# TSG-RAN Meeting #9 *RP-000358*
# Oahu, HI, USA, 20 – 22 September 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 |
|---|---|---|---|---|---|---|---|---|
| R2-001482 | agreed | 25.322 | 059 | 1 | State variables after window change | F | 3.3.0 | 3.4.0 |
| R2-001748 | agreed | 25.322 | 060 | 4 | SDU discard | F | 3.3.0 | 3.4.0 |
| R2-001840 | agreed | 25.322 | 061 | 5 | General RLC corrections | F | 3.3.0 | 3.4.0 |
| R2-001368 | agreed | 25.322 | 066 | | Editorial changes to RLC | F | 3.3.0 | 3.4.0 |
| R2-001839 | agreed | 25.322 | 067 | 4 | Correction to RLC window size range | F | 3.3.0 | 3.4.0 |
| R2-001523 | agreed | 25.322 | 068 | 2 | Window based polling | F | 3.3.0 | 3.4.0 |
| R2-001799 | agreed | 25.322 | 070 | 2 | General corrections to RLC | F | 3.3.0 | 3.4.0 |
| R2-001600 | agreed | 25.322 | 071 | | State Transition in RLC Acknowledged Mode | F | 3.3.0 | 3.4.0 |
| R2-001613 | agreed | 25.322 | 073 | | Clarification of the Length Indicators | F | 3.3.0 | 3.4.0 |
| R2-001755 | agreed | 25.322 | 076 | 1 | RLC corrections | F | 3.3.0 | 3.4.0 |
| R2-001811 | agreed | 25.322 | 077 | 1 | Corrections to reset procedure and length indicator definitions | F | 3.3.0 | 3.4.0 |
| R2-001658 | agreed | 25.322 | 078 | | RLC Modes for SHCCH | F | 3.3.0 | 3.4.0 |
| R2-001812 | agreed | 25.322 | 079 | | CCCH in UM RLC | F | 3.3.0 | 3.4.0 |

**3GPP TSG RAN WG2 meeting #14**
**Paris, France, 03-07 July 2000**

***Document* R2-001482**
*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** **059r1** Current Version: 3.3.0

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

For submission to: TSG-RAN #9 *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:** 2000-06-19

**Subject:** State variables after window change

**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. It should be clarified how the state variables VT(S) and VT(MS) shall be handled in the special case that VT(S)>VT(MS) after a change of window size.

**Clauses affected:** 9.4, 11.3.2

**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.

## 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. 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 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 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 upon the reception of a RESET ACK PDU. 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 a MRW command is transmitted. VT(MRW) is reset upon the reception of a STATUS PDU which suggests the acknowledgement of a MRW command in the receiver or the occurrence of discarding new SDU. 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 ≥ 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.

# 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.

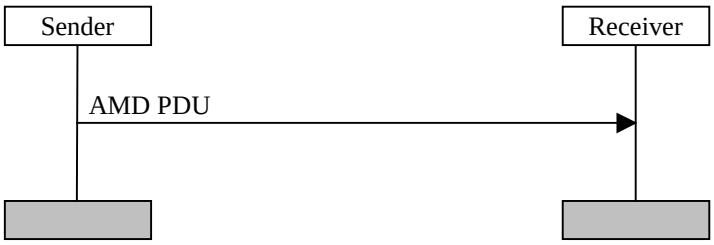


**Figure 11.3: Acknowledged mode data transfer procedure**

## 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 always be retransmitted. In addition, the PU with highest SN that has not yet been acknowledged may be retransmitted if the peer Rx window size is less than half the maximum RLC AM sequence number.

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.

**3GPP TSG RAN WG2 meeting #15**
**Sophia Antipolis, France, 21-25 August 2000**

***Document* R2-001748**
*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 | 060r4 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG Use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 2000-08-04

**Subject:** SDU discard

**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 present SDU discard procedure does not work in some cases and needs to be corrected. Especially, a mechanism is needed to differentiate between an acknowledgement for a MRW SUFI and an acknowledgement for a received PDU. Also, the definition of the MRW SUFI needs to be corrected to correctly reflect the number of discarded SDUs in the receiver.
2. The criteria for detection of obsolete MRW_ACK SUFIs is changed. The new criteria allows the Rx to detect that some parameters in the SUFI is not correct, which can be used for a simple form of detection of undetected errors in the SUFI fields.
3. The definition of the MRW SUFI is corrected to be able to discard SDUs with a SDU size larger then the TX window size.
4. The definition of the N field in the MRW_ACK SUFI is corrected.
5. It should be mentioned that VT(S) is updated when a new MRW command is transmitted.

**Clauses affected:** 9.2.2.11, 9.2.2.11.7, 9.2.2.11.8, 9.2.4, 9.5, 11.6.1, 11.6.2, 11.6.2.2, 11.6.3, 11.6.4, 11.6.5, 11.6.6.1, 11.6.6.2, 11.6.6.2a (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.

### 9.2.2.11 SUFI

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

Length: variable number of bits

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

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

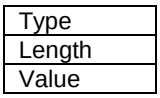


**Figure 9.7: The Structure of a Super-Field**

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

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

The length sub-field gives the length of the variable size part of the following value sub-field and the length of it depends on the super-field type. The value sub-field includes the value of the super-field, e.g. the bitmap in case of a BITMAP super-field, and the length is given by the length of the type sub-field.

### 9.2.2.11.7 The Move Receiving Window Acknowledgement super-field

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

| Type = **MRW_ACK** |
|---|
| N |
| SN_ACK |

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

**N**

Length: 4 bits

The N field shall be set equal to the $N_{LENGTH}$ field in the received MRW SUFI ~~if one or more SDUs are discarded in the receiver. The N field shall be set to zero if no SDU is discarded in the receiver, i.e. the received MRW SUFI is obsolete.~~ if the SN_ACK field is equal to~~s~~ the $SN\_MRW_{LENGTH}$ field~~-~~. Otherwise N shall be set to 0.

