TSG-RAN Meeting #26
Athen, Greece, 08-10 December 2004

RP-040490
Agenda item 8.4

Source: TSG-RAN WG2.

Title: Introduction of the MBMS in RAN. CRs to 25.322.

The following CRs are in RP-040490:

Spec	CR	Rev	Phase	Subject	Cat	Version-Current	Version-New	Doc-2nd-Level	Workitem
25.322	265	1	Rel-6	Inclusion of out of sequence SDU delivery	B	6.1.0	6.2.0	R2-042737	MBMS-RAN
25.322	266	-	Rel-6	Addition of MBMS Logical Channels and UM functionality for 'duplicate avoidance and reordering'.	B	6.1.0	6.2.0	R2-042712	MBMS-RAN

3GPP TSG-RAN WG RAN2 #45 ***R2-042737***
Yokohama, Japan, 15th – 19th November 2004

CR-Form-v7

CHANGE REQUEST

⌘ **25.322** **CR** **265** ⌘ **rev** **1** ⌘ Current version: **6.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☒ Radio Access Network ☒ Core Network ☐

Title:	⌘	Inclusion of out of sequence SDU delivery
Source:	⌘	RAN WG2
Work item code:	⌘	MBMS-RAN
Date:	⌘	24th November 2004
Category:	⌘	**B**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*
Rel-7	*(Release 7*

Reason for change:	⌘	To add out of sequence SDU delivery functionality to UM RLC for MCCH.
Summary of change:	⌘	Section 4.2.1.2: Text describing the functionality of 'out of sequence SDUdelivery' is added. Section 5: Out of sequence delivery is added to the function list. Section 6: Out of sequence delivery is added to the services provided to upper layers for unacknowledged data transfer service. Section 8.2: The UM parameter list is modified to include a parameter indicating use/ no use of duplicate avoidance and reordering configuration and the associated parameters OSD_Window_Size and Timer_OSD timeout period. Section 9.4. The state variable VR(UOH) are added. Section 9.5. Timer_OSD is added. Section 9.6. The parameter OSD_Window_Size is added. 11.2.3: The section is subdivided into two new subsections. 11.2.3.1 contains the behaviour for UM PDU reception without 'out of sequence SDU delivery' which was contained in 11.2.3 and 11.2.3.2 contains the behaviour for UM PDU reception when 'out of sequence SDU delivery' is enabled.
Consequences if not approved:	⌘	The standard will not include UM out of sequence SDU delivery.

Clauses affected:	⌘	4.2.1.2, 5, 6, 8.2, 9.4, 9.5, 9.6, 11.2.3, 11.2.3.1 (new), 11.2.3.2 (new).

		Y	N		
Other specs affected:	⌘	X		Other core specifications	⌘
			X	Test specifications	

	X	O&M Specifications

Other comments: ⌘

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

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).

If a receiving UM RLC entity is configured for out of sequence SDU delivery, it will reassemble SDUs and transfer them to the upper layers as soon as all PDUs that contain the SDU have been received even if earlier PDUs have not yet been received. It will store PDUs pending the retransmission of missing PDUs by the transmitting UM RLC. PDUs are removed from storage after recovery of all of its associated SDUs, or by a sequence number window function or a storage timer. Out of sequence SDU delivery is configured only in the UE and is only used with MCCH.

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

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.
- Out of sequence SDU delivery.

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.
- Out of sequence SDU delivery.

- **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.**

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). For a receiving UM RLC in a UE, an additional parameter indicating use/ no use of out of sequence SDU delivery is included (see subclause 11.2.3.2). If out of sequence SDU delivery is used, the parameters OSD_Window_Size (see subclause 9.6) and the timeout value of Timer_OSD (see subclause 9.5) are included.

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.

9.4 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), VR(US) and VR(USOH) 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 according to subclauses 11.4.2 and 11.4.5.1. 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 $VR(H) \leq x < VR(MR)$, 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) ~~Void~~VR(UOH) –UM out of sequence SDU delivery highest received state variable.

This state variable contains the "Sequence Number" of the highest numbered UMD PDU that has been received.

