ORAN-WG4.CUS.0-v01.00

Technical Specification

O-RAN Fronthaul Working Group

Control, User and Synchronization Plane Specification

Revision History

Date	Revision	Author	Description
2019.03.11	01.00	M. Garyantes	Final version 01.00

Contents

Chapter 1 Introductory Material

1.1 Scope

This Technical Specification has been produced by the O-RAN Alliance.

The contents of the present document are subject to continuing work within O-RAN and may change following formal O-RAN approval. Should the O-RAN Alliance modify the contents of the present document, it will be re-released by O-RAN with an identifying change of release date and an increase in version number as follows:

Release x.y.z

where:

- x the first digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc. (the initial approved document will have x=01).
- y the second digit is incremented when editorial only changes have been incorporated in the document.
- z the third digit included only in working versions of the document indicating incremental changes during the editing process.

The present document specifies the control plane, user plane and synchronization plane protocols used over the fronthaul interface linking the O-DU (O-RAN Distributed Unit) with the O-RU (O-RAN Radio Unit) with a Lower Layer Functional Split-7-2x based architecture (explained below). The scope of this document includes both LTE and NR (5G). A separate document contains the O-RAN M-Plane (management plane) specification.

In the following, “Layer 1” and “Physical Layer” are assumed to be synonymous.

In the main body of this specification (in any “chapter”) the information contained therein is normative meaning binding on any compliant system, unless explicitly described as informative (a capability described as “optional” may or may not be included in a compliant system but if it is included it must comply with the optional capability description). Information contained in an “Annex” to this specification is always informative.

1.2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document in Release 15.

[1] 3GPP TR 21.905: “Vocabulary for 3GPP Specifications”.

[2] 3GPP TR 38.801: “Study on New Radio Access Technology; Radio Access Architecture and Interfaces”.

[3] eCPRI Specification V1.0 “Common Public Radio Interface: eCPRI Interface Specification”.

[4] IEEE Std 1588-2008 “Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems”.

[5] 3GPP TS 36.211: "Evolved Universal Terrestrial Radio Access (E-UTRA); LTE physical layer; General description".

[6] 3GPP TS 38.211 V15.1.0

[7] R1-1800296, "NR OFDM Symbol Generation Option Analysis", Intel, 3GPP TSG RAN WG1 AH#18-01, Vancouver, Canada, Jan. 22-26, 2018

[8] R1-1800802, "OFDM signal generation", Nokia, 3GPP TSG RAN WG1 AH#18-01, Vancouver, Canada, Jan. 22-26, 2018

[9] 3GPP TS 36.213: "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures".

[10] Small cell forum, "Document 082.09.05: FAPI and nFAPI specifications", Release 9, May 2017.

[11] xRAN-FH-MP.0-v01.00: xRAN Fronthaul Working Group Management Plane Specification, Release 01.00, July 2018.

[12] 3GPP TS 38.104 "Base Station (BS) radio transmission and reception", Release 15, v15.2.0 (2018-06).

[13] xRAN-FH-TUT.0-v01.00 xRAN Fronthaul Specification Tutorial, published December 2018.

1.2.1 Synchronization-specific Reference Documents

- ITU-T G.781 (08/2017) Synchronization layer functions
- ITU-T G.810 (08/1996) Definitions and terminology for synchronization networks
- ITU-T G.810 (08/1996) Cor. 1 (11/2011) Definitions and terminology for synchronization networks
- ITU-T G.8260 (08/2015) Definitions and terminology for synchronization in packet networks
- ITU-T G.8260 (08/2015) Amd. 1 (04/2016) Definitions and terminology for synchronization in packet networks
- ITU-T G.8261/Y.1361 (08/2013) Timing and synchronization aspects in packet networks
- ITU-T G.8261/Y.1361 (08/2013) Amd. 1 (01/2015) Timing and synchronization aspects in packet networks
- ITU-T G.8261/Y.1361 (08/2013) Cor. 1 (04/2016) Timing and synchronization aspects in packet networks
- ITU-T G.8262/Y.1362 (01/2015) Timing characteristics of a synchronous Ethernet equipment slave clock
- ITU-T G.8262/Y.1362 (01/2015) Cor. 1 (11/2016) Timing characteristics of a synchronous Ethernet equipment slave clock
- ITU-T G.8264/Y.1364 (08/2017) Distribution of timing information through packet networks
- ITU-T G.8271/Y.1366 (08/2017) Time and phase synchronization aspects of telecommunication networks
- ITU-T G.8271.1/Y.1366.1 (10/2017) Network limits for time synchronization in packet networks
- ITU-T G.8272/Y.1367 (01/2015) Timing characteristics of primary reference time clocks
- ITU-T G.8272/Y.1367 (01/2015) Erratum 1 (08/2015) Timing characteristics of primary reference time clocks
- ITU-T G.8272/Y.1367 (01/2015) Amd 1 (04/2016) Timing characteristics of primary reference time clocks
- ITU-T G.8272.1/Y.1367. (11/2016) Timing characteristics of enhanced primary reference time clocks
- ITU-T G.8272.1/Y.1367. (11/2016) Amd 1 (04/2016) Timing characteristics of enhanced primary reference time clocks
- ITU-T G.8273.2/Y.1368 (08/2013) Framework of phase and time clocks
- ITU-T G.8273/Y.1368 (08/2013) Corr 1(05/2014) Framework of phase and time clocks
- ITU-T G.8273/Y.1368 (01/2015) Amd. 1 (01/2015) Framework of phase and time clocks
- ITU-T G.8273/Y.1368 (08/2013) Amd 2 (08/2015) Framework of phase and time clocks
- ITU-T G.8273.2/Y.1368.2 (01/2017) Timing characteristics of telecom boundary clocks and telecom time slave clocks
- ITU-T G.8273.2/Y.1368.2 (01/2017) Amd. 1 (08/2017) Timing characteristics of telecom boundary clocks and telecom time slave clocks
- ITU-T G.8273.3/Y.1368.3 (10/2017) Timing characteristics of telecom transparent clocks
- ITU-T G8275.1/Y.1369.1 (06/2016) Precision time protocol telecom profile for phase/time
- ITU-T G8275.1/Y.1369.1 (06/2016) Amd. 1 (08/2017) Precision time protocol telecom profile for phase/time
- eCPRI Transport Network V1.1 (01/2018) Requirements for the eCPRI transport network
- IEEE 802.1CM-2018 (06/2018) Time-Sensitive Networking for Fronthaul

1.3 Definitions and Abbreviations

1.3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

BMCA: Best Master Clock Algorithm

C-Plane: Control Plane: refers specifically to real-time control between O-DU and O-RU, and should not be confused with the UE's control plane

c_eAxC: component eAxC: a portion of an eAxC flow assigned to a specific O-DU processing element. Includes the fields BandSector_ID, CC_ID, RU_Port_ID and DU_Port_ID (see sections 3.1.3.1.6 and 3.1.3.2.4)

CA: Carrier Aggregation

DL: DownLink: data flow towards the radiating antenna (generally on the LLS interface)

eAxC: extended Antenna-Carrier: a data flow for a single antenna (or spatial stream) for a single carrier in a single sector. Includes the fields BandSector_ID, CC_ID and RU_Port_ID but not DU_Port_ID (see sections 3.1.3.1.6 and 3.1.3.2.4)

eEEC: enhanced Ethernet Equipment Clock

ePRTC: enhanced Primary Reference Time Clock

FFO: Fractional Frequency Offset. This is defined as Δf/f_norm which is used to describe frequency error, typically on the output of the T-TSC filter in the O-RU. It is the same as the Fractional Frequency Deviation defined in ITU-T G.810.

Hop: physical link between 2 s-plane nodes (where node can be O-DU, switch or O-RU) as defined in IEEE 802.1CM

LAA: Licensed-assisted access: Carrier aggregation with at least one secondary cell operating in the unlicensed spectrum.

LLS: Lower Layer Split: logical interface between O-DU and O-RU when using a lower layer (intra-PHY based) functional split.

LLS-U: Lower Layer Split User-plane: logical interface between O-DU and O-RU when using a lower layer functional split.

LLS-C: Lower Layer Split Control-plane: logical interface between O-DU and O-RU when using a lower layer functional split.

High-PHY: those portions of the PHY processing on the O-DU side of the fronthaul interface, including FEC encode/decode, scrambling, and modulation/demodulation.

Low-PHY: those portions of the PHY processing on the O-RU side of the fronthaul interface, including FFT/iFFT, digital beamforming, and PRACH extraction and filtering.

M-Plane: Management Plane: refers to non-real-time management operations between the O-DU and the O-RU

O-CU: O-RAN Control Unit – a logical node hosting PDCP, RRC, SDAP and other control functions

O-DU: O-RAN Distributed Unit: a logical node hosting RLC/MAC/High-PHY layers based on a lower layer functional split.

O-RU: O-RAN Radio Unit: a logical node hosting Low-PHY layer and RF processing based on a lower layer functional split. This is similar to 3GPP's "TRP" or "RRH" but more specific in including the Low-PHY layer (FFT/iFFT, PRACH extraction).

OTA: Over the Air

OTDOA: Observed Time Delay Of Arrival

PRTC: Primary Reference Time Clock

PTP: Precision Time Protocol

Q<I.F>: denotes a signed two's-complement I+F bit fixed point number with I signed integer bits, and F fractional bits.

S-Plane: Synchronization Plane: refers to traffic between the O-RU or O-DU to a synchronization controller which is generally an IEEE 1588 Grand Master (however, Grand Master functionality may be embedded in the O-DU).

Slot: this is a group of 14 symbols, for LTE and NR. LTE has a separate slot definition within 3GPP which is 7 symbols but that definition is not used in this specification. So for NR, “slot” in this document means slot as 3GPP defines it, but for LTE “slot” in this document correlates to the LTE “TTI” as defined by 3GPP.

Spatial stream: the data flow on the DL associated with precoded data (may be same as layers or different if there is expansion in the precoding), and on UL associated with the number of outputs from the digital beamforming (sometimes called “beams”).

SSM: Synchronization Status Message: part of ITU G.781 and G.8264 standards.

SYNCE: Synchronous Ethernet

T-BC: Telecom Boundary Clock

T-GM: Telecom Grand Master

T-TSC: Telecom Slave Clock

TAE: Time Alignment Error as defined by 3GPP TS36.104 and TS38.104

TE: Time Error as defined by eCPRI network requirement specification

TRX: Refers to the specific processing chain in an O-RU associated with D/A or A/D converters. Due to digital beamforming the number of TRXs may exceed the number of spatial streams, and due to analog beamforming the number of TRXs may be lower than the number of antenna elements.

U-Plane: User Plane: refers to IQ sample data transferred between O-DU and O-RU

UL: UpLink : data flow away from the radiating antenna (generally on the LLS interface)

UNI: User Network Interface as defined by eCPRI network requirement specification

UQ<I.F>: denotes an unsigned I+F bit fixed point number with I unsigned integer bits and F fractional bits

1.3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP 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 3GPP TR 21.905 [1].

eNB	e NodeB (applies to LTE)
gNB	g NodeB (applies to NR)
O-DU	O-RAN Distributed Unit (see definitions section)
O-RU	O-RAN Radio Unit

Chapter 2 Architecture & Requirements

Architectural aspects

The architecture of eNB or gNB with O-DU and O-RUs is shown in **Figure 2-1**. LLS-C and LLS-U provide C-plane and U-plane over LLS interface, respectively.

In this architecture, O-DU and O-RU can be defined as follows.

Lower Layer Split Central Unit (O-DU): a logical node that includes the eNB/gNB functions as listed in section 2.1 split option 7-2x, excepting those functions allocated exclusively to the O-RU. The O-DU controls the operation of O-RUs.

O-RAN Radio Unit (O-RU): a logical node that includes a subset of the eNB/gNB functions as listed in section 2.1 split option 7-2x. The real-time aspects of control & user plane communication with the O-RU are controlled by the O-DU.

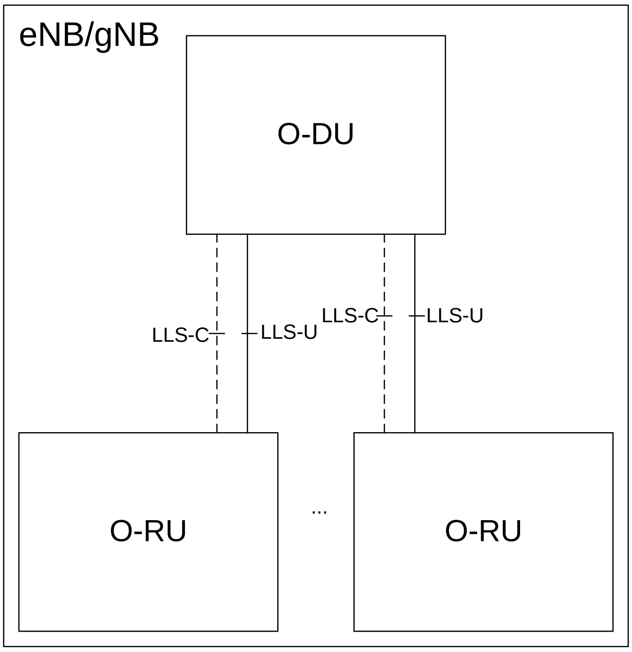


Figure 2-1 : eNB/gNB architecture with O-DU and O-RUs

2.1 Functional Split

When considering the functional split defining a fronthaul interface there are two competing interests:

a) There is a benefit in keeping an O-RU as simple as possible because size, weight, and power draw are primary deciding considerations and the more complex an O-RU, the larger, heavier and more power-hungry the O-RU tends to be;

b) There is a benefit in having the interface at a higher level which tends to reduce the interface throughput relative to a lower-level interface – but the higher-level the interface, the more complex the O-RU tends to be.

To resolve this conundrum, O-RAN has selected a single split point, known as “7-2x” but allows a variation, with the precoding function to be located either “above” the interface in the O-DU or “below” the interface in the O-RU. For the most part the interface is not affected by this decision, but there are some impacts namely to provide the necessary information to the O-RU to execute the precoding operation. O-RUs within which the precoding is not done (therefore of lower complexity) are called “Category A” O-RUs while O-RUs within which the precoding is done are called “Category B” O-RUs. See **Figure 2-2** for a depiction of this dual O-RU concept.

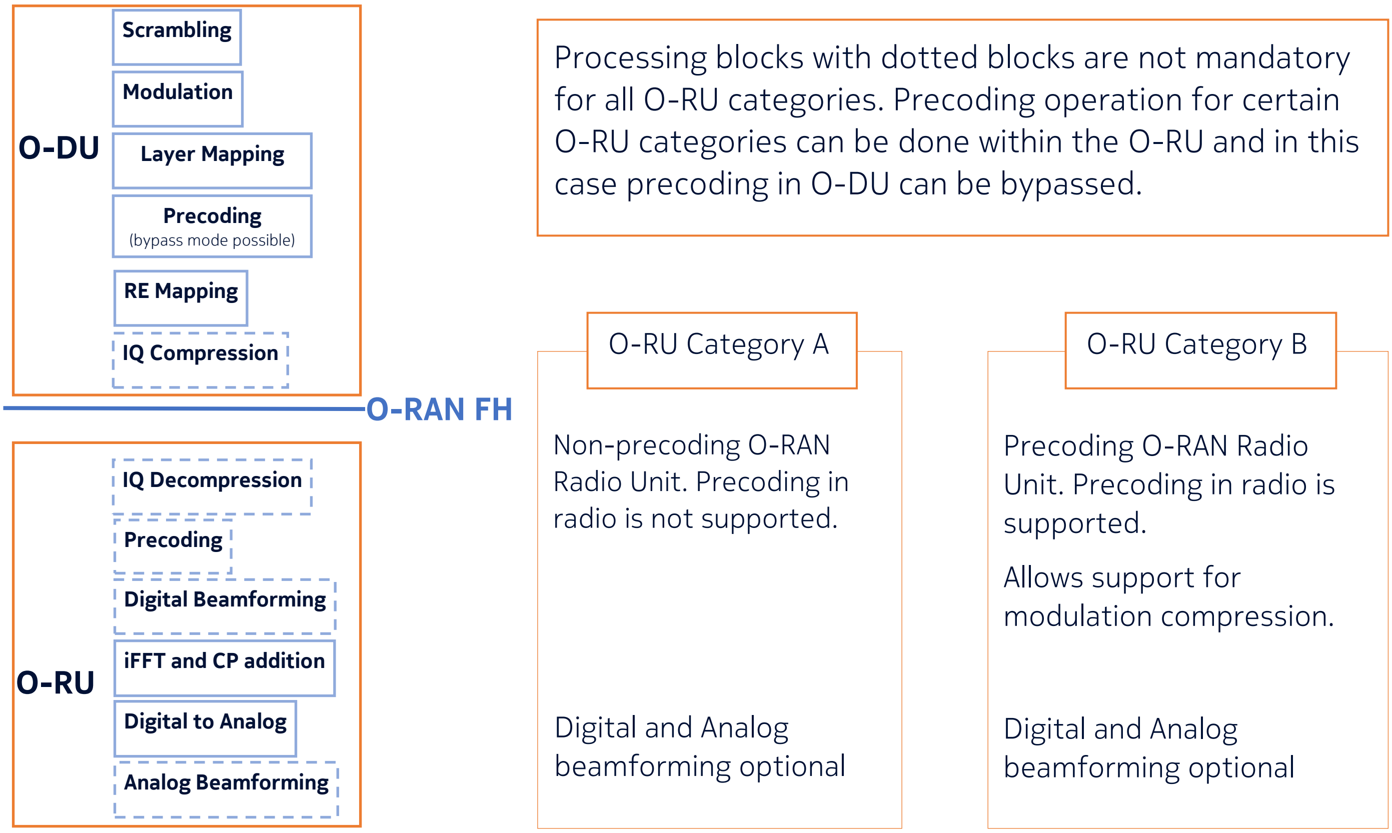


Figure 2-2 : Split Point and Category A and Category B O-RAN Radio Units

The inclusion of these two O-RU categories has certain implications for the LTE and NR functional splits in both DL and UL which are reflected in the following sections. In particular, for a Category B O-RU to implement precoding for LTE TM2-TM4 some special C-Plane instructions need to be provided to the O-RU from the O-DU; this is described in a later section of this document. For LTE TM5-10 and NR, no special instructions are needed because the precoding may be included in a digital beamforming processing block within the O-RU for a Category B O-RU (even for analog beamforming O-RUs), while for a Category A O-RU, the precoding would be executed in the O-DU and any beamforming in the O-RU, if present, would exclude the precoding calculation.

2.1.1 Selected Split 7-2x (DL)

DL functional split for various physical layer channels is illustrated in **Figure 2-3** (LTE Category A O-RUs), **Figure 2-4** (LTE Category B O-RUs), **Figure 2-5** (NR Category A O-RUs), and **Figure 2-6** (NR Category B O-RUs).

When O-RU Category A is supported by O-DU it is mandatory to support a total number of precoded streams of up to 8. Support for more than 8 precoded streams is optional.

For LTE (e.g. TM9) and NR PDSCH with UE specific reference signals, the DL processing chain specified by 3GPP does not include a precoding operation. The detailed precoding operation referred to in **Figure 2-3** and **Figure 2-4** for PDSCH with UE specific reference signals is not further described within this specification.

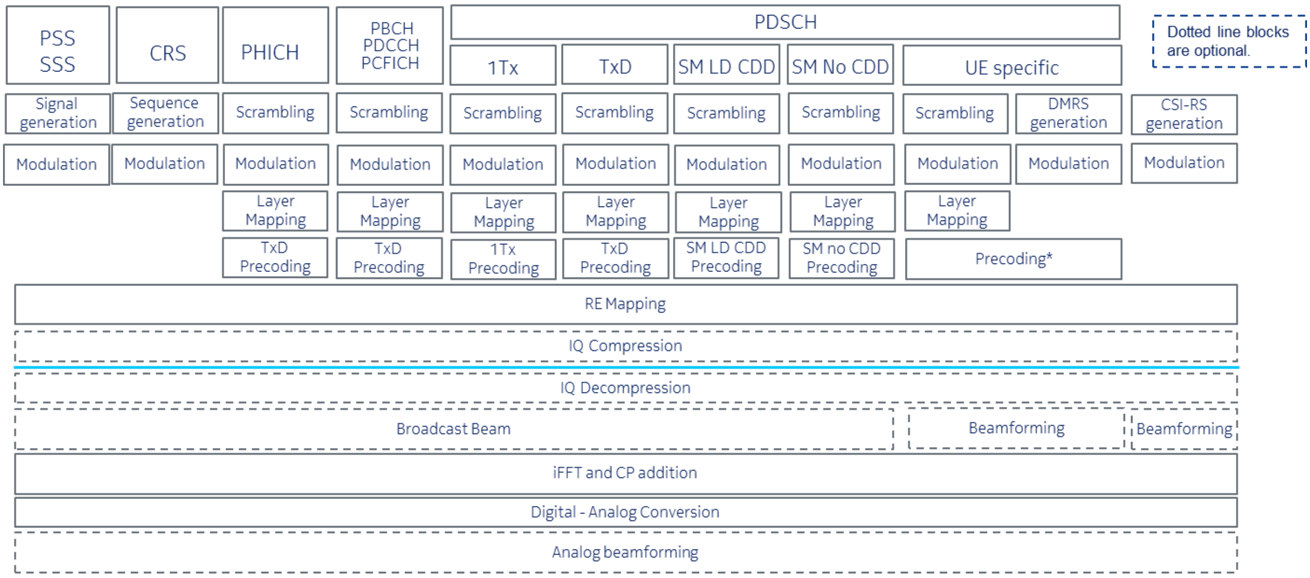


Figure 2-3 : Lower layer DL split description, LTE, Category A O-RUs

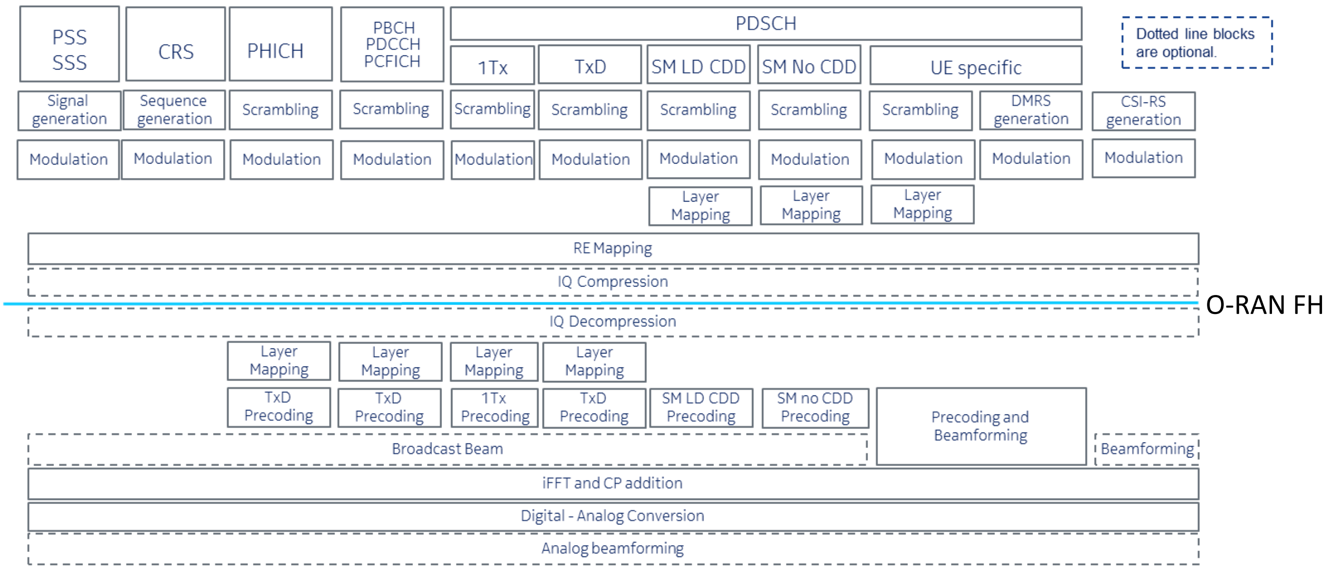


Figure 2-4 : Lower layer DL split description, LTE, Category B O-RUs

Although not explicitly indicated in Figure 2-4, RE mapping operation is separated into two parts for precoding support at O-RU for transmit diversity and spatial multiplexing modes in Category B radios:

1) RE mapping to frequency resources is performed at O-DU

2) RE mapping to antenna ports is performed at O-RU after precoding

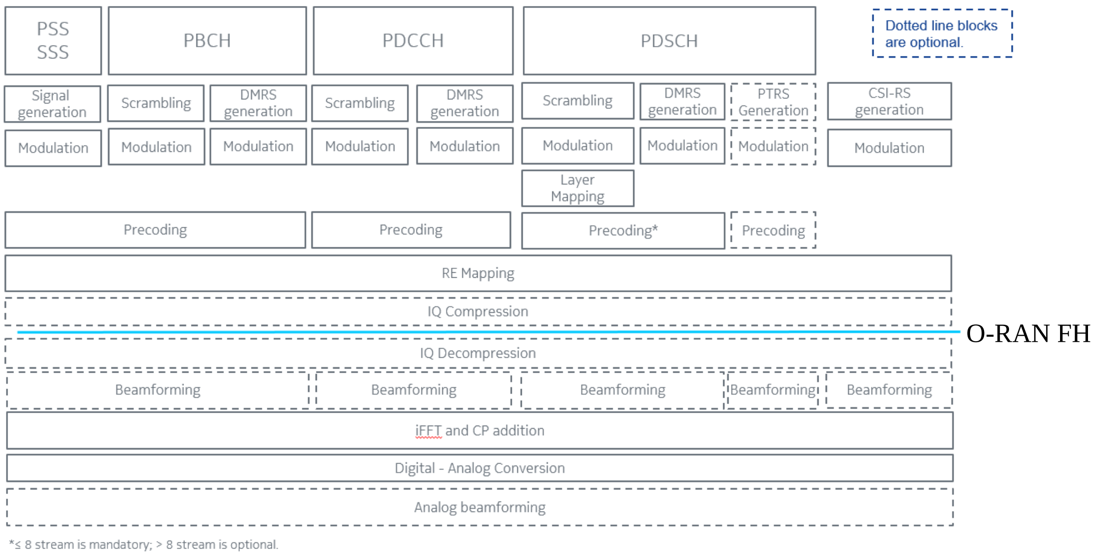


Figure 2-5 : Lower layer DL split description, NR, Category A Radio

Note: the above figure illustrates some of the high level functionalities inside the O-RU and O-DU, and do not bind the internal design of each of the O-RU and O-DU

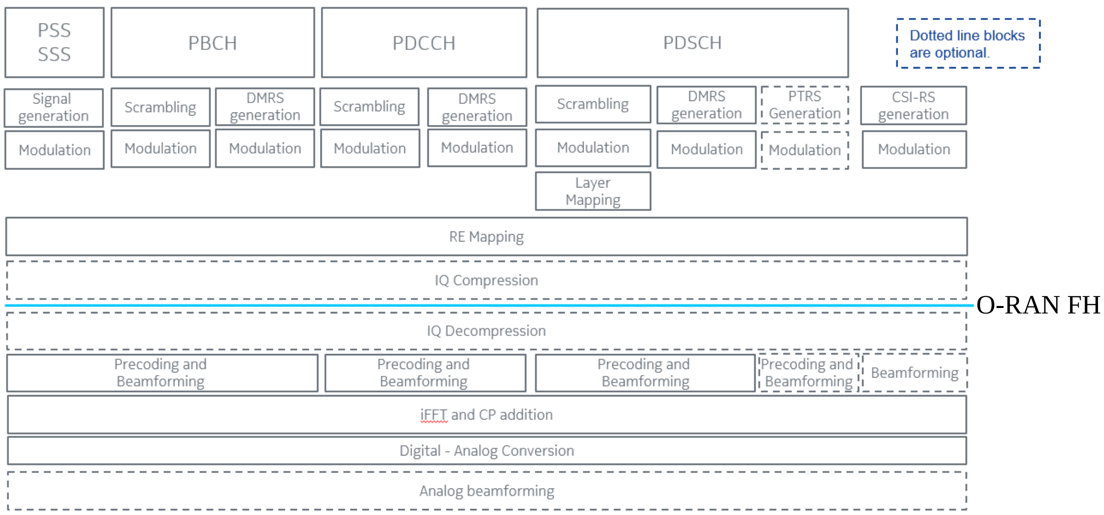


Figure 2-6 : Lower layer DL split description, NR, Category B Radio

Note: the above figure illustrates some of the high level functionalities inside the O-RU and O-DU, and do not bind the internal design of each of the O-RU and O-DU

Split 7-2x

<u>Description:</u>

In the DL, OFDM phase compensation [3GPP TS38.211 clause 5.4], iFFT, CP addition, and digital beamforming functions reside in the O-RU as well as precoding for Category B O-RUs. The rest of the PHY functions including resource element mapping, precoding, layer mapping, modulation, scrambling, rate matching and coding reside in the O-DU. Precoding must be in the O-DU for Category A O-RU support but this only applies if the number of precoder output spatial streams is 8 or less, otherwise precoding must be in the (Category B) O-RU.

The option of including some PHY functionality in a radio unit was not included in 3GPP TR 38.801 Rel 14 study.

<u>Benefits and Justification:</u>

- Interface simplicity: Transfer of user plane data is based on Resource Elements / Physical Resource Blocks, which simplifies the data mapping and limits the required associated control messages
- Transport Bandwidth Scalability: Lower split options (e.g., splits 7-1 and 8) scale based on number of antennas. In contrast, 7-2x interface scales based on "streams", which allows using high number of antennas without higher transport bandwidth. Further, user data transfer can be optimized to send only PRBs that contain user data for purpose of reducing transport bandwidth
- Beamforming Support: The same interface design can support different beamforming techniques (digital, analog, hybrid) as well as different beamforming algorithms. Likewise, deployments using only analog beamforming are also possible with the same interface design.
- Interoperability: Less user specific parameters are used at split 7-2x (when compared to higher split options), which can simplify specification.
- Advanced receivers and inter-cell coordination: this option allows implementation of advanced receivers and coordination features, which are also easier to implement and less restricted when most functions are placed at the O-DU. For example, UL CoMP is not possible when the UL upper-PHY processing is in the O-RU.
- Lower O-RU complexity: Less functions at O-RU (when compared to higher split options) allow limiting the number of required real time calculations as well as required memory requirement, especially for Category A O-RUs.
- Future proof-ness: Placing most functions at O-DU will allow introduction of new features via software upgrades without inflicting HW changes at O-RU (e.g., specification changes due to URLLC or new modulation schemes).
- Interface and functions symmetry: If the same interface and split point is used for DL and UL, specification effort can be reduced.

2.1.2 Selected Split 7-2x (UL)

UL functional split for various physical layer channels and transmission modes are illustrated in **Figure 2-7**. Likewise, digital beamforming in this context, is the function of antenna port selection or antenna port combining.

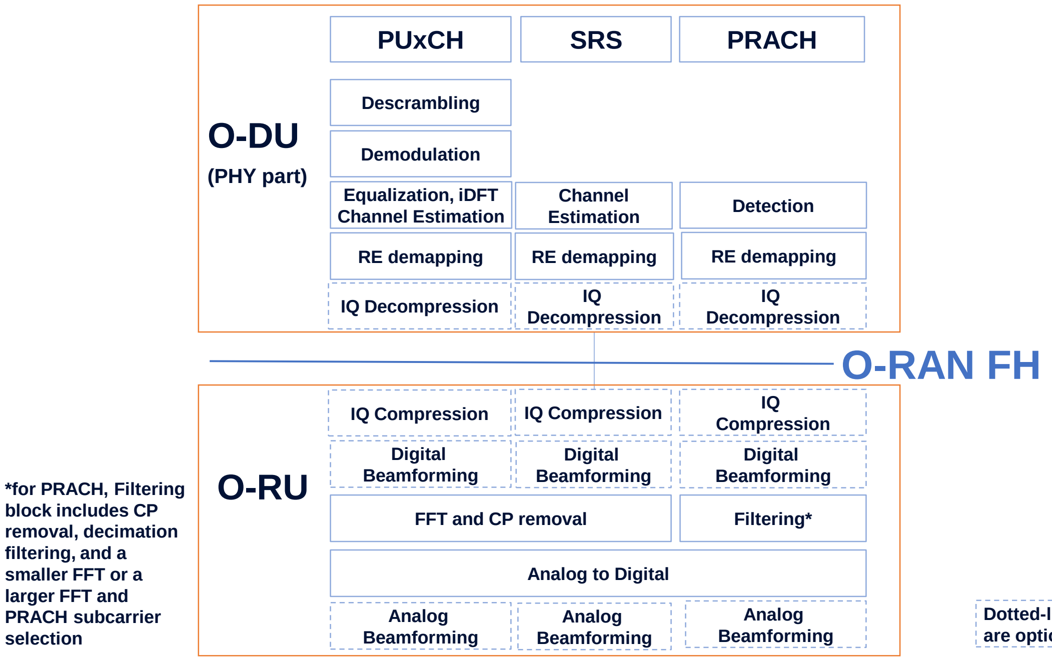


Figure 2-7 : Lower layer UL split description for LTE and NR

Note: the above figure illustrates some of the highlevel functionalities inside the O-RU and O-DU, and do not bind the internal design of each of the O-RU and O-DU.

Split 7-2x

Description:

In the UL, OFDM phase compensation (for all channels except PRACH) [3GPP TS38.211 clause 5.4], FFT, CP removal and digital beamforming functions reside in the O-RU. The rest of the PHY functions including resource element de-mapping, equalization, de-modulation, de-scrambling, rate de-matching and de-coding reside in the O-DU.

The option of including some PHY functionality in a radio unit was not included in 3GPP TR 38.801 Rel 14 study.

Benefits and Justification:

- The benefits defined for Option 7-2x for DL are also applicable for Uplink (See Section 2.1.1).

2.2 Data Flows

An overall reference of required inputs for some major functional blocks, their granularity and input originating source is compiled in **Table 2-1** and **Table 2-2** for DL and UL respectively.

Table 2-1 : Required information for each functional block (DL)

Function	Required Information	Signaling Granularity	Information source
Digital beamforming	• Digital beamforming information • Number of TRXs or Beams	Per RE per beam per slot	Scheduler (O-DU)
iFFT and CP addition	• FFT size • SC spacing • CP length	Per TRX per slot	Scheduler (O-DU)
Analog beamforming	• Analog beamforming info	Per frequency band per slot	Scheduler (O-DU)
Listen Before Talk (LBT)	• LBT parameters	Per MCOT	Scheduler (O-DU)

Table 2-2 : Required Information for each functional block (UL)

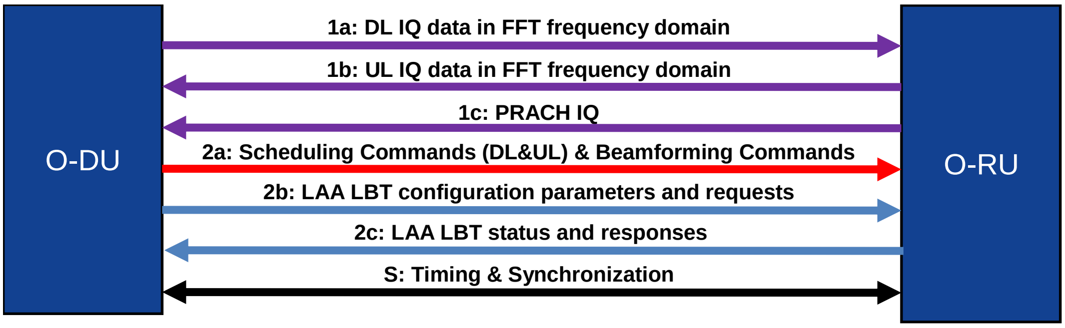
Function	Required Information	Signaling Granularity	Information source
Digital beamforming	• Digital beamforming information • Number of TRXs or Beams	Per RE per TRX per slot	Scheduler (O-DU)
FFT and CP removal	• FFT size • SC spacing • CP length	Per TRX per slot	Scheduler (O-DU)
Analog beamforming	• Analog beamforming info	Per frequency band per slot	Scheduler (O-DU)

In case of lower layer fronthaul based on split option 7-2x for DL and UL, the required external data flows (excluding M-plane) to exchange information between O-DU and O-RU can be categorized as follows. Further, the data flow contents mapping is described in **Table 2-3.**

- User-Plane
  - Data Flow 1a: Flows of IQ Data in FFT frequency domain on DL
  - Data Flow 1b: Flows of IQ Data in FFT frequency domain on UL
  - Data Flow 1c: Flow of PRACH IQ data in FFT frequency domain
- C-Plane

- Data Flow 2a: Scheduling commands (DL and UL) & Beamforming commands
- Data Flow 2b: LAA Listen-Before-Talk (LBT) configuration commands and requests
- Data Flow 2c: LAA LBT status and response messages

- Synchronization-Plane
  - Data Flow S: Timing and Synchronization data



Note: M-Plane flows not represented here

Figure 2-8 : Lower layer fronthaul data flows

Table 2-3 : Data Flow Information Mapping

Plane	ID	Name	Contents	Periodicity
U-Plane	1a	DL Frequency Domain IQ Data	DL user data (PDSCH), control channel data (PDCCH, etc.), …	< slot
	1b	UL Frequency Domain IQ Data	UL user data (PUSCH), control channel data (PUCCH, etc.), …	< slot
	1c	PRACH Frequency Domain IQ Data	UL PRACH data	< slot
C-Plane	2a	Scheduling Commands & Beamforming Commands	Scheduling information, FFT size, CP length, Subcarrier spacing, UL PRACH scheduling DL and UL Beamforming commands (e.g., beam index) and scheduling	~ slot
	2b	LAA LBT configuration parameters and requests	LBT Configuration parameters such as lbtHandle, lbtDeferFactor, lbtBackoffCounter, lbtOffset, MCOT, lbtMode, sfnSf, lbtCWconfig_H, lbtCWconfig_T, lbtTrafficClass.	per MCOT/DRS
	2c	LAA LBT status and responses	LBT DL indication parameters such as lbtHandle, lbtResult, initialPartialSFs, bufferError, lbtCWR_Result	
S-Plane	S	Timing and Synchronization	SyncE SSM & IEEE 1588 PTP packets	

2.3 Latency Requirements

Intra-PHY lower layer fronthaul split has characteristic of a stringent bandwidth and tight latency requirement. This implies use of a special “Fronthaul Service Profile” to be supported by the transport network, and which may differ depending on the operating environment, topology and target use cases. The general concept and latency model is based on eCPRI reference points for delay management definitions (See **Figure 2-9**). However, this specification provides additional details for both DL and UL (**Table 2-4**) latency. The delay parameters and how these are determined are explained below; note that in general it is expected a separate set of timing parameters applies to each O-RU attached to an O-DU. This chapter defines multiple approaches which can be used to provide an inter-operable timing solution.

