TSG-RAN Working Group 2 (Radio layer 2 and Radio layer 3)
Stockholm 8th to 11th March 1999

TSGR2#2(99)058

Agenda Item: **4.2**

Source: **Editor**

Title: **3GPP S2.22 V0.0.1: Description of the RLC Protocol**

Document for:

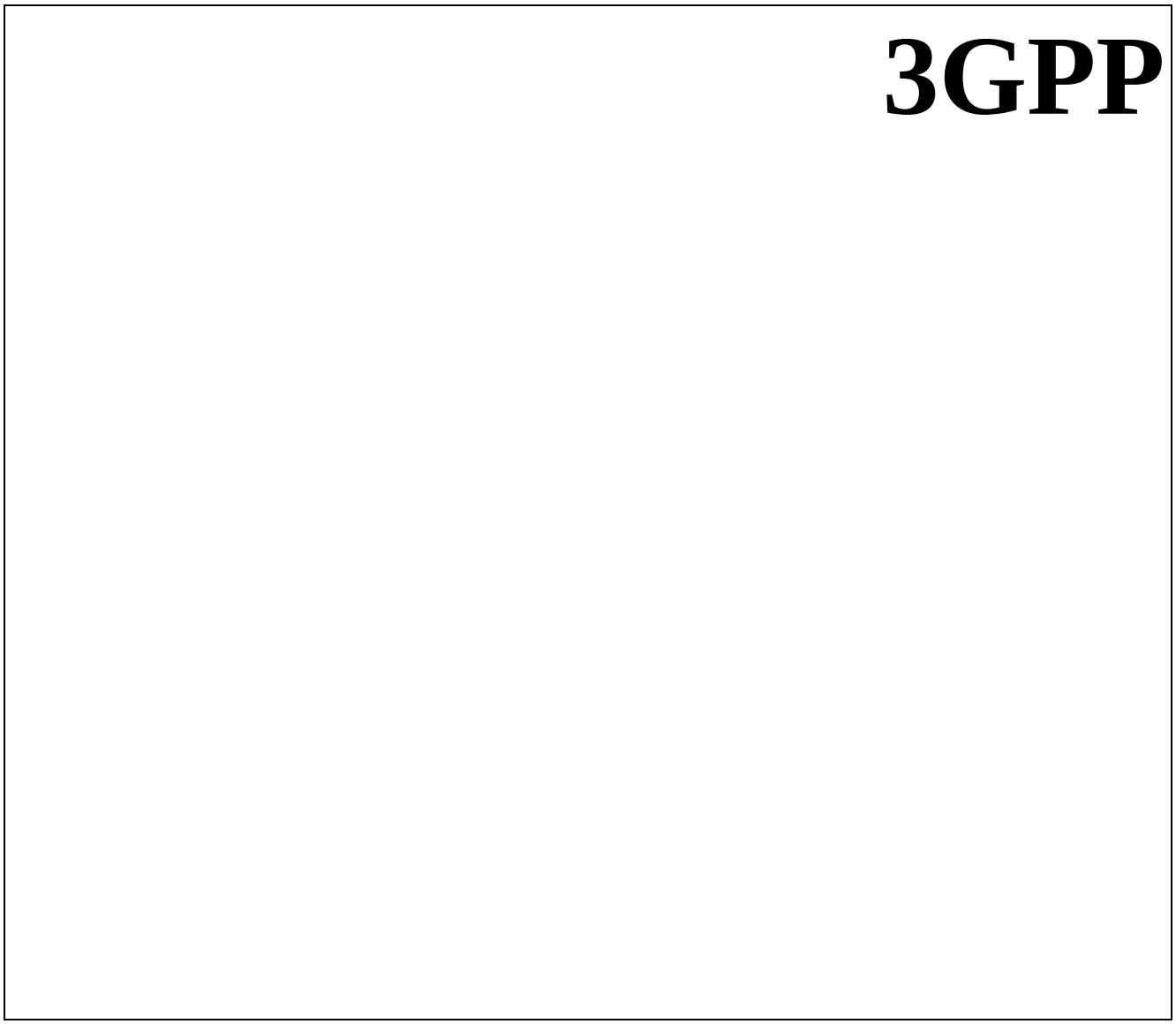
3GPP

TS RAN S2.22 V0.0.1 (1999-01)

Technical Specification

3rd Generation Partnership Project (3GPP);
Technical Specification Group (TSG) RAN;
Working Group 2 (WG2);

Description of the RLC Protocol



Reference
<Workitem> (<Shortfilename>.PDF)

Keywords
Digital cellular telecommunications system,
Universal Mobile Telecommunication System
(UMTS), UTRA, IMT-2000

3GPP

Postal address

Office address

Internet
secretariat@3gpp.org
Individual copies of this deliverable
can be downloaded from
http://www.3gpp.org

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present deliverable may have been declared to ETSI/3GPP. The information pertaining to these essential IPRs, if any, is publicly available for ETSI members and non-members, free of charge. This can be found in the latest version of the ETSI Technical Report: ETR 314: "Intellectual Property Rights (IPRs); Essential or potentially Essential, IPRs notified to ETSI in respect of ETSI standards". The most recent update of ETR 314, is available on the ETSI web server or on request from the Secretariat.
Pursuant to the ETSI Interim IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in the ETR 314, which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP)~~Special Mobile Group (SMG) of the European Telecommunications Standards Institute (ETSI)~~.
The contents of this TS are subject to continuing work within 3GPP TSG-RAN ~~TC SMG~~ and may change following formal TSG RAN ~~TC SMG~~ approval.

1. Scope

The scope of this description is to describe the RLC protocol. A description document is intermediate between a stage 2 document and a protocol specification. Once completed, it should be sufficient for manufacturers to start some " high level design " activities. It should allow as well to assess the complexity of the associated protocol. After the completion of a description document, the drafting of the protocol specification should not have to face difficulties which would impact the other protocols i.e. the radio interface protocol architecture should be stable. This means that some procedures which are felt critical in terms of complexity will need to be studied in more details in the description document so that no problem is faced in the writing of the final protocol.

The following lists typical contents for a description document :
1. list of procedures
2. logical flow diagrams for normal procedures
3. logical description of message (where it should be possible to guess roughly the size of the various information elements)
4. principles for error handling
5. some exceptional procedures which are felt critica
6. It should, as far as possible, have the same format and outline as the final specification

The following is not covered
1. exact message format
2. all scenarios

2. References

[1] UMTS XX.XX, UTRAN Architecture description;
[2] Vocabulary used in the UMTS L2&L3 Expert Group;
[3] [3] S2.01, Radio Interface Protocol Architecture Ver. 0.0.1
~~[4] UMTS YY.01, MS-UTRAN Radio Interface Protocol Architecture ;Stage 2;~~
[4] [4] S2.02, Layer 1; General requirements, Ver. 0.0.1
[5] [5] S2.03, Description of UE States and Procedures in Connected Mode, Ver. 0.0.1
[6] [6] s2.04 , UE Procedures in Idle Mode
~~[7] UMTS YY.02, Layer 1; General requirements;~~
~~[7] UMTS YY.03, UE States and Procedures in Connected Mode;~~
~~[7] UMTS YY.04, Description of procedures in idle Mode;~~

[7] [7] S2.21, Description of the MAC Protocol, Ver. 0.0.1
[8] [8] S2.31b, Description of the RRC Protocol, Ver. 0.0.1
~~[9] UMTS YY.21, Description of MAC protocol;~~
~~[9] UMTS YY.31, Description of RRC protocol.~~

3. Definitions and Abbreviations

ARQ	Automatic Repeat Request
BCCH	Broadcast Control Channel
BCH	Broadcast Channel
C-	Control-
CC	Call Control
CCCH	Common Control Channel
CCH	Control Channel
CCTrCH	Coded Composite Transport Channel
CN	Core Network
CRC	Cyclic Redundancy Check
DC	Dedicated Control (SAP)
DCCH	Dedicated Control Channel
DCH	Dedicated Channel
DL	Downlink
DSCH	Downlink Shared Channel
DTCH	Dedicated Traffic Channel
FACH	Forward Link Access Channel
FCS	Frame Check Sequence
FDD	Frequency Division Duplex
GC	General Control (SAP)
HO	Handover
ITU	International Telecommunication Union
kbps	kilo-bits per second
L1	Layer 1 (physical layer)
L2	Layer 2 (data link layer)
L3	Layer 3 (network layer)
~~LAC~~	~~Link Access Control~~
MAC	Medium Access Control
MS	Mobile Station
MM	Mobility Management
Nt	Notification (SAP)
PCCH	Paging Control Channel
PCH	Paging Channel
PDU	Protocol Data Unit
PHY	Physical layer
PhyCH	Physical Channels
RACH	Random Access Channel
RLC	Radio Link Control
RNTI	Radio Network Temporary Identity
RRC	Radio Resource Control
SAP	Service Access Point
SCCH	Synchronization Control Channel
SCH	Synchronization Channel
SDU	Service Data Unit
TCH	Traffic Channel
TDD	Time Division Duplex
TFI	Transport Format Indicator
TFCI	Transport Format Combination Indicator
TPC	Transmit Power Control

U-	User-
UE	User Equipment
UL	Uplink
UMTS	Universal Mobile Telecommunications System
URA	UTRAN Registration Area
UTRA	UMTS Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network

4. General

4.1. Objective

4.2. Overview on sublayer architecture

4.2.1. Model of RLC

Figure 1 gives an overview model of the RLC layer. The figure illustrates two peer entities, one in the UE and one in the UTRAN. Though it is not shown in the figure the RLC layer may consist of several entities. A RLC entity offers three kinds of data transfer services to the higher layers. The services are transparent mode, unacknowledged mode and acknowledged mode data transfer. The entities have one transmitting side and one receiving side. More detailed 7descriptions of the transmitting and receiving sides are given in subsections 4.2.1 and 4.2.2.

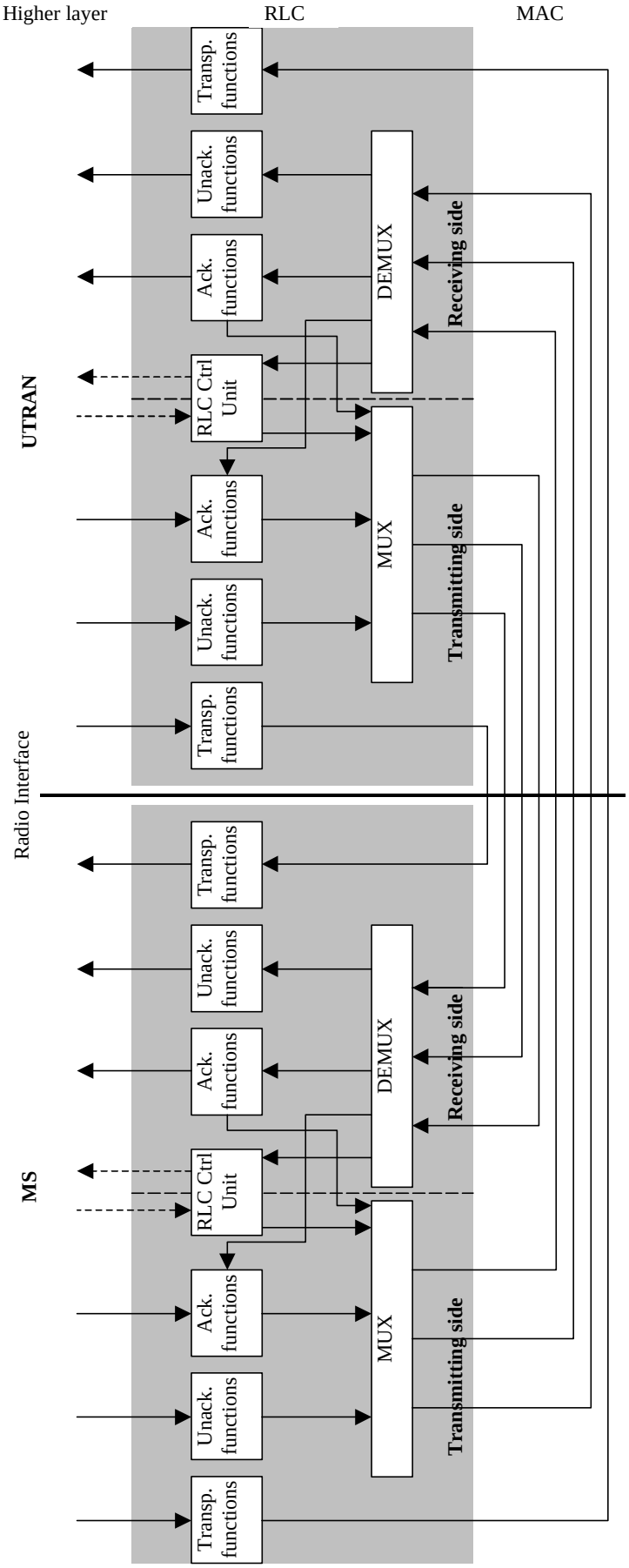


Figure Error! Style not defined.-Error! Bookmark not defined.~~**Figure 1.**~~ **Overview model of RLC.**

4.2.1.1. Model of transmitting side

A model of the transmitting side of a RLC entity is presented in ~~Figure~~ below.

