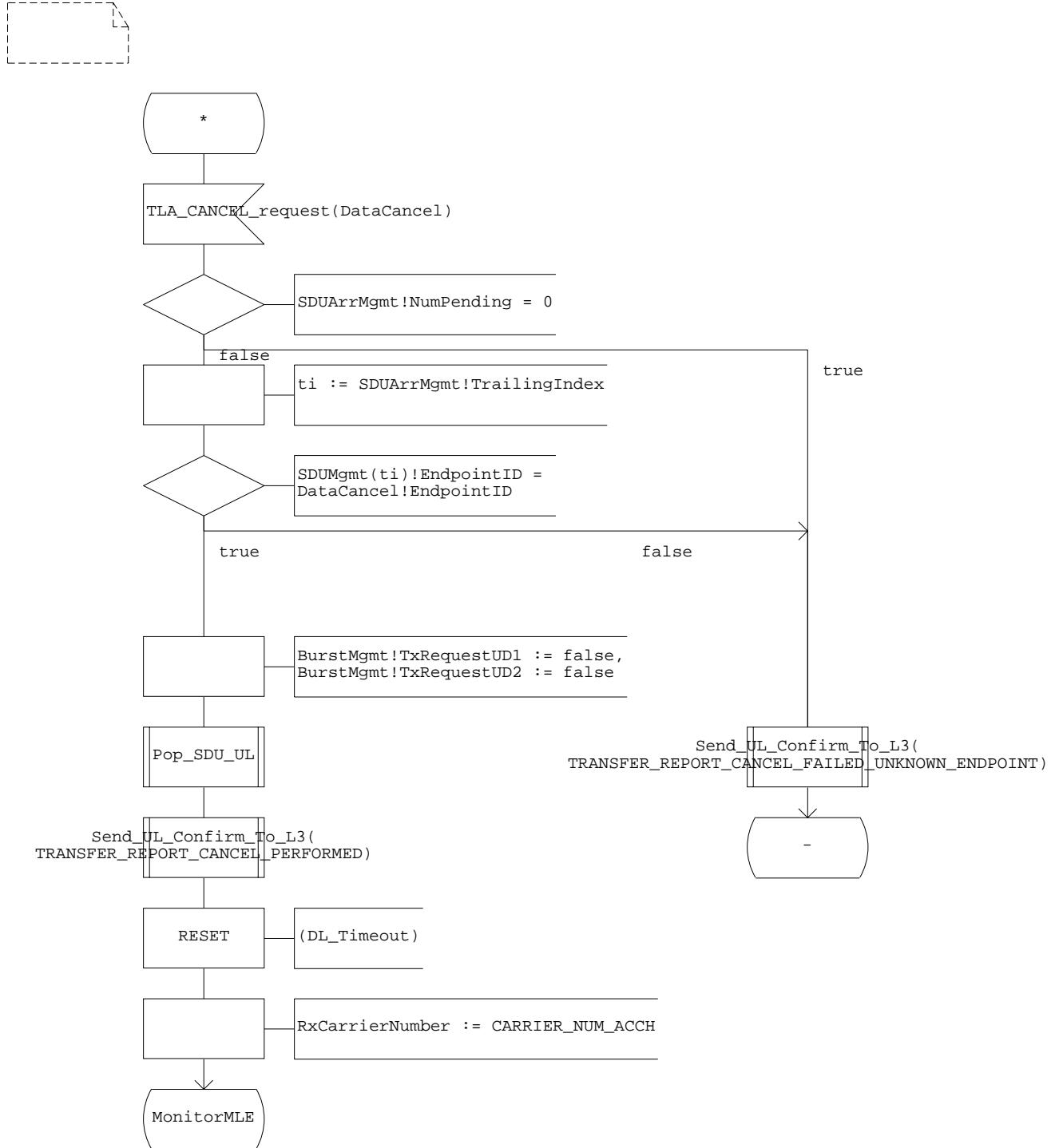






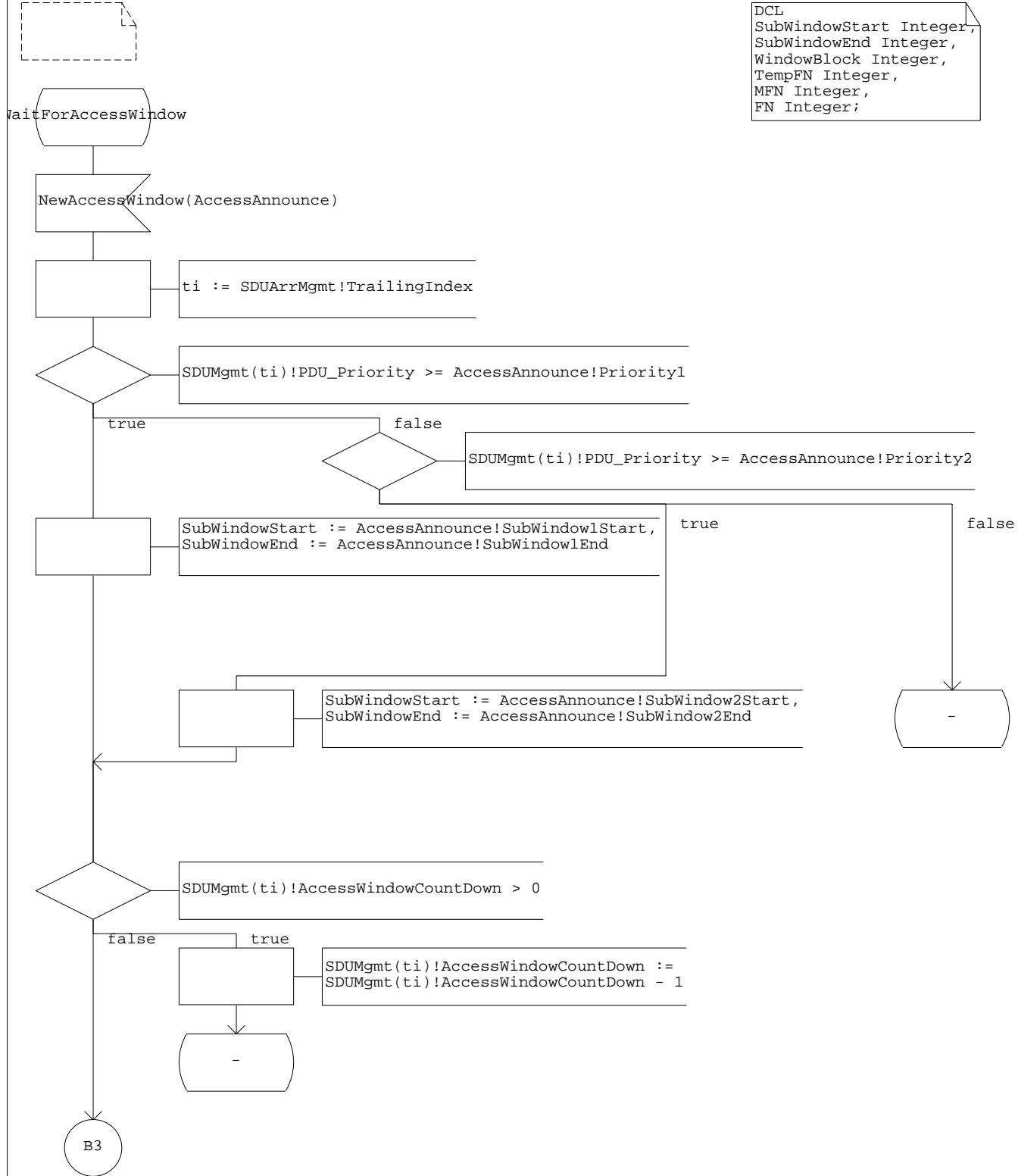
Process MS_UL

7 (15)