The reference points defined for eCPRI are reflected below in **Figure 2-9.** The reference points are:

- O-DU: R1/ R4 – Transmit/ Receive interface at O-DU
- O-RU: R2/ R3 – Receive/ Transmit interface at O-RU
- Ra: Antenna interface at O-RU

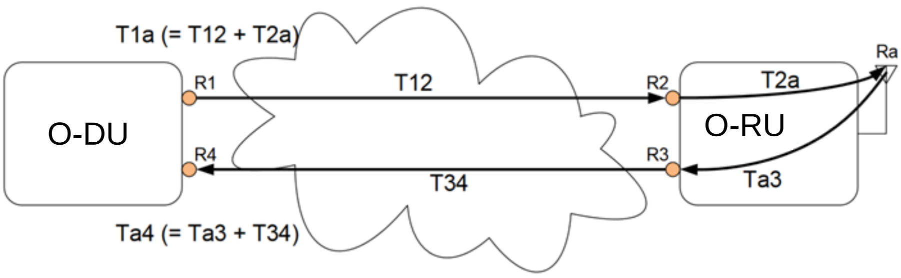


Figure 2-9 : Definition of reference points for delay management (adapted from [3])

Transmission delay between O-DU and O-RU are specified as T12 (downlink) and T34 (uplink). The transmission delay encompasses only the time from when a bit leaves the sender (R1/ R3) until it is received at the receiver (R2/ R4). In an ethernet transport network, these delays may not be constant due to switching delays (i.e. PDV). To account for this, transport delay must be considered as a range with upper and lower bounds:

- Downlink transport delay: T12min/ T12max
- Uplink transport delay: T34min/ T34max

However, fixed timing at Ra is still required. Therefore, Ra is used as a reference point for delay management in the eCPRI model. Therefore, transmission and reception at the reference points are measured relative to Ra, resulting in the following parameters:

Table 2-4 : eCPRI O-DU/ O-RU Delay Model Latency Parameters

		eCPRI	Latency	eCPRI	
				Minimum	Maximum
Downlink	O-DU	T1a	Measured from output at O-DU (R1) to transmission over the air.	T1amin	T1amax
	O-RU	T2a	Measured from reception at O-RU (R2) to transmission over the air	T2amin	T2amax
Uplink	O-DU	Ta4	Measured from reception at O-RU antenna to reception at O-DU port (R4).	Ta4min	Ta4max
	O-RU	Ta3	Measured from reception at O-RU antenna to output at O-RU port (R3)	Ta3min	Ta3max

2.3.1 Timing Parameter Relationships

To ensure proper reception of data at the receiver over the packet interface, there are several relationships between the parameters defined above which must be met. First it is important to understand the nature of the transmission itself. In either direction (downlink/ uplink) it takes some amount of time for the sender to actually transmit the packets onto the transmission media. However, the amount of data for any interval (e.g symbol) can vary thus resulting differing transmit times. This transmission time can be affected by several factors including (but not limited to) transport media rate, air interface bandwidth, and amount of data compression. The maximum amount of time allowed for the transmitter to send all data for an interval (Transmission Window) is defined by T1amax – T1amin. It is important to note that this is the allowed time, based on transport and O-RU characteristics. This is explained in greater detail, and its impacts on O-DU in section 2.3.1.1.

To account for transport variation and transmission time the receiver implements a reception window. This allows packets containing samples for a specific symbol to be received within the window and still be transmitted at Ra at the required time. The size of the Reception Window must account for both the maximum transmission time at the sender and the transport variation through the fronthaul network. The result is the first of the delay relationships which must be met to ensure a working delay solution.

Reception Window ≥ Transmission Window + Transport Variation

Table 2-5 : eCPRI Delay Windows

	Reception Window	Transmission Window	Transport Variation
Downlink	T2amax – T2amin	T1amax – T1amin	T12max – T12min
Uplink	Ta4max – Ta4min	Ta3max – Ta3min	T34max – T34min

The position (in time) of the reception/ transmission windows at the O-RU is fixed relative to the air interface. However, the position of the corresponding windows at the O-DU is a function of the O-RU and transport parameters. For guaranteed reception of packets sent from O-DU to O-RU within the O-RU reception window, the following relationships must also be met[1]:

Table 2-6 : O-DU transmission/ reception window position (in time)

O-DU Timing		Parameter	O-DU Transmit Boundary Relationships
Downlink (Transmit)	No earlier than	T1amax	T1amax ≤ T2amax + T12min
	No later than	T1amin	T1amin ≥ T2amin + T12max
Uplink (Receive)	No earlier than	Ta4min	Ta4min ≤ Ta3min + T34min
	No later than	Ta4max	Ta4max ≥ Ta3max + T34max

2.3.1.1 O-DU Transmission Window

The O-DU transmission window (T1amax – T1amin) is defined by the relationships above based on the O-RU reception window and max transport variation. It does not define the exact timing of transmission from the O-DU. Rather, it defines the boundaries that the O-DU transmission must operate within. The window merely represents the mathematical boundaries imposed on the O-DU as a result of the O-RU and Transport constraints. It is possible to define the constraints for any one of O-DU, transport and O-RU based on knowledge of the other two. However, typically the O-RU constraint is predefined based on the equipment and transport is part of the overall network goals.

The window resulting from the relationships must be greater than or equal to the actual maximum time required by the O-DU to transmit all data for a symbol ($TXmax_{O\text{-}DU}$). That is, the window must be at least large enough that the O-DU can transmit in the worst case within the window. Where, within the window, the O-DU transmits (e.g. beginning, middle, end) and how much of the window is consumed by the O-DU transmission is a matter of O-DU design.

The following downlink example illustrates the concept:

- O-RU parameters: T2amin = 100 usec, T2amax = 260 usec
- Transport Parameters (direct fiber of known length): T12min = 50 usec, T12max = 51 usec

The result indicates an O-DU transmission window as follows:

- T1amax ≤ 260 + 50
- T1amin ≥ 100 + 51

This provides a very large transmit window available to the O-DU. If, for example, the $TXmax_{O\text{-}DU}$ is only 30 usec, then the O-DU can determine where within the window to start its transmission, so long as the transmission completes prior to T1amin.

If, however, this same O-DU were paired with an O-RU with smaller reception window (e.g. T2amin = 100, T2amax = 150) using a transport network with the same T12min, but with 15 usec of PDV (T12max = 65), the result is:

- T1amax ≤ 150 + 50
- T1amin ≥ 100 + 65

The delay solution still works (200 – 165 ≥ 30), but with far less margin, and far less flexibility as to where within the window the O-DU may begin transmission.

[1] Refer to Annex B for explanation of these inequalities.

2.3.2 U-Plane/ C-Plane Timing

The basic delay parameters above describe the general delay model and characteristics of the O-RAN interface. However, the ORAN interface is divided into C-Plane and U-Plane parts. The C-Plane must be available in order to process the corresponding U-Plane packets. To support coordination of C-Plane and U-Plane timing, the O-RAN interface specifies that C-Plane messages must arrive at the O-RU some amount of time in advance (Tcp_adv_dl) of the corresponding U-Plane messages. (Note that it is possible for Tcp_adv_dl to be set to 0, but the O-RAN delay model does not assume this.) As a result, O-RAN has defined the downlink timing relationships and parameters shown in **Figure 2-10**.

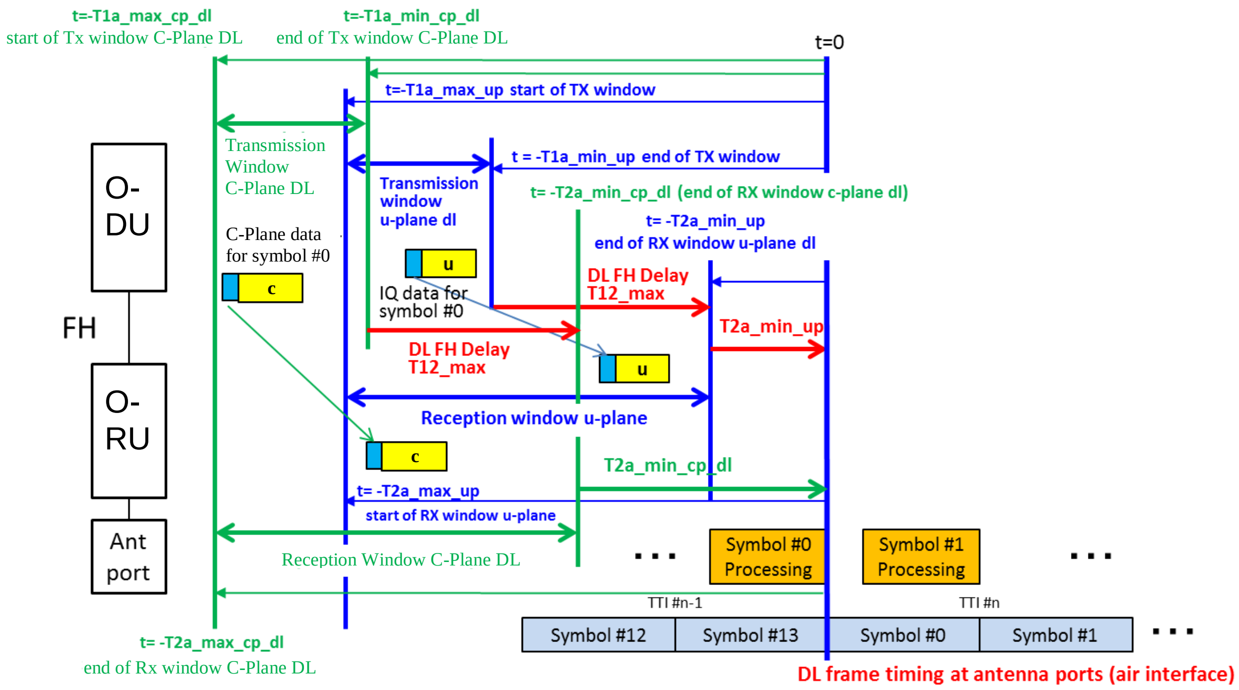


Figure 2-10 : Timing relations per symbol IQ in DL direction (U-Plane and C-Plane)

For simplicity O-RAN assumes that the transmission windows for C-Plane and U-Plane are the same size. Tcp_adv_dl allows for different alignment of the respective transmission/ reception windows as illustrated in the figure.

For uplink, the C-Plane is sent from O-DU to O-RU, while the U-Plane is from O-RU to O-DU. Since uplink C-Plane is from O-DU to O-RU, it uses the R1/ R2 reference points, and corresponding timing parameters. However, the uplink C-Plane timing is not characterized relative to the downlink C-Plane or U-Plane timing, but relative to Ra. The resulting O-RAN delay model is shown in **Figure 2-11**.

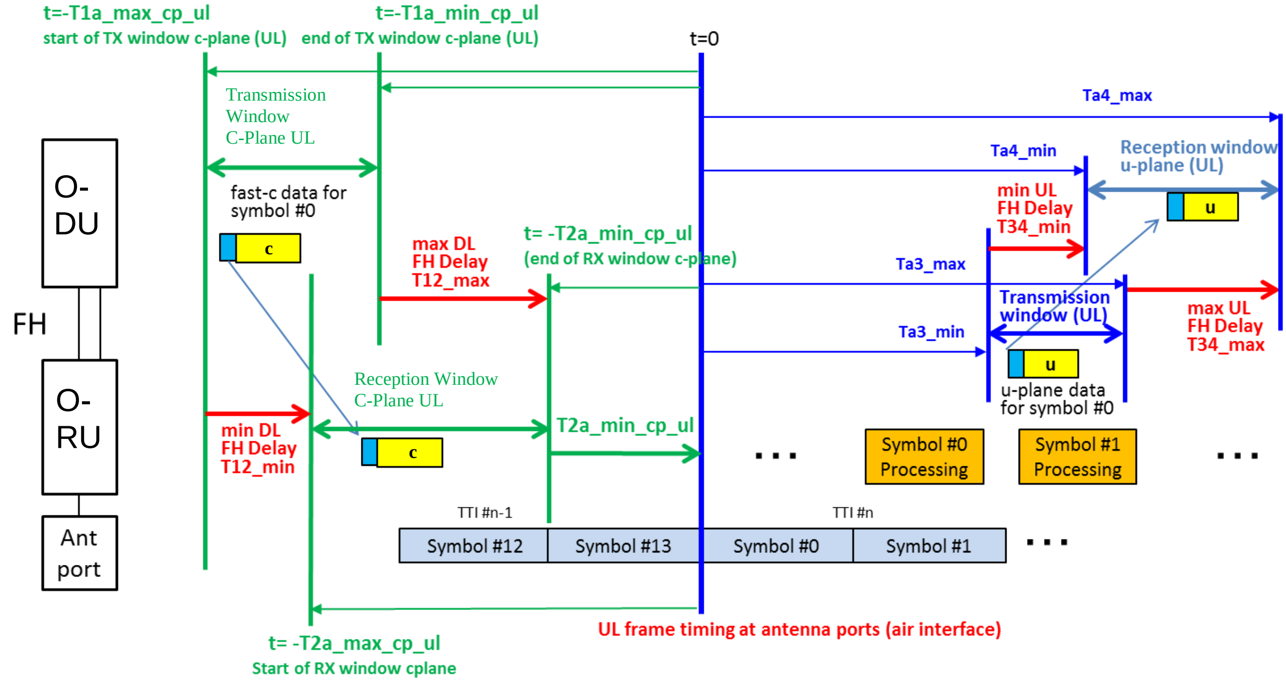


Figure 2-11 : Timing relations per symbol IQ in UL direction (U-Plane and C-Plane)

The resulting O-RAN delay model parameters are summarized on **Table 2-7**. The respective sections within the table for O-RU and O-DU define the delay profile parameters for the equipment.

Table 2-7 : O-RAN Delay Management Model Parameters

	Model Parameters	C-Plane		U-Plane	
		DL	UL	DL	UL
Network	T12min	T12_min	T12_min	T12_min	NA
	T12max	T12_max	T12_max	T12_max	NA
	T34min	NA	NA	NA	T34_min
	T34max	NA	NA	NA	T34_max
O-RU	T2amin	T2a_min_cp_dl	T2a_min_cp_ul	T2a_min_up	NA
	T2amax	T2a_max_cp_dl	T2a_max_cp_ul	T2a_max_up	NA
		Tadv_cp_dl	NA	NA	NA
	Ta3min	NA	NA	NA	Ta3_min
	Ta3max	NA	NA	NA	Ta3_max
O-DU	T1amin	T1a_min_cp_dl	T1a_min_cp_ul	T1a_min_up	NA
	T1amax	T1a_max_cp_dl	T1a_max_cp_ul	T1a_max_up	NA
	Ta4min	NA	NA	NA	Ta4_min
	Ta4max	NA	NA	NA	Ta4_max

The relationships between the various delay model parameters are defined below:

Table 2-8 : Downlink Delay Relationships

	Earliest transmission from O-DU	Latest Transmission from O-DU
U-Plane	T1a_max_up ≤ T2a_max_up + T12_min	T1a_min_up ≥ T2a_min_up + T12_max
C-Plane	T1a_max_cp_dl ≤ T2a_max_cp_dl + T12_min	T1a_min_cp_dl ≥ T2a_min_cp_dl + T12_max

Table 2-9 : Uplink Delay Relationships

	Earliest Reception at O-DU	Latest Reception at O-DU
U-Plane	Ta4min ≤ Ta3min + T34min	Ta4max ≥ Ta3max + T34max
C-Plane	T1a_min_cp_ul ≥ T2a_min_cp_ul + T12_max	T1a_max_cp_ul ≤ T2a_max_cp_ul + T12_min

For the O-RAN interface to operate properly, the transmit and receive windows at the O-DU must be properly aligned. (The O-RU window alignment is always based on Ra.) For the O-DU to align transmit and receive windows, it needs the following:

- **O-RU Delay Characteristics**
- **Transport Network Delay Characteristics**

The expected accuracy of the reported O-RU delay characteristics is 200ns. The 200ns value was decided as a compromise between a very high accuracy (tens of nanoseconds) which makes buffering in the O-RU easier and a more modest accuracy (~0.5 microseconds) which is relatively easy for a well-controlled Ethernet network to accomplish. This accuracy applies only to the reported start and end times of the reception and transmission windows relative to Ra. Ra is a fixed reference point as defined by the respective air interface standards and supported features. As an example, if the O-RU reports T2a_min_up of 20 µs, the O-RU may begin processing the data from 19.8 µs to 20.2 µs prior to the symbol over the air transmission time (Ra). However, the O-RU MUST still transmit over the air with the precision as defined by the air interface standards.

The remainder of this chapter addresses O-RAN supported methods for providing the O-DU with the required information to properly align transmit and receive windows. The following methods are supported:

Table 2-10 : O-DU transmit/ receive window alignment methods

Delay Characteristics	Computed Methods	
	Defined Transport	Measured Transport
O-RU Delay	O-RU specific values available to O-DU	O-RU specific values available to O-DU
Transport Delay	Pre-defined Min/ Max	Min – Measured; PDVmax - SLA

2.3.3 Computed Latency Methods

As previously noted, the goal of delay management for O-RAN interfaces is to ensure that the transmission/ reception windows as the O-DU are properly aligned to support the O-RU and transport network characteristics. The computed methods require the O-DU to compute the required transmit and receive windows based on O-RU delay and transport network characteristics. The O-DU then determines if the O-DU can accommodate the required transmit and receive windows. If so, it sets its transmit and receive windows (based on O-DU design) to the most appropriate settings. As an example, an O-DU may have a fixed set of transmit and receive windows. The O-DU may then select the fixed settings that best fit the transport latency (T12/ T34) and O-RU delay characteristics for the connected O-RUs.

The O-DU should raise some notification if it is not capable of accommodating the required transmit and receive windows.

Alignment of the transmit window at the O-DU requires that the following conditions be met based on the downlink delay relationships defined in **Table 2-8**:

- The O-DU MUST be able to transmit DL U-Plane **(T1a_min_up/ T1a_max_up)**
  - Early enough **(e.g. before T1a_min_up)** to ensure it is received at O-RU before T2a_min_up
  - Late enough **(e.g. after T1a_max_up)** to ensure it is NOT received before T2a_max_up
- Similarly, the O-DU must be able to transmit C-Plane messages within their respective windows

Alignment of the receive window at the O-DU requires that the following conditions be met based on the uplink delay relationships defined in **Table 2-9** :

- The O-DU MUST set its receive window for UL U-Plane **(Ta4_min, Ta4_max)**
  - Early enough to ensure UL U-Plane is not received before Ta4_min
  - Late enough to ensure UL U-Plane is not received after Ta4_max

The following O-DU constraints must be met for delay management to operate properly:

Table 2-11 : O-DU Constraints

O-DU	Constraint		
	Description	**Parameter**	**Relationship**
Transmit Window	maximum required transmit window	$TXmax_{O\text{-}DU}$	$T1a\_max\_up - T1a\_min\_up \geq TXmax_{O\text{-}DU}$
	Maximum supported T1a_max_up	**$T1a\_max\_up_{O\text{-}DU}$**	**$T1a\_max\_up_{O\text{-}DU}$** $\geq T1a\_max\_up$
Receive Window	Maximum supported receive window	**$RXmax_{O\text{-}DU}$**	**$RXmax_{O\text{-}DU}$** $\geq Ta4\_max - Ta4\_min$
	Maximum supported uplink latency relative to Ra	**$Ta4\_max_{O\text{-}DU}$**	**$Ta4\_max_{O\text{-}DU}$** $\geq Ta4\_max$

If any of the O-DU constraints is violated, then proper delay operation cannot be assured.

2.3.3.1 Fronthaul Timing Domain

When using a computed delay approach, the domain over which the O-DU delay parameters apply must be considered. A timing domain is defined as the set of CU and O-RU ports to which the computed O-DU delay parameters apply. The O-DU timing domain may encompass a single CU port, or may encompass multiple CU ports, where the O-DU requires the same O-DU timing parameters to apply to all CU ports in the domain.

Similarly, each CU port in the domain may support O-RAN C/U-Plane traffic to 1 or more O-RU ports. Therefore, a fronthaul timing domain encompasses one or more DU_Port_IDs and one or more RU_port_IDs.

When computing the O-DU transmit/ receive windows, the single set of delay parameters for the O-DU must encompass the worst case of Transport and O-RU delay characteristics within the domain. The following definitions are used:

- Transport Network Parameters for timing domain

  $T12\_min_{ij}$: T12_min between DU_Port_ID(i) and RU_Port_ID(j)

  $T34\_min_{ij}$: T34_min between DU_Port_ID(i) and RU_Port_ID(j)

Note: it is assumed that if a timing domain has multiple T12_min/ T34_min values, then the transport delay characteristics are measured. For measured transport delay it is not possible to measure the maximum delay. T12_max and T34_max in this case is computed by adding an a pre-defined (e.g. via SLA) worst case variation (PDVmax) to the corresponding transport minimum delay values.

Table 2-12: Radio Delay Parameters for O-RU Port "j" within timing domain

	Downlink U-Plane	Uplink U-Plane	Downlink C-Plane[2]	Uplink C-Plane
Minimum	$T2a\_min\_up_j$	$Ta3\_min_j$	$T2a\_min\_cp\_dl_j$	$T2a\_min\_cp\_ul_j$
Maximum	$T2a\_max\_up_j$	$Ta3\_max_j$	$T2a\_max\_cp\_dl_j$	$T2a\_max\_cp\_ul_j$

The resulting parameters to be used for determining the O-DU window for the timing domain are:

Table 2-13 : Transport Network Parameters for Timing Domain

	Downlink	Uplink
Minimum	$T12\_min = MIN(T12\_min_{ij})$	$T34\_min = MIN(T34\_min_{ij})$
Maximum	$T12\_max = MAX(T12\_min_{ij}) + PDVmax$	$T34\_max = MAX(T34\_min_{ij}) + PDVmax$

Table 2-14 : O-RU Delay Parameters for Timing Domain

		U-Plane	C-Plane
Downlink	Minimum	$T2a\_min\_up = MAX(T2a\_min\_up_j)$	$T2a\_min\_cp\_dl = MAX(T2a\_min\_cp\_dl_j)$
	Maximum	$T2a\_max\_up = MIN(T2a\_max\_up_j)$	$T2a\_max\_cp\_dl = MIN(T2a\_max\_cp\_dl_j)$
Uplink	Minimum	$Ta3\_min = MIN(Ta3\_min_j)$	$T2a\_min\_cp\_ul = MAX(T2a\_min\_cp\_ul_j)$
	Maximum	$Ta3\_max = MAX(Ta3\_max_j)$	$T2a\_max\_cp\_ul = MIN(T2a\_max\_cp\_ul_j)$

[2] Tcp_adv_dl may be different across RUs within a domain, therefore T2a_min_cp_dl and T2a_max_cp_dl must be used to determine downlink C-Plane window.

The tables defined above are used to derive the corresponding delay parameters to be used in the timing relationships defined in **Table 2-8** and **Table 2-9** to support the computed methods defined below.

When using a computed delay approach, there are two situations that are considered within O-RAN:

- **Defined Transport Method**: the network delay is pre-defined (usually by the network operator) and the delays are computed based on that definition. The definition generally considers the maximum network latency, with an assumption that a smaller delay can more easily be accommodated.
- **Measured Transport method**: the network delay is estimated based on actual measurements of packet latency on the downlink and uplink (sometimes a symmetric delay is assumed but this is not necessary).

Regardless of whether the Defined Transport Method or Measured Transport Method is used, the calculations are mainly the same. The next two sub-sections describe in more detail these two methods.

2.3.3.2 Defined Transport Method

With the Defined Transport method, the O-DU transmit and receive windows are determined based on pre-defined transport network characteristics, and the delay characteristics of the O-RUs within the timing domain. For this approach to work, the O-RU delay characteristics as defined in **Table 2-14** for each O-RU in the timing domain must be available to the O-DU. The O-DU can adapt its transmit and receive windows to accommodate the O-RU delay characteristics if CU has that capability. The adaptation of the O-DU transmission and reception windows may be limited based on the O-DU design.

Table 2-15 : Delay Profiles

	O-RU delay profile	O-DU delay profile
Downlink	T2a_min_up, T2a_max_up, T2a_min_cp_dl, T2a_max_cp_dl, Tcp_adv_dl	$T1a\_max\_up_{O\text{-}DU}$, $TXmax_{O\text{-}DU}$
Uplink	Ta3_min, Ta3_max, T2a_min_cp_ul, T2a_max_cp_ul	$Ta4\_max_{O\text{-}DU}$, $RXmax_{O\text{-}DU}$

Refer to M-Plane specification for details on how O-RU parameters are obtained. Optionally, an O-RU may adapt its delay profile information, especially for uplink, based on O-DU delay profile and transport delay (T12_min/ T34_min). If this option is supported, then the O-DU must provide its delay profile as well as the T12_min to the O-RU. The exchange of this information should use the same approach as defined for exchange of O-RU delay profile. With this option, the O-RU receives the O-DU delay profile before sending its delay profile to the O-DU. The O-DU is unaware that the O-RU delay profile information has been modified. It simply uses the delay profile provided by the O-RU as it would without this option.

As previously noted, delay characteristics for an O-RU may vary based on air interface properties. To ensure interoperability, O-RAN supported air interface properties which may be used as the basis for supporting different delay characteristics are limited to:

- Channel Bandwidth
- SCS

A set of delay characteristics which applies to a combination of the above properties is referred to as a delay profile. For each supported combination of the above properties that an O-RU supports, a delay profile must be identified. Note it is possible for multiple combinations of the above properties to utilize the same O-RU delay profile. These delay profiles are O-RU specific, and are not within the scope of the O-RAN specification.

When calculating the O-DU transmit and receive window for a timing domain, the O-DU must use the delay profile applicable for each O-RU based on the air interface properties used by the O-RU in the specific network configuration. As previously noted, it is possible that the O-DU design is not capable of adjusting its transmit/ delay windows to meet the required transmit or receive windows.

The O-RU and O-DU may have multiple delay profiles depending on the design. The following **Table 2-16** indicates the contents of each profile.

Table 2-16 : DL and UL Delay Profiles

	O-RU delay profile	O-DU delay profile
Downlink	T2a_min_up, T2a_max_up, T2a_min_cp_dl, T2a_max_cp_dl, Tcp_adv_dl	T1a_max_up$_{\text{O-DU}}$ TXmax$_{\text{O-DU}}$
Uplink	Ta3_min, Ta3_max, T2a_min_cp_ul, T2a_max_cp_ul	Ta4_max$_{\text{O-DU}}$ RXmax$_{\text{O-DU}}$

Using the delay parameters specified for the corresponding profile for each O-RU in the timing domain, the uplink and downlink O-RU parameters to use for the timing domain can be determined as specified in section 2.3.3.1 **Table 2-12**. The transport delay parameters are constant for this method and specified as part of the network. The delay relationships as previously defined in **Table 2-8** and **Table 2-9** can then be applied to determine the O-DU transmit and receive delay parameters. Finally, the O-DU uplink and downlink constraints as defined in **Table 2-11** are applied to ensure that the calculated windows can be supported.

2.3.3.3 Measured Transport Method

With the Measured Transport method, the O-DU transmit and receive windows are determined based on the delay characteristics of the O-RUs as defined in section 2.3.3.2, and measured transport delays between all O-DU ports and O-RU ports in the timing domain. The O-DU must measure the transport delay(s) for all O-RU/O-DU port pairs in the timing domain and adapt its transmit and receive windows to accommodate the measured transport delay characteristics.

O-RAN specifies the use of the One-Way Delay Measurement messages as defined in the eCPRI specification. While other methods are possible, both endpoints must support a common approach. Some of the benefits of using the eCPRI approach are:

- Same ethernet type and address (and IP address/ UDP port) between O-DU and O-RU as used for C-Plane and U-Plane in order to obtain consistent measurements.
- eCPRI One-Way Delay Measurement allows for varying packet sizes to better simulate real traffic.
- eCPRI approach allows for separate T12 and T34 measurements, and measurements in both directions may be initiated by either communication peer.

Note that for this delay measurement eCPRI must be used because no other specified transport mechanism supports one-way delay measurements – this is true even if RoE is used as the C-Plane and U-Plane transport mechanism.

Table 2-17 : eCPRI One-Way Delay Measurement Message

One-Way Delay Measurement (Type 5)									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
ecpriVersion				ecpriReserved			ecpriConcatenation	1	Octet 1
ecpriMessage = 5								1	Octet 2
ecprPayload								2	Octet 3
Measurement ID								1	Octet 5
Action Type								1	Octet 6
TimeStamp (seconds)								6	Octet 7
TimeStamp (nanoseconds)								4	Octet 13
Compensation value (nanoseconds)								8	Octet 17
Dummy bytes								L	Octet 25
									Octet M

The O-DU initiates the measurements for T12 by sending a One-Way Delay Measurement request to the O-RU. The O-RU responds to the One-Way Delay Measurement response message. The request has a timestamp based on the transmission time at the O-DU (t1) and a Compensation value to account for expected processing time from t1 until actual packet transmission at R1.

The O-RU timestamps the received request message (t2). The O-RU creates a response message which contains the receive timestamp (t2) and a compensation value to account for expected delay from packet receipt at R2 to timestamping in the O-RU. O-DU then computes T12 using the formula indicated in **Figure 2-12**. t_{cv1} and t_{cv2} are implementation-specific "compensation" values an O-DU and O-RU may use to accommodate known timestamp processing delays. Note that this measurement process including the definition of the terms in the figure is described in the eCPRI specification [3].

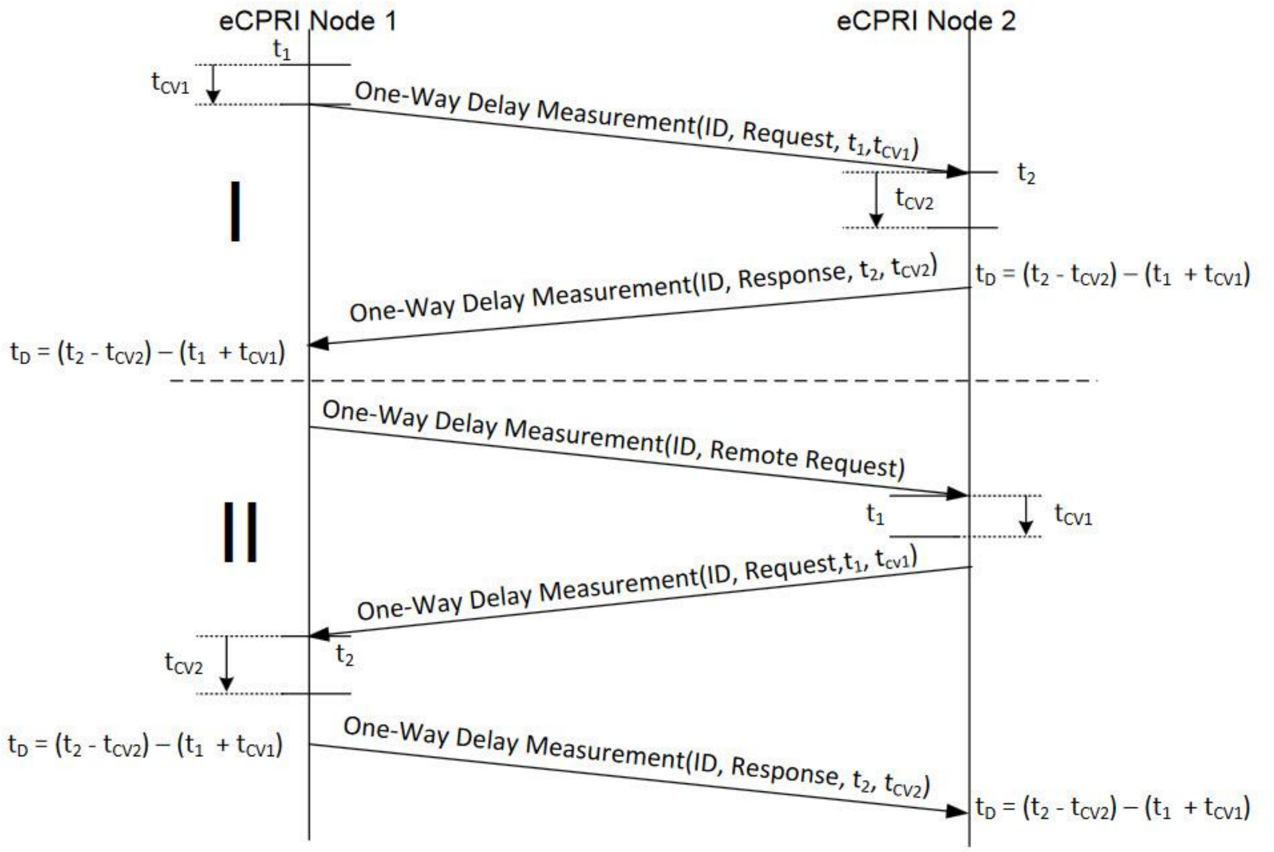


Figure 2-12 : eCPRI One-Way Delay Measurement procedure (Figure 25 of [3])

Note that when measuring the one-way delay, the goal is to determine the minimum transport delay. Since any individual packet can experience different delays through the network due to PDV, it is necessary to perform multiple measurements. Since the desired value is the minimum (T12_min or T34_min), the minimum delay measurement among the various measurements is used to estimate the minimum transport delay. The O-DU may use the estimated T12_min value as the T34_min value. It is optional for the O-DU to request the O-RU to perform a measurement of T34 using the One-Way Delay Measurement remote request.

There is no theoretical upper bound for transport delay. The maximum values are determined based on the measured minimum values plus a maximum allowed transport variation (PDVmax) defined as part of the SLA for the transport network.

The O-DU must measure the delay corresponding to each O-DU/ O-RU port pair within the timing domain. The per port pair minimum delay are then used as per **Table 2-12** to provide the transport parameters to be used in the delay relationships specified in **Table 2-8** and **Table 2-9** . The resulting delay configuration must meet the O-DU constraints defined in **Table 2-11**.

Since the transport delay is necessary to establish correct timing, this measurement is performed initially before sending of C/U-Plane traffic. The measurement should be performed with packets using the same L2 Cos Priority or L3 DSCP Code as the U-Plane data uses, to ensure the measurement accurately reflects the U-Plane transport latency. This could also be done periodically to verify delay if desired. Note however that section 2.4 describes traffic counts which can also be used for error detection. If an error is detected, it may be necessary to re-measure the delay and update the O-DU timing based on the new measurements.

2.3.4 Latency Categories for O-DU with dynamic timing advance

As an aide to selection of equipment for use in specific transport network use cases, O-RAN O-DU and O-RU are categorized based on delay capabilities. Equipment is categorized with a Category [A-Z] and Sub-category [.00-.40]. The category can be used to determine the maximum T12max/ T34max which the equipment can support. Categories are determined as follows:

- **T1a_max**$_{O\text{-}DU}$ ≥ T12max + T2a_min_up
- **Ta4_max**$_{O\text{-}DU}$ ≥ T34max + Ta3_max_up

By definition **T1a_max_up**$_{O\text{-}DU}$ is the earliest that the O-DU can begin transmission. This results in that the earliest that the O-DU can be guaranteed to complete transmission is **T1a_max_up**$_{O\text{-}DU}$ **- TXmax**$_{O\text{-}DU}$**.** The result is that:

- **T1a_max_up**$_{O\text{-}DU}$ **- TXmax**$_{O\text{-}DU}$ - T2a_min_up ≥ T12max

Similarly, for uplink **Ta4_max**$_{O\text{-}DU}$ is the latest that the O-DU may receive any packet from the O-RU. Therefore:

- **Ta4_max**$_{O\text{-}DU}$ - Ta3_max_up ≥ T34max

Both of these are illustrated in the **Figure 2-13** below:

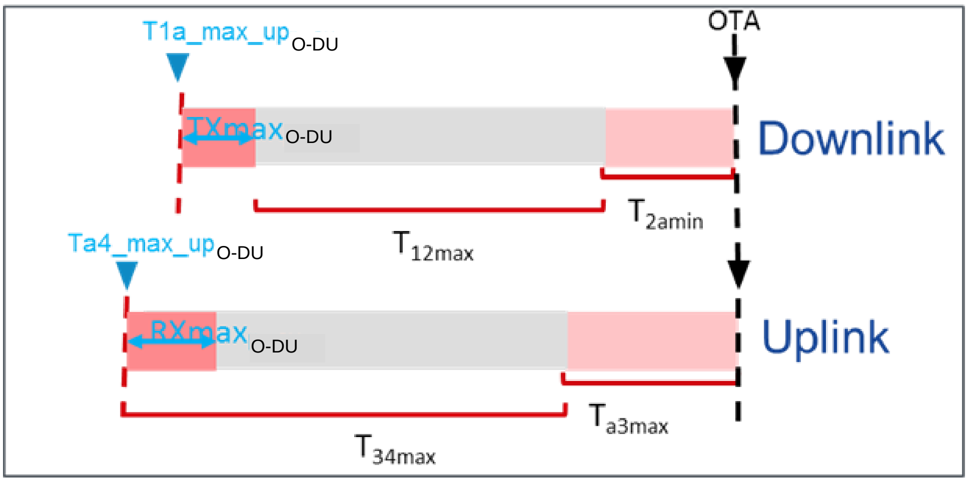


Figure 2-13 : Uplink and Downlink Timing Parameter Relationship

Using these inequalities, O-DU and O-RU can be categories based the following constraints:

Parameters	Constraint			
	Description	lls-CU	RU Processing	Transport
Downlink	Max Transport	T1a_max_up$_{lls\text{-}CU}$ - TXmax$_{lls\text{-}CU}$	T2a_min_up	**T12_max** = T1a_max_up$_{lls\text{-}CU}$ - TXmax$_{lls\text{-}CU}$ – T2a_min_up
Uplink	Max Transport	Ta4_max_up$_{lls\text{-}CU}$	Ta3_max	**T34_max** = Ta4_max_up$_{lls\text{-}CU}$ – Ta3_max

Note that the maximum range that can be supported is based on the minimum value of T12_max/ T34_max from above. This requires separate categorization for UL and DL for each O-DU and O-RU. Therefore, an uplink category pair (AX) and downlink category pair (BY) must be evaluated for each pairing of O-DU and O-RU. The minimum distance (T12_max/ T34_max) represented by the corresponding pairs (AX/ BY) then identifies the maximum transport latency which can be supported by the pair.

Each endpoint has a specific value for the characteristics defined in the table immediately above. In order to categorize endpoints, the endpoints are grouped based on the range in which the maximum transport characteristic falls. The category identification then means that the maximum transport characteristic for the endpoint falls somewhere between the maximum and minimum values for the category.

Since the category identifies that the maximum latency supported for the endpoint falls somewhere within the category range, the resulting category pair (AX) defines a range of T12_max/ T34_max for the combination. The combination can be used to identify a maximum T12_max/ T34_max and minimum T12_max/ T34_max that the combination can support. The category pair identifies a best and worst case for the maximum range the combination can support. The actual maximum range supported by a specific O-DU/ O-RU with a given category combination will fall somewhere within the range. That is, for a given O-DU with category A and O-RU with category X, the T12_max/ T34_max supported by the pair is defined as:

- $\text{Latency\_min}_{AX} \leq (\text{T12\_max} = \text{T34\_max}) \leq \text{Latency\_max}_{AX}$

Additionally, the paired combination can be mapped to supported use cases. Annex B.1 contains the category definitions, Latency_min, Latency_max, and use case tables.

Note that categories are only useful for identifying the maximum possible range (T12_max/ T34_max) values. To be able to usefully identify equipment for a given use-case, it is also necessary to be able to identify T12_min/ T34_min values. This can be defined in terms of variation, where variation indicates how much lower T12_min/ T34_min can be relative to T12_max/ T34_max respectively. That is:

- DL Variation = T12_max – T12_min
- UL Variation = T34_max – T34_min

An additional level of categorization is required to address the transport delay variation (e.g. T12max – T12min) that a given equipment type pair can support. This is addressed using sub-categories. Note that the total variation that can be supported is a combination of the receive window at the receiver, and the transmit variation. However, the receive window will likely be much larger than the transmit variation. Each endpoint is assigned to both an uplink and downlink category. The O-DU DL sub-category is based on transmit variation, while the O-DU UL sub-category is based on receive window. For UL, the roles are reversed and the O-RU DL sub-category is based on receive window, and the UL sub-category is based on transmit variation.

As a result, sub-categories are defined based on the role for the endpoint in the given transmission direction (UL/ DL).