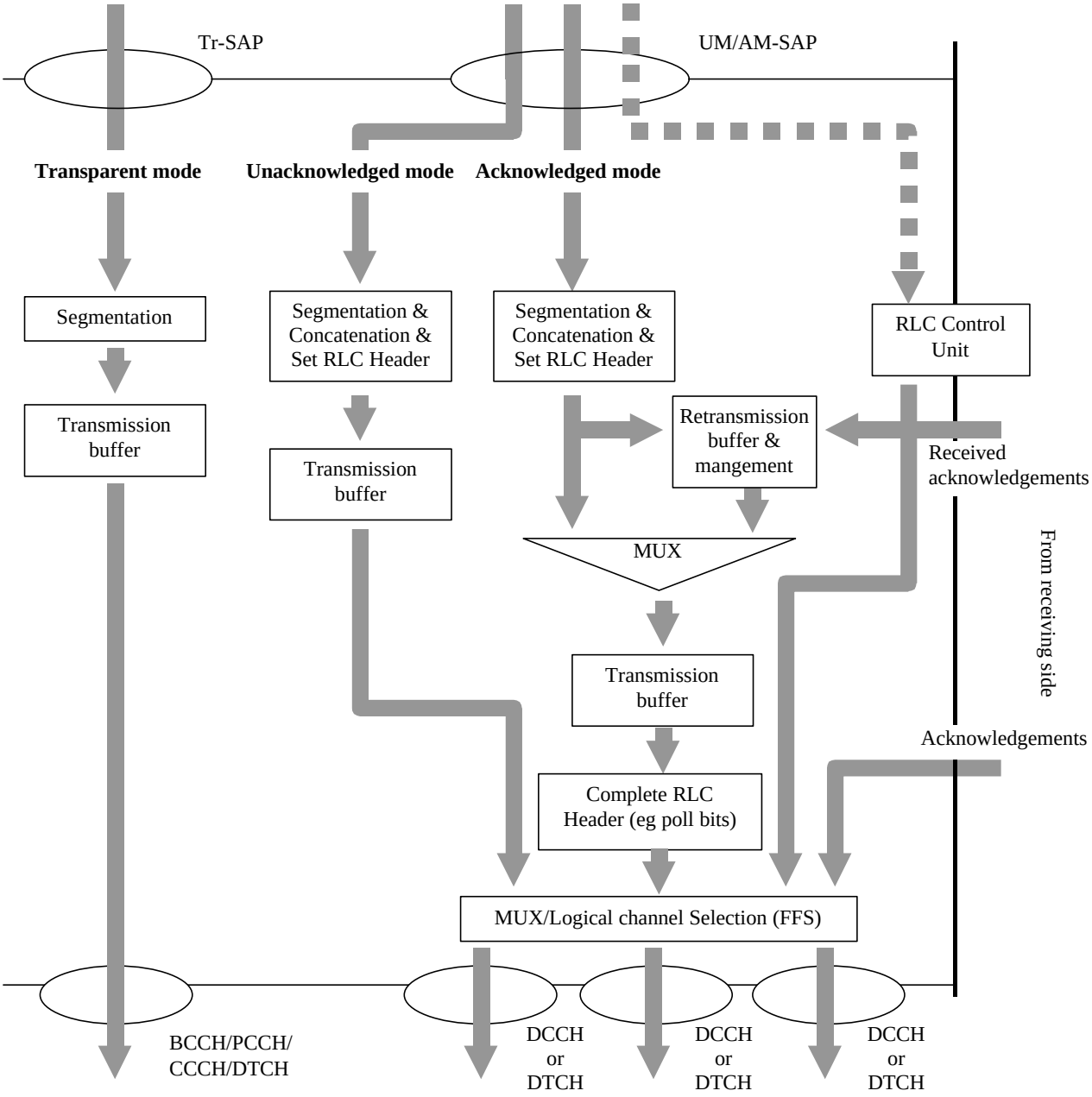


Figure Error! Style not defined.-Error! Bookmark not defined.

~~Figure 2~~. The transmitting side of a RLC entity.

RLC offers three kinds of data transfer services to the higher layers through the RLC-SAP. The services are transparent mode, unacknowledged mode and acknowledged mode data transfer. The transparent mode ~~protocol machines are~~ is independent ~~for the different modes and linkage between them is possible~~from both unacknowledged mode data transfer and acknowledged mode data transfer. The independence between unacknowledged mode and acknowledged mode is FFS. If the logical channel is a DCCH then there is only one DCCH. The dashed line illustrates the possibility to transfer higher layer data during the establishment of an RLC *link [Note: This could be useful in the control plane but it is for further study].* A RLC entity can provide unacknowledged and acknowledged mode data transfer simultaneously, but not transparent mode data transfer. Therefore are these services provided through different SAPs, Tr-SAP and UM/AM-SAP. The data flow through the transmitting side of an RLC entity for these services are described below.

1. **Transparent mode data transfer**
   RLC receives SDUs from the higher layers. RLC might segment the SDUs into appropriate RLC PDUs without adding any overhead. How to perform the segmentation is decided upon when the service is established. RLC delivers the RLC PDUs to MAC through either a BCCH, PCCH~~, DCCH,~~ or a DTCH. The delivery of RLC PDUs to MAC through CCCH is FFS. Which type of logical channel depends on if the higher layer is located in the control plane (BCCH, PCCH, ~~DCCH~~CCCH) or user plane (DTCH).
2. **Unacknowledged mode data transfer**
   RLC receives SDUs from the higher layers. If the SDU is too large it is segmented into appropriate RLC PDUs. The SDU might also be concatenated with other SDUs. RLC adds a header and the PDU is placed in the transmission buffer. The MUX then decides which PDUs and when the PDUs are delivered to MAC. The MUX also decides which logical channel that should be used. The number of logical channels that is needed is decided upon when the service is established (e.g. in the figure there is three logical channels). The type of the logical channels depends on if the higher layer is located in the control plane (DCCH) or in the user plane (DTCH).
3. **Acknowledged mode data transfer**
   RLC receives SDUs from the higher layers. The SDUs are segmented and/or concatenated to PDUs of fixed length. The length of the PDUs is decided upon when the service is established. After that RLC adds a header and the PDU is placed in the retransmission buffer and the transmission buffer. The MUX then decides which PDUs and when the PDUs are delivered to MAC, e.g. it could be useful to send RLC control PDUs on one logical channel and data PDUs on another logical channel. The PDUs are delivered to the MUX via a function that sets the poll bit in the PDUs.
   The retransmission buffer also receives acknowledgements from the receiving side, which are used to indicate retransmissions of PDUs and when to delete a PDU from the retransmission buffer.

The RLC control unit controls the RLC entity and handles control signalling between the peer entities (e.g. establishment and release of a RLC link). It is split between the transmitting and receiving side.

4.2.1.2. Model of receiving side

A model of the receiving side of an RLC entity is presented in Figure below.

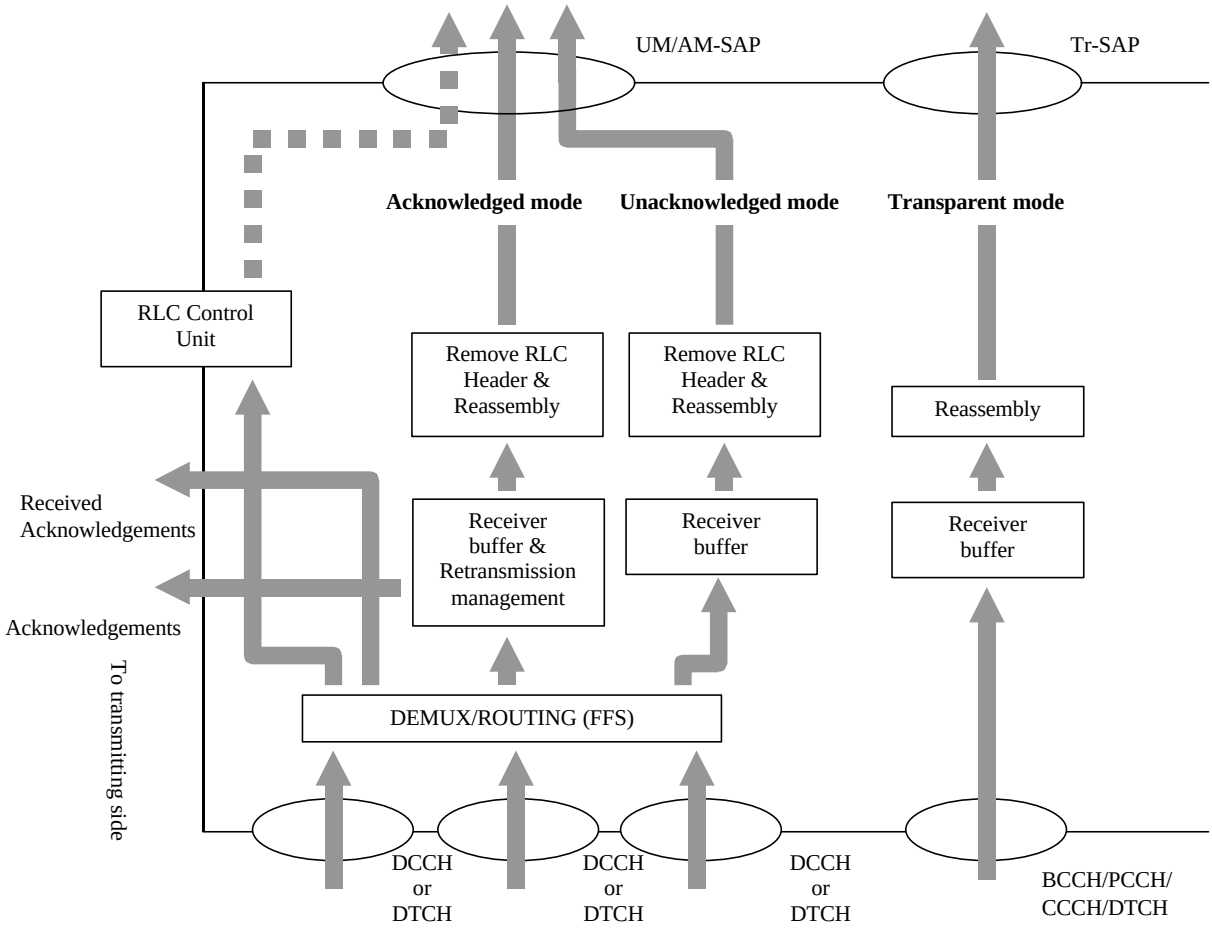


Figure Error! Style not defined.-Error! Bookmark not defined.

~~Figure 3~~. RLC receiver entity.

The data flow through the receiving side of an RLC entity for the three different data transfer services are described below.

1. **Transparent mode data transfer**
   RLC receives PDUs through either a DCCH or a DTCH from the MAC sublayer. RLC reassembles (if segmentation has been performed) the PDUs into RLC SDUs. How to perform the reassembling is decided upon when the service is established. RLC delivers the RLC SDUs to the higher layer through the RLC-SAP.
2. **Unacknowledged mode data transfer**
   RLC receives PDUs through one of the logical channels from the MAC sublayer. RLC removes header from the PDUs and reassembles the PDUs (if segmentation has been performed) into RLC SDUs. After that the SDUs are delivered to the higher layer.
3. **Acknowledged mode data transfer**
   RLC receives PDUs through one of the logical channels from the MAC sublayer. The PDUs are placed in the receiver buffer until a complete SDU has been received. The receiver buffer requests retransmissions of PDUs by sending negative acknowledgements to the peer entity. After that the headers are removed from the PDUs and the PDUs are reassembled into a SDU. Finally the SDU is delivered to the higher layer.

The receiving side also receives acknowledgements from the peer entity. The acknowledgements are passed to the retransmission buffer on the transmitting side.
The RLC control unit controls the RLC entity and handles control signalling between the peer entities (e.g. establishment and release of a RLC link). It is split between the transmitting and receiving side.

5. Functions

For a detailed description of the following functions see [3].

- Connection Control;
- Segmentation and reassembly;
- Concatenation;
- Padding;
- Transfer of user data;
- Error correction;
- In-sequence delivery of higher layer PDUs;
- Duplicate Detection;
- Flow control;
- Protocol error detection and recovery.

The following potential function(s) are regarded as further study items:

- Suspend/resume function;
- ~~Keep-Alive;~~
- ~~FCS error detection and handling;~~
- Ciphering.
- Quick repeat (FFS).

6. Services provided to upper layers

For a detailed description of the following functions see [3].

- **RLC connection establishment/release;**
- **Transparent data transfer Service**

  Following functions are needed to support transparent data transfer:
  - Segmentation and reassembly
  - Transfer of user data;
- **Unacknowledged data transfer Service**

  Following functions are needed to support unacknowledged data transfer:
  - Segmentation and reassembly
  - Concatenation
  - Transfer of user data;
- **Acknowledged data transfer Service**

  Following functions are needed to support acknowledged data transfer:
  - Segmentation and reassembly
  - Concatenation
  - Transfer of user data
  - Error correction
  - In-sequence delivery of higher layer PDUs

  - Duplicate detection
  - Flow Control
  - Protocol error detection and recovery;
- **QoS setting**;
- **Notification of unrecoverable errors**.
- **Multicast delivery of higher layer messages. (FFS)**

~~Table 1. RLC modes and functions in UE downlink side~~

Table Error! Style not defined.-Error! Bookmark not defined.**: RLC modes and functions in UE downlink side**

Service	Functions	CCCH	DCCH	DTCH
Transparent Service	Applicability	+	-	+
	Segmentation	-	-	+
Unacknowledged Service	Applicability	-FFS	+	+
	Segmentation	-	+	+
	Concatenation	-	+	+
Acknowledged Service	Applicability	-	+	+
	Segmentation	-	+	+
	Concatenation	-	+	+
	Flow Control	-	+	+
	Error Correction	-	+	+
	Protocol error correction & recovery	-	+	+

~~Table 2. RLC modes and functions in UE uplink side~~

Table Error! Style not defined.-Error! Bookmark not defined.**: RLC modes and functions in UE uplink side**

Service	Functions	SCCH	BCCH	PCCH	CCCH	DCCH	DTCH
Transparent Service	Applicability	+	+	+	+	-	+
	Reassembly	+	+	+	-	-	+
Unacknowledged Service	Applicability	+	+	+	-FFS	+	+
	Reassembly	+	+	+	-	+	+
Acknowledged Service	Applicability	-	-	-	-	+	+
	Reassembly	-	-	-	-	+	+
	Error correction	-	-	-	-	+	+
	Flow Control	-	-	-	-	+	+
	In sequence delivery	-	-	-	-	+	+
	Duplicate detection	-	-	-	-	+	+
	Protocol error correction & recovery	-	-	-	-	+	+

Table 3. ~~RLC modes and functions in UTRAN downlink side~~

Table Error! Style not defined.-Error! Bookmark not defined.**: RLC modes and functions in UTRAN downlink side**

Service	Functions	SCCH	BCCH	PCCH	CCCH	DCCH	DTCH
Transparent Service	Applicability	+	+	+	+	-	+
	Segmentation	+	+	+	-	-	+
Unacknowledged Service	Applicability	+	+	+	-FFS	+	+
	Segmentation	+	+	+	-	+	+
	Concatenation	+	+	+	-	+	+
Acknowledged Service	Applicability	-	-	-	-	+	+
	Segmentation	-	-	-	-	+	+
	Concatenation	-	-	-	-	+	+
	Flow Control	-	-	-	-	+	+
	Error Correction	-	-	-	-	+	+
	Protocol error correction & recovery	-	-	-	-	+	+

~~Table 4. RLC modes and functions in UTRAN uplink side~~

Table Error! Style not defined.-Error! Bookmark not defined.**: RLC modes and functions in UTRAN uplink side**

Service	Functions	CCCH	DCCH	DTCH
Transparent Service	Applicability	+	-	+
	Reassembly	-	-	+
Unacknowledged Service	Applicability	-FFS	+	+
	Reassembly	-	+	+
Acknowledged Service	Applicability	-	+	+
	Reassembly	-	+	+
	Error correction	-	+	+
	Flow Control	-	+	+
	In sequence delivery	-	+	+
	Duplicate detection	-	+	+
	Protocol error correction & recovery	-	+	+

7. Services expected from MAC

For a detailed description of the following functions see [3].