The initial value of this variable is set according to subclause 11.2.3.2.

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 (or restarted) 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 (or restarted) 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) Void

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 except for the generation of a status report triggered by MAC-hs reset. 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 (or restarted) 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 by triggers others than an indication from lower layers following a MAC-hs reset 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. If a status report is triggered by an indication from lower layers following a MAC-hs reset, a status report shall be transmitted immediately and the timer shall be restarted. 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 (or restarted) 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 (or restarted) 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. 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).

j) Timer_OSD

This timer is used with UM out of sequence SDU delivery. It is used to trigger the deleting of stored PDUs.

The timer is started and stopped according to subclause 11.2.3.2.

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.

i) OSD_Window_Size

This protocol parameter indicates the size of the out of sequence SDU delivery storage window.

11.2.3 Reception of UMD PDU

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

- if "out of sequence SDU delivery" is configured:
  - perform the actions specified in subclause 11.2.3.2;
- else:
  - perform the actions specified in subclause 11.2.3.1.

~~- 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 could have segments or "Length Indicators" indicating the end of the SDUs in the missing UMD PDUs according to subclauses 9.2.2.8 and 9.2.2.9.~~

~~- 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.~~

11.2.3.1 SDU discard and re-assembly

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 could have segments or "Length Indicators" indicating the end of the SDUs in the missing UMD PDUs according to subclauses 9.2.2.8 and 9.2.2.9.
- 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.

11.2.3.2 Out of sequence SDU delivery

To enable the recovery of SDUs from UMD PDUs that are received in different transmissions the receiving function shall store PDUs until all SDUs that are associated with the PDU can be reconstructed or until they are discarded in accordance with the procedures described below. SDUs are transferred to the upper layers as soon as all PDUs that

contain the SDU (and any associated PDU containing the special "Length Indicator" "0000 000 or "0000 0000 0000 000") have been received.

Upon delivery of a set of UMD PDUs from the lower layer, the Receiver shall for each PDU (in the following SN denotes the sequence number of each PDU):

- If the PDU is the first PDU received:
  - VR(UOH) shall be assigned the value SN-1.
- if VR(UOH) > SN > VR(UOH) – OSD_Window_Size then:
  - if a PDU with sequence number SN is already stored:
    - discard the PDU;
  - else:
    - store the PDU in sequence number order.
- else:
  - VR(UOH) = SN thereby advancing the storage window;
  - store the PDU in sequence number order;
  - remove from storage any PDUs whose sequence numbers, SN, are outside of the storage window VR(UOH) > SN > VR(UOH) – OSD_Window_Size;
  - if Timer_OSD is active then Timer_OSD shall be stopped;
  - Timer_OSD shall be started.
- if PDU with sequence number SN was stored:
  - taking account of any consecutively numbered stored PDUs (with lower or higher indexes) and using the values of the "Length Indicators", if any, in each PDU:
    - re-assemble the PDUs into SDUs;
    - submit the RLC SDUs to upper layers through the UM-SAP;
    - remove from storage any PDUs for which all associated SDUs have been re-assembled. PDUs containing the special length indicators “0000 000”, “0000 0000 0000 000” or "1111 1111 1111 011"should not be deleted unless SDUs associated with this length indcator have been recovered or will be capable of recovery.

    Note: If PDUs are removed from storage after SDU recovery then retransmitted PDUs may result in the duplicate transfer of SDUs to the higher layers.
- if Timer_OSD expires:
  - remove from storage all stored PDUs.

3GPP TSG-RAN WG RAN2 #45 ***R2-042712***

Yokohama, Japan, 15th – 19th November 2004

CR-Form-v7

CHANGE REQUEST

⌘ **25.322** **CR** **266** ⌘**rev** ⌘ Current version: **6.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☒ Radio Access Network ☒ Core Network ☐

Title:	⌘	Addition of MBMS Logical Channels and UM functionality for 'duplicate avoidance and reordering'.
Source:	⌘	RAN WG2
Work item code:	⌘	MBMS-RAN
Date:	⌘	19th November 2004
Category:	⌘	**B**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 *(GSM Phase 2)*
R96 *(Release 1996)*
R97 *(Release 1997)*
R98 *(Release 1998)*
R99 *(Release 1999)*
Rel-4 *(Release 4)*
Rel-5 *(Release 5)*
Rel-6 *(Release 6)*
Rel-7 *(Release 7)*

Reason for change: ⌘ The purpose of the change is to:

(i) Add the MBMS logical channels MCCH, MTCH and MSCH.