Downlink variation ($T_{\text{varition\_DL}}$) is defined in **Figure 2-14** as follows:

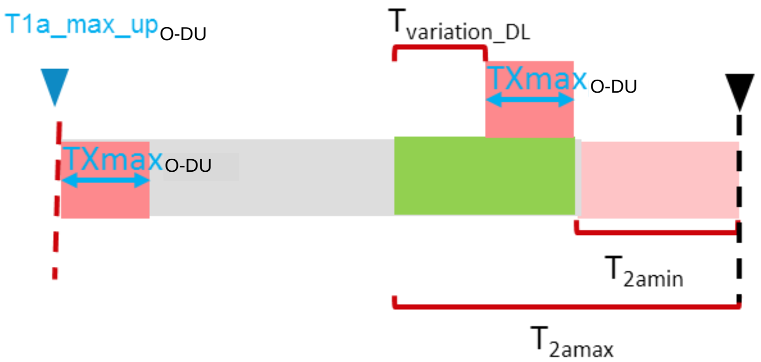


Figure 2-14 : Downlink Transport Variation

$$T_{\text{variation\_DL}} \leq \text{T2a\_max\_up} - \text{T2a\_min\_up} - \text{TXmax}_{\text{O-DU}}$$

Uplink variation ($T_{\text{varition\_UL}}$) is defined in **Figure 2-15** as follows:

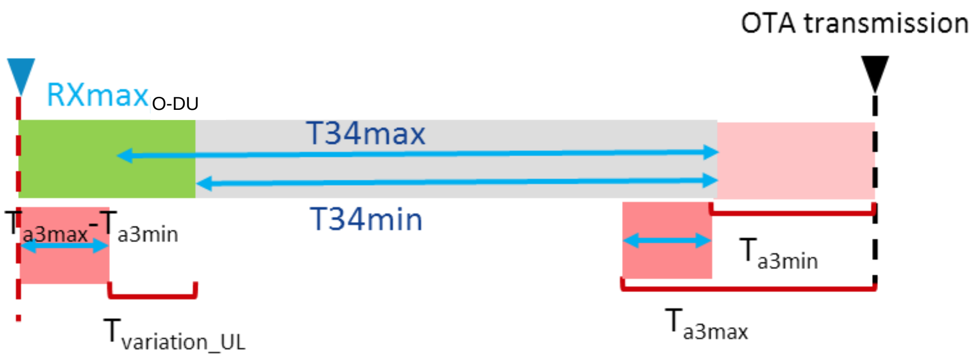


Figure 2-15 : Uplink Transport Variation

$$T_{\text{variation\_UL}} \leq \text{RXmax}_{\text{O-DU}} - (\text{Ta3\_max} - \text{Ta3\_min})$$

Using the respective constraints, the variability can be defined based on O-DU and O-RU delay characteristics as follows:

Parameters	Constraint			
	Description	lls-CU	RU Processing	Transport
Downlink	Max Variation	$TXmax_{lls\text{-}CU}$	T2a_max_up – T2a_min_up	**T12_max – T12_min** = (T2a_max_up – T2a_min_up) - $TXmax_{lls\text{-}CU}$
Uplink	Max Variation	$RXmax_{lls\text{-}CU}$	Ta3_max_up – Ta3_min_up	**T34_max – T34_min** = $RXmax_{lls\text{-}CU}$ – (Ta3_max_up – Ta3_min_up)

Sub-categories for each direction are defined using the endpoint characteristics. Similar to categories, sub-categories indicate that the endpoint has a variation characteristic that falls somewhere in the specified sub-category range. Since transmit variability and receive windows are expected to have different ranges, separate tables are created. However, since sub-categories are numeric [.##], to simplify these values are assigned in a similar way. The ## value represents the time in 10's of µsec for the given endpoint property.

For example, if an O-RU with a category X has a T2a_min – T2a_max = 163 µsec, the sub-category for the O-RU would be [.16]. Making the full category for the O-RU X.16. Note that by using this approach, the receiver sub-category and transmitter sub-category can be directly used to determine the dynamic range in either direction:

TX sub-category: .06

RX sub-category: .16

Resulting $T_{variability}$ = .16 * 1000 - .06 * 1000 = 100 µsec. Note that this provides a lower bound on the maximum dynamic range supported in one direction. The minimum of the UL and DL $T_{variability}$ is used to determine the maximum dynamic range for the combination (e.g. maximum difference between T12_max and T12_min that can be supported).

Annex B.2 contains the sub-category definitions and resulting combination dynamic range use case tables.

With dynamic timing advance, since the O-DU can set its earliest transmission time based on the measured T12_max, the full range of variability is available regardless of the T12_max. For example, if an O-DU/ O-RU combination has 100 usec worth of variability, and a maximum range T12_max of 250 usec, the combination can support the measured T12_max (≤ 250 usec) and T12_min of T12_max – 100 usec. So, if the measured T12_max is 200 usec, the combination can support T12_max of 200 usec and T12_min of 100 usec.

2.3.5 Latency Categories for O-DU with fixed timing advance

The same category concepts apply for O-DU which support fixed transmit/ receive windows. The earliest transmit window and latest receive window timing defines the maximum range that the O-DU can support. $T1a\_max\_up_{O\text{-}DU}$ for the O-DU is equal to the T1a_max_up of the earliest fixed transmit window. Similarly, $Ta4\_max_{O\text{-}DU}$ is equal to the Ta4_max of the latest receive window. The O-DU category is then assigned based on this value.

Fixed transmit and receive windows do however impact variability. The fixed transmit time results in sending packets earlier than absolutely needed to achieve the actual T12_max (i.e. the T12_max value that is either measured or configured by the service provider). The following figure **Figure 2-16** illustrates:

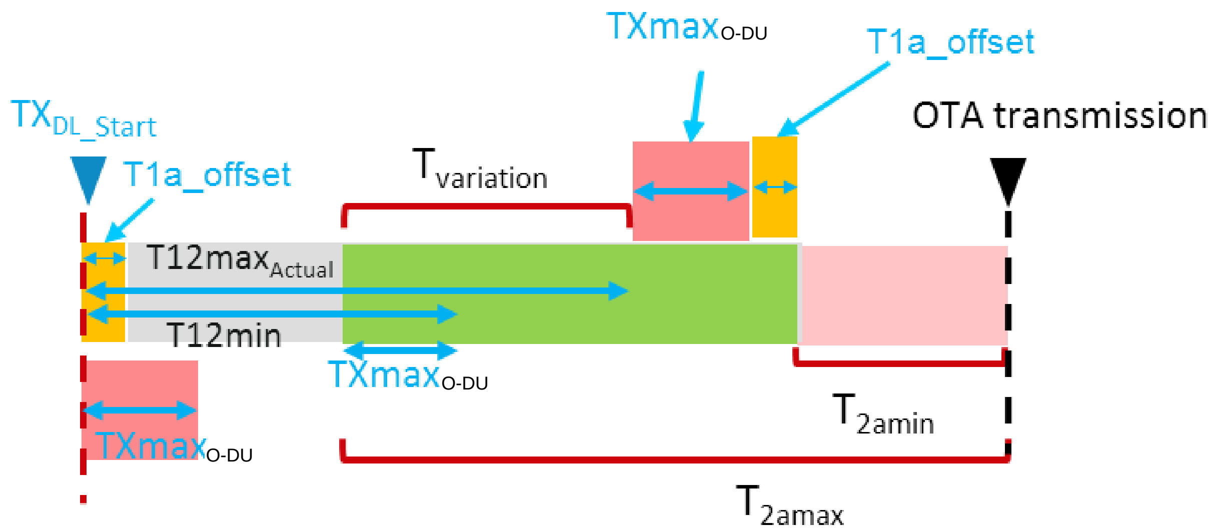


Figure 2-16 : Fixed Transmit Time Illustration

Note that since the TX_{DL_Start} is earlier than required for T12_max by T1a_offset, the latest packets arrive at least T1a_offset before T2a_min_up. This results in T1a_offset worth of O-RU receive buffer being always used, implying that the O-RU needs additional buffer to compensate T1a_offset. Since the TX_{DL_Start} is fixed, the T12min is also fixed at:

$$T12min = TX_{DL\_Start} - (T2a\_max\_up - TXmaxO\text{-}DU)$$

Since T12min is unaffected, the resulting $T_{variation}$ is reduced by T1a_offset. To support T12min, O-RU need to provide T1a_offset + T12max – T12min + TXmax_O_CU

It should also be noted that when T12_max + T2a_min_up = $T1a\_max\_up_{O\text{-}DU}$ (e.g. at maximum range), then T1a_offset = 0. But as the actual T12_max decreases, the amount of receive buffer consumed by T1a_offset increases by the corresponding amount.

Note that the same applies for UL. The result is that the T34_min is defined relative to the T34_max for the combination, not relative to the measured T34_max.

2.3.6 Non-Delay Managed U-Plane Traffic

While most U-Plane data must meet the delay constraints described above, there are certain types of U-Plane traffic for which the reception windows may not be applicable. Not applying the transmission/ reception window constraints to such data allows the transmission of the data over the fronthaul interface to be spread to reduce the peak bandwidth required by the interface.

Figure 2-17 uses SRS to illustrate an example non-delay managed U-Plane traffic.

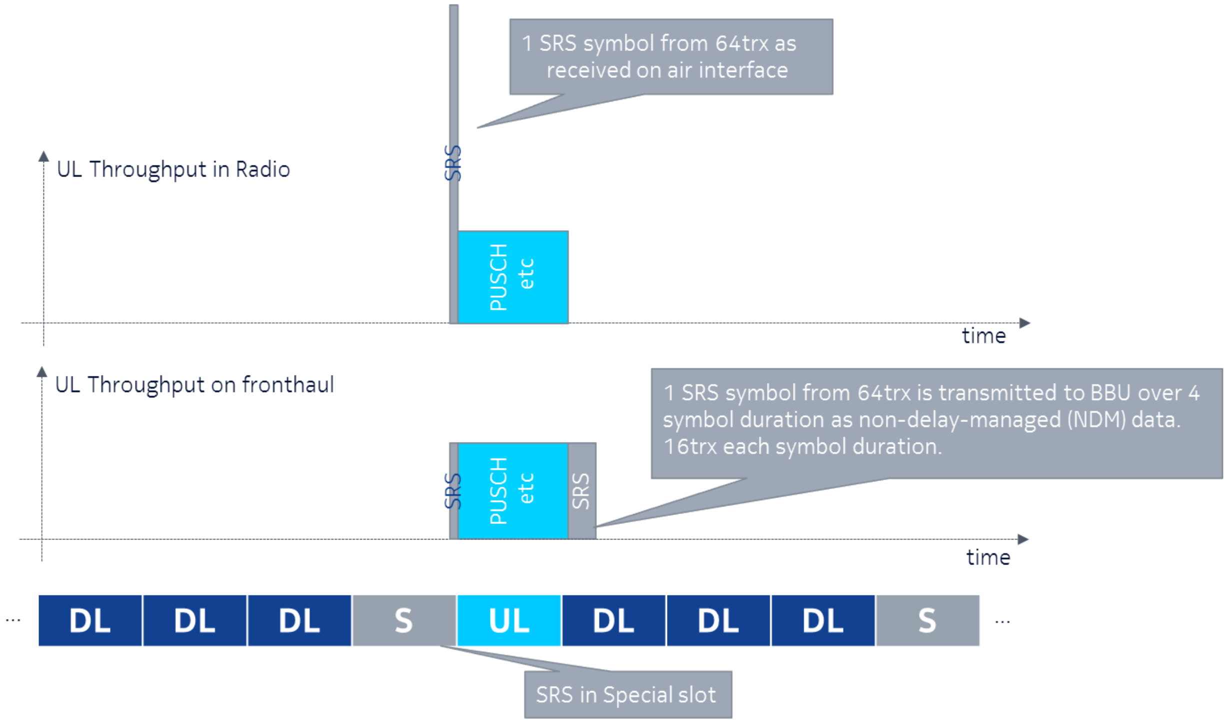


Figure 2-17 : SRS as non-delay managed U-Plane traffic

In the example, a large amount of SRS data is received during a special slot on the air interface. By allowing the SRS data to be transmitted from the O-RU to the O-DU outside of the normal transmit window, and instead allowing it to use otherwise un-used fronthaul interface bandwidth, the required peak fronthaul bandwidth requirement can be substantially reduced.

Non-delay managed U-Plane traffic flows must be uniquely identified from delay managed U-Plane traffic flows. The identification of U-Plane traffic flows using eAxC identifiers is described in section 3.4, and the eAxC must be used to indicate delay-managed versus non-delay-managed traffic (e.g. some eAxC or Pcid-identified packets may carry delay-managed U-Plane traffic while other eAxC or Pcid-identified packets may carry non-delay-managed U-Plane traffic). Non-delay managed U-Plane traffic flows may not use the same eAxC identifiers as delay managed U-Plane traffic flows. Different RU_Port_IDs are allocated within the c_eAxC ID for non-delay managed U-Plane flows from those used for delay managed U-Plane traffic flows in the same direction (UL/DL). It is not required that the RU_Port_IDs used in UL for non-delay managed traffic be the same as those used in the DL.

Non-delay managed traffic may be sent by the sender without regard to the transmission window for the corresponding air interface symbol. Similarly, the receiver will not discard received non-delay managed traffic received outside the normal reception window for the corresponding air interface symbol. However, the air interface symbol time is still contained within the U-Plane packets for reference.

Note that the status of traffic flows being delay-managed versus non-delay-managed (which is managed within the application) is independent of any transport prioritization that may be configured.

2.4 Reception Window Monitoring

The reception window can be used for purpose of error detection. The reception windows at O-RU and O-DU absorb delay variation in the fronthaul. Although its maximum size may vary depending on implementation, the position of windows at O-DU may be adjustable via parametrization as described in section 2.3.

Data flow error detection during fronthaul operation is monitored and reported via counters based on receipt of packets relative to the reception window. The receiving node monitors the actual reception timing and generates counters regarding the status and whether packets were successfully received in proper timing.

The following counters are expected to be supported:

- Data was received on time (within reception window) – no action needed
- Data was received too early – link latency re-measurement may be needed

- Data was received too late – link latency re-measurement may be needed
- Corrupt/Incorrect header packet – alarm may need to be raised
- Duplicated packet – alarm may need to be raised
- Total messages received – allows easier calculation of link statistics

In the above list are listed several system reactions that "may' be needed. The specific reaction will be implementation-specific and depend on such considerations as the number of instances needed before justifying sending an alarm, the ability of the system to measure link latencies, or other design-dependent judgments. Additional details regarding the counts, alarms, and fault handling is outside of the scope of this current document.

Dynamic adaptation of windows at Tx side to handle larger delay values may be employed. For instance, fronthaul distance may be extended by handling a portion (or most) of the fixed delay component at Tx buffer side by adapting the transmission timing based on information provided by Rx side. In other words, rather than accommodating a latency range of zero to "max", it is possible to instead accommodate a longer latency of "offset" to "max+offset" (longer latency but same range). This is described in detail in section 2.3.

Chapter 3 Transport & Protocol Architecture

3.1 Transport Encapsulation Types

3.1.1 Ethernet Encapsulation

Ethernet can be used as transport mechanism for both U-plane and C-plane. In this case, messages are transmitted over standard Ethernet frames (See **Figure 3-1**). Further, both the eCPRI header and payload are contained within the Ethernet data field. For this encapsulation, either the eCPRI Ethertype or the IEEE 1914.3 Ethertype shall be used.

Preamble (8 Bytes)	Destination MAC Address (6 Bytes)	Source MAC Address (6 Bytes)	VLAN Tag (4 Bytes)	Type/Length (Ethertype) (2 Bytes)	Payload (42…1500 Bytes)	FCS (4 Bytes)	IFG (12 Bytes)

Figure 3-1 : Native Ethernet frame with VLAN

3.1.2 IP/UDP Encapsulation

IP/UDP can be used as transport mechanism for both U-plane and C-plane. In this case, messages are transmitted over standard IP packets and the encapsulation mechanism is identified by "IP" or "Jumbo" Ethertype (See **Figure 3-2** and **Figure 3-3**).

Preamble (8 Bytes)	Destination MAC Address (6 Bytes)	Source MAC Address (6 Bytes)	VLAN Tag (4 Bytes)	Type/Length (Ethertype) (2 Bytes)	IPv4 (20 Bytes)	UDP (8 Bytes)	Payload (0…1472 Bytes)	FCS (4 Bytes)	IFG (12 Bytes)

Figure 3-2 : Native IPv4 packet with VLAN

Preamble (8 Bytes)	Destination MAC Address (6 Bytes)	Source MAC Address (6 Bytes)	VLAN Tag (4 Bytes)	Type/Length (Ethertype) (2 Bytes)	IPv6 (40 Bytes)	UDP (8 Bytes)	Payload (0…1452 Bytes)	FCS (4 Bytes)	IFG (12 Bytes)

Figure 3-3 : Native IPv6 packet with VLAN

Note that while the numbers in **Figure 3-1**, **Figure 3-2** and **Figure 3-3** imply standard Ethernet packets, use of Jumbo frames (up to 9000 bytes in length) is not precluded in this specification.

3.1.3 Transport Headers

O-RAN allows for multiple different transport headers within the Ethernet payload to further describe how the application data is to be handled in the C-Plane and U-Plane. In each case the transport header is 8 bytes in length and provides basic data routing capabilities, including description of the data flow type, sending and reception port identifiers, ability to support concatenation of multiple application messages in a single Ethernet packet, and sequence numbering. The following sections describe the possible transport headers.

3.1.3.1 eCPRI Transport Header

The definition of the eCPRI transport header is shown in **Table 3-1** below.

Table 3-1 : eCPRI Transport Header Field Definitions

Section Type : any									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
ecpriVersion				ecpriReserved			ecpriConcatenation	1	Octet 1
ecpriMessage								1	Octet 2
ecpriPayload								2	Octet 3
ecpriRtcid / ecpriPcid								2	Octet 5
ecpriSeqid								2	Octet 7

Each field within the eCPRI Transport Header is further described in the following sub-sections.

3.1.3.1.1 ecpriVersion (eCPRI protocol revision)

Description: This parameter indicates the eCPRI protocol version. NOTE: This parameter is part of the eCPRI common header.

Value range: {0001b=eCPRI version 1.0, 0010b-1111b=Reserved for future eCPRI protocol revisions, 0000b= Reserved for future eCPRI protocol revisions}.

Type: unsigned integer.

Field length: 4 bits

Default Value: 0001b (eCPRI version 1.0).

3.1.3.1.2 ecpriReserved (eCPRI reserved)

Description: This parameter is reserved for eCPRI future use. NOTE: This parameter is part of the eCPRI common header.

Value range: {001b-111b=Reserved}.

Type: unsigned integer

Field length: 3 bits.

Default Value: 000b (reserved fields should always be set to all zeros).

3.1.3.1.3 ecpriConcatenation (eCPRI concatenation indicator)

Description: This parameter indicates when eCPRI concatenation is in use (allowing multiple eCPRI messages in a single Ethernet payload). NOTE: This parameter is part of the eCPRI common header.

Value range: {0b=No concatenation, 1b=Concatenation}.

Type: binary bit.

Field length: 1 bits.

Default Value: 0b (no concatenation).

3.1.3.1.4 ecpriMessage (eCPRI message type)

Description: This parameter indicates the type of service conveyed by the message type. NOTE: This parameter is part of the eCPRI common header. NOTE: In this version of the specification, only values “0000 0000b” and “0000 0010b” and “0000 0101b” are used.

Value range:
0000 0000b = IQ data message;
0000 0010b = Real-time control data message;
0000 0101b = transport network delay measurement message (see section 2.3.3.3 for full message format);
other values not recognized within this version of the specification.

Type: unsigned integer.

Field length: 8 bits.

Valid Values: 0x0 (U-Plane data) or 0x2 (C-Plane data) or 0x5 (network delay measurement messages).

3.1.3.1.5 ecpriPayload (eCPRI payload size)

Description: This parameter is the size in bytes of the payload part of the corresponding eCPRI message. It does not include any padding bytes following the eCPRI message. The maximum supported payload size is 2^{16}-1, but the actual size may be further limited by the maximum payload size of the underlying transport network. NOTE: This parameter is part of the eCPRI common header.

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}.

Type: unsigned integer.

Field length: 16 bits.

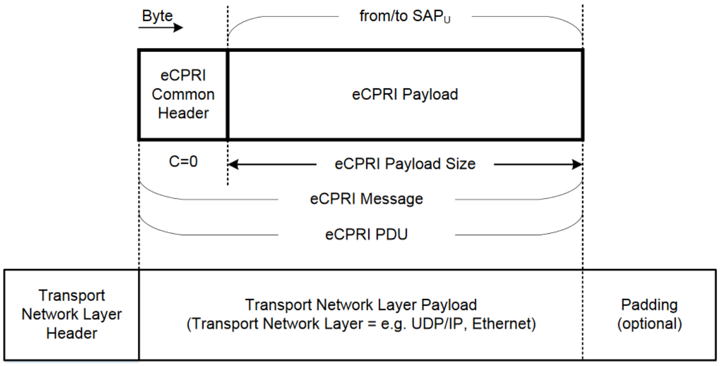


Figure 3-4 : eCPRI payload - no concatenation (source: [3])

3.1.3.1.6 ecpriRtcid / ecpriPcid (real time control data / IQ data transfer message series identifier)

Description: This parameter is a component_eAxC identifier (c_eAxC ID) and identifies the specific data flow associated with each C-Plane (ecpriRtcid) or U-Plane (ecpriPcid) message. It is the analog of CPRI’s “AxC” (antenna-carrier) value so is designated here as “eAxC” (“e” for “extended” to accommodate multiple bands and multiple component carriers). In addition, the “eAxC” is divided into “component eAxC” parts (c_eAxC) because multiple O-DU processors may contribute to a single eAxC and must be identified for correct data routing.

The data marked by an eAxC value has certain common characteristics, including having the same values of the parameters filterIndex, frame structure (from frameStructure or conveyed via M-plane) and cyclic prefix type (determined from frameStructure and cpLength or conveyed via M-plane).. This means that data streams having unique values for those parameters have unique eAxC values. In practical terms this means that PRACH data, SRS data, and mixed-numerology channels have specific eAxC values not shared with other types of data.

One eAxC identifier (eAxC ID) comprises a band and sector identifier (BandSector_ID), a component-carrier identifier (CC_ID) and a spatial stream identifier (RU_Port_ID). One eAxC identifier can consist of one or several c_eAxC IDs having same BandSector_ID, CC-ID, and RU_Port_ID, but different values of DU_Port_ID, the different values correlating with different O-DU processors.

Note: This parameter is specific to eCPRI Message Type = 2 (C-Plane) and Message Type 0 (U-Plane) messages. In this version of the specification, one eAxC contains only one spatial stream (i.e. one beam per subcarrier) at a time. When precoding in the O-RU, then each eAxC contains one layer at a time, except for TxD (LTE TM2) when a single eAxC (single ecpriRtcid and ecpriPcid) represents all TxD layers.

Bit allocation is subdivided as follows:

- DU_Port_ID: Used to differentiate processing units at O-DU (e.g., different baseband cards). It is expected the O-DU will assign these bits, and the O-RU will attach the same value to the UL U-Plane messages carrying the same sectionId data.
- BandSector_ID: Aggregated cell identifier (distinguishes bands and sectors supported by the O-RU).
- CC_ID: distinguishes Carrier Components supported by the O-RU.
- RU_Port_ID: designates logical flows such as data layers or spatial streams, and logical flows such as separate numerologies (e.g. PRACH) or signaling channels requiring special antenna assignments such as SRS. The assignment of RU_Port_ID as part of the eAxC is done by the O-DU via the M-plane.

The bitwidth of each of the above fields is variable and set via M-Plane messaging. This is to allow flexibility given it is expected that not all fields will simultaneously need their maximum range for any given O-RU. It is expected the M-Plane message will configure the O-RU and O-DU with the appropriate bitwidth of each of the four fields, and the NMS that does the actual assignment will assure all 16 bits are allocated (with or without padding).

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b = eAxC ID}

Bit allocations:

0 (msb)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15 (lsb)	Number of Octets
DU_Port_ID				BandSector_ID				CC_ID				RU_Port_ID				2

Type: unsigned integer (concatenated bit fields).

Field length: 16 bits.

3.1.3.1.7 ecpriSeqid (message identifier)

Description: This parameter provides unique message identification and ordering on two different levels. The first octet of this parameter is the Sequence ID, which is used to identify ordering of messages within an eAxC message stream. The Sequence ID field increments and wraps independently for each U-Plane eAxC DL, U-Plane eAxC UL, C-Plane eAxC DL, and C-Plane eAxC UL, even if they share the same eAxC ID. The Sequence ID is used to verify that all messages are received and also to reorder messages that are received out of order. The second octet of this parameter is the Subsequence ID. The Subsequence ID is used to verify ordering and implement reordering when radio-transport-level (eCPRI or IEEE-1914.3) fragmentation occurs. Radio-transport (eCPRI or IEEE-1914.3) fragmentation is a method of splitting U-plane messages containing one or more sections whose length exceeds the maximum packet or message length of the underlying protocol. The Subsequence ID field consists of a 7 bit Subsequence counter and a single bit field, called E-bit. The Subsequence number increments starting from zero for each fragment of a U-plane message. The E bit is used to indicate the last message of the radio-transport level fragments. It is always set to zero except for the last message of the U-plane fragment. In the case of C-plane messages radio-transport fragmentation is not allowed, therefore the Subsequence ID shall be set to zero, and the E bit set to one._ See **Section 3.5** for a description of the fragmentation process.

NOTE: As an alternative to radio-transport-level fragmentation, application fragmentation can be implemented. In this case the application can take the responsibility to ensure all transport messages are not too long (fit within the necessary transport payload size). When this "application layer fragmentation" is used, the subsequence identifier shall always be set to "0", and the E-bit set to "1" (See **Section 3.5**).

Figure 3-5 shows how the sequence numbers are intended to be generated and checked by C-Plane and U-Plane messaging.

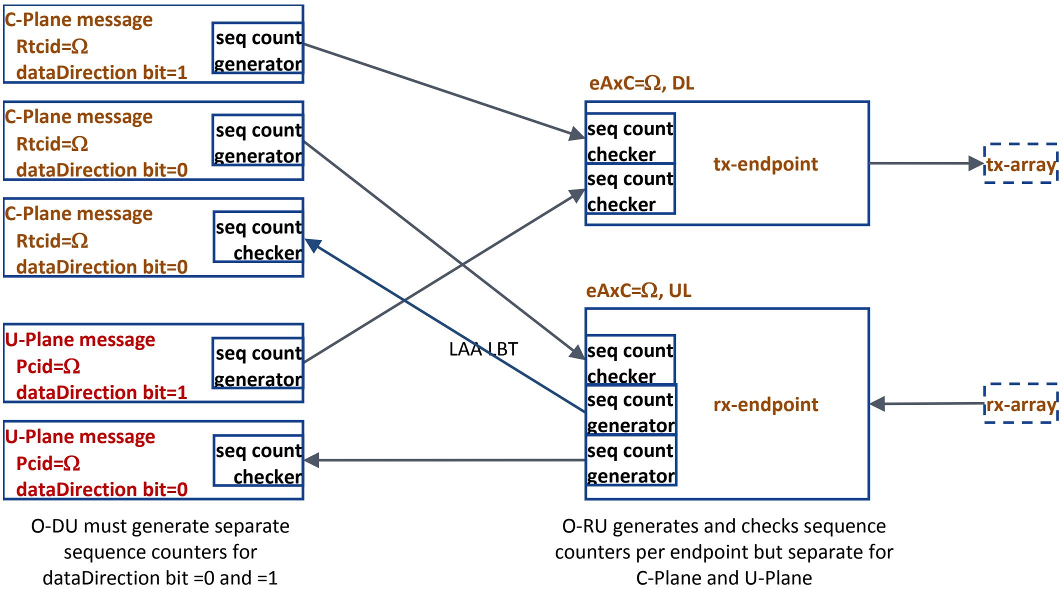


Figure 3-5 : Sequence Count Generation and Checking for C-Plane and U-Plane in O-DU and O-RU

Implications of **Figure 3-5** are as follows:

a) Sequence ID is unique per eAxC (ecpriRtcid or ecpriPcid): different eAxC values have their own Sequence ID generation
b) Sequence ID is generated by the fronthaul interface transmitter in either the O-DU or O-RU
c) The sequence generator and checker for a C-Plane message describing U-Plane DL is independent of the sequence generator and checker for a C-Plane message describing U-Plane UL.

Table 3-2 summarizes this situation consistent with **Figure 3-5**:

Table 3-2 : Description of Shared or Independent Sequence Generators

direction of message flow	type of flow	U-Plane data direction	Same/different eAxC between DL & UL	shared/independent sequence generator
O-DU → O-RU	C-plane	DL (dataDirection bit =1)	same or different	independent
O-DU → O-RU	C-plane	UL (dataDirection bit =0)	same or different	independent
O-RU → O-DU	C-plane	UL (dataDirection bit =0)	same or different	independent
O-DU → O-RU	U-plane	DL (dataDirection bit =1)	same or different	independent
O-RU → O-DU	U-plane	UL (dataDirection bit =0)	same or different	independent

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}

Bit allocations:

msb							lsb		
0	1	2	3	4	5	6	7	Number of Octets	
Sequence ID								1	Octet 1
E bit	Subsequence ID							1	Octet 2

Type: unsigned integer (concatenated bit fields)

Field length: Sequence ID: 8 bits; subsequence ID: 7 bits; E-bit: 1 bit.

3.1.3.2 1914.3 Transport Header

As an alternative to eCPRI as a transport header, IEEE 1914.3 may be used. The definition of the 1914.3 transport header is shown in **Table 3-3** below.

Table 3-3 : 1914.3 Transport Header Field Definitions

Section Type : any									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
RoEsubType								1	Octet 1
RoEflowId								1	Octet 2
RoElength								2	Octet 3
RoEorderInfo								4	Octet 5

Each field within the 1914.3 Transport Header is further described in the following sub-sections.

3.1.3.2.1 RoEsubType (sub type / message type)

Description: This field indicates the payload type within the IEEE 1914.3 Standard for Radio over Ethernet Encapsulations and Mappings (RoE) subType range. RoE allows RoE subTypes in the range 128 to 191 to be mapped to external organizations and companies using a subtype mapping table (below). This table has two fields. The OUI/CID field uses a unique Company ID (CID) value. For O-this version of this specification, the OUI/CID assigned by IEEE to xRAN (0xFAEB6E) shall be used. The payload structure mapping field assigns the RoE subtype to O-RAN message types. When the payload structure mapping field is set to 0xFFFF, this indicates that that RoE subtype is not mapped to an O-RAN message type.

A given O-RAN implementation may choose to have fixed (default) mappings, or alternatively, it may choose to configure the subType mapping table using control packets as described in IEEE 1914.3.

NOTE: The table below indicates the use by O-RAN of the IEEE-defined xRAN subtype mapping; future discussions with IEEE are needed to convert these to O-RAN subtype mapping.

Table 3-4 : RoE Subtype Mapping

RoE subType	OUI/CID subType mapping table (.mapSubtype)	
	OUI/CID Mapping (3 bytes) bit39 <----------------------------------	Payload structure mapping (2 bytes) ---> bit 0
128	xRAN=0xFAEB6E	0x0001, IQ (No concatenation)
129	xRAN=0xFAEB6E	0x0002, IQ (With concatenation)
130	xRAN=0xFAEB6E	0x0003, Ctrl (No concatenation)
131	xRAN=0xFAEB6E	0x0004, Ctrl (With concatenation)
132 to 191	xRAN=0xXXXXXX (don't care)	0xFFFF (IEEE1914.3 default), unused/unmapped by xRAN in this version of the O-RAN specification.

Value range: 128 to 191. Default values are shown below.

Type: unsigned integer.

Field length: 8 bits.

Valid Values: 128 (U-Plane, no concat), 129 (U-Plane, with concat), 130 (C-Plane, no concat), 131 (C0-Plane, with concat).

3.1.3.2.2 RoEflowID (flow identifier)

Description: The RoEflowID is a mechanism which can identify specific flows between end-points. RoEflowID, 0xFF is reserved for RoE control packets. O-RAN has no current use for this field.

Value range: 0 – 0xFE.

Type: unsigned integer.

Field length: 8 bits.

Description: This field is currently unused.

3.1.3.2.3 RoElength (length)

Description: This field is the size in bytes of the payload part of the message. The payload length field value is the total number of octets following the O-RAN common header. It does not include the Ethernet FCS or following bytes.

Value range: 0 – 0xFFFF.

Type: unsigned integer.

Field length: 16 bits.

3.1.3.2.4 RoEorderInfo (order information)

Description: This field is split into seven sub-fields.

Table 3-5 : RoE RoEorderInfo MappingField

Field	Length	Note
DU_Port_ID		Used to differentiate processing units at O-DU (e.g., different baseband cards). It is expected the O-DU will assign these bits, and the O-RU will attach the same value to the UL U-Plane messages carrying the same sectionId data. See sub-clause 3.1.3.1.6 for further information.
BandSector_ID	16 bits	Aggregated cell identifier (distinguishes bands and sectors supported by the O-RU). See sub-clause 3.1.3.1.6 for further information.
CC_ID		Distinguishes Carrier Components supported by the O-RU. See sub-clause 3.1.3.1.6 for further information.
RU_Port_ID		Used to differentiate spatial streams or beams on the O-RU. See sub-clause 3.1.3.1.6 for further information.
Sequence_ID	8 bits	Unique message ordering sequence. See sub-clause 3.1.3.1.7 for further information.
E_Bit	1 bit	Marks the last message pertaining to the section. See sub-clause 3.1.3.1.7 for further information.
Subsequence_ID	7 bits	Unique message ordering sub-sequence. See sub-clause 3.1.3.1.7 for further information.

Value range: 0 to 0xFFFF FFFF.

Type: unsigned integer (concatenated bit fields).

Field length: 32 bits.

3.2 Protocol Architecture

3.2.1 C-plane

Figure 3-6 depicts the protocol stack for C-Plane. Data can be optionally transmitted over IP Layer 3 if supported by the transmitting and receiving nodes.

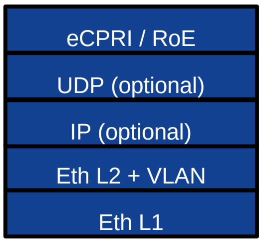


Figure 3-6 : C-plane protocol structure

3.2.2 U-plane

Figure 3-7 depicts the protocol stack for U-Plane. Data can be optionally transmitted over IP Layer 3 if supported by the transmitting and receiving nodes.

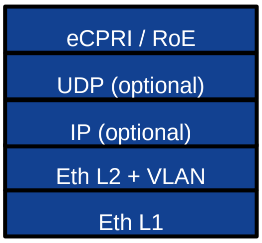


Figure 3-7 : U-plane protocol structure

3.2.3 S-plane

Frequency and time synchronization of O-DUs and O-RUs via Ethernet use Synchronous Ethernet and IEEE 1588-2008 Precision Time Protocol (PTP). Transport of PTP directly over L2 Ethernet (ITU-T G.8275.1 full timing on-path support) is assumed in this version of the specification, whilst transport of PTP over UDP/IP (ITU-T G.8275.2 partial timing support from the network) is also possible albeit with unassured synchronization performance. Security mechanisms will not be used for this plane. The protocol stack for PTP and SyncE over L2 Ethernet is depicted in **Figure 3-8**.

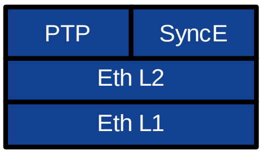


Figure 3-8 : S-plane protocol structure

3.3 Quality of Service

The LLS interface needs to support the ability to distinguish between data flows with different QoS requirements. Configurable priority levels (via the M-Plane) for traffic prioritization of flows that shall be supported on each node on the network path. Default values for the respective O-RAN planes are indicated on **Table 3-6**. Priority marking per packet is needed in each protocol layer. For operation at Layer 2, prioritization is performed by specifying a configurable value for the Priority Code Point (PCP) tag in the IEEE 802.1q VLAN header on the outgoing traffic.

Table 3-6 : Quality of service classes

Plane	L2 CoS Priority (range 0-7)	L3 DSCP Code	Preemption(2)
S-Plane	Default: 7 (1)	Not applicable	non-preemptable
U-Plane	Default: 7	EF (Expedited Forwarding)	non-preemptable
C-Plane	Default: 7	EF (Expedited Forwarding)	non-preemptable
M-Plane	Default: 2	AF2x (Assured Forwarding)	preemptable
Other traffic	Default: 1	BE (Best Effort)	preemptable
(1) Applicable if vLAN is applied which will be possible in the future. (2) Not all networks will support preemption so this only applies to networks supporting preemption			

For U-Plane separate priorities can be supported based on ecpriRtcid / ecpriPcid (or the equivalent fields in RoE packet headers). This allows for different prioritization of some channels (e.g. PRACH/ SRS), or services (URLLC). This requires the ability to configure priority assignments to ecpriRtcid / ecpriPcid via M-Plane messages. The default U-Plane priority can be applied to flows not specifically configured with a different value via the M-Plane. If a node does not support configurable ecpriRtcid / ecpriPcid priorities, then all U-Plane traffic will use the default value.

QoS failure (dropped packets) may be detected using the sequence numbers that make up part of the transport header. Loss of a packet in the U-Plane will generally impact only a specific symbol except in rare cases where a multi-symbol U-Plane message is sent (using the “symInc” field). Loss of a C-Plane packet is more serious, with the possibility of impacting an entire slot’s worth of data. A broader impact is also possible with C-Plane packet loss: if downloaded beamforming weights, beamforming attributes, or channel information is lost then a beamId may be incorrectly associated with an “old” set of beamforming weights thereby possibly impacting multiple slots. However, packet loss can be detected via the sequence number so a recovery action may be taken to re-send messages that may have a persistent impact.

3.4 Data Flow Identification

Differentiation between U/C-Plane and Management Plane traffic can be achieved using the following options:

- Data flow separation based on TCP/UDP (applicable when layer 3 transport is used for the C/U-plane)
- Data flow separation based on VLAN (applicable when layer 2 or layer 3 is used for the C/U-plane transport)
  - Note: The mechanism for assigning VLAN ID to U-Plane and C-Plane is assumed to be via the M-Plane
- Data flow separation based on usage of different MAC addresses (applicable when Layer 2 is used for C/U plane transport)
  - e.g., separate MAC address used for M-Plane and U-Plane, or for baseband load sharing purposes
- Data flow separation based on different EtherTypes (applicable when Layer 2 is used for C/U plane transport)

The U-plane application also needs to uniquely associate different data flows (e.g. spatial streams) each with a unique U/C plane endpoint address. This can be achieved un an O-RU using the eAxC identifier, and in the O-DU using the eAxC identifier in combination with transport-based endpoint identifiers to differentiate O-RUs. Different transport identifiers (e.g. UDP/IP, VLAN, MAC) can be used based on whether layer 2 or layer 3 transport is used for the U/C plane:

- U/C-plane data flow separation using UDP-port identifiers (applicable when layer 3 transport is used for the C/U-plane)
- U/C-plane data flow separation using on VLAN identities (applicable when layer 2 or layer 3 is used for the C/U-plane transport)
- U/C-plane data flow separation based on usage of different MAC addresses (applicable when Layer 2 is used for C/U plane transport)

3.5 Fragmentation

Fragmentation is applied in case data (U-Plane or C-Plane data) with Ethernet transport overheads to be transferred exceed maximum transmission unit (MTU) of the network. This specification allows two methods for fragmentation, application layer fragmentation and transport layer fragmentation:

3.5.1 Application layer fragmentation

- Application creates C-plane or U-Plane messages, which when including overheads can fit to MTU requirements set by network
- Sequence ID: Sequence ID increases for every message, E=1, Subsequence ID=0

An example of application fragmentation is provided in **Figure 3-9,** wherein a long data section is split between multiple U-Plane messages. In the example it is a single data section (0xABC) so described as such in the C-Plane, and in the U-Plane the sectionId is cited multiple times in the multiple U-Plane messages but with differing and consecutive groups of PRBs as indicated by the startPrbu and numPrbu fields. The I and Q word width in the example is 14 bits, which for such a large data section requires multiple U-Plane messages. The ecpriRtcid / ecpriPcid (0xDEF3) is shown with a random example value, as is the sequence ID in the C-Plane (0x99) and independently in the U-Plane (0x40-0x48).

