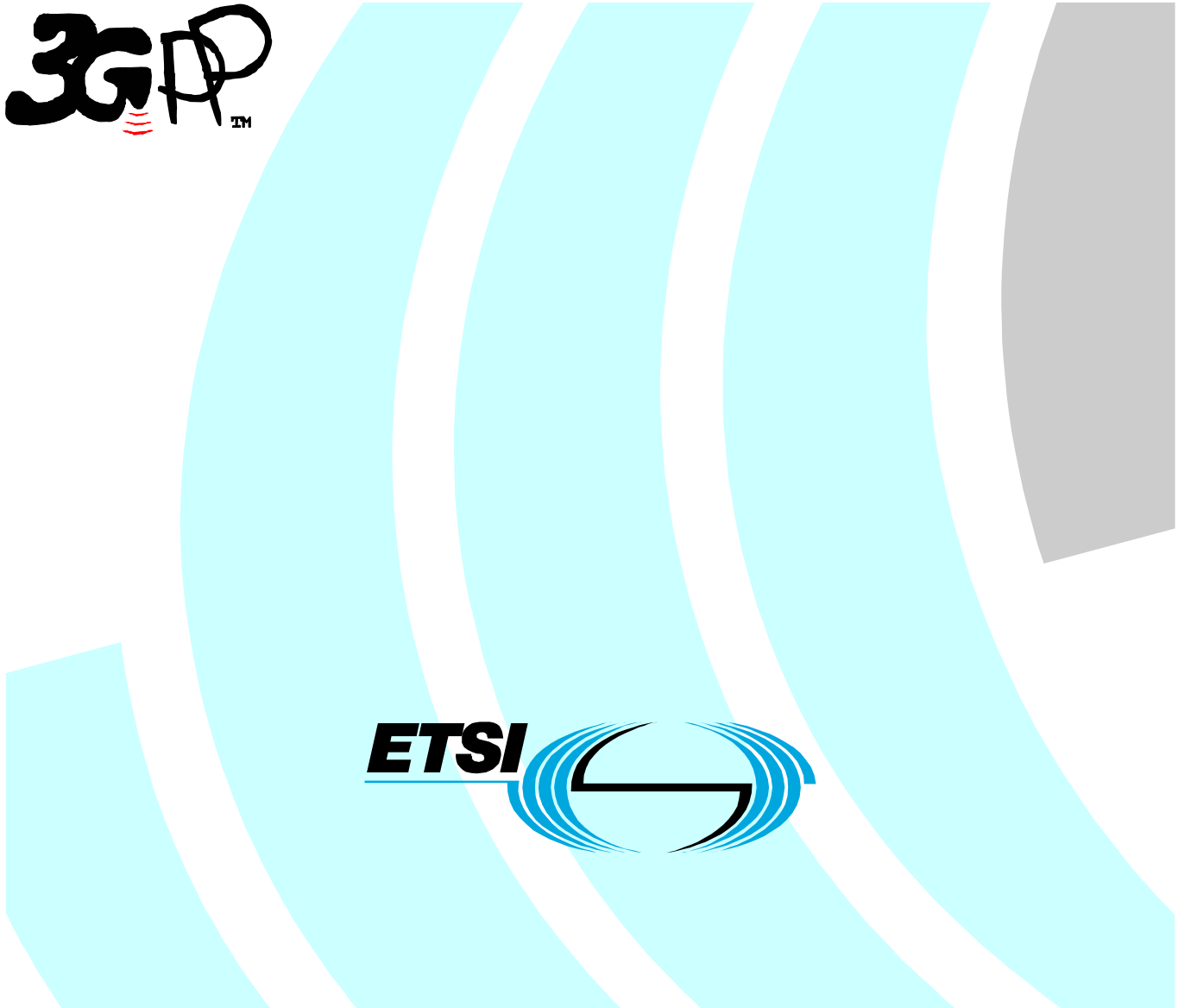
ETSI TS 125 322 V4.9.0 (2003-06)

Technical Specification

Universal Mobile Telecommunications System (UMTS);
Radio Link Control (RLC) protocol specification
(3GPP TS 25.322 version 4.9.0 Release 4)



Reference
RTS/TSGR-0225322v490

Keywords
UMTS

ETSI

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

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

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

Intellectual Property Rights

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

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

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp .

Contents

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document specifies the Radio Link Control protocol for the UE-UTRAN radio interface.

Features for the current Release:

- Transparent mode.
- Unacknowledged mode.
- Acknowledged mode.

2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TS 25.401: "UTRAN Overall Description".

[2] 3GPP TR 25.990: "Vocabulary for UTRAN".

[3] 3GPP TS 25.301: "Radio Interface Protocol Architecture".

[4] 3GPP TS 25.302: "Services provided by the Physical Layer".

[5] 3GPP TS 25.303: "Interlayer procedures in Connected Mode".

[6] 3GPP TS 25.304: "UE Procedures in Idle Mode and Procedures for Cell Reselection in Connected Mode".

[7] 3GPP TS 25.321: "Medium Access Control (MAC); protocol specification".

[8] 3GPP TS 25.331: "Radio Resource Control (RRC); protocol specification".

[9] 3GPP TS 33.102: "3G security; Security architecture".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in [2] apply.

3.2 Abbreviations

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

AM	Acknowledged Mode
AMD	Acknowledged Mode Data

ARQ	Automatic Repeat Request
BCCH	Broadcast Control CHannel
BCH	Broadcast CHannel
C-	Control-
CCCH	Common Control CHannel
CCH	Control CHannel
CCTrCH	Coded Composite Transport CHannel
CRC	Cyclic Redundancy Check
CTCH	Common Traffic CHannel
DCCH	Dedicated Control CHannel
DCH	Dedicated CHannel
DL	DownLink
DSCH	Downlink Shared CHannel
DTCH	Dedicated Traffic CHannel
FACH	Forward link Access CHannel
FDD	Frequency Division Duplex
EPC	Estimated PDU Counter
L1	Layer 1 (physical layer)
L2	Layer 2 (data link layer)
L3	Layer 3 (network layer)
LI	Length Indicator
LSB	Least Significant Bit
MAC	Medium Access Control
MRW	Move Receiving Window
MSB	Most Significant Bit
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
RRC	Radio Resource Control
SAP	Service Access Point
SDU	Service Data Unit
SHCCH	SHared channel Control CHannel
SN	Sequence Number
SUFI	SUper FIeld
TCH	Traffic CHannel
TDD	Time Division Duplex
TFI	Transport Format Indicator
TM	Transparent Mode
TMD	Transparent Mode Data
TTI	Transmission Time Interval
U-	User-
UE	User Equipment
UL	UpLink
UM	Unacknowledged Mode
UMD	Unacknowledged Mode Data
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network

4 General

4.1 Objective

This subclause describes the architecture of the RLC sublayer.

4.2 Overview of the RLC sublayer architecture

The model presented in this subclause is intended to support the definition of the RLC sublayer only, and is not meant to specify or constrain the implementation of the protocol. The RLC sublayer consists of RLC entities, of which there are three types: Transparent Mode (TM), Unacknowledged Mode (UM), and Acknowledged Mode (AM) RLC entities.

4.2.1 Model of the RLC sublayer

Figure 4.1 illustrates different RLC entities in the RLC model.

An UM and a TM RLC entity can be configured to be a transmitting RLC entity or a receiving RLC entity. The transmitting RLC entity transmits RLC PDUs and the receiving RLC entity receives RLC PDUs. An AM RLC entity consists of a transmitting side, and a receiving side, where the transmitting side of the AM RLC entity transmits RLC PDUs and the receiving side of the AM RLC entity receives RLC PDUs.

Elementary procedures (see clause 11) are defined between a "Sender" and a "Receiver". In UM and TM, the transmitting RLC entity acts as a Sender and the peer RLC entity acts as a Receiver. An AM RLC entity acts either as a Sender or as a Receiver depending on the elementary procedure. The Sender is the transmitter of AMD PDUs and the Receiver is the receiver of AMD PDUs. A Sender or a Receiver can reside at either the UE or the UTRAN.

There is one transmitting and one receiving RLC entity for each transparent mode (TM) and unacknowledged mode (UM) service. There is one combined, transmitting and receiving entity for the acknowledged mode (AM) service.

In the present document, "transmitted" is equivalent to "submitted to the lower layer" unless otherwise explicitly stated. Each RLC UM, and TM entity uses one logical channel to send or receive data PDUs. An AM RLC entity can be configured to use one or two logical channels to send or receive data and control PDUs. If two logical channels are configured, they are of the same type (DCCH or DTCH). In figure 4.1, the dashed lines between the AM-Entities illustrate the possibility to send and receive RLC PDUs on separate logical channels, e.g. control PDUs on one and data PDUs on the other. A more detailed description of the different entities is given in subclauses 4.2.1.1, 4.2.1.2 and 4.2.1.3.

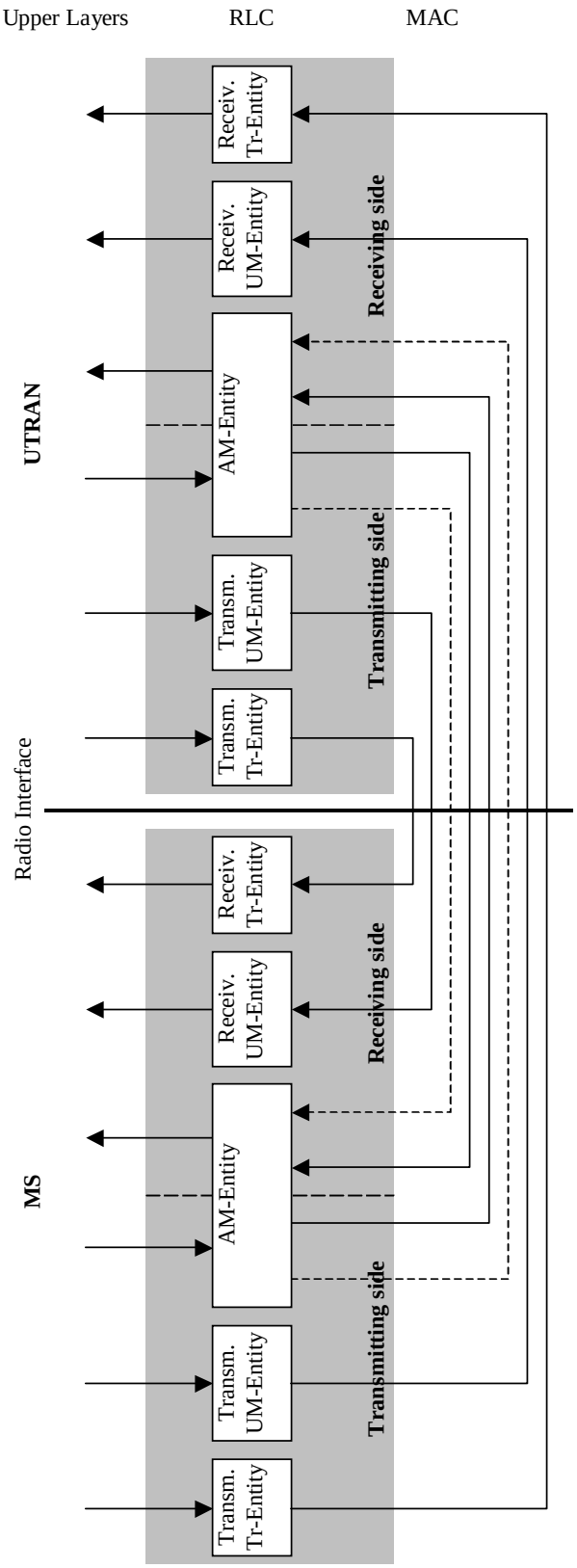


Figure 4.1: Overview model of the RLC sublayer

4.2.1.1 Transparent mode (TM) RLC entities

Figure 4.2 below shows the model of two transparent mode peer RLC entities. The logical channels used to communicate with the lower layer are described in the figure below.

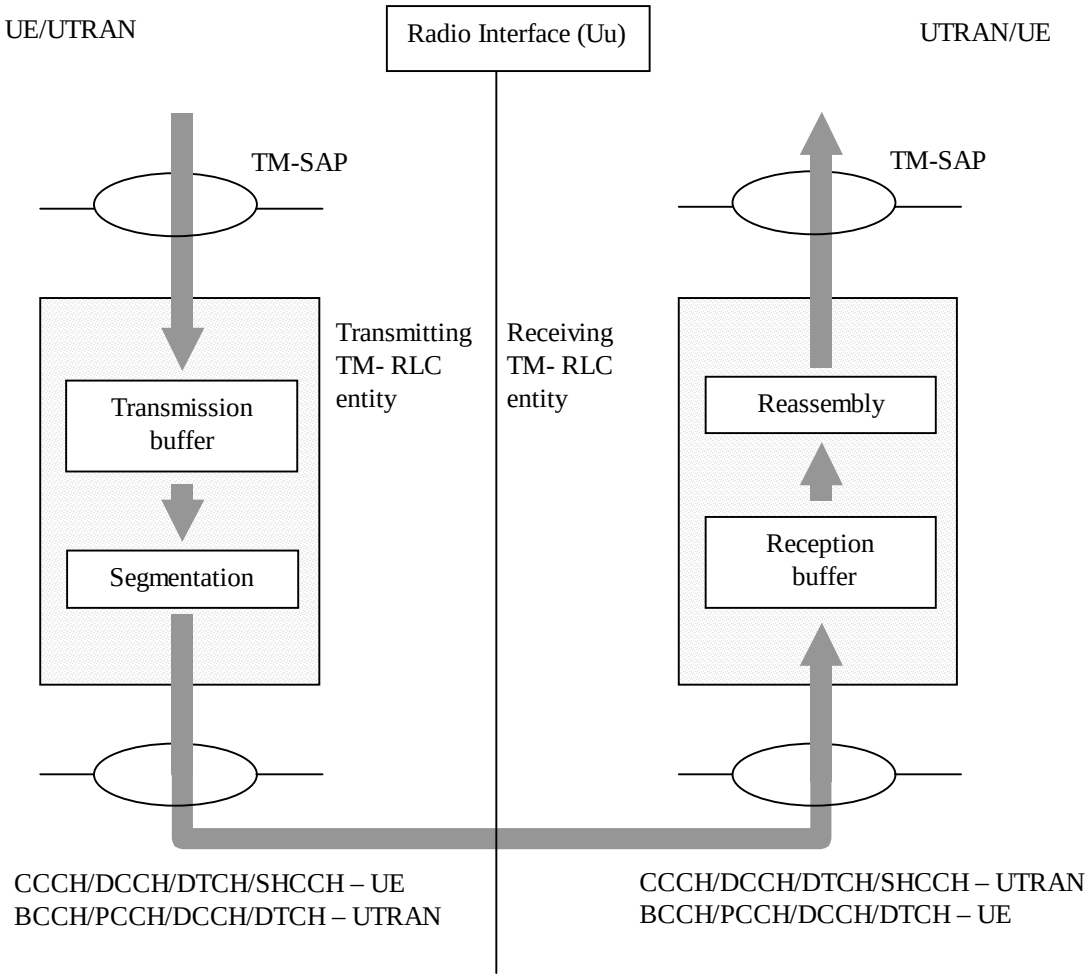


Figure 4.2: Model of two transparent mode peer entities

4.2.1.1.1 Transmitting TM RLC entity

The transmitting TM-RLC entity receives RLC SDUs from upper layers through the TM-SAP.

All received RLC SDUs must be of a length that is a multiple of one of the valid TMD PDU lengths.

If segmentation has been configured by upper layers and a RLC SDU is larger than the TMD PDU size used by the lower layer for that TTI, the transmitting TM RLC entity segments RLC SDUs to fit the TMD PDUs size without adding RLC headers. All the TMD PDUs carrying one RLC SDU are sent in the same TTI, and no segment from another RLC SDU are sent in this TTI.

If segmentation has not been configured by upper layers, then more than one RLC SDU can be sent in one TTI by placing one RLC SDU in one TMD PDU. All TMD PDUs in one TTI must be of equal length.

When the processing of a RLC SDU is complete, the resulting one or more TMD PDU(s) are/is submitted to the lower layer through either a BCCH, DCCH, PCCH, CCCH, SHCCH or a DTCH logical channel.

4.2.1.1.2 Receiving TM RLC entity

The receiving TM-RLC entity receives TMD PDUs through the configured logical channels from the lower layer. If segmentation is configured by upper layers, all TMD PDUs received within one TTI are reassembled to form the RLC SDU.

If segmentation is not configured by upper layers, each TMD PDU is treated as a RLC SDU.

The receiving TM RLC entity delivers RLC SDUs to upper layers through the TM-SAP.

4.2.1.2 Unacknowledged mode (UM) RLC entities

Figure 4.3 below shows the model of two unacknowledged mode peer RLC entities.

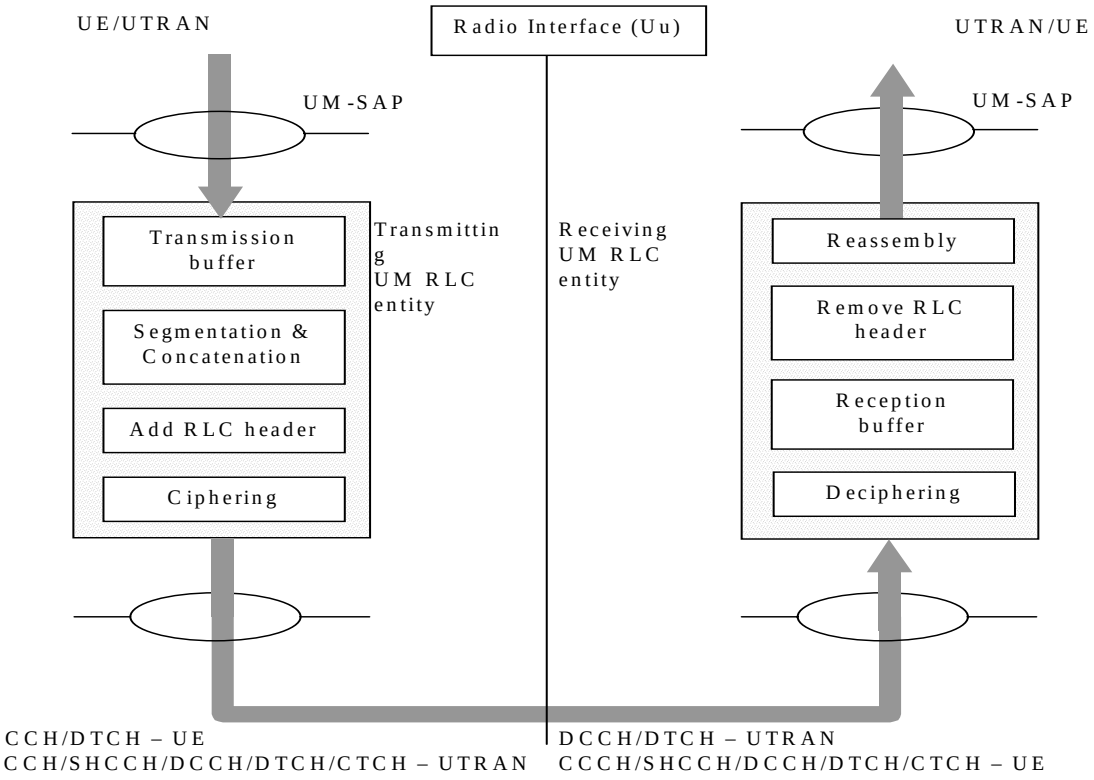


Figure 4.3: Model of two unacknowledged mode peer entities

4.2.1.2.1 Transmitting UM RLC entity

The transmitting UM-RLC entity receives RLC SDUs from upper layers through the UM-SAP.

The transmitting UM RLC entity segments the RLC SDU into UMD PDUs of appropriate size, if the RLC SDU is larger than the length of available space in the UMD PDU. The UMD PDU may contain segmented and/or concatenated RLC SDUs. UMD PDU may also contain padding to ensure that it is of a valid length. Length Indicators are used to define boundaries between RLC SDUs within UMD PDUs. Length Indicators are also used to define whether Padding is included in the UMD PDU.

If ciphering is configured and started, an UMD PDU is ciphered (except for the UMD PDU header) before it is submitted to the lower layer.

The transmitting UM RLC entity submits UMD PDUs to the lower layer through either a CCCH, SHCCH, DCCH, CTCH or a DTCH logical channel.

4.2.1.2.2 Receiving UM RLC entity

The receiving UM-RLC entity receives UMD PDUs through the configured logical channels from the lower layer.

The receiving UM RLC entity deciphers (if ciphering is configured and started) the received UMD PDUs (except for the UMD PDU header). It removes RLC headers from received UMD PDUs, and reassembles RLC SDUs (if segmentation and/or concatenation has been performed by the transmitting UM RLC entity).

RLC SDUs are delivered by the receiving UM RLC entity to the upper layers through the UM-SAP.

4.2.1.3 Acknowledged mode (AM) RLC entity

Figure 4.4 below shows the model of an acknowledged mode RLC entity.

The AM RLC entity can be configured to utilise one or two logical channels. The figure 4.4 shows the model of the AM RLC entity when one logical channel (shown as a solid line) and when two logical channels (shown as dashed lines) are used.

If one logical channel is configured, the transmitting side of the AM RLC entity submits AMD and Control PDUs to the lower layer on that logical channel. And the RLC PDU size shall be the same for AMD PDUs and control PDUs.

In case two logical channels are configured in the uplink, AMD PDUs are transmitted on the first logical channel, and control PDUs are transmitted on the second logical channel. In case two logical channels are configured in the downlink, AMD and Control PDUs can be transmitted on any of the two logical channels.

