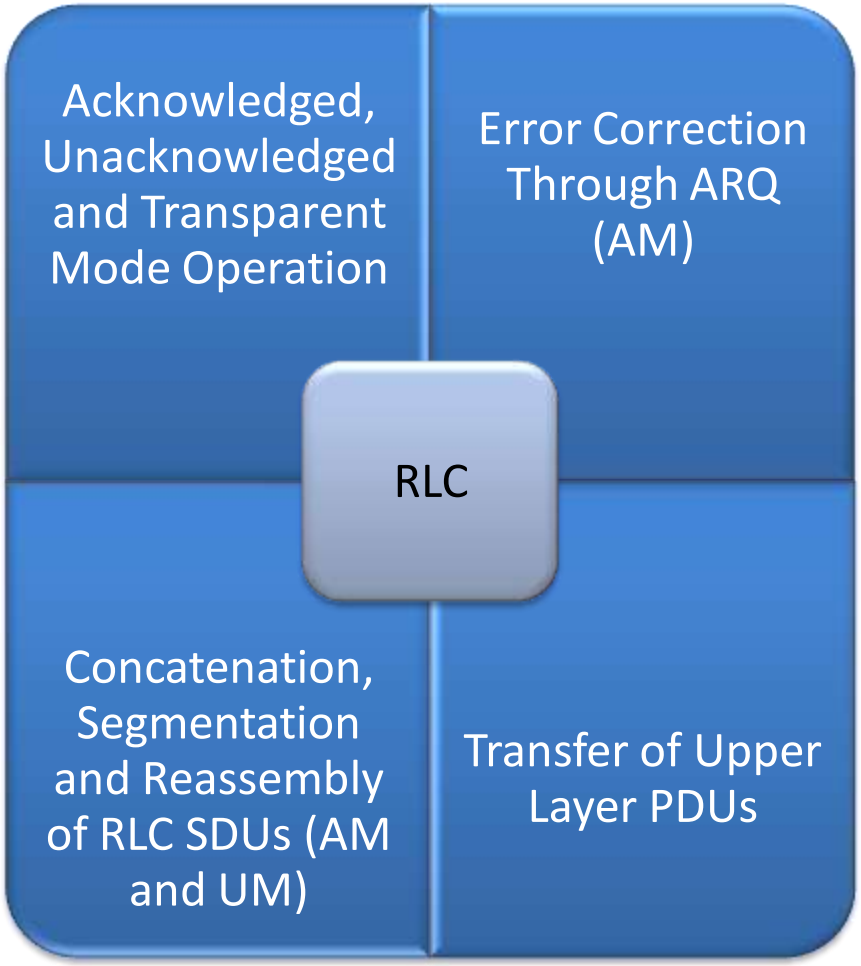
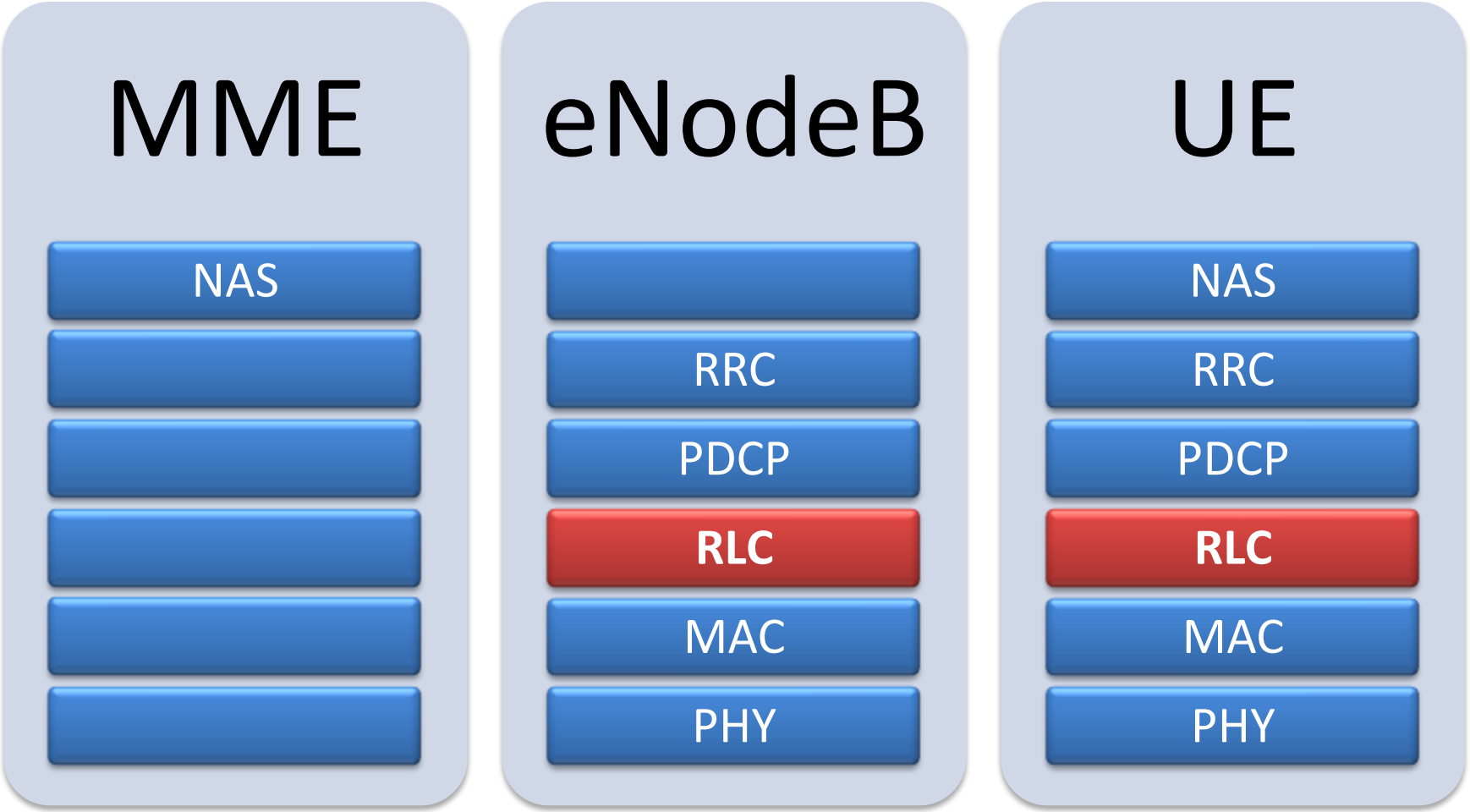
3GPP LTE Radio Link Control (RLC) Sub Layer

LTE RLC Sub Layer Functions

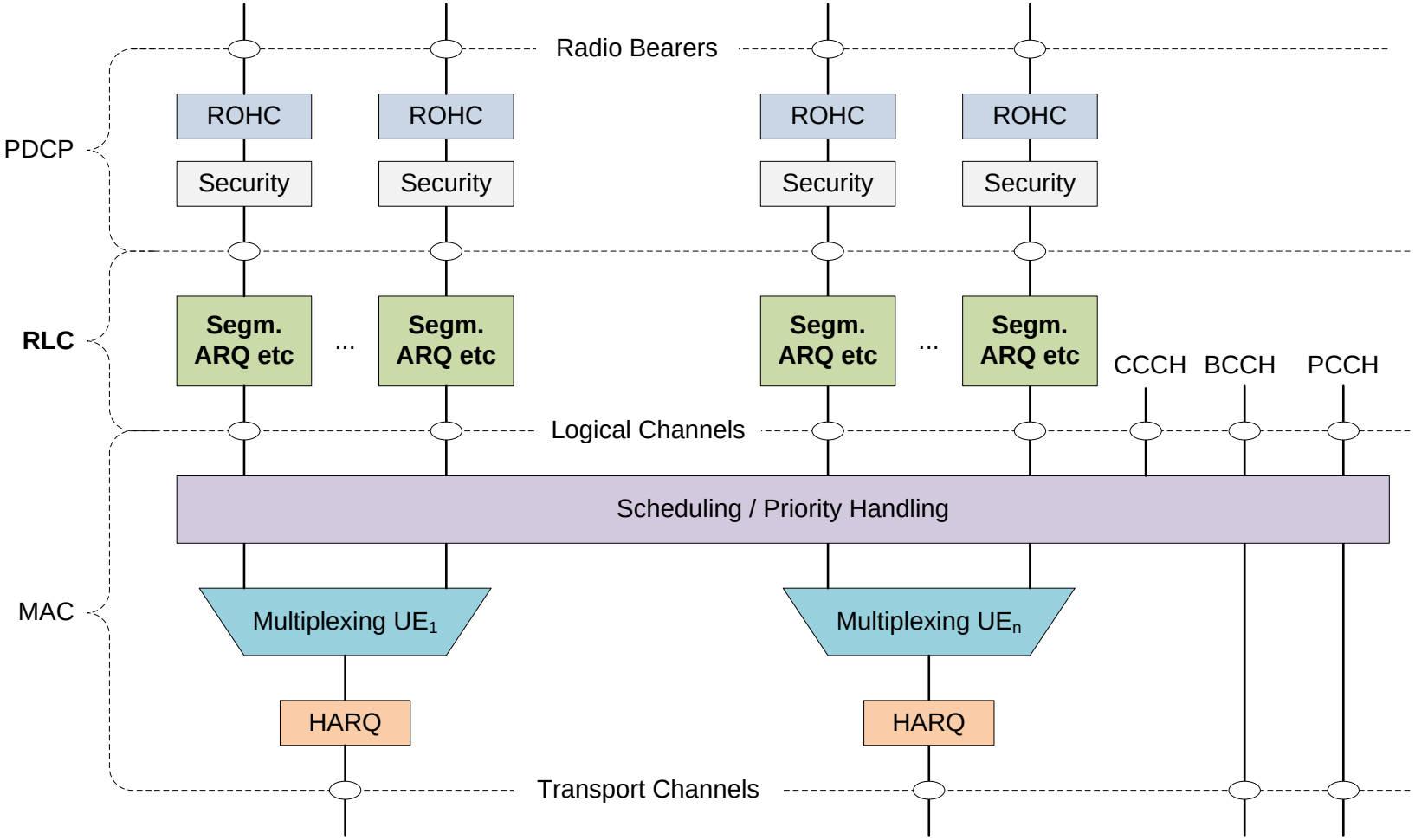


- Transfer of upper layer PDUs;
- Error correction through ARQ (only for AM data transfer)
- Concatenation, segmentation and reassembly of RLC SDUs (UM and AM)
- Re-segmentation of RLC data PDUs (AM)
- Reordering of RLC data PDUs (UM and AM);
- Duplicate detection (UM and AM);
- RLC SDU discard (UM and AM)
- RLC re-establishment
- Protocol error detection and recovery

