TSG-RAN Meeting #10 *RP-000568*
Bangkok, Thailand, 6 - 8 December 2000

Title: **Agreed CRs to TS 25.322**

Source: **TSG-RAN WG2**

Agenda item: **5.2.3**

Doc-1st-	Status-	Spec	CR	Rev	Subject	Cat	Version	Versio
R2-002078	agreed	25.322	080	1	Length Indicator and PDU formats	F	3.4.0	3.5.0
R2-002362	agreed	25.322	083	3	Clarification to the Estimated PDU Counter	F	3.4.0	3.5.0
R2-002407	agreed	25.322	084	2	Model of UM and AM entities	F	3.4.0	3.5.0
R2-002088	agreed	25.322	085	1	General RLC corrections	F	3.4.0	3.5.0
R2-002074	agreed	25.322	086	1	General RLC corrections	F	3.4.0	3.5.0
R2-002361	agreed	25.322	087	5	RLC timers	F	3.4.0	3.5.0
R2-002076	agreed	25.322	088	1	Reset procedure	F	3.4.0	3.5.0
R2-002070	agreed	25.322	089	1	Editorial corrections to RLC	F	3.4.0	3.5.0
R2-002408	agreed	25.322	090	2	RLC UM protocol	F	3.4.0	3.5.0
R2-002400	agreed	25.322	092	2	Clarification to window size parameters, MRW SUFI and window based polling	F	3.4.0	3.5.0
R2-002396	agreed	25.322	093	3	General RLC Corrections	F	3.4.0	3.5.0
R2-002440	agreed	25.322	094	1	RLC Reset handling	F	3.4.0	3.5.0
R2-002266	agreed	25.322	095		Inclusion of stage 3 for ciphering	F	3.4.0	3.5.0

3GPP TSG-RAN WG2 Meeting #16
Beijing, China, 09-13 October 2000

***Document* R2-002078**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 080r1 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X] for information [ ]

strategic [ ] non-strategic [ ] *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-10-4

Subject: Length Indicator and PDU formats

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change:

1. For PDUs with 15-bit Length Indicator, if a PDU contains an LI indicating that an SDU ends with one octet left in the PDU, the last octet of this PDU shall be ignored and shall not be filled with the first octet of the next SDU data, which may not be ready to be transmitted for a long time. Or else, the above mentioned PDU needs to wait to be sent out until the next SDU data is ready.
2. For the above mentioned case, the predefined LI (LI = 11111111 1111011) shall not be included in the next PDU to avoid two LIs indcating the end of one single SDU. Therefore, some clarifications for this predefined LI are needed.
3. The maximum PDU sizes with 15-bit LI should be expressed explicitly.
4. Corrections of Abnormal cases for UMD and AMD transfer procedures are needed.

Clauses affected: 9.2.2.8, 9.2.2.9, 11.2.2.1, 11.2.3, 11.2.4.2, 11.3.2.1, 11.3.2.1.2, 11.3.3, 11.3.4.1, 11.3.4.2, 11.3.4.3.1, 11.3.4.4, 11.3.4.5

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

9.2.2.8 Length Indicator (LI)

The Length Indicator is used to indicate, each time, the end of an SDU occurs in the PU. The Length Indicator points out the number of octets between the end of the last Length Indicator field and up to and including the octet at the end of an SDU segment. Length Indicators are included in the PUs that they refer to. The size of the Length Indicator may be either 7bits or 15bits. The maximum value of a Length Indicator in AM will be no greater than the RLC PDU size – AMD PDU Header – PADDING. The maximum value of a Length Indicator in UM will be no greater than the RLC PDU size – UMD PDU Header – PADDING.

A Length Indicator group is a set of Length Indicators that refer to a PU. Length Indicators that are part of a Length Indicator group must never be reordered within the Length Indicator group or removed from the Length Indicator group.

If there can be more than one Length Indicator, each specifying the end of an SDU in a PU, the order of these Length Indicators must be in the same order as the SDUs that they refer to.

In the case where the end of last segment of an SDU exactly ends at the end of a PDU and there is no LI that indicates the end of the SDU, the next Length Indicator, shall be placed as the first Length Indicator in the following PU and have value LI=0.

In the case where a PDU contains a 15-bit LI indicating that an SDU ends with one octet left in the PDU, the last octet of this PDU shall be ignored and shall not be filled with the first octet of the next SDU data~~., which may or may not be ready to be transmitted yet.~~

In the case where the last segment of an RLC SDU is one octet short of exactly filling the previous RLC PU, and 15-bit Length Indicators are used, the ~~next~~ Length Indicator shall be placed as the first Length Indicator in the following PU and have value LI=111 1111 1111 1011. The remaining one octet in the previous RLC PU shall be ignored.

A PU that has unused space, to be referred to as padding, ~~must~~ shall use a Length Indicator to indicate that this space is used as padding unless the padding size is one octet for PDUs with 15-bit LIs.~~.~~ A padding Length Indicator must be placed after any Length Indicators for a PU.

All unused space in a PU must be located at the end of the PDU, be a homogeneous space and is referred to as padding. Predefined values of the Length Indicator are used to indicate this. 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.

STATUS PDUs can be piggybacked on the AMD PDU by using part or all of the padding space. A Length Indicator must be used to indicate the piggybacked STATUS PDU. This Length Indicator takes space from the padding space or piggybacked STATUS PDU and not the PDU data and will always be the last Length Indicator. Where only part of the padding space is used by a piggybacked STATUS PDU then the end of the piggybacked STATUS PDU is determined 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 PDU. The padding/piggybacked STATUS PDU predefined Length Indicators shall be added after the very last (i.e. there could be more than one SDU that end within a PDU) Length Indicator that indicates the end of the last SDU segment in the PU.

If SDU discard with explicit signalling is used an AMD PDU can contain a maximum number of 15 LIs indicating the end of an SDU and the rest of the AMD PDU space shall be used as padding/piggybacked STATUS PDU.

For AM, 7-bit indicators shall be used if the AMD PDU size is ≤ l26 octets. Otherwise 15-bit indicators shall be used~~. if the AMD PDU size is ≤ 32766.~~ For UM, 7-bit indicators shall be used if the UMD PDU size is ≤ l25 octets. Otherwise 15-bit indicators shall be used~~. if the UMD PDU size is ≤ 32765~~.

The length of the Length Indicator only depends on the size of the largest RLC PDU. The length of the Length Indicator is always the same for all PUs, for one RLC entity.

For Release 99, there is one PU in an AMD PDU.

Length: 7 bits

Bit	Description
0000000	The previous RLC PDU was exactly filled with the last segment of a RLC SDU and there is no LI that indicates the end of the SDU in the previous RLC PDU.
1111100	UMD PDU: The first data octet in this RLC PDU is the first octet of a 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: 15 bits

Bit	Description
000000000000000	The previous RLC PDU was exactly filled with the last segment of an RLC SDU and there is no LI that indicates the end of the 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 LI that indicates the end of the 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 a 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

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

Transparent mode data:

The length of RLC SDUs is not constrained to a multiple of 8 bits.

The RLC SDUs might be segmented. The allowed size for the segments shall be determined from the transport formats of the transport channel [4, 8]. All the RLC PDUs carrying one RLC SDU shall be sent in one transmission time interval. Only segments from one RLC SDU shall be sent in one transmission time interval.

NOTE: If segmentation is not used for the transparent mode RLC entity then more than one RLC SDU can be sent in one transmission time interval using one RLC PDU per RLC SDU. The RLC PDUs need, however, to be of the same size due to L1 limitations.

Unacknowledged mode data and Acknowledged mode data:

The length of RLC SDUs is constrained to a multiple of 8 bits.

RLC SDUs might be segmented. If possible, the last segment of an SDU shall be concatenated with the first segment of the next SDU in order to fill the data field completely and avoid unnecessary padding. The length indicator field is used to point the borders between SDUs.

For PDUs with 15-bit LIs, if an SDU ends with one octet left in a PDU whether the LI indicating the end of the SDU is contained in this PDU or in the next PDU, padding for the last octet of this PDU is necessary and the next SDU shall not be concatenated in this PDU. No LI shall be needed to indicate this kind of one-octet padding.

11.2.2.1 UMD PDU contents to set

The Sequence Number field shall be set equal to VT(US).

The Extension bit shall be set to 1 if the next field is a length indicator field, otherwise it shall be set to zero.

One length indicator field shall be included for each end of a SDU that the PDU includes. The LI ~~length indicator~~ shall be set equal to the number of octets between the end of the header fields and the end of the segment. If padding is needed, another LI ~~length indicator~~ field set to only 1's shall be added unless the padding size is one octet for PDUs with 15-bit LIs. If the PDU is exactly filled with the last segment of a SDU and there is no room for an LI ~~length indicator~~ field, an LI ~~length indicator~~ field set to only 0's shall be included in the next PDU. If the PDU with 15-bit LIs has only one octet left after filling with the last segment of a SDU and there is no room for a 15-bit LI field, an LI field set to the predefined value 11111111 1111011 shall be included in the next PDU.

11.2.3 Reception of UMD PDU

Upon reception of a UMD PDU, the receiver shall update VR(US) state variable according to the received PDU(s).

The PDUs are reassembled into RLC SDUs. If a PDU with sequence number < VR(US) is missing then all SDUs that have segments in this PDU shall be discarded. RLC delivers the RLC SDUs to the higher layer through the UM-SAP.

11.2.4 Abnormal cases

11.2.4.1 Length Indicator value 1111110

Upon reception of an UMD PDU that contains Length Indicator value 1111110 or 111111111111110 ("piggybacked STATUS PDU", in case 7bit or 15 bit Length Indicator field is used, respectively) the receiver shall discard that UMD PDU. This Length Indicator value is not used in unacknowledged mode data transfer.

11.2.4.2 Invalid length indicator value

If the length indicator of a PDU has a value that is larger than the PDU size – the number of octets containing LIs in the PDU – 1 and is not one of the predefined values listed in the table of subclause 9.2.2.8~~-~~, the PDU shall be discarded and treated as a missing PDU.

11.2.4.3 SDU discard without explicit signalling

Upon expiry of the Timer_Discard on the sender side the sender shall discard all PDUs that contain segments of the associated SDU. If the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. The state variable VT(US) shall be updated.

11.3.2.1 AMD PDU contents to set

If the PDU is transmitted for the first time, the Sequence Number field shall be set equal to VT(S) and VT(S) shall be updated.

The setting of the Polling bit is specified in subclause 11.3.2.1.1.

One length indicator field shall be included for each end of a SDU that the PDU includes. The LI ~~length indicator~~ shall be set equal to the number of octets between the end of the header fields and the end of the segment. If the PDU is exactly filled with the last segment of a SDU and there is no room for an LI ~~length indicator~~ field, an LI ~~length indicator~~ field set to only 0's shall be included in the next PDU. If the PDU with 15-bit LIs has only one octet left after filling with the last segment of a SDU and there is no room for a 15-bit LI field, an LI field set to the predefined value 11111111 1111011 shall be included in the next PDU.

How to perform the segmentation of a SDU is specified in subclause 11.3.2.1.2.

11.3.2.1.2 Segmentation of a SDU

Upon reception of a SDU, RLC shall segment the SDU to fit into the fixed size of a PU. The segments are inserted in the data field of a PU. A length indicator shall be added to each PU that includes a border of an SDU, i.e. if a PU does not contain an LI, ~~length indicator~~ the SDU continues in the next PU. The length indicator indicates where the border occurs in the PU. The data after the indicated border can be either a new SDU, padding or piggybacked information. If padding or piggybacking is added another LI ~~length indicator~~ shall be added unless the padding size is one octet for PDUs with 15-bit LIs, see subclauses 9.2.2.8 and 9.2.2.9.

11.3.3 Reception of AMD PDU by the receiver

Upon reception of a AMD PDU, the receiver shall update VR(R), VR(H) and VR(MR) state variables according to the received PU(s).

If any of the PUs includes a Polling bit set to 1, the STATUS PDU transfer procedure shall be initiated.

If the detection of missing PU(s) shall be used and the receiver detects that a PU is missing, the receiver shall initiate the STATUS PDU transfer procedure.

11.3.4 Abnormal cases

11.3.4.1 Timer_Poll timeout

Upon expiry of the Timer_Poll, the sender shall retransmit the poll. The poll can be retransmitted in either a new PDU or a retransmitted PDU.

11.3.4.2 Receiving a PU outside the receiving window

Upon reception of a PU with SN<VR(R) or SN≥VR(MR), the receiver shall discard the PU. The poll bit shall be considered even if a complete PDU is discarded.

11.3.4.3 Timer_Discard timeout

11.3.4.3.1 SDU discard with explicit signalling

Upon expiry of Timer_Discard, the sender shall initiate the SDU discard with explicit signalling procedure.

11.3.4.4 VT(DAT) > MaxDAT

If SDU discard after MaxDAT number of retransmission is used and VT(DAT) > MaxDAT for any PU, the sender shall initiate the SDU discard with explicit signalling procedure for the SDUs to which the PU with VT(DAT)>MaxDAT belongs.

If the SDU discard is not used, the sender shall initiate the RLC reset procedure when VT(DAT) > MaxDAT.

11.3.4.5 Invalid length indicator value

If the length indicator of a PU has a value that is larger than the PU size – the number of octets containing LIs in the PU and is not one of the predefined values listed in the table of subclause 9.2.2.8, the PU shall be discarded and treated as a missing PU.

3GPP TSG-RAN WG2 Meeting #16
Beijing, China, 09-13 October 2000

***Document* R2-002362**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 083r3 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X] for information [ ]

strategic [ ] non-strategic [ ] *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-11-14

Subject: Clarification to the Estimated PDU Counter

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	[X]
A	Corresponds to a correction in an earlier release	[ ]
B	Addition of feature	[ ]
C	Functional modification of feature	[ ]
D	Editorial modification	[ ]

Release:

Phase 2	[ ]
Release 96	[ ]
Release 97	[ ]
Release 98	[ ]
Release 99	[X]
Release 00	[ ]

Reason for change:

1. VR(EP) can be less than ~~zero~~ the number it should be decremented from.
2. The description of the EPC mechanism should be aligned with the names of state variables and timers.
3. ~~The STATUS PDU that triggered the EPC mechanism might never reach the peer sender. The poll timer polling scheme can detect this situation and will send out a poll. The response to this poll will be delayed by the EPC at the receiver. For the cases that the number of requested PUs for retransmission is large, the delay will slow down the RLC traffic. The status report prohibit function of EPC should be deleted to increase the efficiency of RLC traffic control.~~

~~4~~3. Corrections of some minor typing errors are needed.

~~The changes in r2 are marked in yellow.~~

4. The definition “sent on the radio interface” is changed to align with CR87r4.

Clauses affected: 9.4, 9.5, 9.7.2, 9.7.4, 11.4.5.3, ~~11.5.2,~~ 11.5.2.1, 11.5.3, 11.5.4.1

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

9.4 State variables

This sub-clause describes the state variables used in the specification of the peer-to-peer protocol. All state variables are non-negative integers. PUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{12} for AM and 2^7 for UM; the sequence numbers cycle through the entire range: 0 through $2^{12} - 1$ for AM and 0 through $2^7 - 1$ for UM. All arithmetic operations on the following state variables and sequence numbers contained in this specification are affected by the modulus: VT(S), VT(A), VT(MS), VR(R), VR(H), VR(MR), VT(US) and VR(US). When performing arithmetic comparisons of transmitter variables, VT(A) is assumed to be the base. When performing arithmetic comparisons of receiver variables, VR(R) is assumed to be the base.

The RLC maintains the following state variables at the transmitter.

a) VT(S) - Send state variable.

   The sequence number of the next PU to be transmitted for the first time (i.e. excluding retransmission). It is updated after transmission of a PDU, which includes not earlier transmitted PUs and after transmission of a MRW SUFI. The initial value of this variable is 0.

b) VT(A) - Acknowledge state variable.

   The sequence number of the next in-sequence PU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgements. VT(A) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK super-field. The initial value of this variable is 0.

c) VT(DAT).

   This state variable counts the number of times a PU has been transmitted. There is one VT(DAT) for each PU and it is incremented each time the PU is transmitted. The initial value of this variable is 0.

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

   The sequence number of the first PU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1], VT(MS) = VT(A) + Tx_Window_Size. This value represents the upper edge of the transmit window. The transmitter shall not transmit a new PU if VT(S) $\geq$ VT(MS). VT(MS) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK and/or a WINDOW super-field. The PU with SN VT(S)-1 can be transmitted also when VT(S)>VT(MS).

e) VT(US) – UM data state variable.

   This state variable gives the sequence number of the next UMD PDU to be transmitted. It is updated each time a UMD PDU is transmitted. The initial value of this variable is 0.

f) VT(PU).

   This state variable is used when the poll every Poll_PU PU function is used. It is incremented with 1 for each PU that is transmitted. It should be incremented for both new and retransmitted PUs. When it reaches Poll_PU a new poll is transmitted and the state variable is 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 function is used. It is incremented with 1 for each SDU that is transmitted. When it reaches Poll_SDU a new poll is transmitted and the state variable is set to zero. The poll bit should be set in the PU that contains the last segment of the SDU. The initial value of this variable is 0.

h) VT(RST) - Reset state variable.

   It is used to count the number of times a RESET PDU is transmitted. VT(RST) is incremented with 1 each time a RESET PDU is transmitted. VT(RST) is reset only upon the reception of a RESET ACK PDU, i.e. VT(RST) is not reset when a RLC reset occurs which was initiated from the peer RLC entity. The initial value of this variable is 0.

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

It is used to count the number of times a MRW command is transmitted. VT(MRW) is incremented with 1 each time an MRW command is transmitted. VT(MRW) is reset when the discard procedure is terminated. The initial value of this variable is 0.

The RLC maintains the following state variables at the receiver:

a) VR(R) - Receive state variable.

The sequence number of the next in-sequence PU expected to be received. It is set equal to SNmax+1 upon receipt of the next in-sequence PU, where SNmax is the sequence number of the highest received in-sequence PU. The initial value of this variable is 0.

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

The sequence number of the highest expected PU. This state variable is set equal to SN+1 onlywhen a new PU is received with VR(MR)>SN≥VR(H). The initial value of this variable is 0.

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

The sequence number of the first PU not allowed by the receiver [i.e. the receiver will allow up to VR(MR) – 1], VR(MR) = VR(R) + Rx_Window_Size. The receiver shall discard PUs with SN $\geq$ VR(MR).

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

The sequence number of the next PDU to be received. It shall set equal to SN + 1 upon reception of a PDU. The initial value of this variable is 0.

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

The number of PUs that should be received yet as a consequence of the transmission of the latest status report. In acknowledged mode, this state variable is updated at the end of each transmission time interval. It is decremented by the number of PUs that should have been received during the transmission time interval. If VR(EP) is equal to ~~or less than~~ zero, then check if all PUs requested for retransmission in the latest status report have been received.

9.5 Timers

a) Timer_Poll.

This timer is only used when the poll timer trigger is used. It is started when the transmitting side sends a poll to the peer entity. The timer is stopped when receiving a STATUS PDU that contains an acknowledgement of all AMD PDUs with SN up to and including VT(S)-1 at the time the poll was transmitted (or a negative acknowledgement of the same PU). The value of the timer is signalled by RRC.

If the timer expires and no STATUS PDU fulfilling the criteria above has been received, the receiver is polled once more (either by the transmission of a PDU which was not yet sent, or by a retransmission) and the timer is restarted with a new value of VT(S)-1. If there is no PU to be transmitted and all PUs have already been acknowledged, the receiver shall not be polled.

If a new poll is sent when the timer is running it is restarted, with a new value of VT(S)-1.

b) Timer_Poll_Prohibit.

This timer is only used when the poll prohibit function is used. It is used to prohibit transmission of polls within a certain period. A poll shall be delayed until the timer expires if a poll is triggered when the timer is active. Only one poll shall be transmitted when the timer expires even if several polls were triggered when the timer was active. If there is no PU to be transmitted and all PUs have already been acknowledged, a poll shall not be transmitted. This timer will not be stopped by a STATUS PDU. The value of the timer is signalled by RRC.

c) Timer_EPC.

This timer is only used when the EPC function is used and it accounts for the roundtrip delay, i.e. the time when the first retransmitted PU should be received after a status report has been sent. The timer is started when the first STATUS PDU of a status report ~~is begins to be is~~ transmitted and when it expires ~~EPC~~ VR(EP) can start its ~~decrease~~counting-down process (see subclause 9.7.~~4~~3). The value of the timer is signalled by RRC.

9.7.2 STATUS transmission for acknowledged mode

The receiver of AMD PDUs transmits status reports (each status report consists of one or several STATUS PDUs) to the sender in order to inform about which PUs that have been received and not received. There are several triggers for sending a status report. The network (RRC) controls which triggers should be used for each RLC entity, except for one, which is always present. The receiver shall always send a status report when receiving a poll request. Except for that trigger following triggers are configurable:

1) Detection of missing PU(s).

   If the receiver detects one or several missing PUs it shall send a status report to the sender.

2) Timer based STATUS transfer.

   The receiver transmits a status report periodically to the sender. The timer Timer_Status_Periodic controls the time period.

3) The EPC mechanism.

   The timer Timer_EPC is started and the state variable VR(EP) is set when the first STATUS PDU of a status report is transmitted to the peer entity. If not all PUs requested for retransmission have been received before the variable VR(EP) has reached ~~or overpassed~~ zero, ~~Timer EPC has expired~~ a new status report is transmitted to the peer entity. 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. The network (RRC) controls if the STATUS prohibit function shall which functions should be used for each RLC entity. If the prohibit any of the following functions described below is used and Timer_status_Prohibit is active, the sending of the status report shall be delayed, even if any of the conditions above are is fulfilled.:~~

There are two functions that can prohibit the receiver from sending a status report. The network (RRC) controls which functions should be used for each RLC entity. If any of the following functions is used the sending of the status report shall be delayed, even if any of the triggering conditions above are fulfilled:

~~1)~~1) STATUS prohibit.

   The Timer_Status_Prohibit is started when the last STATUS PDU of a status report is transmitted to the peer entity. As long as the timer is running the receiving side is not allowed to send a status report to the peer entity. If a status report was triggered while the timer was running, the status report is transmitted after the timer has expired. The receiver shall only send one status report, even if there are several triggers when the timer running. This timer only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

~~2) The EPC mechanism.~~

~~If the EPC mechanism is active and the sending of a status report is triggered it shall be delayed until the EPC mechanism has ended. The receiver shall only send one status report, even if there are several triggers when the timer is active or the counter is counting down. This mechanism only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.~~

2) The EPC mechanism.

   If the EPC mechanism is active and the sending of a status report is triggered it shall be delayed until the EPC mechanism has ended. The receiver shall only send one status report, even if there are several triggers when the timer is active or the counter is counting down. This mechanism only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

9.7.4 The Estimated PDU Counter

The Estimated PDU Counter is a mechanism used for scheduling the retransmission of status reports in the receiver side. With this mechanism, the receiver will send a new status report in which it requests for PUs 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), ~~the Estimated PDU Counter (EPC),~~ 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 ~~sent out~~transmitted ~~submitted~~ to lower layer (in UTRAN) or ~~sent on the physical layer~~radio interface ~~the~~ successful or unsuccessful transmission of it is indicated by lower layer (in UE) ~~by the receiver to its peer sender~~ to request for retransmitting one or more missing PUs, the variabe VR(EP) is set equal to the number of requested PUs. At least one requested PU is needed to activate the EPC mechanism. The variable VR(EP) ~~EPC~~ is a counter, which is decremented every transmission time interval with the estimated number of PUs that should have been transmitted during that transmission time interval. ~~When the receiver detects that PDUs are missing it generates and sends a status report to the transmitter and sets the EPC equal to the number of requested PUs.~~

A special timer, called Timer_EPC ~~timer~~, controls the maximum time that the variable VR(EP)~~EPC~~ needs to wait before it will start counting down. This timer starts immediately ~~after a transmission of a retransmission request from the receiver (~~after a transmission of a retransmission request from the receiver (when the first STATUS PDU of the status report ~~is~~begins to beis submitted to lower layer ~~transmitted~~ (in UTRAN) or the successful or unsuccessful transmission of it is indicated by lower layer ~~sent on the radio interfacephysical layer~~ (in UE))~~)~~. The timer Timer_EPC ~~timer~~ 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 PUs have been received correctly ~~W~~when VR(EP) ~~the EPC~~ is equal to ~~or less than~~ zero, ~~and not all of these requested PUs have been received correctly,~~ a new status report will be transmitted and the EPC mechanism will be reset accordingly. The timer Timer_EPC ~~timer~~ will be started once more when the first STATUS PDU of the status report is submitted to lower layer ~~transmitted~~ (in UTRAN) or the successful or unsuccessful transmission of it is indicated by lower layer ~~sent on the radio interfacephysical layer~~ (in UE). ~~–~~If all of the requested PUs have been received correctly, the EPC mechanism ends.

11.4.5 Abnormal cases

11.4.5.1 Timer_RST timeout

Upon expiry of Timer_RST the sender shall retransmit the RESET PDU and increase VT(RST) with 1. In the retransmitted RESET PDU the value of the RSN field shall not be incremented.

11.4.5.2 VT(RST) ≥ MaxRST

If VT(RST) becomes larger or equal to MaxRST the RRC layer shall be informed.

11.4.5.3 Reception of the RESET PDU by the sender

Upon reception of a RESET PDU in reset pending state, the sender shall respond with a RESET ACK PDU. The sender resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value. However, VT(RST) and Timer_RST are not reset. The hyper frame number is incremented if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU. The sender shall stay in the reset pending state. The sender shall enter data transfer ready state only upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU.

11.5.2.1 Piggybacked STATUS PDU

It is possible to piggyback a STATUS PDU on an AMD PDU. If a PDU includes padding, a piggybacked STATUS PDU can be inserted instead of the padding. The sending of a piggybacked STATUS PDU follows the same rules as the sending of an ordinary STATUS PDU.

11.5.2.2 STATUS PDU contents to set

The size of the STATUS PDU shall be equal to one of the allowed PDU sizes. The information that needs to be transmitted in a status report can be split into several STATUS PDUs if one STATUS PDU does not accommodate all the information. A SUFI can not be split into several STATUS PDUs. Indication of the same PU shall not be given in more than one STATUS PDU of a STATUS report.

Which SUFI fields to use is implementation dependent, but the status report shall include information about PUs that have been received and information about all PUs detected as missing. No information shall be given for PUs with SN≥VR(H), i.e. PUs that have not yet reached the receiver.

Padding shall be inserted if the SUFI fields do not fill an entire STATUS PDU. If the PDU contains padding the last SUFI field shall be either an ACK SUFI or a NO_MORE SUFI. If there is no padding in the STATUS PDU, NO_MORE SUFI or ACK SUFI does not need to be included in the STATUS PDU.

11.5.3 Reception of the STATUS PDU by the sender

The sender shall upon reception of the STATUS PDU/piggybacked STATUS PDU update the state variables VT(A) and VT(MS) according to the received STATUS PDU/piggybacked STATUS PDU.

If the STATUS PDU includes negative acknowledged PUs, the acknowledged data transfer procedure shall be initiated and the PUs shall be retransmitted. If a PU is indicated as missing more then once in a STATUS PDU, the PU shall be retransmitted only once. Retransmitted PUs have higher priority than new PUs.

11.5.4 Abnormal cases

11.5.4.1 VR(EP)~~EPC~~ reaches ~~or overpasses~~ zero and the requested PUs have not been received

If the EPC mechanism is used and VR(EP) has reached ~~or overpassed~~ ~~0~~ zero and not all PUs 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 PUs have been received and that some new have been lost.