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

**SN_ACK**

Length: 12 bits

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

~~9.2.2.11.7 The Move Receiving Window 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~~ |
| ~~$SN\_MRW_1$~~ |
| ~~...~~ |
| ~~$SN\_MRW_{LENGTH}$~~ |

~~**Figure 9.14: 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. It equals the amount of discarded SDUs within one SUFI.~~

~~**$SN\_MRW_i$**~~

~~Length: 12 bits~~

~~$SN\_MRW_i$ fields enumerate each of the discarded SDUs by indicating the sequence number of the PU in which the succeeding SDU of the i:th discarded SDU begins.~~

~~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. It also indicates the first data byte in the PU with sequence number SN_MRW$_{LENGTH}$ corresponds to the first byte of the SDU to be reassembled next.~~

### 9.2.2.11.8 The Move Receiving Window ~~and Ignore First LI~~ (MRW~~_N_IFL~~) super-field

The 'Move Receiving Window'~~ and ignore first N LIs'~~ 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.~~ It also indicates to the receiver the presence and the amount of the trailing bytes of the discarded SDU in the PU with sequence number SN_MRW$_{LENGTH}$.~~ The format is given in the figure below.

| ~~Type = **MRW_N_IFL**~~ |
|---|
| ~~N~~ |
| ~~LENGTH~~ |
| ~~SN_MRW$_1$~~ |
| ~~...~~ |
| ~~SN_MRW$_{LENGTH}$~~ |

~~**Figure 9.15: The MRW_N_IFL fields in a STATUS PDU**~~

| 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 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

~~The number of LI fields in the PU that shall be ignored in the SN_MRW$_{LENGTH}$. It equals the amount of SDUs in the PU that are discarded from the PU identified by SN_MRW$_{LENGTH}$.~~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.

~~**LENGTH**~~

~~Length: 4 bits~~

~~The number of $SN\_MRW_i$ fields in the super-field of type MRW. It equals the amount of discarded SDUs within one MRW SUFI.~~

~~**$SN\_MRW_i$**~~

~~Length: 12 bits~~

~~$SN\_MRW_i$ fields enumerate each of the discarded SDUs by indicating the sequence number of the PU in which the succeeding SDU of the i:th discarded SDU begins.~~

~~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 LIs and the corresponding data bytes in the PU with sequence number $SN\_MRW_{LENGTH}$.~~

## 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 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.

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 upon the reception of a RESET ACK PDU. 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 a ~~-~~MRW command is transmitted. VT(MRW) is reset when the discard procedure is terminated~~upon the~~

~~reception of a STATUS PDU which suggests the acknowledges~~ement ~~the~~of a ~~transmitted MRW command in the receiver or the occurrence of discarding new SDU~~. 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 ≥ 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.

## 9.5 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 or negative acknowledgement of the AMD PDU with SN=VT(S)-1 at the time the poll was transmitted. The value of the timer is signalled by RRC.

If the timer expires and no STATUS PDU containing an acknowledgement or negative acknowledgement of the AMD PDU with SN=VT(S)-1 at the time the poll was transmitted 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.

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 transmitted and when it expires EPC can start decrease (see subclause 9.7.3). 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 a poll is transmitted (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 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 last STATUS PDU in a status report is transmitted and no new status report containing the mentioned SUFIs can be transmitted before the timer has expired. The timer does not prohibit transmission of the SUFIs MRW, ~~MRW_N_IFL~~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 a status report is transmitted and the timer is restarted. The value of the timer is signalled by RRC.

h) Timer_RST.

It is used to detect the loss of RESET ACK PDU from the peer RLC entity. This timer is set when the RESET PDU is transmitted. And it will be stopped upon reception of RESET ACK PDU. 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 ~~last~~ STATUS PDU containing the MRW SUFI ~~of the status report~~ is first transmitted. Each time the timer expires the ~~status report~~MRW SUFI is retransmitted and the timer is restarted (when the ~~last~~ STATUS PDU ~~of the status report~~ containing the MRW SUFI is retransmitted). It shall be stopped when one of the termination criteria for the SDU discard is fulfilled. ~~when a STATUS PDU is received that indicates that VR(R) ≥ SN_MRW. It shall also be stopped if a new MRW procedure is triggered while it is running.~~ The value of the timer is signalled by RRC.

# 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.

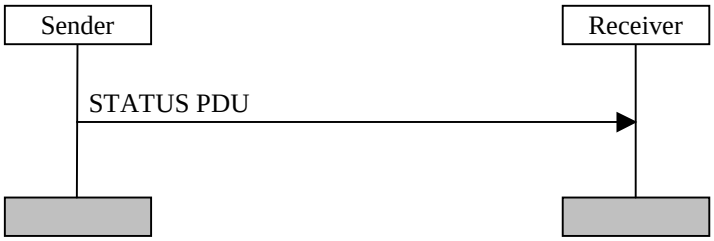


**Figure 11.6: SDU discard with explicit signalling**

## 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 ~~a~~ 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, ~~but~~and VT(S) shall be updated when a new MRW ~~command~~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.

~~If the PU with sequence number SN_MRW$_{LENGTH}$ contains LI indicating trailing data from the discarded SDU, the transmitter shall send SUFI MRW_N_IFL indicating to the receiver to discard the first N LIs and the corresponding data bytes. Otherwise the transmitter shall send SUFI MRW.~~

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 SUFI~~s~~ ~~should~~shall be sent before the ~~STATUS PDU is received indicating the appropriate value of VR(R).~~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 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.~~

The status report shall include the MRW~~/MRW_N_IFL~~ SUFI, other SUFI fields can be used additionally. MRW~~/MRW_N_IFL~~ 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.

~~If the PDU contains padding the last SUFI field shall be a No More Data super field.~~

### 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 ~~indicating the revised value of VR(R)~~. ~~In the MRW_ACK SUFI, the N field shall be set equal to the $N_{LENGTH}$ field in the received MRW SUFI and SN_ACK shall be set equal to VR(R), if one or more SDUs where discarded. If no SDU is discarded, i.e. the received MRW SUFI is obsolete, the N field shall be set to zero.~~.

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.

~~In case of receiving SUFI_MRW, the receiver shall start reassembling the next SDU from the first data byte of the PU with sequence number SN_$MRW_{LENGTH}$.~~

~~If the receiver receives SUFI MRW_N_IFL , it shall discard the first N LIs and the corresponding data bytes and start reassembling the next SDU from the data byte indicated by the N+1:th LI field of the PU with sequence number SN_$MRW_{LENGTH}$.~~

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 a 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 ~~Reception of STATUS PDU if VR(R) ≥ SN_$MRW_{LENGTH}$~~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 N is equal to the $N_{LENGTH}$ indicated in the transmitted MRW SUFI.~~

~~2. On the reception of a STATUS PDU which contains an ACK SUFI indicating VR(R ) > SN_$MRW_{LENGTH}$~~

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.

~~. 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.~~~~when a STATUS PDU is received indicating a value of VR(R) ≥ SN_MRW$_{LENGTH}$.~~

If ~~this occurs~~one of the termination criteria above is fulfilled, Timer_MRW is stopped ~~thereby terminating the procedure~~and the discard procedure is terminated.