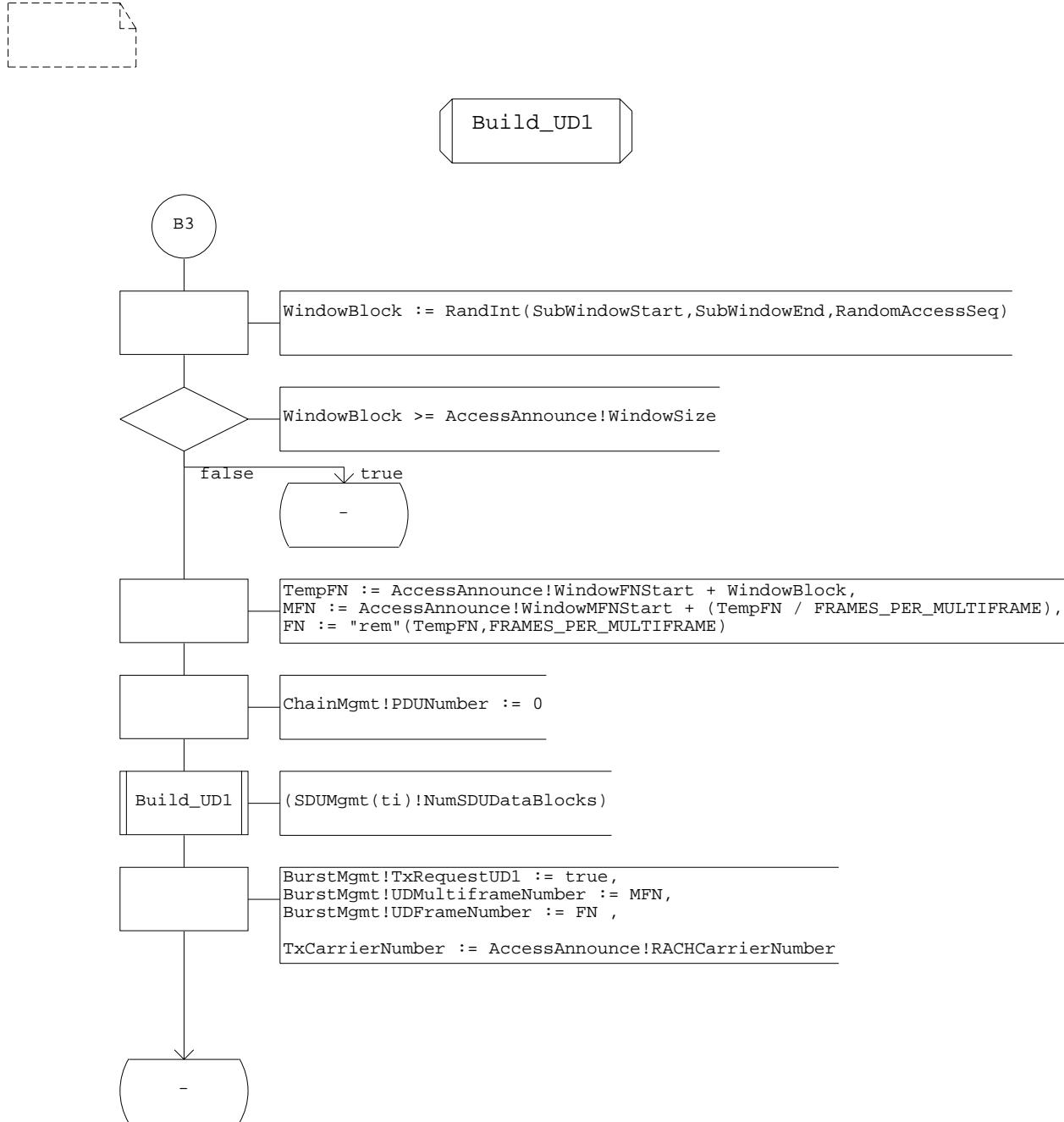
Process MS_UL

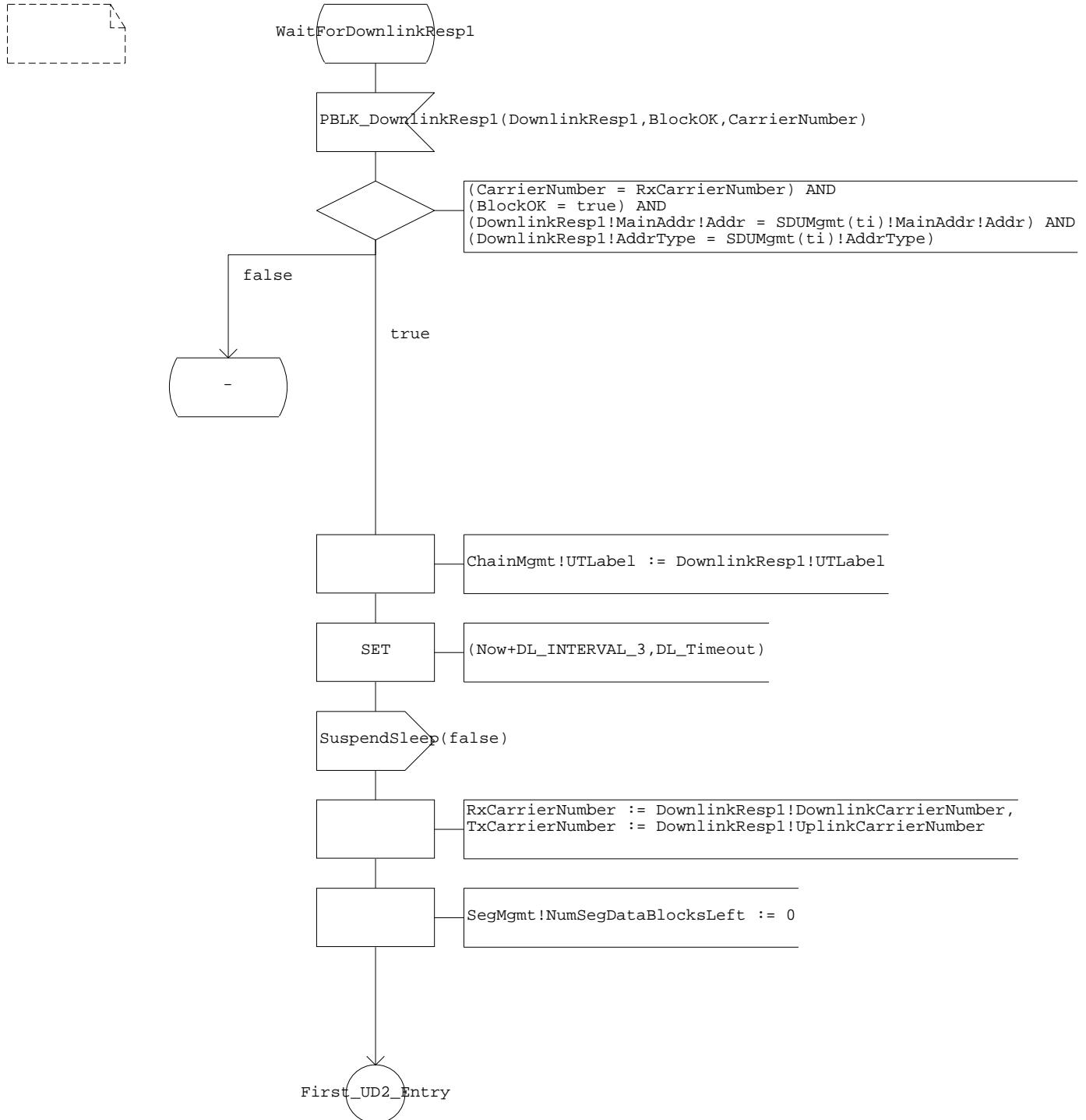
8 (15)



Process MS_UL

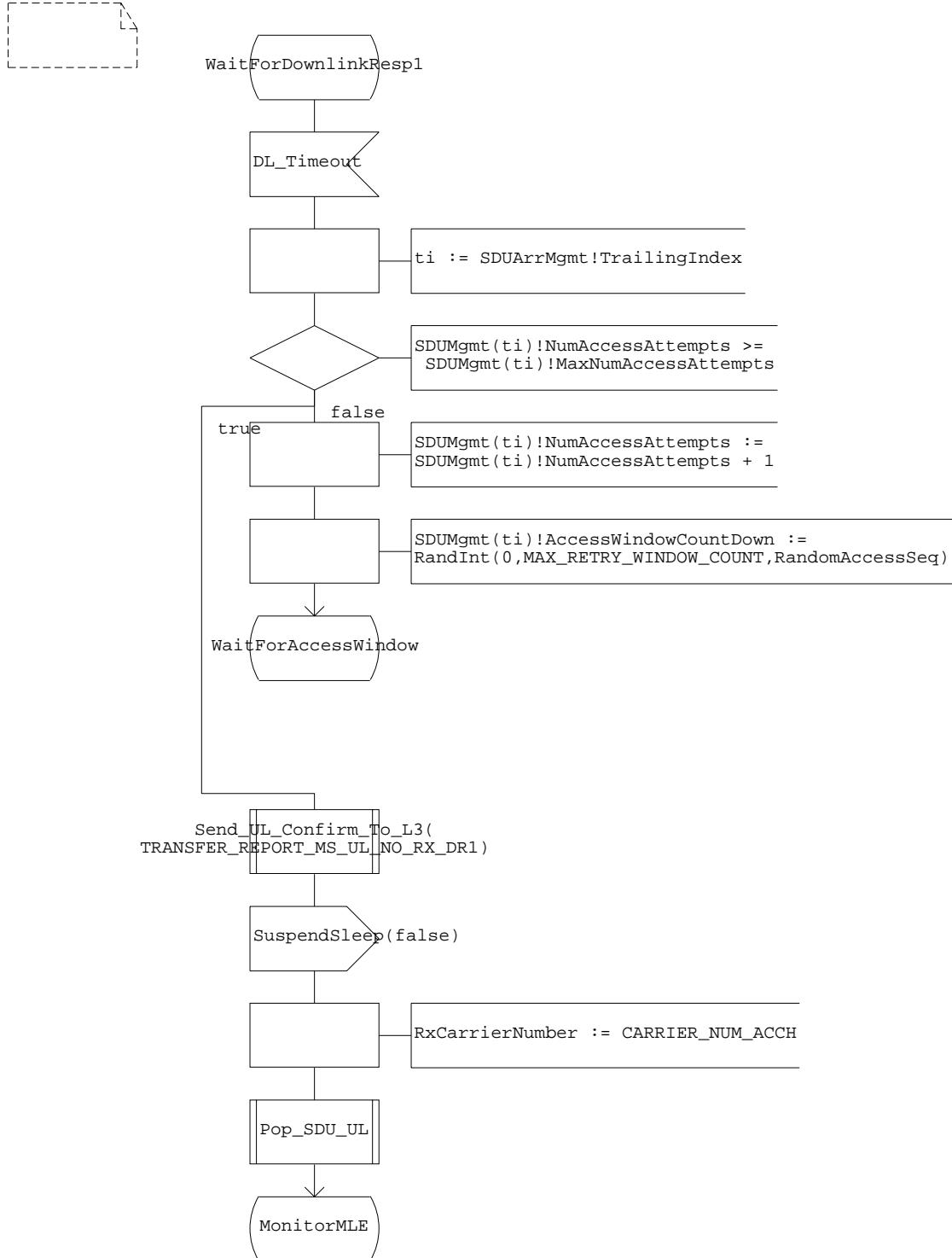
9 (15)