3GPP TSG-RAN WG2 Meeting #17
Sophia Antipolis, France, 13-17 November 2000

***Document* R2-002407**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 084r2 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here* for approval **X** for information strategic non-strategic *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** **16/11/2000**

Subject: Model of UM and AM entities

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change: The current location of ciphering/deciphering in UM and AM entities are not correct. The current models have a problem in using SDU based functions (e.g, SDU discard function) because ciphered PDUs are stored in the transmission and receiver buffers. Moreover, in AM entity, it is difficult to add a piggybacked status PDU into an AMD PDU. Therefore, it is proposed to change the location of ciphering/deciphering in both UM and AM entities.

Revision 2:

- The current RLC UM model does not include the fact that multiple PDU size might be used. Some modifications are needed

Clauses affected: 4.2.1.2, 4.2.1.3

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

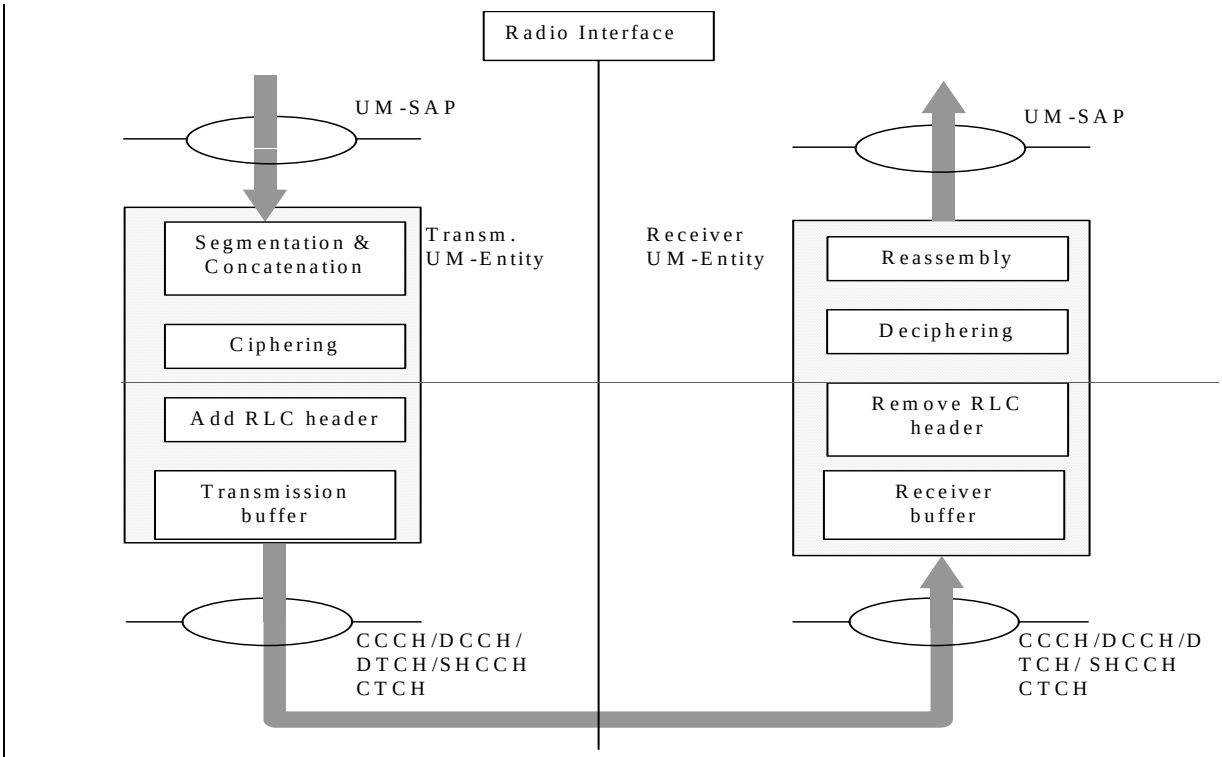
Other comments:

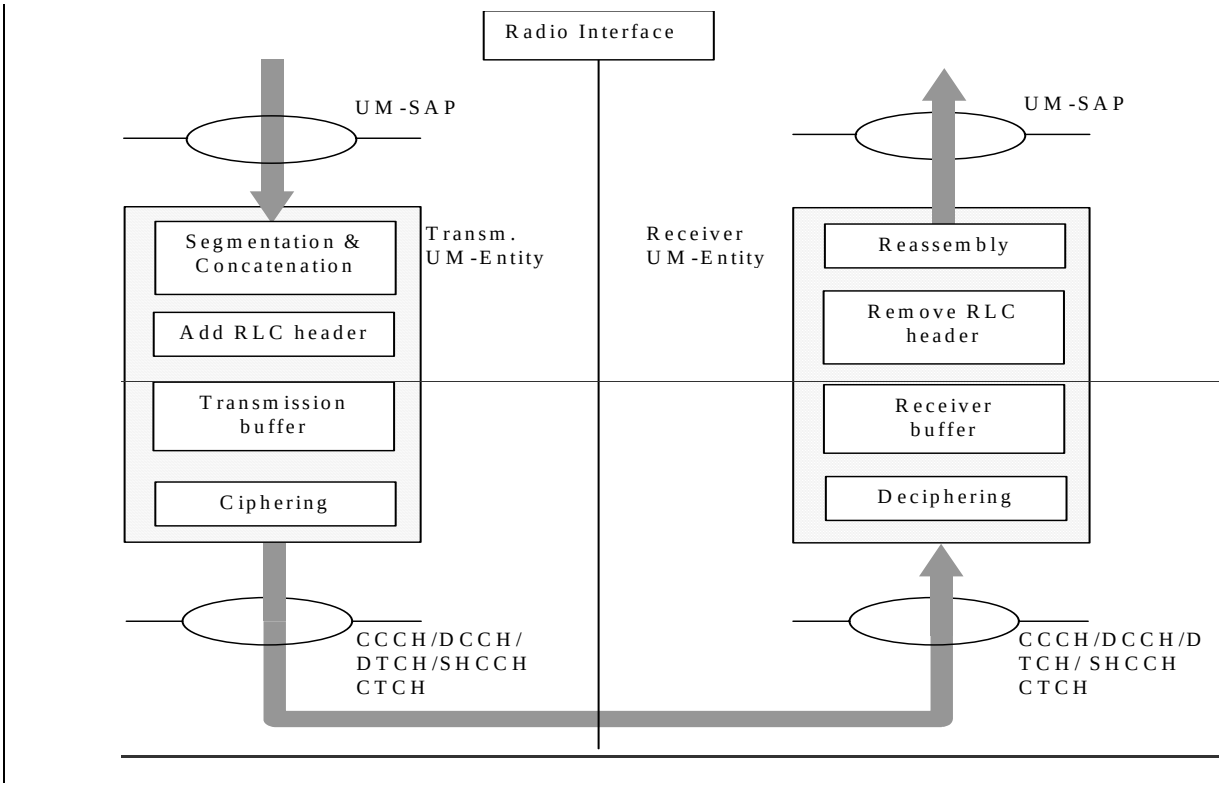

help.doc

<--------- double-click here for help and instructions on how to create a CR.

4.2.1.2 Unacknowledged mode entities

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


Radio Interface
UM-SAP
Segmentation & Concatenation
Transm. UM-Entity
Ciphering
Add RLC header
Transmission buffer
CCCH/DCCH/ DTCH/SHCCH CTCH
Receiver UM-Entity
UM-SAP
Reassembly
Deciphering
Remove RLC header
Receiver buffer
CCCH/DCCH/D TCH/ SHCCH CTCH


Radio Interface
UM-SAP
Segmentation & Concatenation
Transm. UM-Entity
Add RLC header
Transmission buffer
Ciphering
CCCH/DCCH/ DTCH/SHCCH CTCH
Receiver UM-Entity
UM-SAP
Reassembly
Remove RLC header
Receiver buffer
Deciphering
CCCH/DCCH/D TCH/ SHCCH CTCH

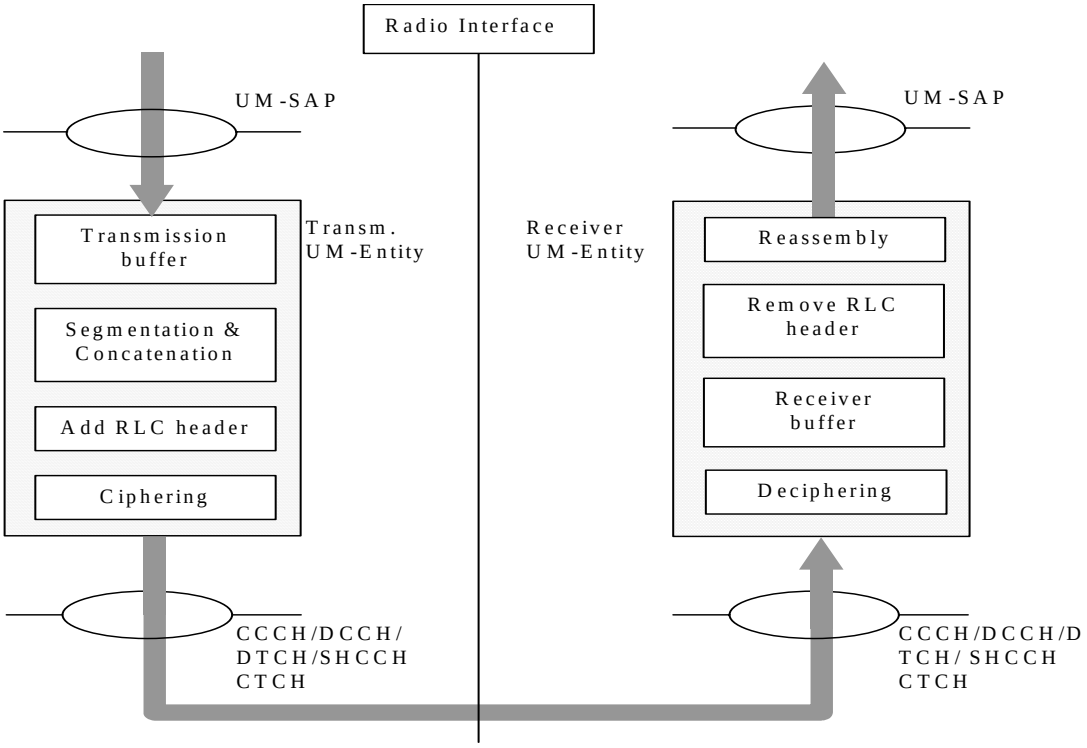


Figure 4.3: Model of two unacknowledged mode peer entities

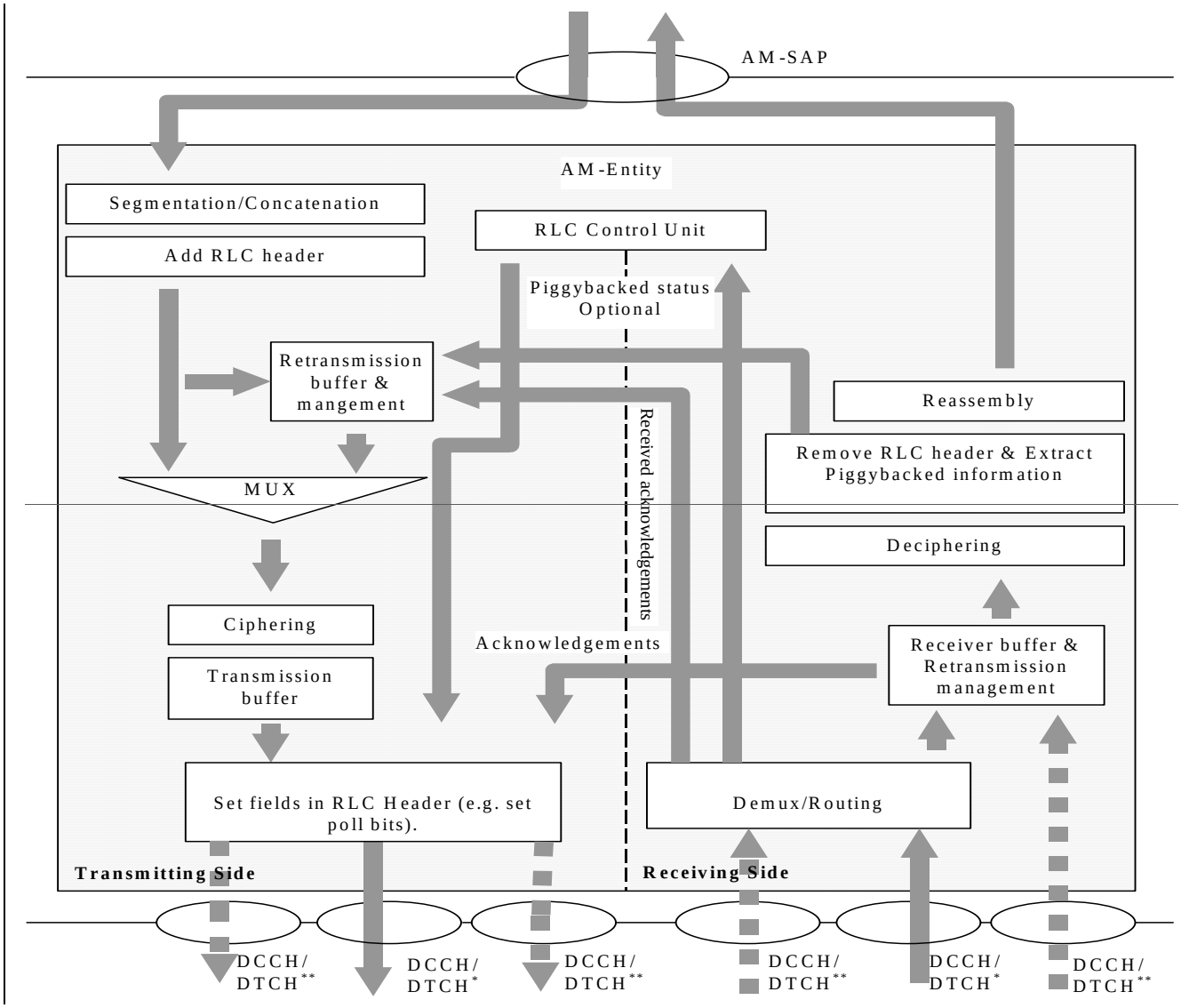
The transmitting UM-entity receives SDUs from the higher layers. RLC might segment the SDUs into RLC PDUs of appropriate size. The SDU might also be concatenated with other SDUs. RLC adds ~~a header and the PDU is placed in the transmission buffer~~. RLC delivers the RLC PDUs to MAC through either a DCCH, CTCH or a DTCH. The CCCH and SHCCH also uses unacknowledged mode, but only for the downlink. Which type of logical channel depends on if the higher layer is located in the control plane (CCCH, DCCH, SHCCH) or user plane (CTCH, DTCH).

The receiving UM-entity receives PDUs through one of the logical channels from the MAC sublayer. RLC removes header from the PDUs and reassembles the PDUs (if segmentation has been performed) into RLC SDUs. The RLC SDUs are delivered to the higher layer.

4.2.1.3 Acknowledged mode entity

Figure 4.4 below shows the model of an acknowledged mode entity, when one logical channel (shown as a solid line) and when two logical channels (shown as dashed lines) are used.

In case two logical channels are used in the uplink the UTRAN can indicate that the first logical channel shall be used for data PDUs and the second logical channel shall be used for control PDUs. If the indication is not given from the UTRAN, data and control PDUs can be sent on either of the two logical channels. The indication of the logical channel mapping is signalled by RRC.


AM-SAP
AM-Entity
Segmentation/Concatenation
Add RLC header
RLC Control Unit
Piggybacked status
Optional
Retransmission
buffer &
mangement
Reassembly
Remove RLC header & Extract
Piggybacked information
Deciphering
MUX
Received acknowledgements
Ciphering
Acknowledgements
Transmission
buffer
Receiver buffer &
Retransmission
management
Set fields in RLC Header (e.g. set
poll bits).
Demux/Routing
Transmitting Side
Receiving Side
DCCH/
DTCH**
DCCH/
DTCH*
DCCH/
DTCH**
DCCH/
DTCH**
DCCH/
DTCH*
DCCH/
DTCH**

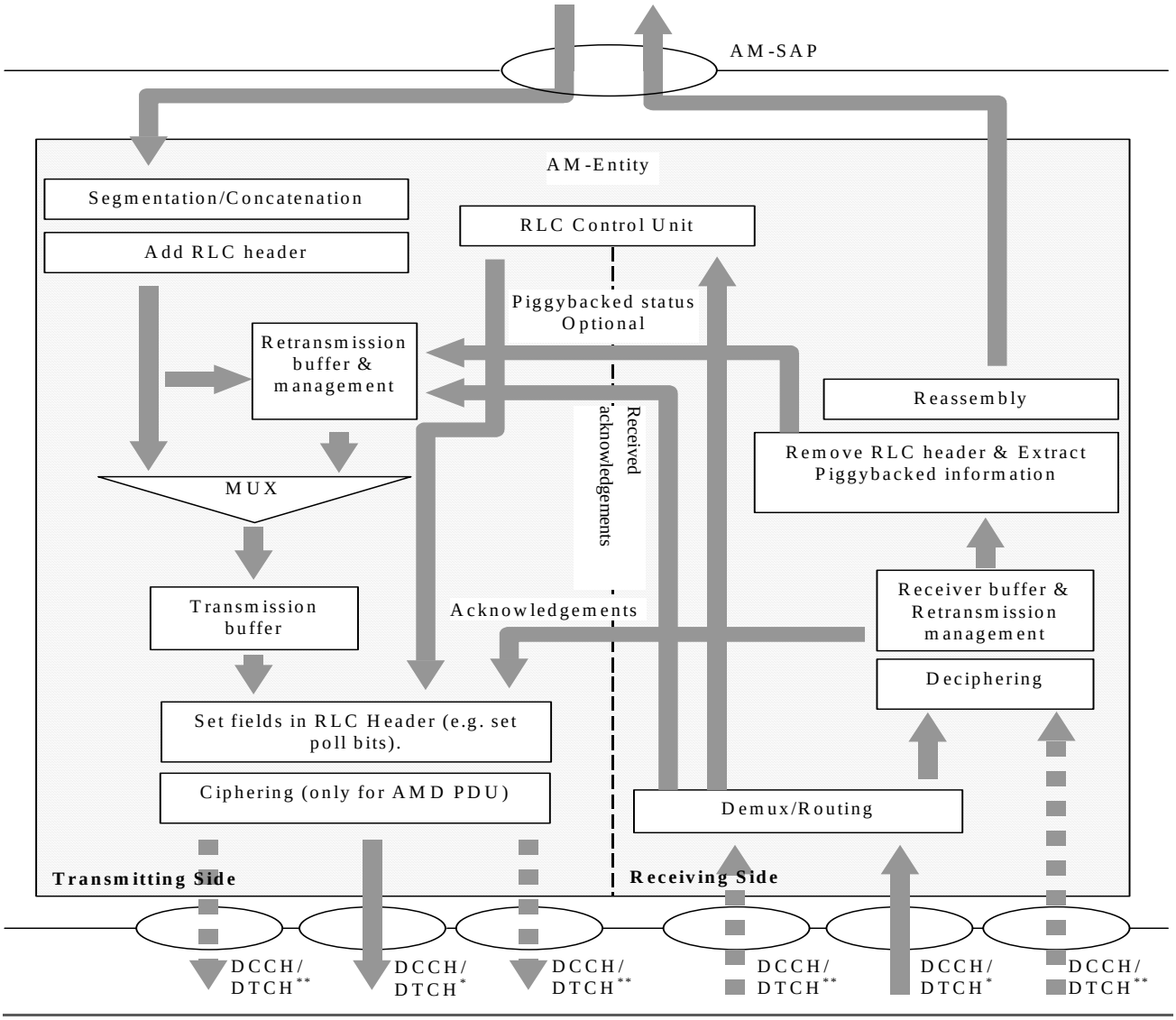


Figure 4.4: Model of an acknowledged mode entity

The transmitting side of the AM-entity receives SDUs from the higher layers. The SDUs are segmented and/or concatenated to PUs of fixed length. PU length is a semi-static value that is decided in bearer setup and can only be changed through bearer reconfiguration by RRC.

For purposes of RLC buffering and retransmission handling, the operation is the same as if there would be one PU per PDU. For concatenation or padding purposes, bits of information on the length and extension are inserted into the beginning of the last PU where data from an SDU is included. Padding can be replaced by piggybacked status information. This includes setting the poll bit.

If several SDUs fit into one PU, they are concatenated and the appropriate length indicators are inserted into the beginning of the PU. After that the PUs are placed in the retransmission buffer and the transmission buffer. One PU is included in one RLC PDU.

The MUX then decides which PDUs and when the PDUs are delivered to MAC. The PDUs are delivered via a function that completes the RLC-PDU header and potentially replaces padding with piggybacked status information.

The ciphering is applied only for AMD PDUs. ~~Includingpiggybacked STATUS PDUs.~~ The fixed 2 octet AMD PDU header is not ciphered. Piggybacked and Padding parts of AMD PDU when existing are ciphered. The other Control PDUs (e.g, STATUS, RESET, and RESET ACK PDU) shall not be ciphered.

When Piggybacking mechanism is applied the padding is replaced by control information, in order to increase the transmission efficiency and making possible a faster message exchange between the peer to peer RLC entities. The

piggybacked control information is not saved in any retransmission buffer. The piggybacked control information is contained in the piggybacked STATUS PDU, which is in turn included into the AMD-PDU. The piggybacked STATUS PDUs will be of variable size in order to match with the amount of free space in the AMD PDU.

The retransmission buffer also receives acknowledgements from the receiving side, which are used to indicate retransmissions of PUs and when to delete a PU from the retransmission buffer.

The Receiving Side of the AM-entity receives PDUs through one of the logical channels from the MAC sublayer. The RLC-PDUs are expanded into separate PUs and potential piggybacked status information are extracted. The PUs are placed in the receiver buffer until a complete SDU has been received. The receiver buffer requests retransmissions of PUs by sending negative acknowledgements to the peer entity. After that the headers are removed from the PDUs and the PDUs are reassembled into a SDU. Finally the SDU is delivered to the higher layer. The receiving side also receives acknowledgements from the peer entity. The acknowledgements are passed to the retransmission buffer on the transmitting side.

3GPP TSG-RAN WG2 Meeting #16
Beijing, China, October 9th – 13th 2000

***Document* R2-002088**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 085r1 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here* — for approval **X** / for information — strategic / non-strategic — *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** Oct. 5th 2000

Subject: General RLC corrections

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X	**Release:**	Phase 2
A	Corresponds to a correction in an earlier release			Release 96
B	Addition of feature			Release 97
C	Functional modification of feature			Release 98
D	Editorial modification			Release 99 X
				Release 00

Reason for change:

1. CRLC-SUSPEND-Req/Conf is missing for UM.
2. A clarification is added for UM Special LI.
3. VT(S) update with MRW SUFI is clarified.
4. Comparison of Tx_Window_Size to 0 is removed from window based polling.
5. In 9.7, the titles of subsections are arranged.
6. The valid range of SN for SUFI LIST, BITMAP or RLIST is corrected.
7. The condition of SDU discard after MaxDAT number of retransmission is corrected.
8. The initiation of SDU discard with explicit signalling procedure is clarified.

Clauses affected: 8.1, 8.2, 9.4, 9.6, 9.7.1, 9.7.3, 9.7.3.1, 9.7.4, 9.7.5, 9.7.6, 10, 11.2.2.1, 11.3.2, 11.3.2.1, 11.3.2.1.1, 11.3.4.4, 11.6.1, 11.6.2, 11.6.5

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:


help.doc

<---------- double-click here for help and instructions on how to create a CR.

8 Elements for layer-to-layer communication

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

8.1 Primitives between RLC and higher 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	Parameter			
	Req.	**Ind.**	**Resp.**	**Conf.**
RLC-AM-DATA	Data, CNF, MUI	Data, DiscardInfo	Not Defined	MUI
RLC-UM-DATA	Data, Use special LI	Data	Not Defined	Not Defined
RLC-TR-DATA	Data	Data	Not Defined	Not Defined
CRLC-CONFIG	E/R, Stop, Ciphering Elements (UM/AM 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 higher layers to request transmission of a higher layer PDU in acknowledged mode.
- RLC-AM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in acknowledged mode and to indicate higher layers of the discarded RLC SDU in the receiving RLC.
- RLC-AM-DATA-Conf is used by RLC to confirm to higher layers the transmission of a RLC SDU.

RLC-UM-DATA-Req/Ind

- RLC-UM-DATA-Req is used by higher layers to request transmission of a higher layer PDU in unacknowledged mode.
- RLC-UM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in unacknowledged mode.

RLC-TR-DATA-Req/Ind

- RLC-TR-DATA-Req is used by higher layers to request transmission of a higher layer PDU in transparent mode.
- RLC-TR-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in transparent mode.

CRLC-CONFIG-Req

This primitive is used by RRC to establish, re-establish, release or reconfigure the RLC. Ciphering elements are included for UM and AM operation.

CRLC-SUSPEND-Req/Conf

This primitive is used by RRC to suspend the RLC. The N parameter indicates that RLC shall not send a PDU with SN>=VT(S)+N for AM and SN>=VT(US)+N for UM, where N is an integer. RLC informs RRC of the VT(S) for AM and VT(US) for UM ~~value~~ in the confirm primitive.

CRLC-RESUME-Req

This primitive is used by RRC to resume RLC when RLC has been suspended.

CRLC-STATUS-Ind

It is used by the RLC to send status information to RRC.

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. The Data parameter may be divided over several RLC PDUs. In case of a RLC-AM-DATA or a RLC-UM-DATA primitive the length of the Data parameter shall be octet-aligned.

2) The parameter Confirmation request (CNF) indicates whether the RLC needs to confirm the correct transmission of the RLC SDU.

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.

4) The parameter E/R indicates (re)establishment, release or modification of RLC If it indicates establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value and RLC shall enter the data transfer ready state. If it indicates re-establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value the RLC buffers shall be emptied and RLC shall enter the data transfer ready state. If it indicates release, all protocol parameters, variables and timers shall be released and RLC shall exit the data transfer ready state. If it indicates modification, the protocol parameters indicated by RRC (e.g. ciphering parameters) shall only be modified with keeping the other protocol parameters, the protocol variables, the protocol timers and the protocol state. RLC shall always be re-established if the PU size is changed.

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

6) The parameter ciphering elements are only applicable for UM and AM operation. These parameters are Ciphering Mode, Ciphering Key, Activation Time (SN to activate a new ciphering configuration) and HFN (Hyper Frame Number).

7) The AM_parameters are only applicable for AM operation. It contains PU size, 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), SDU discard mode (see subclause 9.7.3) and Minimum WSN (see subclause 9.2.2.11.3). The Minimum WSN shall always be greater than or equal to the number of transport blocks in the smallest transport block set.

8) The parameter DiscardInfo indicates the upper layer of each of the discarded RLC SDU. It is applicable only when in-sequence delivery is active and it is purposed to be used when the upper layer requires the reliable data transfer and especially the information of the discarded RLC SDU.

9) The Stop parameter indicates that the RLC entity shall not transmit or receive RLC PDUs.

10) The parameter Use special LI indicates that the LI indicating that a RLC SDU begins in the beginning of a RLC PDU (the first data octet of the PDU is the first octet of an SDU) shall be used. If the RLC SDU does not begin in the beginning of the RLC PDU, or if the LI indicating that a SDU ended exactly or one octet short (only when 15 bit LI is used) in the end of the previous RLC PDU is present, the LI shall not be used.

9.4 State variables

This sub-clause describes the state variables used in the specification of the peer-to-peer protocol. PUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{12} for AM and 2^{7} for UM; the sequence numbers cycle through the entire range: 0 through $2^{12} - 1$ for AM and 0 through $2^{7} - 1$ for UM. All arithmetic operations on the following state variables and sequence numbers contained in this specification are affected by the modulus: VT(S), VT(A), VT(MS), VR(R), VR(H), VR(MR), VT(US) and VR(US). When performing arithmetic comparisons of transmitter variables, VT(A) is assumed to be the base. When performing arithmetic comparisons of receiver variables, VR(R) is assumed to be the base.