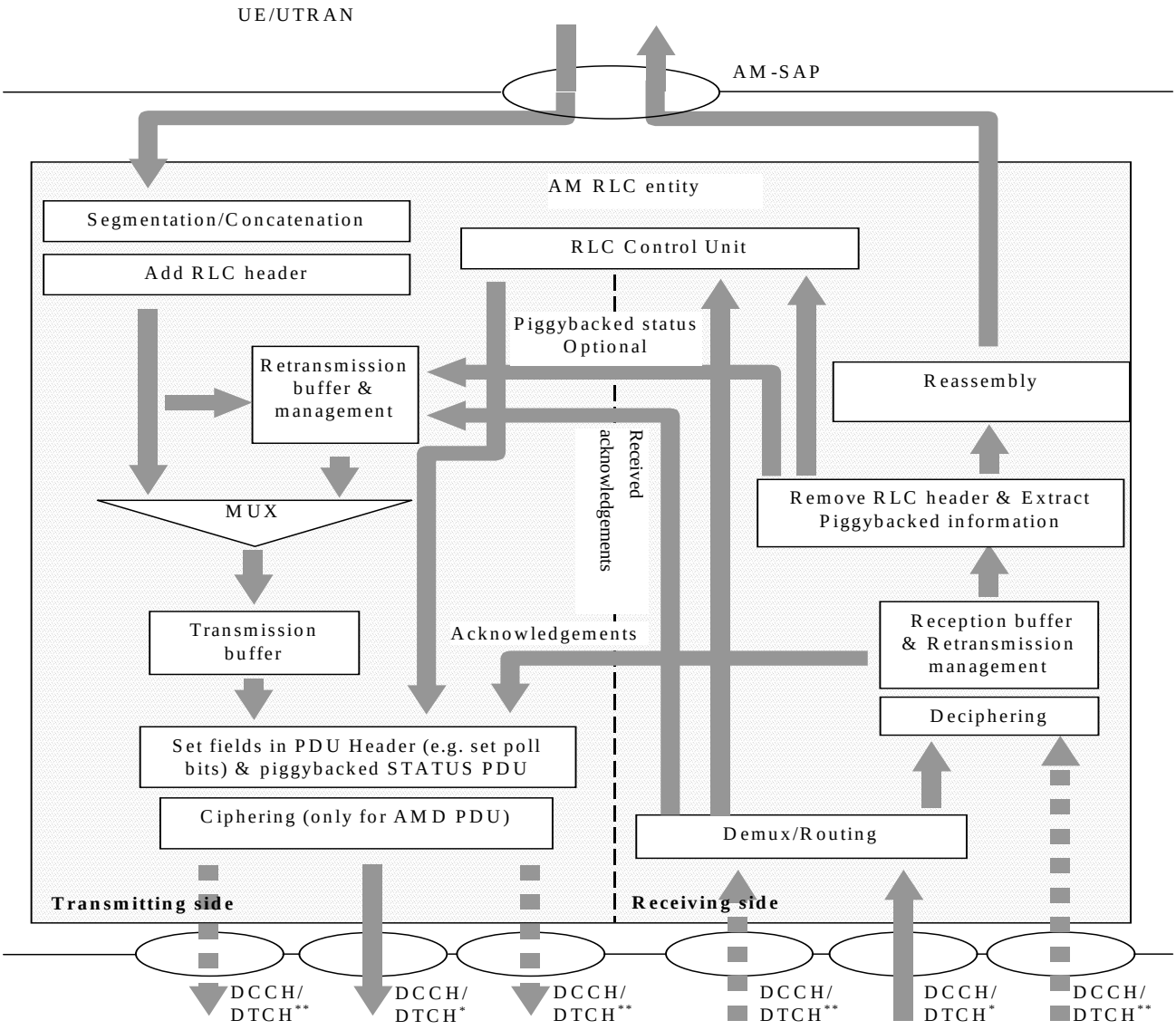


Figure 4.4: Model of an acknowledged mode entity

4.2.1.3.1 Transmitting side

The transmitting side of the AM-RLC entity receives RLC SDUs from upper layers through the AM-SAP.

RLC SDUs are segmented and/or concatenated into AMD PDUs of a fixed length. The segmentation is performed if the received RLC SDU is larger than the length of available space in the AMD PDU. The AMD PDU size is a semi-static value that is configured by upper layers and can only be changed through re-establishment of the AM RLC entity by upper layers. The AMD PDU may contain segmented and/or concatenated RLC SDUs. The AMD PDU may also contain Padding to ensure that it is of a valid size. Length Indicators are used to define boundaries between RLC SDUs within AMD PDUs. Length Indicators are also used to define whether Padding or Piggybacked STATUS PDU is included in the AMD PDU.

After the segmentation and/or concatenation are performed, the AMD PDUs are placed in the Retransmission buffer and at the MUX.

AMD PDUs buffered in the Retransmission buffer are deleted or retransmitted based on the status report found within a STATUS PDU or Piggybacked STATUS PDU sent by the peer AM RLC entity. This status report may contain positive or negative acknowledgements of individual AMD PDUs received by the peer AM RLC entity.

The MUX multiplexes AMD PDUs from the Retransmission buffer that need to be retransmitted, and the newly generated AMD PDUs delivered from the Segmentation/Concatenation function.

The PDUs are delivered to the function that completes the AMD PDU header and potentially replaces padding with piggybacked status information. A Piggybacked STATUS PDUs can be of variable size in order to match the amount of free space in the AMD PDU. The AMD PDU header is completed based on the input from the RLC Control Unit that indicates the values to set in various fields (e.g. Polling Bit). The function also multiplexes, if required, Control PDUs received from the RLC Control Unit (RESET and RESET ACK PDUs), and from the Reception buffer (Piggybacked STATUS and STATUS PDUs), with AMD PDUs.

The ciphering (if configured) is then applied to the AMD PDUs. The AMD PDU header is not ciphered. Piggybacked STATUS PDU and Padding in AMD PDU (when present) are ciphered. Control PDUs (i.e. STATUS PDU, RESET PDU, and RESET ACK PDU) are not ciphered.

The transmitting side of the AM RLC entity submits AMD PDUs to the lower layer through either one or two DCCH or DTCH logical channels.

4.2.1.3.2 Receiving side

The receiving side of the AM-RLC entity receives AMD and Control PDUs through the configured logical channels from the lower layer.

AMD PDUs are routed to the Deciphering Unit, where AMD PDUs (minus the AMD PDU header) are deciphered (if ciphering is configured and started), and then delivered to the Reception buffer.

The AMD PDUs are placed in the Reception buffer until a complete RLC SDU has been received. The Receiver acknowledges successful reception or requests retransmission of the missing AMD PDUs by sending one or more STATUS PDUs to the AM RLC peer entity, through its transmitting side. If a Piggybacked STATUS PDU is found in an AMD PDU, it is delivered to the Retransmission buffer & Management Unit at the transmitting side of the AM RLC entity, in order to purge the buffer of positively acknowledged AMD PDUs, and to indicate which AMD PDUs need to be retransmitted.

Once a complete RLC SDU has been received, the associated AMD PDUs are reassembled by the Reassembly Unit and delivered to upper layers through the AM-SAP.

RESET and RESET ACK PDUs are delivered to the RLC Control Unit for processing. If a response to the peer AM RLC entity is needed, an appropriate Control PDU is delivered, by the RLC Control Unit to the transmitting side of the AM RLC entity. The received STATUS PDUs are delivered to the Retransmission buffer and Management Unit at the transmitting side of the AM RLC entity, in order to purge the buffer of positively acknowledged AMD PDUs, and to indicate which AMD PDUs need to be retransmitted.

5 Functions

The following functions are supported by RLC sublayer. For an overall description of the following functions see [3]:

- Segmentation and reassembly.
- Concatenation.
- Padding.
- Transfer of user data.
- Error correction.
- In-sequence delivery of upper layer PDUs.
- Duplicate detection.
- Flow control.
- Sequence number check.
- Protocol error detection and recovery.
- Ciphering.

- SDU discard.

6 Services provided to upper layers

This clause describes the different services provided by RLC sublayer to upper layers. It also includes the mapping of RLC functions to different RLC services. For a detailed description of the RLC services see [3].

- **Transparent data transfer Service:**

  The following functions are needed to support transparent data transfer:

  - Segmentation and reassembly.
  - Transfer of user data.
  - SDU discard.

- **Unacknowledged data transfer Service:**

  The following functions are needed to support unacknowledged data transfer:

  - Segmentation and reassembly.
  - Concatenation.
  - Padding.
  - Transfer of user data.
  - Ciphering.
  - Sequence number check.
  - SDU discard.

- **Acknowledged data transfer Service:**

  The following functions are needed to support acknowledged data transfer:

  - Segmentation and reassembly.
  - Concatenation.
  - Padding.
  - Transfer of user data.
  - Error correction.
  - In-sequence delivery of upper layer PDUs.
  - Duplicate detection.
  - Flow Control.
  - Protocol error detection and recovery.
  - Ciphering.
  - SDU discard.

- **Maintenance of QoS as defined by upper layers.**
- **Notification of unrecoverable errors.**

6.1 Mapping of services/functions onto logical channels

The following tables show the applicability of services and functions to the logical channels in UL/DL and UE/UTRAN. A '+' in a column denotes that the service/function is applicable for the logical channel in question whereas a '-' denotes that the service/function is not applicable.

Table 6.1: RLC modes and functions in UE uplink side

Service	Functions	CCCH	SHCCH	DCCH	DTCH
Transparent Service	Applicability	+	+	+	+
	Segmentation	-	-	+	+
	Transfer of user data	+	+	+	+
	SDU Discard	-	-	+	+
Unacknowledged Service	Applicability	-	-	+	+
	Segmentation	-	-	+	+
	Concatenation	-	-	+	+
	Padding	-	-	+	+
	Transfer of user data	-	-	+	+
	Ciphering	-	-	+	+
	SDU Discard	-	-	+	+
Acknowledged Service	Applicability	-	-	+	+
	Segmentation	-	-	+	+
	Concatenation	-	-	+	+
	Padding	-	-	+	+
	Transfer of user data	-	-	+	+
	Flow Control	-	-	+	+
	Error Correction	-	-	+	+
	Protocol error correction & recovery	-	-	+	+
	Ciphering	-	-	+	+
	SDU Discard	-	-	+	+

Table 6.2: RLC modes and functions in UE downlink side

Service	Functions	BCCH	PCCH	SHCCH	CCCH	DCCH	DTCH	CTCH
Transparent Service	Applicability	+	+	-	-	+	+	-
	Reassembly	-	-	-	-	+	+	-
	Transfer of user data	+	+	-	-	+	+	-
Unacknowledged Service	Applicability	-	-	+	+	+	+	+
	Reassembly	-	-	+	+	+	+	+
	Deciphering	-	-	-	-	+	+	-
	Sequence number check	-	-	+	+	+	+	+
	Transfer of user data	-	-	+	+	+	+	+
Acknowledged Service	Applicability	-	-	-	-	+	+	-
	Reassembly	-	-	-	-	+	+	-
	Error correction	-	-	-	-	+	+	-
	Flow Control	-	-	-	-	+	+	-
	In sequence delivery	-	-	-	-	+	+	-
	Duplicate detection	-	-	-	-	+	+	-
	Protocol error correction & recovery	-	-	-	-	+	+	-
	Deciphering	-	-	-	-	+	+	-
	Transfer of user data	-	-	-	-	+	+	-
	SDU DIscard	-	-	-	-	+	+	-

Table 6.3: RLC modes and functions in UTRAN downlink side

Service	Functions	BCCH	PCCH	CCCH	SHCCH	DCCH	DTCH	CTCH
Transparent Service	Applicability	+	+	-	-	+	+	-
	Segmentation	-	-	-	-	+	+	-
	Transfer of user data	+	+	-	-	+	+	-
	SDU Discard	-	-	-	-	+	+	-
Unacknowledged Service	Applicability	-	-	+	+	+	+	+
	Segmentation	-	-	+	+	+	+	+
	Concatenation	-	-	+	+	+	+	+
	Padding	-	-	+	+	+	+	+
	Ciphering	-	-	-	-	+	+	-
	Transfer of user data	-	-	+	+	+	+	+
	SDU Discard	-	-	-	-	+	+	-
Acknowledged Service	Applicability	-	-	-	-	+	+	-
	Segmentation	-	-	-	-	+	+	-
	Concatenation	-	-	-	-	+	+	-
	Padding	-	-	-	-	+	+	-
	Transfer of user data	-	-	-	-	+	+	-
	Flow Control	-	-	-	-	+	+	-
	Error Correction	-	-	-	-	+	+	-
	Protocol error correction & recovery	-	-	-	-	+	+	-
	Ciphering	-	-	-	-	+	+	-
	SDU Discard	-	-	-	-	+	+	-

Table 6.4: RLC modes and functions in UTRAN uplink side

Service	Functions	CCCH	SHCCH	DCCH	DTCH
Transparent Service	Applicability	+	+	+	+
	Reassembly	-	-	+	+
	Transfer of user data	+	+	+	+
Unacknowledged Service	Applicability	-	-	+	+
	Reassembly	-	-	+	+
	Deciphering	-	-	+	+
	Sequence number check	-	-	+	+
	Transfer of user data	-	-	+	+
Acknowledged Service	Applicability	-	-	+	+
	Reassembly	-	-	+	+
	Error correction	-	-	+	+
	Flow Control	-	-	+	+
	In sequence delivery	-	-	+	+
	Duplicate detection	-	-	+	+
	Protocol error correction & recovery	-	-	+	+
	Deciphering	-	-	+	+
	Transfer of user data	-	-	+	+
	SDU Discard	-	-	+	+

7 Services expected from MAC

For a detailed description of the service provided by the MAC sublayer to upper layers see [3].

- Data transfer.

8 Elements for layer-to-layer communication

The interaction between the RLC sublayer and other layers are described in terms of primitives where the primitives represent the logical exchange of information and control between the RLC sublayer and other layers. The primitives shall not specify or constrain the implementation.

8.1 Primitives between RLC and upper layers

The primitives between RLC and upper layers are shown in table 8.1.

Table 8.1: Primitives between RLC and upper layers

Generic Name	Parameters			
	Req.	**Ind.**	**Resp.**	**Conf.**
RLC-AM-DATA	Data, CNF, DiscardReq, MUI, UE-ID type indicator	Data, DiscardInfo	Not Defined	Status, MUI
RLC-UM-DATA	Data, UE-ID type indicator, DiscardReq, MUI	Data	Not Defined	MUI
RLC-TM-DATA	Data, UE-ID type indicator, DiscardReq, MUI	Data, Error_Indicator	Not Defined	MUI
CRLC-CONFIG	E/R, Stop (UM/AM only), Continue (UM/AM only), Ciphering Elements (UM/AM only), TM_parameters (TM only), UM_parameters (UM only), AM_parameters (AM only)	Not Defined	Not Defined	Not Defined
CRLC-SUSPEND (UM/AM only)	N	Not Defined	Not Defined	VT(US) (UM only), VT(S) (AM only)
CRLC-RESUME (UM/AM only)	No Parameter	Not Defined	Not Defined	Not Defined
CRLC-STATUS	Not Defined	EVC	Not Defined	Not Defined

Each Primitive is defined as follows:

RLC-AM-DATA-Req/Ind/Conf

- RLC-AM-DATA-Req is used by upper layers to request transmission of an RLC SDU in acknowledged mode.
- RLC-AM-DATA-Ind is used by the AM RLC entity to deliver to upper layers an RLC SDU that has been transmitted in acknowledged mode and to indicate to upper layers of the discarded RLC SDU in the peer RLC AM entity.
- RLC-AM-DATA-Conf is used by the AM RLC entity to confirm to upper layers the reception of an RLC SDU by the peer-RLC AM entity or to inform the upper layers of a discarded SDU.

RLC-UM-DATA-Req/Ind/Conf

- RLC-UM-DATA-Req is used by upper layers to request transmission of an RLC SDU in unacknowledged mode.
- RLC-UM-DATA-Ind is used by the UM RLC entity to deliver to upper layers an RLC SDU that has been transmitted in unacknowledged mode.
- RLC-UM-DATA-Conf is used by the UM RLC entity to inform the upper layers of a discarded SDU.

RLC-TM-DATA-Req/Ind/Conf

- RLC-TM-DATA-Req is used by upper layers to request transmission of an RLC SDU in transparent mode.
- RLC-TM-DATA-Ind is used by the TM RLC entity to deliver to upper layers an RLC SDU that has been transmitted in transparent mode.
- RLC-TM-DATA-Conf is used by the TM RLC entity to inform the upper layers of a discarded SDU.

CRLC-CONFIG-Req

This primitive is used by upper layers to establish, re-establish, release, stop, continue or modify the RLC. Ciphering elements are included for UM and AM operation.

CRLC-SUSPEND-Req/Conf

- CRLC-SUSPEND-Req is used by upper layers to suspend the UM or AM RLC entity.
- CRLC-SUSPEND-Conf is used by the UM or AM RLC entity to confirm that the entity is suspended.

CRLC-RESUME-Req

This primitive is used by upper layers to resume the UM or AM RLC entity after the UM or AM RLC entity has been suspended.

CRLC-STATUS-Ind

It is used by an RLC entity to send status information to upper layers.

8.2 Primitive parameters

Following parameters are used in the primitives:

1) The parameter Data is the RLC SDU that is mapped onto the Data field in RLC PDUs. When AM or UM RLC entities are used, the length of the Data parameter is a multiple of 8 bits, otherwise (TM RLC entity) the length of Data parameter is a bit-string whose length may not be a multiple of 8 bits.

2) The parameter Confirmation Request (CNF) indicates whether the transmitting side of the AM RLC entity needs to confirm the reception of the RLC SDU by the peer-RLC AM entity. If required, once all AMD PDUs that make up the RLC SDU are positively acknowledged by the receiving AM RLC entity, the transmitting AM RLC entity notifies upper layers.

3) The parameter Message Unit Identifier (MUI) is an identity of the RLC SDU, which is used to indicate which RLC SDU that is confirmed with the RLC-AM-DATA-Conf. primitive, or discarded with the RLC-AM/UM/TM-DATA-Conf. Primitive.

4) The parameter E/R indicates establishment, re-establishment, release or modification of an RLC entity, where re-establishment is applicable to AM and UM RLC entities only. If re-establishment is requested, the state variables and configurable parameters are initialised according to subclause 9.7.7. If release is requested, all protocol parameters, variables and timers are released and the RLC entity enters the NULL state. If modification is requested, the protocol parameters indicated by upper layers (e.g. ciphering parameters) are only modified, while keeping the other protocol parameters, such as the protocol variables, protocol timers and protocol state unchanged. AM RLC entities are always re-established if the AMD PDU size is changed. The modification of other protocol parameters does not require a re-establishment.

5) The parameter Event Code (EVC) indicates the reason for the CRLC-STATUS-Ind e.g., unrecoverable errors such as data link layer loss or recoverable status events such as reset.

6) The parameter Ciphering Elements are only applicable for UM and AM operations. These parameters are Ciphering Mode, Ciphering Key, Transmitting Activation Time (Sequence Number to activate a new ciphering configuration at the Sender), Receiving Activation Time (Sequence Number to activate a new ciphering configuration at the Receiver) and HFN (Hyper Frame Number).

7) The AM_parameters are only applicable for AM operation. These parameters are AMD PDU size, In-sequence Delivery Indication (indicating that RLC SDUs are delivered to upper layers in sequence or that they can be delivered out of sequence), Timer values (see subclause 9.5), Protocol parameter values (see subclause 9.6), Polling triggers (see subclause 9.7.1), Status triggers (see subclause 9.7.2), Periodical Status blocking configuration (see subclause 9.7.2), SDU discard mode (see subclause 9.7.3), Minimum WSN (see subclause 9.2.2.11.3), and Send MRW. The Minimum WSN is always greater than or equal to the number of transport blocks in the smallest transport block set. The Send MRW indicates that the information of each discarded RLC SDU is sent to the Receiver, and the MRW SUFI is sent to the Receiver even if no segments of the RLC SDU to be discarded were submitted to a lower layer.

8) The parameter DiscardInfo indicates to upper layer the discarded RLC SDU in the peer-RLC AM entity. It is applicable only when in-sequence delivery is configured and it is to be used when upper layers require the reliable data transfer.

9) The Stop parameter is applicable to AM and UM RLC entities only and indicates to the RLC entity to (see subclause 9.7.6):

   - not transmit nor receive any RLC PDUs.

10) The Continue parameter is applicable to AM and UM RLC entities only and indicates to the RLC entity to continue transmission and reception of RLC PDUs.

11) The UM_parameters are only applicable for UM operation. It contains Timer_Discard value (see subclause 9.5) and largest UMD PDU size (see subclause 9.2.2.8).

12) The TM_parameters are only applicable for TM operation. It contains e.g. segmentation indication (see subclauses 9.2.2.9 and 11.1.2.1), Timer_Discard value (see subclause 9.5) and delivery of erroneous SDU indication (see subclause 11.1.3).

13) The N parameter indicates that an RLC entity will not send a PDU with "Sequence Number">=VT(S)+N for AM and "Sequence Number">=VT(US)+N for UM, where N is a non-negative integer.

14) The VT(S) parameter indicates the value of the Send State Variable for the case of the AM.

15) The VT(US) parameter indicates the value of the UM Data State Variable, for the case of the UM.

16) The Error_Indicator parameter indicates that the RLC SDU is erroneous (see subclause 11.1.3).

17) The parameter UE-ID type indicator indicates the RNTI type (U-RNTI or C-RNTI) to be used for the associated RLC SDU. This parameter is not required at the UE.

18) The parameter DiscardReq indicates whether the transmitting RLC entity needs to inform the upper layers of the discarded RLC SDU. If required, the transmitting RLC entity notifies upper layers when the SDU is discarded.

19) The parameter Status is only applicable for AM operation. This parameter indicates whether a RLC SDU is successfully transmitted or discarded.

9 Elements for peer-to-peer communication

9.1 Protocol data units

The structures defined in this subclause are normative.

9.1.1 Data PDUs

a) TMD PDU (Transparent Mode Data PDU).

The TMD PDU is used to convey RLC SDU data without adding any RLC overhead. The TMD PDU is used by RLC when it is in transparent mode.

b) UMD PDU (Unacknowledged Mode Data PDU).

The UMD PDU is used to convey sequentially numbered PDUs containing RLC SDU data. UMD PDUs are used by RLC when it is configured for unacknowledged data transfer.

c) AMD PDU (Acknowledged Mode Data PDU).

The AMD PDU is used to convey sequentially numbered PDUs containing RLC SDU data. AMD PDUs are used by RLC when it is configured for acknowledged data transfer.

9.1.2 Control PDUs

Control PDUs are only used in acknowledged mode.

a) STATUS PDU and Piggybacked STATUS PDU.

The STATUS PDU and the Piggybacked STATUS PDU are used:

- by the Receiver to inform the Sender about missing and received AMD PDUs in the Receiver;
- by the Receiver to inform the Sender about the size of the allowed transmission window;
- by the Sender to request the Receiver to move the reception window; and
- by the Receiver to acknowledge the Sender about the reception of the request to move the reception window.

b) RESET PDU.

The RESET PDU is used to reset all protocol states, protocol variables and protocol timers of the peer RLC entity in order to synchronise the two peer entities. It is sent by the Sender to the Receiver.

c) RESET ACK PDU.