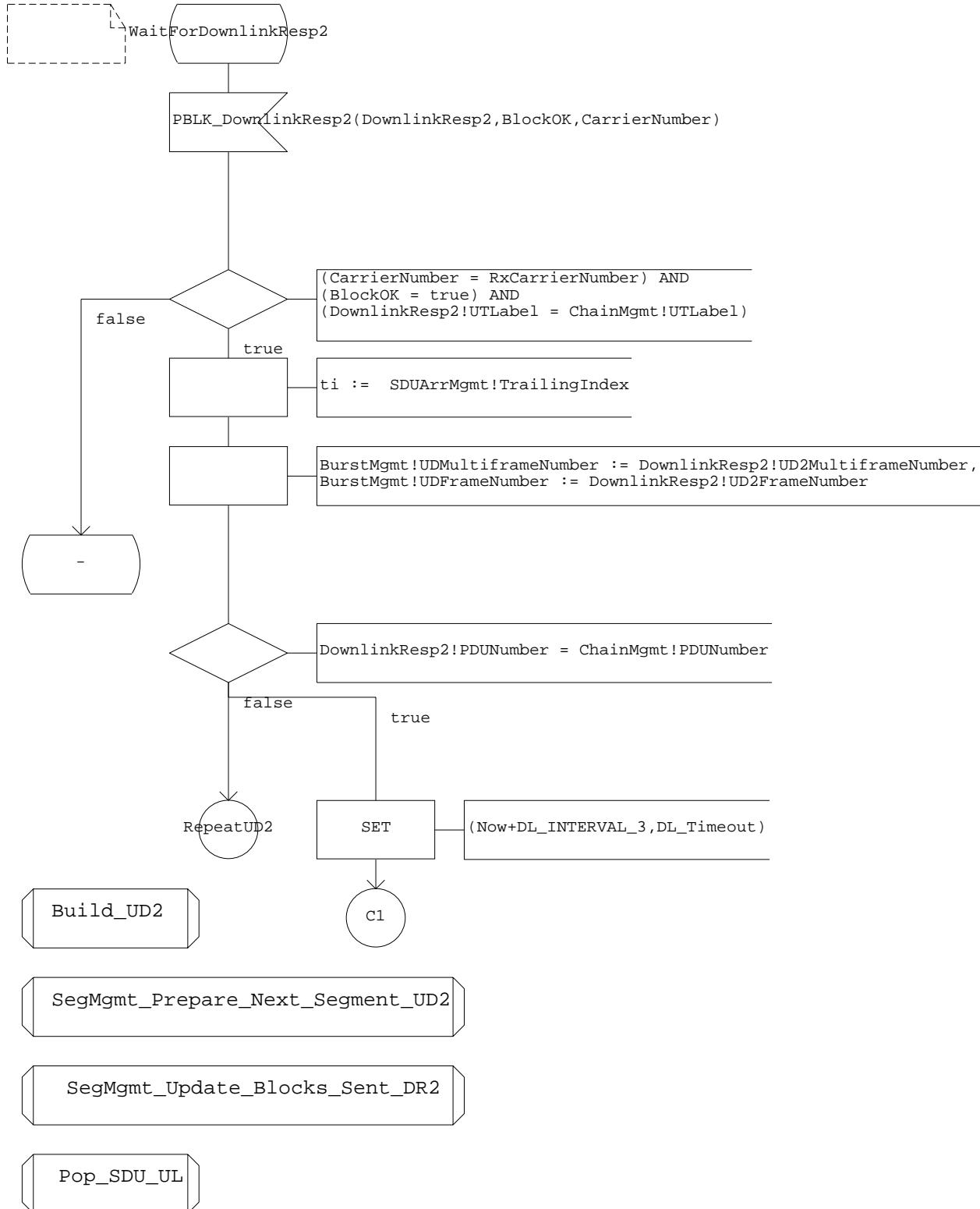
Process MS_UL

11(15)



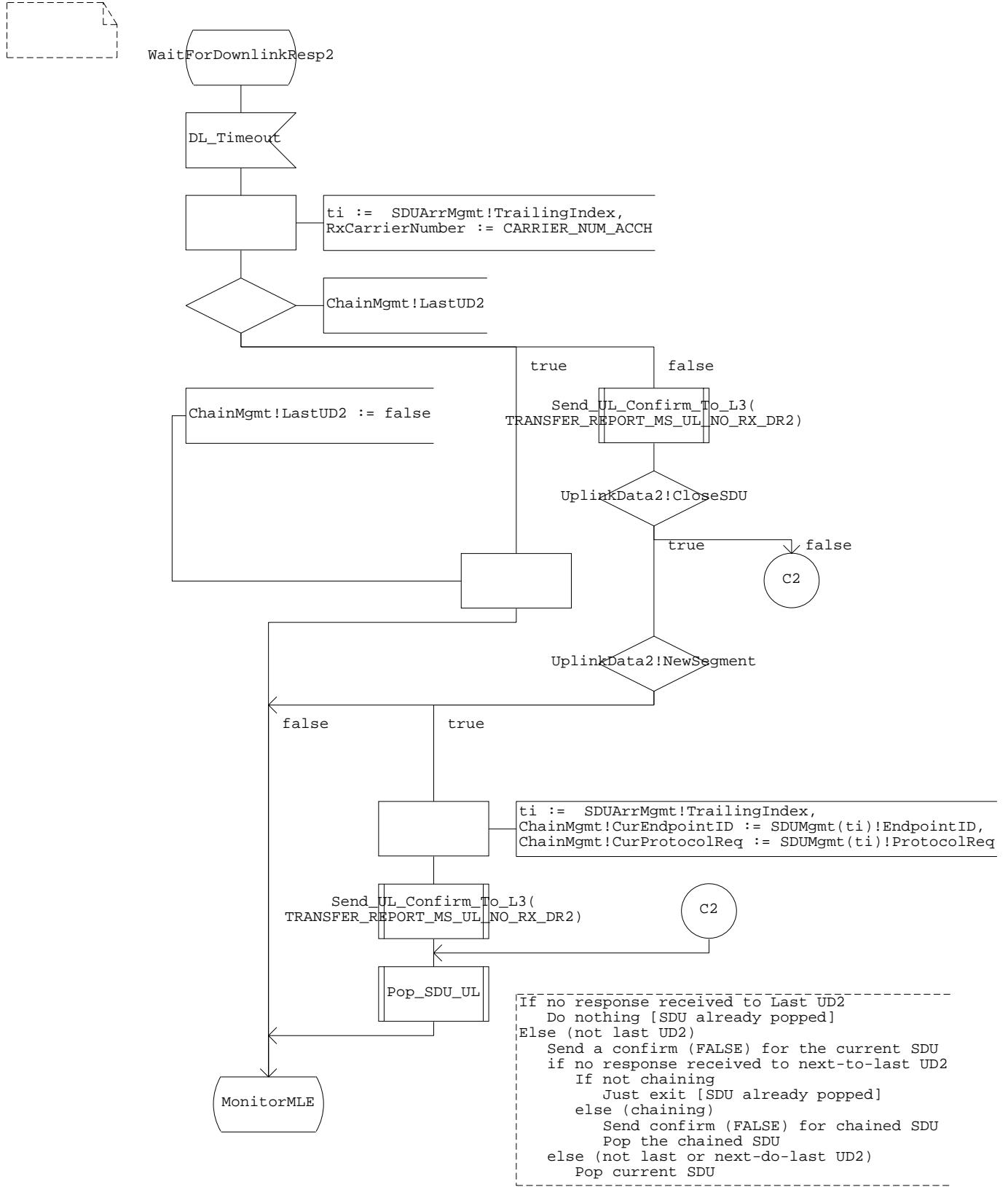
Process MS_UL

12(15)