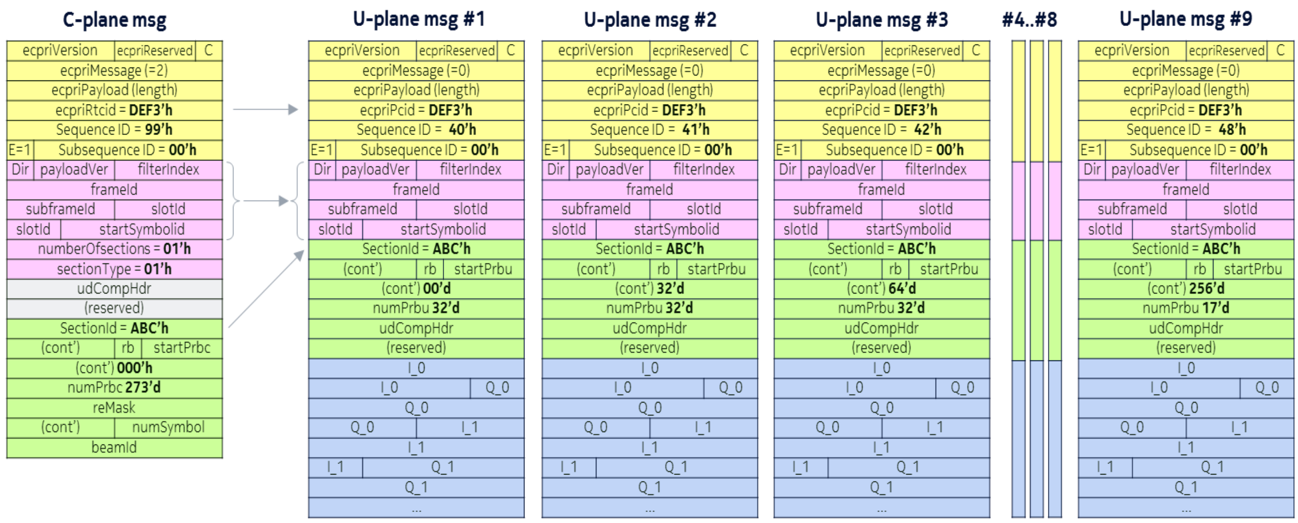


Figure 3-9 : Example of Application-Level Fragmentation of U-Plane Messages

3.5.2 Radio Transport layer (eCPRI or IEEE-1914.3) fragmentation

- Application creates U-Plane messages, which when including overhead may exceed MTU requirements set by network
- Radio transport layer splits message which may contain more than one section into pieces such that the fragments with overheads fit to MTU requirements set by network.
- Sequence ID: Sequence ID remains same for all the fragments. Subsequence ID starts from 0 for the first fragment and counts up for each fragment. Last fragment flagged with E=1, others E=0.

Table 3-7 : Example of Sequence Numbers Usage

Application Layer Fragmentation			
eAxC	SeqId	E	Sub-SeqId
Ω	0	1	0
Ω	1	1	0
Ω	2	1	0
Ω	3	1	0
Ω	4	1	0
Ω	5	1	0
Ω	6	1	0
Ω	7	1	0
Ω	8	1	0
Ω	9	1	0

eCPRI Layer Fragmentation			
eAxC	SeqId	E	Sub-SeqId
Ω	0	0	0
Ω	0	1	1
Ω	1	0	0
Ω	1	0	1
Ω	1	0	2
Ω	1	1	3
Ω	2	0	0
Ω	2	0	1
Ω	2	1	2
Ω	3	1	0

3.5.3 Fragmentation Guideline

Application layer fragmentation should be applied so that maximum size standard IEEE 802.3 Ethernet frames can be used. (Jumbo frames, if used, increase the maximum MTU size).

- In case of L2 only solution, application layer maximum transmission unit size is standard IEEE 802.3 Ethernet frame payload size (1500 bytes) – transport overhead (8 bytes) = 1492 bytes (or larger for Jumbo frames)

When a U-plane section payload is fragmented due to large number of PRBs, it will be divided into multiple groups of PRBs such that each group (including the application headers) can fit to the MTU requirements. Each group of PRBs shall have its respective control fields including the same section ID from the C-plane and startPrbu and numPrbu to identify its contents (See **Figure 3-9**). When an application layer message exceeds the MTU size set via the M-Plane configuration, radio-transport layer fragmentation can be applied at the transport layer and has the advantage of allowing application layer messages to fit whatever is the MTU size without the application having to know about transport-level details.

Chapter 4 Security

Security requirements are depicted in **Table 4-1**.

Table 4-1 : Security requirements for User-Plane, Control-Plane, and Synchronization-Plane

Plane	Integrity (protection from modifications)	Confidentiality (encryption protection)	Availability (protection from packet insertion)	Remarks
U-Plane	No requirement	No requirement	No requirement	User data protected end to end via PDCP protocol
C-Plane	No requirement	No requirement	No requirement	
S-Plane	No requirement	No requirement	No requirement	Optional in IEEE 1588 (PTP). However, not feasible at a reasonable cost.

Chapter 5 C-plane Protocol

5.1 General

5.2 Function

5.2.1 C-Plane Transport

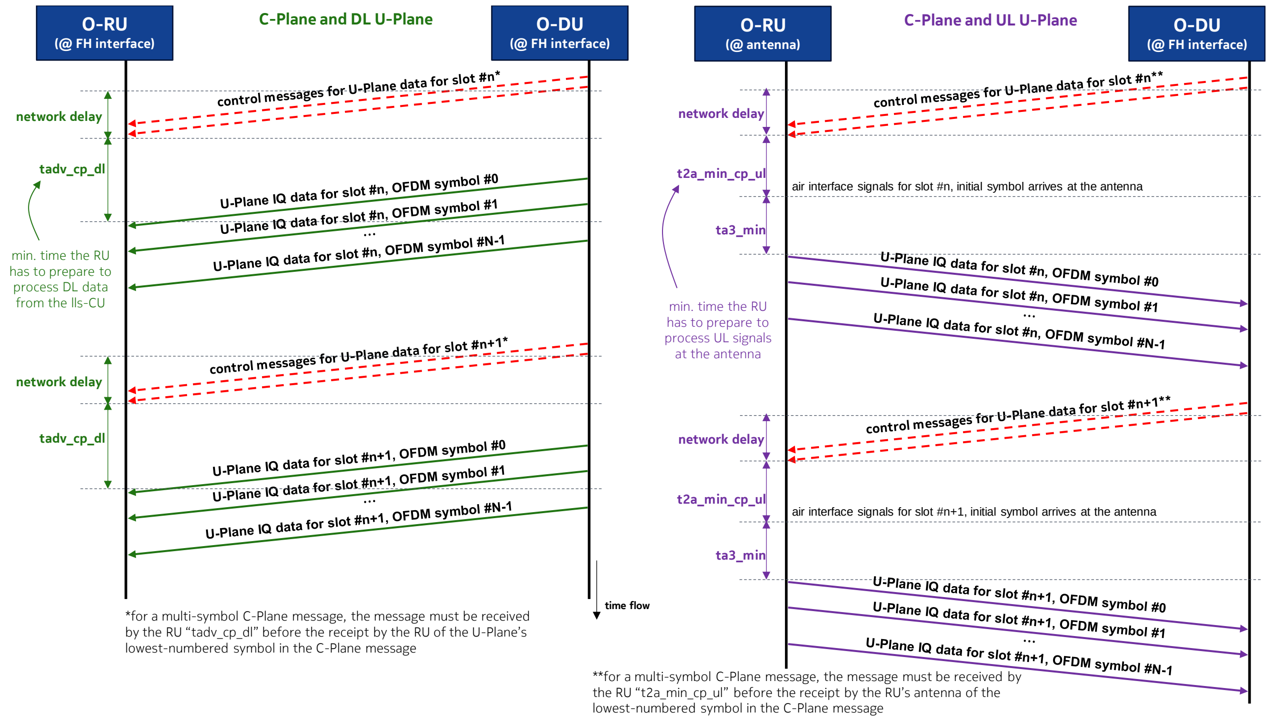
Either eCPRI or IEEE 1914.3 is used as an encapsulation mechanism for the control-plane messages. Due to the nature of these messages (very strict delay constraints), it is assumed that message acknowledgements are not possible. Likewise, it is assumed that a different data flow is used other than the U-Plane channel. Further, C-Plane messages are not concatenated with U-Plane messages within same Ethernet frame.

5.3 Elementary Procedures

5.3.1 Scheduling and Beamforming Commands Transfer procedure

This procedure is used to exchange C-Plane messages between O-DU and O-RU. The main purpose of these messages is to transmit data-associated control information required for processing of user data (e.g., scheduling and beamforming commands). Messages are sent separately for DL related commands and UL related commands (see **Figure 5-1**). See also **Figure 5-7**.

for the special case of LTE LAA wherein there are UL as well as DL C-Plane message flows. For purpose of increased flexibility, C-Plane messages may be sent either jointly or separately depending on the channel for which information is conveyed. For example, PUCCH and PUSCH may be bundled or not bundled into a single C-Plane message depending on implementation.



-

Figure 5-1 : Scheduling and beamforming commands transfer procedure

5.3.2 Mixed Numerology and PRACH Handling

5G NR defines Physical Resource Block (PRB) where the number of subcarriers per PRB is the same for all numerologies (twelve). However, subcarrier spacing may differ, resulting in mixed numerology used in time and frequency domains respectively. Likewise, mixed numerologies may be applied to both DL and UL respectively **(Figure 5-2**).

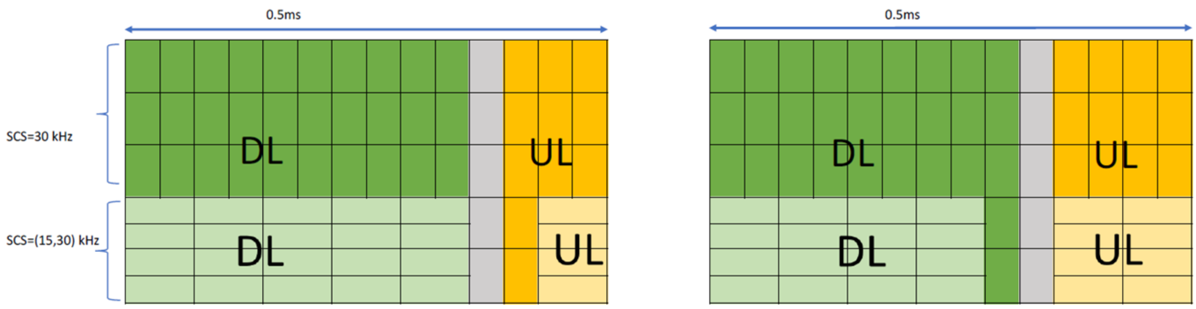


Figure 5-2 : Mixed Numerology in Time and Frequency Domain

__Frequency indexing across multiple numerologies__

- RB indexing (i.e. the physical resource block) is always dependent on the used numerology. For PRACH and SSB the relation is with the physical channel. In case of PRACH, indexing of RBs inside of each PRACH frequency block follows 3GPP specifications and PRACH subcarrier extraction is performed in the CU. In particular, the first RE of the first RB addressed inside section Id shall corresponds with the first guard tone used at the lower edge of the PRACH block frequency block and based on the PRACH SCS. The O-RU, using the filter index knows the PRACH subcarrier position and guard tones and can then adjust its filtering parameters (e.g. filter center, passband). For the SSB case, its own separate frequency offset is assigned.
- For each numerology (and PRACH/SSB), the freqOffset IE defines the relative spacing between the center of RE#0 of RB#0 to the component carrier center (i.e. equivalently DC=0Hz when normalized with regard to the component carrier center), and using resolution of half the SCS of the respective numerology. This concept is depicted in **Figure 5-3**, **Figure 5-4** and **Figure 5-5**.
    - The component carrier center (RF frequency in Hz) is the common reference to all numerologies and PRACH/SSB channels. Further, the relation between DC=0Hz of the carrier and ARFCN (RF frequency in Hz of the carrier center) is to be defined at carrier setup over M-Plane.
    - Spacing resolution of 0.5 x SCS allows DC to be aligned with an RE center, or and RE edge.

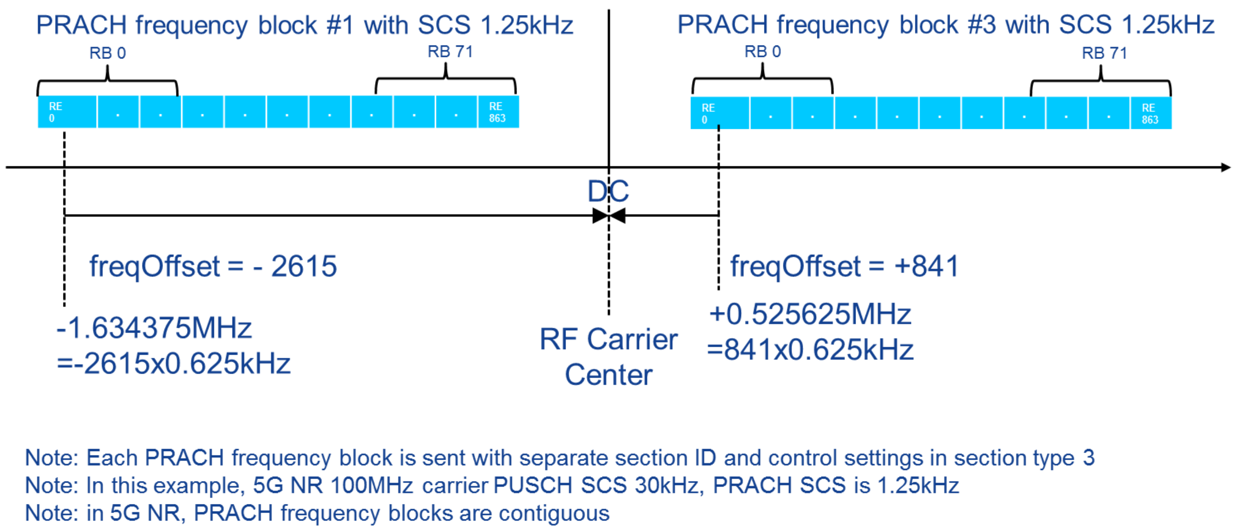


Figure 5-3 : RB Mapping– PRACH Example

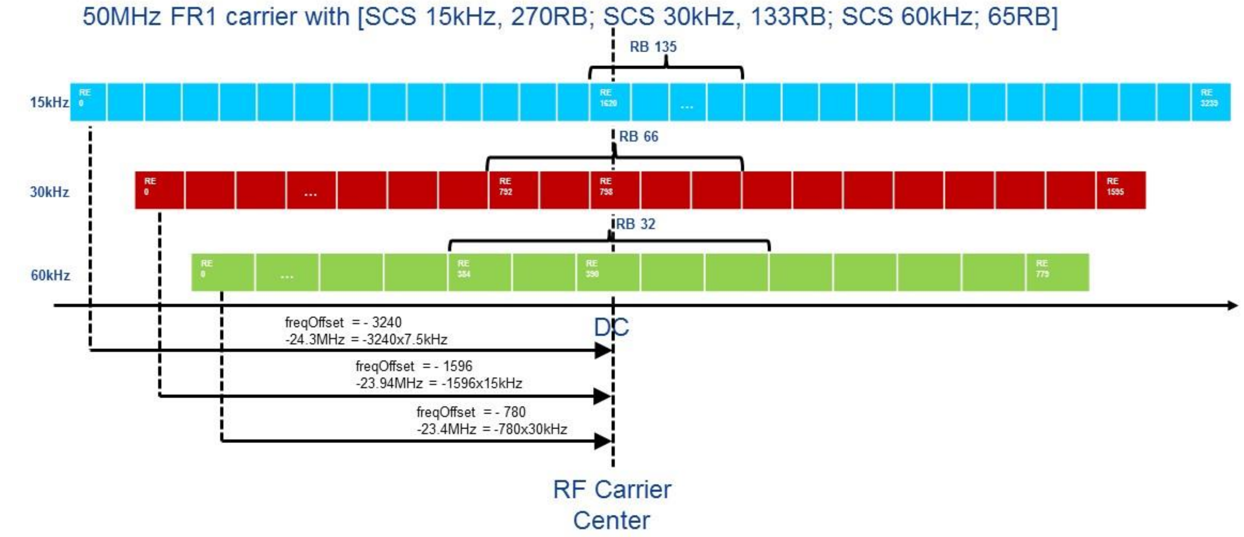


Figure 5-4 : RB Mapping and Support of Mixed Numerologies - Example with Mixed Numerologies

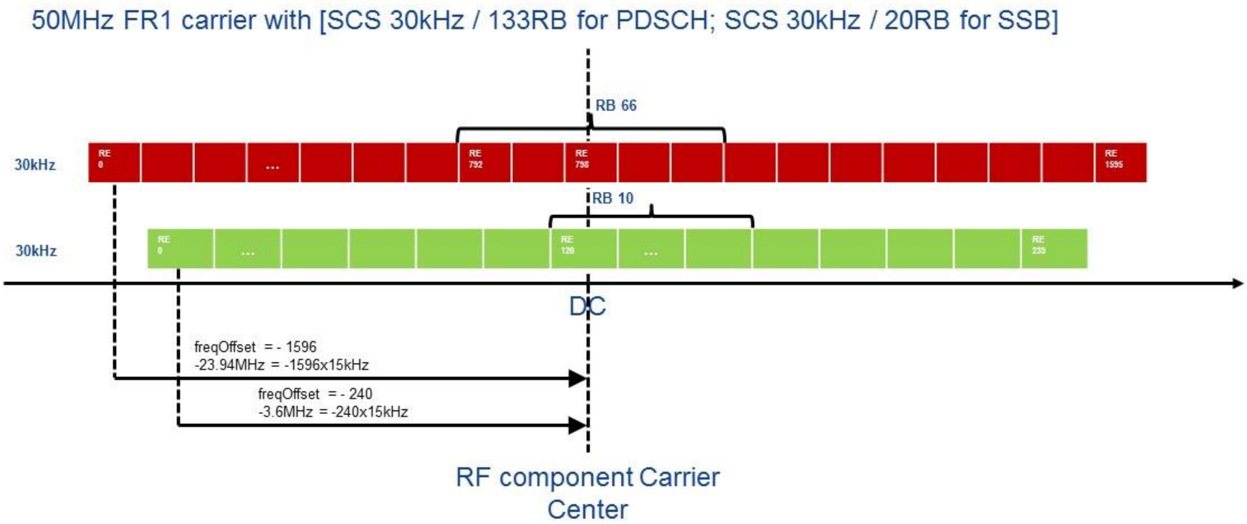


Figure 5-5 : RB Mapping and Support of Mixed Numerologies - Example SSB

<u>**Slot indexing with mixed numerologies**</u>

For a given frequency range FR1 or FR2, the highest possible numerology supported by the O-RU shall be used as the common reference per component carrier for the start of the slot identified by slotId. If the highest numerology supported by the O-RU allows both normal and extended CP then normal CP shall be used as reference. Note that the O-RU shall advance slots with extended CP against the reference. The symbol duration and position in time is calculated from the µ value (SCS from frameStruture in message field or configured via M-plane) and the slotId field in the CU-Plane message. The value of the sectionId field in CU-Plane messages addressed per eAxC shall be unique per slot identified by slotId value.

For example, in FR1 case, the maximum slot numbers per subframe is four, since the maximum SCS is 60kHz

- µ= 0 → slot #0; µ=1 → slot #0 & #2; µ=2 → slot #0, #1, #2, #3 (see **Table 5-1**)

Table 5-1 : cpLength Use

μ=2 60 kHz		μ=1 30 kHz		μ=0 15 kHz	
slotId	symbolId	slotId	symbolId	slotId	symbolId
0	0	0	0	0	0
	1				
	2		1		
	...				
	11		2		1
	12				
	13		...		
1	0				
	1		11		2
	2				
	...		12		
	11				
	12		13		...
	13				
2	0	2	0		
	1				
	2		1		11
	...				
	11		2		
	12				
	13		...		12
3	0				
	1		11		
	2				
	...		12		13
	11				
	12		13		
	13				
0	0	0	0	0	0
	1				
	2		1		
	...				
	11		2		1
	12				
	13				

...

μ=3 120 kHz		μ=2 60 kHz		μ=1 30 kHz	
slotId	symbolId	slotId	symbolId	slotId	symbolId
0	0 1 ... 12 13	0	0	0	0
			1		
			2		1
1	0 1 ... 12 13		...		
			11		2
			12		
			13		...
2	0 1 ... 12 13	2	0		
			1		11
			2		
3	0 1 ... 12 13		...		12
			11		
			12		13
			13		
4	0 1 ... 12 13	4	0	4	0
			1		
			2		1
5	0 1 ... 12 13		...		
			11		2
			12		
			13		...
6	0 1 ... 12 13	6	0		
			1		11
			2		
7	0 1 ... 12 13		...		12
			11		
			12		13
			13		
0	0 1 ... 12 13	0	0	0	0
			1		
			2		1
1	0 1 ... 12 13		...		
			11		2
			12		
			13		

...

PRACH formats with multiple repetitions of preamble

Certain PRACH formats lead to PRACH symbols to be constructed from multiple repetitions of a preamble sequence, with only the Cyclic Prefix (CP) used with the first sequence. Therefore, the O-RU must be informed how to correctly execute CP extraction and FFT. Based on implementation, this may e.g. be achieved by sending each repetition inside a PRACH occasion as a separate symbol (in a separate message and separate Section ID), or by sending a single control message spanning over multiple symbols (e.g. example depicted in Figure 5-6: number of symbols = 4, CP length = 0, time offset duration is adjusted by an equivalent time value of 1152 samples to compensate for setting CP length =0), which reduces the number of C-Plane messages and data sections required.

The example in **Figure 5-6** depicts the scenario with format A2 and 30kHz SCS, in which only CP extraction occurs once, yet there are four associated FFT operations.

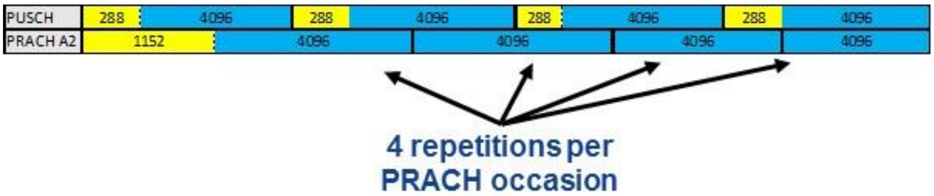


Figure 5-6 : Example of PRACH Format A2

5.3.3 DL Precoding configuration parameters and indications

This sub-section addresses the method of specifying the precoding operation in the O-RU, which is expected when Category B O-RUs are supported. For Category A O-RUs the precoding is implemented in the O-DU so the descriptions in this sub-section are not relevant. Annex I contains more information and examples regarding this precoding.

O-DU

12 REs are always sent on the interface from the O-DU to O-RU.

For 'single Tx':

- For layer mapping at O-RU, complex-valued modulation symbols mapped in a sequence starting with $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)}-1)$ to frequency REs (k,l) and are packed into a PRB. A single eAxC is used for this purpose.
- For CRS mapping at O-RU, all CRS REs belong to a single Tx antenna and are mapped to frequency REs (k,l) for one layer and are packed into a PRB for transmission and are unpacked at O-RU (see Annex I for details).

For TxD:

- For layer mapping at O-RU, complex-valued modulation symbols mapped in a sequence starting with $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)}-1)$ to frequency REs (k,l) and are packed into a PRB. A single eAxC is used for this purpose.
- For CRS mapping at O-RU, all CRS RE's are mapped to frequency REs (k,l) for all layers and are packed into a PRB for transmission and are unpacked at O-RU (see Annex I for details).

For TM3/TM4, TM5/6

- At the O-DU, layer mapped symbols for each layer v, mapped in sequence starting with $x(i)=\begin{bmatrix}x^{(0)}(i) & \ldots & x^{(\upsilon-1)}(i)\end{bmatrix}^T$ to frequency REs (k,l) are packed into each PRB (see Annex I).
- For precoding at the O-RU, different eAxCs are used for each layer (See Annex I).
- For CRS mapping at the O-RU, all CRS RE's for each layer are packed into a PRB for transmission and are unpacked at the O-RU (see Annex I for details).
- All C-plane message parameters are kept the same for precoding purposes.

For TM7-10 and NR

- Precoding in the O-RU may be implemented in various ways that are vendor-defined and vendor-specific, because there is no 3GPP-mandated precoding operation for these cases.

O-RU

From the C-plane precoding section extension, the O-RU determine the transmission scheme using txScheme field.

For 'single Tx':

- At the O-RU, input modulated symbols $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)}-1)$ are to be unpacked and used to perform layer mapping, precoding and antenna port mapping for single tx.

- For transmission on a single antenna port, a single layer is used, $\upsilon = 1$, and the mapping is defined as $x^{(0)}(i) = d^{(0)}(i)$ with $M_{\text{symb}}^{\text{layer}} = M_{\text{symb}}^{(0)}$.
- For single tx transmission on a single antenna port, precoding is defined by $y^{(p)}(i) = x^{(0)}(i)$ where $p \in \{0,4,5,7,8,11,13,107,108,109,110\}$ is the number of the single antenna port used for transmission of the physical channel and $i = 0,1,\ldots,M_{\text{symb}}^{\text{ap}} - 1$, $M_{\text{symb}}^{\text{ap}} = M_{\text{symb}}^{\text{layer}}$.
- For antenna port mapping p={0}, each y(i)=[$y^{(p)}$ (i) $]^T$ RE goes to antenna port y_p(i) after antenna port mapping.
- Since the PRB contains CRS sequences for one antenna port, the RE should extract the CRS RE's using crsSymbolNumber, crsReMask and crsShift (see Annex I for details) and are mapped to the appropriate RE position.

For txScheme 'TxD':

- At the O-RU, input modulated symbols $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)} - 1)$ are to be unpacked and used to perform layer mapping, precoding and antenna port mapping.
- The appropriate precoder is selected based on number of layers and antenna ports.
- For antenna port mapping p={0..N}, each y(i)=[$y^{(p)}$ (i) $]^T$ RE goes to each antenna port y_p(i) after antenna port mapping.
- Since the PRB contains CRS sequences for N antenna ports, the RE should extract the CRS RE's using crsSymbolNumber, crsReMask and crsShift (see Annex I for details) which are mapped to the appropriate RE position and rest of the REs are populated with zero data.

For TM3/TM4, TM5/6

- Input layer mapped symbols $x(i) = \left[x^{(0)}(i) \quad \ldots \quad x^{(\upsilon-1)}(i)\right]^T$ shall be used to perform precoding at the O-RU based on codeBookIndex, numLayers, layerID.
- In closed loop mode, the appropriate precoder is selected per codebook index, number of layers and antenna ports.
- In open loop mode, the codebook index field is ignored.
- The O-RU changes the precoder per RE automatically based on the number of antenna ports and number of layers.
- After precoding, for antenna port mapping p={0..N}, each y(i)=[$y^{(p)}$ (i) $]^T$ RE goes to each antenna port y_p(i) after antenna port mapping.
- Since all PRBs contain CRS sequences for N antenna ports, then based on the layerID (layer 0) extract CRS sequence using crsSymbolNumber, crsReMask and crsShift (see Annex I) for CRS mapping to each of the antenna ports using the reMask bit field; the CRS REs from other layers can be ignored.

For TM7-10 and NR

- One way to implement precoding in the O-RU is via the beamID values, wherein a beamID points to a beamforming vector that also implements the precoding operation.

5.3.4 LAA Commands Transfer procedure

This procedure is used to exchange C-Plane messages between O-DU and O-RU. The main purpose of these messages is to support LAA feature in the O-RU/O-DU. See Annex G for more details on the LAA message flow.

5.3.4.1 LBT procedure overview

The LBT procedure is used to configure the O-RU with the parameters needed to do LBT prior to PDSCH or DRS transmission OTA. The O-RU needs to report the LBT process outcome (either success or failure) in the indication message.

In contrast to licensed spectrum, where the O-RU may continuously send data or reference signals OTA, in unlicensed spectrum, the O-RU can only send discontinuous bursts of data (of length MCOT) or periodic DRS signals. The LBT_DL_CONFIG request message is sent once prior to each OTA transmission on the LAA SCell. This includes both DRS and PDSCH (i.e., MCOT burst) transmissions. For every LBT_DL_CONFIG request message, the O-RU replies with the LBT_DL indication messages which include the LBT outcome and the status of the subframe (transmitted/dropped) (see **Figure 5-7**).

If O-RU can manage a congestion window by itself, the O-DU does not need to send LBT_DL_CONFIG request message to start the LBT procedure at O-RU. Because O-RU can determine the start timing of LBT procedure, the requirements of O-DU for the LBT procedure at the O-RU is a delivery of needed information to adjust a congestion window managed by O-RU. To transmit these information, O-DU sends LBT_CWCONFIG_REQ message, and O-RU notifies to the O-DU by LBT_CWCONFIG_RSP message to indicate that the LBT_CWCONFIG_REQ message is successfully received or not. Also, O-RU can adjust its congestion window based on the information on the LBT_CWCONFIG_REQ message. For every LBT_CWCONFIG_REQ request message, the O-RU replies with the LBT_DL indication message(s) which include the LBT outcome and the status of the subframe (transmitted/dropped).

For the DRS signal, initially, the O-DU must configure the O-RU with the DRS parameters (DMTC period and DMTC offset) via the M-plane.

Before every DRS period, the O-DU must send the LBT_DL_CONFIG request message and the DRS signal. O-RU does LBT and transmits the DRS signal OTA in case of LBT success. The O-RU then must send the LBT_DL indication.

The DRS signal can be transmitted over the fronthaul interface similar to other reference signals such as the PSS/SSS/CRS.

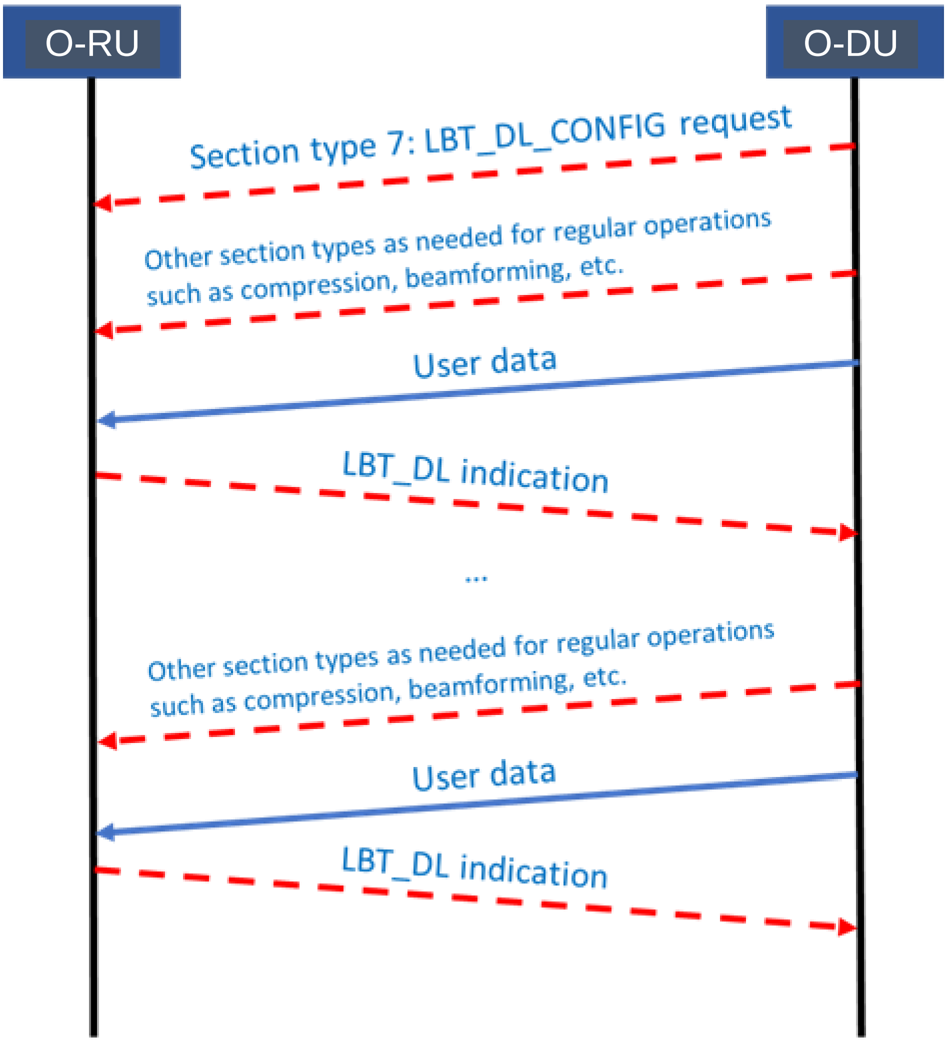


Figure 5-7 : LBT Message Flow

5.3.4.2 Definitions

- D_{OW}: Maximum one-way fronthaul latency
- P_{DU}: Maximum O-DU processing time (reading an upstream O-RU c-plane message, MAC processing, sending the downstream U-plane data)
- P_{RU}: Maximum O-RU processing time (sending an upstream c-plane message, receiving downstream U-plane data, transmitting U-plane data OTA)
- B_{RU}: O-RU Buffer depth (for LAA) ceiled to an integer number of OFDM symbols in microseconds (up to 140 symbols, for example (i.e., MCOT)). This is equivalent to the minimum amount of data that must be buffered at the O-RU. This parameter could be less than or equal to the actual O-RU buffer size communicated via the M-Plane.
- Expired symbol: Symbol where the current time at the O-RU exceeds its target transmission time (i.e., SFN/SF) OTA.
- Normal (partially-filled) SF assumption: T_{start} is the start of the transmission time, which occurs at the first SF (slot) boundary the O-RU encounters after starting the LBT CAT 4 process and after finishing the minimum sensing duration. Initially known to the O-DU (since it sends the LBT start time and the LBT parameters to the O-RU). It can be updated in real-time based on the LBT outcome and the communication between the O-DU and O-RU. Note that here, a “slot” follows the LTE definition of containing seven symbols.

- LBT_DL indication Threshold (LBT_{th}): The latest time the O-DU is expected to receive the LBT_DL indication message from the O-RU.

$$LBT_{th} = T_{start} + (P_{RU} + D_{OW} + P_{DU})$$

- CU Transmission time threshold (CUT_{th}): The time at which the O-DU is required to send the data to the O-RU to be transmitted OTA at time T_{start}

$$CUT_{th} = T_{start} - (P_{RU} + D_{OW} + P_{DU})$$

- x: Minimum time between any two potential start transmission times

$$x = \begin{cases} 1\,ms, & Normal\,SF \\ 0.5\,ms, & Partially - filled\,SF \end{cases}$$

5.3.4.3 General Guidelines for the LAA-procedure

5.3.4.3.1 PDSCH Transmission

- O-DU should avoid buffer overflow or underflow at the O-RU:
  - O-DU should only send B_{RU} worth of data to the O-RU for every single transmission
  - O-DU should plan for the data to be received at the O-RU only P_{RU} before the actual OTA transmission time at the O-RU
- O-RU should avoid buffer overflow or underflow at the O-RU:
  - O-RU should flush its buffer (by dropping any expired symbols) as soon as any symbol becomes expired (i.e., current time is larger than the symbol’s scheduled time)
  - O-RU should immediately send a success LBT indication to the O-DU once the channel is acquired.
  - The O-RU should send an error message to the O-DU if data received is larger than its local buffer
  - The O-RU should send an subframe drop or transmission message to the O-DU when buffered subframe is dropped because scheduled time is passed, or is transmitted after LBT success.
- The O-RU should have a buffer that satisfies the following equation: $B_{RU} \geq 2 \times D_{OW} + P_{DU} + P_{RU}$
- Note on LBT CAT 4 in general:
  - The O-DU can configure the O-RU (via the M-plane) with the threshold on the LBT CAT 4 duration (e.g., 8 ms). Once this threshold has exceeded, the O-RU sends a failure LBT indication to the O-DU,

which in return sends back a new LBT config request. The O-RU can then restart the LBT CAT 4 process.

- The data signal may be received at the O-RU before or after the LBT_PDSCH_REQ
- The O-RU should generate the reservation signal locally whenever needed

5.3.4.3.2 DRS Transmission

- Recall:
  1. PSS/SSS being part of a DRS may occur outside subframe 0 and 5
  2. CRS/CSI-RS/PSS/SSS do not vary with subframe number but are kept unchanged across subframes 0–4 (call it DRS_v1) and 5–9 (DRS_v2).
- Assumption: DRS OTA transmission starts at the SF boundaries.
- The DRS signal may be received before or after the LBT_DRS_REQ
- LBT_DRS_REQ is sent once per DRS window (e.g., DMTC offset = 6 ms is conveyed to the O-RU via the M-plane).
- DRS signal is sent every SF in the 6 SFs of the DRS window until LBT succeeds.
- $DRS_{th,1} = SF0 - (P_{RU} + D_{OW} + P_{DU})$
- $DRS_{th,2} = SF5 - (P_{RU} + D_{OW} + P_{DU})$
- $SF0$: The start of SF0, SF1, SF2, SF3, or SF4
- $SF5$: The start of SF5, SF6, SF7, SF8, or SF9
- At $DRS_{th,1}$, O-DU sends DRS_v1 (i.e., to be transmitted within SFs 0-4)
- At $DRS_{th,2}$, O-DU sends DRS_v2 (i.e., to be transmitted within SFs 5-9)
- Notes:
  1. LBT indication with failure outcome must be sent for every sensing period.

5.3.4.3.3 Congestion Window Information Transmission

- O-DU should send information regarding a congestion window adjustment to O-RU
  - HARQ feedback information for the reference subframe and number of TB are included
- O-RU should adjust its managed congestion window value and notify to O-DU the packet reception status
  - O-RU should adjust its congestion window value based on received information
  - O-RU should immediately send a success LBT_CWCONFIG_RSP to the O-DU when the message is successfully received. If not, O-RU should send a fail LBT_CWCONFIG_RSP.

5.3.5 Symbol Numbering and Duration

For the NR case, data symbol numbering in a slot shall be from zero to thirteen (for extended CP, eleven). For LTE the symbol numbering shall be mapped to the NR numbering as follows:

LTE with normal prefix maps to NR with same µ and normal prefix:

LTE symbol l_{LTE} of slot $2i$ is mapped to NR slot i, symbol $l' = l_{LTE}$
LTE symbol l_{LTE} of slot $2i + 1$ is mapped to NR slot i, symbol $l' = 7 + l_{LTE}$

LTE with extended prefix the following applies:

LTE symbol l_{LTE} of slot $2i$ is mapped to NR slot i, symbol $l' = l_{LTE}$
LTE symbol l_{LTE} of slot $2i + 1$ is mapped to NR slot i, symbol $l' = 6 + l_{LTE}$

The symbol duration can be determined from the SCS provided by “frameStructure” and knowledge of whether LTE versus NR is supported; 3GPP specifications 36.211 and 38.211 provide the necessary formulas.

5.4 Elements for the C-plane Protocol

5.4.1 General

C-Plane messages are encapsulated using a two-layered header approach. The first layer consists of a eCPRI common header or IEEE 1914.3 common header, including corresponding fields used to indicate the message type, while the second layer is an application layer including necessary fields for control and synchronization. Within the application layer, a "section" defines the characteristics of U-plane data to be transferred or received from a beam with one pattern id. In general, the transmission header, application header, and sections are all intended to be aligned on 4-byte boundaries and are transmitted in "network byte order" meaning the most significant byte of a multi-byte parameter is transmitted first.

Table 5-2 described the section types are supported within the C-Plane.