When VT(MRW) reaches MaxMRW, the procedure is terminated and a RLC reset is performed.~~If new SDUs are discarded during the running of the Timer_MRW, a new discard procedure should be initiated no earlier than after the reception of STATUS PDU with VR(R) ≥ SN_MRW$_{LENGTH}$.~~

## 11.6.5 Expiration of timer Timer_MRW

If Timer_MRW expires before the discard procedure is terminated, ~~before a STATUS PDU is received indicating a value of VR(R) greater or equal to the MRW parameter then~~ the MRW SUFI~~STATUS(MRW)~~ shall be retransmitted, VT(MRW) is incremented by one and Timer_MRW restarted. MRW SUFI ~~should~~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 ~~should~~shall be performed.

### 11.6.6.2a 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 If the N field in the received MRW_ACK field is not equal to the N$_{LENGTH}$ field 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.

nicola.magnani@cselt.it

**3GPP TSG RAN WG2 meeting #15**
**Sophia Antipolis, France, 21-25 August 2000**

***Document* R2-001840**
*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 061r5** Current Version: 3.3.0

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

For submission to: TSG-RAN #9 *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 WG22 **Date:** 2000-08-18

**Subject:** General RLC corrections

**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. Transfer of user data should be added in table 6.3
2. The stop criteria for the timer_poll should be corrected
3. ~~The poll window formula should be corrected~~
4. The content of a STATUS PDU needs clarification
5. RLIST was excluded from the list of SUFIs which could report erroneous information in section 10.
6. Section 11.5.2.2 mentions that STATUS PDUs should be transmitted sequentially in order of increasing sequence numbers of the PUs that are indicated in the STATUS PDUs. ~~Also indication of the PU at lower edge of receiver window need not be the first indication.~~
7. Also multiple indications of the same PU in the same STATUS report in different STATUS PDUs needs to be avoided ~~.if window size > 2047~~
8. Multiple retransmissions of the same PU in response to a received STATUS PDU needs to avoided.
9. It needs to be described what action transmitter should take in case it receives a STATUS PDU indicating different status for the same PU.

**Clauses affected:** 6.1, 9.5, 9.6, 9.7.1, 10, 11.3.2.1.1, 11.5.2, 11.5.2.2, 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.

## 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 | - | - | + | + |

## 9.5 ~~9.5~~ Timers

~~When the start time for a timer in the UE is connected with the transmission of a certain PU, the timer shall be started when the PU is transmitted by the physical layer.~~

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 ~~or negative acknowledgement~~ of ~~the~~ 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 ~~containing an acknowledgement or negative acknowledgement of the AMD PDU with SN=VT(S)-1 at the time the poll was transmitted~~ 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 transmitted and when it expires EPC can start decrease (see subclause 9.7.3). 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 a poll is transmitted (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 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 last STATUS PDU in a status report is transmitted and no new status report containing the mentioned SUFIs can be transmitted before the timer has expired. The timer does not prohibit transmission of the SUFIs MRW, MRW_N_IFL, 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 a status report is transmitted and the timer is restarted. The value of the timer is signalled by RRC.

h) Timer_RST.

It is used to detect the loss of RESET ACK PDU from the peer RLC entity. This timer is set when the RESET PDU is transmitted. And it will be stopped upon reception of RESET ACK PDU. 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 last STATUS PDU of the status report is first transmitted. Each time the timer expires the status report is retransmitted and the timer is restarted (when the last STATUS PDU of the status report is retransmitted). It shall be stopped when a STATUS PDU is received that indicates that VR(R) ≥ SN_MRW. It shall also be stopped if a new MRW procedure is triggered while it is running. 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 transmitted when:

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

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

.

~~$$\left[ \frac{Dist(\mathrm{VT(A)}, \mathrm{VT(S)})}{Tx\_Window\_Size} \right] * 100 > Poll\_Window$$~~

~~Where Dist(VT(A),VT(S)) is the distance, within the Tx window, in number of PUs between VT(A) and VT(S). If VT(S)>VT(MS), the whole expression shall be evaluated to 100%.~~

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 ~~if no STATUS PDU containing an acknowledgement or negative acknowledgement of all~~the ~~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 criterion for stopping the timer has not occurred ~~has been received~~ 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.

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.

# 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 ,~~or~~ BITMAP or RLIST is not within the value between "Acknowledge state variable(VT(A))" and "Send state variable(VT(S))", and

~~- Each Sequence Number of missing PU informed by SUFI BITMAP is not within the value between "Acknowledge state variable(VT(A)) 7" and "Send state variable(VT(S))", 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.

# 11.3 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.

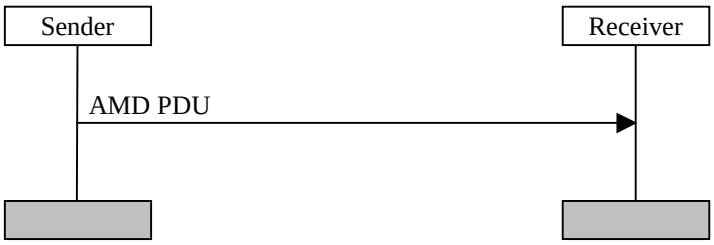


**Figure 11.3: Acknowledged mode data transfer procedure**

## 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 always be retransmitted. In addition, the PU with highest SN that has not yet been acknowledged may be retransmitted if the peer Rx window size is less than half the maximum RLC AM sequence number.

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).

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 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, and

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

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

~~$$\left[\frac{Dist(\mathrm{VT(A)}, \mathrm{VT(S)})}{Tx\_Window\_Size}\right] * 100 > Poll\_Window$$~~

~~Where *Dist*(VT(A),VT(S)) is the distance, within the Tx window, in number of PUs between VT(A) and VT(S). If VT(S)>VT(MS), the whole expression shall be evaluated to 100%~~

7) timer based polling is used and Timer_Poll_Periodic has expired.

8) 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.

If timer based SDU discard with explicit signalling is used and a missing PU is detected the timer Timer_Discard is started.

## 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.

# 11.5 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.

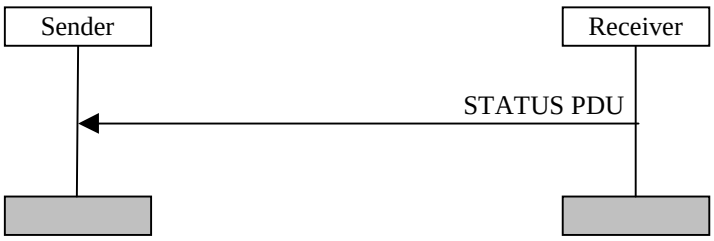


**Figure 11.5: Status report transfer procedure**

## 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. ~~The delay between the transmission of the STATUS PDUs in a status report shall be as low as possible.~~

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.

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.

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.