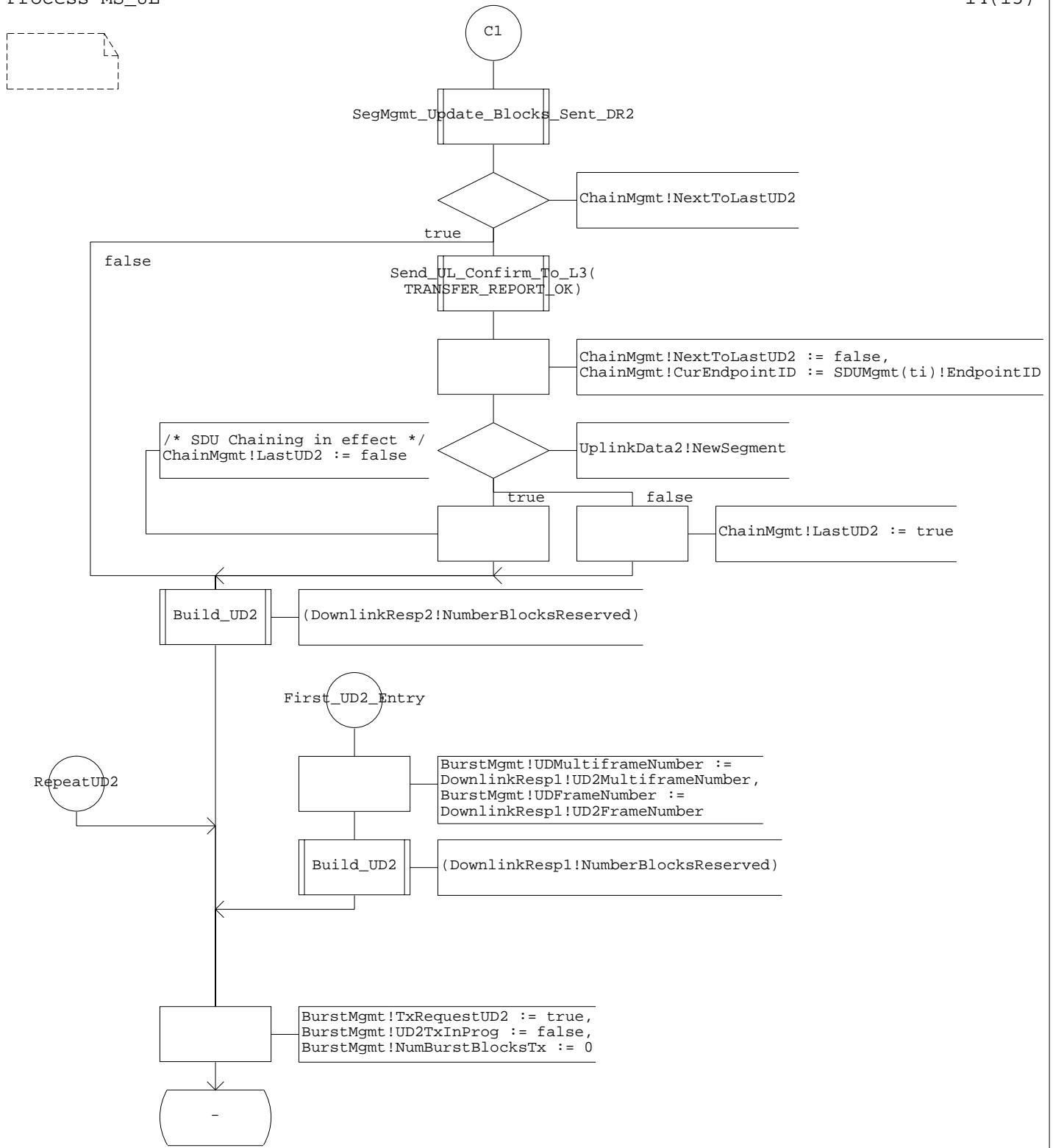
Process MS_UL

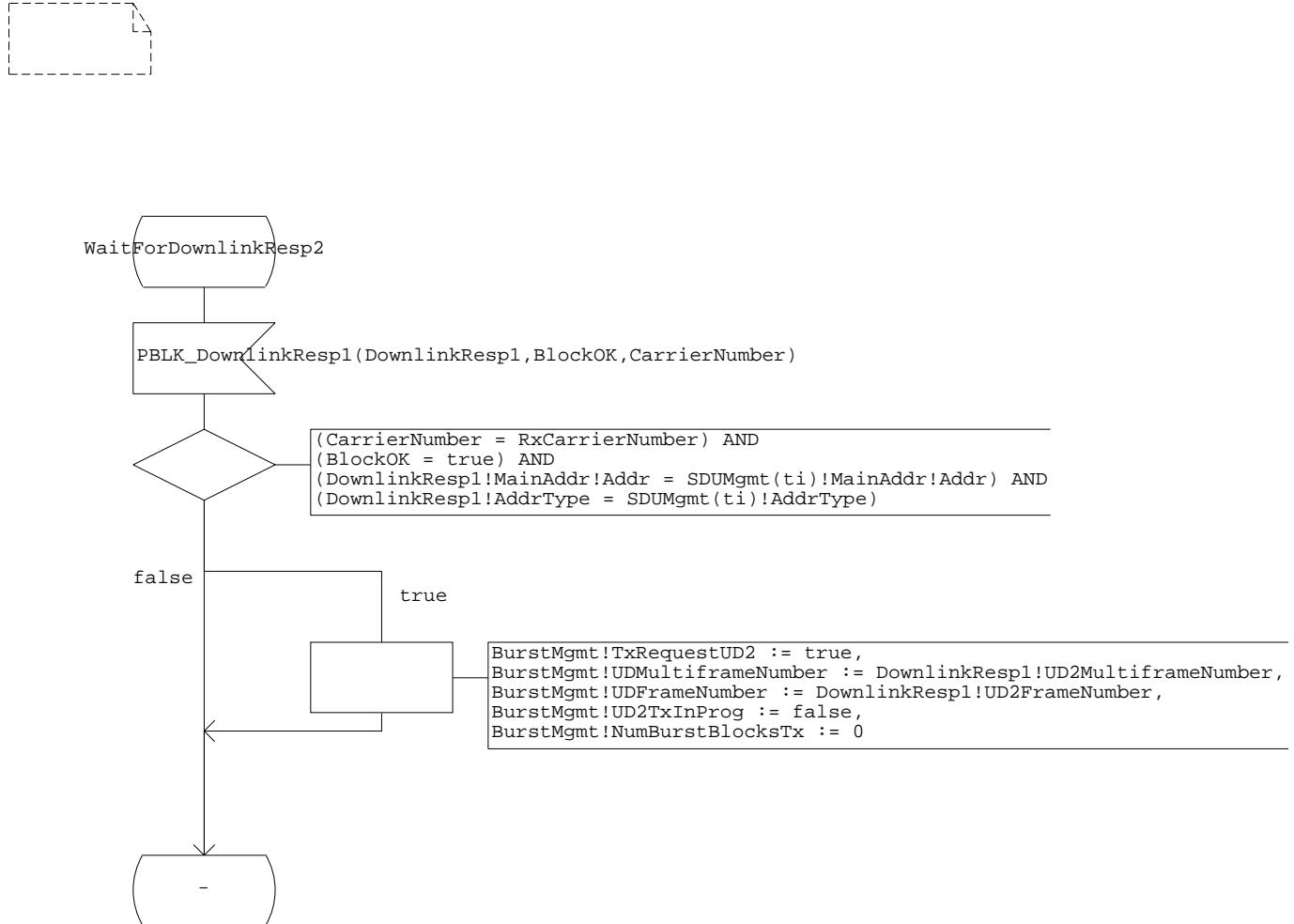
13 (15)