The RESET ACK PDU is an acknowledgement to the RESET PDU. It is sent by the Receiver to the Sender.

Table 9.1: RLC PDU names and descriptions

Data Transfer Mode	PDU name	Description
Transparent	TMD	Transparent mode data
Unacknowledged	UMD	Sequenced unacknowledged mode data
Acknowledged	AMD	Sequenced acknowledged mode data
	STATUS	Solicited or Unsolicited Status Report, Change window size command, SDU discard command, or SDU discard acknowledgement
	Piggybacked STATUS	Piggybacked Solicited or Unsolicited Status Report, Change window size command, SDU discard command, or SDU discard acknowledgement
	RESET	Reset Command
	RESET ACK	Reset Acknowledgement

9.2 Formats and parameters

The formats of RLC PDUs and their parameters defined in this subclause are normative.

9.2.1 Formats

This subclause specifies the format of the RLC PDUs. The parameters of each RLC PDU are explained in subclause 9.2.2.

9.2.1.1 General

An RLC PDU is a bit string. In the figures in subclause 9.2, bit strings are represented by tables in which the first bit is the leftmost one on the first line of the table, the last bit is the rightmost one on the last line of the table, and more generally the bit string is to be read from left to right and then in the reading order of the lines.

Depending on the provided service, RLC SDUs are bit strings, with any non-null length, or bit strings with a multiple of 8 bits in length. An RLC SDU is included into an RLC PDU from first bit onward.

9.2.1.2 TMD PDU

The TMD PDU is used to transfer user data when RLC is operating in transparent mode. No overhead is added to the SDU by RLC. The data length is not constrained to be a multiple of 8 bits.

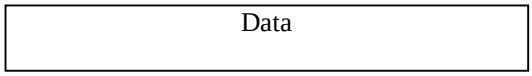


Figure 9.1: TMD PDU

9.2.1.3 UMD PDU

The UMD PDU is used to transfer user data when RLC is operating in unacknowledged mode. The length of the data part shall be a multiple of 8 bits. The UMD PDU header consists of the first octet, which contains the "Sequence Number". The RLC header consists of the first octet and all the octets that contain "Length Indicators".

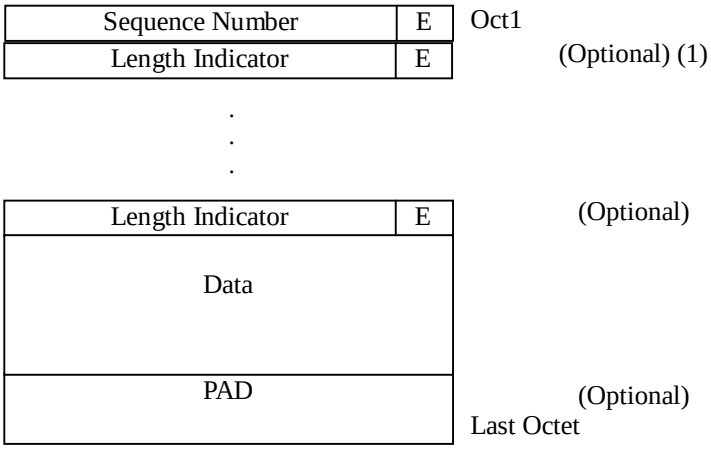
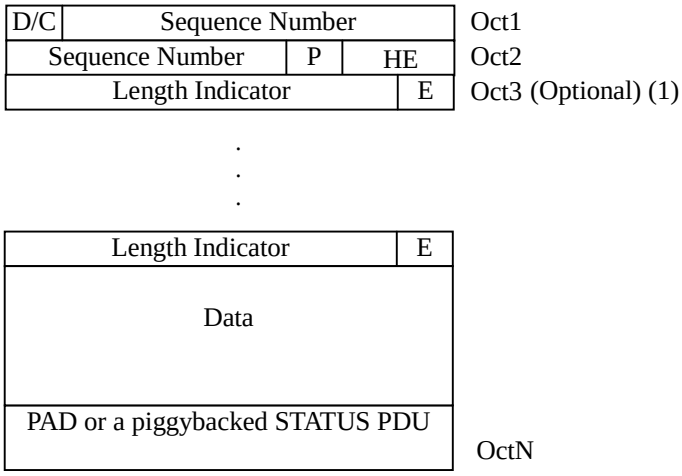


Figure 9.2: UMD PDU

NOTE (1): The "Length Indicator" may be 15 bits.

9.2.1.4 AMD PDU

The AMD PDU is used to transfer user data, piggybacked status information and the Polling bit when RLC is operating in acknowledged mode. The length of the data part shall be a multiple of 8 bits. The AMD PDU header consists of the first two octets, which contain the "Sequence Number". The RLC header consists of the first two octets and all the octets that contain "Length Indicators".



NOTE (1): The "Length Indicator" may be 15 bits.

Figure 9.3: AMD PDU

9.2.1.5 STATUS PDU

The STATUS PDU is used to exchange status information between two RLC AM entities.

The format of the STATUS PDU is given in figure 9.4 below. The length of each super field (SUFI) is dependent on its type and contents.

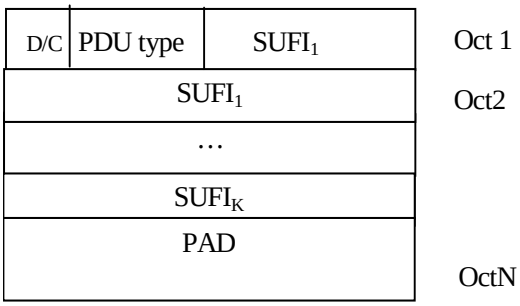


Figure 9.4: STATUS PDU

A STATUS PDU can include super-fields of different types. The size of a STATUS PDU is variable and upper bounded by the maximum RLC PDU size used by the logical channel on which the control PDUs are sent. Padding shall be included to match one of the PDU sizes used by the logical channel on which the control PDUs are sent. The length of the STATUS PDU shall be a multiple of 8 bits.

9.2.1.6 Piggybacked STATUS PDU

The format of the piggybacked STATUS PDU is the same as for the STATUS PDU except that the D/C field is replaced by a reserved bit (R2). This PDU can be piggybacked in an AMD PDU if the data leaves out enough room in the AMD PDU. The PDU Type field is set to "000" and all other values are invalid for this version of the protocol.

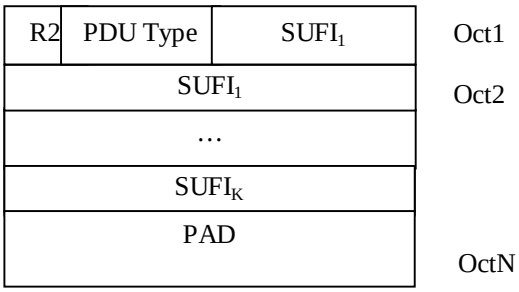


Figure 9.5: Piggybacked STATUS PDU

9.2.1.7 RESET, RESET ACK PDU

The RESET PDU includes a one-bit sequence number field (RSN). The value of this bit is carried over in the RESET ACK PDU sent in response in order to allow the peer entity to identify which RESET PDU it was sent in response to.

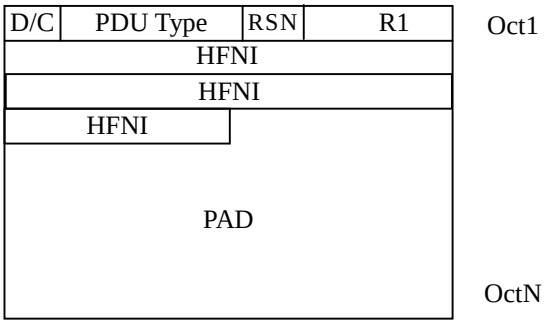


Figure 9.6: RESET, RESET ACK PDU

The size of a RESET or RESET ACK PDU is variable and upper bounded by the maximum RLC PDU size used by the logical channel on which the control PDUs are sent. Padding shall be included to match one of the PDU sizes used by the logical channel on which the control PDUs are sent. The length of the RESET or RESET ACK PDU shall be a multiple of 8 bits.

9.2.2 Parameters

If not otherwise mentioned in the definition of each field, the bits in the parameters shall be interpreted as follows: the left-most bit string is the first and most significant and the right most bit is the last and least significant bit.

Unless otherwise mentioned, integers are encoded in standard binary encoding for unsigned integers. In all cases, including when a value extends over more than one octet as shown in the tables, the bits appear ordered from MSB to LSB when read in the RLC PDU.

9.2.2.1 D/C field

Length: 1bit.

The D/C field indicates the type of an AM PDU. It can be either data or control PDU.

Bit	Description
0	Control PDU
1	Data PDU

9.2.2.2 PDU Type

Length: 3 bit.

The PDU type field indicates the Control PDU type.

Bit	PDU Type
000	STATUS
001	RESET
010	RESET ACK
011-111	Reserved (PDUs with this coding will be discarded by this version of the protocol).

9.2.2.3 Sequence Number (SN)

This field indicates the "Sequence Number" of the RLC PDU, encoded in binary.

PDU type	Length	Notes
AMD PDU	12 bits	Used for retransmission and reassembly
UMD PDU	7 bits	Used for reassembly

9.2.2.4 Polling bit (P)

Length: 1bit.

This field is used to request a status report (one or several STATUS PDUs) from the Receiver.

Bit	Description
0	Status report not requested
1	Request a status report

9.2.2.5 Extension bit (E)

Length: 1bit.

This bit indicates if the next octet will be a "Length Indicator" and E bit.

Bit	Description
0	The next field is data, piggybacked STATUS PDU or padding
1	The next field is Length Indicator and E bit

9.2.2.6 Reserved 1 (R1)

Length: 3 bits.

This field in the RESET PDU and RESET ACK PDU is used to have a multiple of 8 bits in length. Its shall always be coded to "000". Other values are reserved and will be considered invalid for this version of the protocol.

9.2.2.7 Header Extension Type (HE)

Length: 2 bits.

This two-bit field indicates if the next octet is data or a "Length Indicator" and E bit.

Value	Description
00	The succeeding octet contains data
01	The succeeding octet contains a length indicator and E bit
10-11	Reserved (PDUs with this coding will be discarded by this version of the protocol).

9.2.2.8 Length Indicator (LI)

A "Length Indicator" is used to indicate the last octet of each RLC SDU ending within the PDU.

Except for the predefined values reserved for special purposes and listed in the tables below, the "Length Indicator" shall:

- be set to the number of octets between the end of the RLC header and up to and including the last octet of an RLC SDU segment;
- be included in the PDUs that they refer to.

The size of the "Length Indicator" may be either 7 bits or 15 bits. The value of a "Length Indicator" shall not exceed the values specified in subclauses 11.2.4.2 and 11.3.4.5 respectively for UMD and AMD PDUs.

The "Length Indicators" which refer to the same PDU shall:

- not be reordered in case of retransmission;
- be in the same order as the RLC SDUs that they refer to.

For AM:

- if the "AMD PDU size" is ≤ 126 octets:
  - 7-bit "Length Indicators" shall be used.
- else:
  - 15-bit "Length Indicators" shall be used.
- the size of the "Length Indicator" is always the same for all AMD PDUs, for one RLC entity.

For UM:

- if the "largest UMD PDU size" is ≤ 125 octets:
  - 7-bit "Length Indicators" shall be used.
- else:
  - 15-bit "Length Indicators" shall be used.
- between modifications of the "largest UMD PDU size", the size of the "Length Indicator" is the same for all UMD PDUs;

- if the RLC SDU begins in the beginning of the RLC PDU; and
- if the RLC PDU is transmitted in uplink; and
- if the "Length Indicators" indicating that a RLC SDU ended exactly in the end or one octet short (only when 15-bit "Length Indicators" is used) of the previous RLC PDU are not present:
  - if 7-bit "Length Indicator" is used:
    - the "Length Indicator" with value "111 1100" shall be used;
  - if 15-bit "Length Indicator" is used:
    - the "Length Indicator" with value "111 1111 1111 1100" shall be used.
- in downlink:
  - if 7-bit "Length Indicator" is used:
    - the Receiver shall be prepared to receive the "Length Indicator" with value "111 1100";
    - the Receiver shall follow the discard rules in subclause 11.2.3 both when the "Length Indicator" with value "111 1100" is present and when it is absent.
  - if 15-bit "Length Indicator" is used:
    - the Receiver shall be prepared to receive the "Length Indicator" with value "111 1111 1111 1100";
    - the Receiver shall follow the discard rules in subclause 11.2.3 both when the "Length Indicator" with value "111 1111 1111 1100" is present and when it is absent.

In the case where the end of the last segment of an RLC SDU exactly ends at the end of a PDU and there is no "Length Indicator" that indicates the end of the RLC SDU:

- if 7-bit "Length Indicator" is used:
  - a "Length Indicator" with value "000 0000" shall be placed as the first "Length Indicator" in the following PDU;
- if 15-bit "Length Indicator" is used:
  - a "Length Indicator" with value "000 0000 0000 0000" shall be placed as the first "Length Indicator" in the following PDU.

In the case where a PDU contains a 15-bit "Length Indicator" indicating that an RLC SDU ends with one octet left in the PDU, the last octet of this PDU shall:

- be padded by the Sender and ignored by the Receiver though there is no "Length Indicator" indicating the existence of Padding; and
- not be filled with the first octet of the next RLC SDU data.

In the case where 15-bit "Length Indicators" are used in a PDU and the last segment of an RLC SDU is one octet short of exactly filling the PDU:

- if a 15-bit "Length Indicator" is used for the following PDU:
  - the "Length Indicator" with value "111 1111 1111 1011" shall be placed as the first "Length Indicator" in the following PDU;
  - the remaining one octet in the current PDU shall be padded by the Sender and ignored at the Receiver though there is no "Length Indicator" indicating the existence of Padding;
- if a 7-bit "Length Indicator" is used for the following PDU:
  - if RLC is configured for UM mode:

- the "Length Indicator" with value "000 0000" shall be placed as the first "Length indicator" in the following PDU and its "Sequence Number" shall be incremented by 2 before it is transmitted.

For UM and AM RLC:

- if a 7 bit "Length Indicator" is used in a RLC PDU and one or more padding octets are present in the RLC PDU after the end of the last RLC SDU:
  - indicate the presence of padding by including a "Length Indicator" with value "1111111" as the last "Length Indicator" in the PDU.
- if a 15 bit "Length Indicator" is used in a RLC PDU and two or more padding octets are present in the RLC PDU after the end of the last RLC SDU:
  - indicate the presence of padding by including a "Length Indicator" with value "111 1111 1111 1111" as the last "Length Indicator" in the PDU.

NOTE: After the "Length Indicator" indicating the presence of padding has been included in the RLC PDU, the length of the padding may be zero.

If a "Length Indicator" is still awaiting transmission and there is no RLC SDU available, an RLC PDU consisting of this "Length Indicator", the appropriate padding "Length Indicator" and padding may be transmitted.

Predefined values of the "Length Indicator" are used to indicate padding. The values that are reserved for special purposes are listed in the tables below depending on the size of the "Length Indicator". Only predefined "Length Indicator" values can refer to the padding space. These values shall only be placed after all other "Length Indicators" for a PDU.

STATUS PDUs can be piggybacked on the AMD PDU by using part or all of the padding space. A predefined "Length Indicator" shall be used to indicate the presence of a piggybacked STATUS PDU. This "Length Indicator" replaces the padding "Length Indicator". The piggybacked STATUS PDU shall be appended immediately following the PDU data. When only part of the padding space is used, the end of the piggybacked STATUS PDU is indicated by one of the SUFI fields NO_MORE or ACK. Thus no additional "Length Indicator" is required to show that there is still padding in the AMD PDU.

If "SDU discard with explicit signalling" is configured:

- an AMD PDU can contain a maximum number of 15 "Length Indicators" indicating the end of 15 corresponding SDUs; and
- the rest of the AMD PDU space shall be used as padding or as piggybacked STATUS PDU.

Length: 7 bits

Bit	Description
0000000	The previous RLC PDU was exactly filled with the last segment of an RLC SDU and there is no "Length Indicator" that indicates the end of the RLC SDU in the previous RLC PDU.
1111100	UMD PDU: The first data octet in this RLC PDU is the first octet of an RLC SDU. AMD PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol).
1111101	Reserved (PDUs with this coding will be discarded by this version of the protocol).
1111110	AMD PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU. UMD PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol).
1111111	The rest of the RLC PDU is padding. The padding length can be zero.

Length: 15bits

Bit	Description
000000000000000	The previous RLC PDU was exactly filled with the last segment of an RLC SDU and there is no "Length Indicator" that indicates the end of the RLC SDU in the previous RLC PDU.
111111111111011	The last segment of an RLC SDU was one octet short of exactly filling the previous RLC PDU and there is no "Length Indicator" that indicates the end of the RLC SDU in the previous RLC PDU. The remaining one octet in the previous RLC PDU is ignored.
111111111111100	UMD PDU: The first data octet in this RLC PDU is the first octet of an RLC SDU. AMD PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol).
111111111111101	Reserved (PDUs with this coding will be discarded by this version of the protocol).
111111111111110	AMD PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU. UMD PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol).
111111111111111	The rest of the RLC PDU is padding. The padding length can be zero.

9.2.2.9 Data field

RLC SDUs or segments of RLC SDUs are mapped to this field in transparent, unacknowledged and acknowledged modes.

Transparent mode data:

- the length of RLC SDUs is not constrained to a multiple of 8 bits;
- if "Segmentation" is configured:
  - all the RLC PDUs carrying segments of a RLC SDU shall be sent in one TTI;
  - only RLC PDUs carrying segments from a single RLC SDU shall be sent in one TTI;
- otherwise (Segmentation is not configured):
  - TMD PDU size is fixed within a single TTI and is equal to the RLC SDU size.

Unacknowledged mode data and Acknowledged mode data:

- the length of RLC SDUs is constrained to a multiple of 8 bits;
  - the last segment of an RLC SDU shall be concatenated with the first segment of the next RLC SDU in order to fill the data field completely and avoid unnecessary padding. The "Length Indicator" field is used to point the borders between RLC SDUs (see subclause 9.2.2.8).

9.2.2.10 Padding (PAD)

All unused space in a PDU shall be located at the end of the PDU and is referred to as padding. Padding shall have a length such that the PDU as a whole has one of the predefined total lengths.

Padding may have any value and the Receiver and the Sender shall disregard it.

9.2.2.11 SUFI

Which SUFI fields to use is implementation dependent, but when a STATUS PDU includes information about which AMD PDUs have been received and which are detected as missing, information shall not be included about AMD PDUs with "Sequence Number"≥VR(H) i.e. AMD PDUs that have not yet reached the Receiver. Information about AMD PDUs with "Sequence Number"<VR(R) shall not be given except when this is necessary in order to use the BITMAP SUFI, see subclause 9.2.2.11.5.

Length: variable number of bits.

The SUFI can include three sub-fields: type information (type of super-field, e.g. list, bitmap, acknowledgement, etc), length information (providing the length of a variable length field within the following value field) and a value.

Figure 9.7 shows the structure of the super-field. The size of the type sub-field is non-zero but the size of the other sub-fields may be zero.

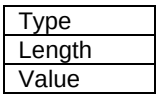


Figure 9.7: The Structure of a Super-Field

The length of the type field is 4 bits and it may have any of following values.

Bit	Description
0000	No More Data (**NO_MORE**)
0001	Window Size (**WINDOW**)
0010	Acknowledgement (**ACK**)
0011	List (**LIST**)
0100	Bitmap (**BITMAP**)
0101	Relative list (**Rlist)**
0110	Move Receiving Window **(MRW)**
0111	Move Receiving Window Acknowledgement **(MRW_ACK)**
1000-1111	Reserved (PDUs with this encoding are invalid for this version of the protocol)

The size and presence of the sub-fields "Length" and "Value" depend on the super-field type and is specified for each super field separately.

9.2.2.11.1 The No More Data super-field

The 'No More Data' super-field indicates the end of the data part of a STATUS PDU and is shown in Figure 9.8 below. It shall always be placed as the last SUFI if it is included in a STATUS PDU. All data after this SUFI shall be regarded as padding and shall be neglected.

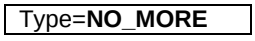


Figure 9.8: NO_MORE field in a STATUS PDU

9.2.2.11.2 The Acknowledgement super-field

The 'Acknowledgement' super-field consists of a type identifier field (ACK) and a sequence number (LSN) as shown in figure 9.9 below. The acknowledgement super-field is also indicating the end of the data part of a STATUS PDU. Thus, no 'NO_MORE' super-field is needed in the STATUS PDU when the 'ACK' super-field is present. The ACK SUFI shall always be placed as the last SUFI if it is included in a STATUS PDU. All data after this SUFI shall be regarded as padding and shall be neglected.

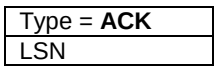


Figure 9.9: The ACK fields in a STATUS PDU

LSN

Length: 12 bits

Acknowledges the reception of all AMD PDUs with "Sequence Number" < LSN (Last Sequence Number) that are *not* indicated to be erroneous in earlier parts of the STATUS PDU. This means that if the LSN is set to a value greater than VR(R), all erroneous AMD PDUs shall be included in the same STATUS PDU and if the LSN is set to VR(R), the erroneous AMD PDUs can be split into several STATUS PDUs. At the transmitter, if the value of the LSN =< the value of the first error indicated in the STATUS PDU, VT(A) will be updated according to the LSN, otherwise VT(A) will be

updated according to the first error indicated in the STATUS PDU. VT(A) is only updated based on STATUS PDUs where ACK SUFI (or MRW_ACK SUFI) is included. The LSN shall not be set to a value > VR(H) nor < VR(R).

9.2.2.11.3 The Window Size super-field

The Window Size super-field consists of a type identifier (WINDOW) and a window size number (WSN) as shown in Figure 9.10 below. The Receiver is always allowed to change the transmission window size of the peer entity during a connection, but the minimum and the maximum allowed value is given by upper layers configuration. The reception window size of the Receiver is not changed.

Type = **WINDOW**
WSN

Figure 9.10: The WINDOW fields in a STATUS PDU

WSN

Length: 12 bits

The value of VT(WS) to be used by the transmitter. The range of the WSN is [0, 2^{12}-1]. The minimum value of VT(WS) is 1. If WSN is zero the SUFI shall be discarded by this version of the protocol. The variable VT(WS) is set equal to WSN upon reception of this SUFI. If WSN is greater than Configured_Tx_Window_Size, VT(WS) shall be set equal to Configured_Tx_Window_Size.