(ii) Extend UM RLC receiver functionality to include a 'duplicate avoidance and reordering' function. The 'duplicate avoidance and reordering' function can be configured only in the UE and is for use with the reception of MBMS MTCH.

Summary of change: ⌘ Section 3.2 Definitions of MBMS, MCCH, MTCH and MSCH are added to the abbreviations.
Section 4.2.1.2 Figure 4.3, is amended to add MCCH, MTCH and MSCH to the lists of logical channels to which UM applies. An additional figure is added to illustrate UM with the duplicate avoidance and reordering function enabled on the receiving side, ciphering/ deciphering removed, multiple inputs to the receiving RLC entity and MTCH as the only logical channel for which this configuration applies.
Section 4.2.1.2.1 MCCH, MTCH and MSCH are added to the list of logical channels that can be transmitted using UM RLC.
Section 4.2.1.2.2. A description of the duplicate avoidance and reordering function and the existence of multiple inputs to the RLC receive function when duplicate avoidance and reordering is used is added.
Section 5. Duplicate avoidance and reordering is added to the function list.
Section 6. Duplicate avoidance and reordering is added to the services provided to upper layers for UM mode.
Section 6.1, MTCH, MCCH and MSCH are added to tables 6.2 and 6.3. In table 6.2, duplicate avoidance and reordering is introduced and associated with receiving MTCH. In table 6.2, out of sequence delivery is introduced and associated with receiving MCCH.

		Section 8.2. The UM parameter list is modified to include a parameter indicating use/ no use of duplicate avoidance and reordering configuration and the associated parameters DAR_Window_Size and Timer_DAR timeout period. Section 9.4. The state variables VR(UDH), VR(UDR) and VR(UDT) are added. Section 9.5. Timer_DAR is added. Section 9.6. The parameter DAR_Window_Size is added. Section 9.7.10. A new section is added to contain a description of duplicate avoidance and reordering. The procedure is based on the MAC-hs re-ordering function that is contained in TS25.321v6.1.0. The initial values specified for state variables are different to those that are use with MAC-hs. Section 11.2.1 MCCH, MSCH and MTCH are added to the lists of logical channels that can be received by UM RLC. Section 11.2.3. The initial sentence is modified to indicate that PDUs can be received from the duplicate avoidance and reordering function.
Consequences if not approved:	⌘	The MBMS logical channels will be absent from the specification. The mechanism for UEs to support MBMS duplicate avoidance and reordering will not be defined.

Clauses affected:	⌘	3.2, 4.2.1.2, 4.2.1.2.1, 4.2.1.2.2, 5, 6, 6.1, 8.2, 9.4, 9.5, 9.6, 9.7.10 (new), 11.2.1, 11.2.3.			
		Y	**N**		
Other specs affected:	⌘	**X**		Other core specifications	⌘ TS25.331 for the signalling of values for the parameters 'DAR_Window_Size' and 'Timer_DAR'.
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