Process MS_UL

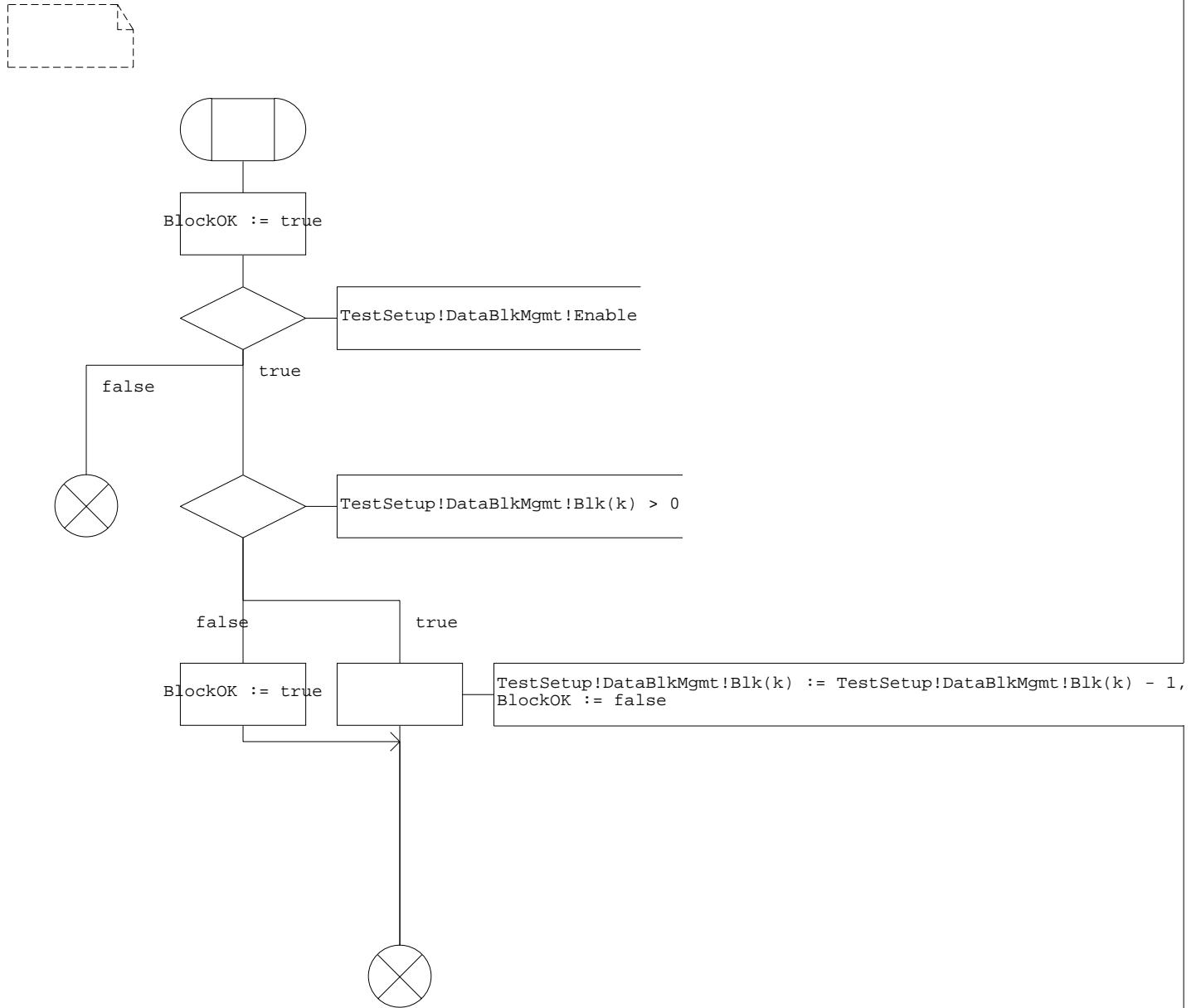
14 (15)





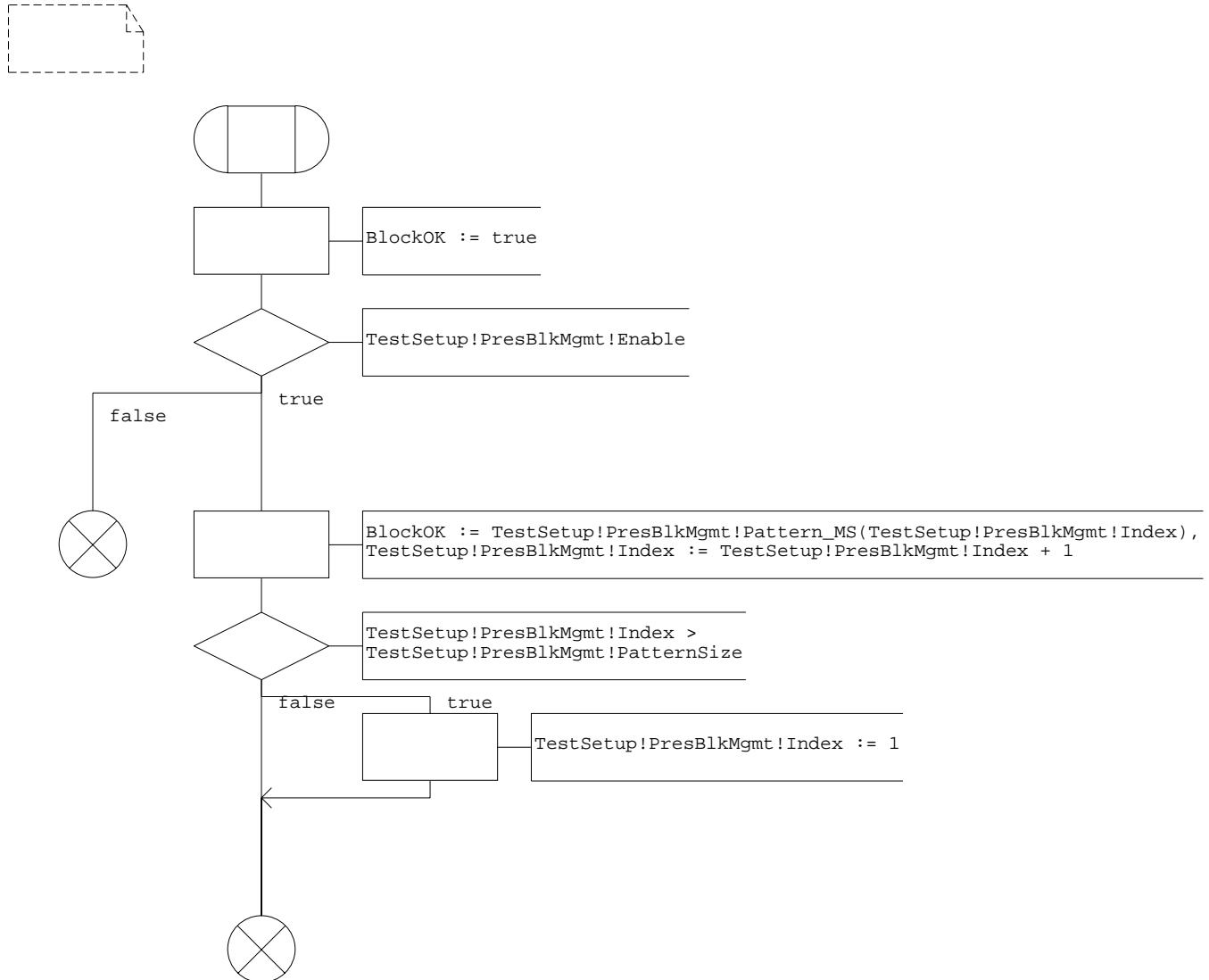
Procedure Bad_Data_Block_Gen_UL

1(1)



Procedure Bad_Presiding_Block_Gen_MS_UL

1(1)

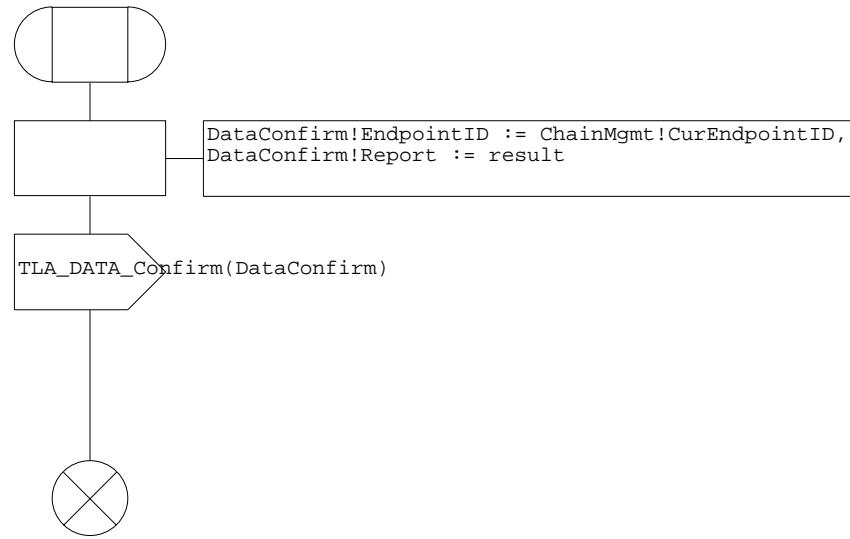


Procedure Send_UL_Confirm_To_L3

1(1)

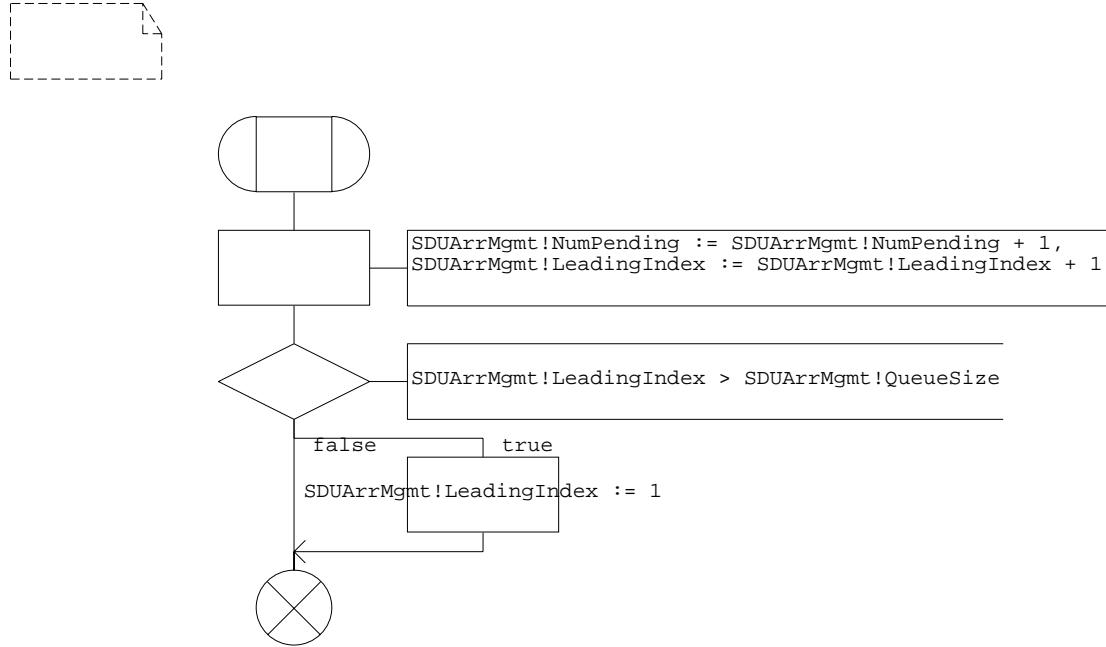
;FPAR IN result TransferReportType;