The RLC maintains the following state variables at the transmitter.

a) VT(S) - Send state variable.

   The sequence number of the next PU to be transmitted for the first time (i.e. excluding retransmission). It is updated after transmission of a PDU, which includes not earlier transmitted PUs and after transmission of a MRW SUFI which includes $SN\_MRW_{LENGTH} \geq VT(S)$. The initial value of this variable is 0.

b) VT(A) - Acknowledge state variable.

   The sequence number of the next in-sequence PU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgements. VT(A) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK super-field. The initial value of this variable is 0.

c) VT(DAT).

   This state variable counts the number of times a PU has been transmitted. There is one VT(DAT) for each PU and it is incremented each time the PU is transmitted. The initial value of this variable is 0.

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

   The sequence number of the first PU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1], VT(MS) = VT(A) + Tx_Window_Size. This value represents the upper edge of the transmit window. The transmitter shall not transmit a new PU if VT(S) $\geq$ VT(MS). VT(MS) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK and/or a WINDOW super-field. The PU with SN VT(S)-1 can be transmitted also when VT(S) $\geq$ VT(MS).

e) VT(US) – UM data state variable.

   This state variable gives the sequence number of the next UMD PDU to be transmitted. It is updated each time a UMD PDU is transmitted. The initial value of this variable is 0.

f) VT(PU).

   This state variable is used when the poll every Poll_PU PU function is used. It is incremented with 1 for each PU that is transmitted. It should be incremented for both new and retransmitted PUs. When it reaches Poll_PU a new poll is transmitted and the state variable is 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 function is used. It is incremented with 1 for each SDU that is transmitted. When it reaches Poll_SDU a new poll is transmitted and the state variable is set to zero. The poll bit should be set in the PU that contains the last segment of the SDU. The initial value of this variable is 0.

h) VT(RST) - Reset state variable.

   It is used to count the number of times a RESET PDU is transmitted. VT(RST) is incremented with 1 each time a RESET PDU is transmitted. VT(RST) is reset only upon the reception of a RESET ACK PDU, i.e. VT(RST) is not reset when a RLC reset occurs which was initiated from the peer RLC entity. The initial value of this variable is 0.

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

It is used to count the number of times a MRW command is transmitted. VT(MRW) is incremented with 1 each time an MRW command is transmitted. VT(MRW) is reset when the discard procedure is terminated. The initial value of this variable is 0.

The RLC maintains the following state variables at the receiver:

a) VR(R) - Receive state variable.

The sequence number of the next in-sequence PU expected to be received. It is set equal to SNmax+1 upon receipt of the next in-sequence PU, where SNmax is the sequence number of the highest received in-sequence PU. The initial value of this variable is 0.

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

The sequence number of the highest expected PU. This state variable is set equal to SN+1 only when a new PU is received with VR(MR)>SN≥VR(H). The initial value of this variable is 0.

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

The sequence number of the first PU not allowed by the receiver [i.e. the receiver will allow up to VR(MR) – 1], VR(MR) = VR(R) + Rx_Window_Size. The receiver shall discard PUs with SN ≥ VR(MR).

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

The sequence number of the next PDU to be received. It shall set equal to SN + 1 upon reception of a PDU. The initial value of this variable is 0.

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

The number of PUs that should be received yet as a consequence of the transmission of the latest status report. In acknowledged mode, this state variable is updated at the end of each transmission time interval. It is decremented by the number of PUs that should have been received during the transmission time interval. If VR(EP) is equal to zero, then check if all PUs requested for retransmission in the latest status report have been received.

9.6 Protocol Parameters

The values of the protocol parameters in this section are signalled by RRC.

a) MaxDAT.

It is the maximum value for the number of retransmissions of a PU. This parameter is an upper limit of counter VT(DAT). When the value of VT(DAT) comes to MaxDAT, error recovery procedure will be performed.

b) Poll_PU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_PU PU. This is an upper limit for the VT(PU) state variable, when VT(PU) reaches Poll_PU a poll is transmitted to the peer entity.

c) Poll_SDU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_SDU SDU. This is an upper limit for the VT(SDU) state variable, when VT(SDU) reaches Poll_SDU a poll is transmitted to the peer entity.

d) Poll_Window.

This parameter indicates when the transmitter should poll the receiver in case of performing window-based polling. A poll is triggered for each PU when:

1) VT(S)<VT(MS), ~~Tx_Window_Size>0,~~ and

$$\left[1 - \frac{(Tx\_Window\_Size + \mathrm{VT(MS)} - \mathrm{VT(S)\text{-}1}) \bmod Tx\_Window\_Size}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$

.

2) VT(S)≥VT(MS)~~, and Tx_Window_Size>0~~.

~~When Tx_Window_Size=0, the transmitter does not perform window-based polling.~~

e) MaxRST.

It is the maximum value for the number of retransmission of RESET PDU. This parameter is an upper limit of counter VT(RST). When the value of VT(RST) comes to MaxRST, the higher layer (RRC) is notified.

f) Tx_Window_Size.

The maximum allowed transmitter window size.

g) Rx_Window_Size.

The maximum allowed receiver window size.

h) MaxMRW.

It is the maximum value for the number of retransmissions of a MRW command. This parameter is an upper limit of counter VT(MRW). When the value of VT(MRW) comes to MaxMRW, error recovery procedure will be performed.

9.7 Specific functions

9.7.1 Polling function for acknowledged mode ~~transfer~~

The transmitter of AMD PDUs may poll the receiver for a status report (consisting of one or several STATUS PDUs). The Polling bit in the AMD PDU indicates the poll request. There are several triggers for setting the polling bit. The network (RRC) controls, which triggers should be used for each RLC entity. Following triggers are possible:

1) Last PU in buffer.

The sender transmits a poll when the last PU available for transmission is transmitted.

2) Last PU in retransmission buffer.

The sender transmits a poll when the last PU to be retransmitted is transmitted.

3) Poll timer.

The timer Timer_Poll is started when a poll is transmitted to the receiver and if the criterion for stopping the timer has not occurred before the timer Timer_Poll expires a new poll is transmitted to the receiver.

4) Every Poll_PU PU.

The sender polls the receiver every Poll_PU PU. Both retransmitted and new PU~~u~~s shall be counted.

5) Every Poll_SDU SDU.

The sender polls the receiver every Poll_SDU SDU.

6) Poll_Window% of transmission window.

The sender polls the receiver when it has reached Poll_Window% of the transmission window.

7) Timer based.

The sender polls the receiver periodically.

Either the trigger "Last PU in buffer" and "Last PU in retransmission buffer" or "Timer based" can be chosen to avoid deadlock for every RLC entity. The network also controls if the poll prohibit function shall be used. The poll bit shall be set to 0 if the poll prohibit function is used and the timer Timer_Poll_Prohibit is active. This function has higher priority than any of the above mentioned triggers.

9.7.2 STATUS transmission for acknowledged mode

The receiver of AMD PDUs transmits status reports (each status report consists of one or several STATUS PDUs) to the sender in order to inform about which PUs that have been received and not received. There are several triggers for sending a status report. The network (RRC) controls which triggers should be used for each RLC entity, except for one, which is always present. The receiver shall always send a status report when receiving a poll request. Except for that trigger following triggers are configurable:

1) Detection of missing PU(s).

   If the receiver detects one or several missing PUs it shall send a status report to the sender.

2) Timer based STATUS transfer.

   The receiver transmits a status report periodically to the sender. The timer Timer_Status_Periodic controls the time period.

3) The EPC mechanism.

   The Timer EPC is started when the first STATUS PDU of a status report is transmitted to the peer entity. If not all PUs requested for retransmission have been received before the Timer EPC has expired a new status report is transmitted to the peer entity. 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. The network (RRC) controls which functions should be used for each RLC entity. If any of the following functions is used the sending of the status report shall be delayed, even if any of the conditions above are fulfilled:

1) STATUS prohibit.

   The Timer_Status_Prohibit is started when the last STATUS PDU of a status report is transmitted to the peer entity. As long as the timer is running the receiving side is not allowed to send a status report to the peer entity. If a status report was triggered while the timer was running, the status report is transmitted after the timer has expired. The receiver shall only send one status report, even if there are several triggers when the timer running. This timer only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

2) The EPC mechanism.

   If the EPC mechanism is active and the sending of a status report is triggered it shall be delayed until the EPC mechanism has ended. The receiver shall only send one status report, even if there are several triggers when the timer is active or the counter is counting down. This mechanism only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

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

The SDU discard function allows to discharge RLC PDU from the buffer on the transmitter side, when the transmission of the RLC PDU does not success for a long time. The SDU discard function allows to avoid buffer overflow. There will be several alternative operation modes of the RLC SDU discard function, and which discard function to use will be given by the QoS requirements of the Radio Access Bearer.

The following is a list of operation modes for the RLC SDU discard function.

Table 9.2: List of criteria's 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 retransmissions	Network controlled

9.7.3.1 Timer based discard, with explicit signalling

This alternative 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 a higher layer, timer monitoring of the transmission time of the SDU is started. If the transmission time exceeds a predefined value for a SDU in acknowledged mode RLC, this SDU is discarded in the transmitter and a Move Receiving Window (MRW) command is sent to the receiver so that AMD PDUs carrying that SDU are discarded in the receiver and the receiver window is updated accordingly. Note that when the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded.

The MRW command is defined as a super-field in the RLC STATUS PDU (see subclause 9.2), and piggy-backed to status information of transmissions in the opposite direction. If the MRW command has not been acknowledged by receiver, it will be retransmitted. Therefore, SDU discard variants requiring peer-to-peer signalling are only possible for full duplex connections.

9.7.3.2 Timer based discard, without explicit signalling

This alternative 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. This function is applied only for unacknowledged and transparent mode RLC and peer-to-peer signalling is never needed. The SDUs are simply discarded in the transmitter, once the transmission time is exceeded.

9.7.3.3 SDU discard after MaxDAT number of retransmissions

This alternative uses the number of retransmissions as a trigger for SDU discard, and is therefore only applicable for acknowledged mode RLC. This makes the SDU discard function dependent of 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. SDU discard is triggered at the transmitter, and a MRW command is necessary to convey the discard information to the receiver, like in the timer based discard with explicit signalling.

9.7.4 The Estimated PDU Counter for acknowledged mode

The Estimated PDU Counter is a mechanism used for scheduling the retransmission of status reports in the receiver side. With this mechanism, the receiver will send a new status report in which it requests for PUs not yet received. The time between two subsequent status report retransmissions is not fixed, but it is controlled by the Estimated PDU Counter (EPC), which adapt this time to the current bit rate, indicated in the TFI, in order to minimise the delay of the status report retransmission.

The EPC is a counter, which is decremented every transmission time interval with the estimated number of PUs that should have been transmitted during that transmission time interval. When the receiver detects that PDUs are missing it generates and sends a status report to the transmitter and sets the EPC equal to the number of requested PUs.

A special timer, called EPC timer, controls the maximum time that the EPC 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 transmitted). The EPC timer 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.

When the EPC is equal to zero and not all of these requested PUs have been received correctly, a new status report will be transmitted and the EPC will be reset accordingly. The EPC timer will be started once more.

9.7.5 Multiple payload units in an RLC PDU for acknowledged mode

The possibility to include multiple payload units (PU) into one RLC AMD PDU is part of the service capabilities of a UE in acknowledged mode. For Release 99, there shall be only one PU per AMD PDU.

A payload unit is the smallest unit that can be separately addressed for retransmission and is of fixed size, containing data and optionally, length indicators and/or padding. The padding space of a PU can be used to piggyback STATUS PDUs.

The size of the PU is set by the RRC.

9.7.6 Local Suspend function for acknowledged and unacknowledged mode ~~transfer~~

The higher layer (RRC) may suspend the RLC entity. The CRLC-SUSPEND-Req indicates this request. The RLC entity shall, when receiving this request, not send RLC PDUs with SN>=VT(S)+N for AM and SN>=VT(US)+N for UM (N is given by the CRLC_SUSPEND-Req primitive). The RLC entity shall acknowledge the CRLC-SUSPEND-Req ordering a suspend with a CRLC-SUSPEND-Conf with the current value of VT(S) for AM and VT(US) for UM. The suspend state is left when a CRLC-RESUME-Req primitive indicating resume is received.

10 Handling of unknown, unforeseen and erroneous protocol data

The list of error cases is reported below:

a) Inconsistent state variables.

   If the RLC entity receives a PDU including "erroneous Sequence Number", state variables between peer entities may be inconsistent. Following shows "erroneous Sequence Number" examples:

   - Each Sequence Number of missing PU informed by SUFI LIST, BITMAP or RLIST is not within the value between "Acknowledge state variable(VT(A))" and "Send state variable(VT(S)) – 1", and

   -- LSN of SUFI ACK is not within the value between "Acknowledge state variable(VT(A))" and "Send state variable(VT(S))".

   In case of error situations the following actions are foreseen:

   1) RLC entity should use RESET procedure in case of an unrecoverable error.

   2) RLC entity should discard invalid PDU.

   3) RLC entity should notify upper layer of unrecoverable error occurrence in case of failed retransmission.

b) Inconsistent status indication of a PU

If a received STATUS PDU indicates different status for the same PU, then the transmitter shall discard the STATUS PDU.

11.2 Unacknowledged mode data transfer procedure

11.2.1 Purpose

The unacknowledged mode data transfer procedure is used for transferring data between two RLC peer entities, which are operating in unacknowledged mode. Figure 11.2 below illustrates the elementary procedure for unacknowledged mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.

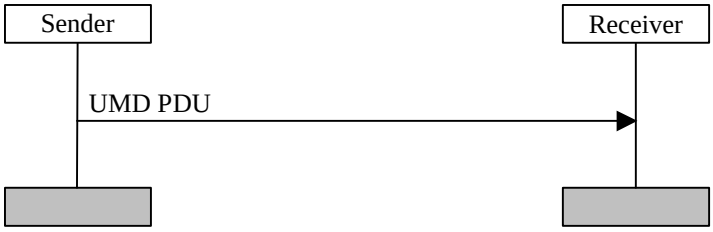


Figure 11.2: Unacknowledged mode data transfer procedure

11.2.2 Initiation

The sender initiates this procedure upon a request of unacknowledged mode data transfer from higher layer.

When the sender is in data transfer ready state it shall segment the data received from the higher layer into PDUs.

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) and MAC decides how many PDUs shall be transmitted in each TTI. In the UE, the PDUs that can not be transmitted in a TTI (i.e. MAC has indicated that some of the available PDUs can not be transmitted) shall be buffered according to the discard configuration set by RRC.

The VT(US) state variable shall be updated for each UMD PDU that is transmitted.

11.2.2.1 UMD PDU contents to set

The Sequence Number field shall be set equal to VT(US).

The Extension bit shall be set to 1 if the next field is a length indicator field, otherwise it shall be set to zero.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number of octets between the end of the header fields and the end of the segment. If padding is needed another length indicator shall be added. If the PDU is exactly filled with the last segment of a SDU and there is no room for a length indicator field, a length indicator field set to only 0's shall be included as the first length indicator in the ~~next~~following PDU.

11.2.3 Reception of UMD PDU

Upon reception of a UMD PDU the receiver shall update VR(US) state variable according to the received PDU(s).

The PDUs are reassembled into RLC SDUs. If a PDU with sequence number < VR(US) is missing then all SDUs that have segments in this PDU shall be discarded. RLC delivers the RLC SDUs to the higher layer through the UM-SAP.

11.2.4 Abnormal cases

11.2.4.1 Length Indicator value 1111110

Upon reception of an UMD PDU that contains Length Indicator value 1111110 or 111111111111110 ("piggybacked STATUS PDU", in case 7bit or 15 bit Length Indicator field is used, respectively) the receiver shall discard that UMD PDU. This Length Indicator value is not used in unacknowledged mode data transfer.

11.2.4.2 Invalid length indicator value

If the length indicator of a PDU has a value that is larger than the PDU size, the PDU shall be discarded and treated as a missing PDU.

11.2.4.3 SDU discard without explicit signalling

Upon expiry of the Timer_Discard on the sender side the sender shall discard all PDUs that contain segments of the associated SDU. If the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. The state variable VT(US) shall be updated.

11.3 Acknowledged mode data transfer procedure

11.3.1 Purpose

The acknowledged mode data transfer procedure is used for transferring of data between two RLC peer entities, which are operating in acknowledged mode. Figure 11.3 below illustrates the elementary procedure for acknowledged mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.

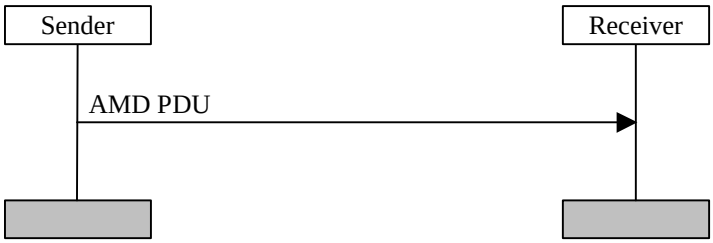


Figure 11.3: Acknowledged mode data transfer procedure

11.3.2 Initiation

The sender initiates this procedure upon a request of acknowledged mode data transfer from higher layer or upon retransmission of PUs. Retransmitted PUs have higher priority than PUs transmitted for the first time.

The sender is only allowed to retransmit PUs that have been indicated missing by the receiver. An exception is the PU with SN VT(S)-1 which can be retransmitted. In addition, a PU that has not yet been acknowledged, may be retransmitted if the configured transmitter window size is less than 2048.

RLC shall segment the data received from the higher layer into PUs. When the sender is in data transfer ready state one or several PUs are included in one AMD PDU, which is sent to the receiver. The 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. In the UE, the PDUs that can not be transmitted in a TTI (i.e. MAC has indicated that some of the available PDUs can not be transmitted) shall be buffered according to the discard configuration set by RRC.

The VT(DAT) state variables shall be updated for each AMD PDU that is transmitted. The PDU shall not include any PU with Sequence Number ≥ VT(MS), except the PU with sequence number VT(S)-1 which may be included also when VT(S) ≥ VT(MS).

If the poll bit is set in any of the AMD PDUs and the timer Timer_Poll shall be used the sender shall start the timer Timer_Poll when the PDU with the set poll bit is delivered to MAC.

If timer based SDU discard is used the timer Timer_Discard shall be started when the RLC entity receives an SDU from higher layer. One timer is used for each SDU that is received from higher layer.

If the trigger for polling, "Every Poll_PU PU", is used the VT(PU) shall be increased by 1 for each PU that is transmitted.

If the trigger for polling, "Every Poll_SDU SDU", is used the VT(SDU) shall be increased by 1 for each SDU that is transmitted.

11.3.2.1 AMD PDU contents to set

If the PDU is transmitted for the first time, the Sequence Number field shall be set equal to VT(S) and VT(S) shall be updated.

The setting of the Polling bit is specified in subclause 11.3.2.1.1.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number of octets between the end of the header fields and the end of the segment. If the PDU is exactly filled with the last segment of a SDU and there is no room for a length indicator field, a length indicator field set to only 0's shall be included as the first length indicator in the ~~next~~following PDU. How to perform the segmentation of a SDU is specified in subclause 11.3.2.1.2.

11.3.2.1.1 Setting of the Polling bit

The Polling bit shall be set to 1 if any of following conditions are fulfilled except when the poll prohibit function is used and the timer Timer_Poll_Prohibit is active (the different triggers are described in 9.7.4):

1) Last PU in buffer is used and the last PU available for transmission is transmitted.
2) Last PU in retransmission buffer is used and the last PU to be retransmitted is transmitted.
3) Poll timer is used and timer Timer_Poll has expired.
4) Every Poll_PU PU is used and when VT(PU)=Poll_PU.
5) Every Poll_SDU is used and VT(SDU)=Poll_SDU and the PDU contains the last segment that SDU.
6) Poll_Window(%) of transmission window is used, ~~Tx_Window_Size>0,~~ VT(S)<VT(MS), and

$$\left[1 - \frac{(Tx\_Window\_Size + \mathrm{VT(MS)} - \mathrm{VT(S)}\text{-}1) \bmod Tx\_Window\_Size}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$

7) Poll_Window (%) of transmission window is used, ~~Tx_Window_Size>0~~ and VT(S)≥VT(MS).
8) Timer based polling is used and Timer_Poll_Periodic has expired.
9) Poll prohibit shall be used, the timer Timer_Poll_Prohibit has expired and one or several polls were prohibited during the time Timer_Poll_Prohibit was active.

11.3.2.1.2 Segmentation of a SDU

Upon reception of a SDU, RLC shall segment the SDU to fit into the fixed size of a PU. The segments are inserted in the data field of a PU. A length indicator shall be added to each PU that includes a border of a SDU, i.e. if a PU does not contain a length indicator the SDU continues in the next PU. The length indicator indicates where the border occurs in the PU. The data after the indicated border can be either a new SDU, padding or piggybacked information. If padding or piggybacking is added another length indicator shall be added, see subclause 9.2.2.8.

11.3.3 Reception of AMD PDU by the receiver

Upon reception of a AMD PDU the receiver shall update VR(R), VR(H) and VR(MR) state variables according to the received PU(s).

If any of the PUs include a Polling bit set to 1 the STATUS PDU transfer procedure shall be initiated.

If the detection of missing PU(s) shall be used and the receiver detects that a PU is missing the receiver shall initiate the STATUS PDU transfer procedure.

11.3.4 Abnormal cases

11.3.4.1 Timer_Poll timeout

Upon expiry of the Timer_Poll the sender shall retransmit the poll. The poll can be retransmitted in either a new PDU or a retransmitted PDU.

11.3.4.2 Receiving a PU outside the receiving window

Upon reception of a PU with SN<VR(R) or SN≥VR(MR) the receiver shall discard the PU. The poll bit shall be considered even if a complete PDU is discarded.

11.3.4.3 Timer_Discard timeout

11.3.4.3.1 SDU discard with explicit signalling

Upon expiry of Timer_Discard the sender shall initiate the SDU discard with explicit signalling procedure.

11.3.4.4 VT(DAT) ~~>~~≥ MaxDAT

If SDU discard after MaxDAT number of retransmission is used and VT(DAT) ~~>~~≥ MaxDAT for any PU the sender shall initiate the SDU discard with explicit signalling procedure for the SDUs to which the PU with VT(DAT)~~>~~ ≥ MaxDAT belongs.

If the SDU discard is not used the sender shall initiate the RLC reset procedure when VT(DAT) ~~>~~≥ MaxDAT.

11.3.4.5 Invalid length indicator value

If the length indicator of a PU has a value that is larger than the PU size, the PU shall be discarded and treated as a missing PU.

11.6 SDU discard with explicit signalling procedure

11.6.1 Purpose

An SDU can be discarded with explicit signalling when MaxDAT number of retransmissions is reached or the transmission time exceeds a predefined value (Timer_Discard) for a SDU in acknowledged mode RLC. Move Receiving Window (MRW) command is sent to the receiver so that AMD PDUs carrying that SDU are discarded in the receiver and the receiver window is updated accordingly. Note that when the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. ~~If one or more segments of a SDU has been transmitted, the SDU shall not be discarded in the transmitter without notification to the receiver.~~

The MRW command is defined as a super-field in the RLC STATUS PDU, and can be piggybacked to status information of transmissions in the opposite direction.

Figure 11.6 below illustrates the elementary procedure for SDU discard with explicit signalling. The sender is the sender of AMD PDUs and it is either the UE or the network and the receiver is the receiver of AMD PDUs and it is either the network or the UE.

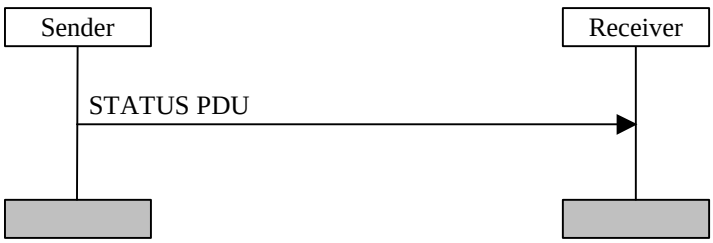


Figure 11.6: SDU discard with explicit signalling

11.6.2 Initiation

This procedure is initiated by the sender when the following conditions are fulfilled:

1) Timer based SDU discard with explicit signalling is used, and Timer_Discard expires for an SDU.

2) SDU discard after MaxDAT number of retransmissions is used, and MaxDAT number of retransmissions is reached ~~or Timer_Discard expires~~ for an SDU ~~in acknowledged mode RLC~~.

The sender shall discard all PUs that contain segments of the associated SDUs. If the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. VT(A) shall be updated when the procedure is terminated, and VT(S) shall be updated when a new MRW SUFI which includes $SN\_MRW_{LENGTH} \geq VT(S)$ is transmitted.

The sender shall transmit a status report 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.

This status report is sent even if the 'STATUS prohibit' is used and the timer 'Timer_Status_Prohibit' or 'Timer_EPC' is active.

The STATUS PDUs have higher priority than data PDUs.

The sender shall start timer Timer_MRW. If a new SDU discard procedure is triggered when Timer_MRW is running, no new MRW SUFIs shall be sent before the current SDU discard procedure is terminated by one of the termination criteria.

11.6.2.1 Piggybacked STATUS PDU

It is possible to piggyback a STATUS PDU on an AMD PDU. If a PDU includes padding a piggybacked STATUS PDU can be inserted instead of the padding.

11.6.2.2 STATUS PDU contents to set

The size of the STATUS PDU shall be equal to one of the allowed PDU sizes. The discard information shall not be split into several MRW SUFIs.

The status report shall include the MRW SUFI, other SUFI fields can be used additionally. MRW SUFI shall convey information about the discarded SDU(s) to the receiver.

In order to discard a single SDU that ends in a PDU with SN≥ VT(A)+Configured_Tx_Window_Size, the LENGTH field in the MRW SUFI shall be set to "0000". If more then one SDU are discarded with the same MRW SUFI, at least the first discarded SDUs must end (i.e. the LI must be located) in a PDU with SN in the interval VT(A)≤ SN <VT(A)+Configured_Tx_Window_Size.

Padding shall be inserted if the SUFI fields do not fill the entire STATUS PDU. If the STATUS PDU contains padding the last SUFI field shall be either an ACK SUFI or a NO_MORE SUFI. If there is no padding in the STATUS PDU, NO_MORE SUFI or ACK SUFI does not need to be included in the STATUS PDU.

11.6.3 Reception of the STATUS PDU by the receiver

The receiver shall upon reception of the STATUS PDU/piggybacked STATUS PDU discard PUs and update the state variables VR(R), VR(H) and VR(MR) according to the received STATUS PDU/piggybacked STATUS PDU. Additionally the receiver should indicate the higher layers of all of the discarded SDUs.

The receiver shall initiate the transmission of a status report containing an MRW_ACK SUFI.

In the MRW_ACK SUFI, SN_ACK shall be set to the new value of VR(R), updated after reception of the MRW SUFI. The N field in the MRW_ACK SUFI shall be set to N_{LENGTH} field in the received MRW SUFI if the SN_ACK field is equal to SN_MRW_{LENGTH}. Otherwise N shall be set to 0.

The last discarded data byte is the byte indicated by the N_{LENGTH}:th LI field of the PU with sequence number SN_MRW_{LENGTH} and the succeeding data byte is the first data byte to be reassembled after the discard. When N_{LENGTH} = 0, the first data byte of the PU with sequence number SN_MRW_{LENGTH} is the first data byte to be reassembled after the discard.

If the MRW SUFI indicates an SN_MRW_i outside the interval $VR(R) \leq SN\_MRW_i < VR(MR)$, the Rx shall consider the sequence number to be below VR(R), unless LENGTH="0000" or at least the first indicated SN_MRW_i in the MRW SUFI is within the interval $VR(R) \leq SN\_MRW_i < VR(MR)$, in which case the sequence number shall be considered to be above or equal to VR(MR).