3.2 Abbreviations

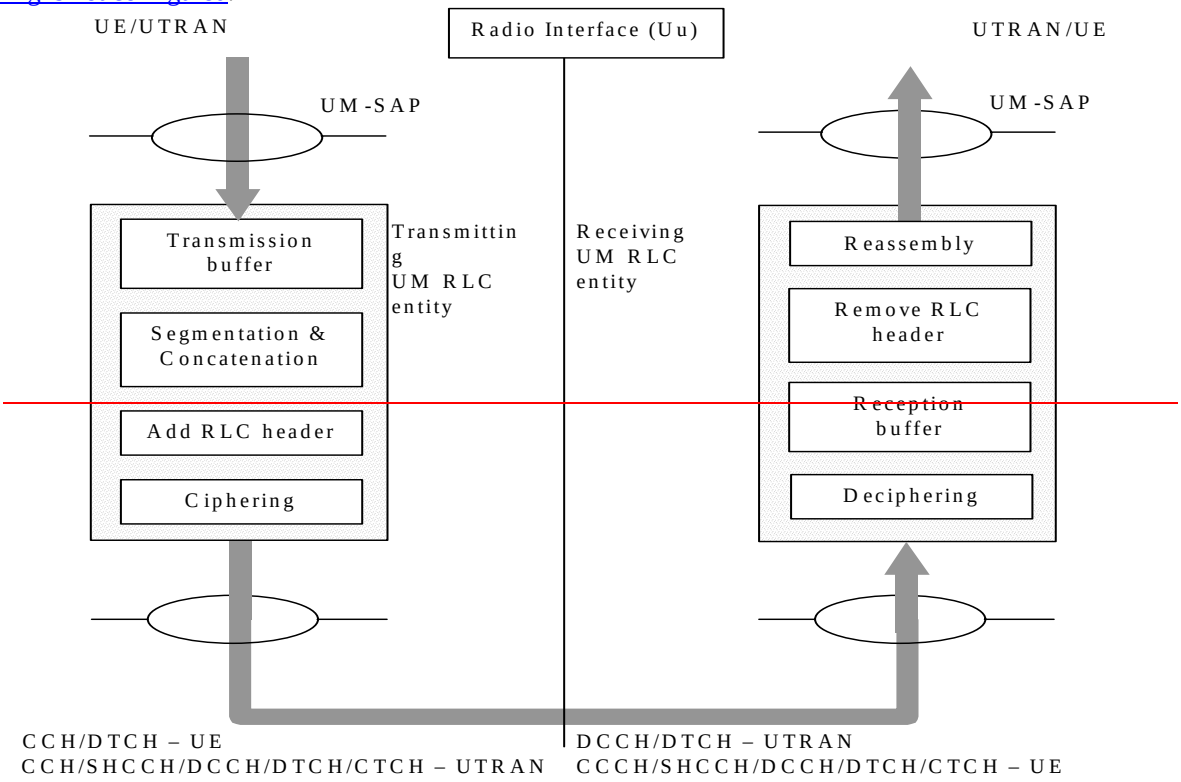
For the purposes of the present document, the following abbreviations apply:

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
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
MBMS	Multmedia Broadcast Multicast Service
MCCH	MBMS point-to-multipoint Control CHannel
MRW	Move Receiving Window
MSB	Most Significant Bit
MSCH	MBMS point-to-multipoint Scheduling CHannel
MTCH	MBMS point-to-multipoint Traffic CHannel
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

4.2.1.2 Unacknowledged mode (UM) RLC entities

Figure 4.3 below shows the model of two unacknowledged mode peer RLC entities when duplicate avoidance and reordering is not configured.