- Data transfer;
- Acknowledged data transfer service by MAC for transmission on RACH/FACH is FFS.

8. Elements for layer-to-layer communication

8.1. Primitives between RLC and higher layers

The primitives between RLC and upper layers are shown in Table 8.1-1.

~~Table 8.1-1 Primitives between RLC and upper layers~~

Table Error! Style not defined.-Error! Bookmark not defined. **: Primitives between RLC and upper layers**

Generic Name	Parameter			
	Req.	ind.	Resp.	conf.
RLC-AM-DATA	MU	MU	Not Defined	Not Defined
RLC-UM-DATA	MU, QR (ffs)	MU	Not Defined	Not Defined
RLC-TR-DATA	MU	MU	Not Defined	Not Defined
MRLC-CONFIGURE				
MRLC RELEASE			Not Defined	Not Defined

Each Primitive is defined as follows:

a) RLC-AM-DATA req./ind.
It is used for acknowledged data transmission mode of point-to-point connection between the same level user entities.

[Editor's note: Confirmation for the RLC-AM-DATA procedure is FFS.]

b) RLC-UM-DATA req./ind.
It is used for unacknowledged data transmission mode of point-to-point connection between the same level user entities.

c) RLC-TR-DATA req./ind.
It is used for trasparent data transmission mode of point-to-point connection between the same level user entities.

d) MRLC-CONFIGURE
FFS

e) MRLC RELEASE
FFS

The parameter Message Unit (MU) is mapped on MU field on RLC PDU transparently in the case of RLC-AM-DATA req. or RLC-UM-DATA req. And the MU field of RLC PDU received is mapped on MU in the case of RLC-AM-DATA ind. or RLC-UM-DATA ind. transparently. Length of MU must be n octets (n is integer).
The Quick Repeat indicator (QR) indicates whether UMD PDU will be transmitted with Quick Repeat or not. It holds one of two values: "Yes" or "No". *(The need of this indicator is FFS)*

9. Elements for peer-to-peer communication

In unacknowledged transmission, only one type of unacknowledged data PDU is exchanged between peer RLC entities In acknowledged transmission, both (acknowledged) data PDUs and control PDUs are exchanged between peer RLC entities.

9.1. Protocol data units

~~(List of PDU´s, encoding of PDU´s (if applicable~~))

Data PDU

a) AMD PDU (Acknowledged Mode Data PDU)

The AMD PDU is used to convey sequentially numbered PDUs containing RLC SDU data. The AMD PDU is used by the RLC when it is in the acknowledged mode.

b) UMD PDU (Unacknowledged Mode Data PDU)

The UMD PDU is used to convey sequentially numbered PDUs containing RLC SDU data. It is used by the RLC when using the unacknowledged data transfer.

Control PDU

a) BGN PDU (Begin)

The BGN PDU is used by a RLC entity in order to establish a RLC link between the entity and its peer entity.

b) BGAK PDU (Begin Acknowledge)

The BGAK PDU is an acknowledgement to the BGN PDU.

c) BGREJ PDU (Begin Reject)

The BGREJ PDU is used to reject the RLC link setup request of the peer RLC entity.

d) END PDU (End)

The END PDU is used by a RLC entity in order to release the RLC link between the entity and its peer entity.

e) ENDAK PDU (End Acknowledge)

The ENDAK PDU is an acknowledgement to the END PDU.

f) STAT PDU (Solicited Status Response) (FFS)

The STAT PDU is used to respond to a status request from the peer RLC entity.

g) USTAT PDU (Unsolicited Status Response) (FFS)

The USTAT PDU is transmitted upon detection of an erroneous transmission of one or more data PDUs. It is used to inform the transmitter side about missing AMD PDUs at the receiver RLC.

Table Error! Style not defined.-Error! Bookmark not defined. **: RLC PDU names and descriptions**

Functionality	PDU name	Description
Establishment	BGN	Request Initialization
	BGAK	Request Acknowledgement
	BGREJ	Connection Reject
Release	END	Disconnect Command
	ENDAK	Disconnect Acknowledgement
Acknowledged Data Transfer	AMD	Sequenced acknowledged mode data
	STAT	Solicited Status Report
	USTAT	Unsolicited Status Report
Unacknowledged Data Transfer	UMD	Sequenced unacknowledged mode data

9.2. Formats and parameters

~~(PDU formats, PDU parameters, information element encodings, other formats employed in specific functions, if applicable, e.g. if a segmentation of PDUs is performed and further protocol information added after segmentation, formats for multiplexing/demultiplexing)~~

[All the section shall be reviewed when the protocol is defined]

AMD PDU

Note: R bit may be H bit. It is FFS.

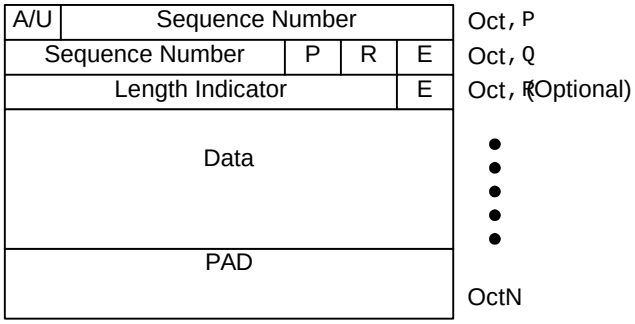


Figure Error! Style not defined.-Error! Bookmark not defined.**. AMD PDU**

UMD PDU

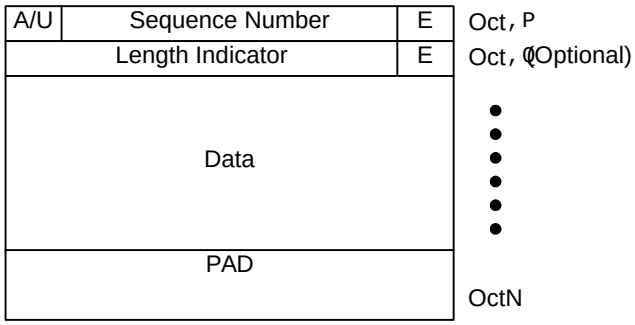


Figure Error! Style not defined.-Error! Bookmark not defined.**. UMD PDU**

BGN PDU

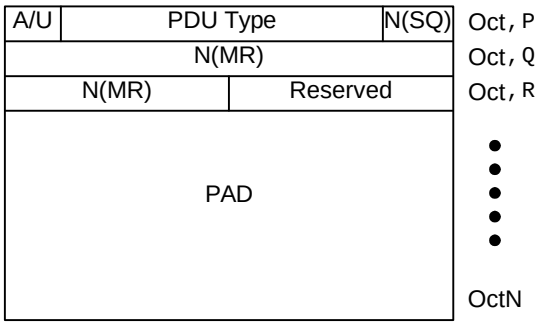


Figure Error! Style not defined.-Error! Bookmark not defined.**. BGN PDU**

BGAK PDU

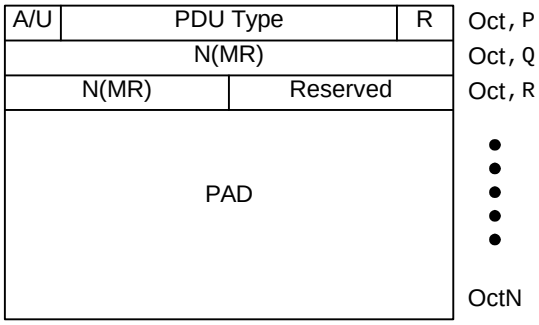


Figure Error! Style not defined.-Error! Bookmark not defined.**. BGAK PDU**

BGREJ, END, ENDAK PDU

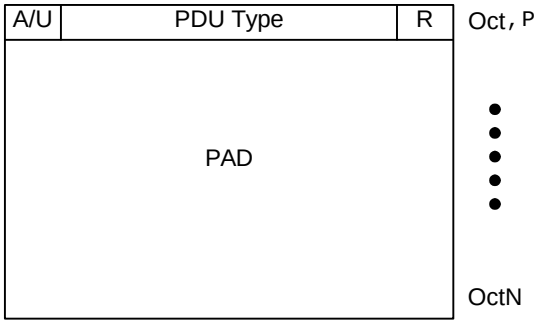


Figure Error! Style not defined.-Error! Bookmark not defined.**. BGREJ, END, ENDAK PDU**

STAT PDU

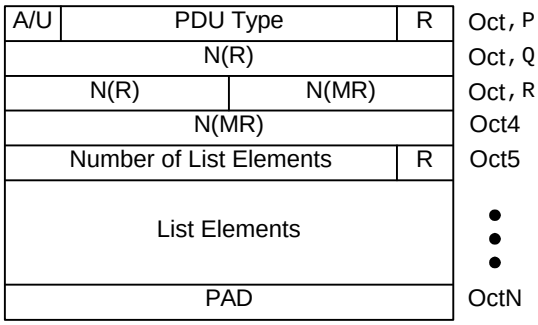


Figure Error! Style not defined.-Error! Bookmark not defined.**. STAT PDU**

USTAT PDU

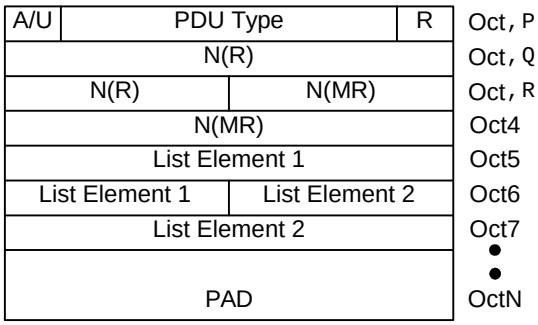


Figure Error! Style not defined.-Error! Bookmark not defined.**. USTAT PDU**

Note: Regarding STAT and USTAT, it is FFS. whether a bitmap type of PDU status indication would be more efficient than List elements.

The RLC PDU parameters are defined as follows:

- A/U bit: 1bit
  This field indicates Acknowledged mode data PDU or Unacknowledged mode data PDU/ Control PDU. If it indicates Acknowledged mode, the PDU is AMD PDU.

Bit	Description
0	Unacknowledged mode data PDU/ Control PDU
1	Acknowledged mode data PDU

- PDU Type: 6bit
  This field indicates the type of Control PDU. They are indicated by the special values of sequence number field.

Bit	PDU Type	Bit	PDU Type
111111	BGN	111010	STAT
111110	BGAK	111001	USTAT
111101	BGREJ	111000 – 110000	Reserved
111100	END		
111011	ENDAK		

- Sequence Number (SN)
  This field indicates the sequence number of the RLC PDU.

PDU type	Length	Notes
AMD PDU	12 bit	Used for retransmission and reassembly
UMD PDU	6 bit	Used for reassembly ***Especially "110000" – "111111" are reserved for PDU Type (Control PDU)***

- Polling bit (P): 1bit
  This field is used to request a status report (STAT PDU) from the receiver RLC.

Bit	Description
0	-
1	Request a status report

- Extension bit (E): 1bit
  This bit indicates whether the next octet will be header information (LI) or data.

Bit	Description
0	The next octet is data
1	The next octet is header information (LI)

- Reserved (R):
  One function of this field is to achieve octet alignment. Other functions are FFS. Where no functions are defined, this field shall be coded as zero. This field ignored by the receiver.

- Length Indicator (LI): 7bit
  This field is optional and is used if concatenation or padding takes pRLCe. It indicates the end of the last segment of a SDU. Especially "0000000" indicates that the previous RLC PDU is exactly filled with the last segment of a RLC SDU, and "1111111" indicates that the rest part of the RLC PDU is padding.

- N(SQ): 1bit
  This field carries the connection sequence value. VT(SQ) is mapped to N(SQ) whenever a new BGN PDU is transmitted. This field is used by the receiver together with VR(SQ) to identify retransmitted BGN PDU.

- N(R): 12bit

VR(R) is mapped to N(R) whenever a STAT or USTAT PDU is generated.

- N(MR): 12bit
  VR(MR) is mapped to N(R) whenever a STAT, USTAT, BGN, or BGAK PDU is generated. This is the basis for credit granting by the receiver.

- Number of List Elements: 7bit
  It indicates the number of list elements that included in the STAT PDU.

- Header extension flag (H): 1bit

This header extension flag indicates that there is an additional control part (SN+H+E) in an acknowledged mode RLC PDU header. The use of this flag is FFS

- Data:
  In this field data from higher layer PDUs is mapped.

9.3. Protocol states

~~(Description of states, provision of state transition diagram(s))~~

This sub-section describes the states of a RLC entity. These states are used in the specification of the peer-to-peer protocol. The states are conceptual and reflect general conditions of the RLC entity in the sequences of signals and PDU exchanges with its user and peer, respectively. In addition, other conditions are used in the description, in order to avoid identification of additional states, as detailed in the SDLs. The basic states are:

– State 1 – Idle
Each RLC entity is conceptually initiated in the Idle state (State 1) and returns to this state upon the release of a connection.

– State 2 – Outgoing Connection Pending
A RLC entity requesting a connection with its peer is in the Outgoing Connection Pending state (State 2) until it receives acknowledgment from its peer.

– State 3 – Incoming Connection Pending
A RLC entity that has received a connection request from its peer and is waiting for its user's response is in the Incoming Connection Pending (State 3).

– State 4 – Outgoing Disconnection Pending
A RLC entity requesting release of the peer-to-peer connection goes to the Outgoing Disconnection Pending state (State 4) until it receives confirmation that the peer entity has released and transited to the Idle state (State 1), after which it does the same.

– State 5 – Data Transfer Ready
Upon successful completion of the connection establishment procedures, both peer RLC entities will be in Data Transfer Ready state (State 5) and acknowledged data transfer can take place.