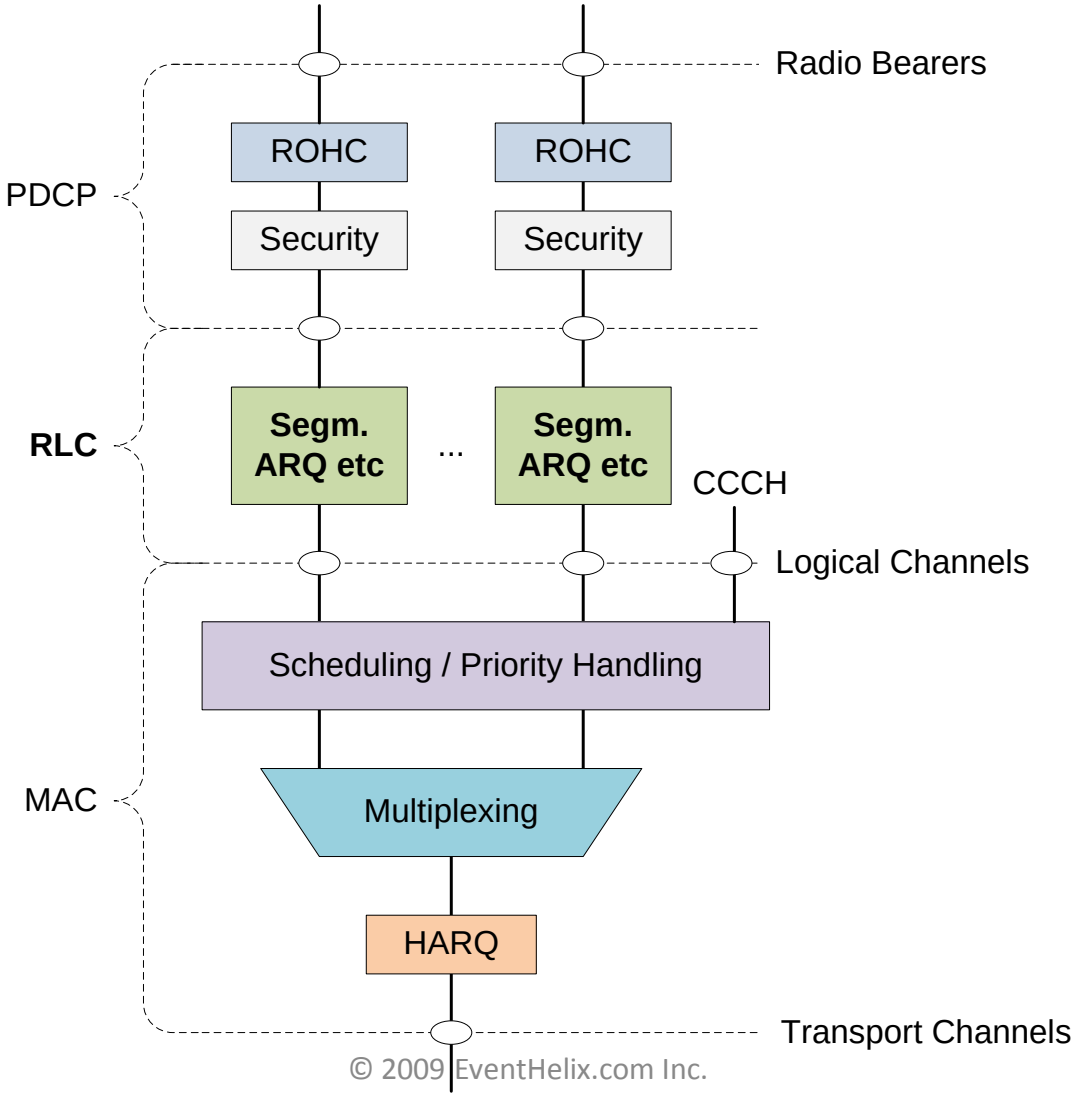
RLC in the LTE Protocol Stack



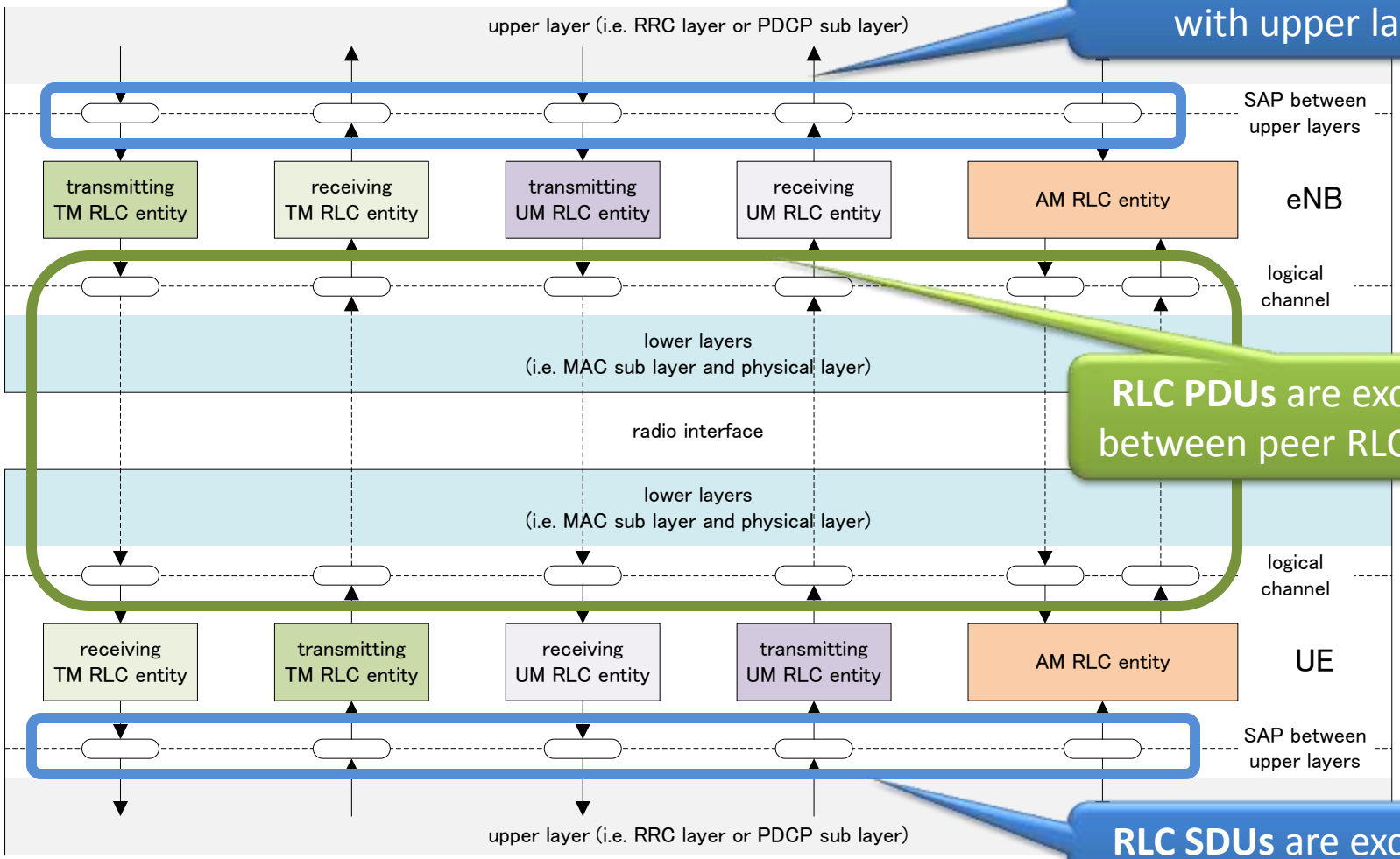
Downlink RLC Sub Layer Interfaces



Uplink RLC Sub Layer Interfaces



LTE RLC Sub Layer



RLC SDUs are exchanged with upper layers

RLC PDUs are exchanged between peer RLC entities

RLC SDUs are exchanged with upper layers

RLC Modes

Transparent Mode

- No segmentation and reassembly of RLC SDUs
- No RLC headers are added
- No delivery guarantees
- Suitable for carrying voice