### 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. ~~When a status report is split into several STATUS PDUs, the STATUS PDUs shall be transmitted sequentially in order of increasing sequence numbers of the PUs that are indicated in the STATUS PDUs, i.e. information about the lower edge of the receiver window shall be transmitted first.~~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 ~~all~~ PUs that have been received ~~or~~ 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 EPC reaches zero and the requested PUs have not been received

If the EPC mechanism is used and VR(EP) has reached 0 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 #14**
**Paris, France, 3-7 July 2000**

***Document R2-001368***
*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 | 066 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG Use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 2000-06-19

**Subject:** Editorial changes 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:**

1. 25.322v3.3. Section 11.3.3 states that a timer discard is started at the receiver . This is obsolete and was initially intended for implicit discard at AM.

**Clauses affected:** 11.3.3

**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.

### 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.
~~If timer based SDU discard with explicit signalling is used and a missing PU is detected the timer Timer_Discard is started.~~

**3GPP TSG-RAN WG2 Meeting #15**
**Sophia Antipolis, France; 21-25 August, 2000**

***Document R2-001839***
*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 | 067r4 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

For submission to: TSG-RAN #9 *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:** 23 August, 2000

**Subject:** Correction to RLC window size range

**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:**

It is proposed to change the RLC window size range to [0, $2^{12}$-1]. This will allow the implementation of efficient flow control schemes. The receiving RLC entity, both in UE or UTRAN, can perform a very efficient flow control by setting the Tx window size to 0.

It is also clarified that the transmission of RRC messages should not be blocked by RLC because of a small value of Minimum WSN.

The changes in Revision 3 are shown in blue:

Some problems have been identified if the Rx is allowed to set Tx window size to zero; please refer to tdoc R2-001614 for an excellent description of the problems.

To solve the problems it is proposed that the Minimum WSN shall always be greater than or equal to the number of transport blocks in the smallest transport block set.

The changes in Revision 3 are shown in green.

**Clauses affected:** 8.1, 9.2.2.11.3

**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.

## 8.1 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, | Data | Not Defined | Not Defined |
| **RLC-TR-DATA** | Data | Data | Not Defined | Not Defined |
| **CRLC-CONFIG** | E/R, Ciphering Elements (UM/AM only), AM_parameters, ~~e.g. Minimum Window Size~~ (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, release or reconfigure the RLC. Ciphering elements are included for UM and AM operation.

~~Following parameters are used in the primitives:~~

~~1) The parameter Minimum Window Size (AM only) indicates the minimum value of the Window Size Number (WSN) in the Window Size Super Field. Value can be either 0 or 1 however the value 0 should not be used for the control plane.~~

**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.

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, all protocol parameters, variables and timers shall be set or reset 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.

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 ~~is~~ 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~~~~When AM RLC is used for the control plane t~~The Minimum WSN ~~should~~shall always be greater than or equal ~~than~~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.

[…]

### 9.2.2.11.3 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 allowed Tx window size to be used by the transmitter. The range of the WSN~~window size~~ is [~~1~~0, $2^{12}$-1]. The minimum value of the window size is 1, if WSN is zero the SUFI shall be discarded by this version of the protocol. The Tx_Window_Size parameter is set equal to WSN upon reception of this SUFI.

**3GPP RAN WG2 #14**
**Paris, France, July 3th – July 7th, 2000**

***Document R2-001523***
*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 | 068r2 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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 | | UTRAN / Radio | X | Core Network | |
|---|---|---|---|---|---|---|---|---|

**Source:** TSG-RAN WG2 **Date:** July 4th 2000

**Subject:** Window based polling

**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 current window based polling algorithm has problem that the unwanted polling is triggered when VT(S) = VT(A). The algorithm should be modified to eliminate such a problem. Also, there should be some explanations for special cases like VT(S)≥VT(MS) and Tx_Window_Size=0.

**Clauses affected:** 9.6, 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.

## 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 ~~transmitted~~ triggered for each PU when:

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

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

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.

### 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).

~~7~~8) ~~t~~Timer based polling is used and Timer_Poll_Periodic has expired.

~~8~~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.

**3GPP TSG RAN WG2 meeting #15**
**Sophia Antipolis, France, 21-25 August 2000**

***Document* R2-001799**
*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 | 070r2 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG Use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 2000-08-18

**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:** If the transmitter window size is less than 2048,restrictions on retransmission of a PU can be removed.

**Clauses affected:** 11.3.2

**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.

# 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 ~~always~~ be retransmitted. In addition, a ~~the~~ PU ~~with highest SN~~ that has not yet been acknowledged, may be retransmitted if the configured transmitter window size is less than 2048. ~~if the peer Rx window size is less than half the maximum RLC AM sequence number.~~

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).

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.

**3GPP TSG RAN WG2 meeting #15**
**Sophia Antipolis, France, 21-25 August, 2000**

***Document* R2-001600**
*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 | 071 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG Use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 2000-08-17

**Subject:** State Transition in RLC Acknowledged Mode

**Work item:** 6.8

| **Category:** *(only one category shall be marked with an X)* | | | **Release:** | |
|---|---|---|---|---|
| F | Correction | X | 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 state model transitions don't handle the situations when a reset occurs in local suspend state or when it gets CRLC suspend command or CRLC resume command in RESET state .

**Clauses affected:** 9.3.3.3, 9.3.3.4

| **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.

## 9.3.3 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) 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, the RLC entity resets the protocol (resets the state variables in 9.4 to their initial value, resets configurable parameters to their configured value, increments the hyper frame number) ~~and~~ 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 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, increments the hyper frame number), sends a RESET ACK PDU and stays in the reset pending state.