9.2.2.11.4 The List super-field

The List Super-Field consists of a type identifier field (LIST), a list length field (LENGTH) and a list of LENGTH number of pairs as shown in figure 9.11 below:

Type = **LIST**
LENGTH
SN_1
L_1
SN_2
L_2
…
SN_{LENGTH}
L_{LENGTH}

Figure 9.11: The List fields in a STATUS PDU

LENGTH

Length: 4 bits

The number of (SN_i , L_i)-pairs in the super-field of type LIST. The value "0000" is invalid and the STATUS PDU is discarded.

SN_i

Length: 12 bits

"Sequence Number" of AMD PDU, which was not correctly received.

L_i

Length: 4 bits

Number of consecutive AMD PDUs not correctly received following AMD PDU with "Sequence Number" SN_i.

9.2.2.11.5 The Bitmap super-field

The Bitmap Super-Field consists of a type identifier field (BITMAP), a bitmap length field (LENGTH), a first sequence number (FSN) and a bitmap as shown in figure 9.12 below:

Type = **BITMAP**
LENGTH
FSN
Bitmap

Figure 9.12: The Bitmap fields in a STATUS PDU

LENGTH

Length: 4 bits

The size of the bitmap in octets equals LENGTH+1, i.e. LENGTH="0000" means that the size of the bitmap is one octet and LENGTH="1111" gives the maximum bitmap size of 16 octets.

FSN

Length: 12 bits

The "Sequence Number" for the first bit in the bitmap. FSN shall not be set to a value lower than VR(R)-7 when the reception window size is less than half the maximum RLC AM "Sequence Number". If the reception window size is larger, FSN shall not be set to a value lower than VR(R).

Bitmap

Length: Variable number of octets given by the LENGTH field.

Status of the "Sequence Number" fields in the interval [FSN, FSN + (LENGTH+1)*8 - 1] indicated in the bitmap where each position (from left to right) can have two different values (0 and 1) with the following meaning (bit_position$\in$ [0,(LENGTH+1)*8 - 1]):

1: Sequence Number = (FSN + bit_position) has been correctly received.

0: Sequence Number = (FSN + bit_position) has not been correctly received.

9.2.2.11.6 The Relative List super-field

The Relative List super-field consists of a type identifier field (RLIST), a list length field (LENGTH), the first sequence number (FSN) and a list of LENGTH number of codewords (CW) as shown in figure 9.13 below.

Type = **RLIST**
LENGTH
FSN
CW_1
CW_2
…
CW_{LENGTH}

Figure 9.13: The RList fields in a STATUS PDU

LENGTH

Length: 4 bits

The number of codewords (CW) in the super-field of type RLIST.

FSN

Length: 12 bits

The "Sequence Number" for the first erroneous AMD PDU in the RLIST, i.e. LENGTH="0000" means that only FSN is present in the SUFI.

CW

Length: 4 bits

The CW consists of 4 bits where the three first bits are part of a number and the last bit is a status indicator and it shall be interpreted as follows:

Code Word	Description
$X_1X_2X_3$ 0	Next 3 bits of the number are $X_1X_2X_3$ and the number continues in the next CW. The most significant bit within this CW is X_1.
$X_1X_2X_3$ 1	Next 3 bits of the number are $X_1X_2X_3$ and the number is terminated. The most significant bit within this CW is X_1. This is the most significant CW within the number.

By default, the number given by the CWs represents a distance between the previous indicated erroneous AMD PDU up to and including the next erroneous AMD PDU.

One special value of CW is defined:

000 1 'Error burst indicator'.

The error burst indicator means that the next CWs will represent the number of subsequent erroneous AMD PDUs (not counting the already indicated error position). After the number of errors in a burst is terminated with XXX 1, the next codeword will again by default be the least significant bits (LSB) of the distance to the next error.

If the last CW, as indicated by the value of the LENGTH field, does not contain a "1" in its rightmost position, or the last CW, as indicated by the value of the LENGTH field does contain a "1" in its rightmost position, but is a special "error burst indicator" CW, the encoding of the RLIST SUFI is invalid, and the STATUS PDU is discarded.

9.2.2.11.7 The Move Receiving Window Acknowledgement super-field

The 'Move Receiving Window Acknowledgement' super-field acknowledges the reception of a MRW SUFI. The format is given in figure 9.14 below.

Type = **MRW_ACK**
N
SN_ACK

Figure 9.14: The MRW-ACK fields in a STATUS PDU

N

Length: 4 bits

The N field shall be set equal to the N_{LENGTH} field in the received MRW SUFI if the SN_ACK field is equal to the SN_MRW_{LENGTH} field. Otherwise N shall be set to 0.

With the aid of this field in combination with the SN_ACK field, it can be determined if the MRW_ACK corresponds to a previously transmitted MRW SUFI.

SN_ACK

Length: 12 bits

The SN_ACK field indicates the updated value of VR(R) after the reception of the MRW SUFI. With the aid of this field in combination with the N field, it can be determined if the MRW_ACK corresponds to a previously transmitted MRW SUFI.

9.2.2.11.8 The Move Receiving Window (MRW) super-field

The 'Move Receiving Window' super-field is used to request the Receiver to move its reception window and optionally to indicate the set of discarded RLC SDUs, as a result of an RLC SDU discard in the Sender. The format is given in figure 9.15 below.

Type = **MRW**
LENGTH
SN_MRW_1
SN_MRW_2
...
SN_MRW_{LENGTH}
N_{LENGTH}

Figure 9.15: The MRW fields in a STATUS PDU

LENGTH

Length: 4 bits

The number of SN_MRW_i fields in the super-field of type MRW.

The values "0001" through "1111" indicate 1 through 15 SN_MRW_i respectively. The value "0000" indicates that one SN_MRW_i field is present and that the RLC SDU to be discarded in the Receiver extends above the configured transmission window in the Sender.

SN_MRW_i

Length: 12 bits

When "Send MRW" is configured, an SN_MRW_i shall be used to indicate the end of each discarded RLC SDU, i.e. the number of SN_MRW_i fields shall equal the number of RLC SDUs discarded by that MRW SUFI. When "Send MRW" is not configured, an SN_MRW_i field shall be used to indicate the end of the last RLC SDU to be discarded in the Receiver and additional ones may optionally be used to indicate the end of other discarded RLC SDUs. SN_MRW_i is the "Sequence Number" of the AMD PDU that contains the "Length Indicator" of the i:th RLC SDU to be discarded in the Receiver (except for SN_MRW_{LENGTH} when $N_{LENGTH} = 0$, see definition of N_{LENGTH}). The order of the SN_MRW_i shall be in the same sequential order as the RLC SDUs that they refer to.

Additionally SN_MRW_{LENGTH} requests the Receiver to discard all AMD PDUs with "Sequence Number" < SN_MRW_{LENGTH}, and to move the reception window accordingly. In addition, when $N_{LENGTH} > 0$, the Receiver has to discard the first N_{LENGTH} "Length Indicators" and the corresponding data octets in the AMD PDU with "Sequence Number" SN_MRW_{LENGTH}.

N_{LENGTH}

Length: 4 bits

N_{LENGTH} is used together with SN_MRW_{LENGTH} to indicate the end of the last RLC SDU to be discarded in the Receiver.

N_{LENGTH} indicates which "Length Indicator" in the AMD PDU with "Sequence Number" SN_MRW_{LENGTH} corresponds to the last RLC SDU to be discarded in the Receiver. $N_{LENGTH} = 0$ indicates that the last RLC SDU ended in the AMD PDU with "Sequence Number" $SN\_MRW_{LENGTH} - 1$ and that the first data octet in the AMD PDU with "Sequence Number" SN_MRW_{LENGTH} is the first data octet to be reassembled next.

9.2.2.12 Reserved 2 (R2)

Length: 1 bit

This bit in the Piggybacked STATUS PDU is used to make the Piggybacked STATUS PDU a multiple of 8 bits in length and for this purpose it is coded as 0. Otherwise the PDU is treated as invalid and hence shall be discarded by this version of the protocol.

9.2.2.13 Reset Sequence Number (RSN)

Length: 1 bit

This field is used to indicate the sequence number of the transmitted RESET PDU. If this RESET PDU is a retransmission of the original RESET PDU then the retransmitted RESET PDU would have the same RSN value as the original RESET PDU. Otherwise it will have the next RSN value. The initial value of this field is zero. The value of this field shall be reinitialised when the RLC is re-established. It shall not be reinitialised when the RLC is reset.

9.2.2.14 Hyper Frame Number Indicator (HFNI)

Length: 20 bit

This field is used to indicate the hyper frame number (HFN) to the peer entity. With the aid of this field the HFN in UE and UTRAN can be synchronised.

9.3 Protocol states

The content presented in this subclause is intended to support the definition of the RLC protocol states only, and is not meant to specify or constrain the implementation of the protocol.

9.3.1 State model for transparent mode entities

Figure 9.16 illustrates the state model for transparent mode RLC entities (both transmitting and receiving). A transparent mode entity can be in one of the following states.

9.3.1.1 NULL State

In the NULL state the RLC entity does not exist and therefore it is not possible to transfer any data through it.

Upon reception of a CRLC-CONFIG-Req from upper layers indicating establishment, the RLC entity:

- is created; and
- enters the DATA_TRANSFER_READY state.

9.3.1.2 DATA_TRANSFER_READY State

In the DATA_TRANSFER_READY state, transparent mode data can be exchanged between the entities according to subclause 11.1.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

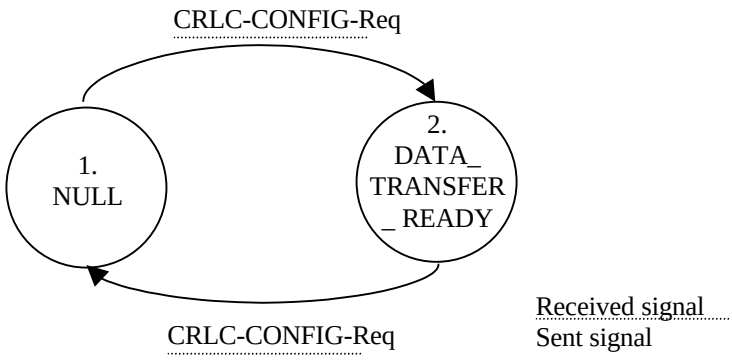


Figure 9.16: The state model for transparent mode entities

9.3.2 State model for unacknowledged mode entities

Figure 9.17 illustrates the state model for unacknowledged mode RLC entities (both transmitting and receiving). An unacknowledged mode entity can be in one of the following states.

9.3.2.1 NULL State

In the NULL state the RLC entity does not exist and therefore it is not possible to transfer any data through it.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating establishment the RLC entity:

- is created; and
- enters the DATA_TRANSFER_READY state.

9.3.2.2 DATA_TRANSFER_READY State

In the DATA_TRANSFER_READY state, unacknowledged mode data can be exchanged between the entities according to subclause 11.2.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating modification, the RLC entity:

- stays in the DATA_TRANSFER_READY state;
- modifies only the protocol parameters and timers as indicated by upper layers.

Upon reception of a CRLC-SUSPEND-Req from upper layers, the RLC entity:

- enters the LOCAL_SUSPEND state.

9.3.2.3 LOCAL_SUSPEND State

In the LOCAL_SUSPEND state, the RLC entity is suspended, i.e. it does not send UMD PDUs with "Sequence Number" greater than or equal to a certain specified value (see subclause 9.7.5).

Upon reception of a CRLC-RESUME-Req from upper layers, the RLC entity:

- enters the DATA_TRANSFER_READY state; and
- resumes the data transmission.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating modification, the RLC entity:

- stays in the LOCAL_SUSPEND state;
- modifies only the protocol parameters and timers as indicated by upper layers.

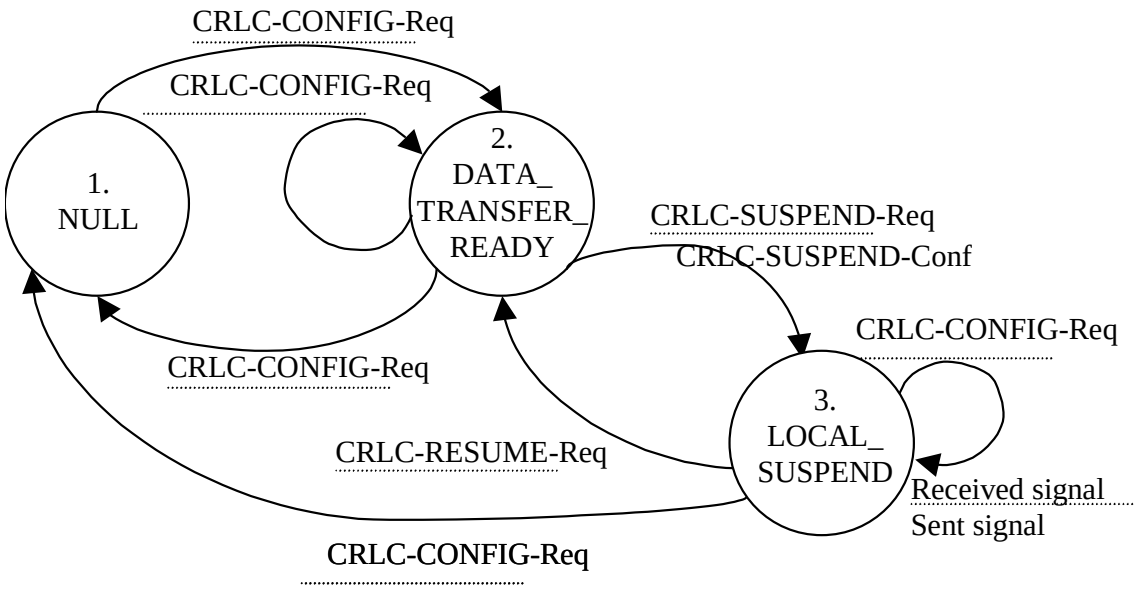


Figure 9.17: The state model for unacknowledged mode entities

9.3.3 State model for acknowledged mode entities

Figure 9.18 illustrates the state model for the acknowledged mode RLC entity (both transmitting and receiving). An acknowledged mode entity can be in one of the following states.

9.3.3.1 NULL State

In the NULL state the RLC entity does not exist and therefore it is not possible to transfer any data through it.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating establishment, the RLC entity:

- is created; and
- enters the DATA_TRANSFER_READY state.

9.3.3.2 DATA_TRANSFER_READY State

In the DATA_TRANSFER_READY state, acknowledged mode data can be exchanged between the entities according to subclause 11.3.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

Upon detection of an initiating condition for the RLC reset procedure described in subclause 11.4.2, the RLC entity:

- initiates the RLC reset procedure (see subclause 11.4); and
- enters the RESET_PENDING state.

Upon reception of a RESET PDU, the RLC entity responds according to subclause 11.4.3.

Upon reception of a RESET ACK PDU, the RLC entity takes no action.

Upon reception of CRLC-SUSPEND-Req from upper layer, the RLC entity is suspended and enters the LOCAL_SUSPEND state.

9.3.3.3 RESET_PENDING State

In the RESET_PENDING state the entity waits for a response from its peer entity and no data can be exchanged between the entities.

Upon reception of a CRLC-CONFIG-Req from upper layer indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

Upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU, the RLC entity:

- acts according to subclause 11.4.4; and
- enters the DATA_TRANSFER_READY state.

Upon reception of a RESET ACK PDU with a different RSN value as in the corresponding RESET PDU, the RLC entity:

- discards the RESET ACK PDU (see subclause 11.4.4); and
- stays in the RESET_PENDING state.

Upon reception of a RESET PDU, the RLC entity:

- responds according to subclause 11.4.3; and
- stays in the RESET_PENDING state.

Upon reception of CRLC-SUSPEND-Req from upper layer, the RLC entity:

- enters the RESET_AND_SUSPEND state.

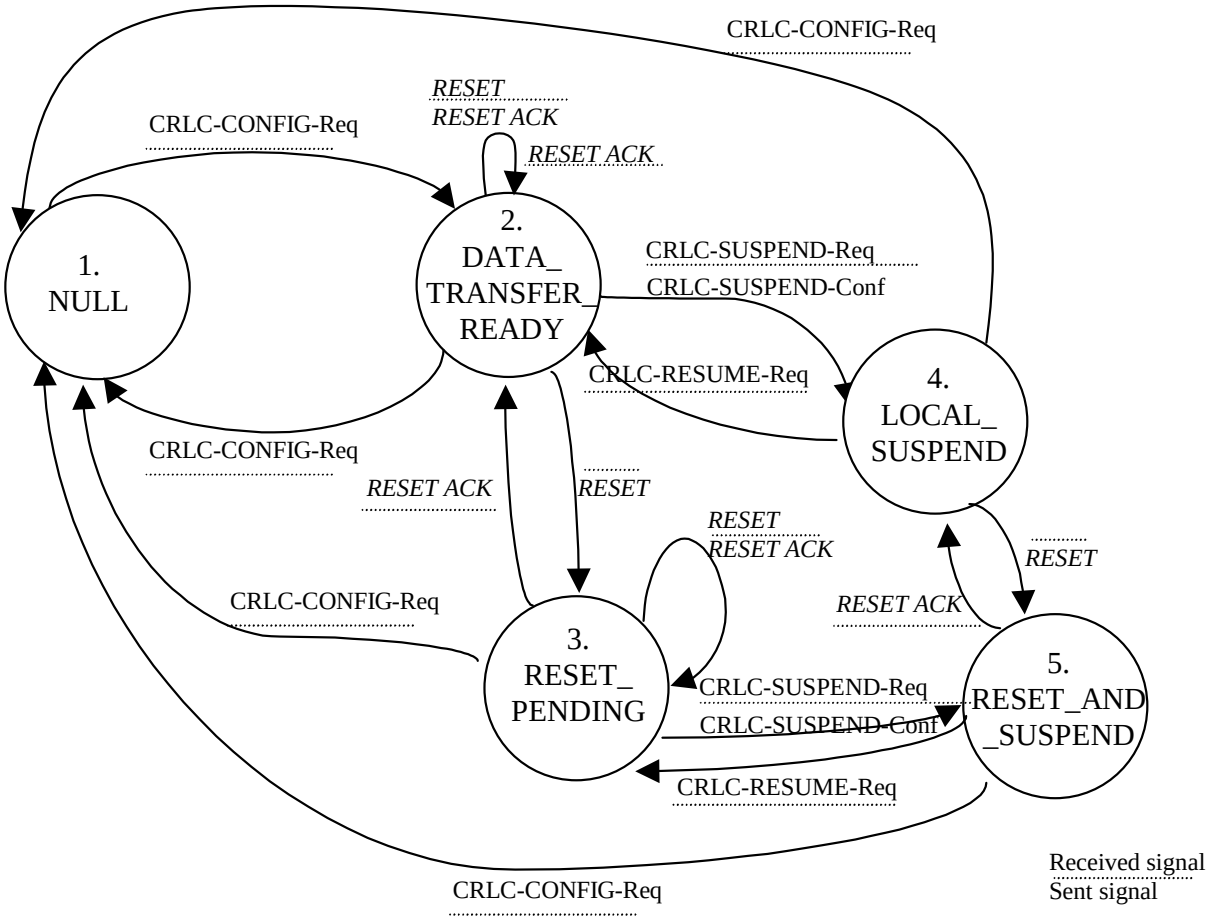


Figure 9.18: The state model for the acknowledged mode entities

9.3.3.4 LOCAL_SUSPEND State

In the LOCAL_SUSPEND state, the RLC entity is suspended, i.e. it does not send AMD PDUs with "Sequence Number" greater than or equal to certain specified value (see subclause 9.7.5).

Upon reception of CRLC-RESUME-Req from upper layers in this state, the RLC entity:

- resumes the data transmission; and
- enters the DATA_TRANSFER_READY state.

Upon reception of CRLC-CONFIG-Req from upper layers indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

Upon detection of an initiating condition for RLC reset procedure described in subclause 11.4.2, the RLC entity:

- initiates the RLC reset procedure (see subclause 11.4); and
- enters the RESET_AND_SUSPEND state.

9.3.3.5 RESET_AND_SUSPEND State

In the RESET_ AND_SUSPEND state, the entity waits for a response from its peer entity or a primitive (CRLC-RESUME-Req) from its upper layer and no data can be exchanged between the entities.

Upon reception of CRLC-CONFIG-Req from upper layer indicating release, the RLC entity:

- enters the NULL state; and
- is considered as being terminated.

Upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU, the RLC entity:

- acts according to subclause 11.4.4; and
- enters the LOCAL_SUSPEND state.

Upon reception of CRLC-RESUME-Req from upper layer in this state, the RLC entity:

- is resumed, i.e. releases the suspend constraint; and
- enters the RESET_PENDING state.

9.4 State variables

The state variables defined in this subclause are normative.

This sub-clause describes the state variables used in AM and UM in order to specify the peer-to-peer protocol. All state variables are non-negative integers. UMD and AMD PDUs are numbered by modulo integer sequence numbers (SN) cycling through the field: 0 to $2^{12} - 1$ for AM and 0 to $2^{7} - 1$ for UM. All arithmetic operations contained in the present document on VT(S), VT(A), VT(MS), VR(R), VR(H) and VR(MR) are affected by the AM modulus. All arithmetic operations contained in the present document on VT(US) and VR(US) are affected by the UM modulus. When performing arithmetic comparisons of state variables or Sequence number values a modulus base shall be used. This modulus base is subtracted (within the appropriate field) from all the values involved and then an absolute comparison is performed. At the Sender, VT(A) and VT(US) shall be assumed to be the modulus base in AM and UM respectively. At the Receiver, VR(R) and VR(US) shall be assumed to be the modulus base in AM and UM respectively.

The RLC shall maintain the following state variables in the Sender.

a) VT(S) - Send state variable.

   This state variable contains the "Sequence Number" of the next AMD PDU to be transmitted for the first time (i.e. excluding retransmitted PDUs). It shall be updated after the aforementioned AMD PDU is transmitted or after transmission of a MRW SUFI which includes SN_MRW_{LENGTH} >VT(S) (see subclause 11.6).

   The initial value of this variable is 0.

b) VT(A) - Acknowledge state variable.

   This state variable contains the "Sequence Number" following the "Sequence Number" of the last in-sequence acknowledged AMD PDU. This forms the lower edge of the transmission window of acceptable acknowledgements. VT(A) shall be updated based on the receipt of a STATUS PDU including an ACK (see subclause 9.2.2.11.2) and/or an MRW_ACK SUFI (see subclause 11.6).

   The initial value of this variable is 0. For the purpose of initialising the protocol, this value shall be assumed to be the first "Sequence Number" following the last in-sequence acknowledged AMD PDU.

c) VT(DAT).