Unacknowledged Mode

- Segmentation and reassembly of RLC SDUs
- RLC Headers are added
- No delivery guarantees
- Suitable for carrying streaming traffic

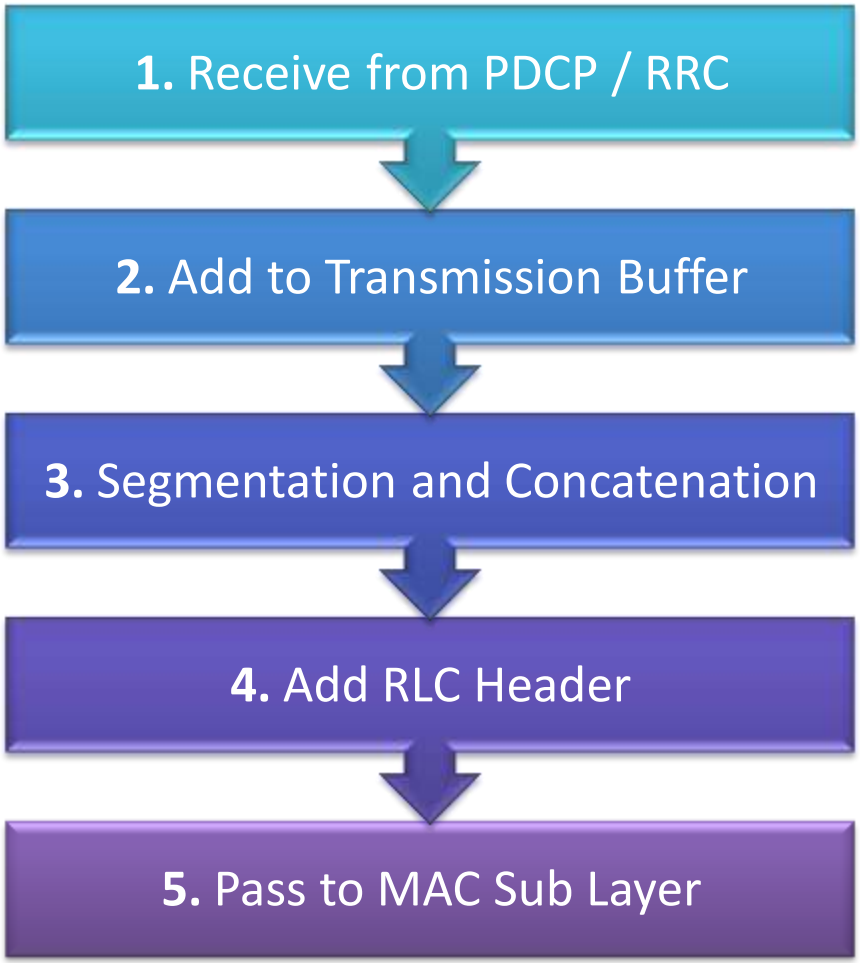
Acknowledged Mode

- Segmentation and reassembly of RLC SDUs
- RLC Headers are added
- Reliable in sequence delivery service
- Suitable for carrying TCP traffic

3GPP LTE Radio Link Control (RLC) Sub Layer

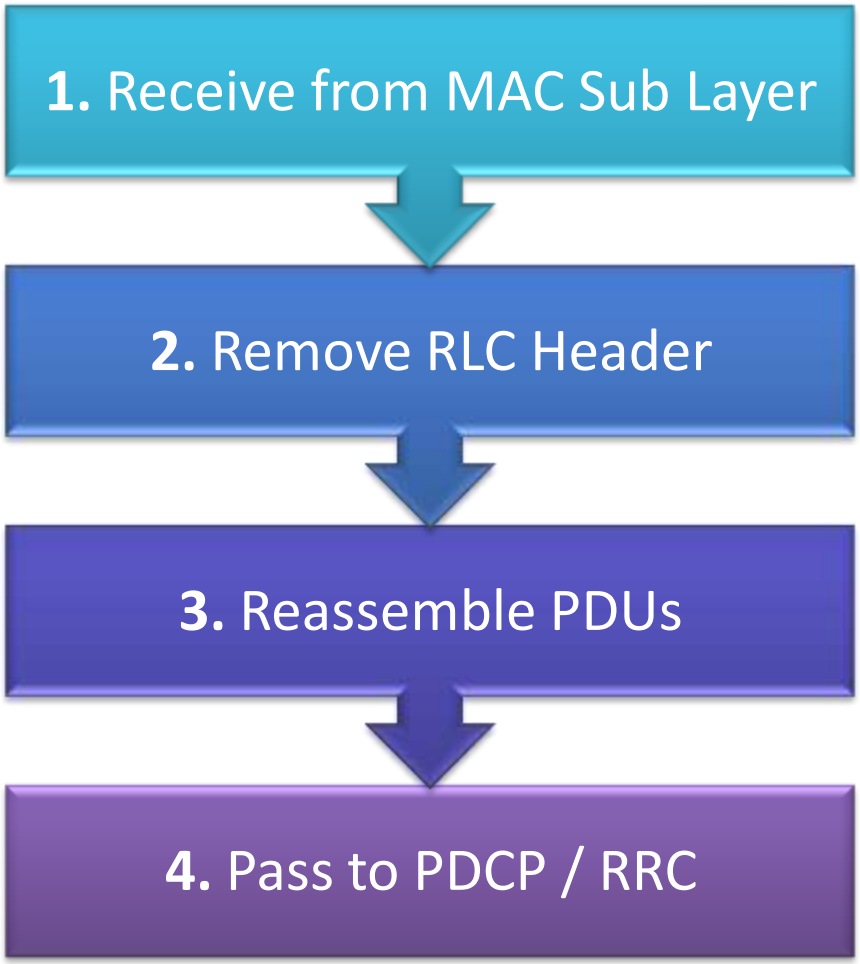
UNACKNOWLEDGED MODE

Unacknowledged Mode Transmit Overview



1. Receive the upper layer SDU from PDCP or RRC.
2. Add the SDU to the transmit buffer.
3. Segment the SDU into RLC PDUs when the MAC scheduler permits transmission.
4. Add the RLC header to the RLC PDU.
5. Pass the RLC PDUs to MAC for transmission over the air.

Unacknowledged Mode Receive Overview



1. The MAC layer passes the received RLC PDUs to the RLC layer.
2. The RLC layer removes the RLC header.
3. The RLC layer assembles an upper layer SDUs if receipt of an RLC PDU completes the assembly of the SDU.
4. Pass the assembled SDUs to the PDCP or RRC layers.

Unacknowledged Mode State Variables

VT(US) Send State Variable

- Holds the value of the SN to be assigned for the next newly generated UMD PDU.
- It is initially set to 0, and is updated whenever the UM RLC entity delivers an UMD PDU with SN = VT(US).

VR(UR) UM receive state variable

- Holds the value of the SN of the earliest UMD PDU that is still considered for reordering.
- It is initially set to 0.

VR(UX) UM *t-Reordering* state variable

- This state variable holds the value of the SN following the SN of the UMD PDU which triggered *t-Reordering*.

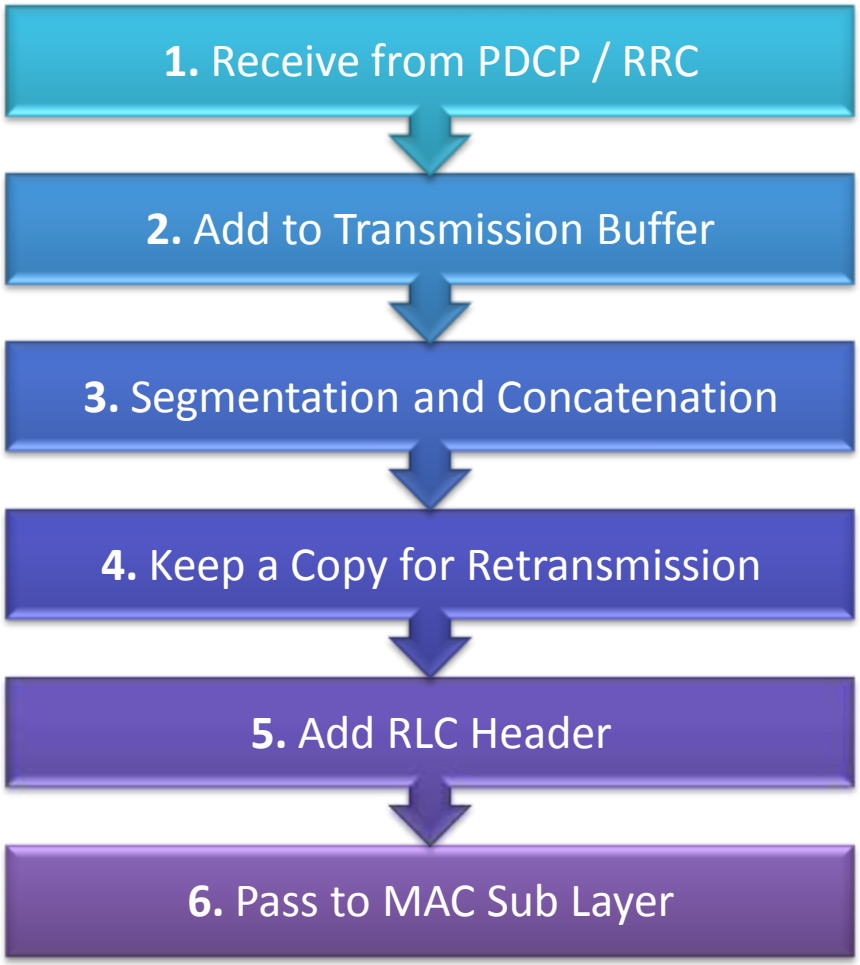
VR(UH) UM highest received state variable

- This state variable holds the value of the SN following the SN of the UMD PDU with the highest SN among received UMD PDUs
- Serves as the higher edge of the reordering window. It is initially set to 0.