11.6.4 Termination

The procedure is terminated in the sender in the following cases:

1. On the reception of a STATUS PDU which contains an MRW_ACK SUFI with $SN\_ACK > SN\_MRW_{LENGTH}$
2. On the reception of a STATUS PDU which contains an ACK SUFI indicating $VR(R) > SN\_MRW_{LENGTH}$
3. On reception of a STATUS PDU which contains an MRW_ACK with $SN\_ACK = SN\_MRW_{LENGTH}$ and N is equal to the N_{LENGTH} indicated in the transmitted MRW SUFI.

If one of the termination criteria above is fulfilled, Timer_MRW is stopped and the discard procedure is terminated.

When VT(MRW) reaches MaxMRW, the procedure is terminated and an RLC reset is performed.

11.6.5 Expiration of timer Timer_MRW

If Timer_MRW expires before the discard procedure is terminated, the MRW SUFI shall be retransmitted, VT(MRW) is incremented by one and Timer_MRW restarted. MRW SUFI shall be exactly the same as previously transmitted even though some new SDUs would have been discarded during the running of the Timer_MRW. If the retransmitted STATUS PDU contains other SUFIs than the MRW SUFI, the status information indicated by these SUFIs shall be updated.

11.6.6 Abnormal cases

11.6.6.1 Obsolete/corrupted MRW command

If the MRW command contains outdated information about the receiver window (receiver window already moved further than MRW command is indicating), the MRW command shall be discarded and a status report containing SUFI MRW_ACK shall be transmitted indicating the value of VR(R) and the N field shall be set to zero.

11.6.6.2 VT(MRW) equals MaxMRW

If the number of retransmission of a MRW command (i.e. VT(MRW)) reaches MaxMRW, an error indication shall be passed to RRC and RESET procedure shall be performed.

11.6.6.3 Reception of obsolete MRW_ACK

The received MRW_ACK shall be discarded in the following cases.

1. If timer Timer_MRW is not active.
2. If the SN_ACK field in the received $MRW\_ACK < SN\_MRW_{LENGTH}$ in the transmitted MRW SUFI.
3. If the SN_ACK field in the received MRW_ACK is equal to the SN_MRW_{LENGTH} in the transmitted MRW SUFI and the N field in the received MRW_ACK field is not equal to the N_{LENGTH} field in the transmitted MRW SUFI.

3GPP TSG RAN WG2 meeting #16
Beijing, China, 09-13 October 2000

***Document* R2-002074**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 086r1 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*
for approval [X]
for information [ ]
strategic [ ]
non-strategic [ ]
(for SMG Use only)

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-09-21

Subject: General RLC corrections

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	[X]	**Release:**	Phase 2	[ ]
A	Corresponds to a correction in an earlier release	[ ]		Release 96	[ ]
B	Addition of feature	[ ]		Release 97	[ ]
C	Functional modification of feature	[ ]		Release 98	[ ]
D	Editorial modification	[ ]		Release 99	[X]
				Release 00	[ ]

Reason for change:

1. ~~The description of RLC suspend/resume is corrected to also include UMD.~~ Changes on suspend/resume are covered by another CR and are deleted in this CR.
2. It should be clarified that an ACK SUFI can be included in each STATUS PDU of a status report.
3. The configured Tx window size shall be used in the SDU discard procedure, not the momentary Tx window size.

Clauses affected: 8.1, 9.2.2.11.8, 9.3.3.4, 9.7.6, 11.5.2.2

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

8.1 Primitives between RLC and higher 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	Parameter			
	Req.	**Ind.**	**Resp.**	**Conf.**
RLC-AM-DATA	Data, CNF, MUI	Data, DiscardInfo	Not Defined	MUI
RLC-UM-DATA	Data, Use special LI	Data	Not Defined	Not Defined
RLC-TR-DATA	Data	Data	Not Defined	Not Defined
CRLC-CONFIG	E/R, Stop, Ciphering Elements (UM/AM 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 higher layers to request transmission of a higher layer PDU in acknowledged mode.
- RLC-AM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in acknowledged mode and to indicate higher layers of the discarded RLC SDU in the receiving RLC.
- RLC-AM-DATA-Conf is used by RLC to confirm to higher layers the transmission of a RLC SDU.

RLC-UM-DATA-Req/Ind

- RLC-UM-DATA-Req is used by higher layers to request transmission of a higher layer PDU in unacknowledged mode.
- RLC-UM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in unacknowledged mode.

RLC-TR-DATA-Req/Ind

- RLC-TR-DATA-Req is used by higher layers to request transmission of a higher layer PDU in transparent mode.
- RLC-TR-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in transparent mode.

CRLC-CONFIG-Req

This primitive is used by RRC to establish, re-establish, release or reconfigure the RLC. Ciphering elements are included for UM and AM operation.

CRLC-SUSPEND-Req/Conf

This primitive is used by RRC to suspend the RLC. The N parameter indicates that RLC shall not send a PDU with SN>=VT(S)+N ~~(in AM) or SN>=VT(US)+N (in UM)~~, where N is an integer. RLC informs RRC of the ~~value of~~ VT(S) ~~(in AM) or VT(US) (in UM)~~~~value~~ value in the confirm primitive.

CRLC-RESUME-Req

This primitive is used by RRC to resume RLC when RLC has been suspended.

CRLC-STATUS-Ind

It is used by the RLC to send status information to RRC.

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. The Data parameter may be divided over several RLC PDUs. In case of a RLC-AM-DATA or a RLC-UM-DATA primitive the length of the Data parameter shall be octet-aligned.

2) The parameter Confirmation request (CNF) indicates whether the RLC needs to confirm the correct transmission of the RLC SDU.

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.

4) The parameter E/R indicates (re)establishment, release or modification of RLC If it indicates establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value and RLC shall enter the data transfer ready state. If it indicates re-establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value the RLC buffers shall be emptied and RLC shall enter the data transfer ready state. If it indicates release, all protocol parameters, variables and timers shall be released and RLC shall exit the data transfer ready state. If it indicates modification, the protocol parameters indicated by RRC (e.g. ciphering parameters) shall only be modified with keeping the other protocol parameters, the protocol variables, the protocol timers and the protocol state. RLC shall always be re-established if the PU size is changed.

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

6) The parameter ciphering elements are only applicable for UM and AM operation. These parameters are Ciphering Mode, Ciphering Key, Activation Time (SN to activate a new ciphering configuration) and HFN (Hyper Frame Number).

7) The AM_parameters are only applicable for AM operation. It contains PU size, 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), SDU discard mode (see subclause 9.7.3) and Minimum WSN (see subclause 9.2.2.11.3). The Minimum WSN shall always be greater than or equal to the number of transport blocks in the smallest transport block set.

8) The parameter DiscardInfo indicates the upper layer of each of the discarded RLC SDU. It is applicable only when in-sequence delivery is active and it is purposed to be used when the upper layer requires the reliable data transfer and especially the information of the discarded RLC SDU.

9) The Stop parameter indicates that the RLC entity shall not transmit or receive RLC PDUs.

10) The parameter Use special LI indicates that the LI indicating that a RLC SDU begins in the beginning of a RLC PDU (the first data octet of the PDU is the first octet of an SDU) shall be used. If the RLC SDU does not begin in the beginning of the RLC PDU, or if the LI indicating that a SDU ended exactly in the end of the previous RLC PDU is present, the LI shall not be used.

9.2.2.11.8 The Move Receiving Window (MRW) super-field

The 'Move Receiving Window' super-field is used to request the RLC receiver to move its receiving window and to indicate the amount of discarded SDUs, as a result of a SDU discard in the RLC transmitter. The format is given in the figure below.

Type = **MRW**
LENGTH
SN_MRW_1
...
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 discarded SDU extends above the configured Tx window in the transmitter.

SN_MRW_i

Length: 12 bits

SN_MRW_i is used to indicate the end of each discarded SDU. SN_MRW_i is the sequence number of the PU that contains the LI of the i:th discarded SDU (except when $N_{LENGTH} = 0$, see definition of N_{LENGTH}).

Additionally SN_MRW_{LENGTH} requests the RLC receiver to discard all PUs with sequence number < SN_MRW_{LENGTH}, and to move the receiving window accordingly. In addition, the receiver has to discard the first N_{LENGTH} LIs and the corresponding data bytes in the PU 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 discarded SDU.

N_{LENGTH} indicates which LI in the PU with sequence number SN_MRW_{LENGTH} corresponds to the last discarded SDU. $N_{LENGTH} = 0$ indicates that the last SDU ended in the PU with sequence number $SN\_MRW_{LENGTH} - 1$ and that the first data byte in the PU with sequence number SN_MRW_{LENGTH} is the first data byte to be reassembled next.

9.3.3.4 Local Suspend State

Upon reception of CRLC-SUSPEND-Req from higher layer (RRC) in Acknowledge Data Transfer Ready State the RLC entity is suspended and the Local Suspend state is entered. In the Local Suspend state RLC shall not send a RLC-PDUs with a SN>=VT(S)+N ~~(in AM) or~~. ~~SN>=VT(US)+N (in UM)~~. Upon reception of CRLC-RESUME-Req from higher layer (RRC) in this state, the RLC entity is resumed and the Data Transfer Ready state is entered.

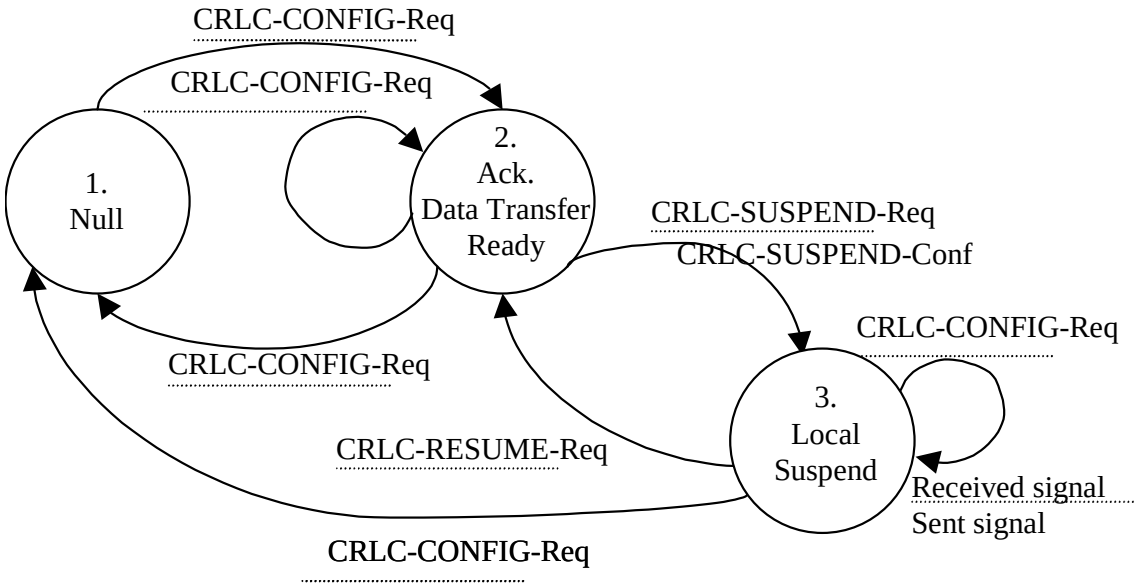


Figure 9.19: The state model for the acknowledged mode entities when local suspend is performed

9.7.6 Local Suspend function ~~for acknowledged mode transfer~~for acknowledged mode transfer

The higher layer (RRC) may suspend the RLC entity. The CRLC-SUSPEND-Req indicates this request. The RLC entity shall, when receiving this request, not send RLC PDUs with SN>=VT(S)+N ~~(in AM) or SN>=VT(US)+N (in UM), where (~~,where N is given by the CRLC_SUSPEND-Req primitive~~)~~. The RLC entity shall acknowledge the CRLC-SUSPEND-Req ordering a suspend with a CRLC-SUSPEND-Conf with the current value of VT(S) ~~(in AM) or VT(US) (in UM)~~. The suspend state is left when a CRLC-RESUME-Req primitive indicating resume is received.

11.5.2.2 STATUS PDU contents to set

The size of the STATUS PDU shall be equal to one of the allowed PDU sizes. The information that needs to be transmitted in a status report can be split into several STATUS PDUs if one STATUS PDU does not accommodate all the information. A SUFI can not be split into several STATUS PDUs. Indication of the same PU 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.

Which SUFI fields to use is implementation dependent, but the status report shall include information about PUs that have been received and information about all PUs detected as missing. No information shall be given for PUs with SN≥VR(H), i.e. PUs that have not yet reached the receiver.

Padding shall be inserted if the SUFI fields do not fill an entire STATUS PDU. If the PDU contains padding the last SUFI field shall be either an ACK SUFI or a NO_MORE SUFI. If there is no padding in the STATUS PDU, NO_MORE SUFI or ACK SUFI does not need to be included in the STATUS PDU.

3GPP TSG-RAN WG2 Meeting #17
Sophia Antipolis, France, 13-17 November

Document R2-002361
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 087r5 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X]
for information [ ]

strategic [ ]
non-strategic [ ]

(for SMG Use only)

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-11-09

Subject: RLC timers

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	[X]	**Release:**	Phase 2	[ ]
A	Corresponds to a correction in an earlier release	[ ]		Release 96	[ ]
B	Addition of feature	[ ]		Release 97	[ ]
C	Functional modification of feature	[ ]		Release 98	[ ]
D	Editorial modification	[ ]		Release 99	[X]
				Release 00	[ ]

Reason for change:

1. The description of the RLC timers are corrected to reflect the decision at WG2 #15 to start RLC timers when a PDU is transmitted on the physical layer (In UE only)
2. The description of poll triggers is corrected.
3. "delivered to MAC" is exchanged by "delivered to lower layer" in several places.
4. The CR 1932 has been merged into this CR

~~The changes in r1 are marked in yellow~~

Changes in r2:

1. "delivered to lower layer" is replaced by "submitted to lower layer"

2. "sent on the physical layer" is replaced by "sent on the radio interface"

3. The definition of transmitted is moved to a new section (4.2.1)

Changes in R4:

1. The definition "sent on the radio interface" is changed to reflect changes in the MAC specification. The new definition is "when successful or unsuccessful transmission of a PDU is indicated by lower layer". The error case that MAC can not transmit a TB is covered by the new definition.

Changes in R5:

1. The sentence defining the word "transmitted" in 4.2.1 is modified.

2. Section 9.7.4 is removed since the changes on this section is done in CR 83r3.

Clauses affected:	4.2.1, 4.2.1.3, 9.5, 9.7.1, 9.7.2, ~~9.7.4~~, 11.3.2, 11.5.2			
Other specs affected:	Other 3G core specifications		→ List of CRs:	
	Other GSM core specifications		→ List of CRs:	
	MS test specifications		→ List of CRs:	
	BSS test specifications		→ List of CRs:	
	O&M specifications		→ List of CRs:	
Other comments:				

help.doc

<--------- double-click here for help and instructions on how to create a CR.

4.2.1 Model of RLC

Figure 4.1 gives an overview model of the RLC layer. The figure illustrates the different RLC peer entities. There is one transmitting and one receiving entity for the transparent mode service and the unacknowledged mode service and one combined transmitting and receiving entity for the acknowledged mode service. In this specification the word transmitted is equivalent to "submitted to lower layer" unless otherwise explicitly stated.~~The RLC entity shall assume a PDU to be transmitted when the PDU is submitted to lower layer~~ The dashed lines between the AM-Entities illustrate the possibility to send the RLC PDUs on separate logical channels, e.g. control PDUs on one and data PDUs on the other. More detailed descriptions of the different entities are given in subclauses 4.2.1.1, 4.2.1.2 and 4.2.1.3.

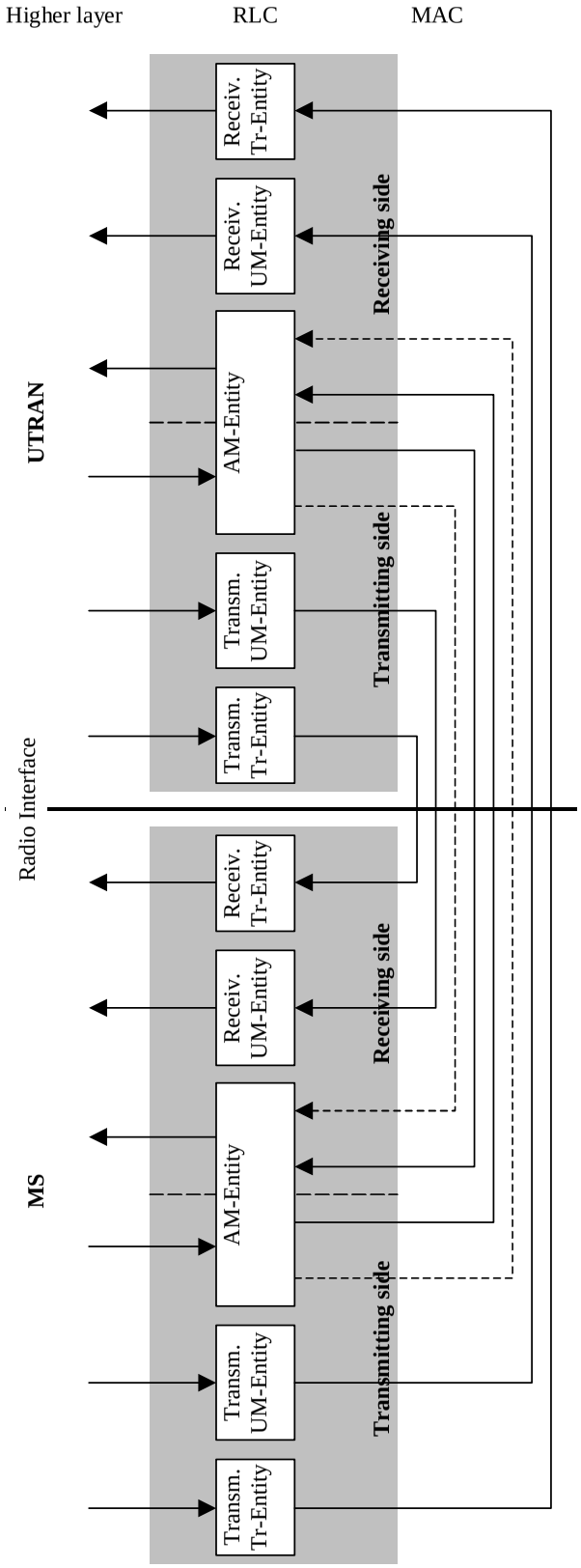


Figure 4.1: Overview model of RLC

4.2.1.3 Acknowledged mode entity

Figure 4.4 below shows the model of an acknowledged mode entity, when one logical channel (shown as a solid line) and when two logical channels (shown as dashed lines) are used.

In case two logical channels are used in the uplink the UTRAN can indicate that the first logical channel shall be used for data PDUs and the second logical channel shall be used for control PDUs. If the indication is not given from the UTRAN, data and control PDUs can be sent on either of the two logical channels. The indication of the logical channel mapping is signalled by RRC.

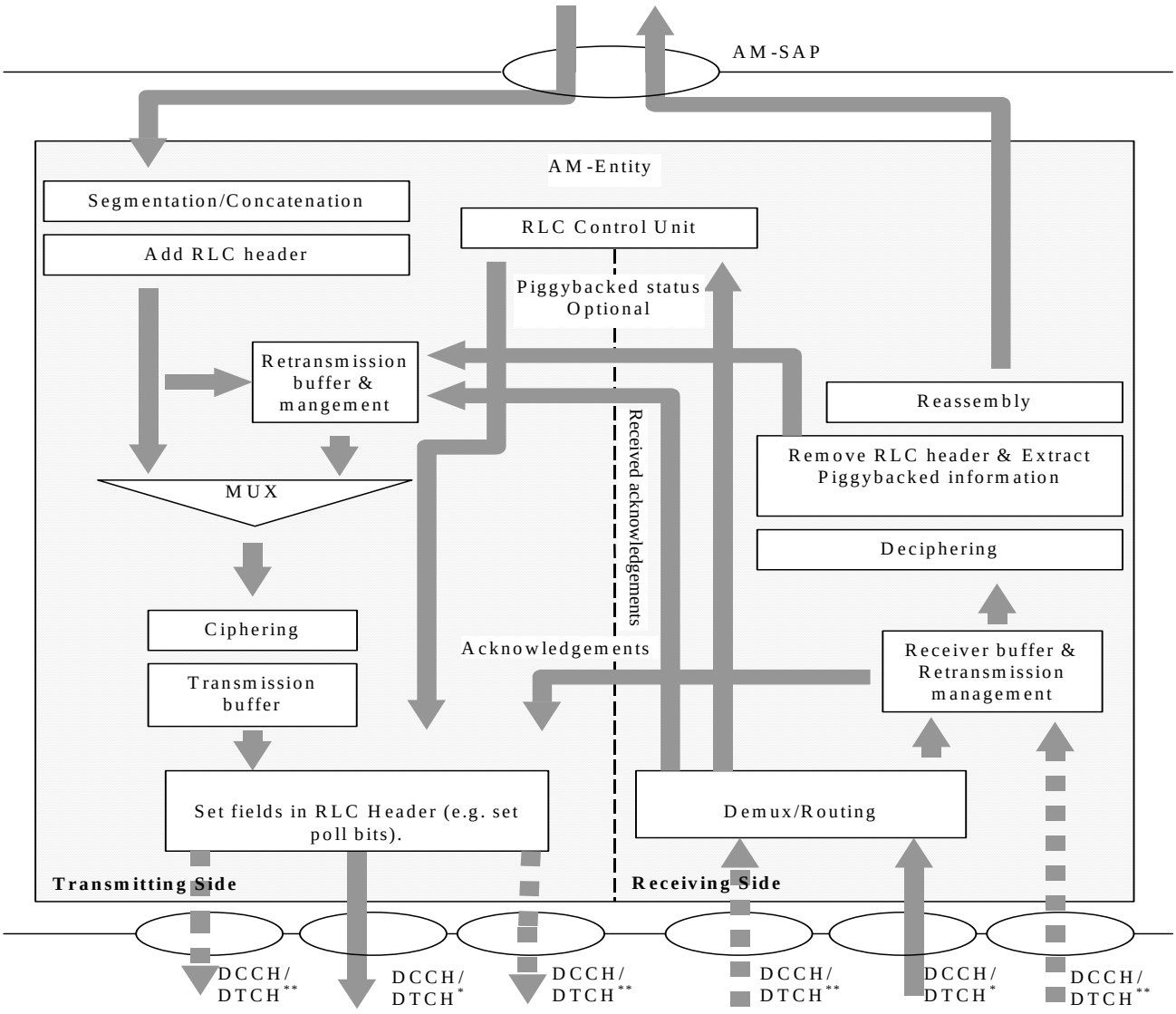


Figure 4.4: Model of a acknowledged mode entity

The transmitting side of the AM-entity receives SDUs from the higher layers. The SDUs are segmented and/or concatenated to PUs of fixed length. PU length is a semi-static value that is decided in bearer setup and can only be changed through bearer reconfiguration by RRC.

For purposes of RLC buffering and retransmission handling, the operation is the same as if there would be one PU per PDU. For concatenation or padding purposes, bits of information on the length and extension are inserted into the beginning of the last PU where data from an SDU is included. Padding can be replaced by piggybacked status information. This includes setting the poll bit.

If several SDUs fit into one PU, they are concatenated and the appropriate length indicators are inserted into the beginning of the PU. After that the PUs are placed in the retransmission buffer and the transmission buffer. One PU is included in one RLC PDU.

The MUX then decides which PDUs and when the PDUs are ~~delivered~~ submitted to ~~MAC~~lower layer. The PDUs are ~~delivered~~ submitted via a function that completes the RLC-PDU header. The RLC entity shall assume a PDU to be transmitted when the PDU is ~~delivered~~submitted to lower layer. The fixed 2 octet AMD PDU header is not ciphered.

When Piggybacking mechanism is applied the padding is replaced by control information, in order to increase the transmission efficiency and making possible a faster message exchange between the peer to peer RLC entities. The piggybacked control information is not saved in any retransmission buffer. The piggybacked control information is contained in the piggybacked STATUS PDU, which is in turn included into the AMD-PDU. The piggybacked STATUS PDUs will be of variable size in order to match with the amount of free space in the AMD PDU.

The retransmission buffer also receives acknowledgements from the receiving side, which are used to indicate retransmissions of PUs and when to delete a PU from the retransmission buffer.

The Receiving Side of the AM-entity receives PDUs through one of the logical channels from the MAC sublayer. The RLC-PDUs are expanded into separate PUs and potential piggybacked status information are extracted. The PUs are placed in the receiver buffer until a complete SDU has been received. The receiver buffer requests retransmissions of PUs by sending negative acknowledgements to the peer entity. After that the headers are removed from the PDUs and the PDUs are reassembled into a SDU. Finally the SDU is delivered to the higher layer. The receiving side also receives acknowledgements from the peer entity. The acknowledgements are passed to the retransmission buffer on the transmitting side.

9.5 Timers

a) Timer_Poll.

This timer is only used when the poll timer trigger is used. It is started when the successful or unsuccessful transmission ~~transmitting side sends~~ of a PDU containing a poll is indicated by lower layer ~~on the physical layer~~radio interface ~~to the peer entity~~(in UE) or ~~transmits~~ a PDU containing a poll is ~~, i.e. delivers~~ submitt~~eds the PDU~~ to lower layer (in UTRAN). The timer is stopped when receiving a STATUS PDU that contains an acknowledgement of all AMD PDUs with SN up to and including VT(S)-1 at the time the poll was ~~transmitted (i.e. delivered~~submitted to lower layer~~),(~~or when a negative acknowledgement of the same PU is received~~)~~. The value of the timer is signalled by RRC.

If the timer expires and no STATUS PDU fulfilling the criteria above has been received, the receiver is polled once more (either by the transmission of a PDU which was not yet sent, or by a retransmission) and the timer is restarted at the time specified above, with a new value of VT(S)-1. ~~If there is no PU to be transmitted and all PUs have already been acknowledged, the receiver shall not be polled.~~

If a new poll is sent when the timer is running the timer~~it~~ is restarted at the time specified above, with a new value of VT(S)-1.

b) Timer_Poll_Prohibit.

This timer is only used when the poll prohibit function is used. It is used to prohibit transmission of polls within a certain period. The timer shall be started when the successful or unsuccessful transmission of a PDU containing a poll is indicated by lower layer~~sent on the physical layer~~radio ~~interface~~ (in UE) or a PDU containing a poll is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN). The prohibit time is calculated from the time a PDU containing a poll is ~~delivered~~submitted to lower layer until the timer has expired. A poll shall be delayed until the prohibit time~~timer~~ expires if a poll is triggered ~~when the timer is active.~~ during the prohibit time. Only one poll shall be transmitted when the prohibit time~~r~~ expires even if several polls were triggered during the prohibit time~~when the timer was active~~. ~~If there is no PU to be transmitted and all PUs have already been acknowledged, a poll shall not be transmitted~~. This timer will not be stopped by a received STATUS PDU. The value of the timer is signalled by RRC.

c) Timer_EPC.

This timer is only used when the EPC function is used and it accounts for the roundtrip delay, i.e. the time when the first retransmitted PU should be received after a status report has been sent. The timer is started when the successful or unsuccessful transmission of the first STATUS PDU of a status report is indicated by lower layer~~sent on the physical layer~~radio ~~interface~~ (in UE) or the first STATUS PDU of a status report is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN) and when it expires EPC can start decrease (see subclause 9.7.~~43~~). The value of the timer is signalled by RRC.

d) Timer_Discard.