9.4. State variables

This sub-clause describes the state variables used in the specification of the peer-to-peer protocol. AMD PDUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{12} and the sequence numbers cycle through the entire range, 0 through 2^{12} – 1. All arithmetic operations on the following state variables and sequence numbers contained in this Recommendation are affected by the modulus: VT(S), VT(A), VT(MS), VR(R), VR(H), and VR(MR). When performing arithmetic comparisons of transmitter variables, VT(A) is assumed to be the base. When performing arithmetic comparisons of receiver variables, VR(R) is assumed to be the base. In addition, the state variables VT(SQ) and VR(SQ) use modulo 2 arithmetic and VT(US) and VT(UR) use modulo 48. The RLC maintains the following state variables at the transmitter.

a) VT(S) - Send state variable
The sequence number of the next AMD to be transmitted for the first time (i.e. excluding retransmission). Incremented after transmission of a AMD for the first time (i.e. excluding retransmission).

b) VT(A) - Acknowledge state variable
The sequence number of the next in-sequence AMD PDU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgments. VT(A) is updated upon acknowledgment of in-sequence AMD PDUs.

c) VT(DAT)
This state variable is used to count the retransmission number of each AMD PDU. VT(DAT) is incremented by sending AMD.

d) VT(MS) - Maximum Send state variable
The sequence number of the first AMD PDU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1]. This value represents the upper edge of the transmit window. The transmitter shall not transmit a new AMD PDU if VT(S) = VT(MS). VT(MS) is updated based on receipt of a USTAT PDU, STAT PDU, BGN PDU, BGAK PDU.

e) VT(CC) - Connection Control state variable
The number of unacknowledged BGN, END PDUs. VT(CC) is incremented upon transmission of a BGN, END PDU. If an END PDU is transmitted in response to a protocol error, RLC does not wait for an ENDAK PDU [i.e. RLC moves directly to state 1 (Idle)] and VT(CC) is not incremented.

f) VT(SQ) - Transmitter Connection Sequence state variable
This state variable is used to allow the receiver to identify retransmitted BGN PDUs. This state variable is initialized to 0 upon creation of the RLC process and incremented and then mapped into the N(SQ) field before the initial transmission of either a BGN PDU.

g) VT(US) - Unit data state variable
This state variable means new sequence number of UMD-PDU which will send next. After new UMD-PDU is sent, VT(US) will be incremented.

h) VT(QR) - Quick repeat state variable (FFS)
This state variable is used to count the retransmission number when UMD-PDU is sent by quick repeat scheme. It is incremented after UMD-PDU is sent and quick repeat will be continued until VT(QR) becomes to equal MaxQR.

The RLC maintains the following state variables at the receiver:
a) VR(R) - Receive state variable
The sequence number of the next in-sequence AMD PDU expected to be received. Incremented upon receipt of the next in-sequence AMD PDU.

b) VR(H) - Highest expected state variable
The sequence number of the next highest expected AMD PDU. This state variable is updated whenever a new AMD PDU is received.

c) VR(MR) - Maximum acceptable Receive state variable
The sequence number of the first AMD PDU not allowed by the receiver [i.e. the receiver will allow up to VR(MR) – 1]. The receiver shall discard AMD PDUs with N(S) = VR(MR), (in one case, such an AMD PDU may cause the transmission of a USTAT). Updating VR(MR) is implementation dependent, but VR(MR) should not be set to a value < VR(H).

d) VR(SQ) - Receiver Connection Sequence state variable
This state variable is used to identify retransmitted BGN PDUs. Upon reception of a BGN PDU, this state variable is compared to the value of N(SQ) and then assigned the value of N(SQ). If the values are different, the PDU is processed and VR(SQ) is set to N(SQ). If they are equal, the PDU is identified as a retransmission.

e) VR(US) - Receiver Send Sequence state variable
The sequence number of the latest UMD PDU to be received. It is used to check the duplication receive. When new UMD PDU is received, VR(US) is compared with N(US). If VR(US),N(US), this PDU is quashed because duplication receive happens. And if not, N(US) is substituted for VR(US).
f) VR(EP) – Estimated PDU Counter state variable (FFS)
The number of PDUs that should have been received after the latest USTAT was sent. In acknowledged mode, this state variable is updated at the end of each transmission time interval. It is incremented by the number of PDUs that should have been received during the transmission time interval. If VR(EP) is equal to the number of requested PDUs in the latest USTAT, then check if all PDUs requested for retransmission have been received.

9.5. Timers

a) Timer_STAT
It is used to detect the loss of the response from receiver side. This timer is set when transmitted AMD PDU requests status report (i.e. P bit is set to “1”). And it will be stopped when the transmitter receive Acknowledgement

of the AMD PDU by STAT PDU or USTAT PDU. When this timer is over, the oldest unconfirmed AMD PDU should be retransmitted with requesting status report, and this timer is set again.

b) Timer_Prohibit

It is used to prohibit transmission of polling message within a certain period. It prohibits only the polling of every RLC SDU. For other polling trigger, even if this timer is active, polling message can be transmitted. This timer is set when AMD PDU with polling is transmitted. When this timer is over no action is performed. (T_STAT =< T_Prohibit)

c) Timer_CC

Timer_CC protects the transmission of PDU between connection establishment and connection release, during re-synchronization or during error recovery. Timer_CC is indicates retransmission interval when confirmation isn't received against BGN PDU and ENDPDU. The value of Timer_CC should be a little larger than the round-trip delay.

d) Timer_QR (FFS)

Transmission interval of quick repeat for UMD PDU.

e) Timer_EPC (FFS)

This timer accounts for the roundtrip delay, i.e. the time when the first retransmitted PDU should be received after a USTAT/USTAT has been sent. The value of Timer_EPC is heavily based on the transmission time interval (corresponding to the Layer 1 interleaving depth). When changing the transmission time interval, then the value of the EPC timer also needs to be changed.

9.6. Protocol Parameters

~~(e.g. max, min values of state variables to be initialized)~~

The value of each RLC protocol parameter is application specific and may be defined in another Recommendation which references this Recommendation.

a) MaxCC

Maximum value for the state variable VT(CC), corresponding to the maximum number of transmissions of a BGN, END.

b) MaxDAT

It is Maximum value for the number of retransmission of AMD PDU. This parameter is an upper limit of counter VT(DAT). When the value of VT(DAT) comes to MaxDAT, error recovery procedure will be performed.

c) MaxQR

Maximum successive transmission number of UMD PDU. This parameter is an upper limit for counter VT(QR).

d) MaxSTAT

Maximum number of list elements placed in a STAT PDU. When the number of list items exceeds MaxSTAT, the STAT message shall be segmented. All of the PDUs carrying the segmented STAT message, except possibly the last one, contain MaxSTAT list items. This parameter is not used by the receiver of a STAT PDU for length checking, but is only used by the sender of the STAT message for segmentation purposes. This parameter should be an odd integer greater than or equal to 3.

e) Credit

This parameter is used to coordinate credit notifications to layer management. When RLC is blocked from transmitting a new AMD PDU due to insufficient credit, "Credit" is assigned the value "No". When RLC is permitted to transmit a new AMD PDU, "Credit" is assigned the value of "Yes". Credit is initially assigned "Yes".

9.7. Specific functions

~~(description of specific protocol functions, if applicable, e.g. flow control)~~

[All the section shall be reviewed when the protocol is defined]

9.7.1. Credit and peer-to-peer flow control

Credit is granted by the RLC receiver to allow the peer RLC transmitter to transmit new AMD PDUs. The process by which a receiver entity determines credit is not subject to standardization, but is related to the buffer availability and the bandwidth/delay of the connection.

Details of the usage of Crediting is FFS.

9.2 Local flow control

RLC events, such as reception of PDUs and external and internal signals, are normally processed in the order in which they occurred. However, events pertaining to the exchange of RLC link status information have priority over data transfer.

An implementation may detect congestion (for example, a long queuing delay) in its lower protocol layers. If so, data transfer should be temporarily suspended in order to give priority to connection control messages. The means by which an RLC entity decides whether or not it is congested depends on the protocol environment, including protocol timer values, and is not subject to standardization.

If a RLC entity detects local congestion ("lower layer busy" in the SDL specification), it can elect to suspend the servicing of RLC-AM-DATA.request, RLC-UM-DATA.request It can also suspend the retransmission of requested AMD PDUs. The data transfer procedures allow this to occur without causing protocol errors.

Therefore, in terms of transmitting PDUs to the peer receiver, all types of PDUs except AMD PDU and UMD PDU are given highest priority. The AMD PDUs and UMD PDUs have equal priority. Among the AMD PDUs, retransmission have priority over new transmission if both types are pending. These priorities are only internal to RLC.

10. Handling of unknown, unforeseen and erroneous protocol data

11. Elementary procedures

(Examples: idle, data transfer, RLC connection setup, RLC connection release, re-synchronisation)

12. SDL diagrams

The resultant SDL diagrams (Timer Prohibit scheme) are followed is shown in ANNEX 1.
Estimated PDU Counter (EPC) scheme (receiving side) (FFS)
Send a status report (USTAT), requesting for the retransmission of K number of missing PDUs.
Start Timer EPC. This timer accounts for the roundtrip delay, i.e. the time when the first retransmitted PDU should be received.
When the timer expires, start counting the received PDUs, or rather the PDUs that should have been received using the state variable VT(EP)
If VT(EP) = K, then check if all PDUs (requested in the status report in step 1) have been received.
If some of the previously missing PDUs are still missing, then repeat the procedure from step 1 for the PDUs that are still missing.
If none of the previously missing PDUs are still missing, then no status report needs to be sent, unless a poll had been transmitted or a new missing PDU has been detected. In case of a poll or a new missing PDU, then repeat the procedure from step 1.
Every poll received during the time when the Timer EPC is active and VT(EP) < K will be discarded by the receiving side, i.e. neither STATs nor USTATs will be sent from the receiving side during this time.

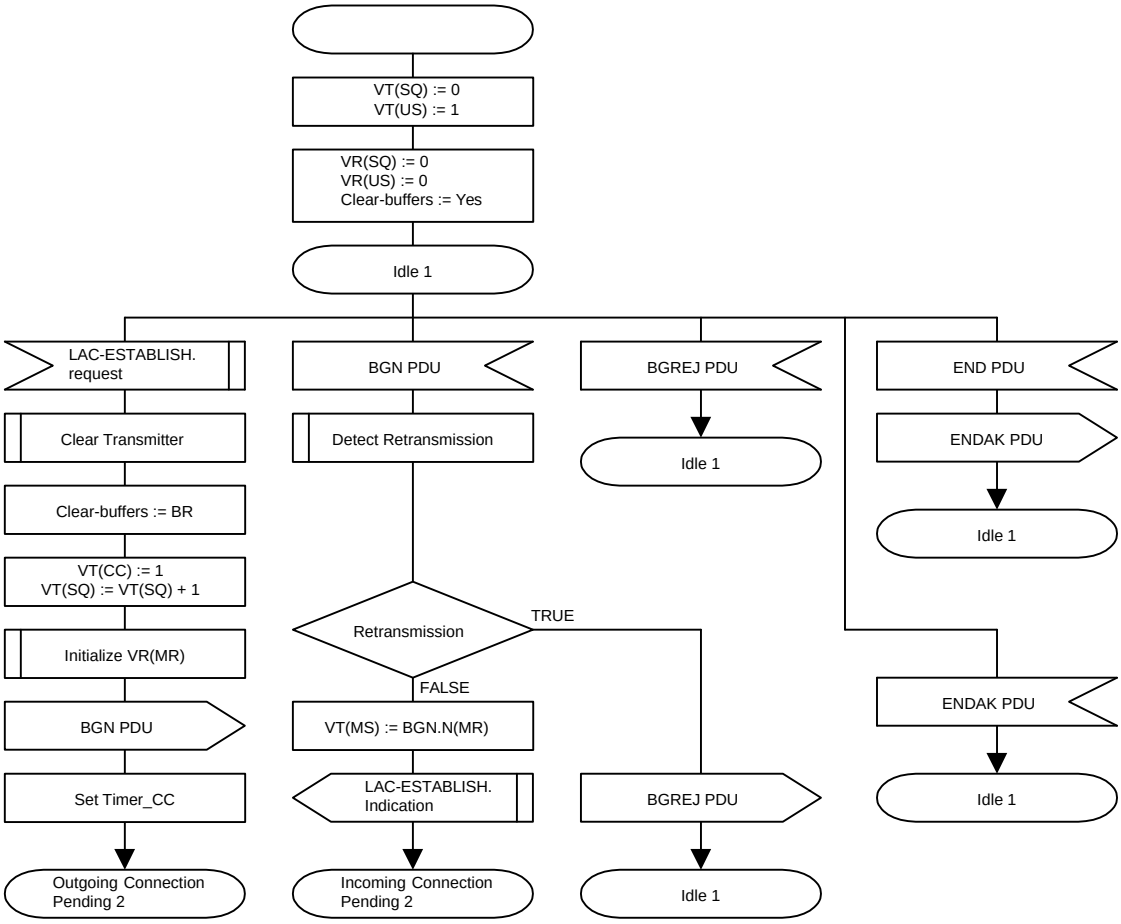


Figure Error! Style not defined.-Error! Bookmark not defined.

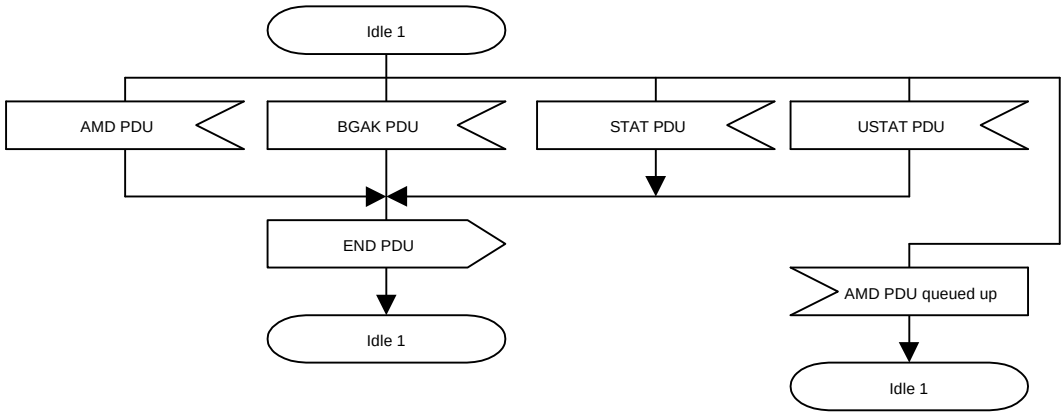


Figure Error! Style not defined.-Error! Bookmark not defined.