DCL
DataConfirm TLA_DataConfirmType;



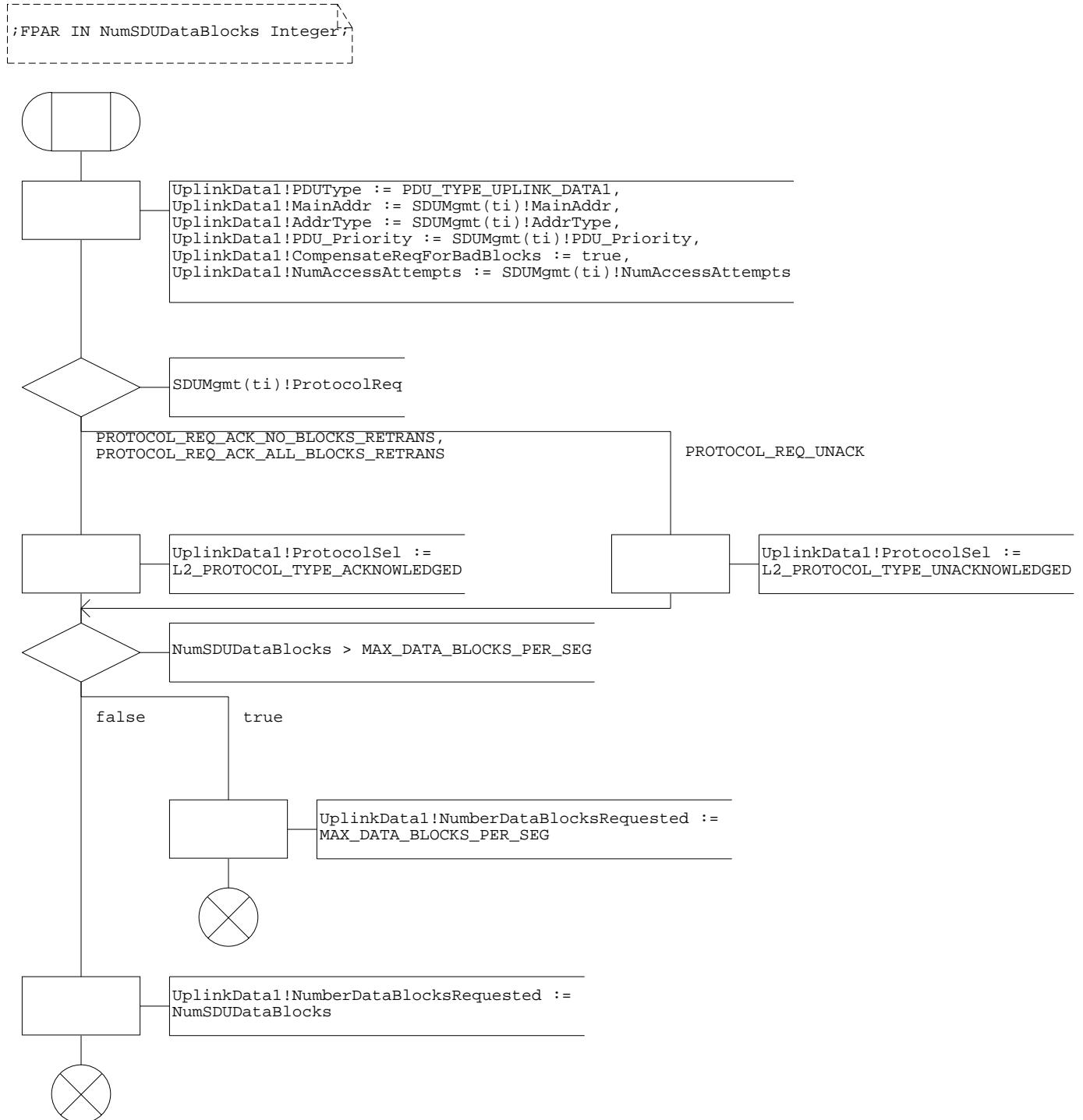
Procedure Push_SDU_UL

1(1)



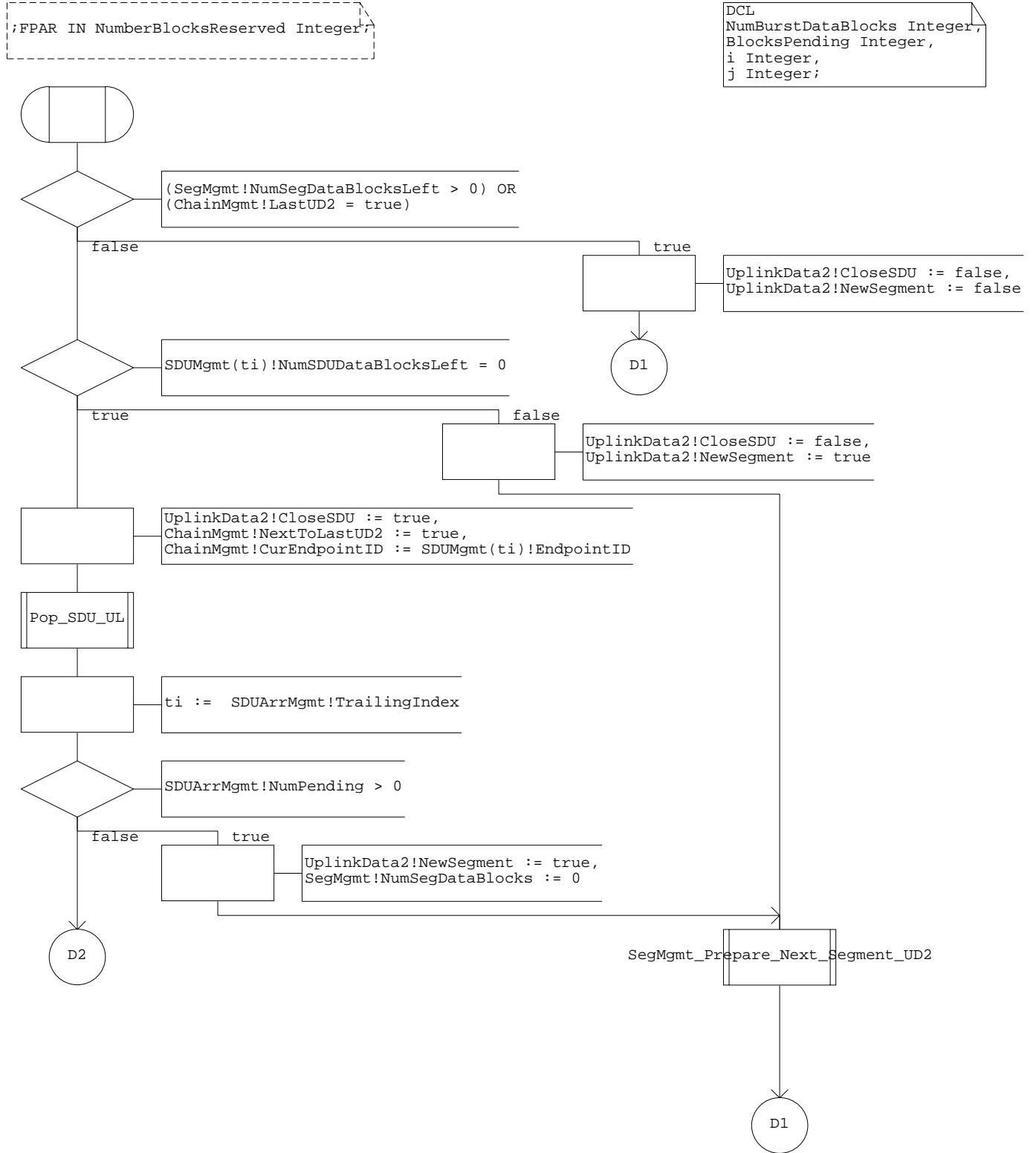
Procedure Build_UD1

1(1)