Table 5-2 : Section Types

Section Type	Target Scenario	Remarks
0	Unused Resource Blocks or symbols in Downlink or Uplink	Indicates to O-RU that certain Resource Blocks or symbols will not be used (idle periods, guard periods). Likewise, there are no associated U-Plane messages containing IQ data for this Section Type. The purpose is to inform the O-RU that transmissions may be halted during the specified idle interval for e.g. power-savings or to provide an interval for calibration.
1	Most DL/UL radio channels	Here "most" refers to channels not requiring time or frequency offsets such as are needed for mixed-numerology channels
2	reserved for future use	
3	PRACH and mixed-numerology channels	Channels requiring time or frequency offsets or different-than-nominal SCS values
4	Reserved for future use	
5	UE scheduling information (UE-ID assignment to section)	Provides scheduling information for UE-IDs
6	Channel information	Sends UE-specific channel information from the O-DU to the O-RU
7	LAA	Messages communicated between O-DU and the O-RU in both directions to configure LBT for PDSCH/DRS transmission and to report the LBT outcome.
8-255	Reserved for future use	

5.4.1.1 Section Extensions

Within a data section the section header in most cases includes a "extension flag" which indicates parameters that apply to the section beyond those within the section header (the extension flag is most often adjacent to the beamId or ueId fields). The presence of this extension flag indicates that following the header a section extension is present. Within the section extension is another extension flag, adjacent to the "extType" field, which indicates that a second (or third, etc.) extension is present. In this way any number of section extensions may be included within a data section. This provides an extensibility for section parameters without the need to continually redefine the section header or create new section types to accommodate future fronthaul specification needs.

The section extension takes the form reminiscent of "TLV": there is a type field specifically "extType" which is a 7-bit field describing the specific extension type, then a one-byte length field specifically "extLen" field detailing how many 4-byte words are contained in the extension (minimum of 1 including the "extType" and "extLen" fields), and then some number of parameters that depend on the "extType" value for their definition. In all cases the section extension will be an integer number of (4-byte) words in length.

Specific valid values of "extType", their meanings and their associated parameters are detailed in subsequent sub-chapters of this spec.

5.4.2 Scheduling and Beamforming Commands

A common frame format is used for C-Plane messages, consisting of a transport layer and an application layer. The application layer is within the transport layer payload and consists of a common header for time reference, followed by information and parameters dependent and specific to the Section Type in use. Multiple sets of section data of the same Section Type value can be lined up one after another within the payload. To minimize packet rate over the interface, transmitter should fill messages with as many subsequent sections (with or without sequential section IDs) as possible. However, sets of section data of different Section Type values are to be sent via separate messages (i.e. different values of Section Type shall not be mixed within a single C-Plane message payload).

- Transport Layer (see section 3.1.3)
- Application Layer
  - Section Type "0" Fields (used for indicating idle or guard periods from O-DU to O-RU)
    - Common Header Fields
      - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit
      - **payloadVersion** (payload version) field: 3 bits
        - value = "1" shall be set (1st protocol version for payload and time reference format)
      - **filterIndex** (filter index) field: 4 bits
      - **frameId** (frame identifier) field: 8 bits
      - **subframeId** (subframe identifier) field: 4 bits
      - **slotID** (slot identifier) field: 6 bits
      - **startSymbolid** (start symbol id) field: 6 bits
      - **numberOfsections** (number of sections) field: 8 bits
      - **sectionType** (section type) field: 8 bits
        - value = "0" shall be set
      - **timeOffset** (time offset) field: 16 bits
      - **frameStructure** (frame structure) field: 8 bits
      - **cpLength** (cyclic prefix length) field: 16 bits
      - **reserved** (reserved for future use) field: 8 bits
    - Section Fields
      - **sectionID** (section identifier) field: 12 bits
      - **rb** (resource block indicator) field: 1 bit
      - **symInc** (symbol number increment command) field: 1 bit
      - **startPrbc** (starting PRB of data section description) field: 10 bits
      - **numPrbc** (number of contiguous PRBs per data section description) field: 8 bits
      - **reMask** (resource element mask) field: 12 bits
      - **numSymbol** (number of symbols) field: 4 bits
      - **reserved** (reserved for future use) field: 16 bits
  - Section Type "1" Fields (used for most Downlink and Uplink radio channels – some channels especially PRACH and mixed-numerology channels need more information elements contained in other section types)
    - Common Header Fields
      - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit

- **payloadVersion** (payload version) field: 3 bits
  - value = “1” shall be set (1st protocol version for payload and time reference format)
- **filterIndex** (filter index) field: 4 bits
- **frameId** (frame identifier) field: 8 bits
- **subframeId** (subframe identifier) field: 4 bits
- **slotID** (slot identifier) field: 6 bits
- **startSymbolid** (start symbol id) field: 6 bits
- **numberOfsections** (number of sections) field: 8 bits
- **sectionType** (section type) field: 8 bits
  - value = “1” shall be set
- **udCompHdr** (user data compression header) field: 8 bits
- **reserved** (reserved for future use) field: 8 bits

- Section Fields
  - **sectionId** (section identifier) field: 12 bits
  - **rb** (resource block identifier) field: 1 bit
  - **symInc** (symbol number increment command) field: 1 bit
  - **startPrbc** (starting PRB of data section description) field: 10 bits
  - **numPrbc** (number of contiguous PRBs per data section description) field: 8 bits
  - **reMask** (resource element mask) field: 12 bits
  - **numSymbol** (number of symbols) field: 4 bits
  - **ef** (extension flag) field: 1 bit
  - **beamId** (beam identifier) field: 15 bits

- Section Type “3” Fields (used for PRACH and mixed-numerology channels):
  - Common Header Fields
    - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit
    - **payloadVersion** (payload version) field: 3 bits
      - value = “1” shall be set (1st protocol version for payload and time reference format)
    - **filterIndex** (filter index) field: 4 bits
    - **frameId** (frame identifier) field: 8 bits
    - **subframeId** (subframe identifier) field: 4 bits
    - **slotID** (slot identifier) field: 6 bits
    - **startSymbolid** (start symbol identifier) field: 6 bits
    - **numberOfsections** (number of sections) field: 8 bits
    - **sectionType** (section type) field: 8 bits
      - value = “3” shall be set
    - **timeOffset** (time offset) field: 16 bits
    - **frameStructure** (frame structure) field: 8 bits
    - **cpLength** (cyclic prefix length) field: 16 bits
    - **udCompHdr** (user data compression header) field: 8 bits
  - Section Fields

- **sectionID** (section identifier) field: 12 bits
- **rb** (resource block identifier) field: 1 bit
- **symInc** (symbol number increment command) field: 1 bit
- **startPrbc** (starting PRB of data section description) field: 10 bits
- **numPrbc** (number of contiguous PRBs per data section description) field: 8 bits
- **reMask** (resource element mask) field: 12 bits
- **numSymbol** (number of symbols) field: 4 bits
- **ef** (extension flag) field: 1 bit
- **beamId** (beam identifier) field: 15 bits
- **freqOffset** (frequency offset) field: 24 bits
- **reserved** (reserved for future use) field: 8 bits

- Section Type "5" Fields (used for UE scheduling information):
  - Common Header Fields
    - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit
    - **payloadVersion** (payload version) field: 3 bits
      - value = "1" shall be set (1st protocol version for payload and time reference format)
    - **filterIndex** (filter index) field: 4 bits
    - **frameId** (frame identifier) field: 8 bits
    - **subframeId** (subframe identifier) field: 4 bits
    - **slotID** (slot identifier) field: 6 bits
    - **startSymbolid** (start symbol identifier) field: 6 bits
    - **numberOfsections** (number of sections) field: 8 bits
    - **sectionType** (section type) field: 8 bits
      - value = "5" shall be set
    - **udCompHdr** (user data compression header) field: 8 bits
    - **reserved** (reserved for future use) field: 8 bits
  - Section Fields
    - **sectionID** (section identifier) field: 12 bits
    - **rb** (resource block identifier) field: 1 bit
    - **symInc** (symbol number increment command) field: 1 bit
    - **startPrbc** (starting PRB of data section description) field: 10 bits
    - **numPrbc** (number of contiguous PRBs per data section description) field: 8 bits
    - **reMask** (resource element mask) field: 12 bits
    - **numSymbol** (number of symbols) field: 4 bits
    - **ef** (extension flag) field: 1 bit
    - **ueId** (UE identifier) field: 15 bits

- Section Type "6" Fields (used for sending channel information for a specific UE ID):
  - Common Header Fields
    - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit

- **payloadVersion** (payload version) field: 3 bits
  - value = "1" shall be set (1st protocol version for payload and time reference format)
- **filterIndex** (filter index) field: 4 bits
- **frameId** (frame identifier) field: 8 bits
- **subframeId** (subframe identifier) field: 4 bits
- **slotID** (slot identifier) field: 6 bits
- **startSymbolid** (start symbol identifier) field: 6 bits
- **numberOfsections** (number of sections) field: 8 bits
- **sectionType** (section type) field: 8 bits
  - value = "6" shall be set
- **numberOfUEs** (number of UE-specific channel information data sets) field: 8 bits
- **reserved** (reserved for future use) field: 8 bits

- Section Fields
  - **ef** (extension flag) field: 1 bit
  - **ueId** (UE identifier) field: 15 bits
  - **regularizationFactor** (regularization factor used for MMSE reception) field: 16 bits
  - **reserved** (reserved for future use) field: 4 bits
  - **rb** (resource block identifier) field: 1 bit
  - **symInc** (symbol number increment command) field: 1 bit
  - **startPrbc** (starting PRB of data section description) field: 10 bits
  - **numPrbc** (number of contiguous PRBs per data section description) field: 8 bits
  - **ciIsample** (channel information value, in-phase sample) field: 16 bits
  - **ciQsample** (channel information value, quadrature sample) field: 16 bits

- Section Type "7" Fields (used to support LAA):
  - Common Header Fields
    - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit
    - **payloadVersion** (payload version) field: 3 bits
      - value = "1" shall be set (1st protocol version for payload and time reference format)
    - **filterIndex** (filter index) field: 4 bits
    - **frameId** (frame identifier) field: 8 bits
    - **subframeId** (subframe identifier) field: 4 bits
    - **slotID** (slot identifier) field: 6 bits
    - **startSymbolid** (start symbol identifier) field: 6 bits
    - **numberOfsections** (number of sections) field: 8 bits
    - **sectionType** (section type) field: 8 bits
      - value = "7" shall be set
    - **reserved** (reserved for future use) field: 8 bits
    - Section Fields
    - **laaMsgType** (LAA message type) field: 4 bits
    - **laaMsgLen** field: 4 bits

- laaMsgType = "0" shall be set for LBT_DL_CONFIG.request: LBT_PDSCH_REQ
  - **lbtHandle** (An opaque handling returned in LBT_PDSCH_RSP) field: 16 bits
  - **lbtDeferFactor** (Defer factor in sensing slots as described in 3GPP TS 36.213 Section 15.1.1) field: 3 bits
  - **lbtBackoffCounter** (LBT backoff counter in sensing slots as described in 3GPP TS 36.213 Section 15.1.1) field: 10 bits
  - **lbtOffset** (LBT start time in microseconds from the beginning of the subframe scheduled by this message) field: 10 bits
  - **MCOT** (LTE TXOP duration in subframes) field: 4 bits
  - **lbtMode** (LBT process type) field: 2 bits:

- laaMsgType = "1" shall be set for LBT_DL_CONFIG.request: LBT_DRS_REQ
  - **lbtHandle** (An opaque handling returned in LBT_DRS_RSP) field: 16 bits
  - **lbtOffset** (LBT start time in microseconds from the beginning of the subframe scheduled by this message) field: 10 bits
  - **lbtMode** (LBT process type) field: 2 bits:
  - **reserved** (reserved for future use) field: 1 bit

- laaMsgType = "2" shall be set for LBT_DL.indication: LBT_PDSCH_RSP
  - **lbtHandle** (An opaque handling returned in LBT_PDSCH_RSP) field: 16 bits
  - **lbtPdschRes** (LBT result of SFN/SF) field: 2 bits
  - **sfStatus** (subfrme status) field: 1 bit
  - **initialPartialSF** (Indicates whether the initial SF in the LBT process is full or partial) field: 1 bit
  - **sfnSf** (SFN/SF of subframe which is dropped or successfully transmitted at O-RU) field: 12 bits
  - **reserved** (reserved for future use) field: 4 bits

- laaMsgType = "3" shall be set for LBT_DL.indication: LBT_DRS_RSP
  - **lbtHandle** (An opaque handling returned in LBT_DRS_RSP) field: 16 bits
  - **lbtDrsRes** (LBT result of SFN/SF) field: 1 bit
  - **reserved** (reserved for future use) field: 15 bits

- laaMsgType = "4" shall be set for LBT_buffer error: LBT_Buffer_Error
  - **lbtHandle** (An opaque handling returned in LBT_DRS_RSP) field: 16 bits
  - **lbtBufErr** (LBT buffer error type) field: 1 bit

- laaMsgType = "5" shall be set for LBT_DL_CONFIG.request: LBT_CWCONFIG_REQ
  - **lbtHandle** (An opaque handle returned in LBT_CWCONFIG_RSP) field: 16 bits
  - **lbtCWConfig_H** (HARQ feedback information regarding number of NACK about reference subframe as described in 3GPP TS 36.213 Section 15.1.3) field: 8 bits
  - **lbtCWConfig_T** (number of TB to manage Congestion Window as described in 3GPP TS 36.213 Section 15.1.3) field: 8 bits
  - **lbtMode** (LBT process type) field: 2 bits

- **lbtTrafficClass** (LBT traffic class) field: 3 bits:

- laaMsgType = “6” shall be set for LBT_DL.indication: LBT_CWCONFIG_RSP
  - **lbtHandle** (An opaque handling returned in LBT_CWCONFIG_RSP) field: 16 bits
  - **lbtCWR_Rst** (Notification LBT_CWCONFIG_REQ message successful or not) field: 1 bit

Table 5-3 : Scheduling and beamforming commands frame format (Section Type “0”)

<table>
<tr><th colspan="10">Section Type 0 : idle / guard periods</th></tr>
<tr><th>0 (msb)</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7 (lsb)</th><th># of bytes</th><th></th></tr>
<tr><td colspan="8">transport header, see section 3.1.3</td><td>8</td><td>Octet 1</td></tr>
<tr><td>dataDirection</td><td colspan="3">payloadVersion</td><td colspan="4">filterIndex</td><td>1</td><td>Octet 9</td></tr>
<tr><td colspan="8">frameId</td><td>1</td><td>Octet 10</td></tr>
<tr><td colspan="4">subframeId</td><td colspan="4">slotId</td><td>1</td><td>Octet 11</td></tr>
<tr><td colspan="2">slotId</td><td colspan="6">startSymbolid</td><td>1</td><td>Octet 12</td></tr>
<tr><td colspan="8">numberOfsections</td><td>1</td><td>Octet 13</td></tr>
<tr><td colspan="8">sectionType = 0</td><td>1</td><td>Octet 14</td></tr>
<tr><td colspan="8">timeOffset</td><td>2</td><td>Octet 15</td></tr>
<tr><td colspan="8">frameStructure</td><td>1</td><td>Octet 17</td></tr>
<tr><td colspan="8">cpLength</td><td>2</td><td>Octet 18</td></tr>
<tr><td colspan="8">Reserved</td><td>1</td><td>Octet 20</td></tr>
<tr><td colspan="8">sectionId</td><td>1</td><td>Octet 21</td></tr>
<tr><td colspan="4">sectionId</td><td>rb</td><td>symInc</td><td colspan="2">startPrbc</td><td>1</td><td>Octet 22</td></tr>
<tr><td colspan="8">startPrbc</td><td>1</td><td>Octet 23</td></tr>
<tr><td colspan="8">numPrbc</td><td>1</td><td>Octet 24</td></tr>
<tr><td colspan="8">reMask[11:4]</td><td>1</td><td>Octet 25</td></tr>
<tr><td colspan="4">reMask[3:0]</td><td colspan="4">numSymbol</td><td>1</td><td>Octet 26</td></tr>
<tr><td colspan="8">reserved (16-bits)</td><td>2</td><td>Octet 27</td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td colspan="8">sectionId</td><td>1</td><td>Octet N</td></tr>
<tr><td colspan="4">sectionId</td><td>rb</td><td>symInc</td><td colspan="2">startPrbc</td><td>1</td><td>N+1</td></tr>
<tr><td colspan="8">startPrbc</td><td>1</td><td>N+2</td></tr>
<tr><td colspan="8">numPrbc</td><td>1</td><td>N+3</td></tr>
<tr><td colspan="8">reMask[11:4]</td><td>1</td><td>N+4</td></tr>
<tr><td colspan="4">reMask[3:0]</td><td colspan="4">numSymbol</td><td>1</td><td>N+5</td></tr>
<tr><td colspan="8">reserved (16-bits)</td><td>2</td><td>N+6</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>Octet M</td></tr>
</table>

shading: yellow is transport header, pink is radio application header, others are repeated sections

Table 5-4 : Scheduling and beamforming commands frame format (Section Type “1”)

<table>
<tr><th colspan="10">Section Type 1 : DL/UL control msgs</th></tr>
<tr><th>0 (msb)</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7 (lsb)</th><th># of bytes</th><th></th></tr>
<tr><td colspan="8">transport header, see section 3.1.3</td><td>8</td><td>Octet 1</td></tr>
<tr><td>dataDirection</td><td colspan="3">payloadVersion</td><td colspan="4">filterIndex</td><td>1</td><td>Octet 9</td></tr>
<tr><td colspan="8">frameId</td><td>1</td><td>Octet 10</td></tr>
<tr><td colspan="4">subframeId</td><td colspan="4">slotId</td><td>1</td><td>Octet 11</td></tr>
<tr><td colspan="2">slotId</td><td colspan="6">startSymbolid</td><td>1</td><td>Octet 12</td></tr>
<tr><td colspan="8">numberOfsections</td><td>1</td><td>Octet 13</td></tr>
</table>

sectionType = 1								1	Octet 14
udCompHdr								1	Octet 15
reserved								1	Octet 16
sectionId								1	Octet 17
sectionId				rb	symInc	startPrbc		1	Octet 18
startPrbc								1	Octet 19
numPrbc								1	Octet 20
reMask[11:4]								1	Octet 21
reMask[3:0]				numSymbol				1	Octet 22
ef = 1	beamId[14:8]							1	Octet 23
beamId[7:0]								1	Octet 24
section extensions as indicated by “ef”								var	Octet 25
…									
sectionId								1	Octet N
sectionId				rb	symInc	startPrbc		1	N+1
startPrbc								1	N+2
numPrbc								1	N+3
reMask[11:4]								1	N+4
reMask[3:0]				numSymbol				1	N+5
ef = 0	beamId[14:8]							1	N+6
beamId[7:0]								1	N+7
section extensions as indicated by “ef”								var	N+8
									Octet M

shading: yellow is transport header, pink is radio application header, others are repeated sections

Table 5-5 : Scheduling and beamforming commands frame format (Section Type “3”)

Section Type 3 : PRACH & mixed-numerology									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
transport header, see section 3.1.3								8	Octet 1
dataDirection	payloadVersion			filterIndex				1	Octet 9
frameId								1	Octet 10
subframeId				slotId				1	Octet 11
slotId		startSymbolid						1	Octet 12
numberOfsections								1	Octet 13
sectionType = 3								1	Octet 14
timeOffset								2	Octet 15
frameStructure								1	Octet 17
cpLength								2	Octet 18
udCompHdr								1	Octet 20
sectionId								1	Octet 17
sectionId				rb	symInc	startPrbc		1	Octet 18
startPrbc								1	Octet 19
numPrbc								1	Octet 20
reMask[11:4]								1	Octet 21
reMask[3:0]				numSymbol				1	Octet 22
ef	beamId[14:8]							1	Octet 23
beamId[7:0]								1	Octet 24
frequencyOffset								3	Octet 25
reserved (8 bits)								1	Octet 28
section extensions as indicated by “ef”								var	Octet 29
…									

sectionId								1	Octet N
sectionId				rb	symInc	startPrbc		1	N+1
startPrbc								1	N+2
numPrbc								1	N+3
reMask[11:4]								1	N+4
reMask[3:0]				numSymbol				1	N+5
ef	beamId[14:8]							1	N+6
beamId[7:0]								1	N+7
frequencyOffset								3	N+8
reserved (8 bits)								1	N+11
section extensions as indicated by “ef”								var	N+12
									Octet M

shading: yellow is transport header, pink is radio application header, others are repeated sections

Table 5-6 : UE scheduling information frame format (Section Type “5”)

Section Type 5 : UE scheduling information conveyance									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
transport header, see section 3.1.3								8	Octet 1
dataDirection	payloadVersion			filterIndex				1	Octet 9
frameId								1	Octet 10
subframeId				slotId				1	Octet 11
slotId		startSymbolid						1	Octet 12
numberOfsections								1	Octet 13
sectionType = 5								1	Octet 14
udCompHdr								1	Octet 15
reserved								1	Octet 16
sectionId								1	Octet 17
sectionId				rb	symInc	startPrbc		1	Octet 18
startPrbc								1	Octet 19
numPrbc								1	Octet 20
reMask[11:4]								1	Octet 21
reMask[3:0]				numSymbol				1	Octet 22
ef	ueId[14:8]							1	Octet 23
ueId[7:0]								1	Octet 24
section extensions as indicated by “ef”								var	Octet 25
…									
sectionId								1	Octet N
sectionId				rb	symInc	startPrbc		1	N+1
startPrbc								1	N+2
numPrbc								1	N+3
reMask[11:4]								1	N+4
reMask[3:0]				numSymbol				1	N+5
ef	ueId[14:8]							1	N+6
ueId[7:0]								1	N+7
section extensions as indicated by “ef”								var	N+8
									Octet M

shading: yellow is transport header, pink is radio application header, others are repeated sections

Table 5-7 : UE channel information frame format (Section Type "6")

<table>
<tr><th colspan="10">Section Type 6 : channel information conveyance</th></tr>
<tr><th>0 (msb)</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7 (lsb)</th><th># of bytes</th><th></th></tr>
<tr><td colspan="8">transport header, see section 3.1.3</td><td>8</td><td>Octet 1</td></tr>
<tr><td>dataDirection</td><td colspan="3">payloadVersion</td><td colspan="4">filterIndex</td><td>1</td><td>Octet 9</td></tr>
<tr><td colspan="8">frameId</td><td>1</td><td>Octet 10</td></tr>
<tr><td colspan="4">subframeId</td><td colspan="4">slotId</td><td>1</td><td>Octet 11</td></tr>
<tr><td colspan="2">slotId</td><td colspan="6">startSymbolid</td><td>1</td><td>Octet 12</td></tr>
<tr><td colspan="8">numberOfsections</td><td>1</td><td>Octet 13</td></tr>
<tr><td colspan="8">sectionType = 6</td><td>1</td><td>Octet 14</td></tr>
<tr><td colspan="8">numberOfUEs</td><td>1</td><td>Octet 15</td></tr>
<tr><td colspan="8">reserved</td><td>1</td><td>Octet 16</td></tr>
<tr><td>ef</td><td colspan="7">ueId[14:8]</td><td>1</td><td>Octet 17</td></tr>
<tr><td colspan="8">ueId[7:0]</td><td>1</td><td>Octet 18</td></tr>
<tr><td colspan="8">regularizationFactor</td><td>2</td><td>Octet 19</td></tr>
<tr><td colspan="4">reserved</td><td>rb</td><td>symInc</td><td colspan="2">startPrbc</td><td>1</td><td>Octet 21</td></tr>
<tr><td colspan="8">startPrbc</td><td>1</td><td>Octet 22</td></tr>
<tr><td colspan="8">numPrbc</td><td>1</td><td>Octet 23</td></tr>
<tr><td colspan="8">ciIsample (first PRB, first antenna)</td><td>var</td><td>Octet 24</td></tr>
<tr><td colspan="8">ciQsample (first PRB, first antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciIsample (first PRB, second antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (first PRB, second antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td colspan="8">ciIsample (first PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (first PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td colspan="8">ciIsample (last PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (last PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">section extensions as indicated by "ef"</td><td>var</td><td></td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td>ef</td><td colspan="7">ueId[14:8]</td><td>1</td><td>Octet N</td></tr>
<tr><td colspan="8">ueId[7:0]</td><td>1</td><td>N+1</td></tr>
<tr><td colspan="8">regularizationFactor</td><td>2</td><td>N+2</td></tr>
<tr><td colspan="4">Reserved</td><td>rb</td><td>symInc</td><td colspan="2">startPrbc</td><td>1</td><td>N+4</td></tr>
<tr><td colspan="8">startPrbc</td><td>1</td><td>N+5</td></tr>
<tr><td colspan="8">numPrbc</td><td>1</td><td>N+6</td></tr>
<tr><td colspan="8">ciIsample (first PRB, first antenna)</td><td>var</td><td>N+7</td></tr>
<tr><td colspan="8">ciQsample (first PRB, first antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciIsample (first PRB, second antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (first PRB, second antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td colspan="8">ciIsample (first PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (first PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">…</td><td></td><td></td></tr>
<tr><td colspan="8">ciIsample (last PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">ciQsample (last PRB, last antenna)</td><td>var</td><td></td></tr>
<tr><td colspan="8">section extensions as indicated by "ef"</td><td>var</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>Octet M</td></tr>
</table>

shading: yellow is transport header, pink is radio application header, others are repeated sections

Table 5-8 : LAA Message, O-DU to O-RU -OR- O-RU to O-DU (Section Type "7")

Section Type 7 : LAA Message,O-DU to O-RU or O-RU toO-DU									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
transport header, see section 3.1.3								8	Octet 1
reserved	payloadVersion			reserved				1	Octet 9
frameId								1	Octet 10
subframeId				slotId				1	Octet 11
slotId		reserved						1	Octet 12
reserved								1	Octet 13
sectionType = 7								1	Octet 14
reserved								1	Octet 15
reserved								1	Octet 16
laaMsgType				laaMsgLen				1	Octet 17
Payload (see below) plus padding to 32-bit boundary								3 or 7	Octet 18+

Note that here, frameId, subframeId and slotId serve to provide a time stamp on the LAA LBT C-Plane message and do not describe any U-Plane data associated with the Section Type 7 C-Plane message.

Section Type 7 : LAA Request Message, O-DU to O-RU **LBT_PDSCH_REQ**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0000b (LBT_PDSCH_REQ)				laaMsgLen = 2 (2 words)				1	Octet 17
lbtHandle								2	Octet 18
lbtOffset[9:2]								1	Octet 20
lbtOffset[1:0]		lbtMode		reserved	lbtDeferFactor			1	Octet 21
lbtBckoffCounter[9:2]								1	Octet 22
lbtBckoffCounter[1:0]		MCOT				reserved		1	Octet 23
reserved								1	Octet 24

Section Type 7 : LAA Request Message, O-DU to O-RU **LBT_DRS_REQ**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0001b (LBT_DRS_REQ)				laaMsgLen = 2 (2 words)				1	Octet 17
lbtHandle								2	Octet 18
lbtOffset[9:2]								1	Octet 20
lbtOffset[1:0]		lbtMode		reserved				1	Octet 21
reserved								1	Octet 22
reserved								1	Octet 23
reserved								1	Octet 24

Section Type 7 : LAA Response Message, O-RU to O-DU **LBT_PDSCH_RSP**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0010b (LBT_PDSCH_RSP)				laaMsgLen = 2 (2 words)				1	Octet 17
lbtHandle								2	Octet 18
lbtPdschRes		inParSF	sfStatus	sfnSf[11:8]				1	Octet 20
sfnSf[7:0]								1	Octet 21
reserved								1	Octet 22
reserved								1	Octet 23
reserved								1	Octet 24

Section Type 7 : LAA Response Message, O-RU to O-DU **LBT_DRS_RSP**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0011b (LBT_DRS_RSP)				laaMsgLen = 1 (1 word)				1	Octet N
lbtHandle								2	N+1
lbtDrsRes	reserved							1	N+3

Section Type 7 : LAA Response Message, O-RU to O-DU **LBT_Buffer_Error**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0100b (LBT_Buffer_Error)				laaMsgLen = 1 (1 word)				1	Octet N
lbtHandle								2	N+1
lbtBufErr	reserved							1	N+3

Section Type 7 : LAA Request Message, O-DU to O-RU **LBT_CWCONFIG_REQ**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0101b (LBT_CWCONFIG_REQ)				laaMsgLen = 2 (2 words)				1	Octet N
lbtHandle								2	N+1
lbtCWConfig_H								1	N+3
lbtCWConfig_T								1	N+4
lbtMode		lbtTrafficClass			reserved			1	N+5
reserved								2	N+6

Section Type 7 : LAA Request Message, O-RU to O-DU **LBT_CWCONFIG_RSP**									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
laaMsgType = 0110b (LBT_CWCONFIG_RSP)				laaMsgLen = 1 (1 word)				1	Octet N
lbtHandle								2	N+1
lbtCWR_Rst	reserved							1	N+2

5.4.3 Coding of Information Elements – Transport Layer

See Section 3.1.3 for transport header information element details.

5.4.4 Coding of Information Elements – Application Layer, Common

5.4.4.1 dataDirection (data direction (gNB Tx/Rx))

Description: This parameter indicates the gNB data direction.

Value range: {0b=Rx (i.e. UL), 1b=Tx (i.e. DL)}.

Type: binary bit.

Field length: 1 bit.

5.4.4.2 payloadVersion (payload version)

Description: This parameter defines the payload protocol version valid for the following IEs in the application layer. In this version of the specification payloadVersion=001b shall be used.

Value range: {000b-111b=Payload version}.

Type: unsigned integer.

Field length: 3 bits.

Default Value: 001b (version 1 assumed).

5.4.4.3 filterIndex (filter index)

Description: This parameter defines an index to the channel filter to be used between IQ data and air interface, both in DL and UL. For most physical channels filterIndex =0000b is used which indexes the standard channel filter, e.g. 100MHz channel filter for 100MHz nominal carrier bandwidth. Another use case is PRACH in UL, where different filter indices can be used for different PRACH formats, assuming that before FFT processing of PRACH data there is a separate PRACH filter or PRACH filter in addition to the standard channel filter in UL. Please note that for PRACH there is typically also a frequency offset (see freqOffset) applied before the PRACH filter.

NOTE: Filter index is commanded from O-DU to O-RU. Likewise, it is not mandatory to command special filters, and filter index = 0000b is also allowed for PRACH.

NOTE: When using filter indices corresponding to PRACH, the first RE of the first PRB addressed inside the section Id shall correspond with the first guard tone used at the lower edge of the PRACH frequency block as specified in 3GPP RAN1 TS 3x.211. In case of fragmentation of 1 data section over multiple U-Plane messages, this is applicable to the first fragmented section.

Value range: {0000b-1111b}.

Table 5-9 : Filter Index

Value of IE "filter Index"	Usage	PRACH preamble formats	Minimum filter pass band
0000b	standard channel filter	N/A	
0001b	UL filter for PRACH preamble formats	0, 1, 2	839 x 1.25kHz = 1048.75 kHz
0010b		3	839 x 5 kHz = 4195 kHz
0011b		A1, A2, A3, B1, B2, B3, B4, C0, C2	139 x Δf^{RA} (See SCS in **Table 5-10**)
0100b	UL filter for NPRACH	0, 1	48 x 3.75KHz = 180 KHz
0101b…111b	Reserved		

Type: unsigned integer.

Field length: 4 bits.

Default Value: 0000b (no special filter).

5.4.4.4 frameId (frame identifier)

Description: This parameter is a counter for 10 ms frames (wrapping period 2.56 seconds), specifically frameId = frame number modulo 256.

Value range: {0000 0000b-1111 1111b}.

Type: unsigned integer.

Field length: 8 bits.

5.4.4.5 subframeId (subframe identifier)

Description: This parameter is a counter for 1 ms sub-frames within 10ms frame.

Value range: {0000b-1111b}.

Type: unsigned integer.

Field length: 4 bits.

5.4.4.6 slotId (slot identifier)

Description: This parameter is the slot number within a 1ms sub-frame. All slots in one sub-frame are counted by this parameter, slotId running from 0 to N_{slot}-1. In this version of the specification the maximum N_{slot}=16, All other values of the 6 bits are reserved for future use.

Value range: {00 0000b-00 1111b=slotID, 01 0000b-11 1111b=Reserved}.

Type: unsigned integer.

Field length: 6 bits.

5.4.4.7 startSymbolid (start symbol identifier)

Description: This parameter identifies the first symbol number within slot, to which the information of this message is applies.

Value range: {00 0000b-11 1111b}.

Type: unsigned integer.

Field length: 6 bits.

5.4.4.8 numberOfsections (number of sections)

Description: This parameter indicates the number of data section descriptions (separate citations of section ID even for multiple citations of the same sectionId) included in this C-Plane message.

Value range: {0000 0000b-1111 1111b}.

Type: unsigned integer.

Field length: 8 bits.

5.4.4.9 sectionType (section type)

Description: This parameter determines the characteristics of U-plane data to be transferred or received from a beam with one pattern id.

Value range: {0000 0000b-1111 1111b=Section Type, see table at the start of this chapter for section type meanings}

Type: unsigned integer.

Field length: 8 bits.

5.4.4.10 udCompHdr (user data compression header)

See section 6.3.3.13 for the description of this parameter.

Note that it is impermissible to specify different values of udCompHdr for the same data section, even if discriminated by different reMask values. Only a single compression method per data section is supported. The O-RU response if the C-Plane specifies multiple udCompHdr values for a single data section is undefined.

5.4.4.11 numberOfUEs (number Of UEs)

Description: This parameter applies to section type 6 messages and indicates the number of UEs (for which channel information is provided) are included in the message. This allows the parser to determine when the last UE's data has been parsed.

Value range: {0000 0000b-1111 1111b}.

Type: unsigned integer.

Field length: 8 bits.

5.4.4.12 timeOffset (time offset)

Description: This parameter defines the time_offset from the start of the slot to the start of the Cyclic Prefix (CP) in number of samples T_s (=1/30.72MHz as specified in 3GPP TS38.211 section 4.1). The value shall be less than the baseline slot length used for slot ID counting as defined in section 5.3.2.

time_offset = timeOffset * t_s

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}.

Type: unsigned integer.

Field length: 16 bits.

5.4.4.13 frameStructure (frame structure)

Description: This parameter defines the frame structure. The first 4 bits define the FFT/iFFT size being used for all IQ data processing related to this message. The second 4 bits define the sub carrier spacing as well as the number of slots per 1ms sub-frame according to 3GPP TS 38.211, taking for completeness also 3GPP TS 36.211 into account. The parameter µ=0…5 from 3GPP TS 38.211 is extended to apply for PRACH processing.

NOTE: The parameter "FFTSize" is meant to facilitate the above calculation and is not meant to strictly dictate the O-RU's method of time-to-frequency conversion.

Value range: {0000 0000b-1111 1111b}

Bit allocations

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
FFT Size				µ (Subcarrier spacing)				1	Octet 1

Table 5-10 : FFT Size

Value of IE "FFT_size"	FFT/iFFT size
0000b	Reserved (no FFT/iFFT processing)
0001b…0110b	Reserved
0111b	128
1000b	256
1001b	512
1010b	1024
1011b	2048
1100b	4096
1101b	1536
1110,1111b	Reserved

Table 5-11 : Sub-Carrier Spacing

Value of IE "SCS"	3GPP "µ"	Subcarrier spacing Δf	Number of slots per 1ms sub-frame: N_{slot}	Slot length
0000b	0	15kHz	1	1ms
0001b	1	30kHz	2	500µs
0010b	2	60kHz	4	250µs
0011b	3	120kHz	8	125µs
0100b	4	240kHz	16	62.5µs
0101b…1011b	NA	Reserved	Reserved	Reserved
1100b	NA	1.25kHz	1	1ms
1101b	NA	3.75kHz (LTE-specific)	1	1ms
1110b	NA	5kHz	1	1ms
1111b	NA	7.5kHz (LTE-specific)	1	1ms

Type: unsigned integer (concatenated bit fields).

Field length: 8 bits.

5.4.4.14 cpLength (cyclic prefix length)

Description: This parameter defines the length CP_length of the Cyclic Prefix (CP) as follows, based on T_s (=1/30.72MHz as specified in 3GPP TS38.211 section 4.1):

CP_length = cpLength * T_s

NOTE: cpLength parameter is used with section types 0 and 3, and is applicable to all symbols addressed by the C-plane message. In addition to the values defined in 3GPP, the value 0 is supported. The slot start time reference differentiation between normal CP and extended CP is based on cpLength and frameStructure. If cpLength is zero then differentiation between normal CP and extended CP time reference is based on cp-type parameter conveyed in the M-Plane. With section type 1, the CP properties shall be configured over M-Plane. In the latter case, the cyclic prefix shall have the following properties:

- cp-type: Normal or extended
- cp-length: same unit as cpLength defined in this section; used for symbol 0 for NR & LTE, and symbol $7*2^{\mu}$ for NR (µ as defined in the table above with "NA" for µ shall be replaced by "0")
- cp-length-other: same unit as cpLength defined in this section; used for other symbols than by cp-length

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}.

Type: unsigned integer.

Field length: 16 bits.

5.4.5 Coding of Information Elements – Application Layer, Sections

5.4.5.1 sectionId (section identifier)

Description: This parameter identifies individual data sections described within the C-Plane message. The purpose of the section ID is to map U-Plane data sections to the corresponding C-Plane message (and Section Type) associated with the data. Two or more C-Plane data section descriptions with same Section ID may be cited corresponding to a single U-Plane data section containing a combined payload for both citations (e.g., for supporting mixed CSI RS and PDSCH). This case is applicable when usage of reMask is complimentary (or orthogonal) and different beam directions (i.e. beamIds) are given the resource elements. See informative reference [13] for a more thorough description of data sections and how they are referenced in C-Plane messages. Data sections are specific to a value of eAxC so different eAxC values may have differently-defined data sections e.g. have different ranges of PRBs contained within.

NOTES:

- In the case of two section descriptions with same Section ID, both sections shall have same rb, startPrbc, numPrbc, udCompHdr, and numSymbol IE fields' content.
- An upper bound on the max number of section IDs that can be addressed per eAxC or per sets of eAxCs, per symbol and per slot, for DL and for UL respectively, shall be conveyed via M-plane messaging as part of the O-RU capabilities description.
- sectionIds are specific to a slot, so sectionId values may be "reused" for each slot, and the sectionId value for one slot has no specified relation to the sectionId value for a different slot.
- The sectionId cited in a C-Plane message must have the same value as the sectionId in the corresponding U-Plane message for the given data section as defined by the frameId, subFrameId, slotId, startSymbolid fields and range of relevant PRBs as indicated by the totality of the specified startPrb(c/u) and numPrb(c/u) fields.

Value range: {0000 0000 0000b-1111 1111 1111b}.

Type: unsigned integer.

Field length: 12 bits.

5.4.5.2 rb (resource block indicator)

Description: This parameter is used to indicate if every RB is used or every other RB is used. The starting RB is defined by startPrbc and total number of used RBs is defined by numPrbc. Example: RB=1, startPrb=1, numPrb=3, then the PRBs used are 1, 3, and 5.

Value range: {0b=every RB used; 1b=every other RB used}.

Type: binary bit.

Field length: 1 bit.

Default Value: 0b (every RB used).

5.4.5.3 symInc (symbol number increment command)

Description: This parameter is used to indicate which symbol number is relevant to the given sectionId. It is expected that for each C-Plane message a symbol number is maintained and starts with the value of startSymbolid. The same value is used for each section in the message as long as symInc is zero. When symInc is one, the maintained symbol number should be incremented to the next symbol, and that new symbol number should be used for that section and each subsequent section until the symInc bit is again detected to be one. In the case of a multiple-symbol data section (numSymbol > 1), the new symbol number shall be the one after the last symbol in the data section. In this manner, multiple symbols may be handled by a single C-Plane message.