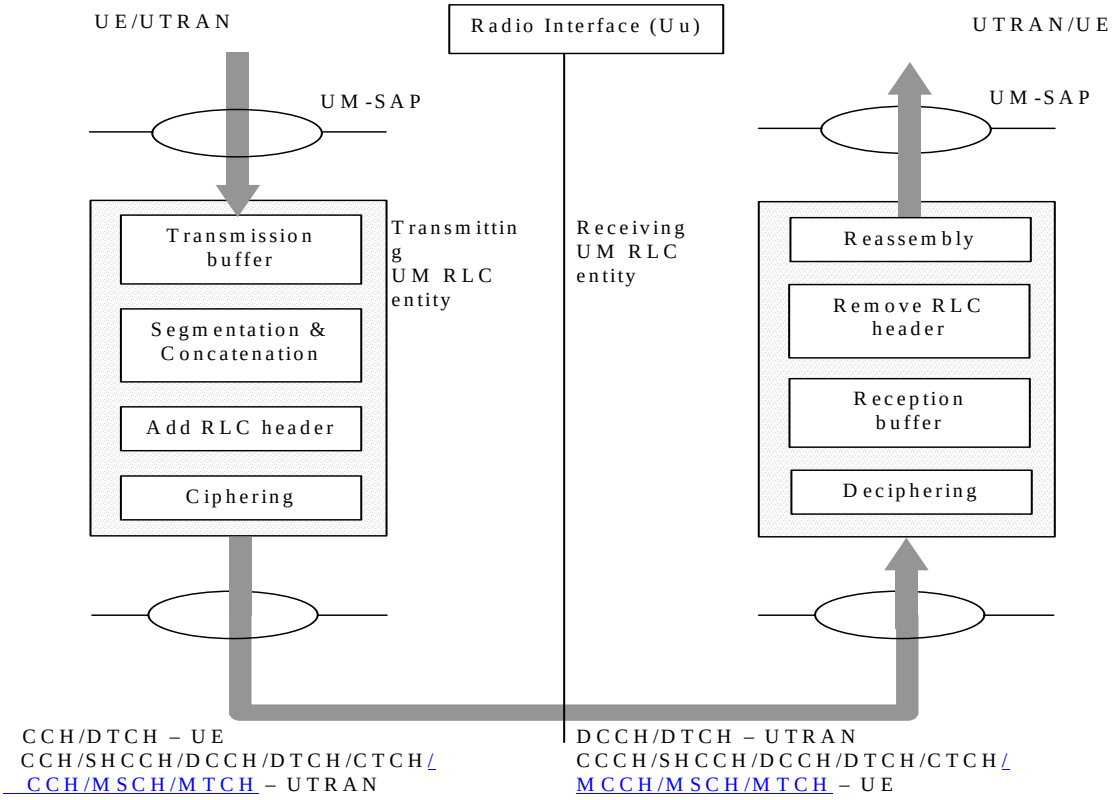


Figure 4.3: Model of two unacknowledged mode peer entities configured for use without duplicate avoidance and reordering

Figure 4.x below shows the model of two unacknowledged mode peer RLC entities configured for duplicate avoidance and reordering. Because duplicate avoidance and reordering is only specified for MTCH in this release, ciphering/ deciphering is omitted.

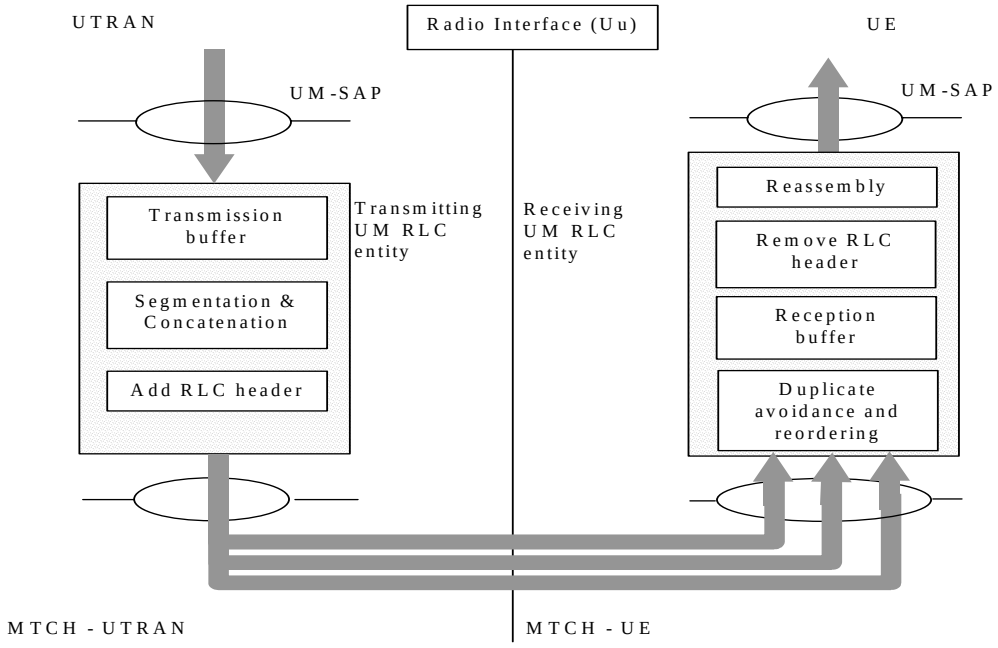


Figure 4.x: Model of two unacknowledged mode peer entities configured for use with duplicate avoidance and reordering