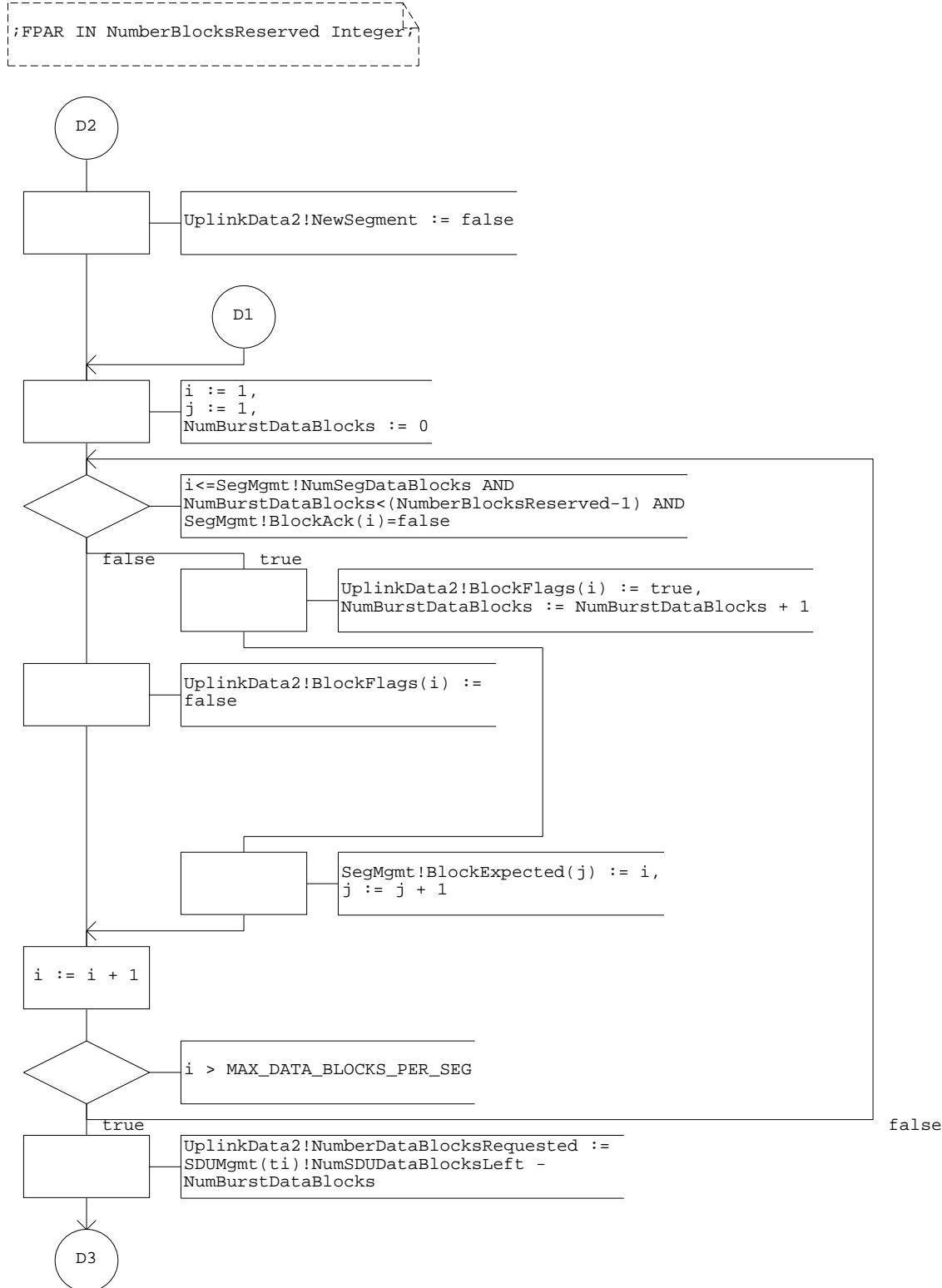
Procedure Build_UD2

1 (3)



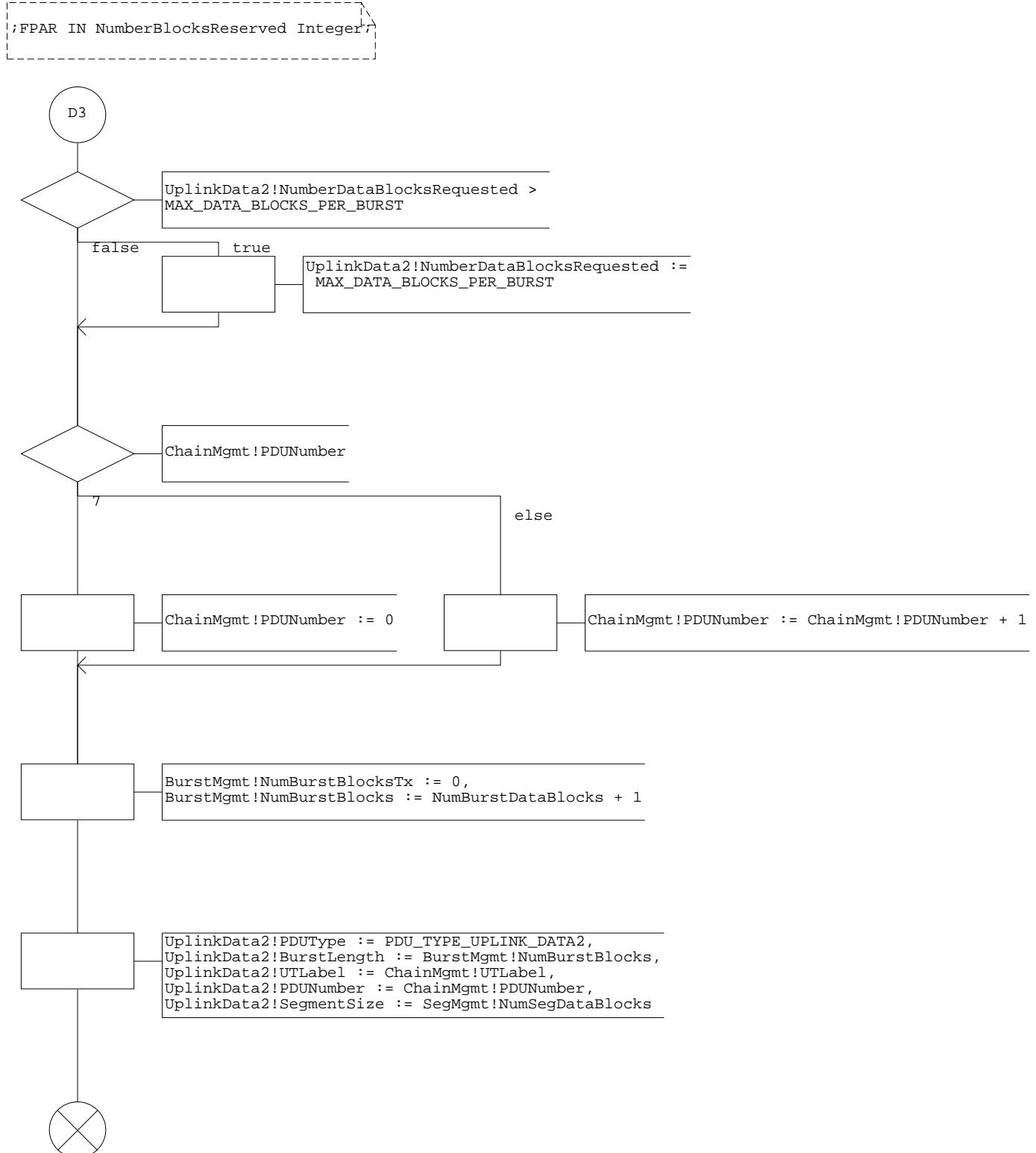
Procedure Build_UD2

2 (3)