A few other points regarding use of symInc in the C-Plane:

a) SymInc may be used when different data sections have a different number of symbols under certain conditions namely that the data section descriptions in the C-Plane message are carefully arranged. The arrangement must be such that the following procedure as applied by the O-RU works correctly:

   symbol[s] - symbol addressed by section description s=1…N (considers section descriptions in one C-Plane message, s=1 identifies the first section description) shall be calculated as follows:

   Let symInc[s] and numSymbols[s] are values of corresponding fields of section description s
   For sake of simplicity let symbol[0] = startSymbolId, symInc[0]=0 and numSymbols[0]=1
   for s=1…N (all section descriptions in a message)

      If symInc[s] = 0 then symbol[s] = symbol[s-1]

      Else symbol[s] = symbol[s-1] + numSymbols[s-1]

b) When a data section description is cited multiple times (e.g. with different reMask values) if the value of symInc is to be set to 1, then only the first invocation of the sectionID shall have symInc=1 and all other invocations of the sectionId shall have symInc=0 in the same C-Plane message. This assures that the above rule can be applied (in the same C-Plane message only) while assuring that all invocations of the data section descriptions will refer to the same symbols, which must be the case whether the invocations are in the same C-Plane message or in separate C-Plane messages.

Use of symInc in the U-Plane is independent of the use of symInc in the C-Plane. The current view is that U-Plane message includes data for a single symbol (to avoid degrading the latency performance of the U-Plane), so use of symInc=1 is prohibited in the U-Plane as of this version of the specification. A future version may allow use of symInc=1 in the U-Plane to allow multiple symbols to be contained in a single U-Plane message.

Value range: {0b=use the current symbol number; 1b=increment the current symbol number and use that}.

Type: binary bit.

Field length: 1 bit.

Default Value: 0b (do not increment the current symbol number).

5.4.5.4 startPrbc (starting PRB of data section description)

Description: This parameter is the starting PRB of a data section described in the C-Plane message. Values of startPrbc and numPrbc must ensure that data sections must never overlap: a single PRB may only exist within one data section for a given value of eAxC. For one C-Plane message, there may be multiple U-Plane messages associated with it and requiring defining from which PRB the control commands are applicable.

Note: freqOffset affects the frequency span for specific range of PRB numbers. Therefore "must never overlap" must consider the value of freqOffset.

Value range: {00 0000 0000b-11 1111 1111b}.

Type: unsigned integer.

Field length: 10 bits.

5.4.5.5 reMask (resource element mask)

Description: This parameter defines the Resource Element (RE) mask within an PRB. Each bit setting in the reMask indicates if the section control is applicable to the RE sent in U-Plane messages (0=not applicable; 1=applicable).

Note that different REs in a PRB may be indicated by different invocations of the same sectionId but with differing reMasks; the maximum number of different reMask values that may be applied to a PRB is an O-RU characteristic that is conveyed from the O-RU to the O-DU via the M-Plane. If any RE in a PRB is never pointed to by an reMask (but other REs in that PRB are), the “missing” RE should be set to zero in the U-Plane, and no beamforming ID or other processing should be applied to the “missing” RE. No RE may be referenced more than once in a data section.

Value range: {0000 0000 0000b-1111 1111 1111b}.

Type: unsigned integer (bit mask).

Field length: 12 bits.

Default Value: 1111 1111 1111b (all REs in the block applicable).

5.4.5.6 numPrbc (number of contiguous PRBs per data section description)

Description: This parameter defines the PRBs within the data section being described. Values of startPrbc and numPrbc must ensure that data sections must never overlap: a single PRB may only exist within one data section for a given value of eAxC.

Value range: {0000 0001b-1111 1111b, 0000 0000b = all PRBs}.

Value 0000 0000b is particularly relevant for NR cases wherein the total number of PRBs may be more than 255.

Type: unsigned integer.

Field length: 8 bits.

5.4.5.7 numSymbol (number of symbols)

Description: This parameter defines number of symbols to which the section control is applicable. At minimum, the section control shall be applicable to at least one symbol. However, possible optimizations could allow for several (up to 14) symbols, if e.g., all 14 symbols use the same beam ID.

Value range: {0001b-1110b=number of symbols, 0000b=reserved, 1111b=reserved}.

Type: unsigned integer.

Field length: 4 bits.

5.4.5.8 ef (extension flag)

Description: This parameter is used to indicate if this section will contain both beamforming index and any extension information (ef=1) or just a beamforming index ef=0)

Value range: {0b=beamforming index provided; 1b=beamforming index and extension information provided}.

Type: binary bit.

Field length: 1 bit.

Default Value: 0b (no weights being sent).

5.4.5.9 beamId (beam identifier)

Description: This parameter defines the beam pattern to be applied to the U-Plane data. beamId = 0 means no beamforming operation will be performed.

Note that the beamId encodes the beamforming to be done on the O-RU. This beamforming may be digital, analog or both ("hybrid beamforming") and the beamId provides all the information necessary for the O-RU to select the correct beam (or weight table or beam attributes from which to create a beam). It is intended that the beamId be global for the O-RU meaning there are 32767 possible beams shared within the O-RU for all Rtcids/Pcids and shared between UL and DL (beamId=0x0000 is reserved for no beamforming). The specific mapping of beamId to e.g. weight table, beam attributes, directionality, beam adjacency or any other beam designator is specific to the O-RU design and must be conveyed via M-Plane from the O-RU to O-DU upon startup.

NOTE: An upper bound on the max number of beamIDs that can be addressed per eAxC or per sets of eAxCs, per symbol and per slot, for DL and for UL respectively, shall be conveyed via M-plane messaging as part of the O-RU capabilities description.

Value range: {000 0000 0000 0001b-111 1111 1111 1111b; 000 0000 0000 0000b means no BF to be done}

Type: unsigned integer.

Field length: 15 bits.

Default Value: 0000 0000 0000 0000b (no beamforming).

5.4.5.10 ueId (UE identifier)

Description: This parameter provides a label for the UE for which the section contents apply. This is used to support channel information sending from the O-DU to the O-RU. This is just a label and the specific value has no meaning regarding types of UEs that may be supported within the system.

Value range: {000 0000 0000 0000b-111 1111 1111 1111b}

Type: unsigned integer.

Field length: 15-6 bits.

5.4.5.11 freqOffset (frequency offset)

Description: This parameter defines the frequency offset with respect to the carrier center frequency before additional filtering (e.g. for PRACH) and FFT processing (in UL) in steps of one half the subcarrier spacings Δf. The frequency offset shall be individual per data section (one cannot use reMask to allow different frequency offsets for different REs in the PRBs).

frequency_offset = freqOffset * Δf * 0.5

Note: Frequency span resulting from frameStructure, freqOffset, startPrbc, numPrbc, and rb must not exceed channel bandwidth configured for eAxC over M-plane.**Value range:** {0000 0000b=no offset, 0000 0001b-0111 1111b=positive frequency offset, 1000 0001b-1111 1111=negative frequency offset}.

Type: signed integer.

Field length: 24 bits.

5.4.5.12 regularizationFactor (regularization Factor)

Description: This parameter provides a signed value to support MMSE operation within the O-RU when beamforming weights are supported in the O-RU, so related to section type 6.

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}

Type: unsigned integer.

Field length: 16 bits.

5.4.5.13 ciIsample, ciQsample (channel information I and Q values)

Description: These values are the channel information complex values relayed from the O-DU to the O-RU, related to section type 6. The order of transmission is first Prbc for the first antenna to the last antenna, then second Prbc for the first to last antenna, and so on until reaching the last Prbc for the first to last antenna. The bit-width per I and Q value is variable (determined by M-Plane messaging) so after the very last Q value, some number of padding (set to zero) bits may be inserted to get to the next byte boundary.

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b} for each I and Q value

Type: signed integer.

Field length: variable bit-with per I and Q value.

5.4.5.14 laaMsgType (LAA message type)

Description: This parameter defines the LAA message type being conveyed within the Section Type 7 C-Plane message. Note that here the C-Plane messages may flow from the O-DU to the O-RU (as usual) or from the O-RU to the O-DU. All of these messages relate to the listen-before-talk (LBT) LAA operation and provide a "handshake" between the O-DU and the O-RU to manage the LBT operation.

Value range: {0001b-1111b}.

Table 5-12 : laaMsgType definition

laaMsgType	lssMsgType definition	lssMsgType meaning
0000b	LBT_PDSCH_REQ	O-DU to O-RU request to obtain a PDSCH channel
0001b	LBT_DRS_REQ	O-DU to O-RU request to obtain the channel and send DRS
0010b	LBT_PDSCH_RSP	O-RU to O-DU response, channel acq success or failure
0011b	LBT_DRS_RSP	O-RU to O-DU response, DRS sending success or failure
0100b	LBT_Buffer_Error	O-RU to O-DU response, reporting buffer overflow
0101b	LBT_CWCONFIG_REQ	O-DU to O-RU request, congestion window configuration

0110b	LBT_CWCONFIG_RSP	O-RU to O-DU response, congestion window config. response
0100b – 1111b	reserved for future methods	

Type: unsigned integer.

Field length: 4 bits.

5.4.5.15 laaMsgLen (LAA message length)

Description: This parameter defines number of 32-bit words in the LAA section, where “1” means one 32-bit word, “2” means 2 32-bit words, etc. – including the word containing the lssMsgLen parameter. Zero is a reserved value.

Value range: {0001b-1111b=number of 32-bit words in the section from 1 to 16 words (4 to 64 bytes)
0000b is a reserved value }

Type: unsigned integer.

Field length: 4 bits.

5.4.5.16 lbtHandle

Description: This parameter provides a label that is included in the configuration request message (e.g., LBT_PDSCH_REQ, LBT_DRS_REQ) transmitted from the O-DU to the O-RU and returned in the corresponding response message (e.g., LBT_PDSCH_RSP, LBT_DRS_RSP).

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b}

Type: unsigned integer.

Field length: 16 bits.

5.4.5.17 lbtDeferFactor (listen-before-talk defer factor)

Description: Defer factor in sensing slots as described in 3GPP TS 36.213 Section 15.1.1. This parameter is used for LBT CAT 4 and can take one of three values: {1,3, 7} based on the priority class. Four priority classes are defined in 3GPP TS 36.213.

Value range: {001b, 011b, 111b} or {1, 3, 7} in decimal

Type: unsigned integer

Field length: 3 bits

5.4.5.18 lbtBackoffCounter (listen-before-talk backoff counter)

Description: LBT backoff counter in sensing slots as described in 3GPP TS 36.213 Section 15.1.1.

Value range: {00 0000 0000b – 11 1111 1111b} (0-1023 decimal)

Type: unsigned integer

Field length: 10 bits

5.4.5.19 lbtOffset (listen-before-talk offset)

Description: LBT start time in microseconds from the beginning of the subframe scheduled by this message

Value range: {00 0000 0000b – 11 1110 0111b} or {0 – 999} in decimal

Type: unsigned integer

Field length: 10 bits

5.4.5.20 MCOT (maximum channel occupancy time)

Description: LTE TXOP duration in subframes as described in 3GPP TS 36.213 Section 15.1.1. The maximum values for this parameter are {2, 3, 8, 10} based on the priority class. Four priority classes are defined in 3GPP TS 36.213.

Value range: {1-10} in decimal

Type: unsigned integer

Field length: 4 bits

5.4.5.21 lbtMode (LBT Mode)

Description: Part of multi-carrier support. Indicates whether full LBT process is carried or partial LBT process is carried (multi carrier mode B according to 3GPP TS 36.213 Section 15.1.5.2).

00b = full LBT (regular LBT, sending reservation signal until the beginning of the SF/slot)
01b = Partial LBT (looking back 25 µsec prior to transmission as indicated in 3GPP TS 36.213 section 15.1.5.2)
10b = Partial LBT (looking back 34 µsec prior to transmission as indicated in 3GPP TS 36.213 section 15.1.5.2)
11b = full LBT and stop (regular LBT, do NOT send reservation signal; O-RU senses the spectrum for the defer factor + a sensing slot right before OTA when the O-DU is ready to transmit data as indicated in 3GPP TS 36.213 section 15.1.1. i.e., right before the SF/slot boundary)

Value range: {00b - 11b}

Type: unsigned integer

Field length: 2 bits

5.4.5.22 lbtPdschRes (LBT PDSCH Result)

Description: LBT result of SFN/SF

- 00b = not sensing – indicates that the O-RU is transmitting data
- 01b = currently sensing – indicates the O-RU has not yet acquired the channel
- 10b = success – indicates that the channel was successfully acquired
- 11b = Failure – indicates expiration of the LBT timer. The LBT process should be reset.

Value range: {00b - 11b}

Type: unsigned integer

Field length: 1 bit

5.4.5.23 sfStatus (subframe status)

Description: indicates whether the subframe was dropped or transmitted

- 0 – subframe was dropped
- 1 – subframe was transmitted

Value range: {0,1}

Type: binary bit

Field length: 1 bit

5.4.5.24 lbtDrsRes (LBT DRS Result)

Description: LBT result of SFN/SF

- 0 – SUCCESS – indicates that DRS is sent
- 1 – FAILURE – indicates that DRS is not sent

Value range: {0,1}

Type: binary bit

Field length: 1 bit

5.4.5.25 initialPartialSF (Initial partial SF)

Description: Indicates whether the initial SF in the LBT process is full or partial.

- 0 – full SF (two slots, 14 symbols)
- 1 – partial SF (only second slot, last 7 symbols)

Value range: {0, 1}

Type: binary bit

Field length: 1 bit

5.4.5.26 lbtBufErr (LBT Buffer Error)

Description: Indicates when an LBT buffer overflow has occurred.

- 0 – reserved
- 1 – buffer overflow – data received at O-RU is larger than the available buffer size

Value range: {0, 1}

Type: binary bit

5.4.5.27 sfnSf (SFN/SF End)

Description: SFN/SF which is dropped by O-RU because of time expired or successfully transmitted

Value range: SFN: {0000 0000b – 1111 1111b} or {0 – 255} in decimal, SF: {0000b – 1001b} or {0 – 9} in decimal

Type: unsigned integer

Field length: 12 bits

5.4.5.28 lbtCWConfig_H (HARQ Parameters for Congestion Window management)

Description: Total number of HARQ NACK feedback messages received for the lbtCWConfig_T transport blocks transmitted over LAA cell(s)to be used for adjusting the Congestion Window.

Value range: {0000 0000b – 1111 1111b} or {0 – 255} in decimal

Type: unsigned integer

Field length: 8 bits

5.4.5.29 lbtCWConfig_T (TB Parameters for Congestion Window management)

Description: Total number of Transport Blocks (TB) transmitted over the LAA cell to be used for adjusting the Congestion Window.

Value range: {0000 0000b – 1111 1111b} or {0 – 255} in decimal

Type: unsigned integer

Field length: 8 bits

5.4.5.30 lbtTrafficClass (Traffic class priority for Congestion Window management)

Description: Channel access priority class as defined in TS 36.213

Value range: {000 - 111} or {0 - 7} in decimal

1 - 4: traffic class priority

0, 5 - 7: reserved

Type: unsigned integer

Field length: 3 bits

5.4.5.31 lbtCWR_Rst (Notification about packet reception successful or not)

Description: Notifies to O-DU whether the O-RU receives LBT_CWCONFIG_REQ message successfully or not

- 0 – SUCCESS – indicates successful reception of LBT_CWCONFIG_REQ
- 1 – FAILURE – indicates failure of receiving LBT_CWCONFIG_REQ

Value range: {0, 1}

Type: binary bit

Field length: 1 bit

5.4.5.32 reserved (reserved for future use)

Description: This parameter is reserved for future use. Transmitter shall send value “0”, while receiver shall ignore the value received.

Value range: {variable}.

Type: variable.

Field length: variable.

5.4.6 Section Extension Commands

The following section extension parameters are defined within the C-Plane:

Table 5-13 : Section Extension Commands

extType	meaning	extLen	extension parameters	octets	meaning
0	reserved	1 (1 word)	reserved reserved	1 1	for future use for future use
1	beamforming weights	var	bfwCompHdr bfwCompParam bfwI (for TRX 0) bfwQ (for TRX 0) … bfwI (for last TRX) bfwQ (for last TRX) zero-pad to 4-byte boundary	1 1 var var var var var	bitWidth(3:0) | compMeth(3:0) depends on compr. method beamforming weight I value beamforming weight Q value beamforming weight I value beamforming weight Q value
2	beamforming attributes	var	bfaCompHdr	2	BF attributes compr. header

			bfAzPt bfZePt bfAz3dd bfZe3dd bfAzSl bfZeSl	var var var var 3b 3b	BF azimuth pointing param BF zenith pointing param BF azimuth beamwidth param BF zenith beamwidth param BF azimuth sidelobe param BF zenith sidelobe param
3	DL Precoding configuration parameters and indications	var (3 or 4 words)	codebookIndex layerId txScheme numLayers crsReMask crsSymNum crsShift beamIdAP1 beamIdAP2 beamIdAP3	1 4b 4b 4b 12b 4b 1b 2 2 2	precoder codebook layer ID for DL Tx transmission scheme number of layers in DL Tx CRS RE Mask CRS symbol number CRS shift command Beam ID, Antenna Port 1 Beam ID, Antenna Port 2 Beam ID, Antenna Port 3
4	modulation compr. params	0	csf modCompScaler	1b 15b	constellation shift flag mod. compr. scale value
5	modulation compression additional scaling parameters	var	mcScaleReMask csf mcScaleOffset	12b 1b 15b	Position of same scaling bits constellation shift flag added mod. compr. scale values
6-127	reserved	1 (1 word)	reserved reserved	1 1	for future use for future use

5.4.6.1 extType (extension type)

Description: This parameter provides the extension type which provides additional parameters specific to the subject data extension. An O-RU or O-DU receiving a "reserved" section extension shall ignore the extension and all parameters contained within it.

Value range: {all zeros – all ones}. See Table 5-14 for values.

Type: unsigned integer.

Field length: 7 bits.

5.4.6.2 ef (extension flag)

Description: This parameter is used to indicate if there is another extension present (ef=1) or this is the last extension ef=0)

Value range: {0b=beamforming index provided; 1b=beamforming index and extension information provided}.

Type: binary bit.

Field length: 1 bit.

5.4.6.3 extLen (extension length)

Description: This parameter provides the length of the section extension in units of 32-bit (or 4-byte) words. The value zero is reserved, so there is always at least one word in the extension (the word containing the extType and extLen).

Value range: {all zeros – all ones} – form one word to 255 words.

Type: unsigned integer.

Field length: 8 bits.

5.4.7 Coding of Information Elements – Application Layer, Section Extensions

5.4.7.1 ExtType=1: Beamforming Weights Extension Type

This section applies to the sending of beamforming weights from the O-DU to the O-RU. When this is done, the weights are sent along with a beamID which is meant to allow those same weights to be used in future C-Plane messages by invoking the same beamID (without the need to send the weights again). This allows downloaded weights to have "persistence" which should save DL throughput by not requiring sending of weights multiple times. This section extension applies only to section types 1 and 3.

Table 5-14 : Extension Type 1 Data Format

ef	extType = 0x1	1	Octet 23
extLen		1	Octet 24
bfwCompHdr		1	Octet 25
bfwCompParam		var	Octet 26
bfwI (for TRX 0)		var	
bfwQ (for TRX0)		var	
remaining beamforming weights bfwI and bfwQ up to L TRXs		var	
zero pad to 4-byte boundary		var	

5.4.7.1.1 bfwCompHdr (beamforming weight compression header)

Description: This parameter defines the compression method and IQ bit width for the beamforming weights in the specific section in the C-Plane message. In this way each set of weights may employ a separate compression method. Note that for the block compression methods, the block size is the entire vector of beamforming weights, not some subset of them.

Value range: {0000 0000b-1111 1111b}

Bit allocations

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
bfwIqWidth				bfwCompMeth				1	Octet 1

Table 5-15 : udIqWidth definition

bfwIqWidth	Bit width of each I and each Q

0000-1111b	value of bfwIqWidth except a value of zero means 16 bits
	e.g. bfwIqWidth = 0000b means I and Q are each 16 bits wide;
	e.g. bfwIqWidth = 0001b means I and Q are each 1 bit wide;
	e.g. bfwIqWidth = 1111b means I and Q are each 15 bits wide

Table 5-16 : bfwCompMeth definition

bfwCompMeth	compression method	udIqWidth meaning
0000b	no compression	bitwidth of each uncompressed I and Q value
0001b	block floating point	bitwidth of each I and Q mantissa value
0010b	block scaling	bitwidth of each I and Q scaled value
0011b	μ-law	bitwidth of each compressed I and Q value
0100b	beamspace compression	bitwidth of each beamspace I and Q coefficient
0101b – 1111b	reserved for future methods	depends on the specific compression method

Type: unsigned integer (concatenated bit fields).

Field length: 15 bits.

Default Value: 1111 0000b (no compression, 16-bit I and Q).

5.4.7.1.2 bfwCompParam (beamforming weight compression parameter)

Description: This parameter applies to the compression method specified by the associated sectionID's bfwCompMeth value.

Value range: {0000 0000b-1111 1111b}.

Bit allocations

bfwCompMeth	0 (msb)	1	2	3	4	5	6	7 (lsb)	compParam size
0000b = no compression	absent								0 octets
0001b = block fl. point	reserved (set to all zeros)				Exponent (unsigned)				1 octet
0010b = block scaling	blockScaler (unsigned, 1 integer bit, 7 fractional bits)								1 octet
0011b = μ-law	compBitWidth				compShift				1 octets
0100b = beamspace	activeBeamspaceCoefficientMask								ceil(N/8)*8 octets
	blockScaler (unsigned,1 integer bit,7 fractional bits)								1 octet
0101b – 1111b	reserved (set to all zeros)								? octets

* N is the total number of weights in the section. N is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane

Type: unsigned integer (concatenated bit fields).

Field length: zero for bfwCompMeth values 0000b and 0011b, 8 bits for bfwCompMeth values 0001b, and 0010b; other bfwCompMeth values may imply other lengths but will always be an integer number of bytes.

5.4.7.1.3 bfwI (beamforming weight in-phase value)

Description: This parameter is the In-phase beamforming weight value. The total number of weights in the section is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane.

Value range: {all zeros – all ones}.

Type: signed integer.

Field length: 1-16 bits.

5.4.7.1.4 bfwQ (beamforming weight quadrature value)

Description: This parameter is the Quadrature beamforming weight value. The total number of weights in the section is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane.

Value range: {all zeros – all ones}.

Type: signed integer.

Field length: 1-16 bits.

5.4.7.2 ExtType=2: Beamforming Attributes Extension Type

This section extension applies only to section types 1 and 3.

The following table shows the format of this section extension.

Table 5-17 : Extension Type 2 Data Format

ef	extType = 0x02		1	Octet 25
extLen			1	Octet 26
bfaCompHdr			2	Octet 27
bfAzPt			var	Octet 29
bfZePt			var	
bfAz3dd			var	
bfZe3dd			var	
zero-padding	bfAzSl	bfZeSl	1	
zero padding to achieve 4-byte alignment as needed				

5.4.7.2.1 bfaCompHdr (beamforming attributes compression header)

Description: This parameter defines the bit width for the beamforming attributes extension parameters. In this way each set of beamforming attributes may employ a different bit width.

Value range: {0000 0000 0000b-1111 1111 1111b}

Bit allocations

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
Reserved		bfAzPtWidth			bfZePtWidth			1	Octet 1
Reserved		bfAz3ddWidth			bfZe3ddWidth			1	Octet 2

Table 5-18 : fzAzPtWidth definition

bfAzPtWidth	Bit width of bfAzPt
000-111b	value of bfAzPtWidth

Table 5-19 : bfZePtWidth definition

bfZePtWidth	Bit width of bfZePt
000-111b	value of bfZePtWidth

Table 5-20 : bfAz3ddWidth definition

bfAz3ddWidth	Bit width of bfAz3dd
000-111b	value of bfAz3ddWidth

Table 5-21 : bfZe3ddWidth definition

bfZe3ddWidth	Bit width of bfZe3dd
000-111b	value of bfZe3ddWidth

For each of the four bitwidth values in this parameter (bfAzPtWidth, bfZePtWidth, bfAz3ddWidth, and bfZe3ddWidth) the following mapping shall be used:

000b = no bits, the field is no applicable (O-RU cannot support it) or the default value shall be used.
001b = 2-bit bitwidth
010b = 3-bit bitwidth
011b = 4-bit bitwidth
100b = 5-bit bitwidth
101b = 6-bit bitwidth
110b = 7-bit bitwidth
111b = 8-bit bitwidth (this is the highest bitwidth anticipated to be needed)

Type: unsigned integer (concatenated bit fields).

Field length: 16 bits (4 bits are reserved).

Default Value: 1111 1111 1111 1111b (, 8-bit azimuth and zenith pointing angle and 8-bit azimuth and zenith beamwidth).

5.4.7.2.2 bfAzPt (beamforming azimuth pointing parameter)

Description: This parameter is the azimuth beamforming pointing angle in degrees. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane.

Value range: {all zeros – all ones}.

Type: signed integer.

Field length: 0-8 bits.

5.4.7.2.3 bfZePt (beamforming zenith pointing parameter)

Description: This parameter is the zenith beamforming pointing angle in degrees. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane.

Value range: {all zeros – all ones}.

Type: unsigned integer.

Field length: 0-8 bits.

5.4.7.2.4 bfAz3dd (beamforming azimuth beamwidth parameter)

Description: This parameter is the azimuth beamforming beamwidth in degrees. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane. The value (000b) corresponds to the minimum valid beamwidth.

Value range: {all zeros – all ones}.

Type: unsigned integer.

Field length: 0-8 bits.

5.4.7.2.5 bfZe3dd (beamforming zenith beamwidth parameter)

Description: This parameter is the zenith beamforming beamwidth in degrees. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane. The value (000b) corresponds to the minimum valid beamwidth.

Value range: {all zeros – all ones}.

Type: unsigned integer.

Field length: 0-8 bits.

5.4.7.2.6 bfAzSl (beamforming azimuth sidelobe parameter)

Description: This parameter is the azimuth beamforming sidelobe suppression value in dB. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane. The value of bfAzSl corresponds to a value of 10 dB for all zeros and increments in 5 dB steps (e.g. 001 corresponds to 15 dB, 010 corresponds to 20 dB, and so on).

Value range: {all zeros – all ones}.

Type: unsigned integer.

Field length: 3 bits.

5.4.7.2.7 bfZeSl (beamforming zenith sidelobe parameter)

Description: This parameter is the zenith beamforming sidelobe suppression value in dB. The valid range of values is O-RU-specific and is conveyed from the O-RU to the O-DU as part of the initialization procedure via the M-Plane. The value of bfZeSl corresponds to a value of 10 dB for all zeros and increments in 5 dB steps (e.g. 001 corresponds to 15 dB, 010 corresponds to 20 dB, and so on).

Value range: {all zeros – all ones}.

Type: unsigned integer.

Field length: 3 bits.

5.4.7.2.8 zero-padding

Description: This parameter is intended to pad out the data to the next 4-byte boundary. Because the preceding parameters are of varying bitwidth, extra padding in most cases will be needed to achieve the 4-byte boundary condition. Transmitter shall send value “0”, while receiver shall ignore the value received.

Value range: {variable}.

Type: variable.

Field length: variable.

5.4.7.3 ExtType=3: DL Precoding Extension Type

This section extension applies only to section types 1 and 3 and is to be used only for LTE TM2-4 not for other LTE transmission modes and not for NR. For other LTE transmission modes and for NR, precoding is assumed to be included in the beamforming operation (that is, encoded in the beamforming weights).

The following table shows the format of this section extension.

Table 5-22 : Extension Type 3 Data Format – first data layer

<table>
<tr><td>ef</td><td colspan="2">extType = 0x03</td><td>1</td><td>Octet N</td></tr>
<tr><td colspan="3">extLen = 0x04 (4 words)</td><td>1</td><td>N+1</td></tr>
<tr><td colspan="3">codebookIndex</td><td>1</td><td>N+2</td></tr>
<tr><td colspan="2">layerId = 0000b or 1111b</td><td>numLayers</td><td>1</td><td>N+3</td></tr>
<tr><td colspan="2">txScheme</td><td>crsReMask[11:8]</td><td>1</td><td>N+4</td></tr>
<tr><td colspan="3">crsReMask[7:0]</td><td>1</td><td>N+5</td></tr>
<tr><td>crsShift</td><td>reserved</td><td>crsSymNum</td><td>1</td><td>N+6</td></tr>
<tr><td colspan="3">reserved</td><td>3</td><td>N+7</td></tr>
<tr><td colspan="3">beamIdAP1</td><td>2</td><td>N+10</td></tr>
<tr><td colspan="3">beamIdAP2</td><td>2</td><td>N+12</td></tr>
<tr><td colspan="3">beamIdAP3</td><td>2</td><td>N+14</td></tr>
</table>

Table 5-23 : Extension Type 3 Data Format – not first data layer

<table>
<tr><td>ef</td><td colspan="2">extType = 0x03</td><td>1</td><td>Octet N</td></tr>
<tr><td colspan="3">extLen = 0x01 (1 word)</td><td>1</td><td>N+1</td></tr>
<tr><td colspan="3">codebookIndex</td><td>1</td><td>N+2</td></tr>
<tr><td colspan="2">layerId ≠ 0000b or 1111b</td><td>numLayers</td><td>1</td><td>N+3</td></tr>
</table>

There may be two or four antenna ports hence two or four beamIDs needed (same beamID for user data and CRS REs). For Antenna Port 0, the beamId is contained in the C-Plane data section header, while the Antenna Ports 1-3 beamIDs are contained in this section extension. When there are two antenna ports, the section extension only contains the second Antenna Port beam ID ("beamIdAP1") and the section extension length is 3 words ("extLen" = 0x3). When there are four antenna ports, the section extension contains the second, third and fourth Antenna Port beam IDs ("beamIdAP1", "beamIdAP2". And "beamIdAP3") and the section extension length is 4 words ("extLen" = 0x4).

For the txScheme indicating TxD, one Pcid is used for all the user data and one section instantiation is needed using the corresponding C-Plane Rtcid, providing all the beam IDs (up to 4) for the user data; a second section instantiation (same sectionId) with a different reMask may be used to provide the CRS RE beam IDs (also up to 4). In the TxD case the layerId is set to 1111b ("TxD").

For the txSchemes indicating spatial multiplexing (LD CDD or no CDD), each layer will have its own Pcid for the user data with a corresponding C-Plane Rtcid conveying the user data's beamId, with this section extension showing a different layer number for each layer. Only within the layer ID zero Rtcid will the CRS REs be provided with their beamIDs (one beamId in the section header and the other beamIDs in this section extension). For the non-zero layer number Rtcids, this section extension will still be provided to guide the precoding operation (provide the layer ID) but will not include beamIds (extLen = 0x1), and the beamId in the section header should be ignored by the O-RU and should be set to the default value by the O-DU.

5.4.7.3.1 codebookIndex (precoder codebook used for transmission)

Description: This parameter defines the indices of the precoder codebook that are used for precoding. It is to be used in conjunction with the numLayers field. (Invalid for TM1, TM2 and TM3)

Value range: {0000 0000b - 1111 1111b}.

Type: unsigned integer.

Field length: 8 bits.

DefaultValue: 0000 0000b (used for invalid mode)

5.4.7.3.2 layerID (Layer ID for DL transmission)

Description: This parameter defines the layer ID that are used for DL transmission in TM1 – TM4.

Value range: {0000b-1111b}. 0000b implies layer0, 0001b implies layer1, 0010b implies layer2, 0011b implies layer3. (for TxD, set to all ones)

Type: unsigned integer.

Field length: 4 bits.

DefaultValue: 1111b (used for TxD mode)

5.4.7.3.3 txScheme (transmission scheme)

Description: This parameter defines the TM scheme used in this section type.

Value range: {0000b-1111b} 0000b: Spatial multiplexing (CDD) 0001b – Spatial multiplexing (no CDD), 0010b – Transmit diversity, 0011b-111b reserved

Type: unsigned integer.

Field length: 4 bits.

5.4.7.3.4 numLayers (number of layers used for DL transmission)

Description: This parameter defines the number of layers that are used for DL transmission in TM1 – TM6.

Value range: {0000b-1111b}. 0000b implies 1 layer, 0001 implies 2 layers, 0010 implies 3 layers, 0011b implies 4 layers,

Type: unsigned integer.

Field length: 4 bits.

5.4.7.3.5 crsReMask (CRS resource element mask)

Description: This parameter defines the CRS Resource Element (RE) mask within a PRB. Each bit setting in the crsReMask indicates if the section control is applicable to the RE sent in U-Plane messages (0=not applicable, 1=applicable)

Value range: {0000 0000 0000b-1111 1111 1111b}

Type: unsigned integer (bit mask).

Field length: 12 bits.

5.4.7.3.6 crsSymlNum (CRS symbol number indication)

Description: This parameter defines the CRS symbol number within a PRB. The value of the crsSymNum index indicates the symbol number to the RE sent in U-Plane messages (0=not applicable, 1=applicable)

Value range: {0000b-1111b}, value indicates symbol number.

0000b – 1101b : use symbol number 0 – 13 respectively;

1110b – 1111b : reserved

Type: unsigned integer.

Field length: 4 bits.

5.4.7.3.7 crsShift (crsShift used for DL transmission)

Description: This parameter indicates the shift pattern to pick up the right index for CRS positions for N Antennas (see **Figure I-8**)

Value range: 0 or 1, implying shift patterns that are layer-dependent according to the table below (see **Tables I-1** through **I-3** for vShift)

	1 Layer	2 Layers	4 Layers
crsShift	**=0 for** $0 \le$ vshift ≤ 5	**0 for** $0 \le$ vshift ≤ 2 **1 for** $3 \le$ vshift ≤ 5	**0 for** $0 \le$ vshift ≤ 2 **1 for** $3 \le$ vshift ≤ 5

Type: binary.

Field length: 1 bit.

5.4.7.3.8 beamIdAP1 (beam id to be used for antenna port 1)

Description: This parameter defines the beam pattern to be applied to the U-Plane data. beamId = 0 means no beamforming operation will be performed. Note that unlike the 15-bit beamId that is part of the section header, this field is 16 bits so in principle (though likely not in fact) there are more IDs available within this field.

Value range: {0000 0000 0000 0001b-1111 1111 1111 1111b}

Type: unsigned integer.

Field length: 16 bits.

Default Value: 0000 0000 0000 0000b (no beamforming).

5.4.7.3.9 beamIdAP2 (beam id to be used for antenna port 2)

Description: This parameter defines the beam pattern to be applied to the U-Plane data. beamId = 0 means no beamforming operation will be performed. Note that unlike the 15-bit beamId that is part of the section header, this field is 16 bits so in principle (though likely not in fact) there are more IDs available within this field.

Value range: {0000 0000 0000 0001b-1111 1111 1111 1111b}

Type: unsigned integer.

Field length: 16 bits.

Default Value: 0000 0000 0000 0000b (no beamforming).

5.4.7.3.10 beamIdAP3 (beam id to be used for antenna port 3)

Description: This parameter defines the beam pattern to be applied to the U-Plane data. beamId = 0 means no beamforming operation will be performed. Note that unlike the 15-bit beamId that is part of the section header, this field is 16 bits so in principle (though likely not in fact) there are more IDs available within this field.

Value range: {0000 0000 0000 0001b-1111 1111 1111 1111b}

Type: unsigned integer.

Field length: 16 bits.

Default Value: 0000 0000 0000 0000b (no beamforming).

5.4.7.4 ExtType=4: Modulation Compression Parameters Extension Type

This section extension applies only to section types 1 and 3. **Table 5-24** shows the section extension format.

Table 5-24 : Section Format for Section Extension 2 (modulation compression parameters)

ef	extType = 0x04	1	Octet N
extLen = 0x01 (1 word)		1	N+1
csf	modCompScaler[14:8]	1	N+2
modCompScaler[7:0]		1	N+3

5.4.7.4.1 csf (constellation shift flag)

Description: This binary flag indicates whether to shift the constellation (csf=1) or not (csf=0). “Shift” means subtract from (during compression) or add to (during decompression) the I and Q samples the value 2^{-N} where “N” depends on the modulation constellation type.

Table 5-25 : constellation shift definition

N	Modulation	Input constellation point I and Q values	Shift value
1	QPSK	-1/2, 1/2	-1/2
2	16QAM	-3/4, -1/4, 1/4, 3/4	-1/4
3	64QAM	-7/8, -5/8, -3/8, -1/8, 1/8, 3/8, 5/8, 7/8	-1/8
4	256QAM	-15/16,-13/16,-11/16,-9/16,-7/16,-5/16,-3/16,-1/16,1/16,3/16,5/16,7/16,9/16,11/16,13/16,15/16	-1/16
5	1024QAM	-31/32, -29/32, -27/32, -25/32, -23/32, -21/32, -19/32, -17/32, -15/32, etc. until 29/32, 31/32	-1/32

Value range: {0b-1b}

Type: binary.

Field length: 1 bit.

5.4.7.4.2 modCompScaler (modulation compression scaler value)

Description: This parameter is the scale factor to apply to the unshifted constellation points during decompression. It is a fractional floating-point value having an unsigned but negative 4-bit exponent and an unsigned fractional 11-bit mantissa.

Value range: { 0 through +(1-2^{-11}) }.

Type: unsigned fractional floating-point value.

$$"mantissa" = \sum_{k=0}^{10} modCompScaler[k] \cdot 2^{k-11}$$

$$"exponent" = \sum_{k=11}^{14} modCompScaler[k] \cdot 2^{k-10}$$

“exponent” is the most significant 4 bits of the 15-bit modCompScaler field and “mantissa” is the least-significant 11 bits of the modCompScaler field. “modCompScaler[k] refers to the k^{th} bit of the modCompScaler field. Therefore, the actual value of modCompScaler is:

$$modCompScaler = mantissa \cdot 2^{-exponent}$$

Field length: 15 bits.

5.4.7.5 ExtType=5: Modulation Compression Additional Parameters Extension Type

This section extension applies only to section types 1, 3 and 5 . **Table 5-26** and **Table 5-27** shows the section extension format when one set and two sets of “mcScaleReMasks, csf and mcScaleOffset values” are conveyed. Please note that section extension type 5 may be used to convey more than 2 sets of “mcScaleReMasks, csf and mcScaleOffset values” in which case the frame structure is extended in similar fashion, i.e. the reserved bits are added at the end of the section extension to maintain 4-byte alignment.

Table 5-26 : Section Format for Section Extension 5 (one scaler value, modulation compression parameters)

ef	extType = 0x05		1	Octet N
extLen (1 word)			1	N+1
mcScaleReMask[11:4]			1	N+2
mcScaleReMask[3:0]	csf	mcScaleOffset [14:12]	1	N+3
mcScaleOffset [11:4]			1	N+4
mcScaleOffset [3:0]	reserved		1	N+5
reserved			1	N+6
reserved			1	N+7

Table 5-27 : Section Format for Section Extension 5 (two scaler values, modulation compression parameters)

ef	extType = 0x05		1	Octet N
extLen (1 word)			1	N+1
mcScaleReMask[11:4]			1	N+2
mcScaleReMask[3:0]	csf	mcScaleOffset [14:12]	1	N+3
mcScaleOffset [11:4]			1	N+4
mcScaleOffset [3:0]	mcScaleReMask[11:8]		1	N+5

mcScaleReMask[7:0]		1	N+6
csf	mcScaleOffset [14:8]	1	N+7
mcScaleOffset [7:0]		1	N+8
reserved		1	N+9
reserved		1	N+10
reserved		1	M+11

5.4.7.5.1 mcScaleReMask (modulation compression power scale RE mask)

Description: This parameter defines the Resource Element (RE) mask to indicate the position of RE with same scaling and modulation type within a PRB. Each bit setting in the mcScaleReMask indicates if the mcScaleOffset and csf fields are applicable to the RE sent in U-Plane messages or not (0=not applicable; 1=applicable).