   This state variable counts the number of times a AMD PDU has been scheduled to be transmitted. There shall be one VT(DAT) for each PDU and each shall be incremented every time the corresponding AMD PDU is scheduled to be transmitted.

   The initial value of this variable is 0.

d) VT(MS) - Maximum Send state variable.

This state variable contains the "Sequence Number" of the first AMD PDU that can be rejected by the peer Receiver, VT(MS) = VT(A) + VT(WS). This value represents the upper edge of the transmission window. The transmitter shall not transmit AMD PDUs with "Sequence Number" $\geq$ VT(MS) unless VT(S) $\geq$ VT(MS). In that case, the AMD PDU with "Sequence Number" = VT(S) - 1 can also be transmitted. VT(MS) shall be updated when VT(A) or VT(WS) is updated.

The initial value of this variable is Configured_Tx_Window_size.

e) VT(US) – UM data state variable.

This state variable contains the "Sequence Number" of the next UMD PDU to be transmitted. It shall be incremented by 1 each time a UMD PDU is transmitted.

The initial value of this variable is 0.

NOTE: For the UTRAN side, the initial value of this variable can be different from 0.

f) VT(PDU).

This state variable is used when the "poll every Poll_PDU PDU" polling trigger is configured. It shall be incremented by 1 for each AMD PDU that is transmitted including both new and retransmitted AMD PDUs. When it becomes equal to the value Poll_PDU, a new poll shall be transmitted and the state variable shall be set to zero.

The initial value of this variable is 0.

g) VT(SDU).

This state variable is used when the "poll every Poll_SDU SDU" polling trigger is configured. It shall be incremented by 1 for a given SDU when the AMD PDU carrying the first segment of this SDU is scheduled to be transmitted for the first time. When it becomes equal to the value Poll_SDU a new poll shall be transmitted and the state variable shall be set to zero. The "Polling bit" shall be set to "1" in the first transmission of the AMD PDU that contains the "Length Indicator" indicating the end of the SDU.

The initial value of this variable is 0.

h) VT(RST) - Reset state variable.

This state variable is used to count the number of times a RESET PDU is scheduled to be transmitted before the reset procedure is completed. VT(RST) shall be incremented by 1 each time a RESET PDU is scheduled to be transmitted. VT(RST) shall only be reset upon the reception of a RESET ACK PDU, i.e. VT(RST) shall not be reset when an RLC reset initiated by the peer RLC entity occurs.

The initial value of this variable is 0.

i) VT(MRW) – MRW command send state variable.

This state variable is used to count the number of times a MRW command is transmitted. VT(MRW) is incremented by 1 each time a timer Timer_MRW expires. VT(MRW) shall be reset when the SDU discard with explicit signalling procedure is terminated.

The initial value of this variable is 0.

j) VT(WS) – Transmission window size state variable.

This state variable contains the size that shall be used for the transmission window. VT(WS) shall be set equal to the WSN field when the transmitter receives a STATUS PDU including a WINDOW SUFI.

The initial value of this variable is Configured_Tx_Window_size.

The RLC shall maintain the following state variables in the Receiver:

a) VR(R) - Receive state variable.

This state variable contains the "Sequence Number" following that of the last in-sequence AMD PDU received. It shall be updated upon the receipt of the AMD PDU with "Sequence Number" equal to VR(R).

The initial value of this variable is 0. For the purpose of initialising the protocol, this value shall be assumed to be the first "Sequence Number" following the last in-sequence received AMD PDU.

b) VR(H) - Highest expected state variable.

This state variable contains the "Sequence Number" following the highest "Sequence Number" of any received AMD PDU. When a AMD PDU is received with "Sequence Number" x such that VR(H)≤x<VR(MR), this state variable shall be set equal to x+1.

The initial value of this variable is 0.

c) VR(MR) - Maximum acceptable Receive state variable.

This state variable contains the "Sequence Number" of the first AMD PDU that shall be rejected by the Receiver, VR(MR) = VR(R) + Configured_Rx_Window_Size.

d) VR(US) - Receiver Send Sequence state variable.

This state variable contains the "Sequence Number" following that of the last UMD PDU received. When a UMD PDU with "Sequence Number" equal to x is received, the state variable shall set equal to x + 1.

The initial value of this variable is 0.

e) VR(EP) - Estimated PDU Counter state variable.

This state variable contains the number of AMD PDUs whose re-transmission is still expected as a consequence of the transmission of the latest status report. At the end of each TTI it is decremented by the total number of AMD PDUs that were received during that time.

9.5 Timers

The timers defined in this subclause are normative. The timers shall be considered active from the time they are started until the time they either expire or are stopped.

a) Timer_Poll.

This timer shall only be used when so configured by upper layers. The value of the timer is signalled by upper layers. In the UE this timer shall be started when the successful or unsuccessful transmission of an AMD PDU containing a poll is indicated by lower layer. In UTRAN it should be started when an AMD PDU containing a poll is submitted to lower layer. If x is the value of the state variable VT(S) after the poll was submitted to lower layer, the timer shall be stopped upon receiving:

- positive acknowledgements for all the AMD PDUs with "Sequence Number" up to and including x - 1; or
- a negative acknowledgement for the AMD PDU with "Sequence Number" = x - 1.

If the timer expires and no STATUS PDU fulfilling the criteria above has been received:

- the Receiver shall be polled once more;
- the timer shall be restarted; and
- the new value of VT(S) shall be saved.

If a new poll is sent when the timer is active, the timer shall be restarted at the time specified above, and the value of VT(S) shall be saved.

b) Timer_Poll_Prohibit.

This timer shall only be used when so configured by upper layers. It is used to prohibit transmission of polls within a certain period. The value of the timer is signalled by upper layers.

In the UE this timer shall be started when the successful or unsuccessful transmission of an AMD PDU containing a poll is indicated by lower layer. In UTRAN it should be started when an AMD PDU containing a poll is submitted to lower layer.

From the time a poll is triggered until the timer expires, polling is prohibited. If another poll is triggered while polling is prohibited, its transmission shall be delayed until the timer expires (see subclause 9.7.1). Only one poll shall be transmitted when Timer_Poll_Prohibit expires even if several polls were triggered in the meantime. This timer shall not be affected by the reception of STATUS PDUs.

When Timer_Poll_Prohibit is not configured by upper layers, polling is never prohibited.

c) Timer_EPC.

This timer shall only be used when the EPC function is configured by upper layers. It is meant to account for the roundtrip delay, i.e. the time between the transmission of a status report and the reception of the first retransmitted AMD PDU. The initial value of the timer is signalled by upper layers.

In the UE, this timer shall be started when the successful or unsuccessful transmission of the first STATUS PDU of a status report is indicated by lower layer. In UTRAN it should be started when the first STATUS PDU of a status report is submitted to lower layer. Only after Timer_EPC expires shall VR(EP) be decremented as described in subclause 9.7.4.

d) Timer_Discard.

This timer shall be used when timer-based SDU discard is configured by upper layers. The value of the timer is signalled by upper layers. In the transmitter, a new timer is started upon reception of an SDU from upper layer.

In UM/TM, if a timer expires before the corresponding SDU is submitted to lower layer, "SDU discard without explicit signalling" specified in subclauses 11.2.4.3 and 11.1.4.2 shall be initiated. In AM, if a timer expires before the corresponding SDU is acknowledged, "SDU discard with explicit signalling" specified in subclause 11.6 shall be initiated.

e) Timer_Poll_Periodic.

This timer shall only be used when "timer based polling" is configured by upper layers. The value of the timer is signalled by upper layers. The timer shall be started when the RLC entity is created. When the timer expires, the RLC entity shall:

- restart the timer;
- if AMD PDUs are available for transmission or retransmission (not yet acknowledged):
  - trigger a poll.

f) Timer_Status_Prohibit.

This timer shall only be used when so configured by upper layers. It is meant to prohibit the Receiver from sending consecutive acknowledgement status reports. A status report is an acknowledgement status report if it contains any of the SUFIs LIST, BITMAP, RLIST or ACK. The value of the timer is signalled by upper layers.

In the UE, this timer shall be started when the successful or unsuccessful transmission of the last STATUS PDU of an acknowledgement status report is indicated by lower layer. In UTRAN it should be started when the last STATUS PDU of an acknowledgement status report is submitted to lower layer.

From the time an acknowledgement status report is triggered until the Timer_Status_Prohibit timer expires, acknowledgement is prohibited. If another such status report is triggered while acknowledgement is prohibited, its transmission shall be delayed until the timer expires (see subclause 9.7.2). The status report may be updated during this time. The transmission of SUFIs MRW, MRW_ACK, WINDOW or NO_MORE is not restricted.

When Timer_Status_Prohibit is not configured by upper layers, acknowledgment is not prohibited.

g) Timer_Status_Periodic.

This timer shall only be used when timer based status reporting is configured by upper layers.

This timer shall be started when the RLC entity is created. When the timer expires the transmission of a status report shall be triggered and the timer shall be restarted. This timer can be blocked by upper layers. The timer shall be restarted when upper layers indicate that it is no longer blocked.

h) Timer_RST.

This timer is meant to handle the loss of a RESET PDU by the peer entity, or the loss of a RESET ACK PDU from the peer entity. The value of the timer is signalled by upper layers.

In the UE this timer shall be started when the successful or unsuccessful transmission of a RESET PDU is indicated by lower layer. In UTRAN it should be started when a RESET PDU is submitted to lower layer.

Timer_RST shall only be stopped upon reception of a RESET ACK PDU (with same RSN as RESET PDU), i.e. this timer shall not be stopped when an RLC reset initiated by the peer RLC entity occurs. If this timer expires, the RESET PDU shall be retransmitted.

i) Timer_MRW.

This timer is used to trigger the retransmission of a status report containing an MRW SUFI field. The value of the timer is signalled by upper layers.

In the UE this timer shall be started when the successful or unsuccessful transmission of a STATUS PDU containing the MRW SUFI is indicated by lower layer. In UTRAN, it should be started when a STATUS PDU containing the MRW SUFI is submitted to lower layer.

Each time the timer expires the MRW SUFI is retransmitted and the timer is restarted. It shall be stopped when one of the termination criteria for the SDU discard with explicit signalling procedure is fulfilled (see subclause 11.6.4).

9.6 Protocol Parameters

The behaviour defined in this subclause is normative. The values of the protocol parameters defined in this subclause are signalled by upper layers.

a) MaxDAT.

The maximum number of transmissions of an AMD PDU is equal to MaxDAT – 1. This protocol parameter represents the upper limit for state variable VT(DAT). When VT(DAT) equals the value MaxDAT, either RLC RESET procedure or SDU discard procedure shall be initiated according to the configuration by upper layers.

b) Poll_PDU.

This protocol parameter indicates how often the transmitter shall poll the Receiver in the case where "polling every Poll_PDU PDU" is configured by upper layers. It represents the upper limit for the state variable VT(PDU). When VT(PDU) equals the value Poll_PDU a poll shall be transmitted to the peer entity.

c) Poll_SDU.

This protocol parameter indicates how often the transmitter shall poll the Receiver in the case where "polling every Poll_SDU SDU" is configured by upper layers. It represents the upper limit for state variable VT(SDU). When VT(SDU) equals the value Poll_SDU a poll shall be transmitted to the peer entity.

d) Poll_Window.

This protocol parameter indicates when the transmitter shall poll the Receiver in the case where "window-based polling" is configured by upper layers. A poll is triggered for each AMD PDU when $J \geq$ Poll_Window, where J is the transmission window percentage defined as:

$$J = \frac{(4096+VT(S)+1-VT(A)) \textbf{ mod } 4096}{VT(WS)} * 100 ,$$

where the constant 4096 is the modulus for AM described in subclause 9.4 and VT(S) is the value of the variable before the AMD PDU is submitted to lower layer.

e) MaxRST.

The maximum number of transmissions of a RESET PDU is equal to MaxRST – 1. This protocol parameter represents the upper limit for state variable VT(RST). When VT(RST) equals the value MaxRST, unrecoverable error shall be indicated to upper layers.

f) Configured_Tx_Window_Size.

This protocol parameter indicates both the maximum allowed transmission window size and the value for the state variable VT(WS).

g) Configured_Rx_Window_Size.

This protocol parameter indicates the reception window size.

h) MaxMRW.

The maximum number of transmissions of an MRW command is equal to MaxMRW. This protocol parameter represents the upper limit for state variable VT(MRW). When VT(MRW) equals the value MaxMRW, the RLC RESET procedure shall be initiated.

9.7 Specific functions

The functions defined in this subclause are normative.

9.7.1 Polling function for acknowledged mode

The Polling function is used by the Sender to request the peer RLC entity for a status report. The "Polling bit" in the AMD PDU indicates the poll request. There are several triggers for initiating the Polling function. Which of the triggers shall be used is configured by upper layers for each RLC entity. The following triggers can be configured:

1) Last PDU in buffer.

When an AMD PDU to be transmitted for the first time is submitted to lower layer, the Sender shall:

- if the AMD PDU is the last AMD PDU scheduled for transmission according to subclause 11.3.2 (i.e. no data received from upper layer remains to be segmented into AMD PDUs); or
- if the AMD PDU is the last AMD PDU that is allowed to transmit according to subclause 11.3.2.2:
  - trigger a poll for this AMD PDU.

2) Last PDU in Retransmission buffer.

When a retransmitted AMD PDU is submitted to lower layer, the Sender shall:

- if the AMD PDU is the last AMD PDU scheduled for retransmission according to subclause 11.3.2; or
- if the AMD PDU is the last of the AMD PDUs scheduled for retransmission that are allowed to transmit according to subclause 11.3.2.2:
  - trigger a poll for this AMD PDU.

3) Poll timer.

The timer Timer_Poll is started and stopped according to subclause 9.5 a). When the timer Timer_Poll expires the Sender triggers the Polling function.

4) Every Poll_PDU PDU.

The Sender triggers the Polling function for every Poll_PDU PDU. Both retransmitted and new AMD PDUs shall be counted.

5) Every Poll_SDU SDU.

The Sender triggers the Polling function for every Poll_SDU SDU. The poll shall be triggered for the first transmission of the AMD PDU that contains the "Length Indicator" indicating the end of the SDU.

6) Window based.

   The Sender triggers the Polling function when the condition described in subclause 9.6 d) ("Poll_Window") is fulfilled.

7) Timer based.

   The Sender triggers the Polling function periodically.

UTRAN should configure RLC to avoid deadlock situations.

The Poll Prohibit function is used by the Sender to delay the initiation of the Polling function. Usage of the Poll Prohibit function is configured by upper layers. The Poll Prohibit function consists of starting the timer Timer_Poll_Prohibit according to subclause 9.5 b) and delaying the Polling function according to the following rules:

When the Polling function is triggered, the Sender shall:

- if polling is not prohibited (see subclause 9.5 b)); and
- if there is one or more AMD PDUs to be transmitted or there are AMD PDUs not acknowledged by the Receiver:
  - initiate the Polling function by setting the polling bit according to subclause 11.3.2.1.1.
- otherwise (if there is no PDU to be transmitted and all PDUs have already been acknowledged):
  - not initiate the Polling function.

Upon expiry of the timer Timer_Poll_Prohibit, the Sender shall:

- if the Polling function was triggered at least once while the timer Timer_Poll_Prohibit was active; and
- if there is one or more AMD PDUs to be transmitted or there are AMD PDUs not acknowledged by the Receiver:
  - initiate the Polling function once by setting the polling bit according to subclause 11.3.2.1.1.
- otherwise (if there is no PDU to be transmitted and all PDUs have already been acknowledged):
  - not initiate the Polling function.

9.7.2 STATUS transmission for acknowledged mode

The Receiver transmits status reports to the Sender in order to inform the Sender about which AMD PDUs have been received and not received. Each status report consists of one or several STATUS PDUs. The Receiver shall trigger the transmission of a status report when receiving a poll request. Additionally, the following triggers for transmission of status reports are configurable by upper layers:

1) Detection of missing PDU(s).

   If the Receiver detects one or several missing AMD PDUs it shall trigger the transmission of a status report to the Sender.

2) Timer based status report transfer.

   The Receiver triggers the transmission of a status report to the Sender periodically. The timer Timer_Status_Periodic controls the time period according to subclause 9.5 g). When "Periodical Status blocking" is configured by upper layers, the trigger shall not be active.

3) The EPC mechanism.

   The timer Timer_EPC is started according to subclause 9.5 c) and the state variable VR(EP) is set and decreased according to subclause 9.7.4. If not all AMD PDUs requested for retransmission have been received before the

variable VR(EP) equalled zero, a new status report is triggered by the Receiver. A more detailed description of the EPC mechanism is given in subclause 9.7.4.

There are two functions that can prohibit the Receiver from sending a status report containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited. Upper layers control which functions should be used for each RLC entity. If any of the following functions is used the transmission of the status report shall be delayed, even if any of the triggering conditions above are fulfilled:

1) STATUS prohibit.

   The timer Timer_Status_Prohibit is started according to subclause 9.5 f). The Receiver is not allowed to transmit a status report while acknowledgement is prohibited (see subclause 9.5 f)). If a status report was triggered during this time, the status report is transmitted after the timer Timer_Status_Prohibit has expired, as described below.

2) The EPC mechanism.

   If the "EPC mechanism" is active and the transmission of a status report is triggered it shall be delayed until the "EPC mechanism" has ended, as described below.

When a status report is triggered the Receiver shall:

- if transmission of status reports is not prohibited by any of the functions "STATUS prohibit" or "EPC mechanism":
  - assemble and transmit the status report to the Sender, as specified in subclauses 11.5.2.2 and 11.5.2.3.
- otherwise (if the status report is prohibited by at least one of the functions "STATUS prohibit" or "EPC mechanism"):
  - if MRW, MRW_ACK or WINDOW SUFIs are required in the status report:
    - send a status report immediately excluding ACK, LIST, BITMAP, and RLIST SUFIs;
  - if ACK, LIST, BITMAP, or RLIST SUFIs are required in the status report:
    - delay sending these SUFIs until the prohibit function terminates.

Upon expiry of the timer Timer_Status_Prohibit or termination of the "EPC mechanism", the Receiver shall:

- if at least one status report was triggered during the time the transmission of a status reports was prohibited that could not be transmitted due to prohibition; and
- if transmission of a status reports is no longer prohibited by any of the functions "STATUS prohibit" or "EPC mechanism":
  - transmit one status report to the Sender, using the procedure described in subclause 11.5.2.3.

9.7.3 SDU discard function for acknowledged, unacknowledged, and transparent mode

The SDU discard function is used by the Sender to discharge RLC PDUs from the RLC PDU buffer, when the transmission of the RLC PDUs does not succeed for a period of time or for a number of transmissions. The SDU discard function allows to avoid buffer overflow. There are several alternative operation modes of the RLC SDU discard function. Upper layers control, which discard function shall be used for each RLC entity.

The following is a list of operation modes for the RLC SDU discard function, which are described in detail in the subsequent subclauses.

Table 9.2: List of criteria that control when to perform SDU discard

Operation mode	Presence
Timer based discard, with explicit signalling	Network controlled
Timer based discard, without explicit signalling	Network controlled
SDU discard after MaxDAT number of transmissions	Network controlled
No_discard after MaxDAT number of transmissions	Network controlled

9.7.3.1 Timer based discard, with explicit signalling

This alternative is only applicable to RLC entities operating in acknowledged mode. It uses a timer based triggering of SDU discard (Timer_Discard). This makes the SDU discard function insensitive to variations in the channel rate and provides means for exact definition of maximum delay. However, the SDU loss rate of the connection is increased as SDUs are discarded.

For every SDU received from upper layers, the Sender shall:

- start a timer Timer_Discard.

When the timer Timer_Discard of a SDU expires, the Sender shall:

- discard the SDU;
- if "Send MRW" is configured, or one or more segments of the discarded SDU were submitted to the lower layer:
  - utilise explicit signalling to inform the Receiver according to subclause 11.6.

NOTE: The support of the configuration "Send MRW" and the functionality connected with this configuration is implementation dependent.

9.7.3.2 Timer based discard, without explicit signalling

This alternative is only applicable to RLC entities operating in unacknowledged or transparent mode. It uses the same timer based trigger for SDU discard (Timer_Discard) as the one described in the subclause 9.7.3.1. The difference is that this discard method does not use any peer-to-peer signalling.

For every SDU received from upper layers, the Sender shall:

- start timer monitoring of the transmission time of the SDU.

When the transmission time exceeds the configured value for a SDU, the Sender shall:

- discard the SDU without explicit signalling (for RLC entities operating in unacknowledged mode apply subclause 11.2.4.3 for updating the state variables).

9.7.3.3 SDU discard after MaxDAT number of transmissions

This alternative uses the number of transmissions as a trigger for SDU discard, and is therefore only applicable for acknowledged mode RLC. This makes the SDU discard function dependent on the channel rate. Also, this variant of the SDU discard function strives to keep the SDU loss rate constant for the connection, on the cost of a variable delay.

If the number of times an AMD PDU is scheduled for transmission reaches MaxDAT, the Sender shall:

- discard all SDUs segments of which are contained in the AMD PDU; and
- utilise explicit signalling to inform the Receiver according to clause 11.6.

9.7.3.4 No_discard after MaxDAT number of transmissions

This alternative uses the number of transmissions, and is therefore only applicable for acknowledged mode RLC.

If the number of times an AMD PDU is scheduled for transmission reaches MaxDAT, the Sender shall:

- initiate the RLC Reset procedure (see subclause 11.3.4.4).

9.7.3.5 SDU discard not configured

If SDU discard has not been configured for an unacknowledged mode RLC entity, SDUs in the transmitter shall not be discarded unless the Transmission buffer is full.

When the Transmission buffer in an unacknowledged mode RLC entity is full, the Sender may:

- if segments of the SDU to be discarded have been submitted to lower layer:
  - discard the SDU without explicit signalling according to subclause 11.2.4.3.
- otherwise, if no segments of the SDU to be discarded have been submitted to lower layer:
  - remove the SDU from the Transmission buffer without utilising any of the discard procedures.

If SDU discard has not been configured for a transparent mode RLC entity, the Sender shall upon reception of new SDUs from upper layer:

- discard all SDUs received from upper layer in previous TTIs that are not yet submitted to lower layer;
- submit the new SDUs in the first possible TTI.

For an acknowledged mode RLC entity, an SDU discard mode is always configured.

