**3GPP TSG-RAN WG2 Meeting #117-electronicR2-220xxxx**

**Online, February 21st - March 3rd, 2022**

|  |
| --- |
| *CR-Form-v12.2* |
| **CHANGE REQUEST** |
|  |
|  | **38.322** | **CR** | **0045** | **rev** | **1** | **Current version:** | **16.2.0** |  |
|  |
| *For* [*HELP*](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **x** | Radio Access Network | **x** | Core Network |  |

|  |
| --- |
|  |
| ***Title:***  | Introduction of NR MBS into 38.322 |
|  |  |
| ***Source to WG:*** | vivo, Huawei, HiSilicon |
| ***Source to TSG:*** | R2 |
|  |  |
| ***Work item code:*** | NR\_MBS-Core |  | ***Date:*** | 2022-03-11 |
|  |  |  |  |  |
| ***Category:*** | ***B*** |  | ***Release:*** | Rel-17 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** | Introduction of MBS support in NR. In this CR, the following RAN2 agreements related to RLC are captured:RAN2#112-e agreements:* RLC UM is supported for PTM transmission of NR MBS.

RAN2#113-e agreements:* The two-step based approach (i.e. BCCH and MCCH) as adopted by LTE SC-PTM is reused for the transmission of PTM configuration for NR MBS delivery mode 2.

RAN2#115-e agreements:* MTCH is specified for PTM transmission of NR MBS.

RAN2#116bis-e agreements:* For both multicast and broadcast, the initial value of RX\_Next\_Highest for broadcast is set to the SN of the first received UMD PDU containing an SN.
* For both multicast and broadcast, it is up to UE implementation to set the initial value of RX\_Next\_Reassembly to a value before RX\_Next\_Highest.
 |
|  |  |
| ***Summary of change:*** | In subclause 3.2, added abbreviations for MBS, MCCH, and MTCH.In subclause 4.2.1.2.1, added clarification that a UM RLC entity can be configured for MCCH or MTCH. In subclause 7.1, added initialization for RX\_Next\_Reassembly and RX\_Next\_Highest when the receiving UM RLC entity is configured for MCCH or MTCH. |
|  |  |
| ***Consequences if not approved:*** | NR MBS is not supported in Rel-17. |
|  |  |
| ***Clauses affected:*** | 3.2, 4.2.1.2.1, 7.1 |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** | **X** |  |  Other core specifications  | TS 38.300 CR 0342 r9TS 38.304 CR 0221 r5TS 38.306 CR 0685TS 38.321 CR 1184 r1TS 38.323 CR 0088TS 38.331 CR 2949 r1TS 37.324 CR 0022 r1 |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** | Revision of R2-2202277. |

*First Modified Subclause*

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

AM Acknowledged Mode

AMD AM Data

ARQ Automatic Repeat request

gNB NR Node B

MBS Multicast/Broadcast Services

MCCH MBS Control Channel

MTCH MBS Traffic Channel

PDU Protocol Data Unit

RLC Radio Link Control

SBCCH Sidelink Broadcast Control Channel

SCCH Sidelink Control Channel

SDU Service Data Unit

SN Sequence Number

STCH Sidelink Traffic Channel

TB Transport Block

TM Transparent Mode

TMD TM Data

UE User Equipment

UM Unacknowledged Mode

UMD UM Data

*Next Modification Subclause*

##### 4.2.1.2.1 General

An UM RLC entity can be configured to submit/receive RLC PDUs through the following logical channels:

- DL/UL DTCH, SCCH, STCH, MCCH, and MTCH.



Figure 4.2.1.2.1-1: Model of two unacknowledged mode peer entities

An UM RLC entity submits/receives the following RLC data PDU:

- UMD PDU.

An UMD PDU contains either one complete RLC SDU or one RLC SDU segment.

NOTE: For groupcast and broadcast of NR sidelink communication only uni-directional UM mode is supported.

*Next Modification Subclause*

## 7.1 State variables

This sub clause describes the state variables used in AM and UM entities in order to specify the RLC protocol. The state variables defined in this clause are normative.

All state variables and all counters are non-negative integers.

All state variables related to AM data transfer can take values from 0 to 4095 for 12 bit SN or from 0 to 262143 for 18 bit SN. All arithmetic operations contained in the present document on state variables related to AM data transfer are affected by the AM modulus (i.e. final value = [value from arithmetic operation] modulo 4096 for 12 bit SN and 262144 for 18 bit SN).

All state variables related to UM data transfer can take values from 0 to 63 for 6 bit SN or from 0 to 4095 for 12 bit SN. All arithmetic operations contained in the present document on state variables related to UM data transfer are affected by the UM modulus (i.e. final value = [value from arithmetic operation] modulo 64 for 6 bit SN and 4096 for 12 bit SN).

When performing arithmetic comparisons of state variables or SN values, a modulus base shall be used.