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, DTCH, MCCH, MSCH or an M~~D~~TCH 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. When duplicate avoidance and reordering is configured there may be one or more than one input from the lower layer. Inputs can be added or removed without changing the buffer contents, state variables or timers within the receiving UM RLC entity. Where duplicate avoidance and reordering is not configured there is only one input from the lower layer and it is not reconfigured.

When configured, duplicate avoidance and reordering is the first receive function that is applied to the input UMD PDU streams in the receiving UM RLC entity. It can only be configured in a UE, it is not used in UTRAN. It completes duplicate detection and re-ordering of the UMD PDUs that are received from the one or more inputs to produce a single ordered sequence of PDUs that is passed to the next in sequence RLC receiver function.

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.

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.

- Duplicate avoidance and reordering.

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.
  - Duplicate avoidance and reordering.

- **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	MCCH	MTCH	MSCH
Transparent Service	Applicability	+	+	-	-	+	+	-	-	-	-
	Reassembly	-	-	-	-	+	+	-	-	-	-
	Transfer of user data	+	+	-	-	+	+	-	-	-	-
Unacknowledged Service	Applicability	-	-	+	+	+	+	+	+	+	+
	Reassembly	-	-	+	+	+	+	+	+	+	+
	Deciphering	-	-	-	-	+	+	-	-	-	-
	Sequence number check	-	-	+	+	+	+	+	+	+	+
	Transfer of user data	-	-	+	+	+	+	+	+	+	+
	Duplicate avoidance and reordering	-	-	-	-	-	-	-	-	+	-
	Out of sequence SDU delivery	-	-	-	-	-	-	-	+	-	-
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	MCCH	MTCH	MSCH
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	-	-	+	+

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). For a receiving UM RLC in a UE, an additional parameter indicating use/ no use of duplicate avoidance and reordering is included (see subclause 9.7.10). If duplicate avoidance and

reordering is used, the parameters DAR_Window_Size (see subclause 9.6) and the timeout value of Timer_DAR (see subclause 9.5) are included.

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.

9.4 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), VR(US), VR(UDH) and VR(USDR) 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 according to subclauses 11.4.2 and 11.4.5.1. 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) ~~Void~~VR(UDR) –UM duplicate avoidance and reordering send state variable.

This state variable contains the "Sequence Number" of the next UMD PDU that is expected to be received in sequence. It's value is set according to subclause 9.7.10.

The initial value of this variable is set according to subclause 9.7.10.

f) VR(UDH) – UM duplicate avoidance and reordering highest received state variable.

This state variable contains the "Sequence Number" of the highest numbered UMD PDU that has been received by the duplicate avoidance and reordering function.

The initial value of this variable is set according to 9.7.10.

g) VR(UDT) – UM duplicate avoidance and reordering timer state variable.

This state variable contains the sequence number of the UMD PDU associated with Timer_DAR when the timer is running. It's value is set according to subclause 9.7.10.

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 (or restarted) 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 (or restarted) 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) Void

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 except for the generation of a status report triggered by MAC-hs reset. 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 (or restarted) 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 by triggers others than an indication from lower layers following a MAC-hs reset 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. If a status report is triggered by an indication from lower layers following a MAC-hs reset, a status report shall be transmitted immediately and the timer shall be restarted. 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 (or restarted) 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 (or restarted) 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. 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).

j) Timer_DAR

This timer is used with the UM duplicate avoidance and reordering function. It is used to trigger the transfer of PDUs to the next in sequence UM RLC receiver function.

The timer is started and stopped according to subclause 9.7.10.

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 ≥ 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.

i) DAR_Window_Size

This protocol parameter indicates the size of the duplicate avoidance and reordering receive window.