3GPP LTE Radio Link Control (RLC) Sub Layer

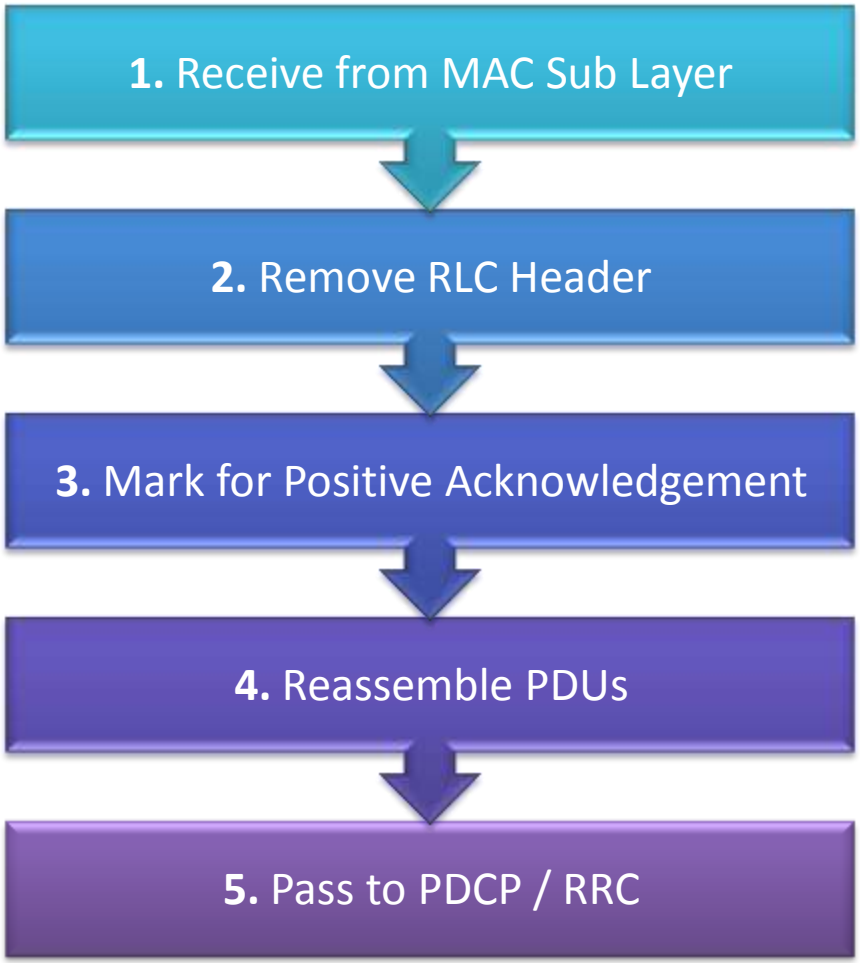
ACKNOWLEDGED MODE

Acknowledged Mode Transmit Overview



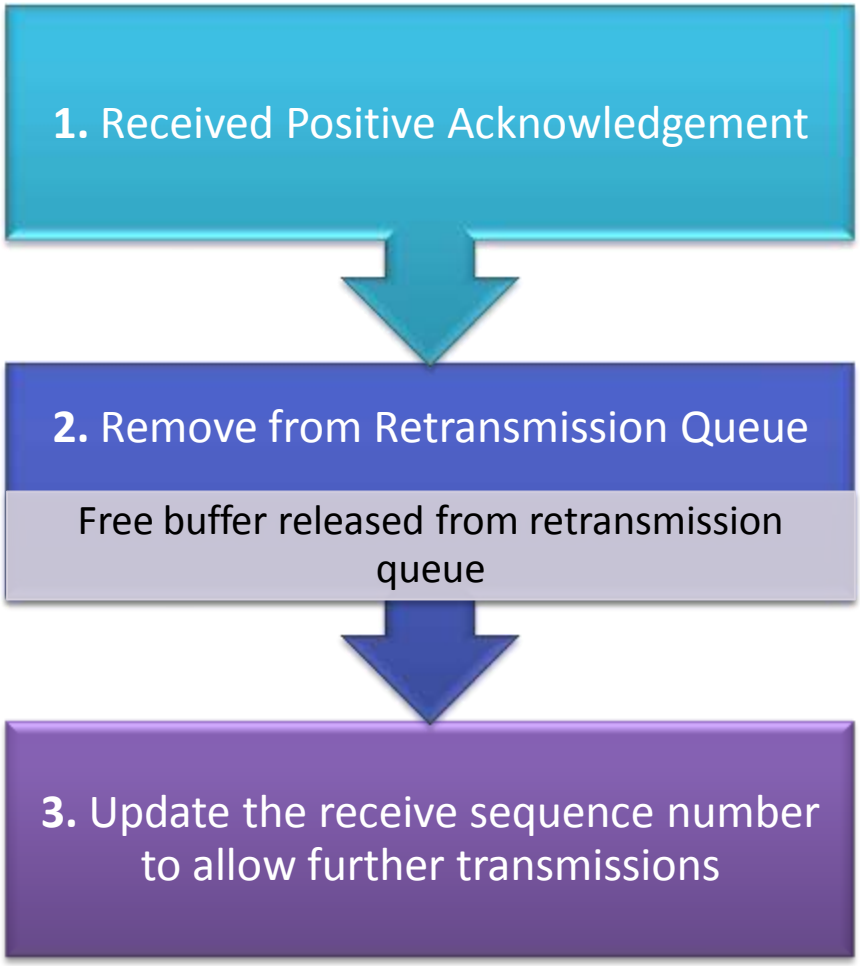
1. Receive the upper layer SDU from PDCP or RRC.
2. Add the SDU to the transmit buffer.
3. Segment the SDU into RLC PDUs when the MAC scheduler permits transmission.
4. Make a copy of the transmit buffer for possible retransmissions.
5. Add the RLC header to the RLC PDUs.
6. Pass the RLC PDUs to MAC for transmission over the air.

Acknowledged Mode Receive Overview



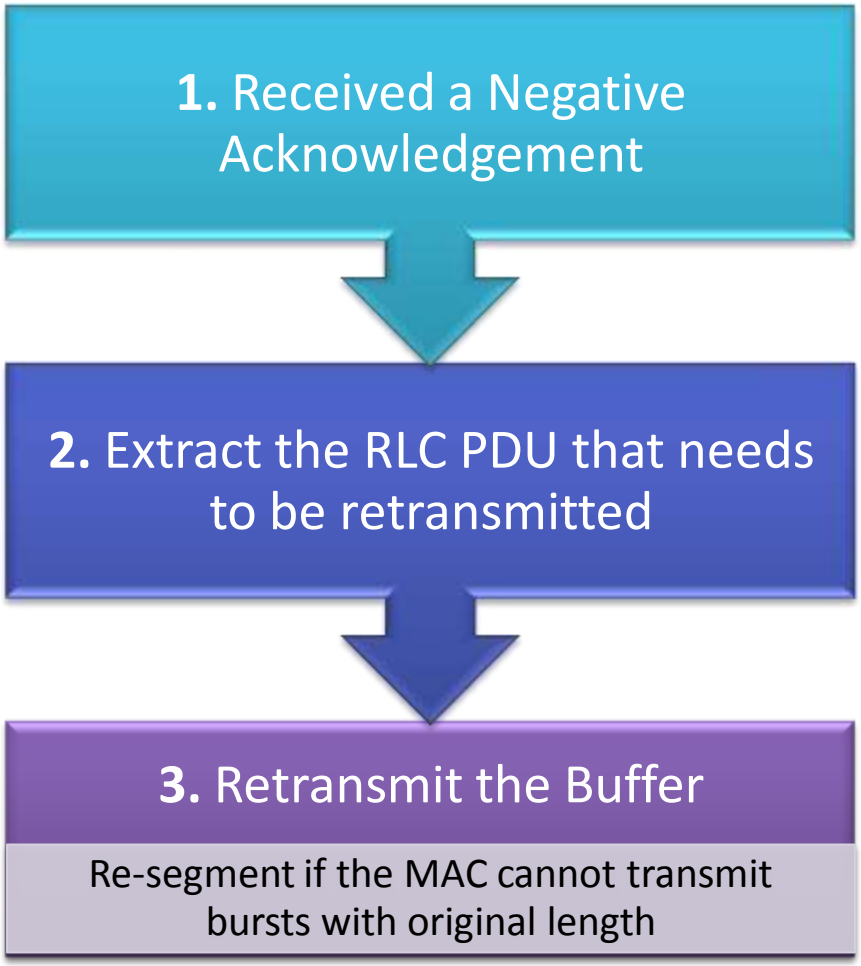
1. The MAC layer passes the received RLC PDU to the RLC layer.
2. The RLC layer removes the RLC header.
3. The RLC PDU is received correctly, so mark the block for positive acknowledgement.
   - Acknowledgements are sent periodically to the remote peer.
4. The RLC layer assembles an upper layer SDUs if receipt of an RLC PDU completes the assembly of the SDU.
5. Pass the assembled SDUs to the PDCP or RRC layers.