9.7.4 The Estimated PDU Counter for acknowledged mode

The Estimated PDU Counter (EPC) is only applicable for RLC entities operating in acknowledged mode. The EPC is a mechanism configured by upper layers used for scheduling the retransmission of status reports in the Receiver. With this mechanism, the Receiver will send a new status report in which it requests for AMD PDUs not yet received. The time between two subsequent status report retransmissions is not fixed, but it is controlled by both the timer Timer_EPC and the state variable VR(EP), which adapt this time to the round trip delay and the current bit rate, indicated in the TFI, in order to minimise the delay of the status report retransmission.

When a status report is triggered by some mechanisms and it is submitted to lower layer (in UTRAN) or the successful or unsuccessful transmission of it is indicated by lower layer (in UE) to request for retransmitting one or more missing AMD PDUs, the variable VR(EP) is set equal to the number of requested AMD PDUs. At least one requested AMD PDU is needed to activate the EPC mechanism. The variable VR(EP) is a counter, which is decremented every transmission time interval with the estimated number of AMD PDUs that should have been received during that transmission time interval on the corresponding logical channel.

The timer Timer_EPC controls the maximum time that the variable VR(EP) needs to wait before it will start counting down. This timer starts immediately after a transmission of a retransmission request from the Receiver (when the first STATUS PDU of the status report is submitted to lower layer (in UTRAN) or the successful or unsuccessful transmission of it is indicated by lower layer(in UE)). The initial value of the timer Timer_EPC is configured by upper layers. It typically depends on the roundtrip delay, which consists of the propagation delay, processing time in the transmitter and Receiver and the frame structure. This timer can also be implemented as a counter, which counts the number of 10 ms radio frames that could be expected to elapse before the first requested AMD PDU is received.

If not all of these requested AMD PDUs have been received correctly when VR(EP) is equal to zero, a new status report will be transmitted and the EPC mechanism will be reset accordingly. The timer Timer_EPC will be started once more when the first STATUS PDU of the status report is submitted to lower layer (in UTRAN) or the successful or unsuccessful transmission of it is indicated by lower layer (in UE). If all of the requested AMD PDUs have been received correctly, the EPC mechanism ends.

9.7.5 Local Suspend function for acknowledged and unacknowledged mode

The upper layers may suspend an RLC entity.

When an RLC entity operating in unacknowledged mode is suspended by upper layers with the parameter N, the RLC entity shall:

- acknowledge the suspend request with a confirmation containing the current value of VT(US);
- not send UMD PDUs with "Sequence Number" SN≥VT(US)+N.

When an RLC entity operating in acknowledged mode is suspended by upper layers with the parameter N, the RLC entity shall:

- acknowledge the suspend request with a confirmation containing the current value of VT(S);
- not send AMD PDUs with "Sequence Number" SN≥VT(S)+N.

When an RLC entity operating in unacknowledged mode is resumed by upper layers, the RLC entity shall:

- resume data transfer procedure.

When an RLC entity operating in acknowledged mode is resumed by upper layers, the RLC entity shall:

- if the RLC entity is suspended and a RLC Reset procedure is not ongoing:
  - resume data transfer procedure.
- otherwise, if the RLC entity is suspended and a RLC Reset procedure is ongoing:
  - remove the suspend constraint;
  - resume the RLC reset procedure according to subclause 11.4.

9.7.6 RLC Stop, RLC Continue function for acknowledged and unacknowledged mode

The upper layer may stop an RLC entity.

When an RLC entity is stopped, the RLC timers are not affected.

When a RLC entity is stopped by upper layers, the RLC entity shall:

- not submit any RLC PDUs to lower layer or receive any RLC PDUs;
- delay triggered Polling functions or status transmissions until the RLC entity is continued.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the stop function until the end of the next TTI.

When a RLC entity is continued by upper layers, the RLC entity shall:

- if the RLC entity is stopped:
  - continue the data transmission and reception;
  - process the triggered Polling functions and status transmissions.
- otherwise, if the RLC is not stopped:
  - take no action.

9.7.7 RLC re-establishment function for acknowledged and unacknowledged mode

The upper layers may re-establish an RLC entity.

The RLC re-establishment function is applicable for AM and UM and is used when upper layers request an RLC entity to be re-established.

When an RLC entity is re-established by upper layers, the RLC entity shall:

- reset the state variables to their initial value;
- set the configurable parameters to their configured value;
- set the hyper frame number (HFN) in UL and DL to the value configured by upper layers;
- if the RLC entity is operating in unacknowledged mode:
  - if it is a receiving UM RLC entity:
    - discard all UMD PDUs.
  - if it is a transmitting UM RLC entity:
    - discard the RLC SDUs for which one or more segments have been submitted to a lower layer;
    - not stop Timer_Discard if the RLC SDU is not discarded.
- otherwise if the RLC entity is operating in acknowledged mode:
  - discard all AMD PDUs and control PDUs in both the receiving side and the transmitting side of the RLC entity;
  - stop all timers described in subclause 9.5 except Timer_Poll_Periodic and Timer_Status_Periodic.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the re-establishment function until the end of the next TTI.

9.7.8 Ciphering for acknowledged and unacknowledged mode

The ciphering function is performed in RLC, according to the following rules if a radio bearer is using a non-transparent RLC mode (AM or UM). The data unit that is ciphered, depends on the transmission mode as described below.

- For RLC UM mode, the ciphering unit is the UMD PDU excluding the first octet, i.e. excluding the UMD PDU header. This is shown below in Figure 9.19.

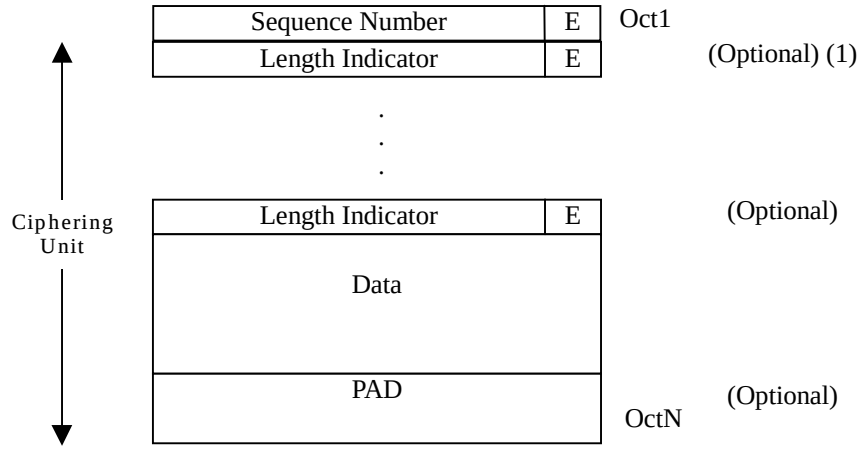


Figure 9.19: Ciphering unit for a UMD PDU

- For RLC AM mode, the ciphering unit is the AMD PDU excluding the first two octets, i.e. excluding the AMD PDU header. This is shown below in Figure 9.20.

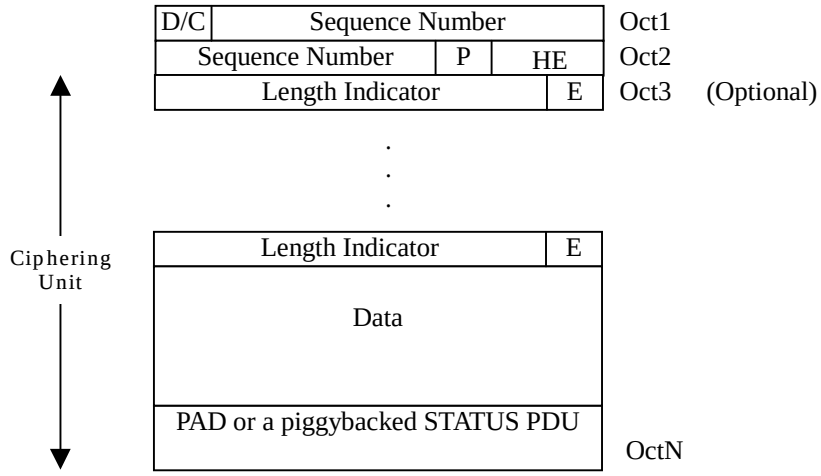


Figure 9.20: Ciphering unit for an AMD PDU

The ciphering algorithm and key to be used are configured by upper layers [8] and the ciphering method shall be applied as specified in [9].

The parameters that are required by RLC for ciphering are defined in [9] and are input to the ciphering algorithm. The parameters required by RLC which are provided by upper layers [8] are listed below:

- RLC AM HFN (Hyper frame number for radio bearers that are mapped onto RLC AM);
- RLC UM HFN (Hyper frame number for radio bearers that are mapped onto RLC UM);
- BEARER (defined as the radio bearer identifier in [9]. It will use the value RB identity –1 as in [8]);
- CK (Ciphering Key).

9.7.9 Reconfiguration of RLC parameters by upper layers

The RLC parameters for an RLC entity may be reconfigured (modifed) by upper layers.

When an RLC parameter is reconfigured by the upper layer, the UE shall:

start using the reconfigured value of the RLC parameter.

If the parameter Configured_Rx_Window_Size is reconfigured:

- the UE shall update the state variable VR(MR), (see clause 9.4);
- for AMD PDUs with "Sequence Number" x such that VR(MR)<=x<VR(H):
  - the UE may discard these AMD PDUs.

If the parameter Configured_Tx_Window_Size is reconfigured:

- the UE shall set the state variable VT(WS) equal to the Configured_Tx_Window_Size;
- the UE shall update the state variable VT(MS), (see clause 9.4);
- for AMD PDUs with "Sequence Number" x such that VT(MS)<=x<VT(S):
  - the UE shall not discard any AMD PDUs that are not positively acknowledged;
  - the UE may discard AMD PDUs that are positively acknowledged.

10 Handling of unknown, unforeseen and erroneous protocol data

Errors and the handling of errors defined in this clause are normative.

10.1 Erroneous Sequence Number

A STATUS PDU or Piggybacked STATUS PDU including "erroneous Sequence Number" is a STATUS PDU or Piggybacked STATUS PDU that contains:

- a LIST, BITMAP or RLIST SUFI in which the "Sequence Number" of at least one AMD PDU that is negatively acknowledged is outside the interval VT(A)≤"Sequence Number"≤ VT(S)-1; or
- an ACK SUFI in which "LSN" is outside the interval VT(A)≤"LSN"≤ VT(S).

If an AM RLC entity receives a STATUS PDU or a Piggybacked STATUS PDU including "erroneous Sequence Number", it shall:

- discard the STATUS PDU or the Piggybacked STATUS PDU;
- initiate the RLC reset procedure (see subclause 11.4).

10.2 Inconsistent status indication

If an AM RLC entity receives a STATUS PDU or a Piggybacked STATUS PDU that indicates different status for the same AMD PDU, it shall:

- discard the STATUS PDU or the Piggybacked STATUS PDU.

10.3 Invalid PDU format

If an UM or AM RLC entity receives a RLC PDU that contains reserved or invalid values (see subclause 9.2), it shall:

- discard the RLC PDU.

10.4 RLC PDU with CRC error

If an UM or AM RLC entity receives a RLC PDU with an error indication, it shall:

- discard the RLC PDU.

If a TM RLC entity receives a RLC PDU with an error indication, it shall:

- if "Delivery of Erroneous SDUs" is configured:
  - process the RLC PDU according to subclause 11.1.3;
- otherwise:
  - discard the RLC PDU.

11 Elementary procedures

Procedures defined in this clause are normative.

This description assumes elementary procedures. Interactions between procedures are not described.

11.1 Transparent mode data transfer procedure

11.1.1 General

The transparent mode data transfer procedure is used for transferring data between two RLC peer entities, which are operating in transparent mode. Data is transferred from Sender to Receiver. This procedure should only apply to entities in DATA_TRANSFER_READY state. Figure 11.1 below illustrates the elementary procedure for transparent mode data transfer.

Channels that can be used are DTCH, CCCH (uplink only), SHCCH (uplink only), BCCH and PCCH. The type of logical channel depends on if the RLC entity is located in the user plane (DTCH) or in the control plane (CCCH/BCCH/SHCCH/PCCH).

Figure 11.1: Transparent mode data transfer procedure

11.1.2 Transmission of TMD PDU

Upon a request of transparent mode data transfer from upper layer, the Sender shall:

- if no SDU discard configuration has been made by upper layers:
  - discard SDUs received in previous TTIs upon reception of new SDUs from upper layers (see subclause 9.7.3.5);
- otherwise (if "Timer Based SDU Discard without explicit signalling" is configured):
  - start a timer Timer_Discard for each SDU received from upper layers (see subclause 9.7.3);
- schedule the RLC SDUs that have been received from upper layer for transmission;
- if one or more RLC SDUs have been scheduled for transmission:
  - notify the lower layer of reception of data from upper layers;
  - perform the actions specified in subclause 11.1.2.2.

11.1.2.1 TMD PDU contents to set

The Sender shall set the data field of the TMD PDU to all or a subset of the data contained in the SDU as described in subclause 11.1.2.2.

11.1.2.2 Submission of TMD PDUs to the lower layer

If one or more RLC SDUs have been scheduled for transmission, according to subclause 11.1.2, the Sender shall:

- if it is configured for segmented operation:
  - inform the lower layer of the size of the next SDU to be sent;
  - segment the SDU according to the PDU size indicated by the lower layer.
- otherwise (the Sender is configured for non-segmented operation):
  - inform the lower layer of the number and size of SDUs available for transmission;

- submit to the lower layer, the requested number of TMD PDUs;
- buffer the SDUs that are not submitted to the lower layer according to the discard configuration (see subclause 9.7.3).

11.1.3 Reception of TMD PDU

Upon delivery by the lower layer of a set of TMD PDUs (received within one TTI), the Receiver shall:

- if it is configured for segmented operation:
  - reassemble the TMD PDUs received in one TTI into one RLC SDU.
- otherwise (it is configured for non-segmented operation):
  - treat each received TMD PDU as a SDU;
- if "Delivery of Erroneous SDUs" is configured as "no":
  - submit only the RLC SDUs received without error to upper layers through the TM-SAP.
- else if "Delivery of Erroneous SDUs" is configured as "yes":
  - submit all RLC SDUs to upper layers through the TM-SAP;
  - provide an error indication for each SDU received in error.
- otherwise if "Delivery of Erroneous SDUs" is configured as "No detect":
  - submit all RLC SDUs to upper layers through the TM-SAP.

If segmentation is performed in transparent mode RLC, an SDU is erroneous if one or more of the TMD PDUs received in a TTI contains an error. If segmentation is not performed, an SDU is erroneous if the corresponding TMD PDU is erroneous.

11.1.4 Abnormal cases

11.1.4.1 Void

11.1.4.2 SDU discard without explicit signalling

Upon expiry of the timer Timer_Discard in the Sender, the Sender shall:

- discard the associated SDU;
- if requested:
  - inform the upper layers of the discarded SDU.

In the case where the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the UE may wait until after it provides MAC with the requested set of PDUs before discarding the afore-mentioned SDU.

11.2 Unacknowledged mode data transfer procedure

11.2.1 General

The unacknowledged mode data transfer procedure is used for transferring data between two RLC peer entities, which are operating in unacknowledged mode. Data is transferred from Sender to Receiver. This procedure should only apply to RLC entities in DATA_TRANSFER_READY state or LOCAL_SUSPEND state. Figure 11.2 below illustrates the elementary procedure for unacknowledged mode data transfer.

Channels that can be used are DTCH, DCCH, CCCH (downlink only), CTCH, SHCCH (downlink only). The type of logical channel depends on if the RLC entity is located in the user plane (DTCH, CTCH) or in the control plane (DCCH/CCCH(downlink only)/SHCCH(downlink only)). One or several PDUs may be transmitted in each transmission time interval (TTI). For each TTI, MAC decides which PDU size shall be used and how many PDUs shall be transmitted.

Figure 11.2: Unacknowledged mode data transfer procedure

11.2.2 Transmission of UMD PDU

Upon a request of unacknowledged mode data transfer from upper layer, the Sender shall:

- if no SDU discard configuration has been made by upper layers:
  - only discard SDUs when the Transmission buffer is full (see subclause 9.7.3);
- if "Timer based SDU Discard without explicit signalling" is configured:
  - start a timer Timer_Discard for each SDU received from upper layer (see subclause 9.7.3);
- schedule the RLC SDUs received from upper layer for transmission;
- if one or more RLC SDUs have been scheduled for transmission:
  - notify the lower layer of reception of data from upper layers;
  - perform the actions specified in subclause 11.2.2.2.

A UMD PDU shall be considered to be a padding PDU if it consists only of an RLC Header with one length indicator (indicating that the rest of the PDU is padding) and padding.

11.2.2.1 UMD PDU contents to set

The Sender shall:

- set the field "Sequence Number" equal to VT(US);
- set a "Length Indicator" field for each SDU that ends in the UMD PDU according to subclause 9.2.2.8.

For each "Extension bit" field in the RLC header, the Sender shall:

- if the next field in the UMD PDU is a "Length Indicator":
  - set the "Extension bit" to "1";
- otherwise if the next field in the UMD PDU is data:
  - set the "Extension bit" to "0".

11.2.2.2 Submission of UMD PDUs to the lower layer

If one or more SDUs have been scheduled for transmission according to subclause 11.2.2, the Sender shall:

- inform the lower layer of the number and size of SDUs scheduled for transmission;
- segment, and if possible concatenate the SDUs according to the PDU sizes indicated by the lower layer;

- submit to the lower layer, the requested number of UMD PDUs;
- update VT(US) for each UMD PDU submitted to the lower layer (see subclause 9.4);
- buffer the SDUs that are not submitted to the lower layer according to the discard configuration (see subclause 9.7.3).

11.2.3 Reception of UMD PDU

Upon delivery of a set of UMD PDUs from the lower layer, the Receiver shall:

- update VR(US) according to each received UMD PDU (see subclause 9.4);
- if the updating step of VR(US) is not equal to one (i.e. one or more UMD PDUs are missing):
  - discard the SDUs that have segments in the missing UMD PDUs.
- if the special "Length Indicator" "1111 100" or "1111 1111 1111 100" is the first "Length Indicator" of a UMD PDU received on the downlink:
  - consider the first data octet in this UMD PDU as the first octet of an RLC SDU.
- reassemble the received UMD PDUs into RLC SDUs;
- submit the RLC SDUs to upper layers through the UM-SAP.

11.2.4 Abnormal cases

11.2.4.1 Length Indicator value reserved for UMD PDU

Upon delivery by the lower layer of an UMD PDU that contains a "Length Indicator" value specified to be reserved for UMD PDUs in this version of the protocol, the Receiver shall:

- ignore that UMD PDU.

11.2.4.2 Invalid length indicator value

If the "Length Indicator" of an UMD PDU has a value that is larger than the PDU size – RLC header size and is not one of the predefined values listed in the table of subclause 9.2.2.8, the Receiver shall:

- ignore the UMD PDU.

11.2.4.3 SDU discard without explicit signalling

Upon expiry of the timer Timer_Discard in the Sender, the Sender shall:

- discard the associated SDU;
- if requested:
  - inform the upper layers of the discarded SDU;
- for the first UMD PDU to be transmitted after the discard operation, the Sender shall:
  - increment VT(US) so that the "Sequence Number" field in this UMD PDU is incremented with two compared with the previous UMD PDU;
  - fill the first data octet in this UMD PDU with the first octet of an RLC SDU;
  - set the first "Length Indicator" in this UMD PDU to indicate that the previous RLC PDU was exactly filled with the last segment of an RLC SDU (to avoid that the Receiver unnecessarily discards an extra SDU).

In the case where the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the UE may wait until after it provides MAC with the requested set of UMD PDUs before discarding the afore-mentioned SDU.

11.3 Acknowledged mode data transfer procedure

11.3.1 General

The acknowledged mode data transfer procedure is used for transferring data between two RLC peer entities, which are operating in acknowledged mode. Data is transferred from Sender to Receiver. This procedure should only apply to RLC entities in DATA_TRANSFER_READY state or LOCAL_SUSPEND state. Figure 11.3 below illustrates the elementary procedure for acknowledged mode data transfer.

The AMD PDUs shall be transmitted on the DCCH logical channel if the Sender is located in the control plane and on the DTCH if it is located in the user plane. One or several PDUs may be transmitted in each transmission time interval (TTI) and MAC decides how many PDUs shall be transmitted in each TTI.

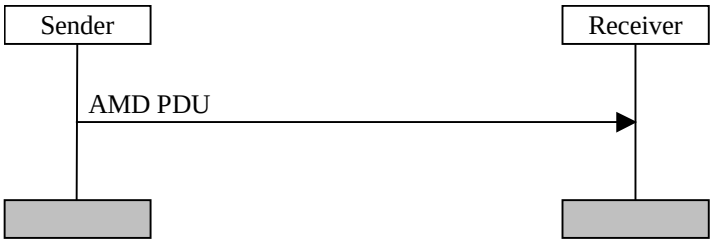


Figure 11.3: Acknowledged mode data transfer procedure

11.3.2 Transmission of AMD PDU

Upon a request of acknowledged mode data transfer from upper layers or upon retransmission of AMD PDUs, the Sender shall:

- when RLC SDUs are received from upper layers:
  - segment the RLC SDUs into AMD PDUs where the fixed PDU size is configured by upper layer;
  - set a "Length Indicator" field for each SDU that ends in the AMD PDU according to subclause 9.2.2.8;
  - if "Timer based SDU Discard with explicit signalling" is configured:
    - start a timer Timer_Discard for each SDU received from upper layer (see subclause 9.7.3);
  - schedule the AMD PDUs for transmission;
- if one or several AMD PDUs have been negatively acknowledged (see subclause 11.5.3):
  - schedule the AMD PDUs that were negatively acknowledged for retransmission;
- if a poll has been triggered by either the poll triggers "Poll timer" or "Timer based" (see subclause 9.7.1); and
- if polling is not prohibited (see subclause 9.5); and
- if no AMD PDU is scheduled for transmission or retransmission:
  - if the value of "Configured_Tx_Window_Size" is larger than or equal to "2048":
    - select the AMD PDU with "Sequence Number" equal to VT(S)-1.
  - otherwise if the "Configured_Tx_Window_Size" is less than "2048";
    - select the AMD PDU with "Sequence Number" equal to VT(S)-1; or

- select an AMD PDU that has not yet been acknowledged by the peer entity;

- schedule the selected AMD PDU for retransmission (in order to transmit a poll).

Each time an AMD PDU is scheduled for transmission or retransmission, the Sender shall:

- increment the value of the corresponding VT(DAT);
- if VT(DAT) = MaxDAT:
  - perform the actions specified in subclause 11.3.3a;
- e1se:
  - notify the lower layer that data is available for transmission;
  - perform the actions specified in subclause 11.3.2.2.