9.7.10 Duplicate avoidance and reordering for unacknowledged mode

The duplicate avoidance and reordering function can be configured for use within a receiving UM RLC entity in the UE. It combines PDU sequences received from several sources and/or repeat transmissions from a single source to form a single ordered PDU sequence that is passed to the header removal and reassembly functions. It completes duplicate detection, discard and re-ordering based on the UM PDU sequence number. Where the UM RLC receives input from several sources, inputs can be added or removed without changing the buffer contents, state variables and timers associated with the duplicate avoidance and reordering function or any subsequent UM RLC function.

The duplicate avoidance and reordering function makes use of the state variable VR(UDR) and a receive window whose span is from VR(UDH) – DAR_Window_Size+1 to VR(UDH) inclusively. For re-ordering the function uses a buffer for the temporary storage of PDUs.

For each PDU received, the duplicate avoidance and reordering function shall (in the following SN denotes the sequence number of each PDU):

Setting initial values of state variables:

- If the PDU is the first PDU received by the duplicate avoidance and reordering function:
  - VR(UDH) is assigned the value SN;
  - VR(UDR) is assigned the value VR(UDH) - DAR_Window_Size + 1.

Duplicate detection and re-ordering:

- if SN is within the receive window:
  - if SN < VR(UDR) or if a PDU with sequence number SN is already stored in the buffer:
    - the PDU shall be discarded;
  - else:
    - the PDU shall be stored in the buffer.
- if SN is outside of the receive window:
  - the PDU shall be stored in the buffer;
  - VR(UDH) shall be assigned the value SN, thereby advancing the receive window;
  - for any stored PDUs with sequence numbers < VR(UDH) - DAR_Window_Size+1 remove the PDU from the buffer and deliver them to the higher RLC function to perform the actions specified in subclause 11.2.3;
  - if VR(UDR) < VR(UDH) - DAR_Window_Size+1;
    - VR(UDR) shall be assigned the value VR(UDH) - DAR_Window_Size+1.
- if PDU with sequence number VR(UDR) is stored in the buffer:
  - or this PDU and any sequence of stored PDUs with consecutive index numbers starting at VR(UDR)+1, remove the PDUs from the buffer and deliver them to the higher RLC function to perform the actions specified in subclause 11.2.3;
  - VR(UDR) shall be assigned the value of x+1 where x is the sequence number of the highest numbered PDU that was delivered to the higher RLC function.

Timer operation:

- if Timer_DAR is not active when a PDU is stored by the duplicate avoidance and reordering function:
  - Timer_DAR shall be started;
  - VR(UDT) shall be assigned the value of the sequence number of the PDU.
- Timer_DAR shall be stopped:
  - if the PDU with sequence number VR(UDT) is removed from the buffer before Timer_DAR expires.
- if Timer_DAR expires:
  - for all stored PDUs with sequence numbers lower or equal to VR(UDT) and for any sequence of stored PDUs with consecutive sequence numbers starting at VR(UDT)+1, remove the PDUs from the buffer and deliver them to the higher RLC function to perform the actions specified in subclause 11.2.3;
  - VR(UDR) shall be assigned the value x+1 where x is the sequence number of the highest numbered PDU that was delivered to the higher RLC function.
- When Timer_DAR is stopped or expires, and there remain PDUs stored by the duplicate avoidance and reordering function:
  - Timer_DAR shall be started;
  - VR(UDT) shall be assigned the sequence number of the highest numbered stored PDU.

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), MCCH, MSCH, MTCH. The type of logical channel depends on if the RLC entity is located in the user plane (DTCH, CTCH, MTCH) or in the control plane (DCCH/CCCH(downlink only)/SHCCH(downlink only)/MCCH/MSCH). 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

11.2.3 Reception of UMD PDU

Upon delivery of a set of UMD PDUs from the lower layer or from the duplicate avoidance and reordering subentity, 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 could have segments or "Length Indicators" indicating the end of the SDUs in the missing UMD PDUs according to subclauses 9.2.2.8 and 9.2.2.9.
- 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.