TX\_Next\_Ack and RX\_Next shall be assumed as the modulus base at the transmitting side and receiving side of an AM RLC entity, respectively. This modulus base is subtracted from all the values involved, and then an absolute comparison is performed (e.g. RX\_Next <= SN < RX\_Next + AM\_Window\_Size is evaluated as [RX\_Next – RX\_Next] modulo 2[*sn-FieldLength*] <= [SN – RX\_Next] modulo 2[*sn-FieldLength*] < [RX\_Next + AM\_Window\_Size – RX\_Next] modulo 2[*sn-FieldLength*]), where *sn-FieldLength* is 12 or 18 for 12 bit SN and 18 bit SN, respectively.

RX\_Next\_Highest– UM\_Window\_Size shall be assumed as the modulus base at the receiving UM RLC entity. This modulus base is subtracted from all the values involved, and then an absolute comparison is performed (e.g. (RX\_Next\_Highest– UM\_Window\_Size) <= SN < RX\_Next\_Highest is evaluated as [(RX\_Next\_Highest– UM\_Window\_Size) – (RX\_Next\_Highest– UM\_Window\_Size)] modulo 2[*sn-FieldLength*] <= [SN – (RX\_Next\_Highest– UM\_Window\_Size)] modulo 2[*sn-FieldLength*] < [RX\_Next\_Highest– (RX\_Next\_Highest– UM\_Window\_Size)] modulo 2[*sn-FieldLength*]), where *sn-FieldLength* is 6 or 12 for 6 bit SN and 12 bit SN, respectively.

The transmitting side of each AM RLC entity shall maintain the following state variables:

a) TX\_Next\_Ack – Acknowledgement state variable

This state variable holds the value of the SN of the next RLC SDU for which a positive acknowledgment is to be received in-sequence, and it serves as the lower edge of the transmitting window. It is initially set to 0, and is updated whenever the AM RLC entity receives a positive acknowledgment for an RLC SDU with SN = TX\_Next\_Ack.

b) TX\_Next – Send state variable

This state variable holds the value of the SN to be assigned for the next newly generated AMD PDU. It is initially set to 0, and is updated whenever the AM RLC entity constructs an AMD PDU with SN = TX\_Next and contains an RLC SDU or the last segment of a RLC SDU.

c) POLL\_SN – Poll send state variable

This state variable holds the value of the highest SN of the AMD PDU among the AMD PDUs submitted to lower layer when POLL\_SN is set according to sub clause 5.3.3.2. It is initially set to 0.

The transmitting side of each AM RLC entity shall maintain the following counters:

a) PDU\_WITHOUT\_POLL – Counter

This counter is initially set to 0. It counts the number of AMD PDUs sent since the most recent poll bit was transmitted.

b) BYTE\_WITHOUT\_POLL – Counter

This counter is initially set to 0. It counts the number of data bytes sent since the most recent poll bit was transmitted.

c) RETX\_COUNT – Counter

This counter counts the number of retransmissions of an RLC SDU or RLC SDU segment (see clause 5.3.2). There is one RETX\_COUNT counter maintained per RLC SDU.

The receiving side of each AM RLC entity shall maintain the following state variables:

a) RX\_Next – Receive state variable

This state variable holds the value of the SN following the last in-sequence completely received RLC SDU, and it serves as the lower edge of the receiving window. It is initially set to 0, and is updated whenever the AM RLC entity receives an RLC SDU with SN = RX\_Next.

b) RX\_Next\_Status\_Trigger – *t-Reassembly* state variable

This state variable holds the value of the SN following the SN of the RLC SDU which triggered *t-Reassembly*.

c) RX\_Highest\_Status – Maximum STATUS transmit state variable

This state variable holds the highest possible value of the SN which can be indicated by "ACK\_SN" when a STATUS PDU needs to be constructed. It is initially set to 0.

d) RX\_Next\_Highest – Highest received state variable

This state variable holds the value of the SN following the SN of the RLC SDU with the highest SN among received RLC SDUs. It is initially set to 0.

Each transmitting UM RLC entity shall maintain the following state variables:

a) TX\_Next – UM send state variable

This state variable holds the value of the SN to be assigned for the next newly generated UMD PDU with segment. It is initially set to 0, and is updated after the UM RLC entity submits a UMD PDU including the last segment of an RLC SDU to lower layers.

Each receiving UM RLC entity shall maintain the following state variables:

a) RX\_Next\_Reassembly – UM receive state variable

This state variable holds the value of the earliest SN that is still considered for reassembly. It is initially set to 0. For groupcast and broadcast of NR sidelink communication, it is initially set to the SN of the first received UMD PDU containing an SN. For the receiving UM RLC entity configured for MCCH or MTCH, it is up to UE implementation to set the initial value of RX\_Next\_Reassembly to a value before RX\_Next\_Highest.

b) RX\_Timer\_Trigger – UM *t-Reassembly* state variable

This state variable holds the value of the SN following the SN which triggered *t-Reassembly*.

c) RX\_Next\_Highest– UM receive state variable

This state variable holds the value of the SN following the SN of the UMD PDU with the highest SN among received UMD PDUs. It serves as the higher edge of the reassembly window. It is initially set to 0. For groupcast and broadcast of NR sidelink communication, it is initially set to the SN of the first received UMD PDU containing an SN. For the receiving UM RLC entity configured for MCCH or MTCH, it is initially set to the SN of the first received UMD PDU containing an SN.

*End of Changes*