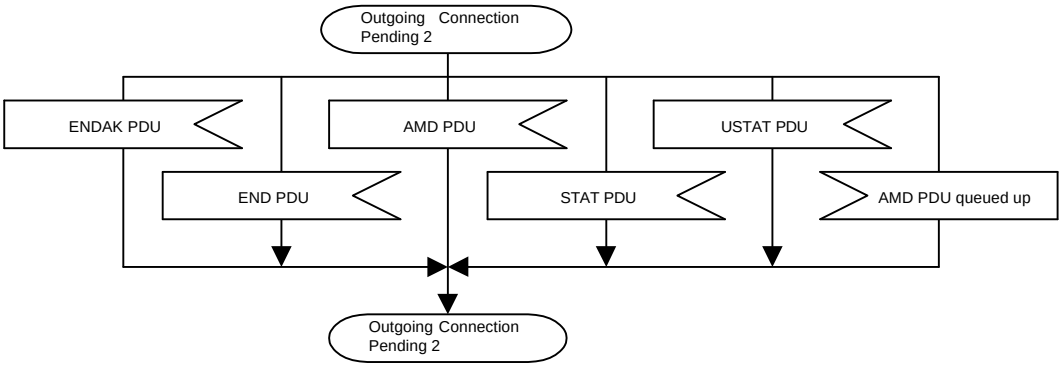


Figure Error! Style not defined.-Error! Bookmark not defined.

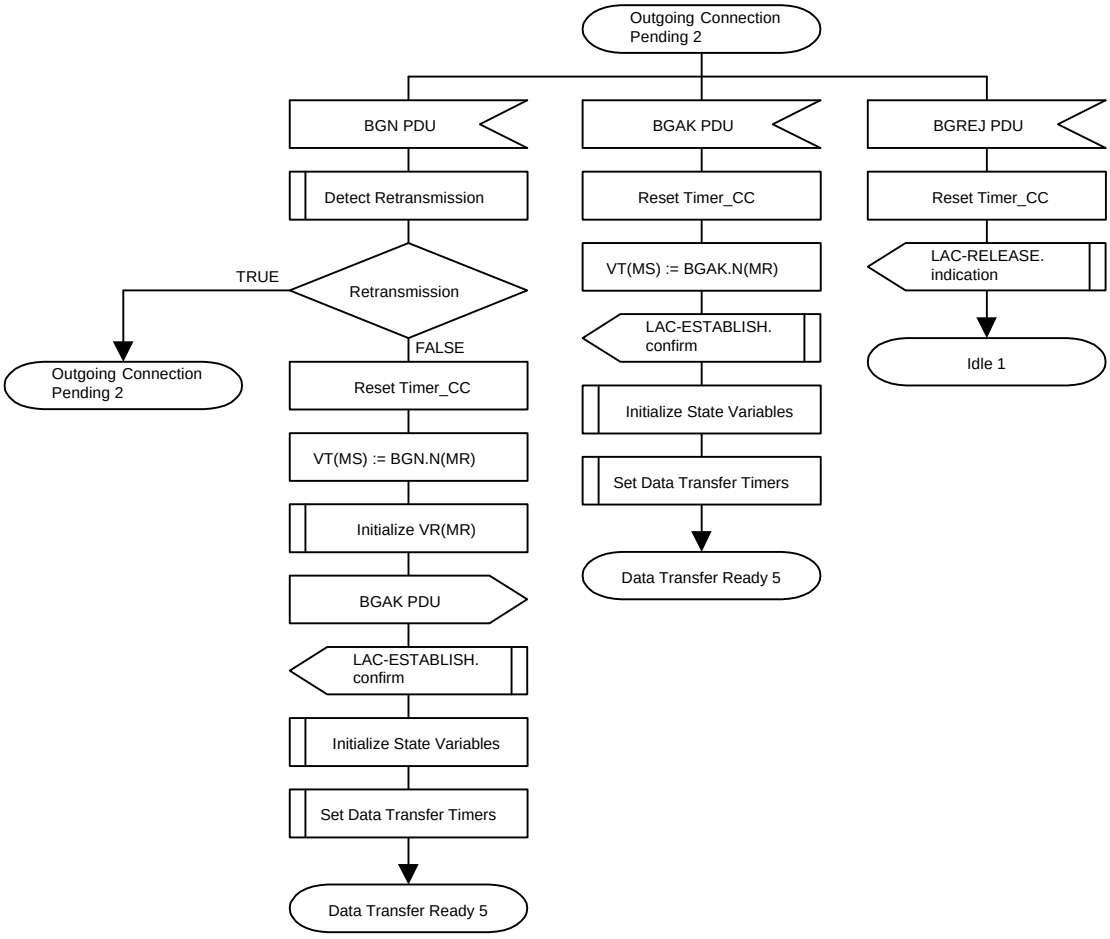


Figure Error! Style not defined.-Error! Bookmark not defined.

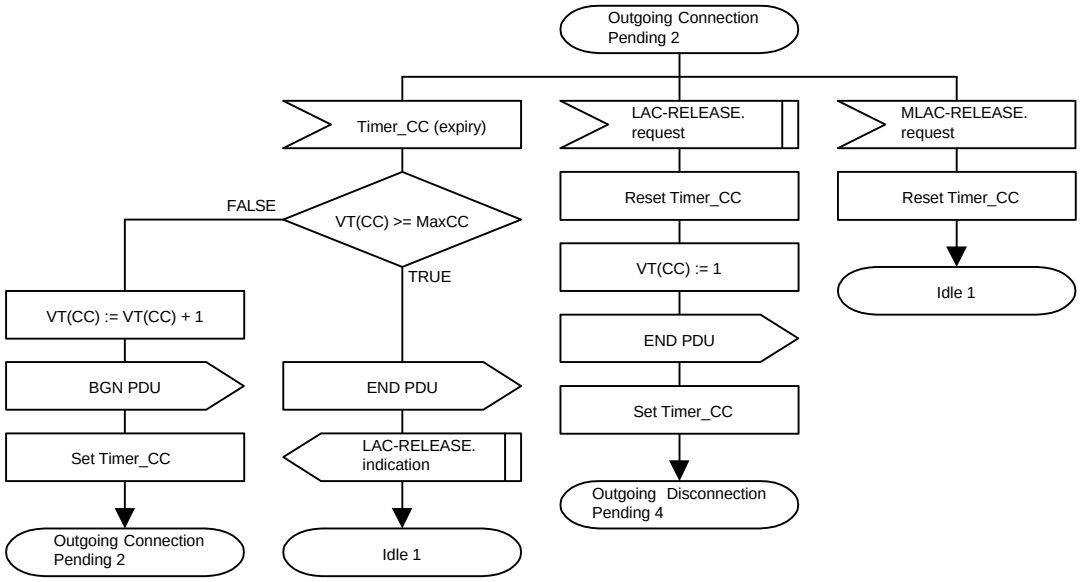


Figure Error! Style not defined.-Error! Bookmark not defined.

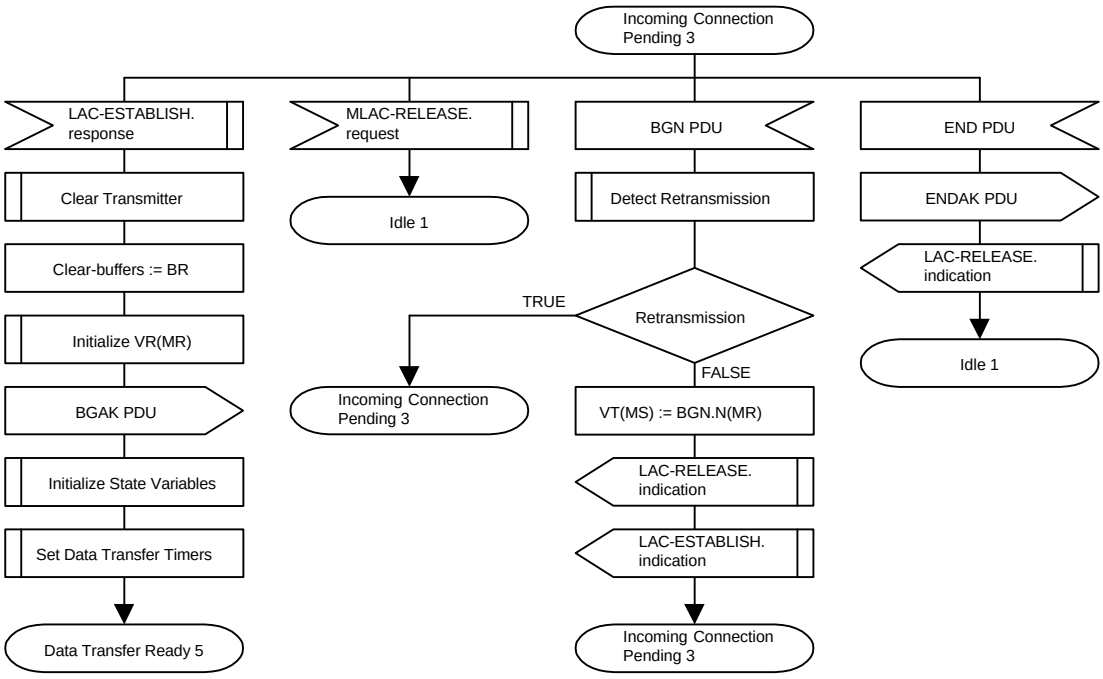


Figure Error! Style not defined.-Error! Bookmark not defined.

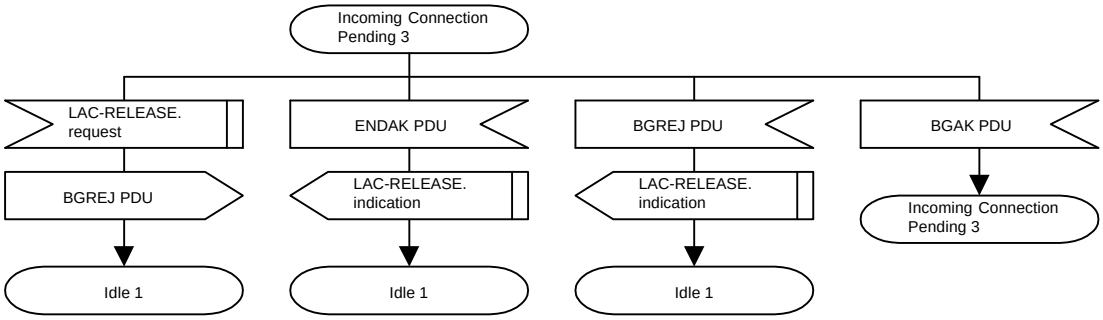


Figure Error! Style not defined.-Error! Bookmark not defined.

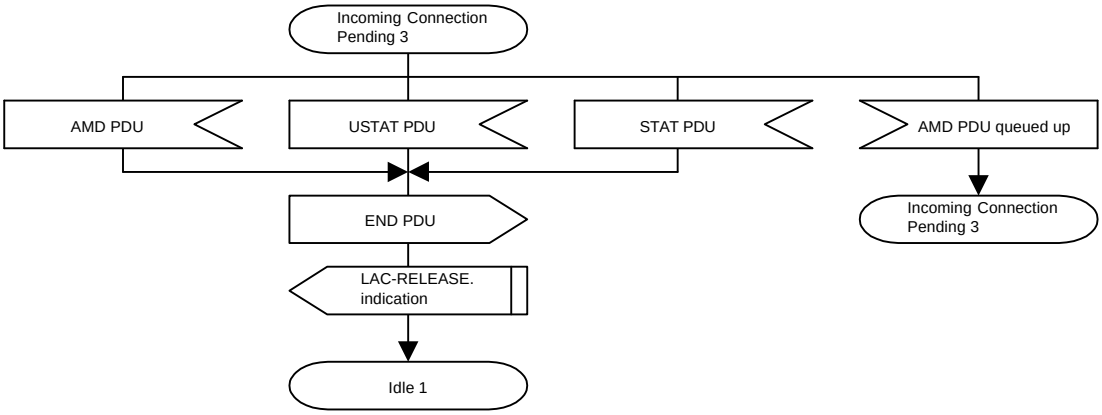


Figure Error! Style not defined.-Error! Bookmark not defined.

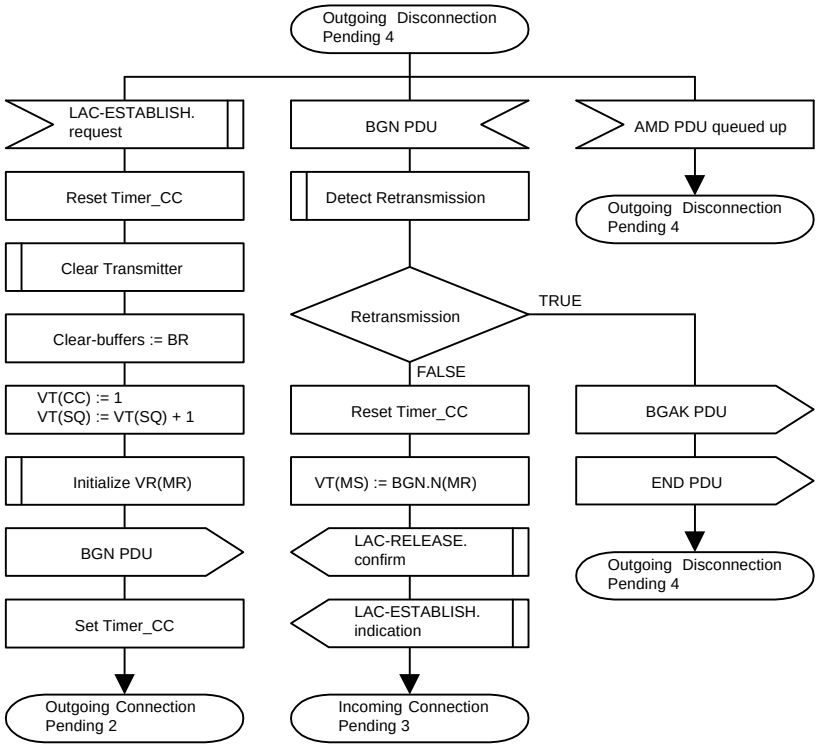


Figure Error! Style not defined.-Error! Bookmark not defined.