This timer is used for the SDU discard function. In the transmitter, the timer is activated upon reception of a SDU from higher layer. One timer is used for each SDU that is received from higher layer. If the SDU has not been acknowledged and/or transmitted when the timer expires, the SDU is discarded. Following which, if the SDU discard function uses explicit signalling, a Move Receiving Window request is sent to the receiver. The value of the timer is signalled by RRC.

e) Timer_Poll_Periodic.

This timer is only used when the timer based polling is used. The timer is started when the RLC entity is created. Each time the timer expires, the timer is restarted and a poll is ~~transmitted~~ triggered (either by the transmission of a PDU which was not yet sent, or by a retransmission)~~ and the timer is restarted~~. If there is no PU to be transmitted and all PUs have already been acknowledged, a poll shall not be ~~transmitted~~ triggered and the timer shall only be restarted. The value of the timer is signalled by RRC.

f) Timer_Status_Prohibit.

This timer is only used when the STATUS prohibit function is used. It prohibits the receiving side from sending status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. The timer is started when the successful or unsuccessful transmission of the last STATUS PDU in a status report is indicated by lower

layersent ~~on the physical layer~~radio interface (in UE) or the last STATUS PDU in a status report is~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN). The prohibit time is calculated from the time the last STATUS PDU of a status report is ~~delivered~~submitted to lower layer until the timer has expired and no new status report containing the mentioned SUFIs can be transmitted ~~before the timer has expired~~ during the prohibit time. The timer does not prohibit transmission of the SUFIs MRW, MRW_ACK, WINDOW or NO_MORE. The value of the timer is signalled by RRC.

g) Timer_Status_Periodic.

This timer is only used when timer based status report sending is used. The timer is started when the RLC entity is created. Each time the timer expires the transmission of a status report is ~~transmitted~~ triggered and the timer is restarted. The value of the timer is signalled by RRC.

h) Timer_RST.

This timer is used to detect the loss of RESET ACK PDU from the peer RLC entity. This timer is ~~set~~ started when the successful or unsuccessful transmission of a RESET PDU is indicated by lower layer ~~sent on the physical layer~~radio interface (in UE) or a RESET PDU is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN). It will only be stopped upon reception of RESET ACK PDU, i.e. this timer is not stopped when an RLC reset occurs which was initiated from the peer RLC entity. If it expires, RESET PDU will be retransmitted. The value of the timer is signalled by RRC.

i) Timer_MRW.

This timer is used as part of the Move Receiving Window protocol. It is used to trigger the retransmission of a status report containing an MRW SUFI field. The timer is started when the successful or unsuccessful transmission of a STATUS PDU containing the MRW SUFI is ~~first~~ indicated by lower layer~~sent on the physical layer~~radio interface (in UE) or a STATUS PDU containing the MRW SUFI is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN). Each time the timer expires the MRW SUFI is retransmitted and the timer is restarted (at the time specified above~~when the STATUS PDU containing the MRW SUFI is retransmitted~~). It shall be stopped when one of the termination criteria for the SDU discard is fulfilled. The value of the timer is signalled by RRC.

9.6 Protocol Parameters

The values of the protocol parameters in this section are signalled by RRC.

a) MaxDAT.

It is the maximum value for the number of retransmissions of a PU. This parameter is an upper limit of counter VT(DAT). When the value of VT(DAT) comes to MaxDAT, error recovery procedure will be performed.

b) Poll_PU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_PU PU. This is an upper limit for the VT(PU) state variable, when VT(PU) reaches Poll_PU a poll is transmitted to the peer entity.

c) Poll_SDU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_SDU SDU. This is an upper limit for the VT(SDU) state variable, when VT(SDU) reaches Poll_SDU a poll is transmitted to the peer entity.

d) Poll_Window.

This parameter indicates when the transmitter should poll the receiver in case of performing window-based polling. A poll is triggered for each PUwhen:

1) VT(S)<VT(MS), Tx_Window_Size>0, and

$$\left[1 - \frac{(Tx\_Window\_Size + \mathrm{VT(MS)} - \mathrm{VT(S)}\text{-}1) \bmod Tx\_Window\_Size}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$

.

2) VT(S)≥VT(MS), and Tx_Window_Size>0

When Tx_Window_Size=0, the transmitter does not perform window-based polling.

e) MaxRST.

It is the maximum value for the number of retransmission of RESET PDU. This parameter is an upper limit of counter VT(RST). When the value of VT(RST) comes to MaxRST, the higher layer (RRC) is notified.

f) Tx_Window_Size.

The maximum allowed transmitter window size.

g) Rx_Window_Size.

The maximum allowed receiver window size.

h) MaxMRW.

It is the maximum value for the number of retransmissions of a MRW command. This parameter is an upper limit of counter VT(MRW). When the value of VT(MRW) comes to MaxMRW, error recovery procedure will be performed.

9.7 Specific functions

9.7.1 Polling function for acknowledged mode transfer

The transmitter of AMD PDUs may poll the receiver for a status report (consisting of one or several STATUS PDUs). The Polling bit in the AMD PDU indicates the poll request. If there is no PU to be transmitted and all PUs have already been acknowledged, the receiver shall not be polled. There are several triggers for setting the polling bit. The network (RRC) controls, which triggers should be used for each RLC entity. Following triggers are possible:

1) Last PU in buffer.

The sender ~~transmits~~ triggers a poll when the last PU available for transmission is transmitted.

2) Last PU in retransmission buffer.

The sender ~~transmits~~ triggers a poll when the last PU to be retransmitted is transmitted.

3) Poll timer.

The timer Timer_Poll is started when the successful or unsuccessful transmission of a PDU containing a poll is indicated by lower layer ~~sent on the physical layer~~radio interface (in UE) or a PDU containing a poll is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN) ~~transmitted to the receiver~~ and if the criterion for stopping the timer has not occurred before the timer Timer_Poll expires a new poll is triggered~~transmitted to the receiver~~.

4) Every Poll_PU PU.

The sender ~~polls the receiver~~ triggers a poll every Poll_PU PU. Both retransmitted and new PU~~u~~s shall be counted.

5) Every Poll_SDU SDU.

The sender ~~polls the receiver~~triggers a poll every Poll_SDU SDU.

6) Poll_Window% of transmission window.

The sender ~~polls the receiver~~triggers a poll when it has reached Poll_Window% of the transmission window.

7) Timer based.

The sender ~~polls the receiver~~triggers a poll periodically.

Either the trigger "Last PU in buffer" and "Last PU in retransmission buffer" or "Timer based" can be chosen to avoid deadlock for every RLC entity. The network also controls if the poll prohibit function shall be used. The poll bit shall be set to 0 if the poll prohibit function is used and the timer Timer_Poll_Prohibit is active. If a poll was triggered during the prohibit time defined in subclause 9.5 b) (Timer_Poll_Prohibit), the poll shall be delayed until the timer expires. Only one poll shall be transmitted when the timer expires even if several polls were triggered during the prohibit time. This function has higher priority than any of the above mentioned triggers.

9.7.2 STATUS transmission for acknowledged mode

The receiver of AMD PDUs transmits status reports (each status report consists of one or several STATUS PDUs) to the sender in order to inform about which PUs that have been received and not received. There are several triggers for sending a status report. The network (RRC) controls which triggers should be used for each RLC entity, except for one, which is always present. The receiver shall always send a status report when receiving a poll request. Except for that trigger following triggers are configurable:

1) Detection of missing PU(s).

If the receiver detects one or several missing PUs it shall ~~send~~ trigger the transmission of a status report to the sender.

2) Timer based STATUS transfer.

The receiver ~~transmits~~ triggers the transmission of a status report periodically to the sender. The timer Timer_Status_Periodic controls the time period.

3) The EPC mechanism.

The Timer EPC is started when the successful or unsuccessful transmission of the first STATUS PDU of a status report is indicated by lower layer ~~sent on the physical layer~~radio interface (in UE) or the first STATUS PDU of a status report is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN)~~transmitted to the peer entity~~. If not all PUs requested for retransmission have been received before the Timer EPC has expired a new status report is transmitted to the peer entity. 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. The network (RRC) controls which functions should be used for each RLC entity. If any of the following functions is used the sending of the status report shall be delayed, even if any of the conditions above are fulfilled:

1) STATUS prohibit.

The Timer_Status_Prohibit is started when the successful or unsuccessful transmission of the last STATUS PDU of a status report is indicated by lower layer ~~sent on the physical layer~~radio interface (in UE) or the last STATUS PDU of a status report is ~~transmitted, i.e. delivered~~submitted to lower layer (in UTRAN) ~~transmitted to the peer entity~~. The prohibit time is calculated from the time the last STATUS PDU of a status report is ~~delivered~~submitted to lower layer until the timer has expired~~As long as the timer is running the receiving side is not allowed to send a status report to the peer entity~~. The receiving side is not allowed to transmit a status report during the prohibit time. If a status report was triggered ~~while the timer was running~~during the prohibit time, the status report is transmitted after the ~~timer has~~prohibit time has expired. The receiver shall only send one status report, even if there are several triggers ~~when the timer running~~during the prohibit time. This timer only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

2) The EPC mechanism.

If the EPC mechanism is active and the sending of a status report is triggered it shall be delayed until the EPC mechanism has ended. The receiver shall only send one status report, even if there are several triggers when the timer is active or the counter is counting down. This mechanism only prohibits the transmission of status reports containing any of the SUFIs LIST, BITMAP, RLIST or ACK. Status reports containing other SUFIs are not prohibited.

11.3 Acknowledged mode data transfer procedure

11.3.1 Purpose

The acknowledged mode data transfer procedure is used for transferring of data between two RLC peer entities, which are operating in acknowledged mode. Figure 11.3 below illustrates the elementary procedure for acknowledged mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.

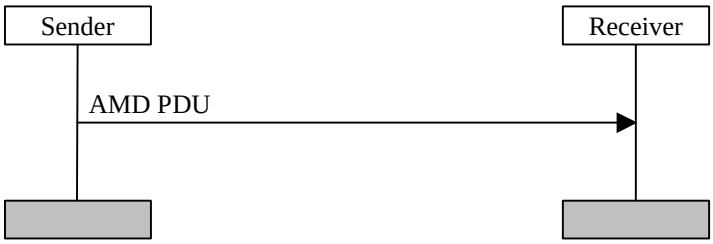


Figure 11.3: Acknowledged mode data transfer procedure

11.3.2 Initiation

The sender initiates this procedure upon a request of acknowledged mode data transfer from higher layer or upon retransmission of PUs. Retransmitted PUs have higher priority than PUs transmitted for the first time.

The sender is only allowed to retransmit PUs that have been indicated missing by the receiver. An exception is the PU with SN VT(S)-1 can be retransmitted. In addition, a PU that has not yet been acknowledged, may be retransmitted if the configured transmitter window size is less than 2048.

RLC shall segment the data received from the higher layer into PUs. When the sender is in data transfer ready state one or several PUs are included in one AMD PDU, which is sent to the receiver. The 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. In the UE, the PDUs that can not be transmitted in a TTI (i.e. MAC has indicated that some of the available PDUs can not be transmitted) shall be buffered according to the discard configuration set by RRC.

The VT(DAT) state variables shall be updated for each AMD PDU that is transmitted. The PDU shall not include any PU with Sequence Number ≥ VT(MS) , except the PU with sequence number VT(S)-1 which may be included also when VT(S) > VT(MS).

If the poll bit is set in any of the AMD PDUs and the timer Timer_Poll shall be used the sender shall start the timer Timer_Poll when the successful or unsuccessful transmission of a PDU with the set poll bit is indicated by lower layer ~~sent on the physical layer~~radio interface (in UE) or ~~delivered~~submitted to lower layer (in UTRAN).~~delivered to MAC.~~

If timer based SDU discard is used the timer Timer_Discard shall be started when the RLC entity receives an SDU from higher layer. One timer is used for each SDU that is received from higher layer.

If the trigger for polling, "Every Poll_PU PU", is used the VT(PU) shall be increased by 1 for each PU that is transmitted.

If the trigger for polling, "Every Poll_SDU SDU", is used the VT(SDU) shall be increased by 1 for each SDU that is transmitted.

11.3.2.1 AMD PDU contents to set

If the PDU is transmitted for the first time, the Sequence Number field shall be set equal to VT(S) and VT(S) shall be updated.

The setting of the Polling bit is specified in subclause 11.3.2.1.1.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number of octets between the end of the header fields and the end of the segment. If the PDU is exactly

filled with the last segment of a SDU and there is no room for a length indicator field a length indicator field set to only 0's shall be included in the next PDU. How to perform the segmentation of a SDU is specified in subclause 11.3.2.1.2.

11.3.2.1.1 Setting of the Polling bit

The Polling bit shall be set to 1 if any of following conditions are fulfilled except when the poll prohibit function is used and the timer Timer_Poll_Prohibit is active (the different triggers are described in 9.7.4):

1) Last PU in buffer is used and the last PU available for transmission is transmitted.

2) Last PU in retransmission buffer is used and the last PU to be retransmitted is transmitted.

3) Poll timer is used and timer Timer_Poll has expired.

4) Every Poll_PU PU is used and when VT(PU)=Poll_PU.

5) Every Poll_SDU is used and VT(SDU)=Poll_SDU and the PDU contains the last segment that SDU.

6) Poll_Window(%) of transmission window is used, Tx_Window_Size>0, VT(S)<VT(MS), and

$$\left[1 - \frac{(Tx\_Window\_Size + \mathrm{VT(MS)} - \mathrm{VT(S)} - 1) \bmod Tx\_Window\_Size}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$

.

7) Poll_Window (%) of transmission window is used, Tx_Window_Size>0 and VT(S)≥VT(MS).

8) Timer based polling is used and Timer_Poll_Periodic has expired.

9) Poll prohibit shall be used, the timer Timer_Poll_Prohibit has expired and one or several polls were prohibited during the time Timer_Poll_Prohibit was active.

11.5 STATUS report transfer procedure

11.5.1 Purpose

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. The receiver is the receiver of AMD PDUs and it is either the UE or the network and the sender is the sender of AMD PDUs and it is either the network or the UE.

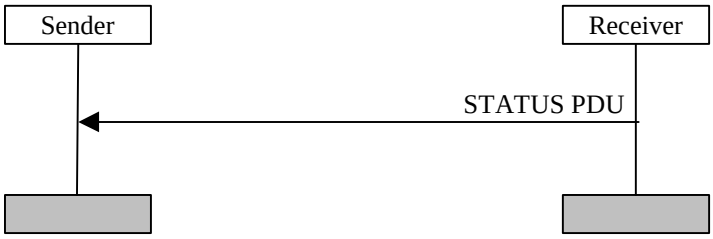


Figure 11.5: Status report transfer procedure

11.5.2 Initiation

The receiver in any of following cases initiates this procedure:

1) The poll bit in a received AMD PDU is set to 1.

2) Detection of missing PUs is used and a missing PU is detected.

3) The timer based STATUS transfer is used and the timer Timer_Status_Periodic has expired.

The receiver shall transmit a status report on the DCCH logical channel if the receiver is located in the control plane and on the DTCH if it is located in the user plane. Separate logical channels can be assigned for AMD PDU transfer and for Control PDU transfer.

The STATUS PDUs have higher priority than data PDUs.

There are two functions that can prohibit the receiver from sending a status report. If any of following conditions are fulfilled the sending of the status report shall be delayed, even if any of the conditions above are fulfilled:

1) STATUS prohibit is used and the timer Timer_Status_Prohibit is active.

   The status report shall be transmitted after the Timer_Status_Prohibit has expired. The receiver shall send only one status report, even if there are several triggers when the timer is ~~running~~active. The rules for when the timer Timer_status_Prohibit is active are defined in subclause 9.5.

2) The EPC mechanism is used and the timer Timer_EPC is active or VR(EP) is counting down.

   The status report shall be transmitted after the VR(EP) has reached 0. The receiver send only one status report, even if there are several triggers when the timer is active or the counter is counting down. The rules for when the timer Timer_EPC is active are defined in subclause 9.5.

If the timer based STATUS transfer shall be used and the Timer_Status_Periodic has expired it shall be restarted.

If the EPC mechanism shall be used the timer Timer_EPC shall be started and the VR(EP) shall be set equal to the number PUs requested to be retransmitted.

3GPP TSG RAN WG2 meeting #16
Beijing, China, 09-13 October 2000

***Document* R2-002076**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 088r1 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*
for approval [X] for information [ ]
strategic [ ] non-strategic [ ] *(for SMG Use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-09-21

Subject: Reset procedure

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

1. The current RLC reset procedure can lead to unsynchronised hyper frame number (HFN) values between UE and UTRAN, which will lead to ciphering failure as discussed on the email reflector. To correct this mechanism, HFN values are indicated in the RESET PDU and RESET ACK PDU. With this correction, RLC can maintain synchronisation of HFN values during reset.

2. Minor editorial updates after RLC-adhoc has been made in 11.4.1.

Clauses affected: 9.2.1.7, 9.2.2.X (new), 9.3.3.2, 9.3.3.3, 11.4.1, 11.4.2.1, 11.4.3, 11.4.3.1, 11.4.4, 11.4.5.3

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

9.2.1.7 RESET, RESET ACK PDU

The RESET PDU and RESET ACK PDU has a one-bit sequence number field (RSN). With the aid of this field the Receiver can define whether the received RESET PDU is transmitted by the Sender for the first time or whether it is a retransmission of a previous RESET PDU..

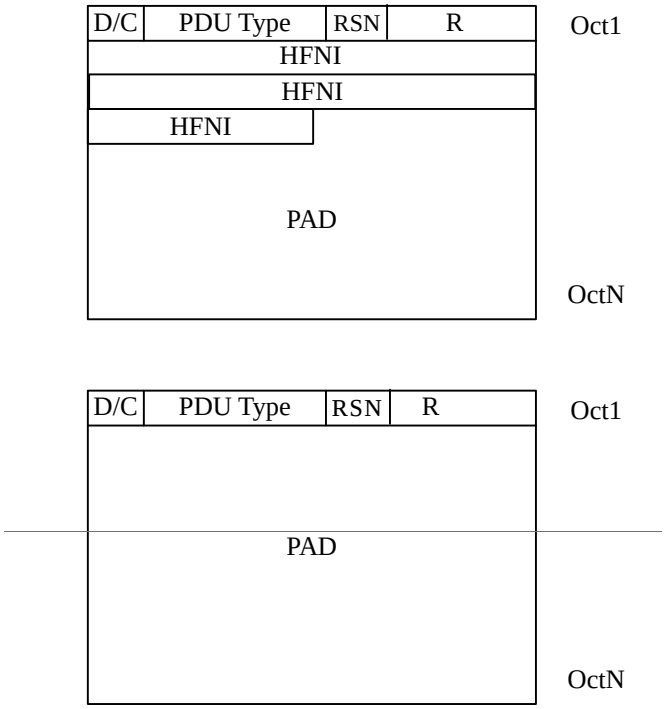


Figure 9.6: RESET, RESET ACK PDU

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 sequence number value as the original RESET PDU. Otherwise it will have the next reset sequence number. The initial value of this field is zero. The value of this field shall be reinitialized when the RLC is re-established. It shall not be reinitialized when the RLC is reset.

9.2.2.14~~X~~ 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.

9.3.3 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 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 an CRLC-CONFIG-Req from higher layer the RLC entity is created and acknowledged data transfer ready state is entered.

9.3.3.2 Acknowledged Data Transfer Ready State

In the acknowledged data transfer ready state, acknowledged mode data can be exchanged between the entities. Upon reception of a CRLC-CONFIG-Req from higher layer the RLC entity is terminated and the null state is entered.

Upon errors in the protocol, the RLC entity sends a RESET PDU to its peer and enters the reset pending state.

Upon reception of a RESET PDU, the RLC entity resets the protocol (resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value, ~~increments the hyper frame number if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU~~ sets the hyper frame number HFN (DL HFN when the RESET is received in UE or UL HFN when the RESET is received in UTRAN) equal to the HFNI field in the RESET PDU~~)~~ and responds to the peer entity with a RESET ACK PDU.

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

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 CRLC-CONFIG-Req from higher layer the RLC entity is terminated and the null state is entered.

Upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU, the RLC entity resets the protocol (resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value, ~~-~~ sets the hyper frame number HFN (DL HFN when the RESET ACK is received in UE or UL HFN when the RESET ACK is received in UTRAN) equal to the HFNI field in the RESET ACK ~~increments the hyper frame number~~) and one of the following state transitions take place.

The RLC entity enters the acknowledged data transfer ready state if Reset Pending State was entered from Acknowledged Data Transfer Ready State or if Reset Pending State was entered from Local Suspend State and a CRLC-RESUME–Req was received in Reset Pending State.

The RLC entity enters into Local Suspend State if Reset Pending State was entered from Local Suspend State or if Reset Pending State was entered from Acknowledged Data Transfer Ready State and a CRLC-SUSPEND-Req was received in Reset Pending State.

Upon reception of a RESET ACK PDU with a different RSN value as in the corresponding RESET PDU the RESET ACK PDU is discarded.~~-~~

Upon reception of a RESET PDU, the RLC entity resets the protocol (resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value, sets the hyper frame number HFN (DL HFN when the RESET is received in UE or UL HFN when the RESET is received in UTRAN) equal to the HFNI field in the RESET PDU~~increments the hyper frame number if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU~~), sends a RESET ACK PDU and stays in the reset pending state.

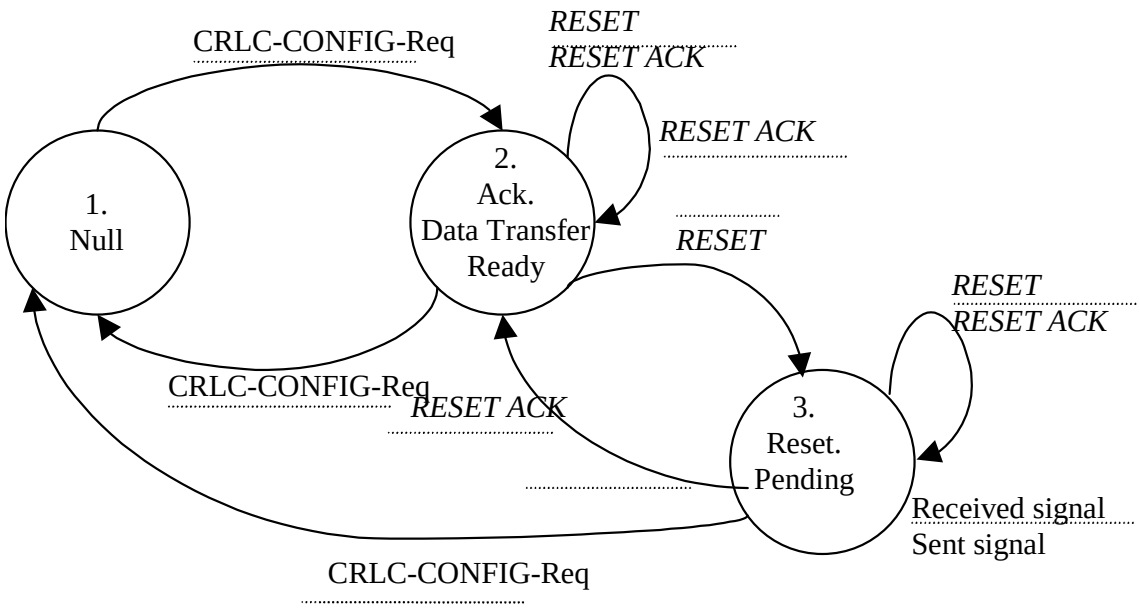


Figure 9.18: The state model for the acknowledged mode entities when reset is performed

9.3.3.4 Local Suspend State

Upon reception of CRLC-SUSPEND-Req from higher layer (RRC) in Acknowledge Data Transfer Ready State the RLC entity is suspended and the Local Suspend state is entered. In the Local Suspend state RLC shall not send a RLC-PDUs with a SN>=VT(S)+N. Upon reception of CRLC-RESUME-Req from higher layer (RRC) in this state, the RLC entity is resumed and the Data Transfer Ready state is entered.

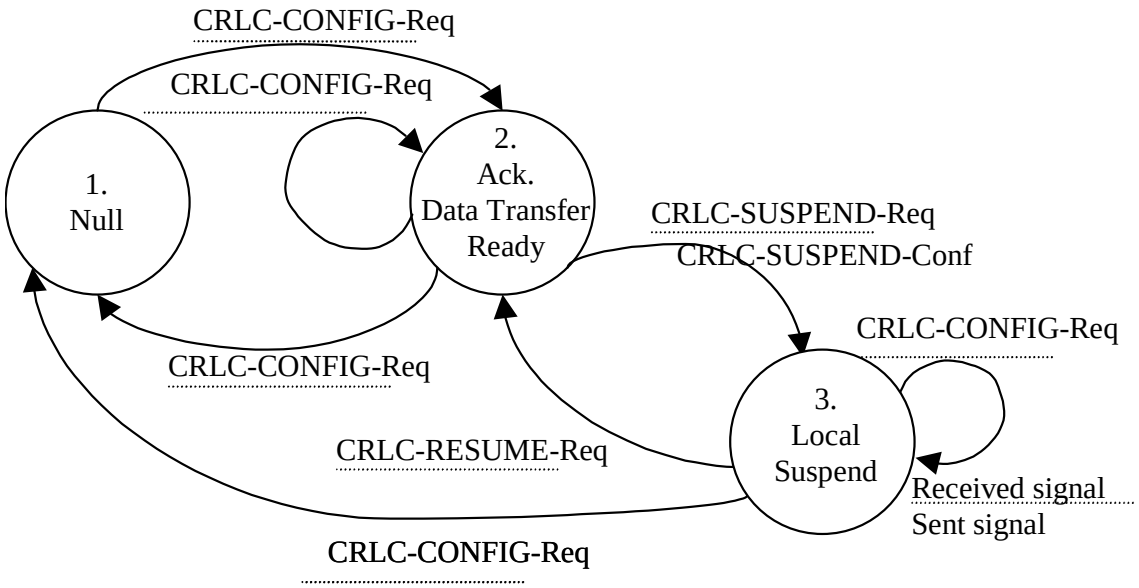


Figure 9.19: The state model for the acknowledged mode entities when local suspend is performed

11.4 RLC reset procedure

11.4.1 Purpose

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 a RLC reset. The sender can be either the UE or the network and the receiver is either the network or the UE. 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~~currently used~~ UL HFN and DL HFN used by the RLC entity are exchanged between UE and UTRAN. After the reset procedure is terminated, the UL HFN and DL HFN shall be increased with one in both UE and UTRAN, and the updated HFN values shall be used after the reset procedure.

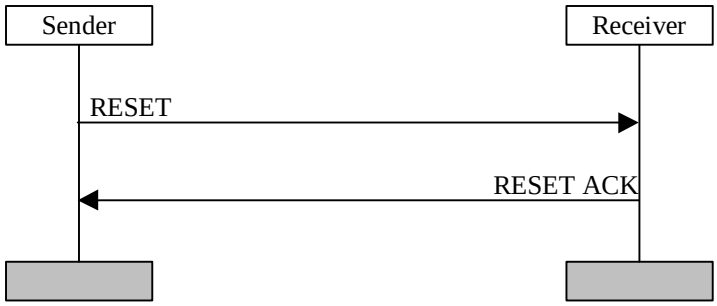


Figure 11.4: RLC reset procedure

11.4.2 Initiation

The procedure shall be initiated when a protocol error occurs.

The sender sends the RESET PDU when it is in data transfer ready state and enters reset pending state. The sender shall start the timer Timer_RST and increase VT(RST) with 1. The RESET PDU 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.

The RESET PDU has higher priority than data PDUs.

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, i.e., 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 size of the RESET PDU shall be equal to one of the allowed PDU sizes. The hyper frame number indicator field (HFNI) shall be set equal to the currently used HFN (DL HFN when the RESET is sent by UTRAN or UL HFN when the RESET is sent by the UE). The RSN field shall indicate the sequence number of the RESET PDU. 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 respond with a RESET ACK PDU. The receiver resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value.