Acknowledged Mode: Received Positive Acknowledgement - Overview



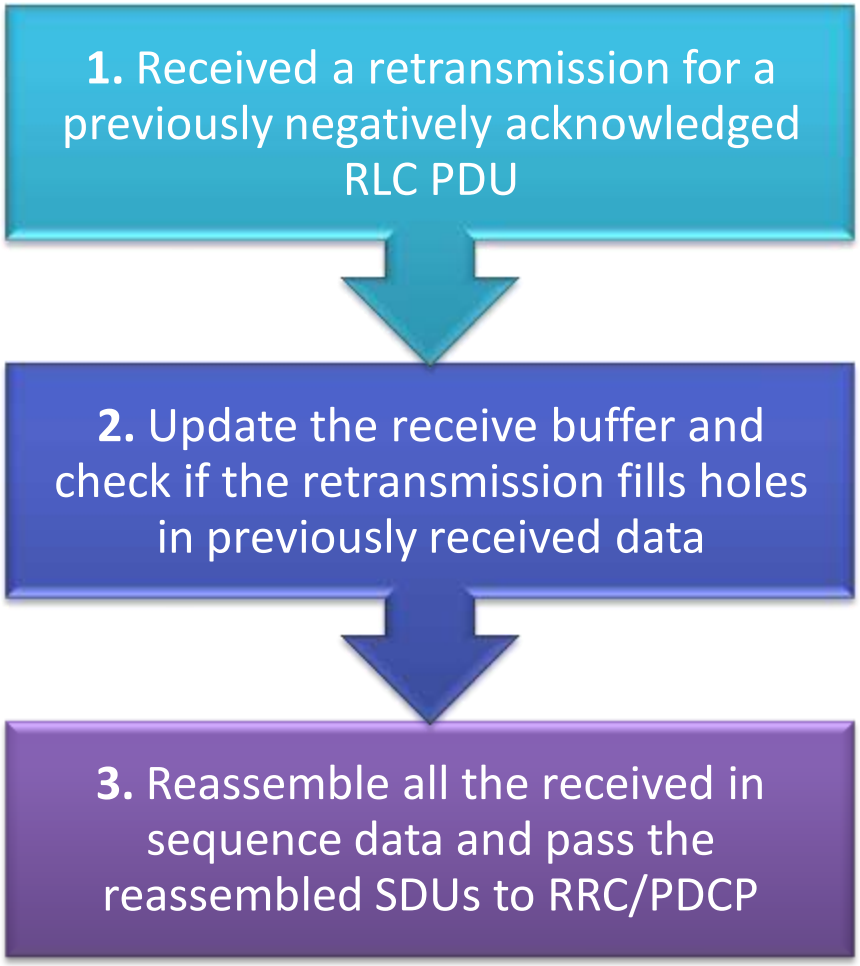
1. A positive acknowledgement is received from the remote end.
2. Access the retransmission queue and remove the buffer as it has been acknowledged.
3. Update the received sequence numbers to advance the sliding window.

Acknowledged Mode: Received Negative Acknowledgement - Overview



1. A negative acknowledgement is received from the remote end.
2. Access the retransmission queue and extract the buffer for retransmission.
3. Retransmit the buffer
   - If MAC does not support the original transmission rate, re-segment the RLC block into the smaller available block size

Acknowledged Mode: Received Retransmission - Overview



1. A retransmission for a previously negatively acknowledged RLC PDU is received.
2. Update the received data buffer
   - The received buffer may fill a hole in the previously received data.
3. Assemble all the in sequence received data into SDUs
   - Pass the received SDUs to the RRC or PDCP layers.

Acknowledged Mode State Variables

Transmit

- **VT(A):** Acknowledged State Variable
- **VT(MS):** Maximum Send State Variable
- **VT(S):** Send State Variable
- **POLL_SN:** Poll Send State Variable
- **PDU_WITHOUT_POLL** Counter
- **BYTE_WITHOUT_POLL** Counter
- **RETX_COUNT** Counter

Receive

- **VR(R):** Receive State Variable
- **VR(MR):** Maximum Accepted Receive State Variable
- **VR(X):** Reordering State Variable
- **VR(MS):** Maximum STATUS Transmit State Variable
- **VR(H):** Highest Received State Variable

Acknowledged Mode Transmit State Variables

VT(A) Acknowledgement state variable

- Holds the value of the SN of the next AMD PDU for which a positive acknowledgment is to be received in-sequence
- **Serves as the lower edge of the transmitting window.**
- It is initially set to 0, and is updated whenever a positive acknowledgment for an AMD PDU with SN = VT(A) is received

VT(MS) Maximum send state variable

- This state variable equals VT(A) + AM_Window_Size
- **It serves as the higher edge of the transmitting window.**

VT(S) 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 delivers an AMD PDU with SN = VT(S).

POLL_SN Poll send state variable

- This state variable holds the value of VT(S)-1 upon the most recent transmission of a RLC data PDU with the poll bit set to “1”. It is initially set to 0.

Acknowledged Mode Transmit Procedure

The Transmit AM RLC entity maintains a transmitting window such that

Transmit Serial Number (SN) falls within the Transmit window [**VT(A) <= SN < VT(MS)**]

Deliver a new AMD PDU to lower layer, the Transmit AM RLC entity shall:

AMD PDU SN= VT(S)

VT(S) = VT(S) + 1

Transmit AM RLC entity receives a STATUS PDU with positive acknowledgement for a RLC data PDU

VT(A) = Smallest SN awaiting acknowledgement.

Positive acknowledgements have been received for all AMD PDUs associated with a transmitted RLC SDU:

Deliver the RLC SDU to the upper layers

Acknowledged Mode Receive State Variables

VR(R) Receive state variable

- Holds the value of the SN following the last in-sequence completely received AMD PDU
- **It serves as the lower edge of the receiving window.**
- It is initially set to 0, and is updated whenever an AMD PDU with SN = VR(R) is received.

VR(MR) Maximum acceptable receive state variable

- This state variable equals VR(R) + AM_Window_Size, and it holds the value of the SN of the first AMD PDU that is beyond the receiving window
- **Serves as the higher edge of the receiving window**.

VR(X) *t-Reordering* state variable

- This state variable holds the value of the SN following the SN of the RLC data PDU which triggered *t-Reordering*.

VR(MS) 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.

VR(H) Highest received state variable

- This state variable holds the value of the SN following the SN of the RLC data PDU with the highest SN among received RLC data PDUs. It is initially set to 0.

RLC Configurable Parameters

maxRetxThreshold	• This parameter is used by the transmitting side of each AM RLC entity to limit the number of retransmissions of an AMD PDU.
pollPDU	• This parameter is used by the transmitting side of each AM RLC entity to trigger a poll for every *pollPDU* PDUs.
pollByte	• This parameter is used by the transmitting side of each AM RLC entity to trigger a poll for every *pollByte* bytes.
sn-FieldLength	• This parameter gives the UM SN field size in bits.

RLC Priority

- The transmitting side of an AM RLC entity shall prioritize transmission of RLC control PDUs over RLC data PDUs.
- The transmitting side of an AM RLC entity shall prioritize retransmission of RLC data PDUs over transmission of new AMD PDUs.

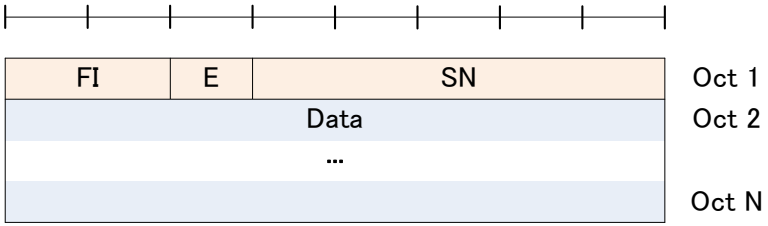
3GPP LTE Radio Link Control (RLC) Sub Layer