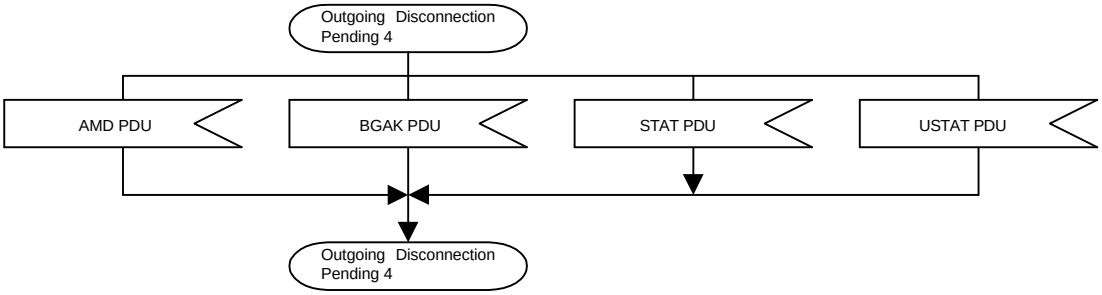


Figure Error! Style not defined.-Error! Bookmark not defined.

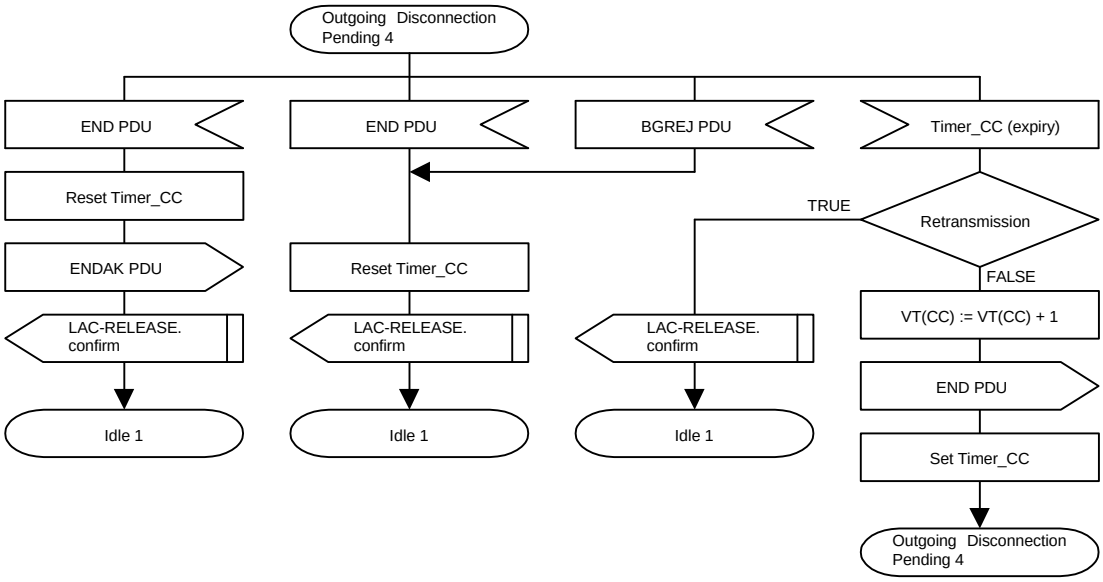


Figure Error! Style not defined.-Error! Bookmark not defined.

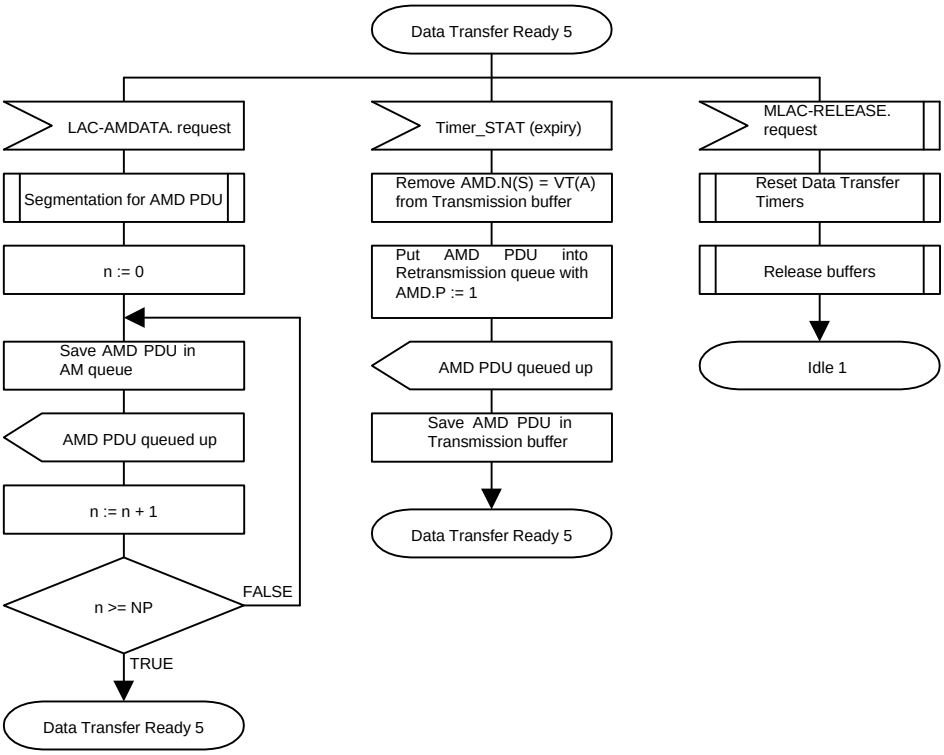


Figure Error! Style not defined.-Error! Bookmark not defined.

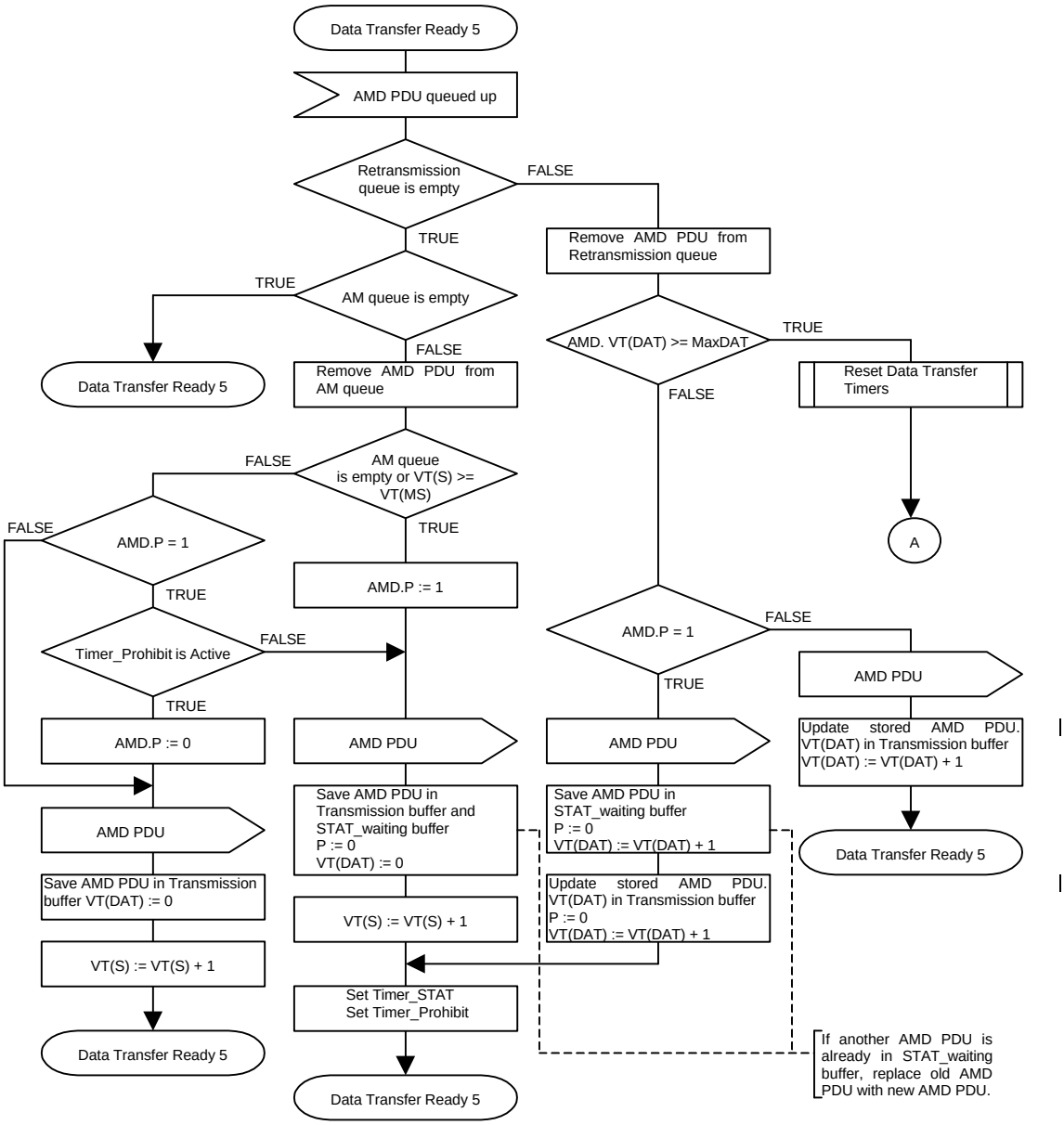


Figure Error! Style not defined.-Error! Bookmark not defined.

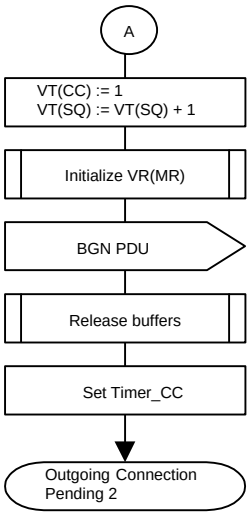


Figure Error! Style not defined.-Error! Bookmark not defined.

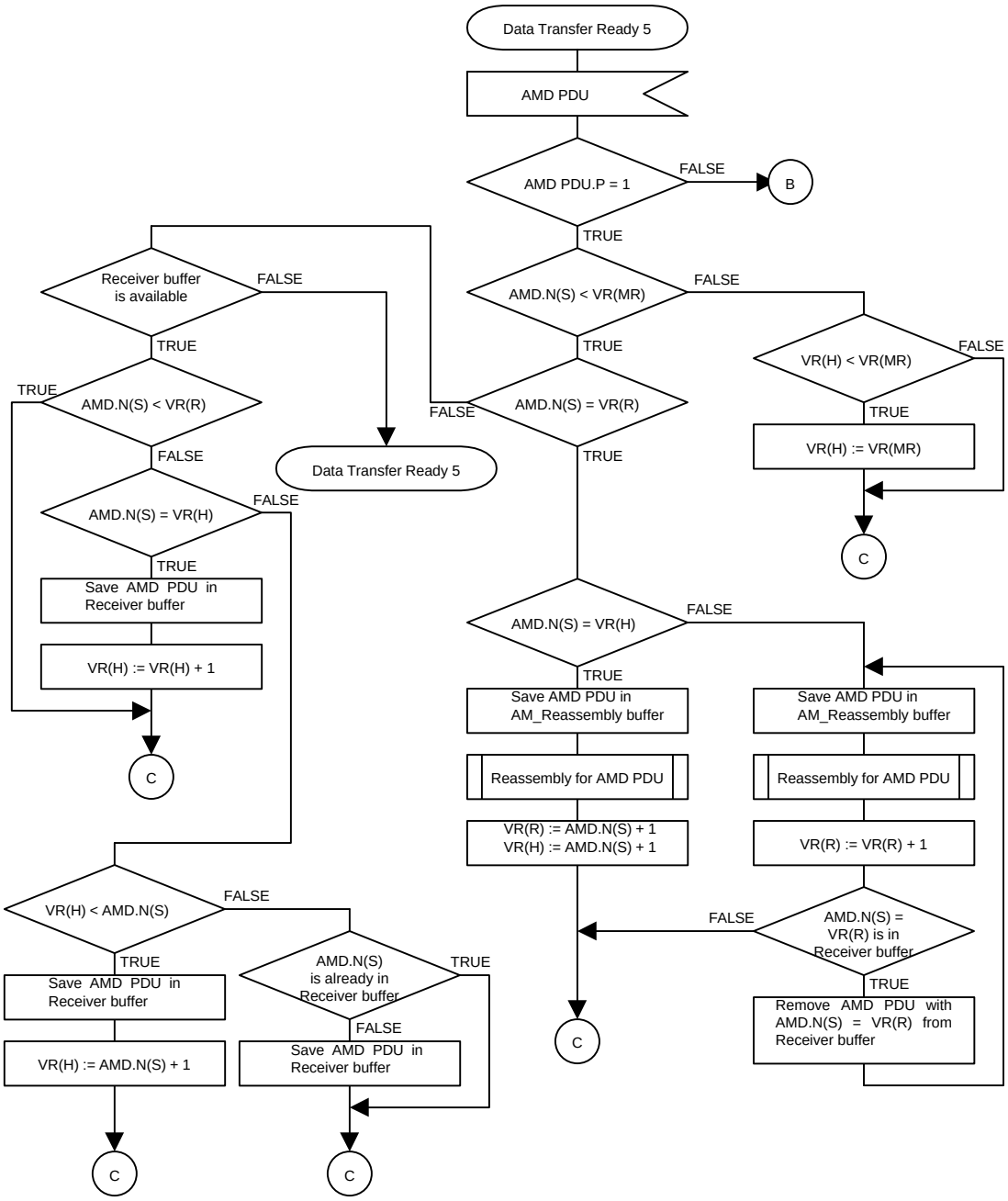


Figure Error! Style not defined.-Error! Bookmark not defined.