~~In the received RESET PDU the Receiver shall check the value of RSN (Reset Sequence Number) field. If the value of the RSN field is different from the RSN value in the previously received RESET PDU the~~ When a RESET PDU is received, the r~~R~~receiver shall ~~increase the value of the~~set the HFN (DL HFN when the RESET is received in UE or UL HFN when the RESET is received in UTRAN) equal to the HFNI field in the received RESET PDU.~~by one.~~

~~If the value of the RSN is equal to the RSN value in the previously received RESET PDU, (i.e. the RESET PDU is a retransmitted RESET PDU) the value of the HFN shall not be increased and only a RESET ACK PDU shall be sent to the peer RLC entity.~~

The RESET ACK PDU 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.

The RESET ACK PDU has higher priority than data PDUs.

11.4.3.1 RESET ACK PDU contents to set

The size of the RESET ACK PDU shall be equal to one of the allowed PDU sizes. The RSN field shall always be set to the same value as in the corresponding RESET PDU. ~~.~~The hyper frame number indicator field (HFNI) shall be set equal to the currently used HFN (DL HFN when the RESET ACK is sent by UTRAN or UL HFN when the RESET ACK is sent by the UE).

11.4.4 Reception of the RESET ACK PDU by the sender

When the sender is in reset pending state and receives a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU the Timer_RST shall be stopped and the value of the HFN (DL HFN when the RESET ACK is received in UE or UL HFN when the RESET ACK is received in UTRAN) shall be ~~increased by one~~set equal to the HFNI field in the received RESET ACK PDU. The sender resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value. The sender shall enter data transfer ready state.

Upon reception of a RESET ACK PDU with a different RSN value as in the corresponding RESET PDU the RESET ACK PDU is discarded.

Upon reception of a RESET ACK PDU in data transfer ready state the RESET ACK PDU is discarded.

11.4.5 Abnormal cases

11.4.5.1 Timer_RST timeout

Upon expiry of Timer_RST the sender shall retransmit the RESET PDU and increase VT(RST) with 1. In the retransmitted RESET PDU the value of the RSN field shall not be incremented.

11.4.5.2 VT(RST) ≥ MaxRST

If VT(RST) becomes larger or equal to MaxRST the RRC layer shall be informed.

11.4.5.3 Reception of the RESET PDU by the sender

Upon reception of a RESET PDU in reset pending state the sender shall respond with a RESET ACK PDU. The sender resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value. However, VT(RST) and Timer_RST are not reset. The hyper frame number, HFN (DL HFN when the RESET is received in UE or UL HFN when the RESET is received in UTRAN) is set equal to the HFNI field in the received RESET PDU~~incremented if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU~~. The sender shall stay in the reset pending state. The sender shall enter data transfer ready state only upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU.

3GPP TSG-RAN WG2 Meeting #16
Beijing, China, 9-13 October, 2000

***Document* R2-002070**
e.g. for 3GPP use the format TP-99xxx or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322	**CR**	**089r1**	Current Version:	3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑* | for approval **X** | for information | strategic | non-strategic | *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from:* ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-10-02

Subject: Editorial corrections to RLC

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

The separate description of RLC re-establishment has been removed. Instead it should be clarified (in 25.303) that in SRNS relocation the UE RLC is re-established in such a way that all data buffers are flushed.

Clarification that VT(A) can only be updated when an ACK SUFI (or MRW_ACK SUFI) is included in the STATUS PDU.

Clarification that the case where the SN_ACK field in the received MRW_ACK is greater than SN_MRW_{LENGTH} in the transmitted MRW SUFI is valid only if the N field in the received MRW_ACK is zero. If the N field is not zero the MRW_ACK is discarded.

Clauses affected: 8.2, 9.2.2.11.2, 11.6.1, 11.6.4, 11.6.6.3

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:


help.doc

<---------- double-click here for help and instructions on how to create a CR.

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. The Data parameter may be divided over several RLC PDUs. In case of a RLC-AM-DATA or a RLC-UM-DATA primitive the length of the Data parameter shall be octet-aligned.

2) The parameter Confirmation request (CNF) indicates whether the RLC needs to confirm the correct transmission of the RLC SDU.

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.

4) The parameter E/R indicates (re-)establishment, release or modification of RLC. If it indicates (re-)establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value and RLC shall enter the data transfer ready state. ~~If it indicates re-establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value the RLC buffers shall be emptied and RLC shall enter the data transfer ready state.~~ If it indicates release, all protocol parameters, variables and timers shall be released and RLC shall exit the data transfer ready state. If it indicates modification, the protocol parameters indicated by RRC (e.g. ciphering parameters) shall only be modified with keeping the other protocol parameters, the protocol variables, the protocol timers and the protocol state. RLC shall always be re-established if the PU size is changed.

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

6) The parameter ciphering elements are only applicable for UM and AM operation. These parameters are Ciphering Mode, Ciphering Key, Activation Time (SN to activate a new ciphering configuration) and HFN (Hyper Frame Number).

7) The AM_parameters are only applicable for AM operation. It contains PU size, 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), SDU discard mode (see subclause 9.7.3) and Minimum WSN (see subclause 9.2.2.11.3). The Minimum WSN shall always be greater than or equal to the number of transport blocks in the smallest transport block set.

8) The parameter DiscardInfo indicates the upper layer of each of the discarded RLC SDU. It is applicable only when in-sequence delivery is active and it is purposed to be used when the upper layer requires the reliable data transfer and especially the information of the discarded RLC SDU.

9) The Stop parameter indicates that the RLC entity shall not transmit or receive RLC PDUs.

10) The parameter Use special LI indicates that the LI indicating that a RLC SDU begins in the beginning of a RLC PDU (the first data octet of the PDU is the first octet of an SDU) shall be used. If the RLC SDU does not begin in the beginning of the RLC PDU, or if the LI indicating that a SDU ended exactly in the end of the previous RLC PDU is present, the LI shall not be used.

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.

Type = **ACK**
LSN

Figure 9.9: The ACK fields in a STATUS PDU

LSN

Length: 12 bits

Acknowledges the reception of all PUs with sequence numbers < 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 different value than VR(R) all erroneous PUs must be included in the same STATUS PDU and if the LSN is set to VR(R) the erroneous PUs 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 should not be set to a value > VR(H).

11.6 SDU discard with explicit signalling procedure

11.6.1 Purpose

An SDU can be discarded with explicit signalling when MaxDAT number of retransmissions is reached or the transmission time exceeds a predefined value (Timer_Discard) for a SDU in acknowledged mode RLC. Move Receiving Window (MRW) command is sent to the receiver so that AMD PDUs carrying that SDU are discarded in the receiver and the receiver window is updated accordingly. Note that when the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. If one or more segments of a SDU has been transmitted, the SDU shall not be discarded in the transmitter without notification to the receiver.

The MRW command is defined as a super-field in the RLC STATUS PDU, and can be piggybacked to status information of transmissions in the opposite direction.

Figure 11.6 below illustrates the elementary procedure for SDU discard with explicit signalling. The sender is the sender of AMD PDUs and it is either the UE or the network and the receiver is the receiver of AMD PDUs and it is either the network or the UE.

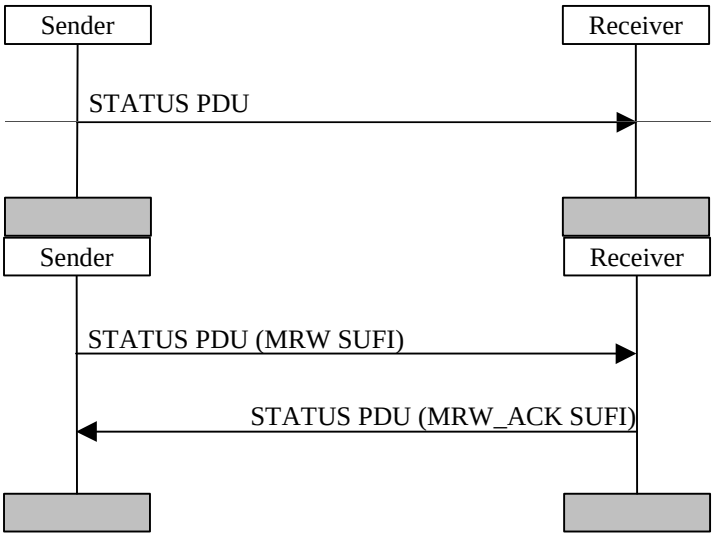


Figure 11.6: SDU discard with explicit signalling

11.6.2 Initiation

This procedure is initiated by the sender when the following conditions are fulfilled:

1) SDU discard with explicit signalling is used.

2) MaxDAT number of retransmissions is reached or Timer_Discard expires for a SDU in acknowledged mode RLC.

The sender shall discard all PUs that contain segments of the associated SDUs. If the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded. VT(A) shall be updated when the procedure is terminated, and VT(S) shall be updated when a new MRW SUFI is transmitted.

The sender shall transmit a status report 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.

This status report is sent even if the "STATUS prohibit" is used and the timer "Timer_Status_Prohibit" or "Timer_EPC" is active.

The STATUS PDUs have higher priority than data PDUs.

The sender shall start timer Timer_MRW. If a new SDU discard procedure is triggered when Timer_MRW is running, no new MRW SUFIs shall be sent before the current SDU discard procedure is terminated by one of the termination criteria.

11.6.2.1 Piggybacked STATUS PDU

It is possible to piggyback a STATUS PDU on an AMD PDU. If a PDU includes padding a piggybacked STATUS PDU can be inserted instead of the padding.

11.6.2.2 STATUS PDU contents to set

The size of the STATUS PDU shall be equal to one of the allowed PDU sizes. The discard information shall not be split into several MRW SUFIs.

The status report shall include the MRW SUFI, other SUFI fields can be used additionally. MRW SUFI shall convey information about the discarded SDU(s) to the receiver.

In order to discard a single SDU that ends in a PDU with SN≥ VT(A)+Configured_Tx_Window_Size, the LENGTH field in the MRW SUFI shall be set to "0000". If more then one SDU are discarded with the same MRW SUFI, at least the first discarded SDUs must end (i.e. the LI must be located) in a PDU with SN in the interval VT(A)≤ SN <VT(A)+Configured_Tx_Window_Size.

Padding shall be inserted if the SUFI fields do not fill the entire STATUS PDU. If the STATUS PDU contains padding the last SUFI field shall be either an ACK SUFI or a NO_MORE SUFI. If there is no padding in the STATUS PDU, NO_MORE SUFI or ACK SUFI does not need to be included in the STATUS PDU.

11.6.3 Reception of the STATUS PDU by the receiver

The receiver shall upon reception of the STATUS PDU/piggybacked STATUS PDU discard PUs and update the state variables VR(R), VR(H) and VR(MR) according to the received STATUS PDU/piggybacked STATUS PDU. Additionally the receiver should indicate the higher layers of all of the discarded SDUs.

The receiver shall initiate the transmission of a status report containing an MRW_ACK SUFI.

In the MRW_ACK SUFI, SN_ACK shall be set to the new value of VR(R), updated after reception of the MRW SUFI. The N field in the MRW_ACK SUFI shall be set to N_{LENGTH} field in the received MRW SUFI if the SN_ACK field is equal to SN_MRW_{LENGTH}. Otherwise N shall be set to 0.

The last discarded data byte is the byte indicated by the N_{LENGTH}:th LI field of the PU with sequence number SN_MRW_{LENGTH} and the succeeding data byte is the first data byte to be reassembled after the discard. When N_{LENGTH} = 0, the first data byte of the PU with sequence number SN_MRW_{LENGTH} is the first data byte to be reassembled after the discard.

If the MRW SUFI indicates an SN_MRW_i outside the interval VR(R)≤ SN_MRW_i <VR(MR) , the Rx shall consider the sequence number to be below VR(R), unless LENGTH="0000" or at least the first indicated SN_MRW_i in the MRW SUFI is within the interval VR(R)≤ SN_MRW_i <VR(MR), in which case the sequence number shall be considered to be above or equal to VR(MR).

11.6.4 Termination

The procedure is terminated in the sender in the following cases:

1. On the reception of a STATUS PDU which contains an MRW_ACK SUFI with SN_ACK > SN_MRW_{LENGTH} and the N field is equal to zero.
2. On the reception of a STATUS PDU which contains an ACK SUFI indicating VR(R) > SN_MRW_{LENGTH}
3. On reception of a STATUS PDU which contains an MRW_ACK with SN_ACK = SN_MRW_{LENGTH} and N is equal to the N_{LENGTH} indicated in the transmitted MRW SUFI.

If one of the termination criteria above is fulfilled, Timer_MRW is stopped and the discard procedure is terminated.

When VT(MRW) reaches MaxMRW, the procedure is terminated and an RLC reset is performed.

11.6.5 Expiration of timer Timer_MRW

If Timer_MRW expires before the discard procedure is terminated,the MRW SUFI shall be retransmitted, VT(MRW) is incremented by one and Timer_MRW restarted. MRW SUFI shall be exactly the same as previously transmitted even though some new SDUs would have been discarded during the running of the Timer_MRW. If the retransmitted STATUS PDU contains other SUFIs than the MRW SUFI, the status information indicated by these SUFIs shall be updated.

11.6.6 Abnormal cases

11.6.6.1 Obsolete/corrupted MRW command

If the MRW command contains outdated information about the receiver window (receiver window already moved further than MRW command is indicating), the MRW command shall be discarded and a status report containing SUFI MRW_ACK shall be transmitted indicating the value of VR(R) and the N field shall be set to zero.

11.6.6.2 VT(MRW) equals MaxMRW

If the number of retransmission of a MRW command (i.e. VT(MRW)) reaches MaxMRW, an error indication shall be passed to RRC and RESET procedure shall be performed.

11.6.6.3 Reception of obsolete MRW_ACK

The received MRW_ACK shall be discarded in the following cases.

1. If timer Timer_MRW is not active.
2. If the SN_ACK field in the received MRW_ACK < SN_MRW_{LENGTH} in the transmitted MRW SUFI.
3. ~~3.~~ If the SN_ACK field in the received MRW_ACK is equal to the SN_MRW_{LENGTH} in the transmitted MRW SUFI and the N field in the received MRW_ACK ~~field~~ is not equal to the N_{LENGTH} field in the transmitted MRW SUFI.
4. If the SN_ACK field in the received MRW_ACK > SN_MRW_{LENGTH} in the transmitted MRW SUFI and the N field in the received MRW_ACK is not equal to zero.

3GPP TSG RAN WG2 Meeting #17
Sophia Antipolis, France, 13-17 November 2000

Document ***R2-002408***

e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 090r2 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X] for information [ ]

strategic [ ] non-strategic [ ] *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 16.11.2000

Subject: RLC UM protocol

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	[X]	**Release:** Phase 2	[ ]
A	Corresponds to a correction in an earlier release	[ ]	Release 96	[ ]
B	Addition of feature	[ ]	Release 97	[ ]
C	Functional modification of feature	[ ]	Release 98	[ ]
D	Editorial modification	[ ]	Release 99	[X]
			Release 00	[ ]

Reason for change:

1 - The current RLC UM model does not include the fact that multiple PDU size might be used. Some modifications are needed on :
- the model for the UMD PDU delivery to MAC,
- the Discard procedure without explicit signalling.

2. editorial correction about LI field size choice.

Revision R1:

- Figure 3.4 modification according to tdoc R2_002087 (CR084r1 on 25.322 : Model of UM and AM entities)

- Editorial corrections and clarifications

Revision R2 :

Remove modification on section 4.2.1.2. Modification is transferred to CR 084r2

Clauses affected: ~~4.2.1.2~~, 9.2.2.8, 11.2.2, 11.2.4.3

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

<--------- double-click here for help and instructions on how to create a CR.

9.2.2.8 Length Indicator (LI)

The Length Indicator is used to indicate, each time, the end of an SDU occurs in the PU. The Length Indicator points out the number of octets between the end of the last Length Indicator field and up to and including the octet at the end of an SDU segment. Length Indicators are included in the PUs that they refer to. The size of the Length Indicator may be either 7bits or 15bits. The maximum value of a Length Indicator in AM will be no greater than the RLC PDU size – AMD PDU Header – PADDING. The maximum value of a Length Indicator in UM will be no greater than the RLC PDU size – UMD PDU Header – PADDING.

A Length Indicator group is a set of Length Indicators that refer to a PU. Length Indicators that are part of a Length Indicator group must never be reordered within the Length Indicator group or removed from the Length Indicator group.

If there can be more than one Length Indicator, each specifying the end of an SDU in a PU, the order of these Length Indicators must be in the same order as the SDUs that they refer to.

In the case where the end of last segment of an SDU exactly ends at the end of a PDU and there is no LI that indicates the end of the SDU, the next Length Indicator, shall be placed as the first Length Indicator in the following PU and have value LI=0.

In the case where the last segment of an RLC SDU is one octet short of exactly filling the previous RLC PU, and 15-bit Length Indicators are used, the next Length Indicator shall be placed as the first Length Indicator in the following PU and have value LI=111 1111 1111 1011. The remaining one octet in the previous RLC PU shall be ignored.

A PU that has unused space, to be referred to as padding, must use a Length Indicator to indicate that this space is used as padding. A padding Length Indicator must be placed after any Length Indicators for a PU.

All unused space in a PU must be located at the end of the PDU, be a homogeneous space and is referred to as padding. Predefined values of the Length Indicator are used to indicate this. 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.

STATUS PDUs can be piggybacked on the AMD PDU by using part or all of the padding space. A Length Indicator must be used to indicate the piggybacked STATUS PDU. This Length Indicator takes space from the padding space or piggybacked STATUS PDU and not the PDU data and will always be the last Length Indicator. Where only part of the padding space is used by a piggybacked STATUS PDU then the end of the piggybacked STATUS PDU is determined 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 PDU. The padding/piggybacked STATUS PDU predefined Length Indicators shall be added after the very last (i.e. there could be more than one SDU that end within a PDU) Length Indicator that indicates the end of the last SDU segment in the PU.

If SDU discard with explicit signalling is used an AMD PDU can contain a maximum number of 15 LIs indicating the end of an SDU and the rest of the AMD PDU space shall be used as padding/piggybacked STATUS PDU.

For AM, 7bit indicators shall be used if the AMD PDU size is ≤ l26 octets. Otherwise 15bit indicators shall be used. For UM, 7bit indicators shall be used if the UMD PDU size is ≤ l25 octets. Otherwise 15bit indicators shall be used.

The length of the Length Indicator only depends on the size of the largest RLC PDU. The length of the Length Indicator is always the same for all UMD PDUs or AMD PUs, for one RLC entity.

For Release 99, there is one PU in an AMD PDU.

Length: 7bit

Bit	Description
0000000	The previous RLC PDU was exactly filled with the last segment of a RLC SDU and there is no LI that indicates the end of the SDU in the previous RLC PDU.
1111100	UMD PDU: The first data octet in this RLC PDU is the first octet of a 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: 15bit

Bit	Description
000000000000000	The previous RLC PDU was exactly filled with the last segment of an RLC SDU and there is no LI that indicates the end of the 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. 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 a 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.

11.2 Unacknowledged mode data transfer procedure

11.2.1 Purpose

The unacknowledged mode data transfer procedure is used for transferring data between two RLC peer entities, which are operating in unacknowledged mode. Figure 11.2 below illustrates the elementary procedure for unacknowledged mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.

Figure 11.2: Unacknowledged mode data transfer procedure

11.2.2 Initiation

The sender initiates this procedure upon a request of unacknowledged mode data transfer from higher layer.

When the sender is in data transfer ready state it shall segment the data received from the higher layer into PDUs.

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) ~~and~~. For each TTI, MAC decides which PDU sizes shall be used and how many PDUs shall be transmitted ~~in each TTI. In the UE, the PDUs that can not be transmitted in a TTI (i.e. MAC has indicated that~~

~~some of the available PDUs can not be transmitted) shall be buffered according to the discard configuration set by RRC.~~

The VT(US) state variable shall be updated for each UMD PDU that is transmitted.

11.2.2.1 UMD PDU contents to set

The Sequence Number field shall be set equal to VT(US).

The Extension bit shall be set to 1 if the next field is a length indicator field, otherwise it shall be set to zero.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number octets between the end of the header fields and the end of the segment. If padding is needed another length indicator shall be added. If the PDU is exactly filled with the last segment of a SDU and there is no room for a length indicator field a length indicator field set to only 0's shall be included in the next PDU.

11.2.3 Reception of UMD PDU

Upon reception of a UMD PDU the receiver shall update VR(US) state variable according to the received PDU(s).

The PDUs are reassembled into RLC SDUs. If a PDU with sequence number < VR(US) is missing then all SDUs that have segments in this PDU shall be discarded. RLC delivers the RLC SDUs to the higher layer through the UM-SAP.

11.2.4 Abnormal cases

11.2.4.1 Length Indicator value 1111110

Upon reception of an UMD PDU that contains Length Indicator value 1111110 or 111111111111110 ("piggybacked STATUS PDU", in case 7bit or 15 bit Length Indicator field is used, respectively) the receiver shall discard that UMD PDU. This Length Indicator value is not used in unacknowledged mode data transfer.

11.2.4.2 Invalid length indicator value

If the length indicator of a PDU has a value that is larger than the PDU size, the PDU shall be discarded and treated as a missing PDU.

11.2.4.3 SDU discard without explicit signalling

Upon expiry of the Timer_Discard on the sender side the sender shall discard ~~all PDUs that contain segments of~~ the associated SDU. ~~If the concatenation function is active, PDUs carrying segments of other SDUs that have not timed out shall not be discarded.~~ The next UMD PDU shall carry the first segment of the oldest SDU not discarded. The state variable VT(US) shall be updated so that the receiver can~~shall~~ detect at least one~~some~~ missing PDUs. To avoid that the receiver should discard one extra SDU, a LI field shall be added in the first PDU transmitted after a Discard Operation. The value of the LI field shall be either the value indicating that the previous SDU filled exactly the previous RLC PDU or the value indicating that the first data octet in this RLC PDU is the first octet of a RLC SDU.

3GPP TSG-RAN WG2 Meeting #17
Sophia Antipolis, France, 13-17 November 2000

Document R2-002400
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 092r2 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X]
for information [ ]

strategic [ ]
non-strategic [ ]
(for SMG use only)

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-11-8

Subject: Clarification to window size parameters, ~~and~~ MRW SUFI and window based polling

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change:

1. Both the maximum allowed transmitter window size signalled by RRC and the current transmitter window size decided by the receiver use the same variable name, Tx_Window_Size. Clarification between these two is needed.
2. Configured_Tx_Window_Size used in Subclause 11.6.2.2 is not defined and need to be clarified.
3. The format of MRW SUFI need to be clarified.
4. VT(MS) need to be clarified.
5. Minor editorial corrections in Clause 3 and Subclause 11.3.2 are needed.
6. Tx_Window_Size is defined as a state variable.
7. CR091r1 is combined in this CR to make the change of Tx_Window_Size more self-contained.

Clauses affected: 3, 9.2.2.11.3, 9.2.2.11.8, 9.4, 9.6, 9.7.1, 11.3.2, 11.3.2.1.1

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

3 Abbreviations

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

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
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
MRW	Move ~~Receive~~ Receiving Window
MSB	Most Significant Bit
PCCH	Paging Control Channel
PCH	Paging Channel
PDU	Protocol Data Unit
PU	Payload 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
TTI	Transmission Time Interval
U-	User-
UE	User Equipment
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network

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 Tx window size of the peer entity during a connection, but the minimum and the maximum allowed value is given by RRC configuration. The Rx window 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)~~allowed Tx window size~~ to be used by the transmitter. The range of the WSN is [0, ~~2~~^{12}~~-1~~~~Configured_Tx_Window_Size~~2^{12}-1]. The minimum value of ~~the window size~~ VT(WS) is 1, if WSN is zero the SUFI shall be discarded by this version of the protocol.~~–~~ ~~The value of Tx_Window_Size parameter~~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.

9.2.2.11.8 The Move Receiving Window (MRW) super-field

The 'Move Receiving Window' super-field is used to request the RLC receiver to move its receiving window and to indicate the amount of discarded SDUs, as a result of a SDU discard in the RLC transmitter. The format is given in the figure 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 discarded SDU extends above the Tx window in the transmitter.

SN_MRW_i

Length: 12 bits

SN_MRW_i is used to indicate the end of each discarded SDU. SN_MRW_i is the sequence number of the PU that contains the LI of the i:th discarded SDU (except when $N_{LENGTH} = 0$, see definition of N_{LENGTH}).

Additionally SN_MRW_{LENGTH} requests the RLC receiver to discard all PUs with sequence number < SN_MRW_{LENGTH}, and to move the receiving window accordingly. In addition, the receiver has to discard the first N_{LENGTH} LIs and the corresponding data bytes in the PU 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 discarded SDU.

N_{LENGTH} indicates which LI in the PU with sequence number SN_MRW_{LENGTH} corresponds to the last discarded SDU. $N_{LENGTH} = 0$ indicates that the last SDU ended in the PU with sequence number $SN\_MRW_{LENGTH} - 1$ and that the first data byte in the PU with sequence number SN_MRW_{LENGTH} is the first data byte to be reassembled next.

9.4 State variables

This sub-clause describes the state variables used in the specification of the peer-to-peer protocol. PUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{12} for AM and 2^{7} for UM; the sequence numbers cycle through the entire range: 0 through $2^{12} - 1$ for AM and 0 through $2^{7} - 1$ for UM. All arithmetic operations on the following state variables and sequence numbers contained in this specification are affected by the modulus: VT(S), VT(A), VT(MS), VR(R), VR(H), VR(MR), VT(US) and VR(US). When performing arithmetic comparisons of transmitter variables, VT(A) is assumed to be the base. When performing arithmetic comparisons of receiver variables, VR(R) is assumed to be the base.

The RLC maintains the following state variables at the transmitter.

a) VT(S) - Send state variable.

   The sequence number of the next PU to be transmitted for the first time (i.e. excluding retransmission). It is updated after transmission of a PDU, which includes not earlier transmitted PUs and after transmission of a MRW SUFI. The initial value of this variable is 0.

b) VT(A) - Acknowledge state variable.

   The sequence number of the next in-sequence PU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgements. VT(A) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK super-field. The initial value of this variable is 0.

c) VT(DAT).

   This state variable counts the number of times a PU has been transmitted. There is one VT(DAT) for each PU and it is incremented each time the PU is transmitted. The initial value of this variable is 0.

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

   The sequence number of the first PU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1], VT(MS) = VT(A) + ~~Tx_Window_Size~~VT(WS). This value represents the upper edge of the transmit window. The transmitter shall not transmit a new PU if VT(S) $\geq$ VT(MS). ~~The initial value of Tx_Window_Size is Configured_Tx_Window_Size. Tx_Window_Size is updated based on receipt of a STATUS PDU including a WINDOW super-field.~~ VT(MS) is updated when either VT(A) or ~~TX_Window_Size~~VT(WS) is updated. ~~based on receipt of a STATUS PDU including an ACK and/or MRW_ACK and/or a WINDOW super-field.~~ The PU with SN $\leq$ VT(S)-1 can be transmitted also when VT(S)>VT(MS).

e) VT(US) – UM data state variable.

   This state variable gives the sequence number of the next UMD PDU to be transmitted. It is updated each time a UMD PDU is transmitted. The initial value of this variable is 0.

f) VT(PU).

   This state variable is used when the poll every Poll_PU PU function is used. It is incremented with 1 for each PU that is transmitted. It should be incremented for both new and retransmitted PUs. When it reaches Poll_PU a new poll is transmitted and the state variable is 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 function is used. It is incremented with 1 for each SDU that is transmitted. When it reaches Poll_SDU a new poll is transmitted and the state variable is set to zero. The poll bit should be set in the PU that contains the last segment of the SDU. The initial value of this variable is 0.

h) VT(RST) - Reset state variable.