Note that different REs in a PRB may be indicated by different invocations of mcScaleReMask within extension field type 5. If any RE in a PRB is never pointed to by a mcScaleReMask (but other REs in that PRB are), the "missing" RE should be considered to represent not populated REs (e.g. no user data to transmit).

There is a relationship between the mcScaleReMask values and the section's reMask: no bit in any of the mcScaleReMasks should be set (=1) in a position where the reMask has a zero, and every reMask bit that is set (=1) should have exactly one bit =1 in one of the mcScaleReMasks. If these rules are violated, the O-RU's reaction is undefined.

Value range: {0000 0000 0000b-1111 1111 1111b}.

Type: unsigned integer (bit mask).

Field length: 12 bits.

Default Value: 1111 1111 1111b (all REs in the block applicable).

5.4.7.5.2 csf (constellation shift flag)

Description: refer to section 5.4.7.4.1

5.4.7.5.3 mcScaleOffset (scaling value for modulation compression)

Description: This parameter is the scale factor to apply to the unshifted constellation points during decompression. It is a fractional floating-point value having an unsigned but negative 4-bit exponent and an unsigned fractional 11-bit mantissa.

Value range: {0 through +(1-2^{-11}) }.

Type: unsigned integer.

$$"mantissa" = \sum_{k=0}^{10} mcScaleOffset[k] \cdot 2^{k-11}$$

$$"exponent" = \sum_{k=11}^{14} mcScaleOffset[k] \cdot 2^{k-10}$$

"exponent" is the most significant 4 bits of the 15-bit mcScaleOffset field and "mantissa" is the least-significant 11 bits of the mcScaleOffset field. "mcScaleOffset[k] refers to the k^{th} bit of the mcScaleOffset field. Therefore, the actual value of mcScaleOffset is:

$$mcScaleOffset = mantissa \cdot 2^{-exponent}$$

Field length: 15 bits.

Chapter 6 U-plane Protocol

6.1 General

6.1.1 U-plane Transport

Either eCPRI or IEEE 1914.3 is used as an encapsulation mechanism for the user-plane messages. Due to the nature of these messages (very strict delay constraints), it is assumed that message acknowledgements are not possible. Likewise, different data flows may be used for the User-Plane and Control-Plane messages.

6.1.2 U-plane Data Compression

U-Plane IQ data, (both DL and UL) including user data, PRACH and control channels may be transmitted in compressed format. There are several envisioned compression methods including an “uncompressed” format. The method of compression is variable based on sectionId but is constant for every section in a single C-Plane message. Likewise, the I and Q bitwidth of the compressed samples is variable based on sectionId but is constant for every section in a single C-Plane message. The block compression methods are performed on Physical Resource Block (PRB) basis (i.e. 12 x Resource Elements per PRB). More specific details on the supported compression methods may be found in Annex A.

An optional capability configures static IQ format and compression method, in which case the M-Plane defines the static IQ format (bitwidth) and compression method. In this case there are four fewer bytes in the U-Plane section header, at the cost of less flexibility. For static IQ data format and compression method, there are actually **four** sets of data formats and compression methods: for LTE DL, LTE UL, NR DL and NR UL. This allows the taking advantage of the static format while still allowing some compression flexibility e.g. allowing modulation compression on the DL (which only applies to DL) and some other compression method on the UL

6.1.3 Digital Power Scaling

6.1.3.1 Definition of IQ Power in dBFS

IQ power level in dBFS (dB full scale) is a logarithmic representation of the power level for an IQ sample carried over the digital interface. IQ power level in dBFS is proportional to logarithm of I^2+Q^2:

IQ power level [dBFS] = $10\cdot\log_{10}(I^2+Q^2) - 10\cdot\log_{10}(FS)$

where FS = $\max(I^2) = \max(Q^2) = \max(I^2+Q^2)$ with max over all IQ values that can be represented by interface

For frequency domain IQ data, 0 dBFS is the maximum power level which can be carried by one subcarrier. The smallest non-zero IQ power level is defined by the interface resolution.

It is expected that an O-RU will normalize any received DL value to its internal representation of full scale so that a 0 dBFS can be properly handled.

Example:

I=min I, Q=0

With 9bit mantissa 2’s complement + 4bit exponent compression: $\min I = -256\cdot2^{15} = -2^{23}$ →

0 dBFS ⇔ average(I^2+Q^2) = 2^{46}

Interface resolution ⇔ $1/2^{46}$ ⇔ -138.47dBFs

6.1.3.2 Definition of Gain over Fronthaul Interface

The gain of an array defines the relation between the levels of a test signal seen at its input and output, also called digital power scaling. The gain of an array can be calculated from the gain of one element of the array while assuming all elements have same gain**. Figure 6-1** depicts the gain relations between the digital interface and the RF reference plane to an assumed lossless antenna (i.e., antenna insertion losses are counted as part of the gain in both DL and UL direction).

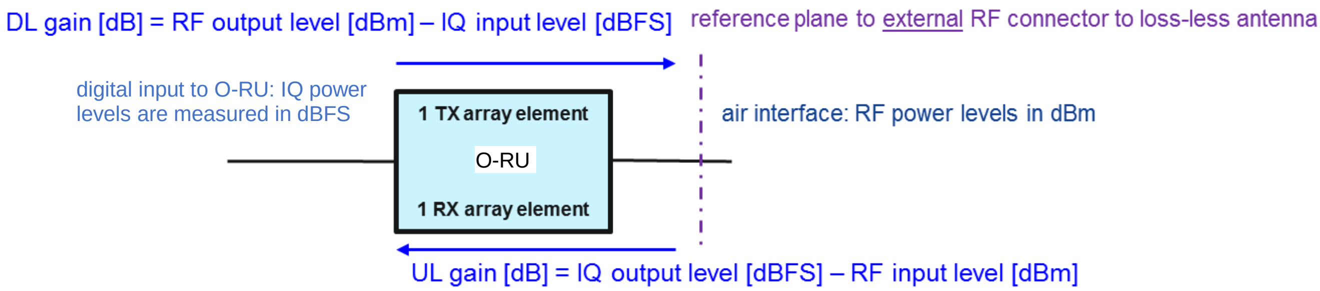


Figure 6-1 : DL and UL Gain over fronthaul interface

Both DL and UL gains in [dB] are described by the following relations

DL gain [dB] = RF output level [dBm] – IQ input level [dBFS]

UL gain [dB] = IQ output level [dBFS] – RF input level [dBm]

Where:

‘RF output level [dBm]’ and ‘RF input level [dBm]’ are RF signal level in dBm at reference plane to an assumed lossless antenna when the antenna is integrated inside the O-RU, or at TAB connector if antenna is not integrated inside the O-RU. The rms output power [dBm] is measured per array element as Total Radiated Power TRP (i.e., excluding any antenna directivity gain). The rms input power [dBm] is measured per array element after including the antenna directivity gain. Note that the antenna directivity gain resuts from the array element directivity + $10\cdot\log_{10}$(number of elements). Hence, when evaluating the array element gain, the antenna directivity is reduced to the array element directivity.

‘IQ input level [dBFS]’ and ‘IQ output level [dBFS]’ are $10\cdot\log_{10}(\text{average}(I^2+Q^2))$ normalized such that 0dBFS is the maximum nominal (r.m.s.) power level which can be achieved with a constant IQ signal with arbitrary phase (i.e., for frequency-domain IQ signal for **one sub-carrier,** constant over time).

In O-RUs supporting beamforming, the actual DL gain and UL gain of the array element can be impacted by the gain level of beam weight used and which can change dynamically during operation. **Such dependency on the beam pattern shall be excluded when describing any configurable gain or reported gain (as O-RU capability).**

For an exact definition of DL and UL gain, the respective input test signals need to be declared by the O-RU vendor; this is not in scope of this specification. [For information only: This can be e.g., CW signal at carrier center or test signals/reference signals defined by 3GPP; for DL gain those being defined to test accuracy of the maximum output power (3GPP TS36.141 section 6.2→ 38.141) and for UL gain those being defined to test absolute accuracy of Received Interference Power (3GPP TS 36.133 section 10.1.1 → 38.133)]. In **Figure 6-2** the details of DL gain are described. For every tx-array, O-RU reports (as capability) the maximal configurable DL gain of **one element of the array** (alternatively the nominal power per tx-array element mapped to 0dBFS). In addition, the O-DU configures over the M-Plane the DL gain to be used per tx-array element for a certain carrier configuration.

The values for the max configurable DL gain and configured DL gain assume:

- No power loss/gain due to beamforming weights;
- all available DL power can be allocated to one array carrier in a single eAxC;

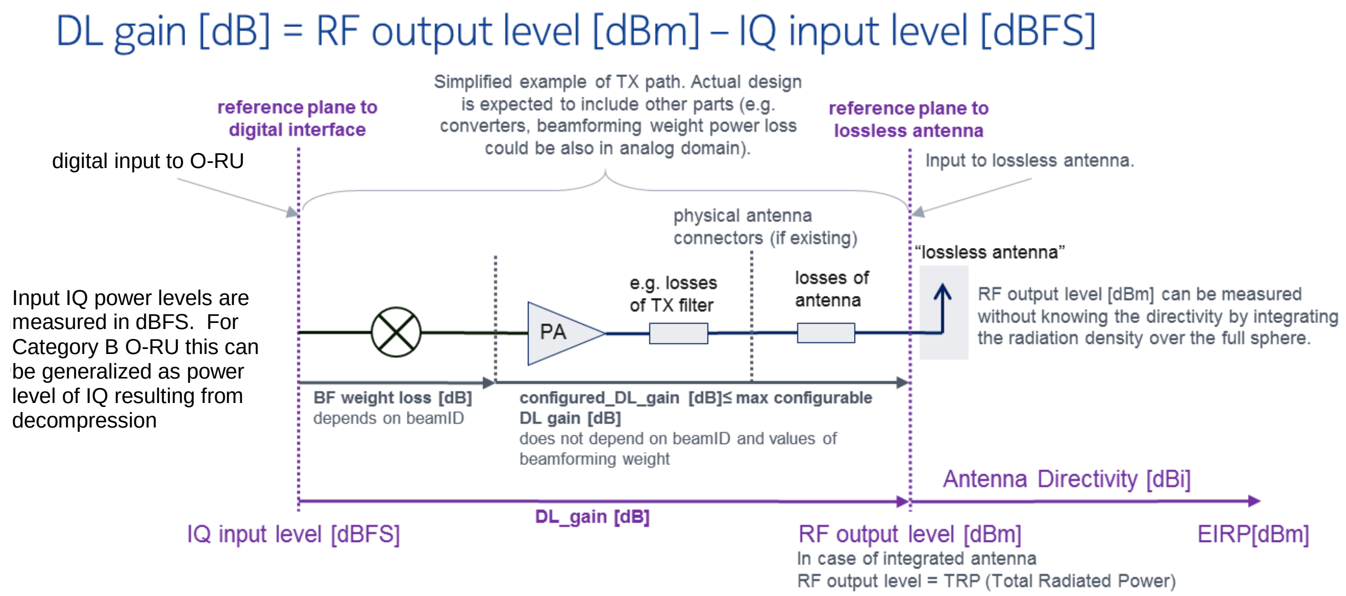


Figure 6-2 : Details of DL Gain

In **Figure 6-3** the details of UL gain are described. For every rx-array, O-RU reports (as capability) an UL gain_correction_range in dB of **one element of the array (applicable to all the elements of the array)**. The gain_correction_range is signed, has a max and min value, and a step size. It shall be allowed to report at least a range of ±20dB and a resolution of 0.01dB. In addition, the O-DU configures over the M-Plane a gain_correction value to be used per rx-array element for a certain carrier configuration. The O-RU can then configure its internal UL gain of the rx-array element for that carrier (rx-array carrier element) if the IQ compression method is configured as static. In case the compression method is dynamic, the internal gain of the O-RU will be dynamic and depending on compression information received in realtime over the C-Plane.

The values for the configured UL gain assume:

- No power loss/gain due to beamforming weights.

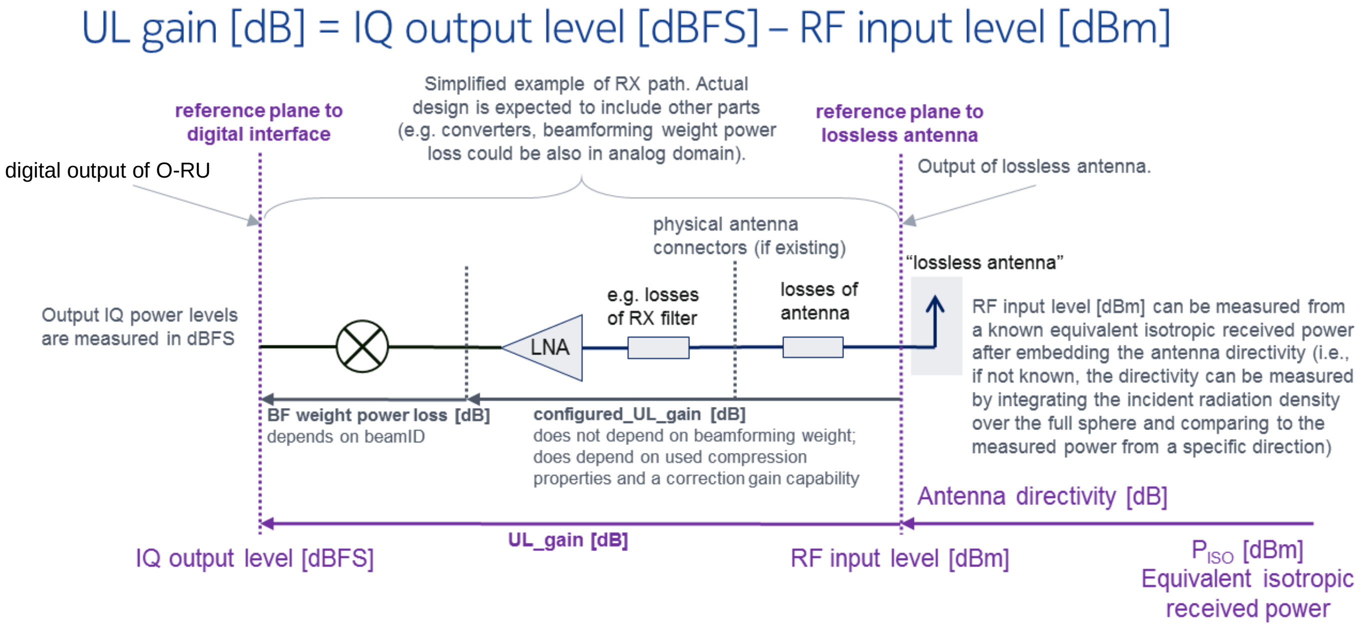


Figure 6-3 : Details of UL Gain

6.1.3.2.1 DL Gain Guideline

The DL gain of a TX array carrier element must be provided as part of the O-RU carrier set-up procedure. The configured (by the O-DU over the M-Plane) DL gain of a TX array carrier element (i.e. "TX array carrier element'' refers to the TX array element serving the respective carrier configured on the respective TX array) is defined by mapping 0dBFs carried over any subcarrier (of the carrier) on the digital interface to the maximum RMS power the TX array carrier element is supposed to output at the assumed-lossless antenna connector port (i.e., including all insertion losses). In single-carrier operation, the configured DL gain applies to the carrier. In multiple-carrier operation, the configured DL gain should be lower accommodating the number of multiple carriers and their bandwidths. It is the responsibility of the O-DU to scale down the level of the signals at the interface to avoid saturation in the O-RU for single or multiple-carrier configurations.

Configured DL_gain [in dB] ~≤ maximum TX power per array element [in dBm], valid for each individual spatial stream served under this TX array carrier element.

NOTE 1: the configured DL gain of a TX array carrier is equivalent to mapping 0dBFs carried over any subcarrier (of the carrier) on the digital interface to the maximum RMS power the TX array carrier element is supposed to output + $10 \cdot \log_{10}$ (the number of array elements).

NOTE 2: for a category A O-RU, the tx-array output rms power is measured over 1 polarization

NOTE 3: for a category B O-RU, the tx-array output rms power shall be measured over all polarizations addressed by the respective precoding operations

6.1.3.2.2 UL Gain Definition

The UL gain or scaling of an rx-array carrier element (i.e. "rx-array carrier element'' refers to the rx-array element serving the respective carrier configured on the respective rx-array) is defined by mapping -152dBm at the assumed-lossless antenna port to the smallest power level an IQ sample can carry over the digital interface (i.e., average(I^2+Q^2) = 1) while considering a configured gain_correction value and the IQ compression properties to be used; valid for each individual spatial stream received by an rx-array carrier element and for any carrier. In addition, in order to avoid saturation over the interface when beamforming is used over the rx-array carrier, the largest power level that can be received at the assumed-lossless antenna port by the rx-array carrier element is equivalent to 0dBFS - $10 \cdot \log_{10}$(number of array elements). The configured gain-correction allows to adjust the level of the smallest & largest receivable power.

The following principles apply:

1. the UL gain of an rx-array carrier is equal to the UL gain of the rx-array carrier element, and all rx-array carrier elements are considered to have the same gain
2. the minimum and maximum power levels that can be received by the rx-array carrier are scaled by $+10 \cdot \log_{10}$(number of array elements) from the respective values of the rx-array carrier element (i.e. when signal is received with equal levels from all the elements of the rx array meaning no tapering used and all rx elements have same design properties). In case an rx-array has only one rx-array element, then the scale factor is unity. The reason behind this principle is to allow applying tapering (beam ID dependent) to an rx-array without impacting the quality of the IQ data being transferred over the interface (i.e., in a worst case scenario, the signal is received from only one rx-array element from the rx-array).
3. assumed-lossless antenna port means antenna losses are considered part of the unit under test.
4. the -152dBm is considered as the smallest level that can be measured by a narrow subcarrier of 1.25kHz (i.e., CW tone) for a system with 3dB of equivalent noise figure and without being impacted by the interface noise (i.e. 20dB margin considered).
5. Configured_UL_gain [in dB] = Interface resolution [dBFs] - (-152 dBm) + gain_correction [dB]; valid for each received individual spatial stream (i.e. configured_UL_gain is configured by O-RU and not by the O-DU).

The UL gain depends on the digital interface resolution [dBFS] representing the smallest level that can be used. The interface resolution depends on the compression scheme which can be static or dynamic. For this reason, when multiple compression methods or IQ bitwidths are used for data streams received from an rx-array carrier element, the configured gain must accommodate all the intended compression methods and IQ bitwidths.

For example, when using block floating point compression, the interface resolution is defined as

Interface resolution [dBFs]=$-20 \times \log_{10}\left(2^{Mantissa_{bits}-1} \times 2^{2^{Exponent}-1}\right)$

It is assumed that the gain_correction is 0dB by default, unless a different value is needed due to special circumstances (details beyond the scope of this specification).

The gain_correction is configured as one value per rx array carrier and does not change once an array carrier is activated. The O-RU applies a configured value of gain_correction regardless of used compression.

Example with rx-array formed by 10 array elements and gain_correction of 0dB:

9bit mantissa 2's complement + 4bit exponent → Interface resolution=-138.5dBFS

Default UL gain [dB]=-138.5dBFs + 152dBm +0dB =13.5dB

The UL gain and power scaling are summarized in **Table 6-1.**

Table 6-1 : Example of UL gain and power scaling for an rx array with 10 elements and for block floating point compression based on 9-bit mantissa and 4-bit exponent

Interface resolution -138.5dBFS	*rx-array carrier element*	*Rx-array carrier with 10 elements*
UL Gain	*13.5dB*	*13.5dB*
Lowest received power level without degradation due to interface (excluding losses due to beam weights)	*-152dBm → -138.5dBFS*	*-142dBm → -128.5dBFS*
Largest possible received power level	*-23.5dBm → -10dBFS*	*-13.5dBm → 0dBFS*

6.1.3.3 TX Power Budget Guideline for Category A O-RUs

This section describes a guideline for handling of power budget in a category A O-RU. In general, care must be taken to avoid exceeding the maximum rms power rating of a tx-array element.

In this section, $m_{a,k}$ is the maximum rms power rating (in W) of tx-array element *k* of a tx-array *a* with *K* elements. For simplicity it is assumed all *K* elements of the array have the same maximum rms power rating (i.e., for every array *a* and every *k* and *k´*, $m_{a,k} = m_{a,k'}$). **For an O-RU with tx-array *a*, the maximum rms power rating of tx-array element shall be reported as read-only parameter. This will be a common value for all array elements of the tx-array *a*.**

NOTE 1: maximum-power-rating can be reported as form of gain when mapped to 0dBFS.

The maximum rms power rating of array can be derived from $m_{a,k}$ by scaling linearly with the number of elements *K* (i.e., $m_a=K\cdot m_{a,k}$)

In addition, let $g_{c,a,k}$ be the gain (in dB) configured for tx-array element k of tx-array a for array carrier c. It is considered that the configured gain is same to every element (i.e., for every tx-array carrier c, every tx-array a and every element k and $k´$, $g_{c,a,k} = g_{c,a,k'}$). **Hence, the O-DU should configure the O-RU by providing a value for $g_{c,a,k}$ in parameter 'gain' of tx-array-carrier.**

The total gain of array carrier *c* served on array *a* can be derived from the configured gain $g_{c,a,k}$ as: $g_{c,a}$ [dB]= $g_{c,a,k}$ [dB]+$10\cdot\log_{10}(K)$.

Hence, the power $p_{c,a,k}$ of array carrier *c* served on tx-array element *k* of array *a* can be derived as $p_{c,a,k}$ [dBm]= $g_{c,a,k}$ [dB]+0dBFS. As a result, the total power of $p_{c,a}$ of array carrier *c* served on tx-array *a* is derived as $p_{c,a}$ [dB]= $p_{c,a,k}$ [dB]+$10\cdot\log_{10}(K)$. Finally, the total power $p_{a.k}$ used on tx-array element *k* of tx-array *a* serving all configured carriers is derived as

$$p_{a,k} = 10 \cdot \log_{10} \sum_{c} 10^{\frac{p_{c,a,k}}{10}}.$$

In scenarios where a tx-array element k is shared between multiple tx-arrays, the O-DU shall ensure that the configured gains $g_{c,a,k}$ are constrained by:

$$\sum_{a} \sum_{c} 10^{\frac{g_{c,a,k}+0dBFS}{10}} \le 1000 \cdot m_k$$

where $m_k = \min_a(m_{a,k})$. The index a spans over every tx-array a that shares array element k and has array carrier configured. The summing over c includes every array carrier c that is configured for tx-array a.

NOTE 2: in general $m_{a,k}$ can be different for different values of tx-array a. However, tx-arrays not used in a given configuration do not contribute to the above constraint.

For every configured array carrier c over a tx-array a, the O-DU shall ensure that the input power levels of all resource elements used over all eAxCs x are constrained by:

$$\sum_x \sum_n 10^{\frac{|RE_{n,x,c,a}|^2}{10}} \leq 1$$

Where $|RE_{n,x,c,a}|^2$ represents that input power level in dBFS of an RE n of an eAxC x of array carrier c configured over tx-array a. The summing over x includes every eAxC x of array carrier c that is used simultaneously in DL. The summing over n includes every RE of eAxC x that is used simultaneously in DL.

The final constraint is related to the beamforming weights to be used in order to make the power budget practical from an implementation perspective. **For any beamforming weight $w_{k,x}$ (a complex number) to be used with array element k and eAxC x, the entity controlling the generation of the weight (i.e., O-DU or O-RU) shall ensure that the $|w_{k,x}|^2 \leq 1$**

6.2 Elementary Procedures

6.2.1 IQ Data Transfer procedure

This procedure is used to transfer frequency domain IQ data samples between the O-DU and O-RU. Data is transmitted symbol by symbol as U-Plane messages. The data-associated control information is typically sent every slot (or for LTE, TTI) in a different data flow with its own header and encapsulated payload (C-Plane messages). An overview of the logic for transmission of both the data-associated control information and IQ data is depicted in **Figure 6-4**

.

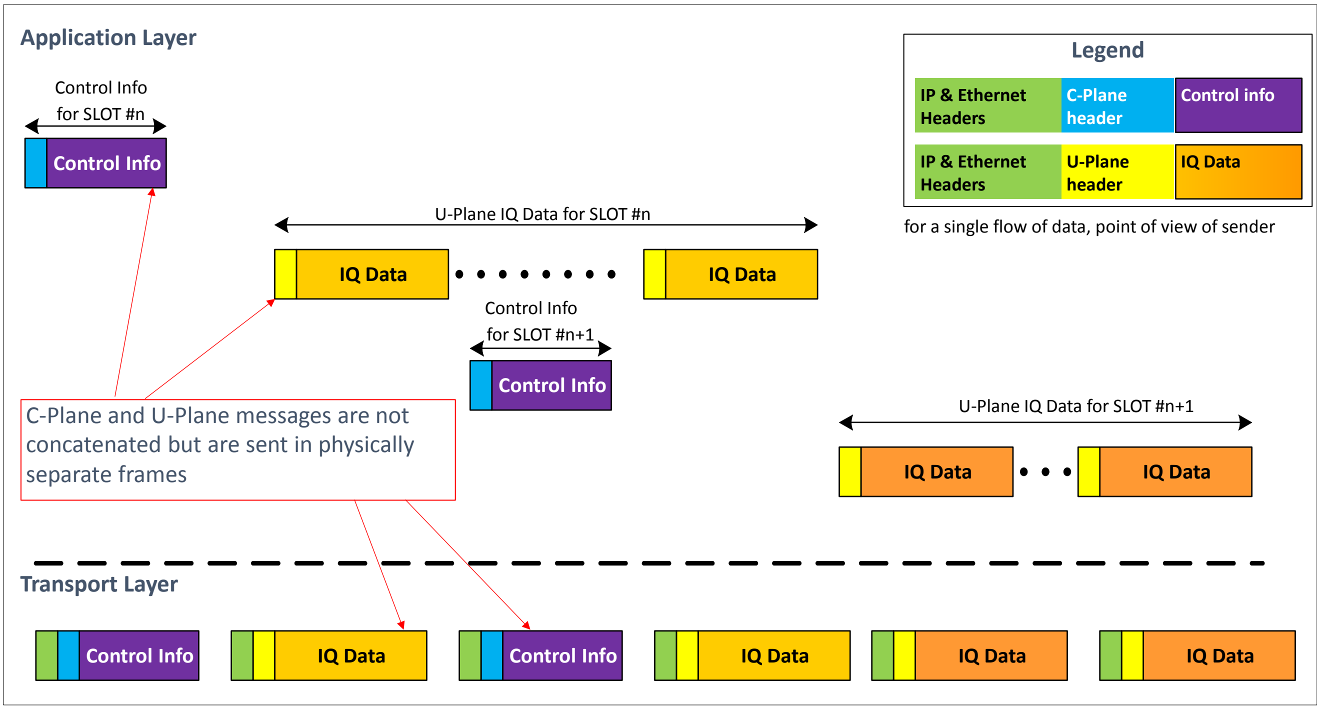


Figure 6-4 : DL IQ data transfer overview

The IQ transfer procedure is comprised of two types of messages:

- Message containing scheduling and beamforming commands information (i.e. data-associated control)
- Message containing frequency domain IQ samples

The data-associated control information is bundled in messages within a single data flow (albeit possibly in multiple C-Plane messages). However, control information for UL and DL is sent separately (See **Figure 6-5**). As shown in the figure, the C-Plane messages for a given slot are exchanged between the O-DU and O-RU are followed by the U-Plane data for that slot, one symbol at a time, after which the C-Plane messages for the next slot are exchanged between the O-DU and the O-RU followed by the U-Plane data for that next slot. There is a limited intention to support "look-ahead" wherein the C-Plane data for a future slot may be sent prior to the actual U-Plane data, this is governed by delay management parameters such as T1a_max_cp_dl, T1a_min_cp_dl and T2a_min_cp_dl. **Figure 6-5** shows the case where control information for slot #n+1 is sent prior to U-Plane data for slot #n. A greater look-ahead capability would result in a need for a larger O-RU buffer to store look-ahead C-Plane messages so is not supported.

Also shown in the figure is the fact that there is a certain period of time between the O-RU's receipt of C-Plane messages for a symbol and the need for the O-RU to process U-Plane data for that symbol. In particular, in the DL there is a period of time, "tadv_cp_dl" which provides for some number of microseconds for the O-RU to e.g. update beamforming weights prior to DL data arriving from the O-DU. In the UL there is a period of time, "t2a_min_cp_ul" between the O-RU's receiving the C-Plane messages governing the processing of UL data and the receipt of UL signals at the O-RU's antennas. These time intervals, when combined with network delays and other processing latencies, result in the RAN's HARQ loop to be closed allowing feedback in the air interface processing.

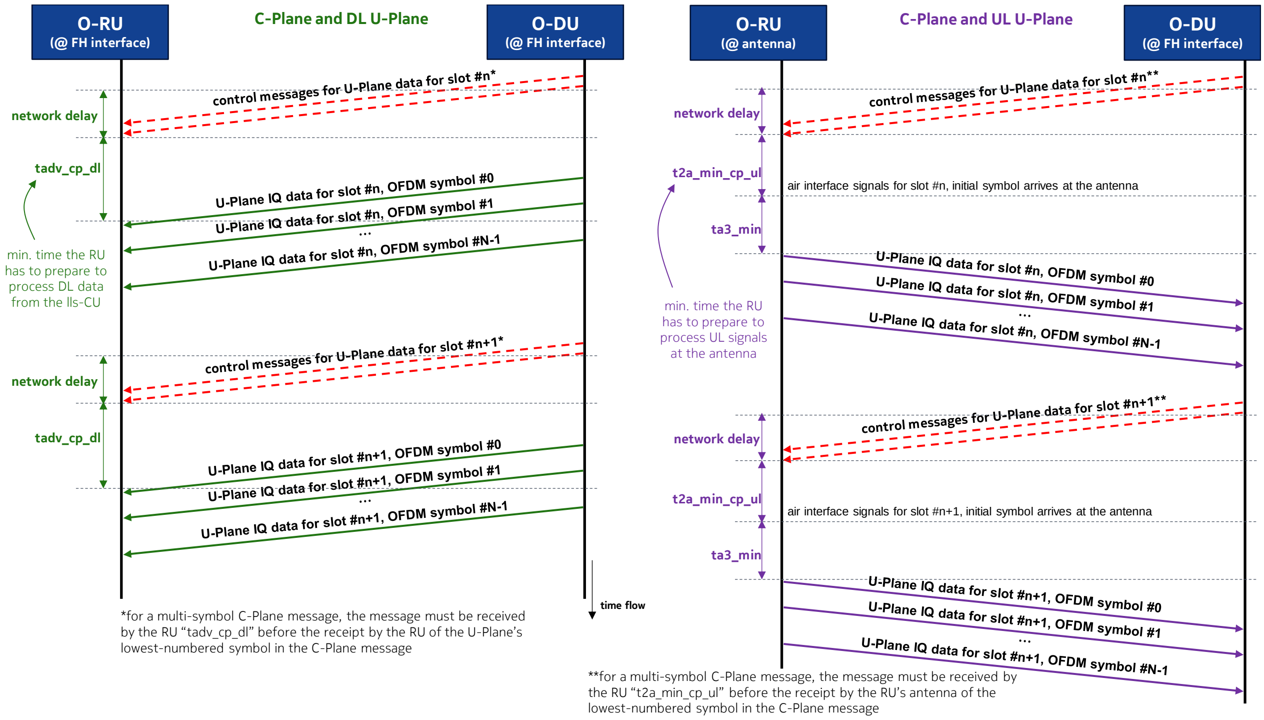


Figure 6-5 : C-Plane and U-Plane message transfer procedure (DL & UL shown)

Regardless of the level of compression, user-plane data may exceed the maximum packet size (e.g., MTU of 1500 Bytes for regular not Jumbo frames) even in case of one symbol of granularity. Thus, IQ data for each symbol may require packetization over multiple packets.

User plane messages are sent as resource blocks ("PRBs") and the data for each PRB shall start on a byte boundary. If due to the IQ bit-width being used the natural end of the data in the PRB does not fall on a byte boundary, then zero bits shall be appended until a byte boundary is reached.

PRACH data is handled similarly, such that the PRACH REs are packaged into 12-RE blocks analogous with data PRBs. Each 12-RE block shall be padded with zero bits to end on a byte boundary. In cases when there is not an even factor of 12 REs in the PRACH data (e.g. 839 PRACH REs), added zero-value REs shall be added to fill out the block to 12 REs, and then if needed zero-padding will then be appended to reach a byte boundary.

6.3 Elements for the U-plane Protocol

6.3.1 General

U-Plane messages are encapsulated using a two-layered header approach. The first layer consists of an eCPRI or IEEE 1914.3 common header, including fields used to indicate the message type, while the second layer is an application layer including necessary fields for control and synchronization.

6.3.2 DL/UL Data

A common frame format is used for U-Plane messages consisting of a transport layer and an application layer. The application layer is within the transport payload and consists of a common header for time reference, followed by information and parameters dependent and specific to the Section Type in use. Data from multiple sections of the same value can be lined up one after another within the payload. To minimize packet rate over the interface, transmitter should fill messages with as many subsequent sections as possible. However, data from sections of different Section Types is to be sent via separate messages (i.e. data from different values of Section Type shall not be mixed within a single U-Plane message payload). Furthermore, whenever necessary, stuffing bits are to be added at the end of a sections after (possibly compressed) I and Q sample data to achieve 1-Byte alignment. Note that within a resource block, 12 REs are always sent across the interface. If some REs are meant to be blank some data (perhaps zero value) must be sent anyway because the U-Plane data parser is expecting exactly 12 complex RE values per resource block.

- Transport Layer – see section 3.1.3
- Application Layer
- Section Type “1” Fields (used for most Downlink and Uplink physical radio channels):
  - <u>Common Header Fields</u>
    - **dataDirection** (data direction (gNB Tx/Rx)) field: 1 bit
    - **payloadVersion** (payload version) field: 3 bits
      - Value = “1” shall be set (1st protocol version for payload and time reference format)
    - **filterIndex** (filter index) field: 4 bits
    - **frameId** (frame identifier) field: 8 bits
    - **subframeId** (subframe identifier) field: 4 bits
    - **slotID** (slot identifier) field: 6 bits
    - **symbolId** (symbol identifier) field: 6 bits
  - <u>Section header fields</u>
    - **sectionID** (section identifier) field: 12 bits
    - **rb** (resource block indicator) field: 1 bit
    - **symInc** (symbol number increment command) field: 1 bit
    - **startPrbu** (starting PRB of user plane section) field: 10 bits
    - **numPrbu** (number of contiguous PRBs per data section) field: 8 bits
    - **udCompHdr** (user data compression header) field, not always present: 8 bits
    - **reserved** (reserved for future use) field, only present with udCompHdr: 1 byte

  - <u>PRB fields</u>
    - **udCompParam** (user data compression parameter) field: 8 bits

- ○ **iSample** (in-phase sample) field: 1-16 bits
- ○ **qSample** (quadrature sample) field: 1-16 bits
- ○ Section Type “3” Fields (used for PRACH and mixed-numerology channels):
  - ○ Timing header, section header and PRB fields same as for Section Type “1”
- ○ Section Type “5” Fields (used for UE scheduling information):
  - ○ Timing header, section header and PRB fields same as for Section Type “1”
- ○ Section Type “6” Fields (used for sending channel information for a specific UE ID):
  - ○ Timing header, section header and PRB fields same as for Section Type “1”

Table 6-2 : IQ data frame format

Section Type 1,3 : DL/UL IQ data msgs									
0 (msb)	1	2	3	4	5	6	7 (lsb)	# of bytes	
transport header, see section 3.1.3								8	Octet 1
dataDirection	payloadVersion			filterIndex				1	Octet 9
frameId								1	Octet 10
subframeId				slotId				1	Octet 11
slotId		symbolid						1	Octet 12
sectionId								1	Octet 13
sectionId				rb	symInc	startPrbu		1	Octet 14
startPrbu								1	Octet 15
numPrbu								1	Octet 16
udCompHdr (not always present)								1	Octet 17
reserved (not always present)								1	Octet 18
udCompParam (not always present)								1	Octet 17/19
iSample (1^{st} RE in the PRB)								1*	Octet 18/20
qSample (1^{st} RE in the PRB)								1*	Octet 19/21*
…									
iSample (12^{th} RE in the PRB)								1*	Octet 40/42*
qSample (12^{st} RE in the PRB)								1*	Octet 41/43*
udCompParam (not always present)								1*	Octet 42/44*
iSample (1^{st} RE in the PRB)								1*	Octet 43/45*
qSample (1^{st} RE in the PRB)								1*	Octet 44/46*
…									
iSample (12^{th} RE in the PRB)								1*	Octet 65/67*
qSample (12^{st} RE in the PRB)								1*	Octet 66/68*
…									
sectionId								1	Octet M
sectionId				rb	symInc	startPrbu		1	M+1
startPrbu								1	M+2
numPrbu								1	M+3
udCompHdr (not always present)								1	M+4
reserved (not always present)								1	M+5
udCompParam (not always present)								1	M+4/6
iSample (1^{st} RE in the PRB)								1*	M+5/7
qSample (1^{st} RE in the PRB)								1*	M+6/8*
…									
iSample (12^{th} RE in the PRB)								1*	M+27/29*
qSample (12^{st} RE in the PRB)								1*	M+28/30*
udCompParam (not always present)								1*	M+29/31*

iSample (1st RE in the PRB)	1*	M+30/32*
qSample (1st RE in the PRB)	1*	M+31/33*
…		
iSample (12th RE in the PRB)	1*	M+52/54*
qSample (12st RE in the PRB)	1*	M+53/55*

shading: yellow is transport header, pink is radio application header, others are repeated sections
*Octet count given if the iqWidth = 8 but other iqWidth values are possible

6.3.3 UL/DL Data Coding of Information Elements

See section 3.1.3 for transport header information element details.

6.3.3.1 dataDirection (data direction (gNB Tx/Rx))

See section (5.4.4.1)

6.3.3.2 payloadVersion (payload version)

See section (5.4.4.2)

6.3.3.3 filterIndex (filter index)

See section (5.4.4.3)

6.3.3.4 frameId (frame identifier)

See section (5.4.4.4)

6.3.3.5 subframeId (subframe identifier)

See section (5.4.4.5)

6.3.3.6 slotId (slot identifier)

See section (5.4.4.6)

6.3.3.7 symbolId (symbol identifier)

Description: This parameter identifies a symbol number within a slot.

Value range: {00 0000b-11 1111b}

Type: unsigned integer.

Field length: 6 bits.

6.3.3.8 sectionId (section identifier)

See section (5.4.5.1)

6.3.3.9 rb (resource block indicator)

See section (5.4.5.2)

6.3.3.10 symInc (symbol number increment command)

See section (5.4.5.3)

6.3.3.11 startPrbu (startingPRB of user plane section)

Description: This parameter is the starting PRB of a user plane data section. Values of startPrbu and numPrbu must ensure that data sections must never overlap: a single PRB may only exist within one data section for a given value of eAxC. For one C-Plane message, there may be multiple U-Plane messages associated with it and requiring defining from which PRB the contained IQ data are applicable.

Note: freqOffset affects the frequency span for specific range of PRB numbers. Therefore "must never overlap" must consider the value of freqOffset.

Value range: {00 0000 0000b-11 1111 1111b}.

Type: unsigned integer.

Field length: 10 bits.

6.3.3.12 numPrbu (number of PRBs per user plane section)

Description: This parameter defines the PRBs where the user plane data section is valid. Values of startPrbu and numPrbu must ensure that data sections must never overlap: a single PRB may only exist within one data section for a given value of eAxC.

Value range: {0000 0001b-1111 1111b, 0000 0000b = all PRBs}.

Value 0000 0000b is particularly relevant for NR cases wherein the total number of PRBs may be more than 255.

Type: unsigned integer.

Field length: 8 bits.