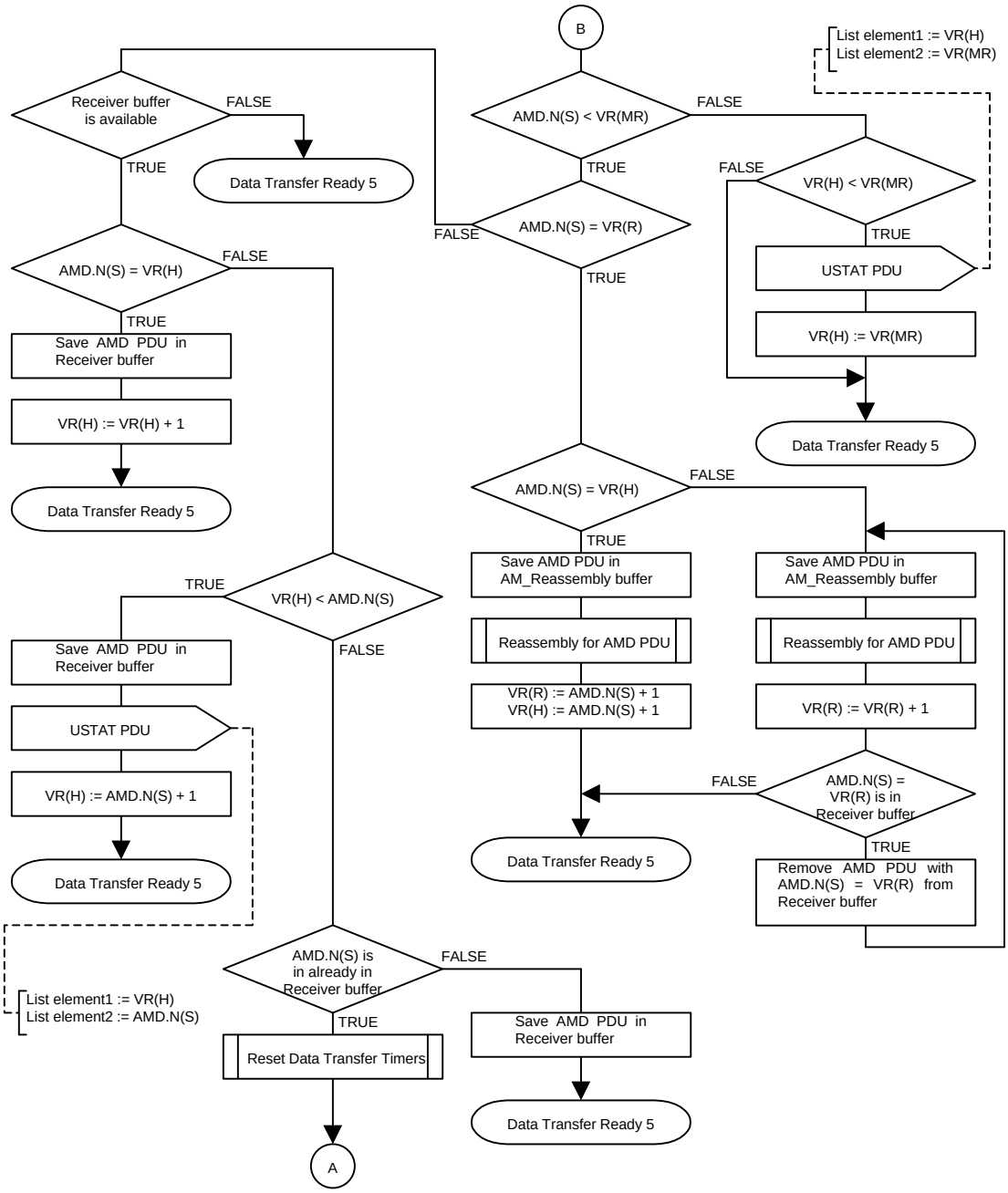


Figure Error! Style not defined.-Error! Bookmark not defined.

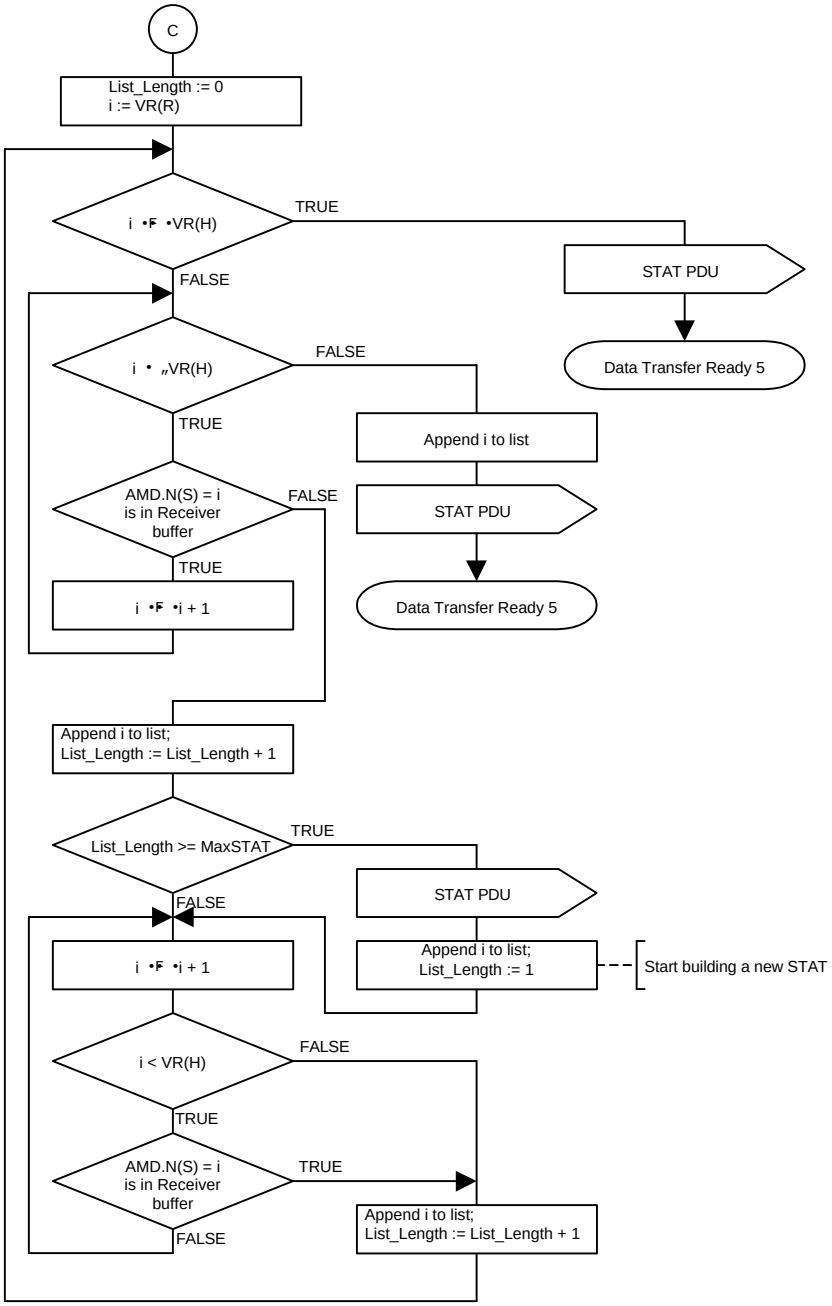


Figure Error! Style not defined.-Error! Bookmark not defined.

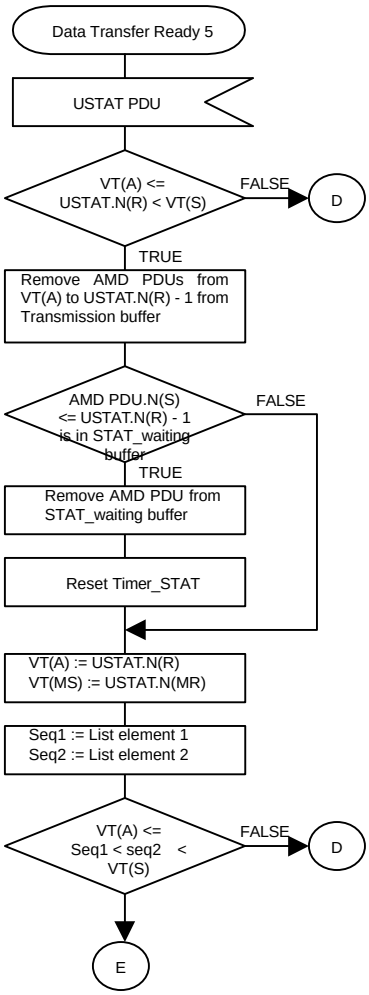


Figure Error! Style not defined.-Error! Bookmark not defined.

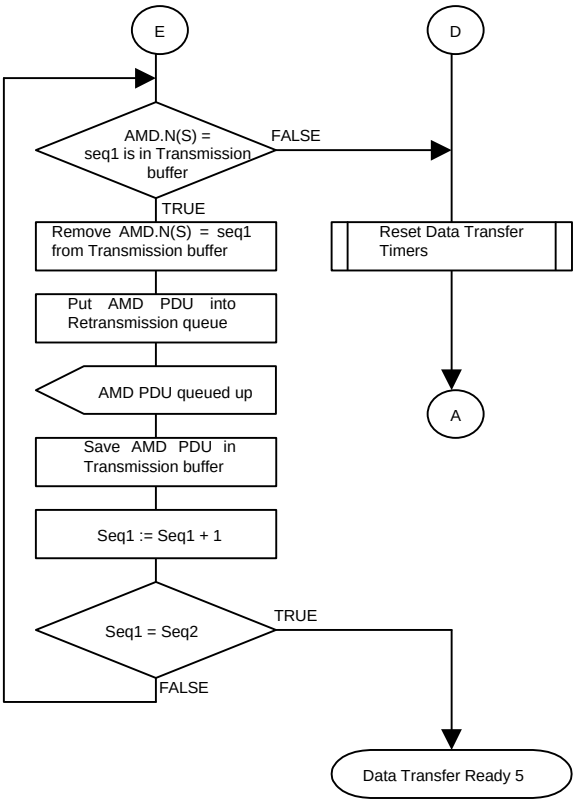


Figure Error! Style not defined.-Error! Bookmark not defined.

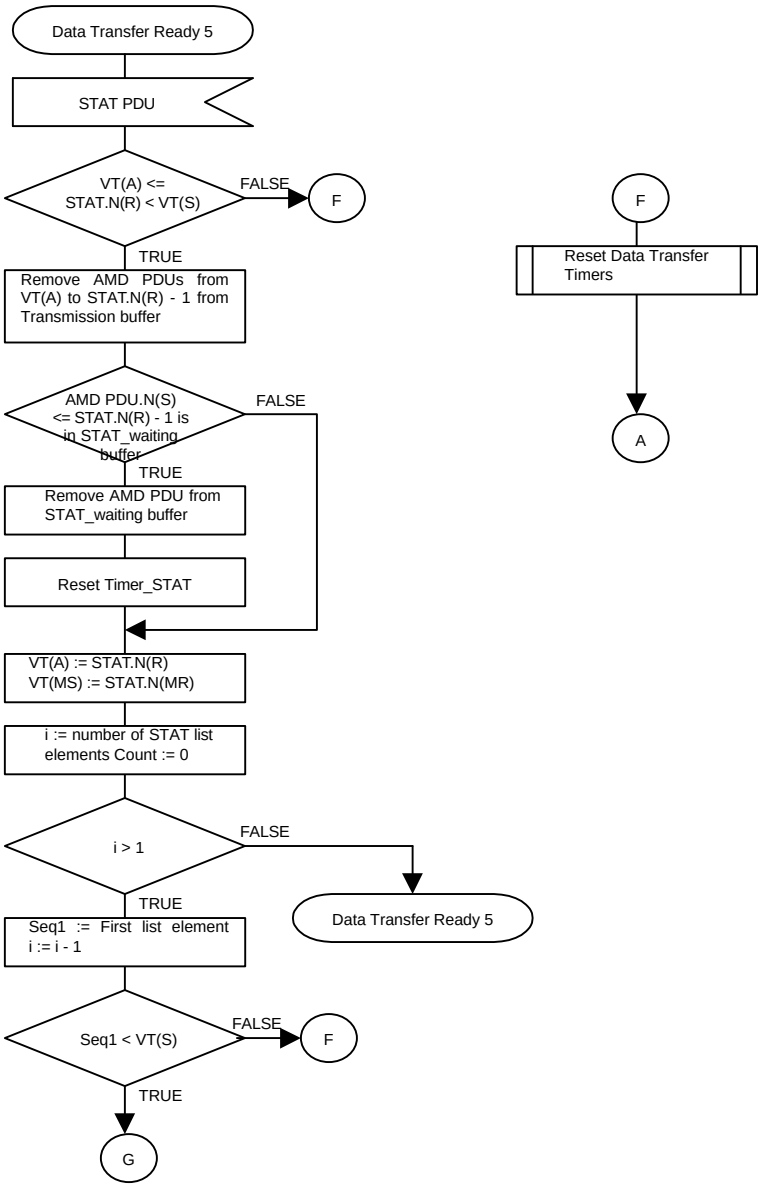


Figure Error! Style not defined.-Error! Bookmark not defined.