   It is used to count the number of times a RESET PDU is transmitted. VT(RST) is incremented with 1 each time a RESET PDU is transmitted. VT(RST) is reset only upon the reception of a RESET ACK PDU, i.e. VT(RST) is not reset when a RLC reset occurs which was initiated from the peer RLC entity. The initial value of this variable is 0.

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

It is used to count the number of times a MRW command is transmitted. VT(MRW) is incremented with 1 each time an MRW command is transmitted. VT(MRW) is reset when the discard procedure is terminated. The initial value of this variable is 0.

j) VT(WS) – Transmitter window size state ~~v~~ variable.

The size that shall be used for the transmitter window. VT(WS) is set equal to the WSN field when the transmitter receives a STATUS PDU including a Window Size super-field. The initial value of this variable is Configured_Tx_Window_size.

The RLC maintains the following state variables at the receiver:

a) VR(R) - Receive state variable.

The sequence number of the next in-sequence PU expected to be received. It is set equal to SNmax+1 upon receipt of the next in-sequence PU, where SNmax is the sequence number of the highest received in-sequence PU. The initial value of this variable is 0.

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

The sequence number of the highest expected PU. This state variable is set equal to SN+1 only when a new PU is received with VR(MR)>SN≥VR(H). The initial value of this variable is 0.

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

The sequence number of the first PU not allowed by the receiver [i.e. the receiver will allow up to VR(MR) – 1], VR(MR) = VR(R) + Configured_Rx_Window_Size. The receiver shall discard PUs with SN $\geq$ VR(MR).

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

The sequence number of the next PDU to be received. It shall set equal to SN + 1 upon reception of a PDU. The initial value of this variable is 0.

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

The number of PUs that should be received yet as a consequence of the transmission of the latest status report. In acknowledged mode, this state variable is updated at the end of each transmission time interval. It is decremented by the number of PUs that should have been received during the transmission time interval. If VR(EP) is equal to zero, then check if all PUs requested for retransmission in the latest status report have been received.

9.6 Protocol Parameters

The values of the protocol parameters in this section are signalled by RRC.

a) MaxDAT.

It is the maximum value for the number of retransmissions of a PU. This parameter is an upper limit of counter VT(DAT). When the value of VT(DAT) comes to MaxDAT, error recovery procedure will be performed.

b) Poll_PU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_PU PU. This is an upper limit for the VT(PU) state variable, when VT(PU) reaches Poll_PU a poll is transmitted to the peer entity.

c) Poll_SDU.

This parameter indicates how often the transmitter should poll the receiver in case of polling every Poll_SDU SDU. This is an upper limit for the VT(SDU) state variable, when VT(SDU) reaches Poll_SDU a poll is transmitted to the peer entity.

d) Poll_Window.

This parameter indicates when the transmitter should poll the receiver in case of performing window-based polling. The range of values of this parameter shall be $0 \leq$ Poll_Window ≤ 100. A poll is triggered for each PU when: J $\geq$ Poll_Window, where J is the window transmission percentage defined by

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

where the constant 4096 is the modulus for AM described in Subclause 9.4.

~~1) VT(S)<VT(MS), Tx_Window_Size>0, and~~

~~$$\left[1 - \frac{(Tx\_Window\_Size + VT(MS) - VT(S)-1) \bmod Tx\_Window\_Size}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$~~

~~2) VT(S)≥VT(MS), and Tx_Window_Size>0~~

~~When Tx_Window_Size=0, the transmitter does not perform window-based polling.~~

e) MaxRST.

It is the maximum value for the number of retransmission of RESET PDU. This parameter is an upper limit of counter VT(RST). When the value of VT(RST) comes to MaxRST, the higher layer (RRC) is notified.

f) Configured_Tx_Window_Size.

The maximum allowed transmitter window size.

g) Configured_Rx_Window_Size.

The ~~maximum~~ allowed receiver window size.

h) MaxMRW.

It is the maximum value for the number of retransmissions of a MRW command. This parameter is an upper limit of counter VT(MRW). When the value of VT(MRW) comes to MaxMRW, error recovery procedure will be performed.

9.7 Specific functions

9.7.1 Polling function for acknowledged mode transfer

The transmitter of AMD PDUs may poll the receiver for a status report (consisting of one or several STATUS PDUs). The Polling bit in the AMD PDU indicates the poll request. There are several triggers for setting the polling bit. The network (RRC) controls, which triggers should be used for each RLC entity. Following triggers are possible:

1) Last PU in buffer.

   The sender transmits a poll when the last PU available for transmission is transmitted.

2) Last PU in retransmission buffer.

   The sender transmits a poll when the last PU to be retransmitted is transmitted.

3) Poll timer.

   The timer Timer_Poll is started when a poll is transmitted to the receiver and if the criterion for stopping the timer has not occurred before the timer Timer_Poll expires a new poll is transmitted to the receiver.

4) Every Poll_PU PU.

   The sender polls the receiver every Poll_PU PU. Both retransmitted and new Pus shall be counted.

5) Every Poll_SDU SDU.

   The sender polls the receiver every Poll_SDU SDU.

6) ~~Poll_Window% of transmission window.~~Window based.

   The sender polls the receiver when it has reached Poll_Window% of the transmission window.

7) Timer based.

   The sender polls the receiver periodically.

Either the trigger "Last PU in buffer" and "Last PU in retransmission buffer" or "Timer based" can be chosen to avoid deadlock for every RLC entity. The network also controls if the poll prohibit function shall be used. The poll bit shall be set to 0 if the poll prohibit function is used and the timer Timer_Poll_Prohibit is active. This function has higher priority than any of the above mentioned triggers.

11.3 Acknowledged mode data transfer procedure

11.3.1 Purpose

The acknowledged mode data transfer procedure is used for transferring of data between two RLC peer entities, which are operating in acknowledged mode. Figure 11.3 below illustrates the elementary procedure for acknowledged mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.

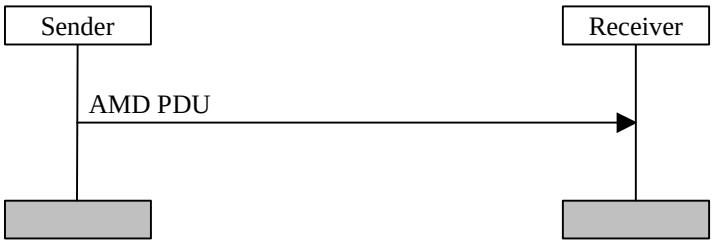


Figure 11.3: Acknowledged mode data transfer procedure

11.3.2 Initiation

The sender initiates this procedure upon a request of acknowledged mode data transfer from higher layer or upon retransmission of PUs. Retransmitted PUs have higher priority than PUs transmitted for the first time.

The sender is only allowed to retransmit PUs that have been indicated missing by the receiver. An exception is the PU with SN VT(S)-1 can be retransmitted. In addition, a PU that has not yet been acknowledged, may be retransmitted if ~~the configured transmitter window size~~Configured_Tx_Window_Size is less than 2048.

RLC shall segment the data received from the higher layer into PUs. When the sender is in data transfer ready state one or several PUs are included in one AMD PDU, which is sent to the receiver. The 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. In the UE, the PDUs that can not be transmitted in a TTI (i.e. MAC has indicated that some of the available PDUs can not be transmitted) shall be buffered according to the discard configuration set by RRC.

The VT(DAT) state variables shall be updated for each AMD PDU that is transmitted. The PDU shall not include any PU with Sequence Number ≥ VT(MS) , except the PU with sequence number VT(S)-1 which may be included also when VT(S) > VT(MS).

If the poll bit is set in any of the AMD PDUs and the timer Timer_Poll shall be used, the sender shall start the timer Timer_Poll when the PDU with the set poll bit is delivered to MAC.

If timer based SDU discard is used, the timer Timer_Discard shall be started when the RLC entity receives an SDU from higher layer. One timer is used for each SDU that is received from higher layer.

If the trigger for polling, "Every Poll_PU PU", is used, the VT(PU) shall be increased by 1 for each PU that is transmitted.

If the trigger for polling, "Every Poll_SDU SDU", is used, the VT(SDU) shall be increased by 1 for each SDU that is transmitted.

11.3.2.1.1 Setting of the Polling bit

The Polling bit shall be set to 1 if any of following conditions are fulfilled except when the poll prohibit function is used and the timer Timer_Poll_Prohibit is active (the different triggers are described in 9.7.14):

1) Last PU in buffer is used and the last PU available for transmission is transmitted.

2) Last PU in retransmission buffer is used and the last PU to be retransmitted is transmitted.

3) Poll timer is used and timer Timer_Poll has expired.

4) Every Poll_PU PU is used and when VT(PU)=Poll_PU.

5) Every Poll_SDU is used and VT(SDU)=Poll_SDU and the PDU contains the last segment of that SDU.

6) Window based pollingPoll_Window(%) of transmission window is used, Tx_Window_Size>0, VT(S)<VT(MS), and J≥Poll_Window, where J is defined in subclause 9.6.

$$\left\lceil 1 - \frac{(Tx\_Window\_Size + VT(MS) - VT(S)-1) \bmod Tx\_Window\_Size}{Tx\_Window\_Size} \right\rceil * 100 > Poll\_Window$$

.

7) Poll_Window (%) of transmission window is used, Tx_Window_Size>0 and VT(S)≥VT(MS).

87) Timer based polling is used and Timer_Poll_Periodic has expired.

98) Poll prohibit shall be used, the timer Timer_Poll_Prohibit has expired and one or several polls were prohibited during the time Timer_Poll_Prohibit was active.

3GPP TSG-RAN WG2 Meeting #17
Sophia Antipolis, France, 13-17 November

Tdoc R2-002396

CR-Form-v3

CHANGE REQUEST

⌘ **25.322** **CR** **093** ⌘ rev **r3** ⌘ Current version: **3.4.0** ⌘

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

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☒ Radio Access Network ☐ Core Network ☐

Title:	⌘	General RLC Corrections
Source:	⌘	TSG-RAN WG2
Work item code:	⌘	***Date:*** ⌘ 2000-11-16
Category:	⌘	**F** ***Release:*** ⌘ R99

Use one of the following categories:	*Use one of the following releases:*	
F *(essential correction)*	*2*	*(GSM Phase 2)*
A *(corresponds to a correction in an earlier release)*	*R96*	*(Release 1996)*
B *(Addition of feature),*	*R97*	*(Release 1997)*
C *(Functional modification of feature)*	*R98*	*(Release 1998)*
D *(Editorial modification)*	*R99*	*(Release 1999)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.	*REL-4*	*(Release 4)*
	REL-5	*(Release 5)*

Reason for change: ⌘

Summary of change: ⌘

1. The segmentation option is removed from PCCH
2. A continue parameter is added in the CRLC-CONFIG-Req primitive
3. A new section is added to define the stop and continue actions

4 Updates in r1: clarification that AMD PDU size shall be equal to control PDU size when one logical channel is used for an AM entity.

5. It is clarified that PDUs containing only padding shall not be transmitted.

Updates in r2:

6. The format of the STATUS PDU is clarified

7. The maximum RLC PDU size is clarifed

Updates in r3: clarifications that padding PDUs are not allowed is removed after discussions in L2 adhoc.

Consequences if not approved: ⌘ Inconsistency between specs. Unclear functionality.

Clauses affected: ⌘ 4.2.1.3, 6.1, 8.1, 8.2, 9.2.1.5, 9.2.1.6, 9.2.2.8, 9.4, 9.7.x (new), 11.2.2.1, 11.3.2.1, 11.5.2.2

Other specs affected: ⌘

- ☐ Other core specifications ⌘
- ☐ Test specifications
- ☐ 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/3G_Specs/CRs.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://www.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2000-09 contains the specifications resulting from the September 2000 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.

4.2.1.3 Acknowledged mode entity

Figure 4.4 below shows the model of an acknowledged mode entity, when one logical channel (shown as a solid line) and when two logical channels (shown as dashed lines) are used.

In case two logical channels are used in the uplink ~~the UTRAN can indicate that~~ the first logical channel shall be used for data PDUs and the second logical channel shall be used for control PDUs. ~~If the indication is not given from the UTRAN, data and control PDUs can be sent on either of the two logical channels. The indication of the logical channel mapping is signalled by RRC.~~ In case one logical channel is used, the RLC PDU size shall be the same for AMD PDUs and control PDUs.

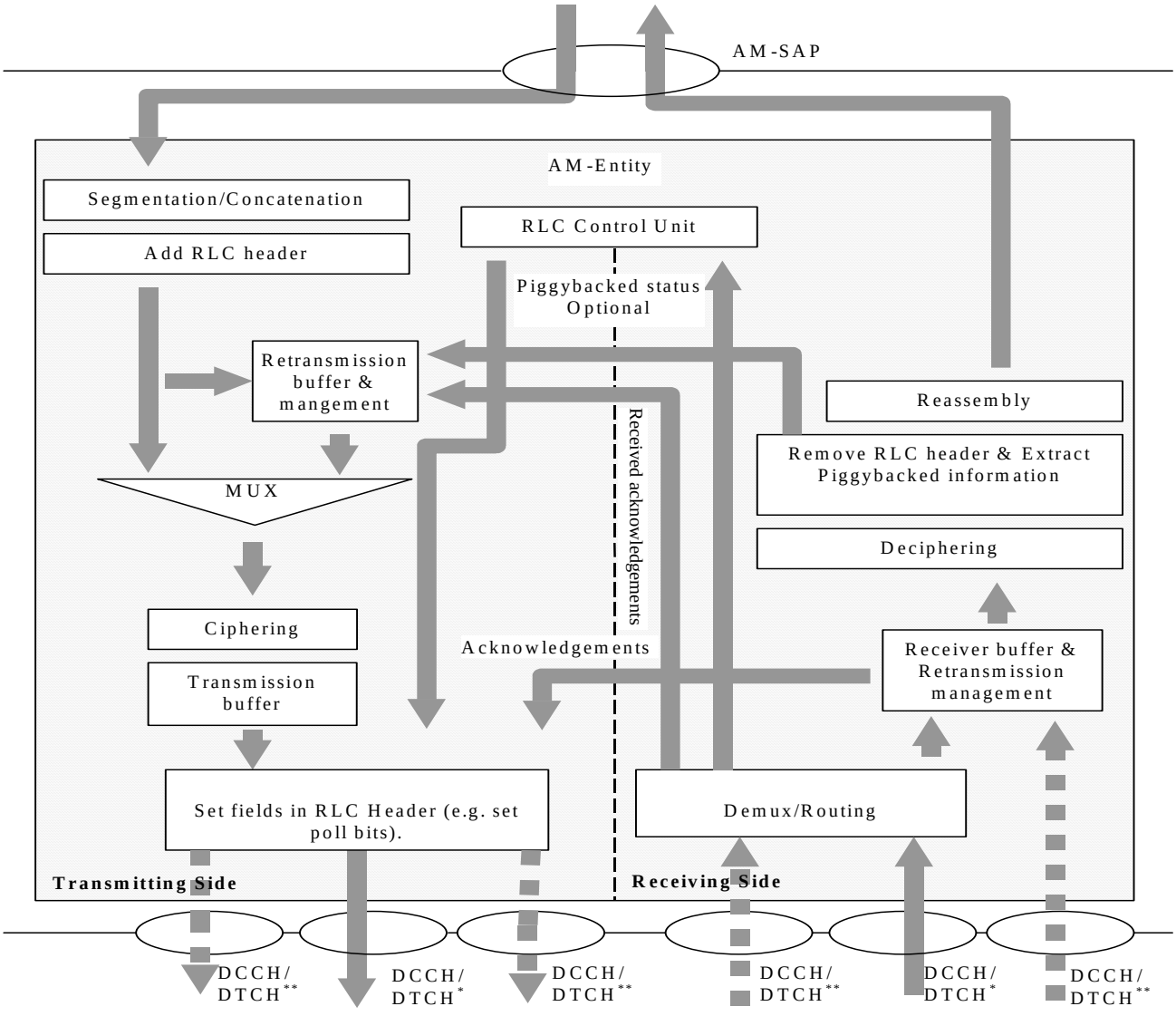


Figure 4.4: Model of a acknowledged mode entity

The transmitting side of the AM-entity receives SDUs from the higher layers. The SDUs are segmented and/or concatenated to PUs of fixed length. PU length is a semi-static value that is decided in bearer setup and can only be changed through bearer reconfiguration by RRC.

For purposes of RLC buffering and retransmission handling, the operation is the same as if there would be one PU per PDU. For concatenation or padding purposes, bits of information on the length and extension are inserted into the beginning of the last PU where data from an SDU is included. Padding can be replaced by piggybacked status information. This includes setting the poll bit.

If several SDUs fit into one PU, they are concatenated and the appropriate length indicators are inserted into the beginning of the PU. After that the PUs are placed in the retransmission buffer and the transmission buffer. One PU is included in one RLC PDU.

The MUX then decides which PDUs and when the PDUs are delivered to MAC. The PDUs are delivered via a function that completes the RLC-PDU header. The fixed 2 octet AMD PDU header is not ciphered.

When Piggybacking mechanism is applied the padding is replaced by control information, in order to increase the transmission efficiency and making possible a faster message exchange between the peer to peer RLC entities. The piggybacked control information is not saved in any retransmission buffer. The piggybacked control information is contained in the piggybacked STATUS PDU, which is in turn included into the AMD-PDU. The piggybacked STATUS PDUs will be of variable size in order to match with the amount of free space in the AMD PDU.

The retransmission buffer also receives acknowledgements from the receiving side, which are used to indicate retransmissions of PUs and when to delete a PU from the retransmission buffer.

The Receiving Side of the AM-entity receives PDUs through one of the logical channels from the MAC sublayer. The RLC-PDUs are expanded into separate PUs and potential piggybacked status information are extracted. The PUs are placed in the receiver buffer until a complete SDU has been received. The receiver buffer requests retransmissions of PUs by sending negative acknowledgements to the peer entity. After that the headers are removed from the PDUs and the PDUs are reassembled into a SDU. Finally the SDU is delivered to the higher layer. The receiving side also receives acknowledgements from the peer entity. The acknowledgements are passed to the retransmission buffer on the transmitting side.

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	+	+	+	+
Unacknowledged Service	Applicability	-	-	+	+
	Segmentation	-	-	+	+
	Concatenation	-	-	+	+
	Padding	-	-	+	+
	Transfer of user data	-	-	+	+
	Ciphering	-	-	+	+
Acknowledged Service	Applicability	-	-	+	+
	Segmentation	-	-	+	+
	Concatenation	-	-	+	+
	Padding	-	-	+	+
	Transfer of user data	-	-	+	+
	Flow Control	-	-	+	+
	Error Correction	-	-	+	+
	Protocol error correction & recovery	-	-	+	+
	Ciphering	-	-	+	+

Table 6.2: RLC modes and functions in UE downlink side

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

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	+	+	-	-	+	+	-
Unacknowledged Service	Applicability	-	-	+	+	+	+	+
	Segmentation	-	-	+	+	+	+	+
	Concatenation	-	-	+	+	+	+	+
	Padding	-	-	+	+	+	+	+
	Ciphering	-	-	-	-	+	+	-
	Transfer of user data	-	-	+	+	+	+	+
Acknowledged Service	Applicability	-	-	-	-	+	+	-
	Segmentation	-	-	-	-	+	+	-
	Concatenation	-	-	-	-	+	+	-
	Padding	-	-	-	-	+	+	-
	Transfer of user data	-	-	-	-	+	+	-
	Flow Control	-	-	-	-	+	+	-
	Error Correction	-	-	-	-	+	+	-
	Protocol error correction & recovery	-	-	-	-	+	+	-
	Ciphering	-	-	-	-	+	+	-

Table 6.4: RLC modes and functions in UTRAN uplink side

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

7 Services expected from MAC

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

- Data transfer.

8 Elements for layer-to-layer communication

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

8.1 Primitives between RLC and higher 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	Parameter			
	Req.	**Ind.**	**Resp.**	**Conf.**
RLC-AM-DATA	Data, CNF, MUI	Data, DiscardInfo	Not Defined	MUI
RLC-UM-DATA	Data, Use special LI	Data	Not Defined	Not Defined
RLC-TR-DATA	Data	Data	Not Defined	Not Defined
CRLC-CONFIG	E/R, Stop,Continue, Ciphering Elements (UM/AM 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 higher layers to request transmission of a higher layer PDU in acknowledged mode.
- RLC-AM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in acknowledged mode and to indicate higher layers of the discarded RLC SDU in the receiving RLC.
- RLC-AM-DATA-Conf is used by RLC to confirm to higher layers the transmission of a RLC SDU.

RLC-UM-DATA-Req/Ind

- RLC-UM-DATA-Req is used by higher layers to request transmission of a higher layer PDU in unacknowledged mode.
- RLC-UM-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in unacknowledged mode.

RLC-TR-DATA-Req/Ind

- RLC-TR-DATA-Req is used by higher layers to request transmission of a higher layer PDU in transparent mode.
- RLC-TR-DATA-Ind is used by RLC to deliver to higher layers RLC SDUs, that have been transmitted in transparent mode.

CRLC-CONFIG-Req

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

CRLC-SUSPEND-Req/Conf

This primitive is used by RRC to suspend the RLC. The N parameter indicates that RLC shall not send a PDU with SN>=VT(S)+N, where N is an integer. RLC informs RRC of the VT(S) value in the confirm primitive.

CRLC-RESUME-Req

This primitive is used by RRC to resume RLC when RLC has been suspended.

CRLC-STATUS-Ind

It is used by the RLC to send status information to RRC.

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. The Data parameter may be divided over several RLC PDUs. In case of a RLC-AM-DATA or a RLC-UM-DATA primitive the length of the Data parameter shall be octet-aligned.

2) The parameter Confirmation request (CNF) indicates whether the RLC needs to confirm the correct transmission of the RLC SDU.

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.

4) The parameter E/R indicates (re)establishment, release or modification of RLC If it indicates establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value and RLC shall enter the data transfer ready state. If it indicates re-establishment, the state variables in 9.4 shall be set to their initial value, the configurable parameters shall be set to their configured value the RLC buffers shall be emptied and RLC shall enter the data transfer ready state. If it indicates release, all protocol parameters, variables and timers shall be released and RLC shall exit the data transfer ready state. If it indicates modification, the protocol parameters indicated by RRC (e.g. ciphering parameters) shall only be modified with keeping the other protocol parameters, the protocol variables, the protocol timers and the protocol state. RLC shall always be re-established if the PU size is changed.

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

6) The parameter ciphering elements are only applicable for UM and AM operation. These parameters are Ciphering Mode, Ciphering Key, Activation Time (SN to activate a new ciphering configuration) and HFN (Hyper Frame Number).

7) The AM_parameters are only applicable for AM operation. It contains PU size, 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), SDU discard mode (see subclause 9.7.3) and Minimum WSN (see subclause 9.2.2.11.3). The Minimum WSN shall always be greater than or equal to the number of transport blocks in the smallest transport block set.

8) The parameter DiscardInfo indicates the upper layer of each of the discarded RLC SDU. It is applicable only when in-sequence delivery is active and it is purposed to be used when the upper layer requires the reliable data transfer and especially the information of the discarded RLC SDU.

9) The Stop parameter indicates that the RLC entity shall not transmit or receive RLC PDUs. The Continue parameter indicates that the RLC entity shall continue transmission and reception of RLC PDUs.

10) The parameter Use special LI indicates that the LI indicating that a RLC SDU begins in the beginning of a RLC PDU (the first data octet of the PDU is the first octet of an SDU) shall be used. If the RLC SDU does not begin in the beginning of the RLC PDU, or if the LI indicating that a SDU ended exactly in the end of the previous RLC PDU is present, the LI shall not be used.

9.2.1.5 STATUS PDU

The STATUS PDU is used to report the status between two RLC AM entities. Both receiver and transmitter status information may be included in the same STATUS PDU.

The format of the STATUS PDU is given in Figure 9.4 below. The Figure shows an example and the length of each SUFI is dependent on the SUFI type.

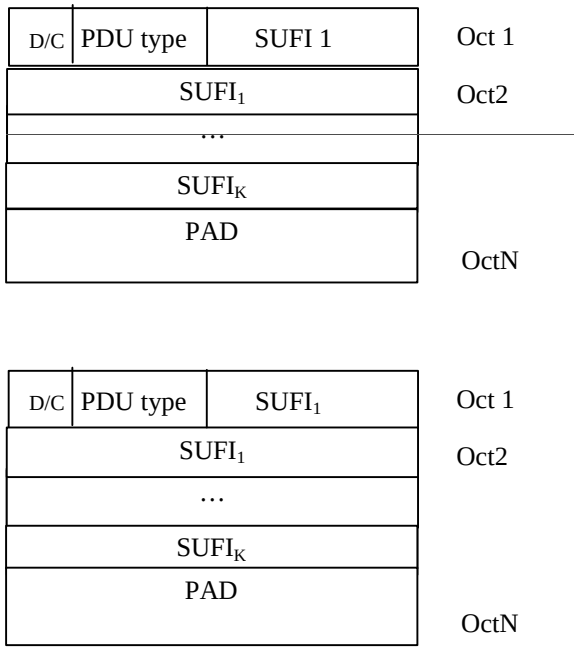


Figure 9.4: Status Information Control PDU (STATUS PDU)

Up to K ~~different~~ super-fields ($SUFI_1$-$SUFI_K$) can be included into one STATUS PDU, in which each super-field can be of different type. The size of a STATUS PDU is variable and upper bounded by the maximum RLC PDU size used by an RLC entity. Padding shall be included to exactly fit one of the PDU sizes used by the entity. The length of the STATUS PDU shall be an integer number of octets.

9.2.1.6 Piggybacked STATUS PDU

The format of the piggybacked STATUS PDU is the same as the ordinary Control PDU except that the D/C field is replaced by a reserved bit (R). This PDU can be used to piggyback STATUS PDU in an AMD PDU if the data does not fill the complete AMD PDU. The PDU Type field is set to zero and all other values are invalid for this version of the protocol and the PDU is discarded.

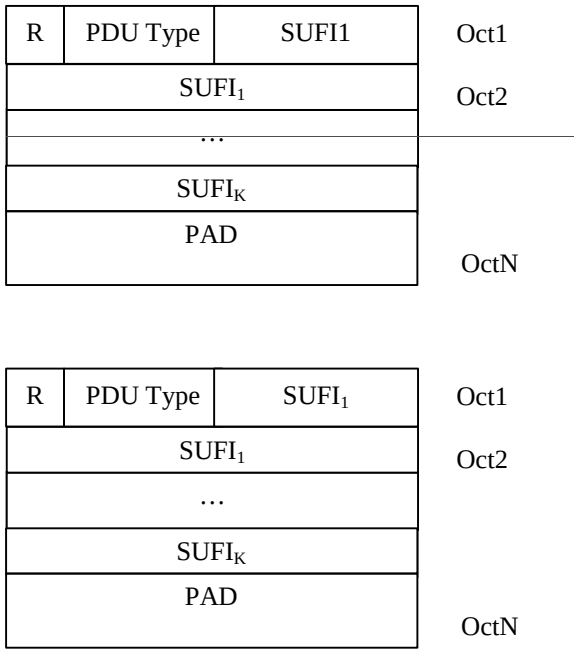


Figure 9.5: Piggybacked STATUS PDU

9.2.2.8 Length Indicator (LI)

The Length Indicator is used to indicate, each time, the end of an SDU occurs in the PU. The Length Indicator points out the number of octets between the end of the last Length Indicator field and up to and including the octet at the end of an SDU segment. Length Indicators are included in the PUs that they refer to. The size of the Length Indicator may be either 7bits or 15bits. The maximum value of a Length Indicator in AM will be no greater than the RLC PDU size – AMD PDU Header – PADDING. The maximum value of a Length Indicator in UM will be no greater than the RLC PDU size – UMD PDU Header – PADDING.