RLC PDU FORMATS

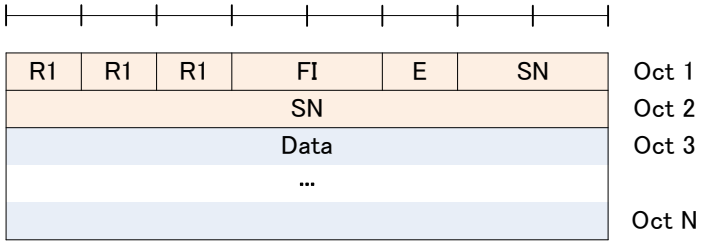
TMD PDU

- Transparent Mode PDUs just contain Data
- No headers are included

UMD PDU-1



UMD PDU with 5 bit Serial Number

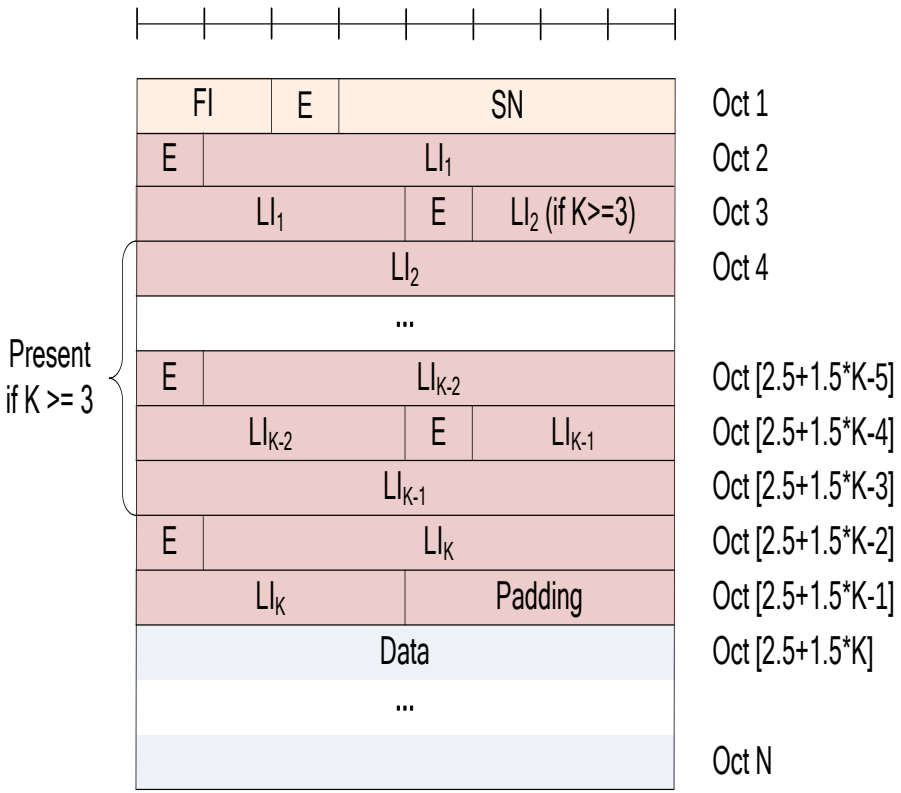


UMD PDU with 10 bit Serial Number

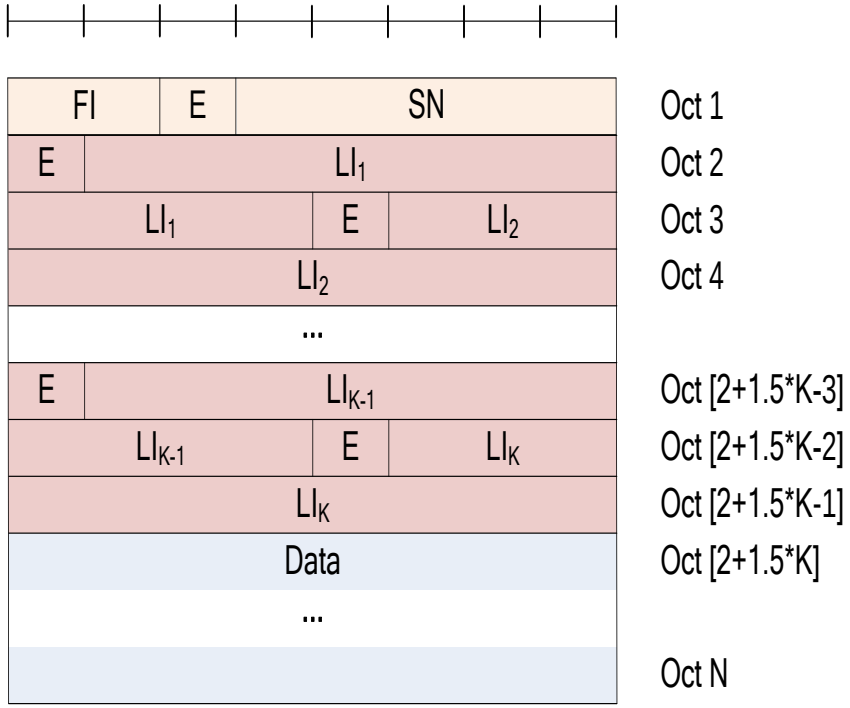
- An UM RLC entity is configured by RRC to use either a 5 bit SN or a 10 bit SN.
- An UMD PDU header needs to be extended when more than multiple Data field elements need to be sent.
  - In that which case an E and a LI are present for every Data field element except the last.
  - Furthermore, when an UMD PDU header consists of an odd number of LI(s), four padding bits follow after the last LI.
  - See next two slides

UMD PDU -2

UMD PDU (5 bit SN) with Odd Number of LIs

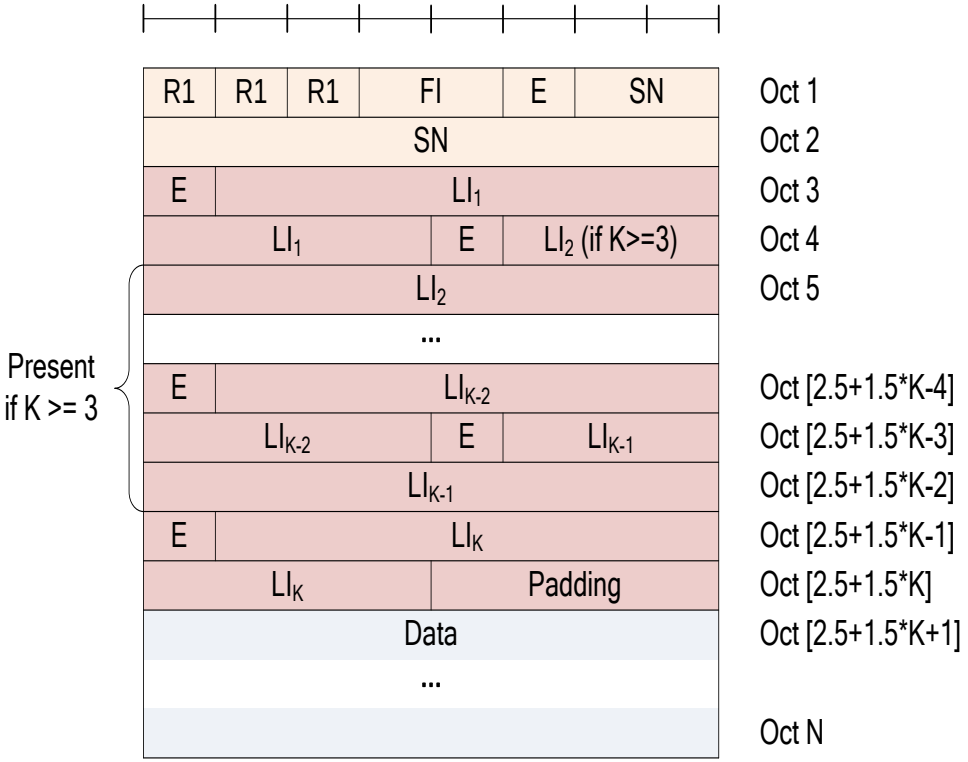


UMD PDU (5 bit SN) with Even Number of LIs

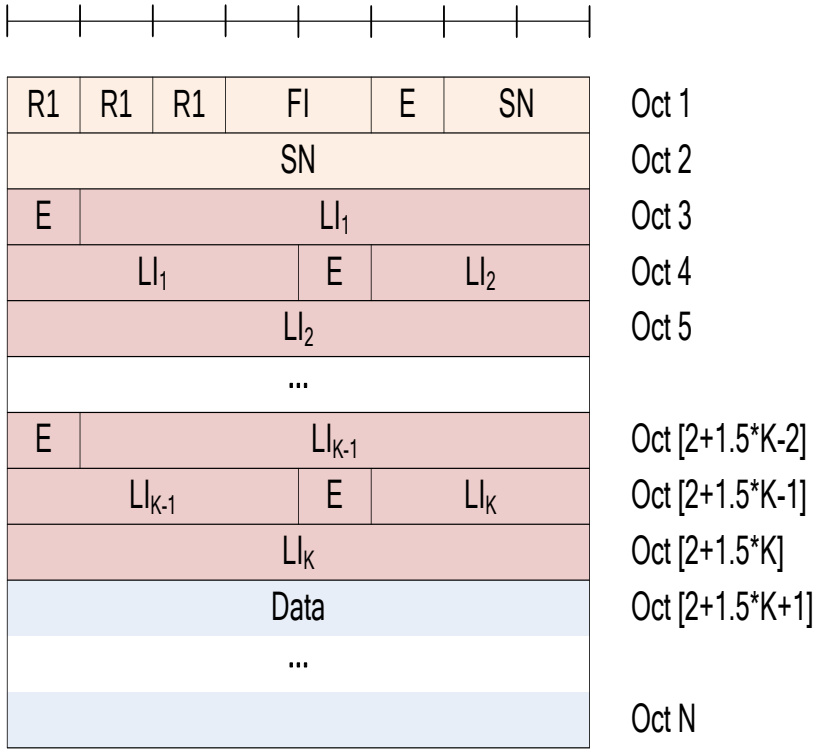


UMD PDU -3

UMD PDU (10 bit SN) with Odd Number of LIs



UMD PDU (10 bit SN) with Even Number of LIs



UMD PDU Fields - 1

Sequence Number (SN) – 5 or 10 bit

- The SN field indicates the sequence number of the corresponding UMD PDU.
- The sequence number is incremented by one for every UMD PDU.

Extension bit (E) – 1 bit

- The E field indicates whether Data field follows or a set of E field and LI field follows.

Framing Info (FI) – 2 bit

- The FI field indicates whether a RLC SDU is segmented at the beginning and/or at the end of the Data field.
- Specifically, the FI field indicates whether the first byte of the Data field corresponds to the first byte of a RLC SDU, and whether the last byte of the Data field corresponds to the last byte of a RLC SDU.