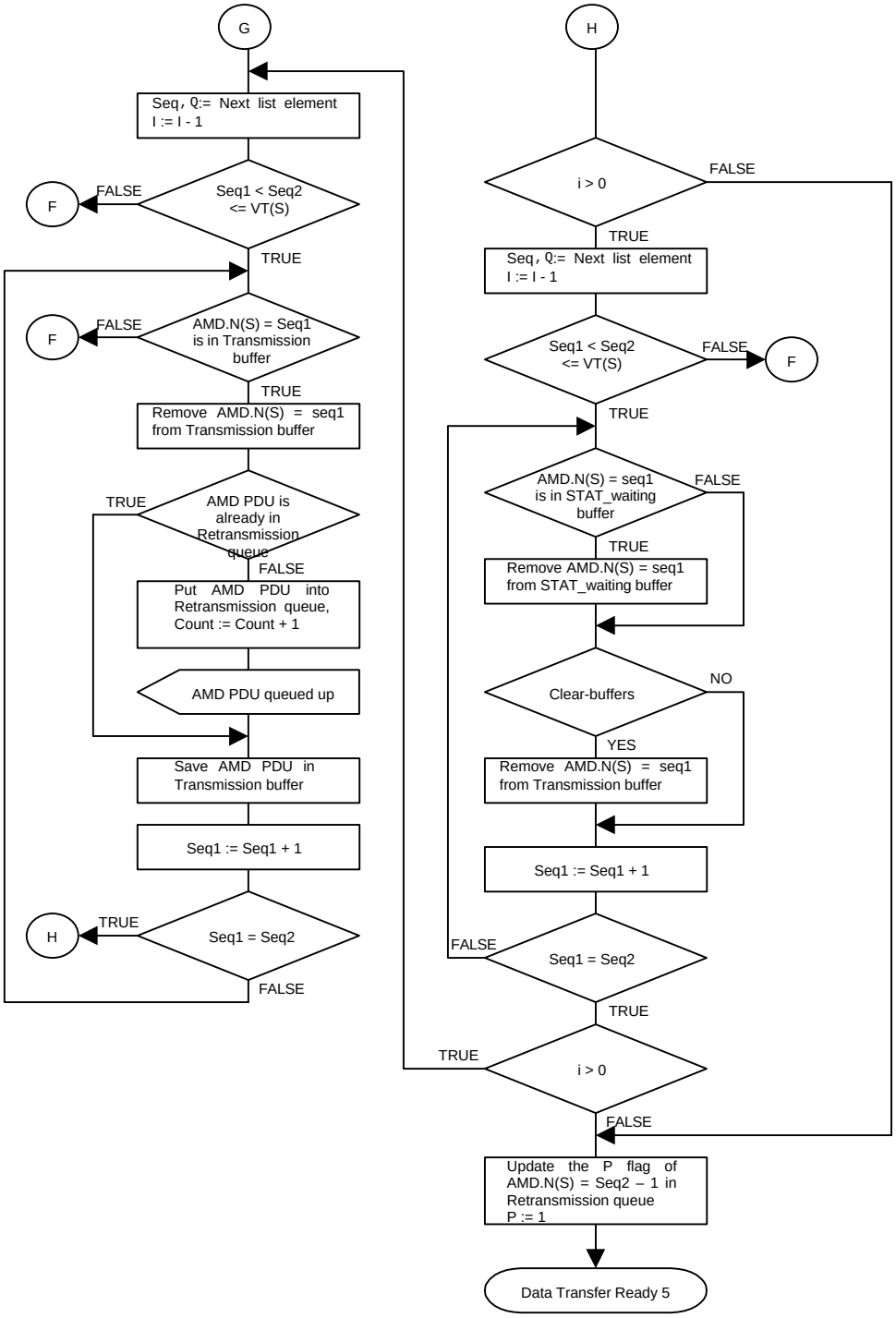


Figure Error! Style not defined.-Error! Bookmark not defined.

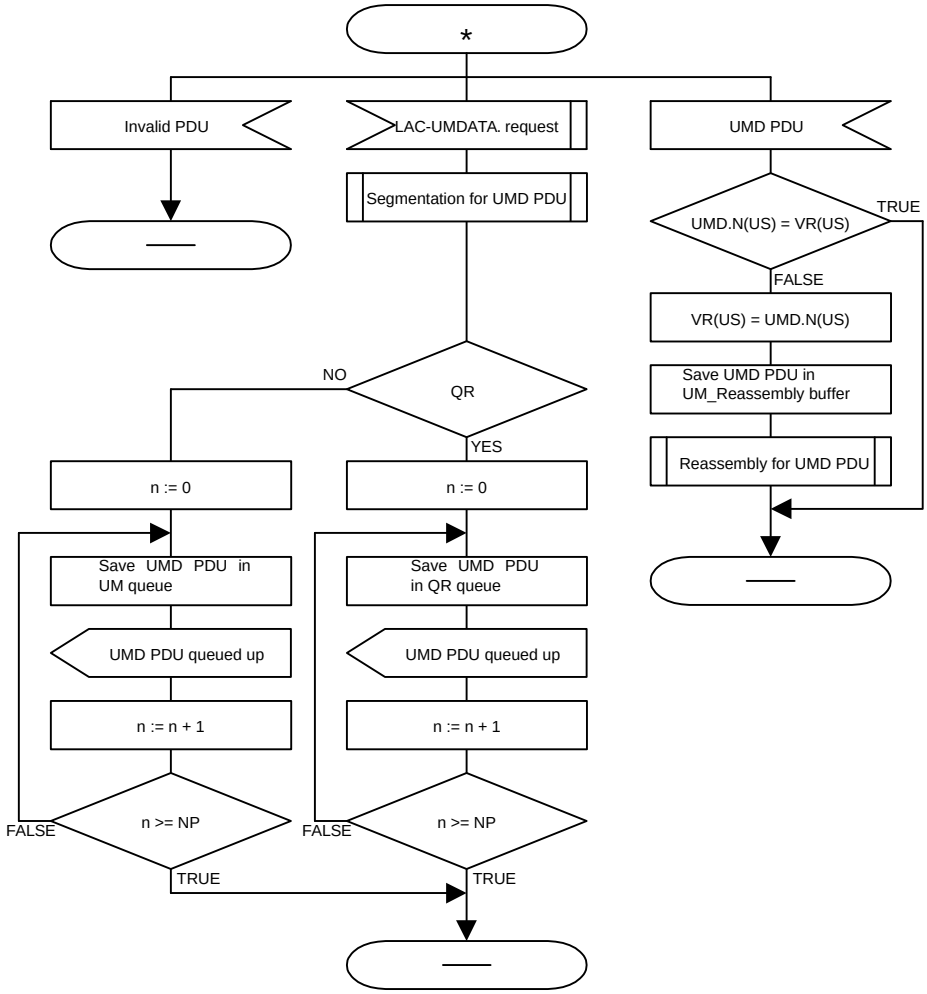


Figure Error! Style not defined.-Error! Bookmark not defined.

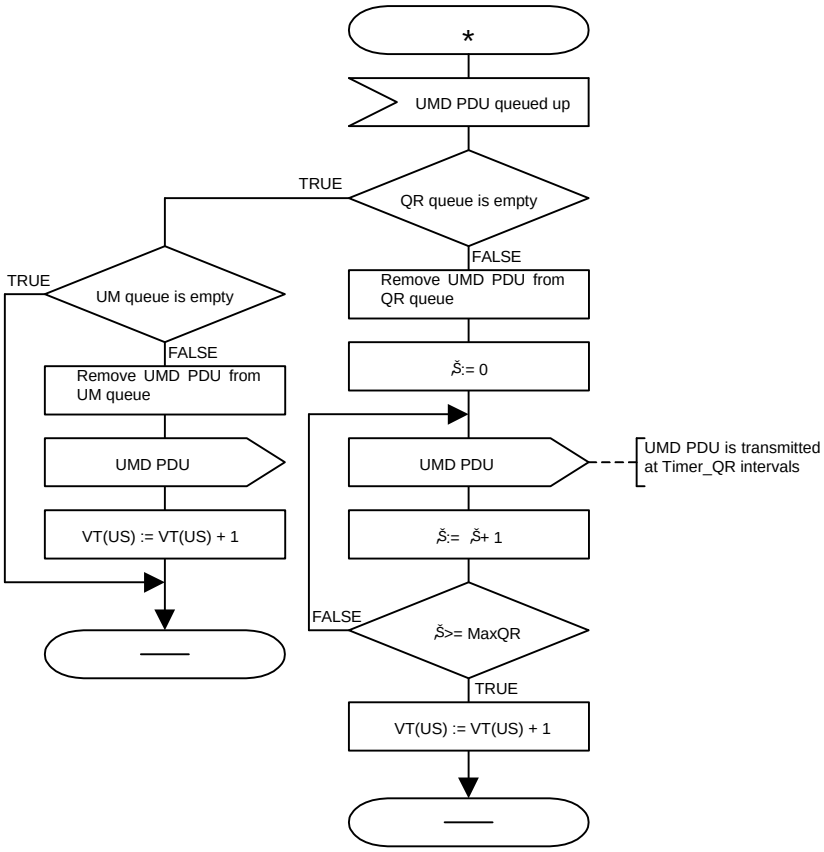


Figure Error! Style not defined.-Error! Bookmark not defined.

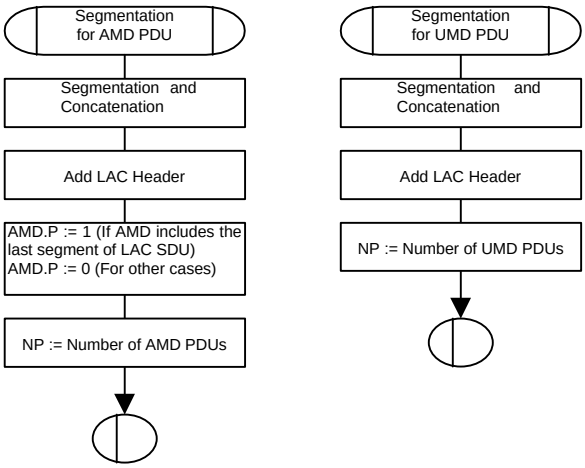


Figure Error! Style not defined.-Error! Bookmark not defined.

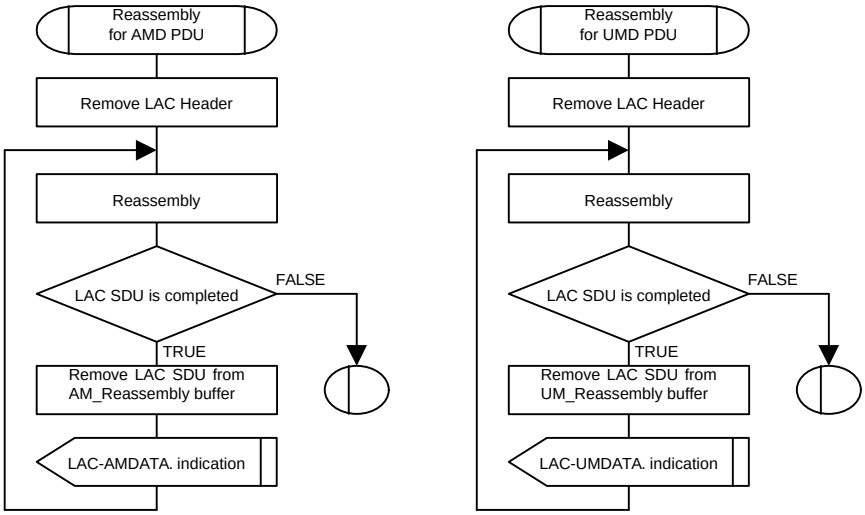


Figure Error! Style not defined.-Error! Bookmark not defined.

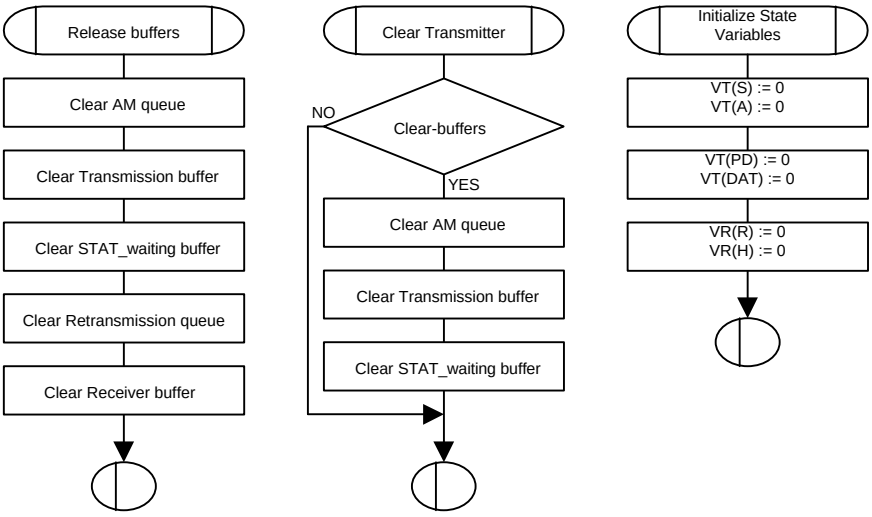


Figure Error! Style not defined.-Error! Bookmark not defined.

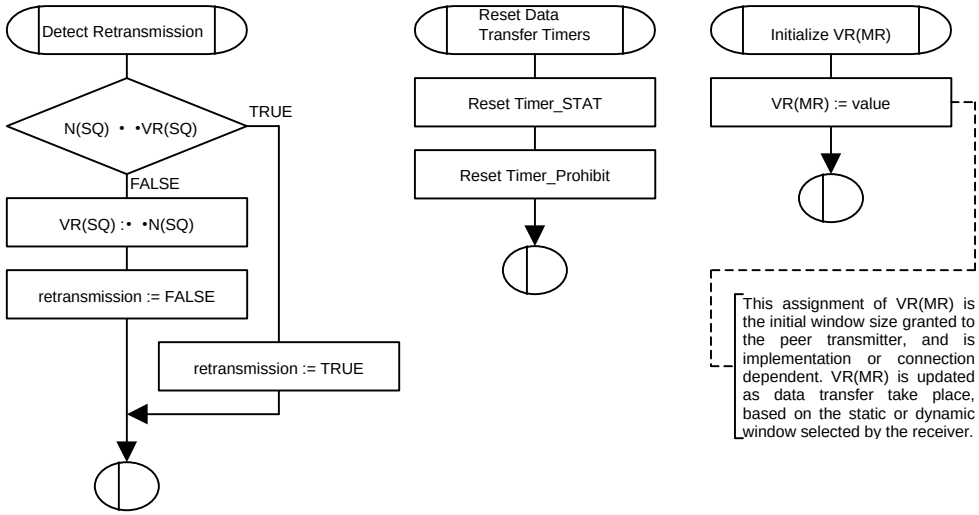


Figure Error! Style not defined.-Error! Bookmark not defined.

Appendix

1. Recommended values

1.1 PDU length

The length of the data field in AMD / UMD PDUs is k (>=0) octets.

1.2 MaxCC

4

1.3 MaxDAT

[FFS]

1.4 MaxQR

[FFS]

1.5 MaxSTAT

This parameter should be an odd integer greater than or equal to 3.

1.6 Timer_STAT

[FFS]

1.7 Timer_Prohibit

[FFS]

1.8 Timer_CC

1 sec

1.9 Timer_QR

[FFS]

13. . History

Document history		
Date	**Version**	**Comment**
January 1999	0.0.1	Document created. Based on TSG RAN WG2 Tdoc 016/99, 006/99 and 021/99..

Rapporteur (temporarily) for 3GPP S2.22 is:

Riccardo Santaniello
CSELT

Tel. : +39 011 228 7422
Fax : +39 011 228 7055

Email : riccardo.santaniello@cselt.it

This document is written in Microsoft Word version 6.0c/95.