A Length Indicator group is a set of Length Indicators that refer to a PU. Length Indicators that are part of a Length Indicator group must never be reordered within the Length Indicator group or removed from the Length Indicator group.

If there can be more than one Length Indicator, each specifying the end of an SDU in a PU, the order of these Length Indicators must be in the same order as the SDUs that they refer to.

In the case where the end of last segment of an SDU exactly ends at the end of a PDU and there is no LI that indicates the end of the SDU, the next Length Indicator, shall be placed as the first Length Indicator in the following PU and have value LI=0.

In the case where the last segment of an RLC SDU is one octet short of exactly filling the previous RLC PU, and 15-bit Length Indicators are used, the next Length Indicator shall be placed as the first Length Indicator in the following PU and have value LI=111 1111 1111 1011. The remaining one octet in the previous RLC PU shall be ignored.

A PU that has unused space, to be referred to as padding, must use a Length Indicator to indicate that this space is used as padding. A padding Length Indicator must be placed after any Length Indicators for a PU.

All unused space in a PU must be located at the end of the PDU, be a homogeneous space and is referred to as padding. Predefined values of the Length Indicator are used to indicate this. 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.

STATUS PDUs can be piggybacked on the AMD PDU by using part or all of the padding space. A Length Indicator must be used to indicate the piggybacked STATUS PDU. This Length Indicator takes space from the padding space or piggybacked STATUS PDU and not the PDU data and will always be the last Length Indicator. Where only part of the padding space is used by a piggybacked STATUS PDU then the end of the piggybacked STATUS PDU is determined 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 PDU. The padding/piggybacked STATUS PDU predefined Length Indicators shall be added after the very last (i.e. there could be more than one SDU that end within a PDU) Length Indicator that indicates the end of the last SDU segment in the PU.

If SDU discard with explicit signalling is used an AMD PDU can contain a maximum number of 15 LIs indicating the end of an SDU and the rest of the AMD PDU space shall be used as padding/piggybacked STATUS PDU.

For AM, 7bit indicators shall be used if the AMD PDU size is ≤ l26 octets. Otherwise 15bit indicators shall be used. For UM, 7bit indicators shall be used if the UMD PDU size is ≤ l25 octets. Otherwise 15bit indicators shall be used.

The length of the Length Indicator only depends on the size of the largest RLC PDU. The length of the Length Indicator is always the same for all PUs, for one RLC entity.

If the maximum RLC PDU size for an RLC entity is not explicitly configured (e.g. on FACH), the length of the Length Indicator is determined by the maximum configured TB size for the transport channel on which the logical channel is mapped.

For Release 99, there is one PU in an AMD PDU.

Length: 7bit

Bit	Description
0000000	The previous RLC PDU was exactly filled with the last segment of a RLC SDU and there is no LI that indicates the end of the SDU in the previous RLC PDU.
1111100	UMD PDU: The first data octet in this RLC PDU is the first octet of a 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: 15bit

Bit	Description
000000000000000	The previous RLC PDU was exactly filled with the last segment of an RLC SDU and there is no LI that indicates the end of the 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. 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 a 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.

9.4 State variables

This sub-clause describes the state variables used in the specification of the peer-to-peer protocol. PUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{12} for AM and 2^7 for UM; the sequence numbers cycle through the entire range: 0 through $2^{12} - 1$ for AM and 0 through $2^7 - 1$ for UM. All arithmetic operations on the following state variables and sequence numbers contained in this specification are affected by the modulus: VT(S), VT(A), VT(MS), VR(R), VR(H), VR(MR), VT(US) and VR(US). When performing arithmetic comparisons of transmitter variables, VT(A) is assumed to be the base. When performing arithmetic comparisons of receiver variables, VR(R) is assumed to be the base.

The RLC maintains the following state variables at the transmitter.

a) VT(S) - Send state variable.

   The sequence number of the next PU to be transmitted for the first time (i.e. excluding retransmission). It is updated after transmission of a PDU, which includes not earlier transmitted PUs and after transmission of a MRW SUFI. The initial value of this variable is 0.

b) VT(A) - Acknowledge state variable.

   The sequence number of the next in-sequence PU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgements. VT(A) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK super-field. The initial value of this variable is 0.

c) VT(DAT).

   This state variable counts the number of times a PU has been transmitted. There is one VT(DAT) for each PU and it is incremented each time the PU is transmitted. The initial value of this variable is 0.

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

   The sequence number of the first PU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1], VT(MS) = VT(A) + Tx_Window_Size. This value represents the upper edge of the transmit window. The transmitter shall not transmit a ~~new~~ PU with ~~if VT(S)~~SN $\geq$ VT(MS). VT(MS) is updated based on receipt of a STATUS PDU including an ACK and/or MRW_ACK and/or a WINDOW super-field. The PU with SN VT(S)-1 can be transmitted also when VT(S)>VT(MS).

e) VT(US) – UM data state variable.

   This state variable gives the sequence number of the next UMD PDU to be transmitted. It is updated each time a UMD PDU is transmitted. The initial value of this variable is 0.

f) VT(PU).

   This state variable is used when the poll every Poll_PU PU function is used. It is incremented with 1 for each PU that is transmitted. It should be incremented for both new and retransmitted PUs. When it reaches Poll_PU a new poll is transmitted and the state variable is 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 function is used. It is incremented with 1 for each SDU that is transmitted. When it reaches Poll_SDU a new poll is transmitted and the state variable is set to zero. The poll bit should be set in the PU that contains the last segment of the SDU. The initial value of this variable is 0.

h) VT(RST) - Reset state variable.

   It is used to count the number of times a RESET PDU is transmitted. VT(RST) is incremented with 1 each time a RESET PDU is transmitted. VT(RST) is reset only upon the reception of a RESET ACK PDU, i.e. VT(RST) is not reset when a RLC reset occurs which was initiated from the peer RLC entity. The initial value of this variable is 0.

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

It is used to count the number of times a MRW command is transmitted. VT(MRW) is incremented with 1 each time an MRW command is transmitted. VT(MRW) is reset when the discard procedure is terminated. The initial value of this variable is 0.

The RLC maintains the following state variables at the receiver:

a) VR(R) - Receive state variable.

The sequence number of the next in-sequence PU expected to be received. It is set equal to SNmax+1 upon receipt of the next in-sequence PU, where SNmax is the sequence number of the highest received in-sequence PU. The initial value of this variable is 0.

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

The sequence number of the highest expected PU. This state variable is set equal to SN+1 onlywhen a new PU is received with VR(MR)>SN≥VR(H). The initial value of this variable is 0.

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

The sequence number of the first PU not allowed by the receiver [i.e. the receiver will allow up to VR(MR) – 1], VR(MR) = VR(R) + Rx_Window_Size. The receiver shall discard PUs with SN $\geq$ VR(MR).

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

The sequence number of the next PDU to be received. It shall set equal to SN + 1 upon reception of a PDU. The initial value of this variable is 0.

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

The number of PUs that should be received yet as a consequence of the transmission of the latest status report. In acknowledged mode, this state variable is updated at the end of each transmission time interval. It is decremented by the number of PUs that should have been received during the transmission time interval. If VR(EP) is equal to zero, then check if all PUs requested for retransmission in the latest status report have been received.

9.7.6 Local Suspend function for acknowledged mode transfer

The higher layer (RRC) may suspend the RLC entity. The CRLC-SUSPEND-Req indicates this request. The RLC entity shall, when receiving this request, not send RLC PDUs with SN>=VT(S)+N (N is given by the CRLC_SUSPEND-Req primitive). The RLC entity shall acknowledge the CRLC-SUSPEND-Req ordering a suspend with a CRLC-SUSPEND-Conf with the current value of VT(S). The suspend state is left when a CRLC-RESUME-Req primitive indicating resume is received.

9.7.7 RLC stop, RLC Continue function

The higher layer may stop the RLC entity. The stop parameter in the CRLC-CONFIG-Req primitive indicates this request. The RLC entity shall, when receiving this request, not submit any RLC PDUs to lower layer or receive any RLC PDUs. The data transmission and reception is continued when the continue parameter in the CRLC-CONFIG-Req primitive is received. If the continue parameter is received when the RLC entity is not stopped, no action shall be taken.

When the RLC entity is stopped, the RLC timers are not affected. triggered polls and status transmissions are delayed until the RLC entity is continued.

11.2.2.1 UMD PDU contents to set

The Sequence Number field shall be set equal to VT(US).

The Extension bit shall be set to 1 if the next field is a length indicator field, otherwise it shall be set to zero.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number octets between the end of the header fields and the end of the segment. If padding is needed another length indicator shall be added~~. Padding shall only be used when it is necessary, i.e. PDUs containing only padding shall not be transmitted.~~ If the PDU is exactly filled with the last segment of a SDU and there is no room for a length indicator field a length indicator field set to only 0's shall be included in the next PDU.

11.3.2.1 AMD PDU contents to set

If the PDU is transmitted for the first time, the Sequence Number field shall be set equal to VT(S) and VT(S) shall be updated.

The setting of the Polling bit is specified in subclause 11.3.2.1.1.

One length indicator field shall be included for each end of a SDU that the PDU includes. The length indicator shall be set equal to the number of octets between the end of the header fields and the end of the segment~~. If padding is needed another length indicator shall be added. Padding shall only be used when it is necessary, i.e. PDUs containing only padding shall not be transmitted.~~ If the PDU is exactly filled with the last segment of a SDU and there is no room for a length indicator field a length indicator field set to only 0's shall be included in the next PDU. How to perform the segmentation of a SDU is specified in subclause 11.3.2.1.2.

11.5.2.2 STATUS PDU contents to set

The size of the STATUS PDU shall be equal to one of the allowed PDU sizes. The information that needs to be transmitted in a status report can be split into several STATUS PDUs if one STATUS PDU does not accommodate all the information. A SUFI can not be split into several STATUS PDUs. Indication of the same PU shall not be given in more than one STATUS PDU of a STATUS report.

Which SUFI fields to use is implementation dependent, but the status report shall include information about PUs that have been received and information about all PUs detected as missing. No information shall be given for PUs with SN≥VR(H), i.e. PUs that have not yet reached the receiver. ~~STATUS PDUs that do not contain any status information, i.e. containing only a NO_MORE SUFI shall not be transmitted.~~

Padding shall be inserted if the SUFI fields do not fill an entire STATUS PDU. If the PDU contains padding the last SUFI field shall be either an ACK SUFI or a NO_MORE SUFI. If there is no padding in the STATUS PDU, NO_MORE SUFI or ACK SUFI does not need to be included in the STATUS PDU.

3GPP TSG-RAN WG2 Meeting #17 ***Tdoc R2-002440***
Sophia Antipolis, France, 13-17 November

CR-Form-v3

CHANGE REQUEST

⌘ **25.322** **CR** **094** ⌘ rev **r1** ⌘ Current version: **3.4.0** ⌘

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

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☒ Radio Access Network ☐ Core Network ☐

Title:	⌘	RLC reset handling
Source:	⌘	TSG-RAN WG2
Work item code:	⌘	***Date:*** ⌘ 2000-11-09
Category:	⌘	**F** ***Release:*** ⌘ R99

Use one of the following categories:
F *(essential 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)

Reason for change:	⌘	RLC buffer handling at reset is not specified
Summary of change:	⌘	It is specified how unacknowledged RLC SDUs shall be handled at RLC reset. Depending on RRC configuration, unacknowledged SDUs are either 1) transmitted after the reset with new sequence numbers or 2) discarded in case of reset. Updates in r1: Reset behaviour is aligned with discussion in the plenary.
Consequences if not approved:	⌘	

Clauses affected:	⌘	9.3.3.2, 9.3.3.3, 11.4.3, 11.4.4, 11.4.5.3
Other specs affected:	⌘	☒ Other core specifications ⌘ 25.331 ☐ Test specifications ☐ 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/3G_Specs/CRs.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://www.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2000-09 contains the specifications resulting from the September 2000 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.

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 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 an CRLC-CONFIG-Req from higher layer the RLC entity is created and acknowledged data transfer ready state is entered.

9.3.3.2 Acknowledged Data Transfer Ready State

In the acknowledged data transfer ready state, acknowledged mode data can be exchanged between the entities. Upon reception of a CRLC-CONFIG-Req from higher layer the RLC entity is terminated and the null state is entered.

Upon errors in the protocol, the RLC entity sends a RESET PDU to its peer and enters the reset pending state.

Upon reception of a RESET PDU, the RLC entity resets the protocol (see subclause 11.4.3)~~resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value~~, increments the hyper frame number if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU~~)~~ and responds to the peer entity with a RESET ACK PDU.

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

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 CRLC-CONFIG-Req from higher layer the RLC entity is terminated and the null state is entered.

Upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU, the RLC entity resets the protocol (see subclause 11.4.4)~~resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value~~, increments the hyper frame number~~)~~ and one of the following state transitions take place.

The RLC entity enters the acknowledged data transfer ready state if Reset Pending State was entered from Acknowledged Data Transfer Ready State or if Reset Pending State was entered from Local Suspend State and a CRLC-RESUME–Req was received in Reset Pending State.

The RLC entity enters into Local Suspend State if Reset Pending State was entered from Local Suspend State or if Reset Pending State was entered from Acknowledged Data Transfer Ready State and a CRLC-SUSPEND-Req was received in Reset Pending State.

Upon reception of a RESET ACK PDU with a different RSN value as in the corresponding RESET PDU the RESET ACK PDU is discarded..

Upon reception of a RESET PDU, the RLC entity resets the protocol (see subclause 11.4.3)~~resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value)~~, increments the hyper frame number if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU~~)~~, sends a RESET ACK PDU and stays in the reset pending state.

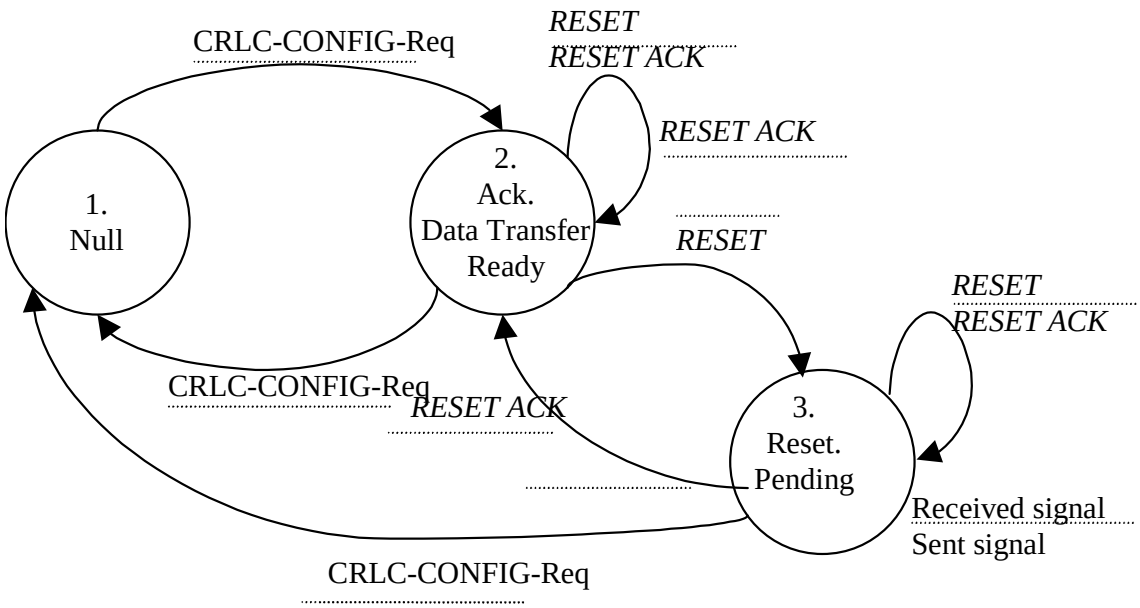


Figure 9.18: The state model for the acknowledged mode entities when reset is performed

9.3.3.4 Local Suspend State

Upon reception of CRLC-SUSPEND-Req from higher layer (RRC) in Acknowledge Data Transfer Ready State the RLC entity is suspended and the Local Suspend state is entered. In the Local Suspend state RLC shall not send a RLC-PDUs with a SN>=VT(S)+N. Upon reception of CRLC-RESUME-Req from higher layer (RRC) in this state, the RLC entity is resumed and the Data Transfer Ready state is entered.

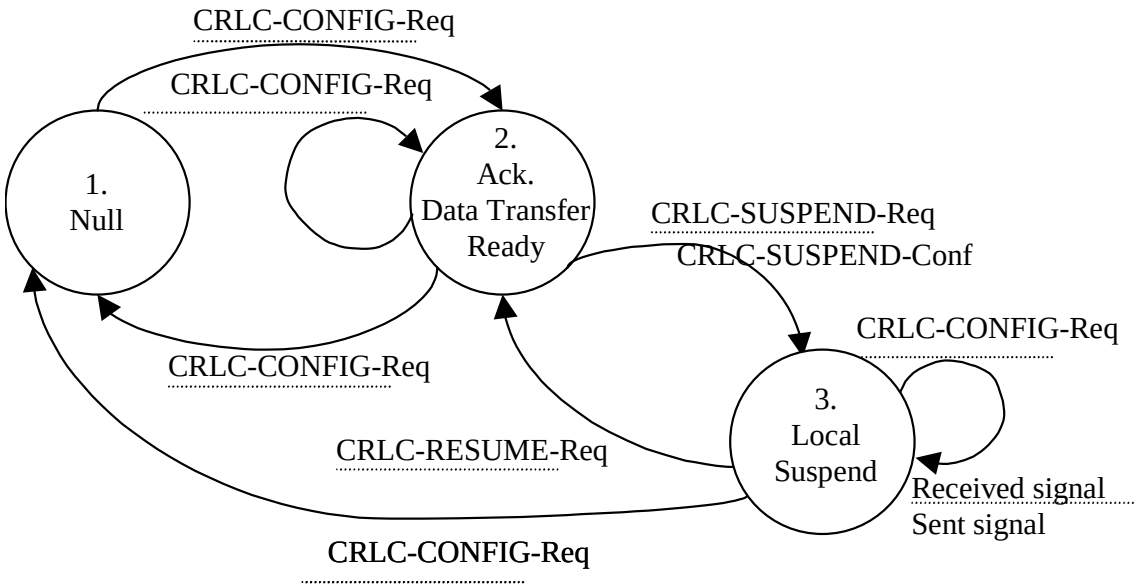


Figure 9.19: The state model for the acknowledged mode entities when local suspend is performed

11.4 RLC reset procedure

11.4.1 Purpose

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 a RLC reset. The sender can be either the UE or the network and the receiver is either the network or the UE.

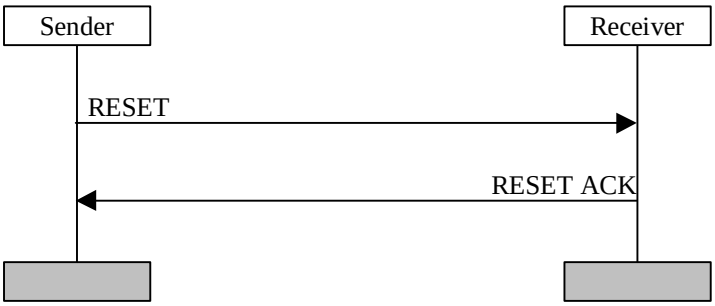


Figure 11.4: RLC reset procedure

11.4.2 Initiation

The procedure shall be initiated when a protocol error occurs.

The sender sends the RESET PDU when it is in data transfer ready state and enters reset pending state. The sender shall start the timer Timer_RST and increase VT(RST) with 1. The RESET PDU 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.

The RESET PDU has higher priority than data PDUs.

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, i.e., 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 size of the RESET PDU shall be equal to one of the allowed PDU sizes. The RSN field shall indicate the sequence number of the RESET PDU. This sequence number is incremented every time a new RESET PDU is transmitted.

11.4.3 Reception of the RESET PDU by the receiver

Upon reception of a RESET PDU the receiver shall respond with a RESET ACK PDU. The receiver resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value. ~~All RLC SDUs in the Rx that has not been delivered to higher layer shall be discarded. Depending on the configuration, the RLC SDUs not yet acknowledged in the Tx shall either be transmitted after the reset with new sequence numbers or the SDUs in the Tx shall be discarded. All control PDUs are discarded independent of the configuration.~~ Both the transmitter and receiver side of the AM RLC entity ~~is~~are reset. All RLC PDUs in the AM RLC receiver shall be discarded. The RLC SDUs in the AM RLC transmitter that w~~as~~ere transmitted before the reset shall be discarded.

In the received RESET PDU the Receiver shall check the value of RSN (Reset Sequence Number) field. If the value of the RSN field is different from the RSN value in the previously received RESET PDU the Receiver shall increase the value of the HFN by one.

If the value of the RSN is equal to the RSN value in the previously received RESET PDU, (i.e. the RESET PDU is a retransmitted RESET PDU) the value of the HFN shall not be increased and only a RESET ACK PDU shall be sent to the peer RLC entity.

The RESET ACK PDU 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.

The RESET ACK PDU has higher priority than data PDUs.

11.4.3.1 RESET ACK PDU contents to set

The size of the RESET ACK PDU shall be equal to one of the allowed PDU sizes. The RSN field shall always be set to the same value as in the corresponding RESET PDU.

11.4.4 Reception of the RESET ACK PDU by the sender

When the sender is in reset pending state and receives a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU the Timer_RST shall be stopped and the value of the HFN shall be increased by one. The sender resets the state variables in 9.4 to their initial value and resets configurable parameters to their configured value. Both the transmitter and receiver side of the AM RLC entity is reset. All RLC PDUs in the AM RLC receiver shall be discarded. The RLC SDUs in the AM RLC transmitter that w~~as~~ere transmitted before the reset shall be discarded.~~All RLC SDUs in the Rx that has not been delivered to higher layer shall be discarded. Depending on the configuration, the RLC SDUs not yet acknowledged in the Tx shall either be transmitted after the reset with new sequence numbers or the SDUs in the Tx shall be discarded. All control PDUs are discarded independent of the configuration.~~

The sender shall enter data transfer ready state.

Upon reception of a RESET ACK PDU with a different RSN value as in the corresponding RESET PDU the RESET ACK PDU is discarded.

Upon reception of a RESET ACK PDU in data transfer ready state the RESET ACK PDU is discarded.

11.4.5 Abnormal cases

11.4.5.1 Timer_RST timeout

Upon expiry of Timer_RST the sender shall retransmit the RESET PDU and increase VT(RST) with 1. In the retransmitted RESET PDU the value of the RSN field shall not be incremented.

11.4.5.2 VT(RST) ≥ MaxRST

If VT(RST) becomes larger or equal to MaxRST the RRC layer shall be informed.

11.4.5.3 Reception of the RESET PDU by the sender

Upon reception of a RESET PDU in reset pending state the sender shall respond with a RESET ACK PDU. The sender resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value. However, VT(RST) and Timer_RST are not reset. Both the transmitter and receiver side of the AM RLC entity ~~is~~are reset. All RLC PDUs in the AM RLC receiver shall be discarded. The RLC SDUs in the AM RLC transmitter that w~~as~~ere transmitted before the reset shall be discarded. ~~All RLC SDUs in the Rx that has not been delivered to higher layer shall be discarded. Depending on the configuration, the RLC SDUs not yet acknowledged in the Tx shall either be transmitted after the reset with new sequence numbers or the SDUs in the Tx shall be discarded. All control PDUs are discarded independent of the configuration.~~

The hyper frame number is incremented if the RSN field indicates that the RESET PDU is not a retransmitted RESET PDU. The sender shall stay in the reset pending state. The sender shall enter data transfer ready state only upon reception of a RESET ACK PDU with the same RSN value as in the corresponding RESET PDU.

3GPP TSG RAN WG2#17
Sophia Antipolis, France, 13th – 17th November, 2000

Document R2-002266
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.322 CR 095 Current Version: 3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG-RAN #10 *list expected approval meeting # here ↑*

for approval [X] for information [ ]

strategic [ ] non-strategic [ ] *(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from:* ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: TSG-RAN WG2 **Date:** 2000-11-13

Subject: Inclusion of stage 3 for ciphering

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	[X]
A	Corresponds to a correction in an earlier release	[ ]
B	Addition of feature	[ ]
C	Functional modification of feature	[ ]
D	Editorial modification	[ ]

Release:

Phase 2	[ ]
Release 96	[ ]
Release 97	[ ]
Release 98	[ ]
Release 99	[X]
Release 00	[ ]

Reason for change: Stage 3 aspects of security have been included.

Clauses affected: 11.7(new)

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:


help.doc

<--------- double-click here for help and instructions on how to create a CR.

11. 7 Ciphering

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 RLC UM PDU header. This is shown below in Figure 11.7.1.

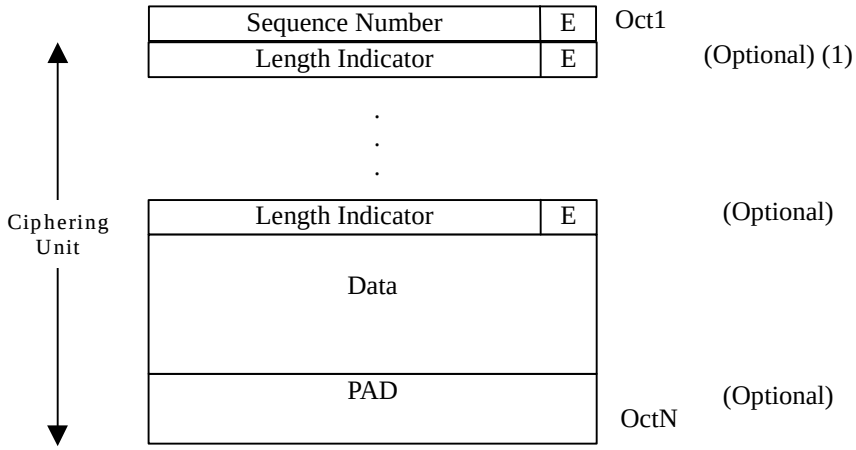


Figure 11.7.1: Ciphering unit for a UMD PDU

- For RLC AM mode, the ciphering unit is the AMD PDU excluding the two first octets, i.e. excluding the RLC AM PDU header. This is shown below in Figure 11.7.2.

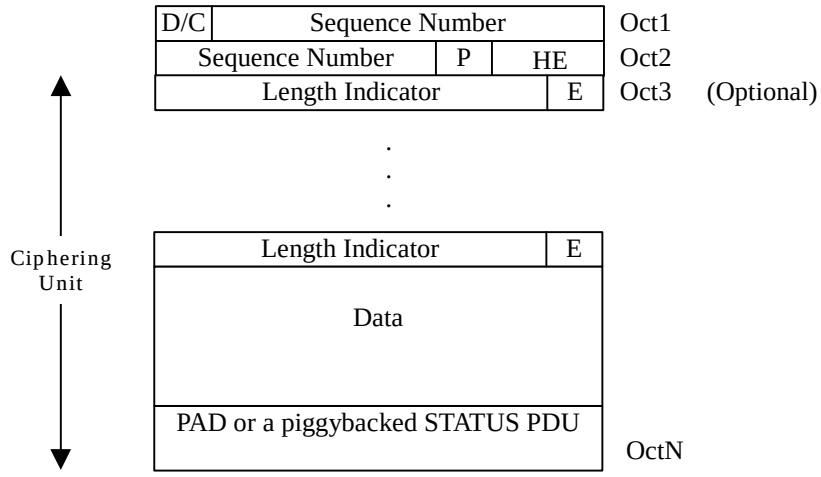


Figure 11.7.1: Ciphering unit for a AMD PDU

The ciphering algorithm and key to be used are configured by upper layers [7] and the ciphering method shall be applied as specified in [10].

The parameters that are required by RLC for ciphering are defined in [10] and are input to the ciphering algorithm. The parameters required by RLC which are provided by upper layers [7] 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 (Radio Bearer ID)
- CK (Ciphering Key)