UMD PDU Fields - 2

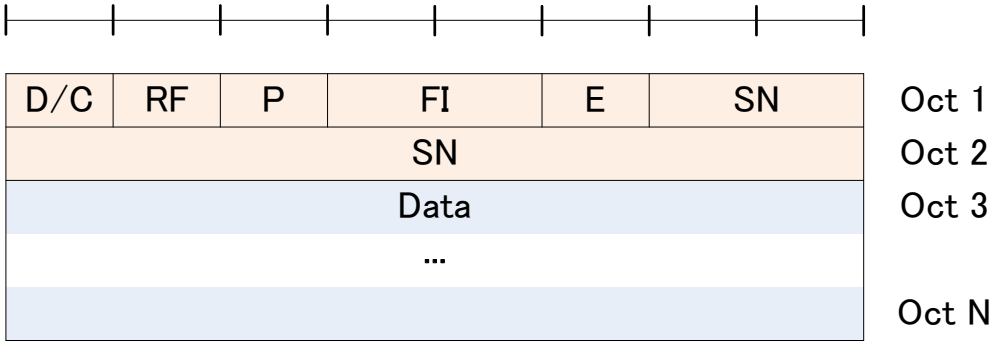
Reserved 1 (R1) – 1 bit

- The R1 field is a reserved field for this release of the protocol.

Length Indicator (LI) - 11 bit

- The LI field indicates the length in bytes of the corresponding Data field element present in the RLC data PDU delivered/received by an UM or an AM RLC entity.
- The first LI present in the RLC data PDU header corresponds to the first Data field element present in the Data field of the RLC data PDU,
- The second LI present in the RLC data PDU header corresponds to the second Data field element present in the Data field of the RLC data PDU, and so on.
- The value 0 is reserved.

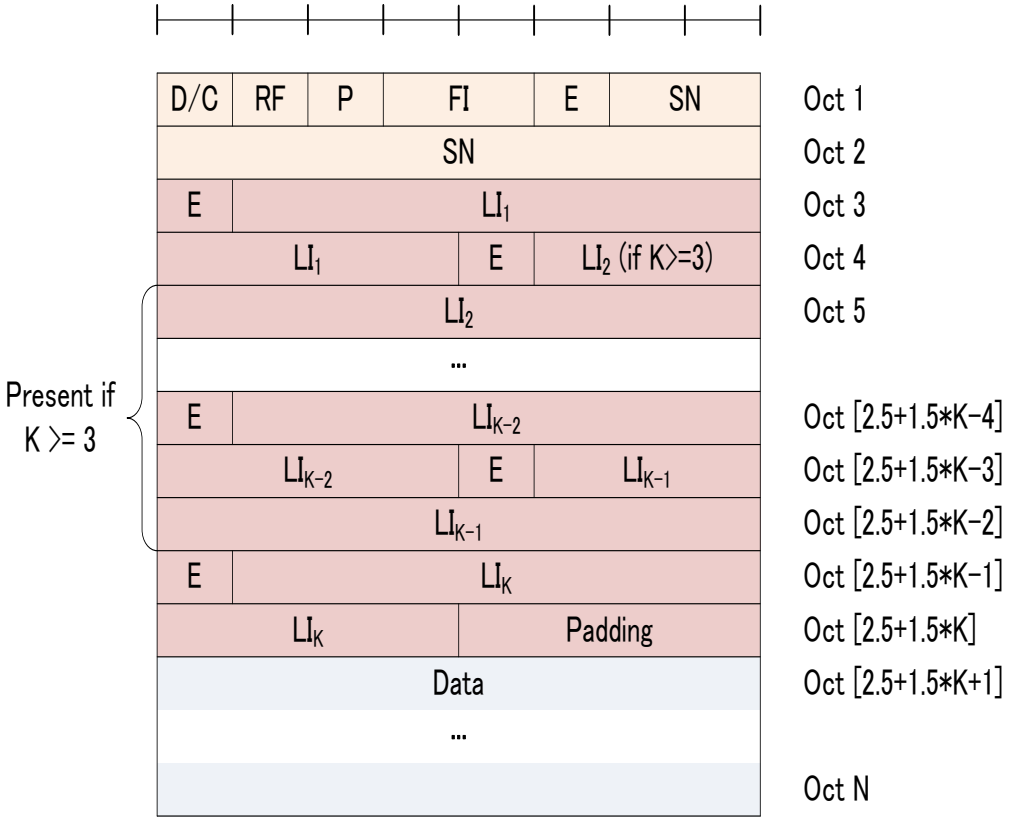
AMD PDU - 1



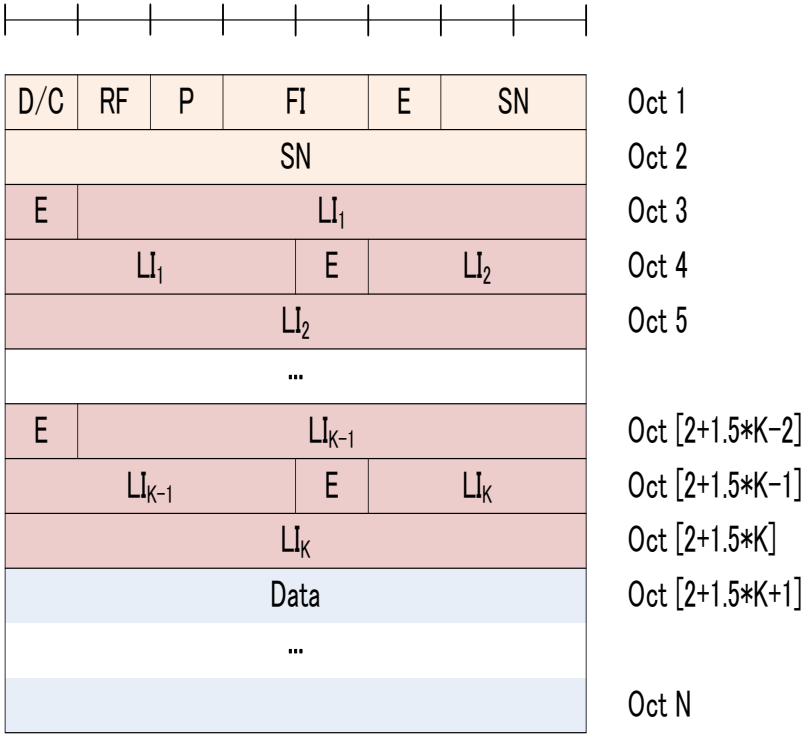
- AMD PDU consists of a Data field and an AMD PDU header.
- AMD PDU header consists of a fixed part (fields that are present for every AMD PDU) and an extension part
- An AMD PDU header needs to be extended when more than multiple Data field elements need to be sent.
  - In that which case an E and a LI are present for every Data field element except the last.
  - When an UMD PDU header consists of an odd number of LI(s), four padding bits follow after the last LI.
  - See next slide

AMD PDU - 2

Odd Number of LIs



Event Number of LIs



AMD PDU Specific Fields

Data/Control (D/C) – 1 bit

- The D/C field indicates whether the RLC PDU is a RLC data PDU or RLC control PDU.

Re-segmentation Flag (RF) – 1 bit

- The RF field indicates whether the RLC PDU is an AMD PDU or AMD PDU segment.

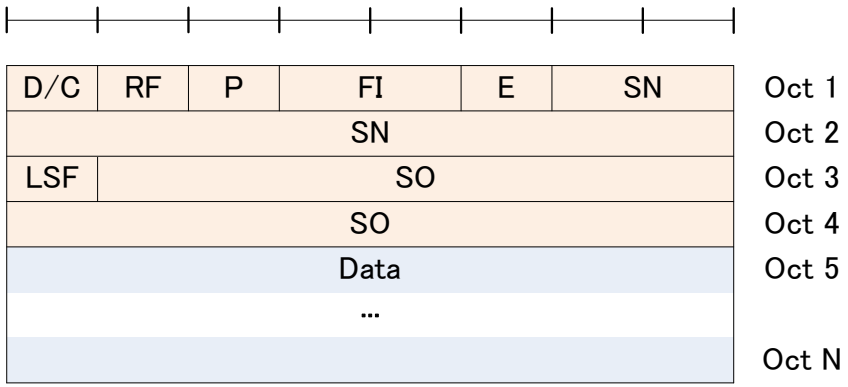
Polling bit (P) – 1 bit

- The P field indicates whether or not the transmitting side of an AM RLC entity requests a STATUS report from its peer AM RLC entity.

Sequence Number (SN) - 10 bit

- The SN field indicates the sequence number of the corresponding AMD PDU.
- For an AMD PDU segment, the SN field indicates the sequence number of the original AMD PDU from which the AMD PDU segment was constructed from.
- The sequence number is incremented by one for every AMD PDU.