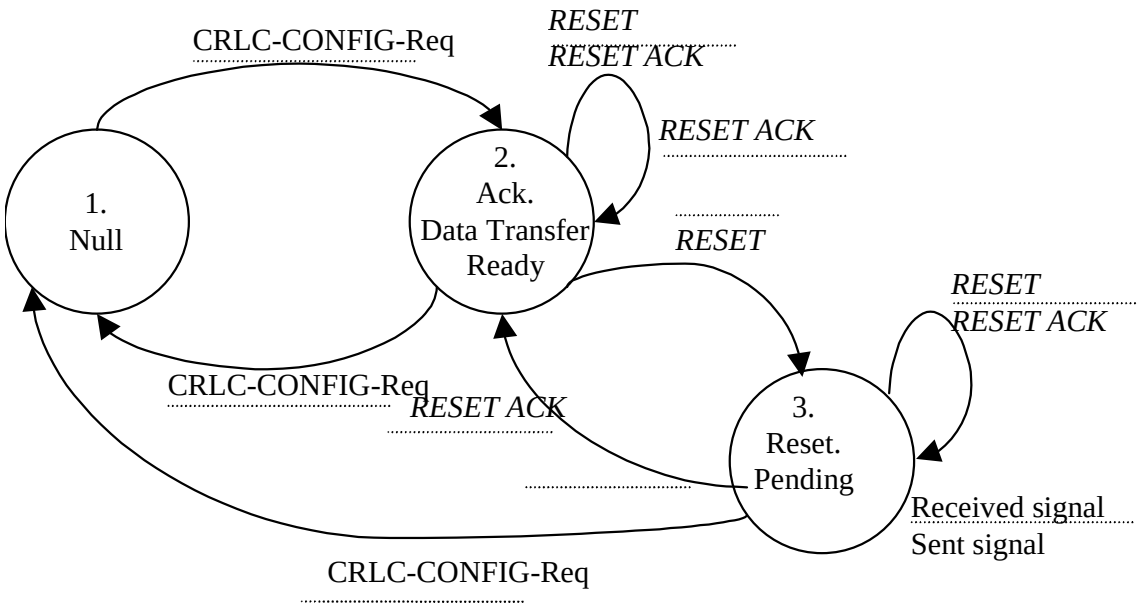


**Figure 9.18: The state model for the acknowledged mode entities when reset is performed**

### 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.

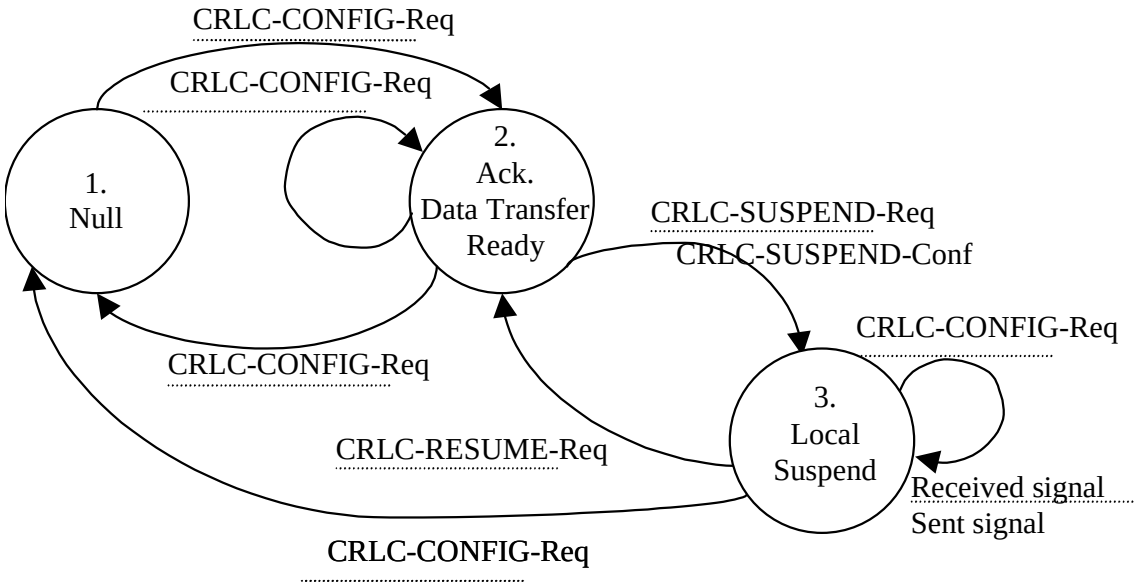


**Figure 9.19: The state model for the acknowledged mode entities when local suspend is performed**

**Figure** Error! No text of specified style in document.**.1**

**3GPP TSG-RAN WG2 Meeting #15**
**Sophia Antipolis, France, August 21st – 25th 2000**

**Document R2-001613**
*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 | 073 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

For submission to: TSG-RAN #9 *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:** Aug. 9th 2000

**Subject:** Clarification of the Length Indicators

**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:** Some clarifications for LI = 00..00 and LI = 11..11 are needed to eliminate the possible misunderstandings.

**Clauses affected:** 9.2.2.8

**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 ~~S~~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 – UM 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 next 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 last RLC PU, and 15-bit Length Indicators are used, the next Length Indicator shall be placed as the first Length Indicator in the next PU and have value LI=111 1111 1111 1011.

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.

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 UM 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.

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 an RLC SDU and there is no LI that indicates the end of the SDU in the previous RLC PDU. |
| **1111100** | 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** | AM PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU.<br>UM 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 last RLC PDU. |
| **111111111111100** | 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** | AM PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU.<br>UM 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 |

**3GPP RAN WG2#15**
**Sophia Antipolis, France, 21 – 25 August, 2000**

***Document R2-001755***
*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 | 076r1 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 2000-08-21

**Subject:** 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 CRLC-CONFIG-Req primitive is updated with an indication that an SRNS relocation is being performed. Upon reception of this indication, RLC shall be suspended.
2) Editorial modification creating a new chapter for the primitive parameters. Bullet point~~s~~ 9 ~~and 10 are~~is new.

~~3)When the UE transmits a RRC message on the UL CCCH it receives a response on DL (in UM). As the UE has not received any prior data in DL, it does not know the current RLC SN in DL. Thus, the UE does not know if the first received PDU is the first transmitted PDU or if some PDUs have been lost.~~
~~According to the current specification, the UE must then discard all data up to the first received LI, in order to assure that only complete SDUs are delivered to higher layer. To avoid that the RRC message in DL is discarded, the definition of LIs is modified.~~
~~3)~~The Special LI parts of CR 076 have been removed as these are now in another CR. Therefore bullet 10 in the new section 8.2 can be ignored.

**Clauses affected:** 8.1, 8.2(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.

# 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) ~~, SRNS Relocation (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, 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, 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.

~~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, all protocol parameters, variables and timers shall be set or reset 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.~~

~~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 is 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).~~

~~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.~~

## 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, all protocol parameters, variables and timers shall be set or reset 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.

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 is 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).

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 ~~SRNS Relocation~~Stop ~~parameter is only applicable for AM operation. T~~indicates that the RLC entity shall ~~be suspended (~~not ~~data transmission or reception~~ transmit or receive ~~of~~ RLC PDUs ~~by the RLC entity). and will be released by RRC once the relocation procedure has been completed. The radio bearer must also be configured to support lossless SRNS relocation.~~

~~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, the LI shall not be used.~~

**3GPP TSG RAN WG2#15**
**Sophia Antipolis, France, 21-25 August 2000**

***Document R2-001811***
*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 | 077r1 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

For submission to: TSG-RAN #9 *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:** 2000-08-21

**Subject:** Corrections to reset procedure and length indicator definitions

**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:**

- It has been clarified when RLC shall be re-established
- The case of two simultaneous reset procedures has been covered: VT(RST), Timer_RST shall not be reset when a RLC reset occurs which was initiated from the peer RLC entity
- General clarifications on reset procedure
- It has been specified what the one remaining octet shall be in case of LI='The last segment of an RLC SDU was one octet short of exactly filling the previous RLC PDU'
- General clarifications on length indicators

- Changes in Revision 1 are shown in yellow.