In AM, a PDU shall be considered to be a padding PDU if it is:

- an AMD PDU consisting only of an RLC Header with one "Length Indicator" (indicating that the rest of the PDU is padding) and padding; or
- a STATUS PDU consisting only of a NO_MORE SUFI.

11.3.2.1 AMD PDU contents to set

If the AMD PDU is transmitted for the first time, the Sender shall:

- set the "Sequence Number" field equal to VT(S);
- set a "Length Indicator" field for each SDU that ends in the AMD PDU according to subclause 9.2.2.8;
- set the "Polling bit" to the value specified in subclause 11.3.2.1.1.

Otherwise if the AMD PDU is retransmitted:

- use the same value of the "Sequence Number" field as in the original transmission of the AMD PDU;
- if the "Length Indicator" fields needed in the AMD PDU according to subclause 9.2.2.8 has changed due to that a piggybacked STATUS PDU is included in the AMD PDU or a piggybacked STATUS PDU was included in the previous transmission of the AMD PDU:
  - update the "Length Indicator" fields according to 9.2.2.8.
- set the "Polling bit" to the value specified in subclause 11.3.2.1.1.

11.3.2.1.1 Setting of the Polling bit

The Sender shall:

- if a poll has been triggered by one or several poll triggers (see subclause 9.7.1):
  - if polling is not prohibited, see subclause 9.5:
    - set the "Polling bit" in the AMD PDU header to "1";
- otherwise:
  - set the "Polling bit" in the AMD PDU header to "0".

11.3.2.1.2 Void

11.3.2.2 Submission of AMD PDUs to lower layer

If one or more AMD PDUs have been scheduled for transmission or retransmission according to subclause 11.3.2, the Sender shall:

- not submit any AMD PDUs to lower layer that is not allowed to transmit. AMD PDUs are only allowed to transmit:
  - if the AMD PDU has a "Sequence Number" < VT(MS) or the AMD PDU has a "Sequence Number" equal to VT(S)-1; and
  - if the AMD PDU is not restricted to be transmitted by the local suspend function, see subclause 9.7.5.
- inform the lower layer of both the numbers of AMD PDUs scheduled and allowed for transmission or retransmission;
- set the AMD PDU contents according to subclause 11.3.2.1;
- submit to the lower layer the requested number of AMD PDUs;
- treat retransmissions with higher priority than AMD PDUs transmitted for the first time;
- update the state variables in clause 9.4 for each AMD PDU submitted to lower layer except VT(DAT) which has already been updated, see subclause 11.3.2;
- if the "Polling bit" is set to "1" in any of the AMD PDUs; and
- if the timer Timer_Poll is configured;
  - start the timer Timer_Poll according to subclause 9.5;
- buffer the AMD PDUs that are not submitted to the lower layer according to the discard configuration (see subclause 9.7.3).

11.3.3 Reception of AMD PDU by the Receiver

Upon reception of an AMD PDU, the Receiver shall:

- update VR(R), VR(H) and VR(MR) state variables for each received AMD PDU (see clause 9.4);
- if a received AMD PDU includes a "Polling bit" set to "1", or "Missing PDU Indicator" is configured and the Receiver detects that a PDU is missing:
  - initiate the STATUS PDU transfer procedure;
- reassemble the received AMD PDUs into RLC SDUs;
- if "In-Sequence Delivery" is configured:
  - deliver the RLC SDUs in-sequence (i.e. in the same order as the RLC SDUs were originally transmitted by the peer entity) to upper layers through the AM-SAP.
- otherwise:
  - deliver the RLC SDUs in arbitrary order to upper layers through the AM-SAP.

11.3.3a Reached maximum number of attempts

If VT(DAT) = MaxDAT, the Sender shall:

- if "No_discard after MaxDAT number of transmissions" is configured:
  - initiate the RLC reset procedure, see subclause 11.4.
- if "SDU discard after MaxDAT number of transmissions" is configured:

- initiate the "SDU discard with explicit signalling" procedure for the corresponding SDU, see subclause 11.6.

11.3.4 Abnormal cases

11.3.4.1 Void

11.3.4.2 Receiving an AMD PDU outside the reception window

Upon reception of an AMD PDU with "Sequence Number" outside the interval VR(R)≤SN<VR(MR), the Receiver shall:

- discard the AMD PDU;
- if the "polling bit" in the discarded AMD PDU is set to "1":
  - initiate the STATUS PDU transfer procedure.

11.3.4.3 Timer_Discard timeout

11.3.4.3.1 SDU discard with explicit signalling

Upon expiry of the timer Timer_Discard, the Sender shall:

- initiate the SDU discard with explicit signalling procedure, see subclause 11.6.2.

In the case where the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the UE may wait until after it provides MAC with the requested set of PDUs before discarding the afore-mentioned SDUs.

11.3.4.4 Void

11.3.4.5 Invalid length indicator value

If the "Length Indicator" of an AMD PDU has a value that is larger than the PDU size – RLC header size and is not one of the predefined values listed in the table of subclause 9.2.2.8, the Receiver shall:

- ignore that AMD PDU.

11.3.4.6 Length Indicator value reserved for AMD PDU

Upon delivery by the lower layer of an AMD PDU that contains a "Length Indicator" value specified to be reserved for AMD PDUs in this version of the protocol, the Receiver shall:

- ignore that AMD PDU.

11.3.4.7 Void

11.3.4.8 Receiving an AMD PDU within the reception window more than once (Handling of Duplicates)

Upon reception of an AMD PDU with a "Sequence Number" within the interval VR(R)≤SN<VR(MR), for which "Sequence Number" an AMD PDU has already been received, the Receiver shall:

- discard the AMD PDU;

- consider the AMD PDU with this “Sequence Number” as having been correctly received in the next status report to be transmitted;

- if the "polling bit" in the discarded AMD PDU is set to "1":

  - initiate the STATUS PDU transfer procedure.

11.4 RLC reset procedure

11.4.1 General

The RLC reset procedure is used to reset two RLC peer entities, which are operating in acknowledged mode. Figure 11.4 below illustrates the elementary procedure for an RLC reset. During the reset procedure the hyper frame numbers (HFN) in UTRAN and UE are synchronised. Two HFNs used for ciphering needs to be synchronised, DL HFN in downlink and UL HFN in uplink. In the reset procedure, the highest UL HFN and DL HFN used by the RLC entity in the transmitting sides, i.e. the HFNs associated with AMD PDUs of "Sequence Number"=VT(S)-1 if at least one AMD PDU had been transmitted or of "Sequence Number"=0 if no AMD PDU had been transmitted, are exchanged between UE and UTRAN.

The RESET PDUs and the RESET ACK PDUs have higher priority than AMD PDUs.

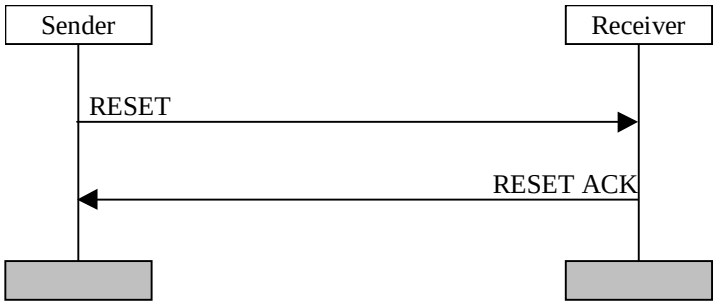


Figure 11.4: RLC reset procedure

11.4.2 Initiation

The Sender shall:

- if one of the following triggers is detected:

1) "No_Discard after MaxDAT number of transmissions" is configured and VT(DAT) equals the value MaxDAT (see subclause 9.7.3.4);

2) VT(MRW) equals the value MaxMRW;

3) A STATUS PDU including "erroneous Sequence Number" is received (see clause 10);

   - stop transmitting any AMD PDU or STATUS PDU;

   - increment VT(RST) by 1;

   - if VT(RST) = MaxRST:

     - perform the actions specified in subclause 11.4.4a.

   - else (if VT(RST) < MaxRST):

     - submit a RESET PDU to the lower layer;

     - start the timer Timer_RST.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the RLC reset procedure until the end of the next TTI.

When a reset procedure has been initiated it can only be ended upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU, or upon request of re-establishment or release from upper layer, a reset procedure is not interrupted by the reception of a RESET PDU from the peer entity.

11.4.2.1 RESET PDU contents to set

The Sender shall:

- set the HFNI field to the currently highest used HFN (DL HFN when the RESET PDU is sent by UTRAN or UL HFN when the RESET PDU is sent by the UE);
- set the RSN field to the sequence number of the RESET PDU. The sequence number of the first RESET PDU after the AM entity is established or re-established shall be "0". This sequence number is incremented every time a new RESET PDU is transmitted, but not when a RESET PDU is retransmitted.

11.4.3 Reception of the RESET PDU by the Receiver

Upon reception of a RESET PDU the Receiver shall:

- if the RSN value in the RESET PDU is the same as the RSN value in the last received RESET PDU:
  - only submit a RESET ACK PDU to the lower layer with the contents set exactly as in the last transmitted RESET ACK PDU (i.e., in this case the RLC entity is not reset).
- if the RESET PDU is the first RESET PDU received since the entity was (re-)established or the RSN value is different from the RSN value in the last received RESET PDU:
  - submit a RESET ACK PDU to the lower layer with the content set as specified in subclause 11.4.3.1;
  - reset the state variables described in subclause 9.4 except VT(RST) to their initial values;
  - stop all the timers described in subclause 9.5 except Timer_RST, Timer_Discard, Timer_Poll_Periodic and Timer_Status_Periodic;
  - reset configurable parameters to their configured values;
  - discard all RLC PDUs in the receiving side of the AM RLC entity;
  - discard all RLC SDUs that were transmitted before the reset in the transmitting side of the AM RLC entity;
  - set the HFN (DL HFN when the RESET PDU is received in UE or UL HFN when the RESET PDU is received in UTRAN) equal to the HFNI field in the received RESET PDU;
  - increase with one the UL HFN and DL HFN, and the updated HFN values shall be used for the first transmitted and received AMD PDUs after the reset procedure.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the RLC SDUs discard in the transmitting side of the AM RLC entity until the end of the next TTI.

11.4.3.1 RESET ACK PDU contents to set

The Receiver shall:

- set the hyper frame number indicator field (HFNI) to the currently highest used HFN (DL HFN when the RESET ACK PDU is sent by UTRAN or UL HFN when the RESET ACK PDU is sent by the UE);
- set the RSN field to the same value as in the corresponding received RESET PDU.

11.4.4 Reception of the RESET ACK PDU by the Sender

Upon reception of a RESET ACK PDU, the Sender shall:

- if the Sender has already transmitted a RESET PDU which has not been yet acknowledged by a RESET ACK PDU:
  - if the received RSN value is the same as the one in the corresponding RESET PDU:
    - set the HFN value (DL HFN when the RESET ACK PDU is received in UE or UL HFN when the RESET ACK PDU is received in UTRAN) to the HFNI field of the received RESET ACK PDU;
    - reset the state variables described in subclause 9.4 to their initial values;
    - stop all the timers described in subclause 9.5 except Timer_Discard, Timer_Poll_Periodic and Timer_Status_Periodic;
    - reset configurable parameters to their configured values;
    - discard all RLC PDUs in the receiving side of the AM RLC entity;
    - discard all RLC SDUs that were transmitted before the reset in the transmitting side of the AM RLC entity;
    - increase with one the UL HFN and DL HFN, and the updated HFN values shall be used for the first transmitted and received AMD PDUs after the reset procedure;
  - otherwise (if the received RSN value is not the same as the one in the corresponding RESET PDU):
    - discard the RESET ACK PDU;
- otherwise (if the Sender has not transmitted a RESET PDU which has not been yet acknowledged by a RESET ACK PDU):
  - discard the RESET ACK PDU.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the RLC SDUs discard in the transmitting side until the end of the next TTI.

11.4.4a Reached maximum number of attempts

If VT(RST) = MaxRST, the Sender shall:

- terminate the ongoing RLC RESET procedure;
- stop the timer Timer_RST if it was started;
- indicate unrecoverable error to upper layer.

11.4.5 Abnormal cases

11.4.5.1 Timer_RST timeout

If Timer_RST expires before the reset procedure is terminated, the Sender shall:

- increment VT(RST) by one;
- if VT(RST)<MaxRST:
  - set the RESET PDU as previously transmitted (even if additional SDUs were discarded in the mean-time);
  - transmit the RESET PDU;
  - restart Timer_RST.
- else (if VT(RST) = MaxRST):
  - perform the actions specified in subclause 11.4.4a.

11.4.5.2 Void

11.4.5.3 Reception of the RESET PDU by the Sender

Upon reception of a RESET PDU, the Sender shall:

- submit a RESET ACK PDU to the lower layer with the content set as specified in subclause 11.4.3.1;
- reset the state variables described in subclause 9.4 except VT(RST) to their initial values;
- stop all the timers described in subclause 9.5 except Timer_RST, Timer_Discard, Timer_Poll_Periodic and Timer_Status_Periodic;
- reset configurable parameters to their configured values;
- discard all RLC PDUs in the receiving side of the AM RLC entity;
- discard all RLC SDUs that were transmitted before the reset in the transmitting side of the AM RLC entity;
- set the HFN (DL HFN when the RESET PDU is received in UE or UL HFN when the RESET PDU is received in UTRAN) equal to the HFNI field in the received RESET PDU.

NOTE: If the TFC selection exchange has been initiated by sending the RLC Entity Info parameter to MAC, the RLC entity may delay the RLC SDUs discard in the transmitting side until the end of the next TTI.

11.5 STATUS report transfer procedure

11.5.1 General

The status report transfer procedure is used for transferring of status information between two RLC peer entities, which are operating in acknowledged mode. Figure 11.5 below illustrates the elementary procedure for status report transfer. A status report consists of one or several STATUS PDUs.

In case two logical channels are configured in the uplink, control PDUs are transmitted on the second logical channel. In case two logical channels are configured in the downlink, control PDUs can be transmitted on any of the two logical channels.

The STATUS PDUs have higher priority than AMD PDUs.

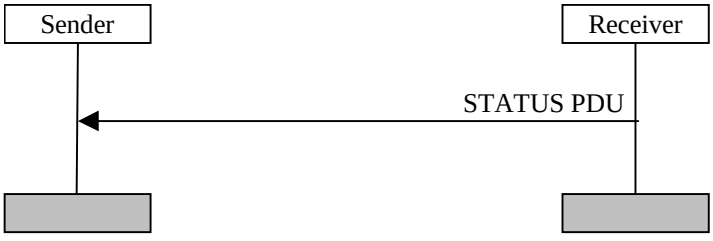


Figure 11.5: Status report transfer procedure

11.5.2 Initiation

The Receiver shall:

- if one of the following triggers is detected:

1) The "Polling bit" in a received AMD PDU is set to "1";
2) "Missing PDU Indicator" is configured and a missing AMD PDU is detected;
3) The "Timer based STATUS transfer" is configured and the timer Timer_Status_Periodic has expired:

- act on the trigger as specified in subclause 9.7.2.

11.5.2.1 Piggybacked STATUS PDU

The Receiver may:

- if STATUS PDU(s) to be sent fit into padding octets in AMD PDU(s) to be sent:
  - piggyback a STATUS PDU on the AMD PDU to be sent.

Submission of a piggybacked STATUS PDU in an AMD PDU to the lower layer follows the same rules as an ordinary STATUS PDU.

11.5.2.2 STATUS PDU contents to set

On triggering of a status report, the Receiver shall:

- if neither the "STATUS prohibit" nor "EPC mechanism" are active:
  - include negative acknowledgements for all AMD PDUs detected as missing;
  - include positive acknowledgements for all AMD PDUs received up to at least VR(R);
- if an MRW SUFI assembled as specified in subclause 11.6.2.2 had not been sent:
  - optionally include the MRW SUFI;
- if an MRW_ACK SUFI assembled as specified in subclause 11.6.2.2 is awaiting transmission:
  - optionally include the MRW_ACK SUFI;
- if the Sender's transmission window is to be updated:
  - optionally include the WINDOW SUFI;
- if all SUFIs can be accommodated in one STATUS PDU:
  - construct the status report using one STATUS PDU, using one of the allowed PDU sizes;
  - if the SUFIs included do not fill the entire STATUS PDU:
    - terminate the STATUS PDU with the ACK or NO_MORE SUFI;
    - use padding in the remainder of the STATUS PDU;
  - otherwise (SUFIs included fill the entire STATUS PDU):
    - ACK or NO_MORE SUFIs need not be included in that STATUS PDU;
- otherwise (the status report is segmented):
  - construct STATUS PDUs including only complete SUFIs using one of the allowed PDU sizes. The set of STATUS PDUs shall accommodate all the SUFIs to form the complete status report. Indication of the same AMD PDU shall not be given in more than one STATUS PDU of a status report, but the ACK SUFI can be present in more than one STATUS PDU of a status report;
  - if any STATUS PDU constructed is not entirely filled with SUFIs:
    - terminate that STATUS PDU with the ACK or NO_MORE SUFI;
    - use padding in the remainder of that STATUS PDU.
  - otherwise (SUFIs included fill the entire STATUS PDU):
    - ACK or NO_MORE SUFIs should not be included in that STATUS PDU.

Which SUFI fields to use is implementation dependent. Bitmap SUFI is used to indicate both received and/or missing AMD PDUs. List SUFI and/or Relative List SUFI are used to indicate missing AMD PDUs only. Acknowledgement SUFI is used to indicate the received AMD PDUs. (For SUFI details see 9.2.2.11.) No information shall be given for AMD PDUs with "Sequence Number"≥VR(H), i.e. AMD PDUs that have not yet reached the Receiver.

11.5.2.3 Submission of STATUS PDUs to the lower layer

The Receiver shall:

- inform the lower layer of the STATUS PDUs scheduled for transmission;
- submit to the lower layer, the requested number of PDUs (STATUS PDUs, piggybacked AMD/STATUS PDUs and optionally AMD PDUs, see also subclause 11.3.2.2);
- if "Timer based STATUS transfer" is configured and the timer Timer_Status_Periodic has expired:
  - restart the timer Timer_Status_Periodic according to subclause 9.5 f);
- if the "EPC mechanism" is configured:
  - start the timer Timer_EPC according to subclause 9.5 c), and set VR(EP) equal to the number of AMD PDUs requested to be retransmitted;
- if the STATUS PDU includes the MRW SUFI:
  - start the timer Timer_MRW according to subclause 9.5 i).

11.5.3 Reception of the STATUS PDU by the Sender

Upon reception of the STATUS PDU/piggybacked STATUS PDU, the Sender shall:

- if an RLC SDU is positively acknowledged by the STATUS PDU:
  - if requested:
    - inform the upper layers of the reception of the RLC SDU by the peer AM RLC entity.
- update the state variables VT(A) and VT(MS) according to the received STATUS PDU/piggybacked STATUS PDU;
- if the STATUS PDU includes negatively acknowledged AMD PDUs:
  - initiate the acknowledged data transfer procedure; and
  - retransmit these AMD PDUs. Retransmitted AMD PDUs shall have higher priority than AMD PDUs to be transmitted for the first time;
  - if an AMD PDU is negatively acknowledged more than once in a STATUS PDU:
    - retransmit the AMD PDU only once.
- if the STATUS PDU includes the MRW SUFI:
  - take the actions specified in subclause 11.6.3.
- if the STATUS PDU includes the MRW_ACK SUFI:
  - take the actions specified in subclause 11.6.4.
- if the STATUS PDU includes the WINDOW SUFI:
  - update the current transmission window size, VT(WS).

11.5.4 Abnormal cases

11.5.4.1 VR(EP) equals zero and the requested AMD PDUs have not been received

If the EPC mechanism is configured and VR(EP) equals zero and not all AMD PDUs requested for retransmission have been received, the Receiver shall:

- retransmit the status report. The retransmitted status report may contain new or different SUFI fields in order to indicate that some previously lost AMD PDUs have been received and that some additional AMD PDUs have been lost.

11.6 SDU discard with explicit signalling procedure

11.6.1 General

The SDU discard with explicit signalling procedure is used for discarding SDUs and transferring the discard information between two peer entities, which are operating in acknowledged mode. The Sender shall discard an SDU that has not been successfully transmitted for a period of time or for a number of transmissions, and send a Move Receiving Window (MRW) SUFI to the Receiver. According to the MRW SUFI, the Receiver shall discard AMD PDUs carrying that SDU and update the reception window. Figure 11.6 below illustrates the elementary procedure for SDU discard with explicit signalling.

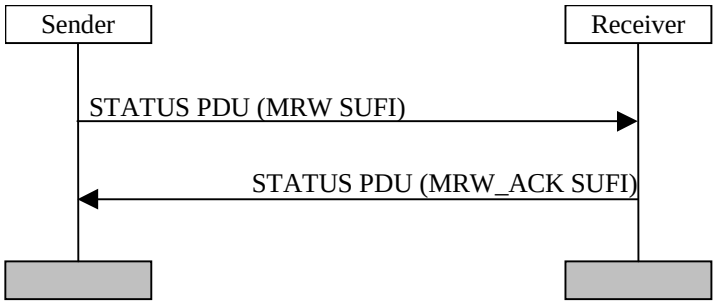


Figure 11.6: SDU discard with explicit signalling

11.6.2 Initiation

The Sender shall initiate the SDU discard with explicit signalling procedure if one of the following triggers is detected:

- "Timer based SDU discard with explicit signalling" is configured, Timer_Discard expires for an SDU, and one or more segments of the SDU have been submitted to lower layer;
- "Timer based SDU discard with explicit signalling" is configured, Timer_Discard expires for an SDU, and "Send MRW" is configured;
- "SDU discard after MaxDAT number of transmissions" is configured, and MaxDAT number of transmissions is reached (i.e. VT(DAT) ≥ MaxDAT) for an AMD PDU.

Upon initiation of the SDU discard with explicit signalling procedure, the Sender shall:

- if "Timer based SDU discard with explicit signalling" is configured:
  - discard all SDUs up to and including the SDU for which the timer Timer_Discard expired.
- if "SDU discard after MaxDAT number of transmissions" is configured:
  - discard all SDUs that have segments in AMD PDUs with "Sequence Number" SN inside the interval VT(A) ≤ SN ≤ X, where X is the value of the "Sequence Number" of the AMD PDU with VT(DAT) ≥ MaxDAT.
- if requested:

- inform the upper layers of the discarded SDUs.