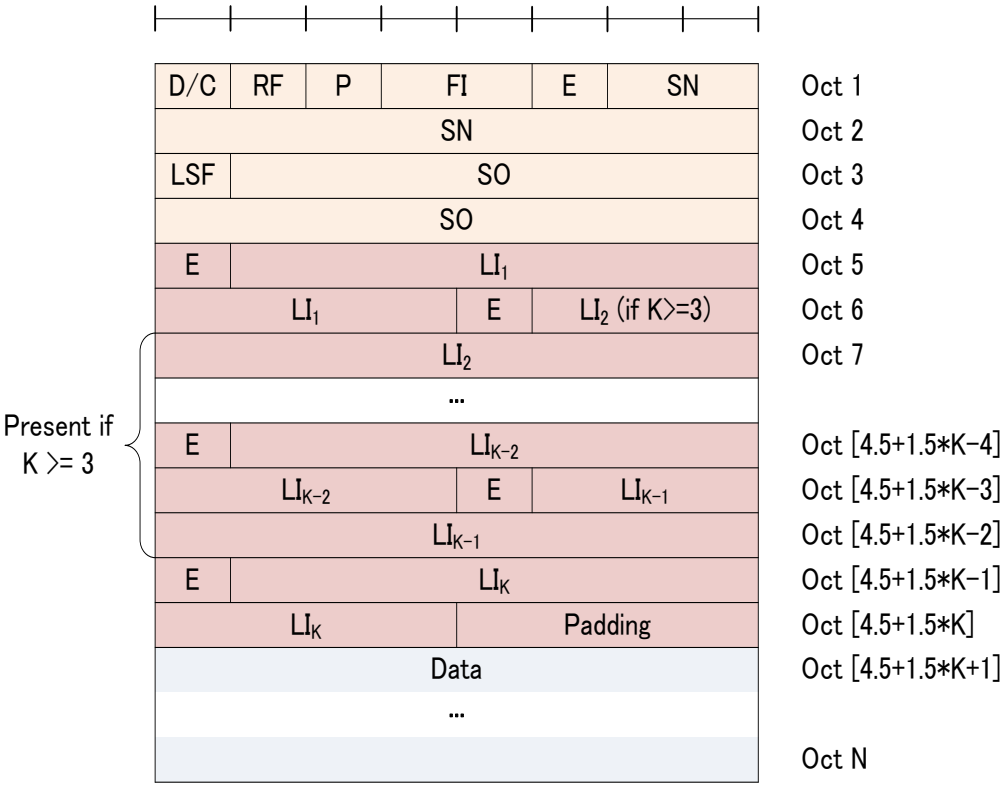
AMD PDU Segment - 1



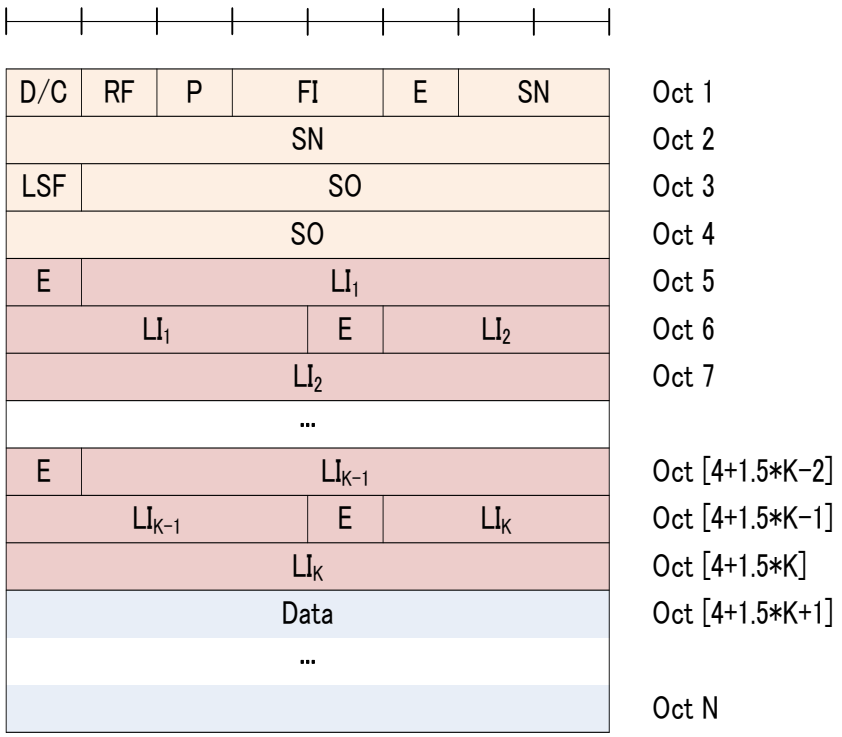
- AMD PDUs can be further segmented into AMD PDU Segments
- AMD PDU segment header consists of a fixed part and an extension part.
- An AMD PDU segment header needs to be extended when more than multiple Data field elements need to be sent.
  - In that which case an E and a LI are present for every Data field element except the last.
  - When an UMD PDU header consists of an odd number of LI(s), four padding bits follow after the last LI.
  - See next slide

AMD PDU Segment - 2

Odd Number of LIs



Even Number of LIs



AMD PDU Segment Specific Fields

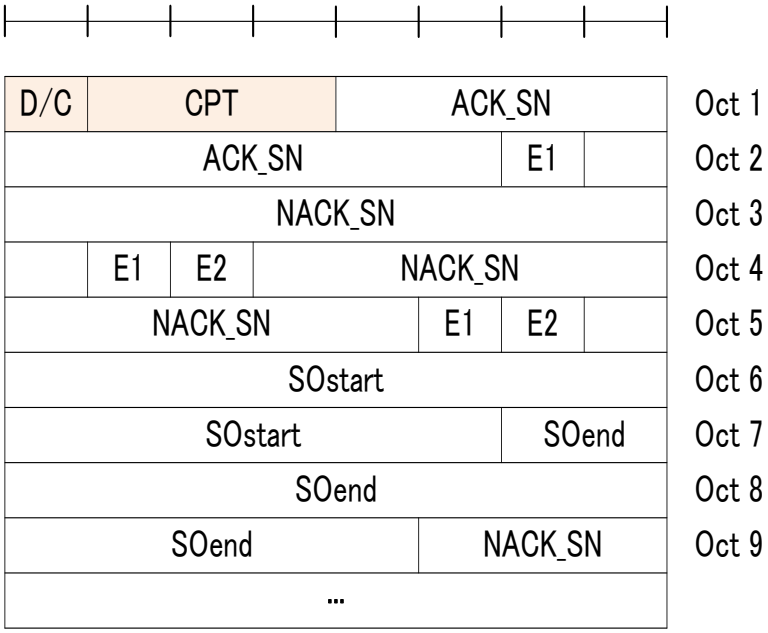
SO start (SOstart) - 15 bit

- The SOstart field indicates the portion of the AMD PDU with SN = NACK_SN that has been detected as lost at the receiving side of the AM RLC entity.
- The SOstart field indicates the position of the first byte of the portion of the AMD PDU in bytes within the Data field of the AMD PDU. The first byte in the Data field of the original AMD PDU is referred by the SOstart field value 0.

SO end (SOend) – 15 bit

- The SOend field indicates the portion of the AMD PDU with SN = NACK_SN that has been detected as lost at the receiving side of the AM RLC entity.
- The SOend field indicates the position of the last byte of the portion of the AMD PDU in bytes within the Data field of the AMD PDU.
- The special SOend value "111111111111111" is used to indicate that the missing portion of the AMD PDU includes all bytes to the last byte of the AMD PDU.

STATUS PDU



- STATUS PDU is used to send acknowledgements for received PDUs.
- Consists of payload and a RLC control PDU header.
- The payload starts from the first bit following the RLC control PDU header, and it consists of:
  - One ACK_SN and one E1
  - Zero or more sets of a NACK_SN, an E1 and an E2
  - Possibly a set of a SOstart and a SOend for each NACK_SN.
  - When necessary one to seven padding bits are included in the end of the STATUS PDU to achieve octet alignment.

Status PDU Fields

Acknowledgement SN (ACK_SN) – 10 bit

- The ACK_SN field indicates the SN of the next not received RLC Data PDU which is not reported as missing in the STATUS PDU.
- When the transmitting side of an AM RLC entity receives a STATUS PDU, it interprets that all AMD PDUs up to but not including the AMD PDU with SN = ACK_SN have been received by its peer AM RLC entity,
  - excluding those AMD PDUs indicated in the STATUS PDU with NACK_SN and portions of AMD PDUs indicated in the STATUS PDU with NACK_SN, SOstart and SOend.

Negative Acknowledgement SN (NACK_SN) – 10 bit

- The NACK_SN field indicates the SN of the AMD PDU (or portions of it) that has been detected as lost at the receiving side of the AM RLC entity.

Control PDU Type (CPT) – 3 bit

- The CPT field indicates the type of the RLC control PDU.
- A value of 0 represents the STATUS PDU. All other values are reserved.

Explore More

Specification	Title
3GPP TS 36.322	Evolved Universal Terrestrial Radio Access (E-UTRA) **Radio Link Control (RLC) protocol specification**
3GPP TS 36.300	Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); **Overall description**; Stage 2
3GPP TS 36.321	Evolved Universal Terrestrial Radio Access (E-UTRA); **Medium Access Control (MAC) protocol specification**
3GPP TS 36.211	Evolved Universal Terrestrial Radio Access (E-UTRA); **Physical channels and modulation**

Thank You

EventHelix.com
- telecommunication design
- systems engineering
- real-time and embedded systems

Thank you for visiting EventHelix.com. The following links provide more information about telecom design tools and techniques:

Links	Description
EventStudio System Designer 4.0	Sequence diagram based systems engineering tool.
VisualEther Protocol Analyzer 1.0	Wireshark based visual protocol analysis and system design reverse engineering tool.
Telecom Call Flows	GSM, SIP, H.323, ISUP, LTE and IMS call flows.
TCP/IP Sequence Diagrams	TCP/IP explained with sequence diagrams.
Real-time and Embedded System Articles	Real-time and embedded systems, call flows and object oriented design articles.