**Clauses affected:** 8.1, 9.2.2.8, 9.2.2.13, 9.3.3.2, 9.3.3.3, 9.4, 9.5, 11.4.5.1, 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.

# 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, | Data | Not Defined | Not Defined |
| **RLC-TR-DATA** | Data | Data | Not Defined | Not Defined |
| **CRLC-CONFIG** | E/R, 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, 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.

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 all protocol parameters, variables and timers shall be set to the configured value or reset 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 all protocol parameters, variables and timers shall be set to the 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. In the UE, RLC shall always be re-established when SRNS relocation has been performed.

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 is 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).

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.

### 9.2.2.8 Length Indicator (LI)

The Length Indicator is used to indicate, each time, the end of an *S*DU 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, the next Length Indicator, shall be placed as the first Length Indicator in the following~~next~~ 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~~last~~ RLC PU, and 15-bit Length Indicators are used, the next Length Indicator shall be placed as the first Length Indicator in the following~~next~~ PU and have value LI=111 1111 1111 1011. The remaining one octet in the previous RLC PU shall be ignored~~padding~~.

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 ~~there can be a maximum number of 15 SDUs in one AMD PDU~~ 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.

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

Length: 7bit

| Bit | Description |
|---|---|
| **0000000** | The previous RLC PDU was exactly filled with the last segment of a RLC SDU. |
| **1111100** | 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.<br>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. |

Length: 15bit

| Bit | Description |
|---|---|
| **000000000000000** | The previous RLC PDU was exactly filled with the last segment of a RLC SDU. |
| **111111111111011** | The last segment of an RLC SDU was one octet short of exactly filling the previous~~last~~ RLC PDU. The remaining one octet in the previous RLC PDU ~~was padding~~is ignored. |
| **111111111111100** | 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).l |
| **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. |

#### 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.

### 9.3.3 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) 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, increments the hyper frame number) and enters the acknowledged 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 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, 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.

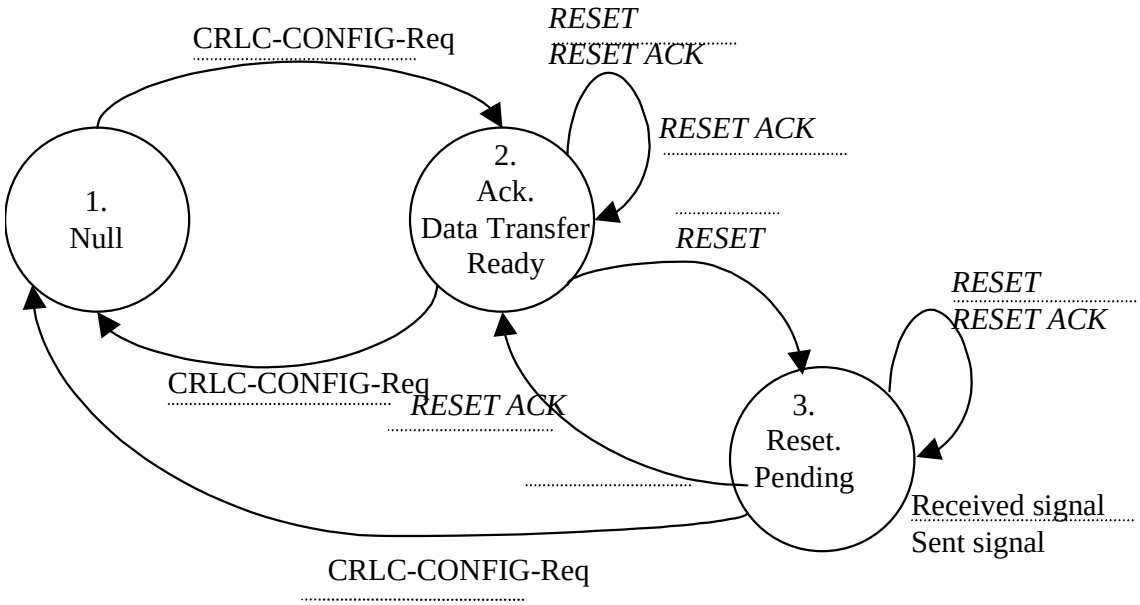


**Figure 9.18: The state model for the acknowledged mode entities when reset is performed**

### 9.3.3.4 Local Suspend State

Upon reception of CRLC-SUSPEND-Req from higher layer (RRC) 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) the RLC entity is resumed and the Data Transfer Ready state is entered.

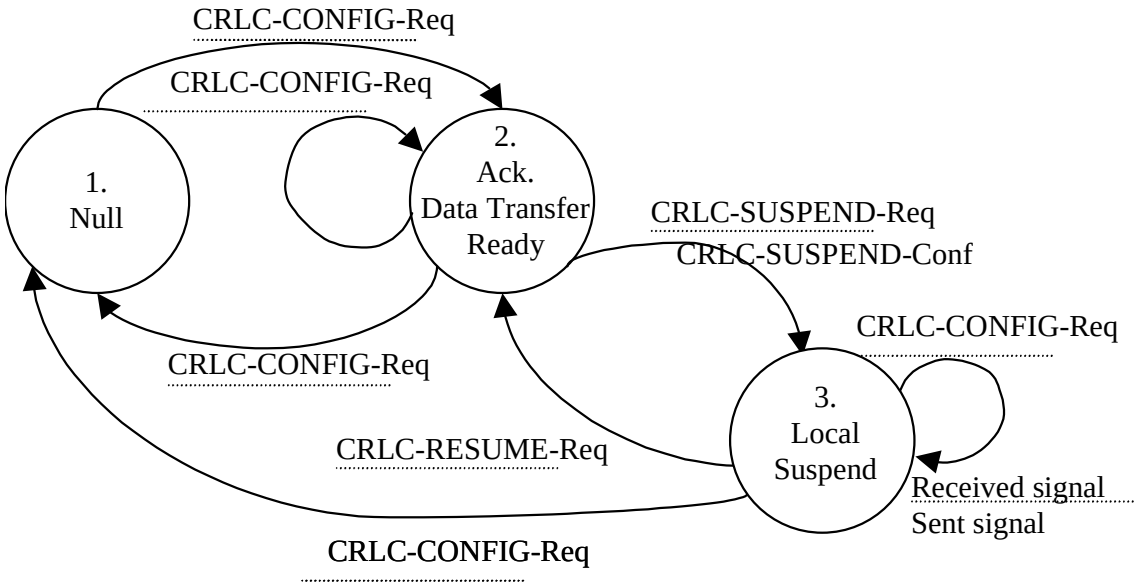


**Figure 9.19: The state model for the acknowledged mode entities when local suspend is performed**

## 9.4 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. 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 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 a WINDOW super-field.