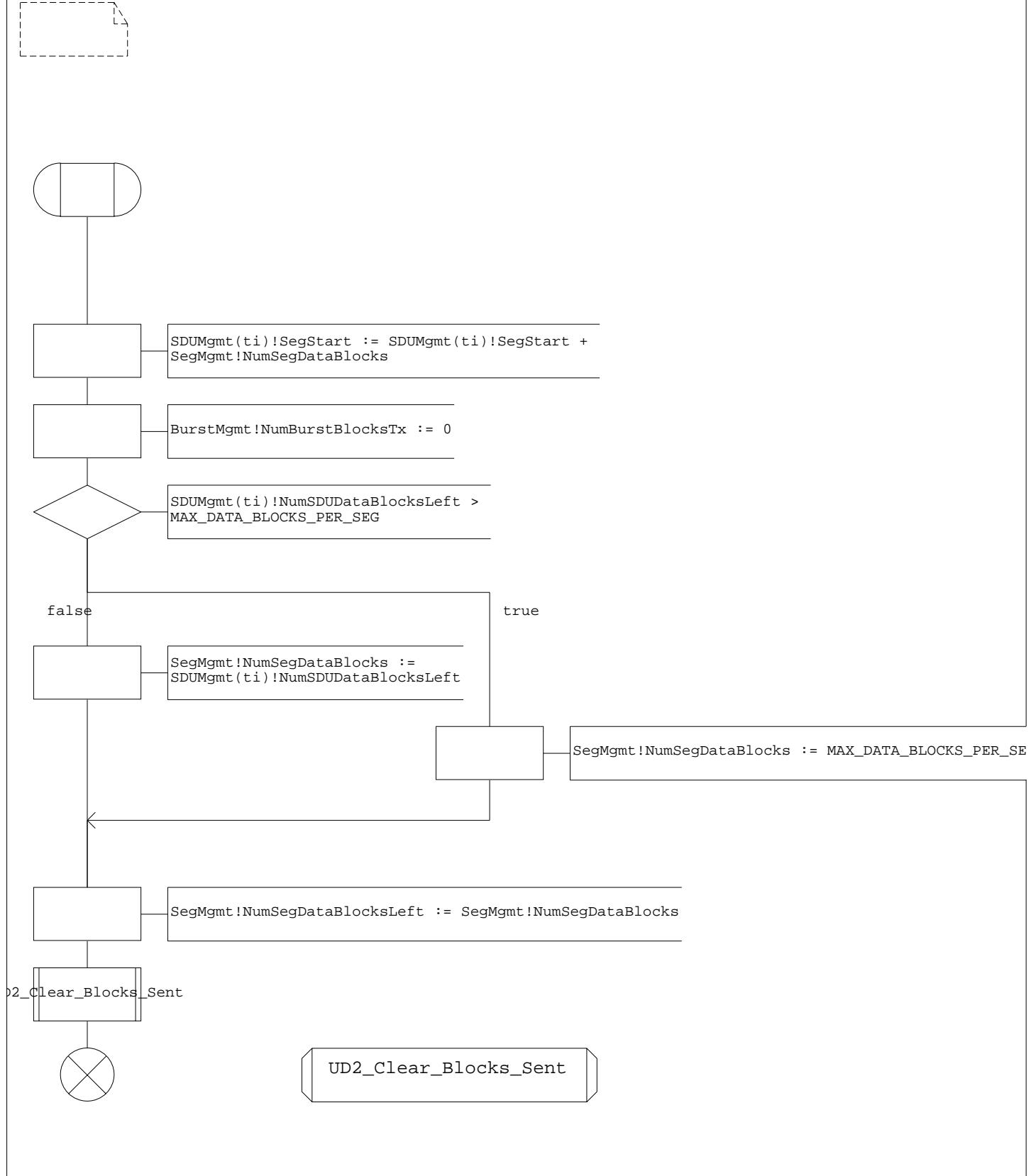
Procedure Build_UD2

3(3)



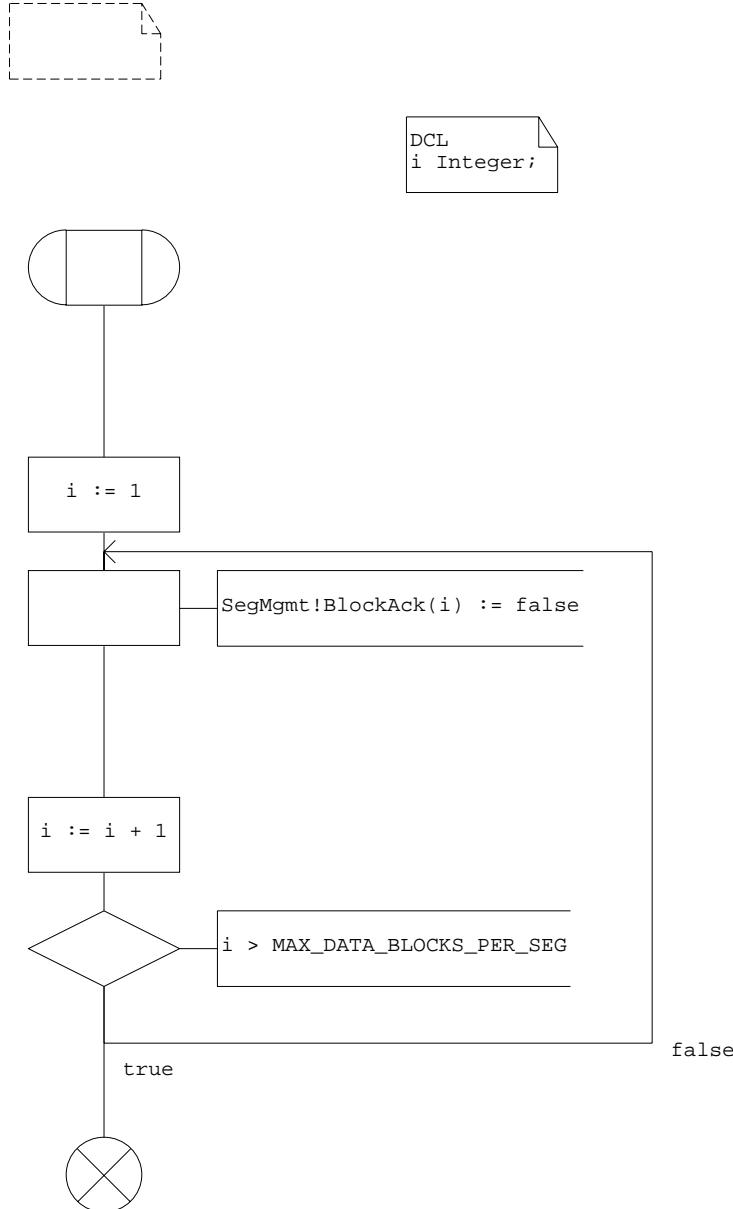
Procedure SegMgmt_Prepares_Next_Segment_UD2

1(1)



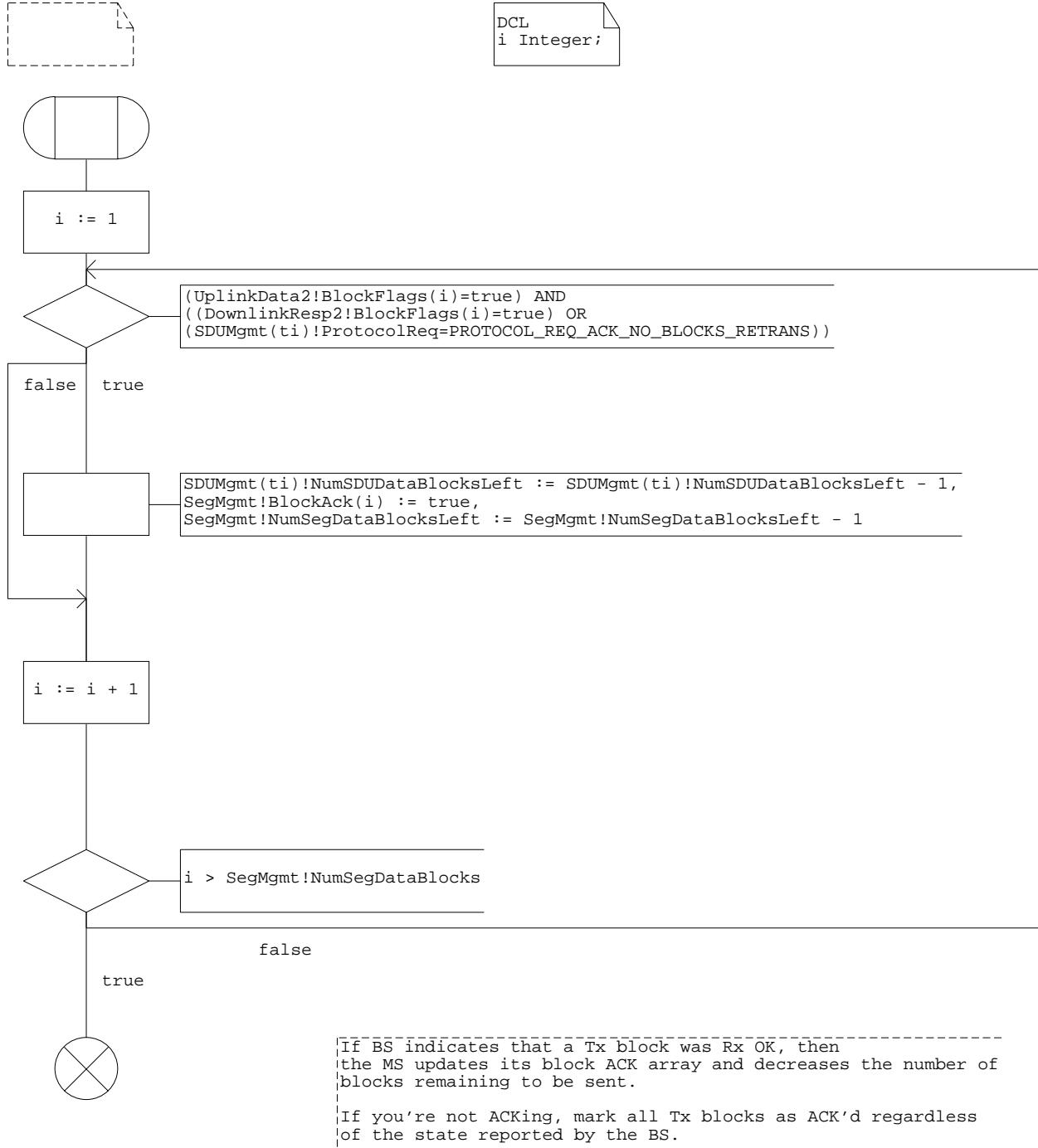
Procedure UD2_Clear_Blocks_Sent

1(1)



Procedure SegMgmt_Update_Blocks_Sent_DR2

1(1)



Procedure Pop_SDU_UL

1(1)