- discard all AMD PDUs including segments of the discarded SDUs, unless they also carry a segment of a SDU whose timer has not expired;

- if more than 15 discarded SDUs are to be informed to the Receiver (see subclause 11.6.2.2):

  - if "Send MRW" is not configured:

    - assemble an MRW SUFI with the discard information of the SDUs.

  - otherwise ("Send MRW" is configured):

    - assemble an MRW SUFI with the discard information of the first 15 SDUs; and

    - include the discard information of the rest SDUs in another MRW SUFI which shall be sent by the next SDU discard with explicit signalling procedure (after the current SDU discard with explicit signalling procedure is terminated).

- otherwise (less than or equal to 15 discarded SDUs are to be informed to the Receiver):

  - assemble an MRW SUFI with the discard information of the SDUs.

- schedule and submit to lower layer a STATUS PDU/piggybacked STATUS PDU containing the MRW SUFI;

- if SN_MRW_{LENGTH} in the MRW SUFI >VT(S):

  - update VT(S) to SN_MRW_{LENGTH}.

- start a timer Timer_MRW according to subclause 9.5.

If a new SDU discard with explicit signalling procedure is triggered when the timer Timer_MRW is active, no new MRW SUFIs shall be sent before the current SDU discard with explicit signalling procedure is terminated by one of the termination criteria specified in subclause 11.6.4.

11.6.2.1 Void

11.6.2.2 STATUS PDU contents to set

The Sender shall:

- if "Send MRW" is configured:

  - if the last discarded SDU ended in an AMD PDU, and its "Length Indicator" is present in the same AMD PDU, and no new SDU is present inside this AMD PDU:

    - set the last SN_MRW_i field in the MRW SUFI to 1 + "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the last discarded SDU;

    - set the N_{LENGTH} field in the MRW SUFI to "0000".

  - otherwise:

    - set the last SN_MRW_i field in the MRW SUFI to the "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the last discarded SDU;

    - set the N_{LENGTH} field in the MRW SUFI so that the last data octet to be discarded in the Receiver shall be the octet indicated by the N_{LENGTH}:th "Length Indicator" field of the AMD PDU which contains the "Length Indicator" of the last discarded SDU;

  - set each of the other SN_MRW_i fields in the MRW SUFI to the "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the i:th discarded SDU.

- otherwise ("Send MRW" is not configured):

- if the last SDU to be discarded in the Receiver ended in an AMD PDU, and its "Length Indicator" is present in the same AMD PDU, and no new SDU is present inside this AMD PDU:
  - set the last SN_MRW_i field in the MRW SUFI to 1 + "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the last SDU to be discarded in the Receiver;
  - set the N_{LENGTH} field in the MRW SUFI to "0000".
- otherwise:
  - set the last SN_MRW_i field in the MRW SUFI to the "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the last SDU to be discarded in the Receiver;
  - set the N_{LENGTH} field in the MRW SUFI so that the last data octet to be discarded in the Receiver shall be the octet indicated by the N_{LENGTH}:th "Length Indicator" field of the AMD PDU which contains the "Length Indicator" of the last SDU to be discarded in the Receiver;
- optionally set each of the other SN_MRW_i fields in the MRW SUFI to the "Sequence Number" of the AMD PDU which contains the "Length Indicator" of the i:th SDU to be discarded in the Receiver;

- if the MRW SUFI contains only one SN_MRW_i field and the value of SN_MRW_i field ≥ VT(A)+Configured_Tx_Window_Size:
  - set the LENGTH field in the MRW SUFI to "0000".
- otherwise:
  - set the LENGTH field in the MRW SUFI to the number of SN_MRW_i fields in the same MRW SUFI. In this case, SN_MRW_1 shall be in the interval $VT(A) \leq SN\_MRW_1 < VT(A)+Configured\_Tx\_Window\_Size$.

11.6.3 Reception of the STATUS PDU by the Receiver

Upon reception of the STATUS PDU/piggybacked STATUS PDU containing an MRW SUFI, the Receiver shall:

- if the LENGTH field in the received MRW SUFI is "0000":
  - consider SN_MRW_1 to be above or equal to VR(R).
- otherwise:
  - consider SN_MRW_1 to be less than VR(MR);
- consider all the SN_MRW_is other than SN_MRW_1 to be in sequential order within the list and sequentially above or equal to SN_MRW_{i-1}.
- discard AMD PDUs up to and including the PDU with sequence number $SN\_MRW_{LENGTH}-1$;
- if the N_{LENGTH} field in the received MRW SUFI is "0000":
  - reassemble from the first data octet of the AMD PDU with sequence number SN_MRW_{LENGTH} after the discard.
- otherwise:
  - discard further the data octets in the AMD PDU with sequence number SN_MRW_{LENGTH} up to and including the octet indicated by the N_{LENGTH}:th "Length Indicator" field of the PDU with sequence number SN_MRW_{LENGTH};
  - reassemble from the succeeding data octet in the AMD PDU with sequence number SN_MRW_{LENGTH} after the discard;
- if "Send MRW" is configured:
  - inform upper layers about all of the discarded SDUs that were not previously delivered to upper layer or discarded by other MRW SUFIs;

- update the state variables VR(R), VR(H) and VR(MR) according to the received STATUS PDU/piggybacked STATUS PDU;
- assemble a MRW_ACK SUFI according to subclause 11.6.3.1;
- schedule and submit to lower layer a STATUS PDU/piggybacked STATUS PDU containing the MRW_ACK SUFI.

11.6.3.1 STATUS PDU contents to set

The Receiver shall:

- set the SN_ACK field in the MRW_ACK SUFI to the new value of VR(R), updated after reception of the MRW SUFI;
- if the SN_ACK field in the MRW_ACK SUFI is set equal to the SN_MRW_{LENGTH} field in the received MRW SUFI:
  - set the N field in the MRW_ACK SUFI to the N_{LENGTH} field in the received MRW SUFI.
- otherwise:
  - set the N field in the MRW_ACK SUFI to "0000".
- include the MRW_ACK SUFI in the next STATUS PDU/piggybacked STATUS PDU to be transmitted, according to subclause 11.5.2.

11.6.4 Termination

The Sender shall terminate the SDU discard with explicit signalling procedure if one of the following criteria is fulfilled:

- a STATUS PDU/piggybacked STATUS PDU containing an MRW_ACK SUFI is received, and the SN_ACK field in the received MRW_ACK SUFI > the SN_MRW_{LENGTH} field in the transmitted MRW_SUFI, and the N field in the received MRW_ACK SUFI is set equal to "0000";
- a STATUS PDU/piggybacked STATUS PDU containing an MRW_ACK SUFI is received, and the SN_ACK field in the received MRW_ACK SUFI = the SN_MRW_{LENGTH} field in the transmitted MRW_SUFI, and the N field in the received MRW_ACK SUFI is set equal to the N_{LENGTH} field in the transmitted MRW SUFI;
- a STATUS PDU/piggybacked STATUS PDU containing an ACK SUFI is received, and this STATUS PDU/piggybacked STATUS PDU indicates that all AMD PDUs up to and including the AMD PDU with "Sequence Number" equal to the SN_MRW_{LENGTH} field in the transmitted MRW SUFI has been received or discarded by the peer entity.

Upon termination of the SDU discard with explicit signalling procedure, the Sender shall:

- stop the timer Timer_MRW;
- update VT(A) and VT(MS) according to the received STATUS PDU/piggybacked STATUS PDU;

The Sender shall not confirm to upper layers the SDUs that are requested to be discarded.

11.6.4a Reached maximum number of attempts

If VT(MRW) = MaxMRW, the Sender shall:

- terminate the SDU discard with explicit signalling procedure;
- stop the timer Timer_MRW if it was started;
- initiate the RLC RESET procedure (see subclause 11.4).

11.6.5 Expiration of timer Timer_MRW

If Timer_MRW expires before the discard procedure is terminated, the Sender shall:

- increment VT(MRW) by one;
- if VT(MRW)<MaxMRW:
  - set the MRW SUFI as previously transmitted (even if additional SDUs were discarded in the mean-time);
  - include the MRW SUFI in a new status report (if other SUFIs are included, their contents shall be updated);
  - transmit the status report by either including it in a STATUS PDU or piggybacked in an AMD PDU;
  - restart Timer_MRW for this discard procedure.
- else (if VT(MRW) = MaxMRW):
  - perform the actions specified in subclause 11.6.4a.

11.6.6 Abnormal cases

11.6.6.1 Reception of obsolete/corrupted MRW SUFI by the Receiver

If the received MRW SUFI contains outdated information about the reception window (reception window already moved further than MRW SUFI is indicating), the Receiver shall:

- discard the MRW SUFI;
- set the SN_ACK field in the MRW_ACK SUFI to the current value of VR(R);
- set the N field in the MRW_ACK SUFI to "0000";
- include the MRW_ACK SUFI in the next STATUS PDU/piggybacked STATUS PDU to be transmitted, according to subclause 11.5.2.

11.6.6.2 Void

11.6.6.3 Reception of obsolete/corrupted MRW_ACK SUFI by the Sender

The Sender shall discard the received MRW_ACK SUFI if one of the following cases occurs:

- the timer Timer_MRW is not active; or
- the SN_ACK field in the received MRW_ACK SUFI < the SN_MRW_{LENGTH} field in the transmitted MRW SUFI; or
- the SN_ACK field in the received MRW_ACK SUFI = the SN_MRW_{LENGTH} field in the transmitted MRW SUFI, and the N field in the received MRW_ACK SUFI is not equal to the N_{LENGTH} field in the transmitted MRW SUFI; or
- the SN_ACK field in the received MRW_ACK SUFI > the SN_MRW_{LENGTH} field in the transmitted MRW SUFI, and the N field in the received MRW_ACK SUFI is not equal to "0000".

11.7 Void

11.8 Void

Annex A (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
10/1999	RP-05	RP-99465	-		Approved at TSG-RAN #5 and placed under Change Control	-	3.0.0
12/1999	RP-06	RP-99641	001		RLC: Editorial corrections	3.0.0	3.1.0
	RP-06	RP-99641	002	1	Editorial changes on RLC protocol specification	3.0.0	3.1.0
	RP-06	RP-99643	003	1	MRW procedure	3.0.0	3.1.0
	RP-06	RP-99643	004		SDU Discard Functionality	3.0.0	3.1.0
	RP-06	RP-99643	005	2	Change in RLC control PDU format	3.0.0	3.1.0
	RP-06	RP-99642	006	1	Editorial corrections regarding CTCH	3.0.0	3.1.0
	RP-06	RP-99641	007		Updated RLC SDL	3.0.0	3.1.0
	RP-06	RP-99642	011		RLC Editorial Changes	3.0.0	3.1.0
	RP-06	RP-99642	013		Editorial Modification on RLC specification	3.0.0	3.1.0
	RP-06	RP-99641	014		Editorial changes	3.0.0	3.1.0
	RP-06	RP-99642	015		Change to one PU in a AMD PDU	3.0.0	3.1.0
	RP-06	RP-99643	016	1	Introduction of RLC suspend state	3.0.0	3.1.0
	RP-06	RP-99641	017	1	RLC editorial corrections	3.0.0	3.1.0
01/2000	-	-	-		Editorial corrections in title and Annex A (SDL)	3.1.0	3.1.1
	-	-	-		Correction of persistent error regarding SDL in Table of Contents	3.1.1	3.1.2
03/2000	RP-07	RP-000040	018	1	RLC editorial changes	3.1.2	3.2.0
	RP-07	RP-000040	021	1	Corrections to RLC	3.1.2	3.2.0
	RP-07	RP-000040	025	2	Corrections to RLC	3.1.2	3.2.0
	RP-07	RP-000040	026	1	STATUS PDUs	3.1.2	3.2.0
	RP-07	RP-000040	027	1	Clarification of RLC AMD Model	3.1.2	3.2.0
	RP-07	RP-000040	028		Corrections to Timer_discard procedures	3.1.2	3.2.0
	RP-07	RP-000040	029	1	Segmentation of RLC SDUs	3.1.2	3.2.0
	RP-07	RP-000040	030	2	Modification of SDU discard to support virtual PDCP sequence numbers	3.1.2	3.2.0
	RP-07	RP-000040	031		Removal of SCCH	3.1.2	3.2.0
	RP-07	RP-000040	032		Updated RLC SDL	3.1.2	3.2.0
	RP-07	RP-000040	033	1	RLC Editorial Changes	3.1.2	3.2.0
	RP-07	RP-000040	034		Order of bit transmission for RLC PDUs	3.1.2	3.2.0
06/2000	RP-08	RP-000220	038		Corrections to RLC	3.2.0	3.3.0
	RP-08	RP-000220	039		Correction to the description of the MRW SUFI fields	3.2.0	3.3.0
	RP-08	RP-000220	040	1	Editorial corrections to length indicators and local suspend rate	3.2.0	3.3.0
	RP-08	RP-000220	041	4	Clarification of the RESET PDU	3.2.0	3.3.0
	RP-08	RP-000220	043	1	Clarification of RLC/MAC interaction	3.2.0	3.3.0
	RP-08	RP-000220	044	2	General RLC corrections	3.2.0	3.3.0
	RP-08	RP-000220	045		Clarification of RLC Transparent Mode operation	3.2.0	3.3.0
	RP-08	RP-000220	048		Editorial corrections to abbreviations, SCCH, BCCH	3.2.0	3.3.0
	RP-08	RP-000220	052		Updated RLC SDL	3.2.0	3.3.0
	RP-08	RP-000220	053		Correction to RLC	3.2.0	3.3.0
	RP-08	RP-000220	055		RLC Logical Channel mapping	3.2.0	3.3.0
	RP-08	RP-000220	057		Correction of EPC timer mechanism	3.2.0	3.3.0
09/2000	RP-09	RP-000358	059	1	State variables after window change	3.3.0	3.4.0
	RP-09	RP-000358	060	4	SDU discard	3.3.0	3.4.0
	RP-09	RP-000358	061	5	General RLC corrections	3.3.0	3.4.0
	RP-09	RP-000358	066		Editorial changes to RLC	3.3.0	3.4.0
	RP-09	RP-000358	067	4	Correction to RLC window size range	3.3.0	3.4.0
	RP-09	RP-000358	068	2	Window based polling	3.3.0	3.4.0
	RP-09	RP-000358	070	2	General corrections to RLC	3.3.0	3.4.0
	RP-09	RP-000358	071		State Transition in RLC Acknowledged Mode	3.3.0	3.4.0
	RP-09	RP-000358	073		Clarification of the Length Indicators	3.3.0	3.4.0
	RP-09	RP-000358	076	1	RLC corrections	3.3.0	3.4.0
	RP-09	RP-000358	077	1	Corrections to reset procedure and length indicator definitions	3.3.0	3.4.0
	RP-09	RP-000358	078		RLC Modes for SHCCH	3.3.0	3.4.0
	RP-09	RP-000358	079		CCCH in UM RLC	3.3.0	3.4.0
12/2000	RP-10	RP-000568	080	1	Length Indicator and PDU formats	3.4.0	3.5.0
	RP-10	RP-000568	083	3	Clarification to the Estimated PDU Counter	3.4.0	3.5.0
	RP-10	RP-000568	084	2	Model of UM and AM entities	3.4.0	3.5.0
	RP-10	RP-000568	085	1	General RLC corrections	3.4.0	3.5.0
	RP-10	RP-000568	086	1	General RLC corrections	3.4.0	3.5.0
	RP-10	RP-000568	087	5	RLC timers	3.4.0	3.5.0
	RP-10	RP-000568	088	1	Reset procedure	3.4.0	3.5.0
	RP-10	RP-000568	089	1	Editorial corrections to RLC	3.4.0	3.5.0

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
	RP-10	RP-000568	090	2	RLC UM protocol	3.4.0	3.5.0
	RP-10	RP-000568	092	2	Clarification to window size parameters, MRW SUFI and window based polling	3.4.0	3.5.0
	RP-10	RP-000568	093	3	General RLC Corrections	3.4.0	3.5.0
	RP-10	RP-000568	094	1	RLC Reset handling	3.4.0	3.5.0
	RP-10	RP-000568	095		Inclusion of stage 3 for ciphering	3.4.0	3.5.0
03/2001	RP-11	RP-010026	097	1	Clarification on LIST SUFI and RLIST SUFI	3.5.0	3.6.0
	RP-11	RP-010026	098	1	Corrections and clarifications for SDU discard without explicit signalling	3.5.0	3.6.0
	RP-11	RP-010026	099	1	Tr mode operation	3.5.0	3.6.0
	RP-11	RP-010026	100	1	Timer based discard with explicit signalling	3.5.0	3.6.0
	RP-11	RP-010026	101		Annex updates	3.5.0	3.6.0
	RP-11	RP-010026	103		Clarification on MRW SUFI and SDU discard procedure	3.5.0	3.6.0
	RP-11	RP-010026	104	1	General clarification on SN arithmetic comparison	3.5.0	3.6.0
	RP-11	RP-010026	105	2	General clarification on RLC header and PDU header	3.5.0	3.6.0
	RP-11	RP-010026	106	1	Clarification on the primitives between RLC and higher layers	3.5.0	3.6.0
	RP-11	RP-010026	107	1	Clarification on the model of AM entity	3.5.0	3.6.0
	RP-11	RP-010026	109	2	Clarification on UMD transfer procedure	3.5.0	3.6.0
	RP-11	RP-010026	110	1	RLC status transmission in CELL_PCH and URA_PCH	3.5.0	3.6.0
	RP-11	RP-010026	111		Re-establishment description	3.5.0	3.6.0
	RP-11	RP-010026	112	1	Clarifications on the RESET and RESET ACK PDU sizes	3.5.0	3.6.0
	RP-11	RP-010026	113	1	Editorial corrections and clarifications	3.5.0	3.6.0
	RP-11	RP-010026	114	1	Clarifications on the RLC-AM-DATA-Conf primitive	3.5.0	3.6.0
	RP-11	RP-010026	116		Removal of the payload unit concept	3.5.0	3.6.0
	RP-11	RP-010026	118	2	Padding Blocks and TFC selection pre-empting	3.5.0	3.6.0
	RP-11	-	-		Upgrade to Release 4 - no technical change	3.6.0	4.0.0
06/2001	RP-12	RP-010309	120		Clarification on ACK SUFI	4.0.0	4.1.0
	RP-12	RP-010309	122		MRW SUFI clarification and enhancement	4.0.0	4.1.0
	RP-12	RP-010309	124		Clarification on AM states	4.0.0	4.1.0
	RP-12	RP-010309	126		Clarification on HFN update in RESET procedure	4.0.0	4.1.0
	RP-12	RP-010309	128		Clarification of RLC Discard	4.0.0	4.1.0
	RP-12	RP-010309	130		Removal of reference to RRC	4.0.0	4.1.0
	RP-12	RP-010309	132		Clarification in the LI Parameters section	4.0.0	4.1.0
	RP-12	RP-010309	136		Cleanup of RLC services and functions	4.0.0	4.1.0
	RP-12	RP-010309	138		Clarification on RLC re-establishment	4.0.0	4.1.0
	RP-12	RP-010309	140		Corrections and clarifications to the LIST and RLIST SUFI types	4.0.0	4.1.0
09/2001	RP-13	RP-010542	142		General clarifications	4.1.0	4.2.0
	RP-13	RP-010542	150		Correction to RLC state variables	4.1.0	4.2.0
12/2001	RP-14	RP-010761	152		General clarifications	4.2.0	4.3.0
	RP-14	RP-010761	156		Send state variable for Timer_Poll and window based polling	4.2.0	4.3.0
	RP-14	RP-010761	158		Unexpected data interruption during transmission scheduling	4.2.0	4.3.0
	RP-14	RP-010761	162		UE-ID Type Indicator	4.2.0	4.3.0
	RP-14	RP-010761	164		Removal of obsolete Send MRW option	4.2.0	4.3.0
	RP-14	RP-010771	160		Content of retransmitted RESET ACK PDU	4.2.0	4.3.0
	RP-14	RP-010771	166		Usage of UM RLC Special Length Indicator	4.2.0	4.3.0
	RP-14	RP-010771	170		Indication of SDU transmission result	4.2.0	4.3.0
03/2002	RP-15	RP-020068	172		Clarification on MRW SUFI and SDU discard with explicit signalling procedure	4.3.0	4.4.0
	RP-15	RP-020068	176		SDU discard termination	4.3.0	4.4.0
	RP-15	RP-020068	180		Initial value of VT(US)	4.3.0	4.4.0
06/2002	RP-16	RP-020327	185		Handling abnormal UMD PDUs and AMD PDUs	4.4.0	4.5.0
	RP-16	RP-020327	188		Clarification of the use of Length Indicators	4.4.0	4.5.0
	RP-16	RP-020327	191	1	Correction to MaxDAT, MaxRST and MaxMRW	4.4.0	4.5.0
	RP-16	RP-020327	194		Clarification on polling functions	4.4.0	4.5.0
09/2002	RP-17	RP-020539	197		Correction to the behaviour after expiration of Timer_MRW during the SDU discard with explicit signalling procedure.	4.5.0	4.6.0
	RP-17	RP-020539	200		Corrections to RLC retransmissions	4.5.0	4.6.0
	RP-17	RP-020637	203	1	Corrections to RLC RESET procedure and Length Indicators	4.5.0	4.6.0
	RP-17	RP-020539	206		Corrections on handling of timers during a RLC reset or re-establishment	4.5.0	4.6.0
	RP-17	RP-020551	208		Corrections on indication of SDU transmission result	4.5.0	4.6.0
12/2002	RP-18	RP-020719	211		RB id in ciphering	4.6.0	4.7.0
03/2003	RP-19	RP-030101	215		Correction to VT(MRW) definition	4.7.0	4.8.0
06/2003	RP-20	RP-030292	219	2	Handling of erroneous PDUs	4.8.0	4.9.0
	RP-20	RP-030292	224		Setting of the “Polling bit” in the “Every Poll_SDU SDU” function	4.8.0	4.9.0
	RP-20	RP-030297	221		Receiver behaviour when detecting an AMD PDU duplicate	4.8.0	4.9.0
	RP-20	RP-030297	226		RLC window size reconfigurations	4.8.0	4.9.0

History

Document history		
V4.0.0	March 2001	Publication
V4.1.0	June 2001	Publication
V4.2.0	September 2001	Publication
V4.3.0	December 2001	Publication
V4.4.0	March 2002	Publication
V4.5.0	June 2002	Publication
V4.6.0	September 2002	Publication
V4.7.0	December 2002	Publication
V4.8.0	March 2003	Publication
V4.9.0	June 2003	Publication