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~~n~~ 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 upon the reception of a STATUS PDU which suggests the acknowledgement of a MRW command in the receiver or the occurrence of discarding new SDU. 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 ≥ 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.

## 9.5 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 or negative acknowledgement of the AMD PDU with SN=VT(S)-1 at the time the poll was transmitted. The value of the timer is signalled by RRC.

If the timer expires and no STATUS PDU containing an acknowledgement or negative acknowledgement of the AMD PDU with SN=VT(S)-1 at the time the poll was transmitted 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.

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 transmitted and when it expires EPC can start decrease (see subclause 9.7.3). 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 a poll is transmitted (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 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 last STATUS PDU in a status report is transmitted and no new status report containing the mentioned SUFIs can be transmitted before the timer has expired. The timer does not prohibit transmission of the SUFIs MRW, MRW_N_IFL, 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 a status report is transmitted and the timer is restarted. The value of the timer is signalled by RRC.

h) Timer_RST.

~~It~~This timer is used to detect the loss of RESET ACK PDU from the peer RLC entity. This timer is set when the RESET PDU is transmitted. ~~And i~~It will only be stopped upon reception of RESET ACK PDU, i.e. this timer is not stopped when a~~n~~ 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 last STATUS PDU of the status report is first transmitted. Each time the timer expires the status report is retransmitted and the timer is restarted (when the last STATUS PDU of the status report is retransmitted). It shall be stopped when a STATUS PDU is received that indicates that VR(R) ≥ SN_MRW. It shall also be stopped if a new MRW procedure is triggered while it is running. The value of the timer is signalled by RRC.

## 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.

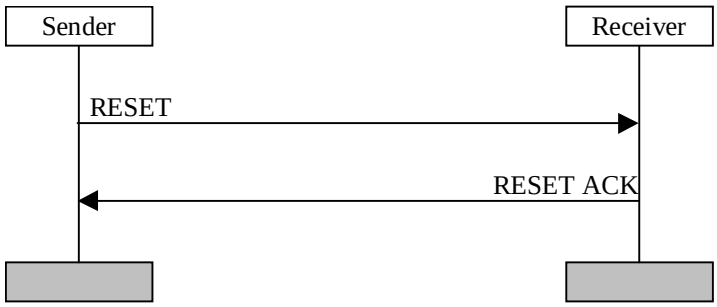


**Figure 11.4: RLC reset procedure**

## 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.

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. 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 ~~value~~ 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 ~~and increments the hyper frame number~~. 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 retransmit the RESET PDU and increase VT(RST) with 1.~~ The sender shall~~can~~ 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 Meeting WG2#15**
**Sophia Antipolis, France, 21 -25 August 2000**

***Document R2-001658***
*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** | **078** | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** 9/8/00

**Subject:** RLC modes for SHCCH

**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 SHCCH is presently defined to operate in transparent mode or unacknowledged mode in downlink direction. The transparent mode is proposed to be removed in order to minimise the amount of options that have to be supported

**Clauses affected:** 4.2.1.1, 4.2.1.2, 6.1, 11.1.2

**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.

#### 4.2.1.1 Transparent mode entities

Figure 4.2 below shows the model of two transparent mode peer entities.

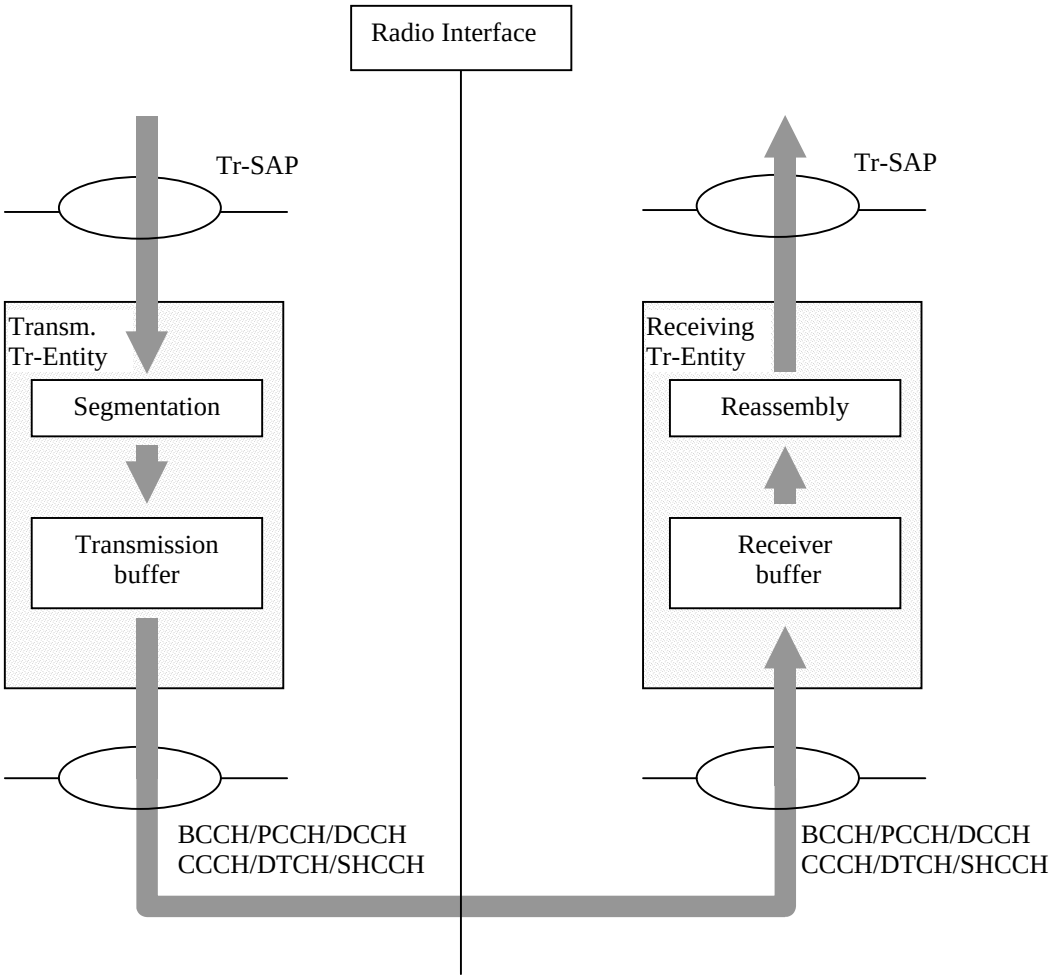


**Figure 4.2: Model of two transparent mode peer entities**

The transmitting Tr-entity receives SDUs from the higher layers through the Tr-SAP. RLC might segment the SDUs into appropriate RLC PDUs without adding any overhead. How to perform the segmentation is decided upon when the service is established. RLC delivers the RLC PDUs to MAC through either a BCCH, DCCH, PCCH, SHCCH or a DTCH. The CCCH and SHCCH also uses transparent mode, but only for the uplink. Which type of logical channel depends on if the higher layer is located in the control plane (BCCH, DCCH, PCCH, CCCH, SHCCH) or user plane (DTCH).

The Tr-entity receives PDUs through one of the logical channels from the MAC sublayer. RLC reassembles (if segmentation has been performed) the PDUs into RLC SDUs. How to perform the reassembling is decided upon when the service is established. RLC delivers the RLC SDUs to the higher layer through the Tr-SAP.

#### 4.2.1.2 Unacknowledged mode entities

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

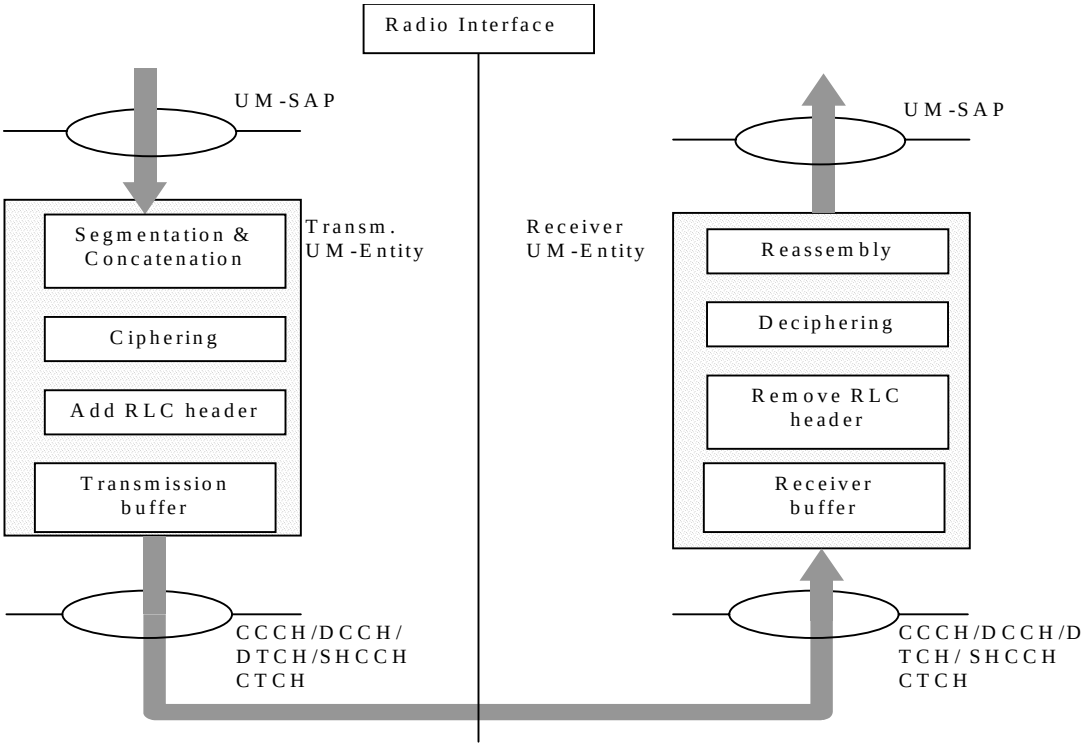


**Figure 4.3: Model of two unacknowledged mode peer entities**

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~~, a SHCCH (downlink only)~~, 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.

## 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 | - | - | - | - | + | + | - |
| **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 | - | - | + | + |

### 11.1.1 Purpose

The transparent mode data transfer procedure is used for transferring of data between two RLC peer entities, which are operating in transparent mode. Figure 11.1 below illustrates the elementary procedure for transparent mode data transfer. The sender can be either the UE or the network and the receiver is either the network or the UE.


**Figure 11.1: Transparent mode data transfer procedure**

### 11.1.2 Initiation

The sender initiates this procedure upon a request of transparent mode data transfer from higher layer. When the sender is in data transfer ready state it shall put the data received from the higher layer into TrD PDUs. If required RLC shall perform segmentation.

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

**3GPP RAN WG2#15**
**Sophia Antipolis, France, 21 – 25 August, 2000**

***Document* R2-001812**
*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 | 079 | Current Version: | 3.3.0 |
|---|---|---|---|---|

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

| For submission to: | TSG-RAN #9 | for approval | X | strategic | | (for SMG use only) |
|---|---|---|---|---|---|---|
| *list expected approval meeting # here ↑* | | for information | | non-strategic | | |

*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:** | 2000-08-21 |
|---|---|---|---|

**Subject:** CCCH in UM 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:**

1) When the UE transmits a RRC message on the UL CCCH it receives a response on DL (in UM). As the UE has not received any prior data in DL, it does not know the current RLC SN in DL. Thus, the UE does not know if the first received PDU is the first transmitted PDU or if some PDUs have been lost.
According to the current specification, the UE must then discard all data up to the first received LI, in order to assure that only complete SDUs are delivered to higher layer. To avoid that the RRC message in DL is discarded, the definition of LIs is modified.

**Clauses affected:** 8.1, 9.2.2.8

**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, 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, 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, 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.

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, all protocol parameters, variables and timers shall be set or reset 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.

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 is 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).

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 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.

### 9.2.2.8 Length Indicator (LI)

The Length Indicator is used to indicate, each time, the end of an *S*DU 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 – UM 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, the next Length Indicator, shall be placed as the first Length Indicator in the next 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 last RLC PU, and 15-bit Length Indicators are used, the next Length Indicator shall be placed as the first Length Indicator in the next PU and have value LI=111 1111 1111 1011.

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.

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 UM 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.

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

Length: 7bit

| Bit | Description |
|---|---|
| **0000000** | The previous RLC PDU was exactly filled with the last segment of a RLC SDU. |
| **1111100** | ~~Reserved (PDUs with this coding will be discarded by this version of the protocol).~~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** | AM PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU. UM PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol). |
| **1111111** | The rest of the RLC PDU is padding. |

Length: 15bit

| Bit | Description |
|---|---|
| **000000000000000** | The previous RLC PDU was exactly filled with the last segment of a RLC SDU. |
| **111111111111011** | The last segment of an RLC SDU was one octet short of exactly filling the last RLC PDU. |
| **111111111111100** | UMD PDU: ~~Reserved (PDUs with this coding will be discarded by this version of the protocol).~~ 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).l |
| **111111111111110** | AM PDU: The rest of the RLC PDU includes a piggybacked STATUS PDU. UM PDU: Reserved (PDUs with this coding will be discarded by this version of the protocol). |
| **111111111111111** | The rest of the RLC PDU is padding. |