6.3.3.13 udCompHdr (user data compression header)

Description: This parameter defines the compression method and IQ bit width for the user data in the data section. This field is absent from U-Plane messages when the static IQ format and compression method is configured via the M-Plane. In this way a single compression method and IQ bit width is provided (per UL and Dl, per LTE and NR) without adding more overhead to U-Plane messages.

Value range: {0000 0000b-1111 1111b}

Bit allocations

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
udIqWidth				udCompMeth				1	Octet 1

Table 6-3 : udIqWidth definition

udIqWidth	Bit width of each I and each Q
0000-1111b	16 for udIqWidth=0, otherwise equals udIqWidth e.g. udIqWidth = 0000b means I and Q are each 16 bits wide; e.g. udIQWidth = 0001b means I and Q are each 1 bit wide; e.g. udIqWidth = 1111b means I and Q are each 15 bits wide

Table 6-4 : udCompMeth definition

udCompMeth	compression method	udIqWidth meaning
0000b	no compression	bitwidth of each uncompressed I and Q value
0001b	block floating point	bitwidth of each I and Q mantissa value
0010b	block scaling	bitwidth of each I and Q scaled value
0011b	μ-law	bitwidth of each compressed I and Q value
0100b	modulation compression	bitwidth of each compressed I and Q value
0100b – 1111b	reserved for future methods	depends on the specific compression method

Type: unsigned integer (concatenated bit fields).

Field length: 8 bits.

6.3.3.14 reserved (reserved for future use)

Description: This parameter provides 1 byte for future definition, should be set to all zeros by the sender and ignored by the receiver. This field is only present when udCompHdr is present, and is absent when the static IQ format and compression method is configured via the M-Plane.

Value range: {0000 0000b-1111 1111b}, but shall be set to all zeros.

Type: unsigned integer.

Field length: 8 bits.

6.3.3.15 udCompParam (user data compression parameter)

Description: This parameter applies to whatever compression method is specified by the associated sectionID's compMeth value.

Value range: {0000 0000b-1111 1111b}.

Bit allocations

<table>
<tr><th>udCompMeth</th><th>0 (msb)</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7 (lsb)</th><th>compParam size</th></tr>
<tr><td>0000b = no compression</td><td colspan="8">absent</td><td>0 octets</td></tr>
<tr><td>0001b = block fl. point</td><td colspan="4">reserved (set to all zeros)</td><td colspan="4">exponent (unsigned)</td><td>1 octet</td></tr>
<tr><td>0010b = block scaling</td><td colspan="8">sblockScaler (unsigned, 1 integer bit, 7 fractional bits)</td><td>1 octet</td></tr>
<tr><td>0011b = μ-law</td><td colspan="4">compBitWidth</td><td colspan="4">compShift</td><td>1 octet</td></tr>
<tr><td>0100b = modulation compr.</td><td colspan="8">absent</td><td>0 octets</td></tr>
<tr><td>0101b – 1111b</td><td colspan="8">reserved (set to all zeros)</td><td>? octets</td></tr>
</table>

Type: variable.

Field length: zero for udCompMeth values 0000b and 0100b, 8 bits for udCompMeth values 0001b, 0010b and 0011b; other udCompMeth values may imply other lengths but will always be an integer number of bytes.

6.3.3.16 iSample (in-phase sample)

Description: This parameter is the In-phase sample value.

Value range: {all zeros – all ones}.

Type: signed integer.

Field length: 1-16 bits.

6.3.3.17 qSample (quadrature sample)

Description: This parameter is the Quadrature sample value.

Value range: {all zeros – all ones}.

Type: signed integer.

Field length: 1-16 bits.

6.3.4 DL Data Precoding

- Section extension '3' is used for C-plane and associated sectionID for U-plane
- O-RU must understand that for this section extension, O-RU should read 12 REs which have CRS reference signals in that PRB.
- O-RU must understand the crsShift and crsReMask field to map appropriately the CRS REs to each antenna port

Table 6-5 : DL Data Precoding Example

<table>
<tr><th>C-plane example
DL Precoding configuration parameters and indications (Section ext "3")</th><th>O-RU Outcome</th></tr>
<tr><td>

Section ext = 3
txScheme='txD'
codeBookIndex= '00000000' (invalid)
numLayer = 4
layerId = '0000' (TxD)
crsReMask= 001001001001 (Assuming MSB..LSB)
crsSymNum=0000
crsShift=0
beamIdAP1
beamIdAP2
beamIdAP3

</td><td>
Based on numLayers, crsShift, crsSymNum and crsReMask bit positions, CRS ports are mapped to the appropriate RE position and rest are left blank.
<table>
<tr><td>CRS (4 Layer)</td><td>If crsSymNum</td><td>crsReMask Bit Position</td></tr>
<tr><td>(crs_Ant0)</td><td>0</td><td>{0,6} + vShift</td></tr>
<tr><td>(crs_Ant1)</td><td>0</td><td>{3,9} + vShift</td></tr>
</table>
</td></tr>
</table>

6.3.5 Data Transfer for Special Cases

Uplink IQ data transfer is covered in Section 6.3.2. In particular, PRACH and other common channels as well as SRS and other reference signal channels use the same frequency domain IQ data packetization as with user data channels (PDSCH, PUSCH).

6.3.5.1 Data Message Mapping and Packetization

See Section 6.3.2.

Chapter 7 Counters and KPIs

7.1 Counters

This chapter provides details of the specific performance counters that are expected to be supported related to the fronthaul interface. Some will be considered mandatory and some optional, and there are differences for the DL versus the UL.

The tables below provide some counter definitions, it is expected there will be more added in future versions of the specification.

Table 7-1 : Common Counters for both DL and UL

#	Counter name	Counter meaning
1	Rx_on_time	Data was received on time (applies to user data reception window)
2	Rx_early	Data was received too early (applies to user data reception window)
3	Rx_late	Data was received too late (applies to user data reception window)
4	Rx_corrupt	Corrupt/Incorrect header packet
5	Rx_pkt_dupl	Duplicated packet
6	Total_msgs_rcvd	Total messages received (on all links)

Table 7-2 : Counters specific to DL

#	Counter name	Counter meaning
1		None identified yet
2		
3		
4		

Table 7-3 : Counters specific to UL

#	Counter name	Counter meaning
1		None identified yet
2		
3		
4		

Chapter 8 Specification Mandatory and Optional Capabilities

8.1 General

This chapter provides details regarding which capabilities within the specification are mandatory and which are optional. The list will in general be different for the O-DU versus the O-RU because in many cases, the O-DU will need to implement multiple options as mandatory to ensure interoperability with O-RUs that have optional capabilities. For example, the ability to support many compression methods may be mandatory in the O-DU while in O-RUs there may be only a single mandatory compression method to allow simplicity in O-RU design (while vendors may enhance their O-RU product offering by implementing some of the optional compression methods).

Table 8-1 : O-RAN-Supported LTE and NR Channels

Physical Channel Support	
LTE DL Channels	PDSCH, PBCH, PCFICH, PDCCH, ePDCCH, MPDCCH, PHICH, CRS, MBSFN RS, UE-RS, DMRS for ePDCCH/MPDCCH, PRS, CSI-RS, PSS, SSS, Discovery RS
LTE UL Channels	PUSCH, PUCCH, DMRS-PUSCH, DMRS-PUCCH, SRS, PRACH (incl. eMTC)
Narrow band IoT DL Channels	NB-DMRS, NB-PDSCH, NB-PBCH, NB-PDCCH, NB-RS, NB-PRS, NB-PSS, NB-SSS
Narrow band IoT UL Channels	NB-PUSCH, NB-PRACH
NR DL Channels	PDSCH, PDCCH, DMRS-PDSCH, PTRS-PDSCH, DMRS-PDCCH, DRMS-PBCH, CSI-RS, PSS, SSS, SS Block/PBCH
NR UL Channels	PUSCH, PUCCH, PRACH, DMRS-PUSCH, PTRS-PUSCH, DMRS-PUCCH, SRS
System Capability Support	
Technologies	LTE TDD, FDD (normal and extended CP) NR TDD, FDD
Channel Bandwidth	LTE: 1.4, 3, 5, 10, 15, 20 MHz
	NR: up to 400MHz
Subcarrier Spacing	LTE: 15kHz, 7.5kHz, 1.25kHz LTE PRACH: 1.25kHz, 7.5kHz NB-IoT PRACH: 3.75kHz
	NR: 15, 30, 60, 120, 240 kHz
	NR Multi Numerology
	NR PRACH: 1.25, 5, 15, 30, 60, 120 kHz
LTE Specific Features	DL Transmission Modes : TM1 - TM10

	UL Transmission Modes : TM1, TM2
	Carrier Aggregation
	eMBMS
	TTI-Bundling
	Semi-Persistent Scheduling (SPS)
	MIMO (SU/MU-MIMO)
	UE TAS (Tx Antenna Selection)
	FeICIC (ABS)
	CoMP (DL/UL), JT
	Short TTI
	eMTC
	NB-IOT (in band/gurad band/standalone)
	License Assisted Access (LAA)
	Sidelink (Proximity Services)
	Dynamic TDD (eIMTA)
	Mission Critical PS-LTE Features (MCPTT, ..)
	Positioning (PRS, OTDOA etc)
	V2X
	Distributed Antenna Sysstem Support
	CBRS Support
	EN-DC
	SSBlock
	BW Part
	Supplimentary UL
	Mini-slot
NR Specific Features	LTE-NR Co-existence
	Analog Beamforming
	Digital Beamforming
	Hybrid Beamforming
Beamforming	O-RU Support for 2 - 256 TRXU
	L2 : Ethernet
Transport	L3 : IPv4, IPv6
	QoS over Fronthaul

Table 8-2 : O-RAN Mandatory and Optional Features

Category	Feature	O-DU Support	O-RU Support	Note
O-RU Category Support	**Support for CAT-A O-RU (up to 8 spatial streams)**	**Mandatory**	NA	
	Support for CAT-A O-RU (> 8 spatial streams)	Optional	NA	
	Support for CAT-B O-RU (precoding in O-RU)	**Mandatory**	NA	
Beamforming	**Beam Index based**	**Mandatory**	**Mandatory**	Applies to UE specific BF for all O-RUs

	Real-time BF Weights	**Mandatory**	**Mandatory**	Applies to O-RUs capable of BF using RT Weights
	Real-Time Beamforming Attributes	Optional	Optional	
	UE Channel Info	Optional	Optional	
Bandwidth Saving	**Programmable static-bit-width Fixed Point IQ**	**Mandatory**	**Mandatory**	
	Real-time variable-bit-width	Optional	Optional	
	Compressed IQ			
	- Block floating point compression	Optional	Optional	
	- Block scaling compression	Optional	Optional	
	- µ-law compression	Optional	Optional	
	- modulation compression	Optional	Optional	
	- beamspace compression	Optional	Optional	
	Variable Bit Width per Channel (per data section)	Optional	Optional	
	Static configuration of U-Plane IQ format and compression header	Optional	Optional	
	Use of "symInc" flag to allow multiple symbols in a C-Plane section	Optional	Optional	
Energy Savings	Transmission blanking	Optional	Optional	
O-DU - O-RU Timing	**Pre-configured Transport Delay Method**	**Mandatory**	**Mandatory**	
	Measured Transport Method (eCPRI Msg 5)	Optional	Optional	
Synchronization	**G.8275.1**	**NA**	**Mandatory**	
	G.8275.2	NA	Optional	
	GNSS based sync	NA	Optional	
	SyncE	NA	Optional	
Transport Features	**L2 : Ethernet**	**Mandatory**	**Mandatory**	
	L3 : IPv4, IPv6 (CUS Plane)	Optional	Optional	
	QoS over Fronthaul	**Mandatory**	**Mandatory**	
	Prioritization of different U-Plane traffic types	Optional	Optional	
	Support of Jumbo Ethernet frames	Optional	Optional	
	eCPRI	**Mandatory**	**Mandatory**	
	support of eCPRI concatenation	Optional	Optional	
	IEEE 1914.3	Optional	Optional	
	Application fragmentation	**Mandatory**	**Mandatory**	C-Plane and U-Plane
	Transport fragmentation	Optional	Optional	U-Plane (not C-Plane)
Other features	LAA LBT O-DU Congestion Window mgmt	**Mandatory**	**Mandatory**	Only for RUS supporting LAA
	LAA LBT O-RU Congestion Window mgmt	Optional	Optional	

Chapter 9 S-Plane Protocol

9.1 General

9.1.1 Overview

Time and frequency synchronization can be distributed to the O-DU and O-RU in different manner. However, synchronization accuracy is mostly impacted by implementation (e.g., timestamping near the interfaces, number of hops) than by the technology itself. The following synchronization options are available over an Ethernet network:

- Frequency synchronization where clocks are aligned in frequency
- Phase synchronization where clocks are aligned in phase
- Time synchronization where clocks are aligned to a common base time

Together the above parameters define a profile for the network, requiring a set of features and option selections for bridges and end stations operation. Further, the profile also states the conformance requirements for supporting equipment and user applications.

This edition of the document considers frequency, phase and time synchronization of all the network elements (O-DUs, intermediate switches and O-RUs). Frequency-only configurations (like LTE FDD or 5G FDD) are For Further Study.

This edition of the document considers macro BTS O-DUs and not small cells that have O-DU and O-RU in the same box and therefore no need for fronthaul link.

9.2 Synchronization Baseline

9.2.1 List of Reference Documents

See section 1.2.1.

9.2.2 Clock Model and Synchronization Topology

Different O-RAN synchronization topologies are necessary to address different deployment market need. The following 4 topology configurations are considered by O-RAN as compliant topologies. A configuration label is used for easier reference through this specification:

- Configuration C1: network timing from O-DU to O-RU via point-to-point topology between central site and remote site
- Configuration C2: network timing from O-DU to O-RU between central sites and remote sites. One or more Ethernet switches are allowed in the fronthaul network. Interconnection among switches and fabric topology (for example mesh, ring, tree, spur etc.) are deployment decisions which are out of the scope of this O-RAN specification.
- Configuration C3: network timing from PRTC/T-GM to both O-DU and O-RU between central sites and remote sites. One or more Ethernet switches are allowed in the fronthaul network. Interconnection among switches and fabric topology (for example mesh, ring, tree, spur etc.) are deployment decisions which are out of the scope of this O-RAN specification
- Configuration C4: Local time source traceable to a PRTC (typically a GNSS receiver) providing the timing to O-RU

Note applying to both C2 and C3: Only current types and classes (PRTC/T-GM, T-BC, T-TC) under G.8271.1 are covered by this document. Consideration of upcoming clock types and classes inside ITU-T G.8271.1 new revision, or outside this recommendation, is For Future Study.

9.2.2.1 Topology configuration C1 and C2 Synchronization

Configuration C1 is based on point-to-point connection between O-DU and O-RU using network timing option. As shown in **Figure 9-1** below, it is basically the simplest topology for network timing option, where O-DU directly synchronizes O-RU.

Configuration C2 is similar to C1 with O-DU acting as PTP and SyncE master to distribute network timing toward O-RU. One or more Ethernet switches are allowed between the central site (hosting O-DUs) and the remote sites (hosting O-RUs). The allowed number of switches in the PTP and SyncE path (between PTP and SyncE master and slaves) is limited by frequency and time error contributions by all clocks in the chain.

With Full Timing Support and Annex H detailed budget split, the allowed network noise limit in the budget can be met by 2 T-BC Class B switches as shown in **Figure 9-1** below. Additional T-BC switches may be allowed if total noise limit can be met. The synchronization SYNCE+PTP master is located at the O-DU. Further, all Ethernet switches in the fronthaul function as Telecom Boundary Clocks as specified by ITU-T G.8273.2. T-TC switch is also allowed as T-BC replacement with the same expectation based on G.8271.1.

With Partal Timing Support, non-T-BC switches may also be deployed. Further investigation is required to specify appropriate frequency and timing budgets and network configuration to ensure 4G and NR TAE requirements as described in **Table 9-3** and 3GPP frequency accuracy requirements can be met.

Interconnection among switches and fabric topology (for example mesh, ring, tree, spur etc.) are deployment decisions which are out of the scope of this O-RAN spec.

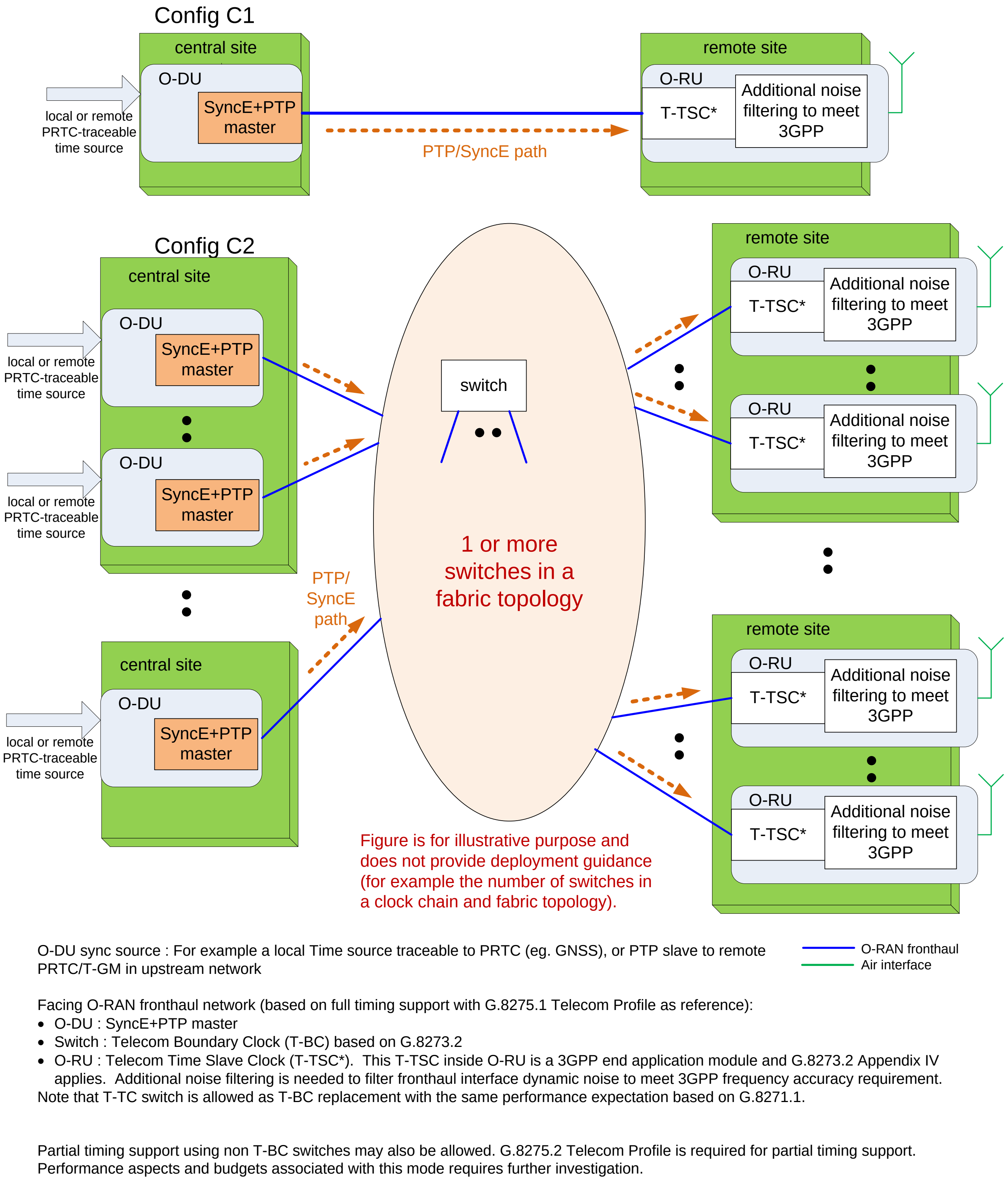


Figure 9-1 : Configurations C1 and C2 synchronization

Notes for O-DU:

O-DU acts as SYNCE and PTP master towards the fronthaul interface, but there are different possible sub-configurations based on the O-DU sync source:

- if O-DU sync source is from a local time source traceable to a PRTC (typically a GNSS receiver), it may act as a T-GM, or as a specific PLL with higher jitter and wander filtering capability.

- if O-DU sync source is from a remote PRTC through a network (typically PTP, with or without SYNCE, from upstream backhaul), O-DU acts as PTP (with optional SyncE) slave toward the upstream network. Two sub-configurations are possible:
    - If the PTP profile used in the upstream network is different (typically ITU-T G.8275.2) from the fronthaul one (which is typically ITU-T G.8275.1), then the O-DU acts as an InterWorkingFunction clock to bridge between the profiles, as defined in ITU-T G.8271.2 (but no clock specification exists for such IWF). This is For Further Study.
    - If the PTP profile used in the upstream network is same as the fronthaul one (typically ITU-T G.8275.1), then the O-DU may act as a combined PTP Slave and Master with higher jitter and wander filtering capability. Note: O-DU acting as a ITU-T G.8273.2 T-BC clock does not provide enough wander cleaning to guarantee the 15 ppb limit and is therefore outside the scope of C1/C2.

When multiple O-DU are connected to the central site aggregation switch supporting T-BC in configuration C2, only one active Master shall be in the clock chain. One O-DU shall serve as active GrandMaster to all O-RUs, including the ones controlled by other O-DUs over the M-plane. If Master redundancy is needed, another O-DU shall serve as backup Master.

Notes for O-RU:

All O-RUs are either:

- An end application including an application-specific O-RU slave clock which can be compliant to G.8273.2 T-TSC regarding TE budgeting but follows additional requirements to ensure 3GPP air interface compliance.
  T-TSC is per IEEE 802.1CM interface condition Case 1.
- An end application connected to an external T-TSC per IEEE 802.1CM interface condition Case 2. This case may be excluded by O-RAN since the O-RU generally does not provide a separate PPS/ToD interface for external T-TSC connection, and there may be some performance concerns about the 1pps distribution interface compared to PTP over Ethernet.

9.2.2.2 Topology configuration C3 Synchronization

Configuration C3 is similar to C2 except frequency and time distribution is made by the fronthaul network itself (not by the O-DU). That means that one or more PRTC/T-GM (acting as SYNCE+PTP master) is implemented in the fronthaul network to distribute network timing toward O-DU and O-RU. One or more Ethernet switches are allowed between the central site (hosting O-DUs) and the remote sites (hosting O-RUs). The permitted number switches in the PTP and SyncE path (between PTP master and SyncE master and slaves) is limited by frequency and time error contributions by all clocks in the chain.

With Full Timing Support, all Ethernet switches in the fronthaul function as T-BC as specified by ITU-T G.8273.2. T-TC switch is also allowed as T-BC replacement with the same expectation based on G.8271.1.

Partial Timing Support using non T-BC switches may also be allowed. G.8275.2 Telecom Profile is required for Partial Timing Support. Performance aspects and budgets associated with this mode requires further investigation.

Interconnection among switches and fabric topology (for example mesh, ring, tree, spur etc.) are deployment decisions which are out of the scope of this O-RAN spec.

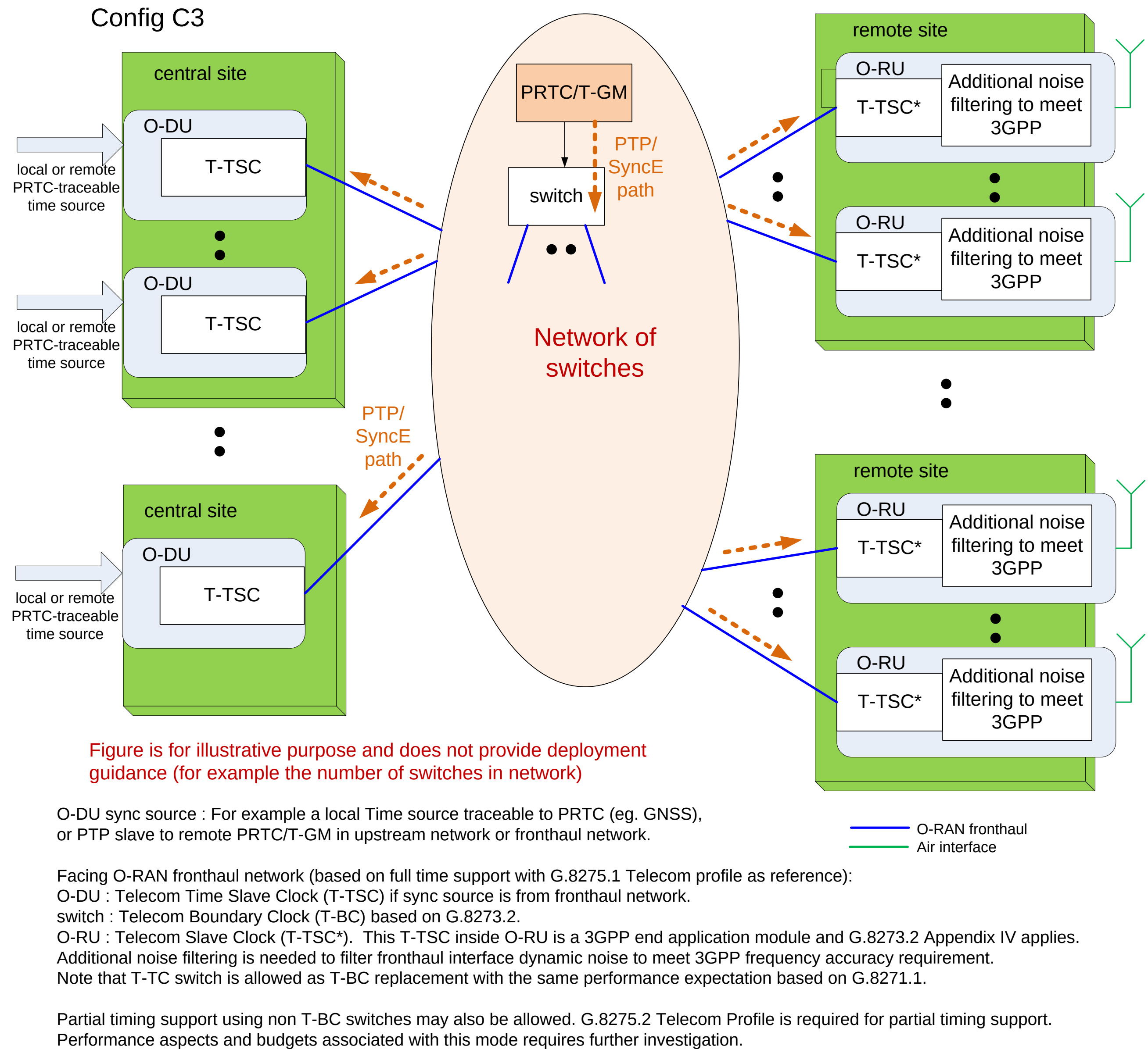


Figure 9-2 : Configuration C3 synchronization

Notes for O-DU:

Unlike C1 & C2, O-DU does not act as SYNCE and PTP Master towards the fronthaul interface. It can select its own synchronization from local or remote PRTC-traceable time source like in C1/C2, but can also select the same SYNCE and PTP distribution from the fronthaul as the O-RU.

One possible C3 implementation consists in having one of the ITU-T G.8273.2-compatible T-BC of the chain being the O-DU.

Notes for O-RU:

Same as C1 & C2.

Notes for PRTC/T-GM (as operator choice in deployment):

- PRTC-traceable time source (typically a GNSS receiver) can be embedded either in an external T-GM connected to any T-BC, or an embedded function inside any T-BC in the network,

- Multiple PRTC/T-GMs can offer redun-dancy. ITU-T G.8275.1 BMCA is used to optimize the time distribution through the clock chain.

9.2.2.3 Topology configuration C4 Synchronization

O-RAN maintains network timing distribution as the preferred approach within the fronthaul network, however, there could be some deployment use cases that prevent the fronthaul network (or only a section of the network) from being upgraded to G.8271.1 compliance and meeting the target performance at the O-RU. To cover these use cases, the O-RAN synchronization strategy shall allow local PRTC-traceable time source (typically GNSS) timing option at the O-RU.

It should be noted that such timing support at O-RU requires extra timing interface or embedded function.

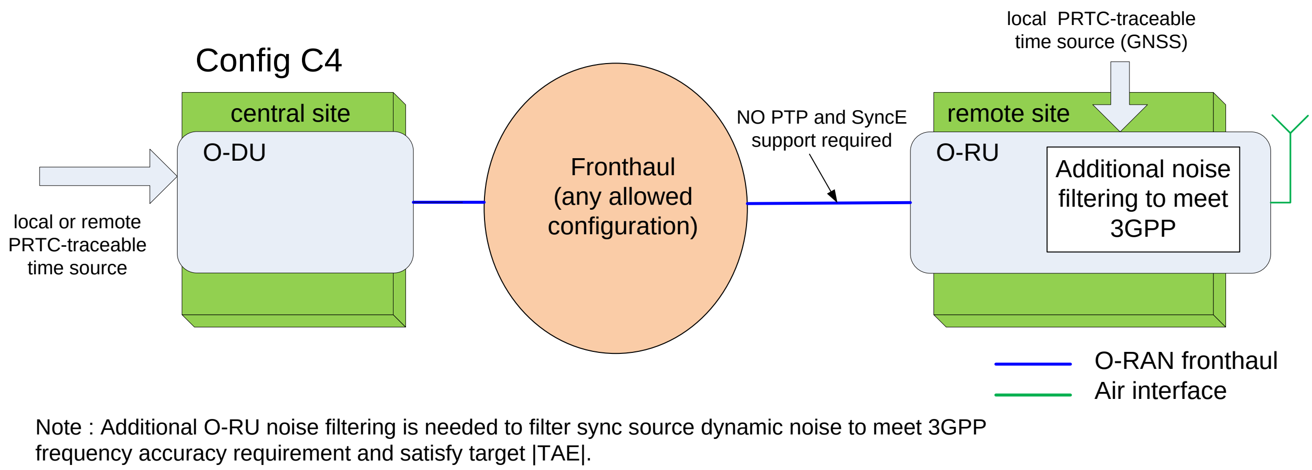


Figure 9-3 : Configuration C4 synchronization

9.2.3 Clock Synchronization

The following requirements are provided for clock synchronization:

- SyncE may be used within the fronthaul network distribution to improve PTP performance per ITU-T G.8271.1 and G.8273.2. In C1 and C2 configurations, the O-DU shall drive SyncE (where used) regardless its own selected time source. In C3 configuration, the network shall deliver SyncE (where used) to all switches and O-RUs. However, an O-RU is not required to use SyncE to achieve clock (frequency) synchronization. An O-RU can use PTP alone to achieve both frequency and phase/time synchronization.
- When used for clock synchronization, SyncE shall comply with the following ITU-T specifications:
  - Definitions: ITU-T G.8260
  - Architecture: ITU-T G.8261
  - SSM transport channel and format: ITU-T G.8264
  - Clock specifications: ITU-T G.8262 (EEC)
  - Functional model and SSM processing: ITU-T G.781
- ITU-T G.8271.1 shall be consulted for guidance on balancing the need to meet a target air interface performance at the O-RU with the maximum number of hops allowed. The network operator shall plan for specific fronthaul network deployment based on this guidance.

9.2.4 Profiles

9.2.4.1 SyncE

An implementation providing SSM shall:

- Support ITU-T G.781 Option 1 Quality Level (per section 5.4 Quality Code of ITU-T G.781)

    - Within QL code Option 1, the accepted QL is to be limited to PRC.
- Support ITU-T G.781 other options, is for further study. ITU-T G.8271.1 analysis has only been done with a synchronous Ethernet network based on option 1 EECs.
- Support ITU-T G.8264 message types, format, transmission and reception (per Table 11.3 of ITU-T G.8264) if SyncE is used.
    - Support of extended SSM TLV (per the 2017 edition of ITU-T G.8264) is optional, and For Further Study

9.2.4.2 PTP

9.2.4.2.1 Full Timing Support

Use of IEEE Std 1588-2008 (hereafter referred as IEEE 1588 or PTP) for time/phase synchronization shall be according to its clauses referred by ITU-T G.8275.1 (Full Timing Support).

Notes:

- The T-TSC inside the O-RU and O-DU are considered as T-TSC inside 3GPP end application modules. Such T-TSC may not provide a 1PPS measurement interface, and ITU-T G.8273.2 Appendix IV applies: the combined performance within each module may not behave as a stand-alone T-TSC described in the normative section of the recommendation.

- Both Congruent and Non-congruent SyncE and PTP distribution, may optionally be supported, as per ITU-T G.8271.1.

9.2.4.2.2 Partial Timing Support

Support of Partial Timing Support using ITU-T G.8275.2 Telecom Profile is currently considered as permissible but requires additional considerations:

- Partial Timing Support allows switches with no T-BC or T-TC, hence there is no guarantee of synchronization performance based on ITU-T standard specification such as G.8273.2. As a result, the system operator must ensure the network components will have adequate performance to meet frequency and phase error budgets to allow an accurate detection of frequency accuracy and phase for proper network operation. Such budgets and implications on performance require further investigation.
- O-RUs (and O-DU as T-TSC from fronthaul in configuration C3) must support L3 (UDP/IP) which is considered “optional” in this version of the CUS-Plane specification.
- O-RUs (and O-DU as T-TSC from fronthaul in configuration C3) must support unicast communication with the GM.
- For configurations C1 and C2, the O-DU must implement G.8275.2 PTP master function.

Note finally that Partial Timing Support is not finalized in the ITU, which has considered this timing method only for relatively coarse timing accuracy (1.5 μsec).

9.2.5 Synchronization Accuracy

9.2.5.1 Jitter

Within the O-RAN fronthaul network, all network equipment (NE) supporting SyncE transport across the network shall comply with input and output jitter requirements specified in ITU-T G.8262 (for EEC).

9.2.5.2 Wander

Within the O-RAN fronthaul network, all network equipment (NE) supporting SyncE transport across the network shall comply with input and output wander requirements specified in ITU-T G.8262 (for EEC).

9.2.5.3 Air interface frequency error

The O-RAN fronthaul network shall ensure O-RU meeting a +/-50ppb air interface frequency error requirement. 3GPP TS 36.104 (for LTE macro cells) and TS 38.104 (for 5G macro ceO) specify +/-50ppb as the short-term average error in 1ms duration applicable to both LTE and 5G technologies. Refer to section 9.3.2 for more detail information.

9.2.5.4 Air interface maximum time error

The O-RAN fronthaul network shall ensure O-RU meeting the following air interface time alignment error (|TAE| absolute or relative) requirements based on different features in LTE and 5G technologies. For features covered by 3GPP, they are specified in TS 36.104 (for LTE) and TS38.104 (for 5G).

The following figure shows the reference points to define the network time error |TE| vs air interface time alignment error |TAE| and the concept of relative vs absolute.

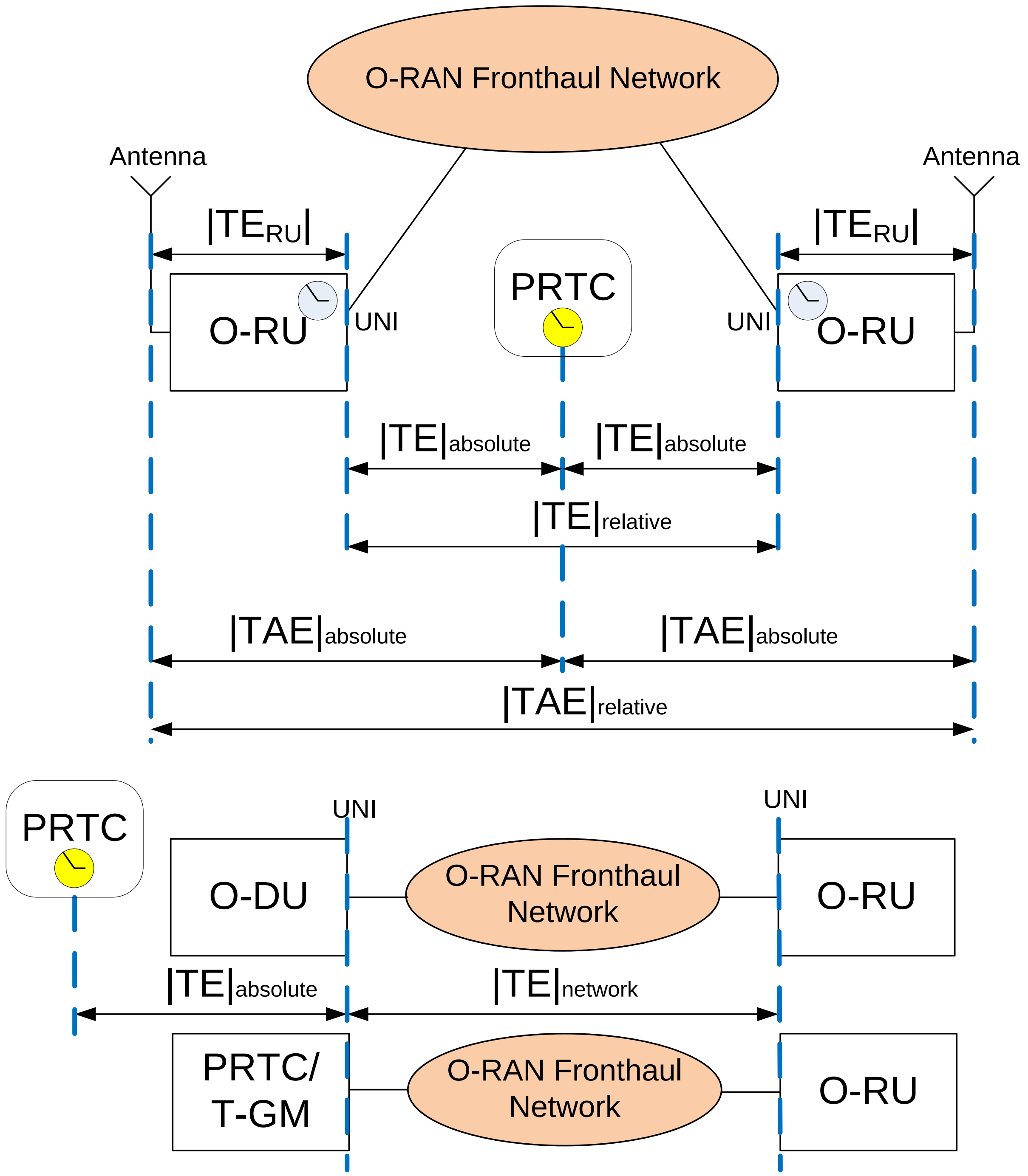


Figure 9-4 : Definition of |TE| and |TAE| and UNI

For LTE features:

Table 9-1 : LTE features with time alignment error requirement at the air interface

LTE Features	|TAE| relative or absolute	Corresponding Timing Category in 802.1CM/eCPRI standard (Informative)
TDD	Relative |TAE| ≤ 3µs (cell radius < 3km) Relative |TAE| ≤ 10us (cell radius > 3km) (note 1) TS36.133	Timing Category C
Dual connectivity	Relative |TAE| ≤ 3µs (note 1) TS36.133	Timing Category C
MIMO or Tx Diversity	Relative |TAE| ≤ 65ns TS36.104	Timing Category A+ (note 2)
CA (intraband contiguous)	Relative |TAE| ≤ 130ns TS36.104	Timing Category A (note 2)
CA (interband or intraband non-contiguous)	Relative |TAE| ≤ 260ns TS36.104	Timing Category B (note 2)
OTDOA	Absolute |TAE| at O-RU antenna << 1.5us, ~100-200 ns (not defined by 3GPP)	Not covered since it is not defined by 3GPP

Note 1: For TDD and dual connectivity features, relative |TAE| requirement is applied to any pair of O-RUs within RAN without any cluster boundary limit. Hence relative |TAE| spec indirectly leads to a per O-RU requirement of absolute |TAE| at O-RU antenna = 1.5µs.

Note 2: When these features are supported within 1 O-RU, relative |TAE| is impacted by O-RU internal |TE| only. When these features are supported by multiple cooperating O-RUs, relative |TAE| is also impacted by network relative |TE| where 802.1CM Timing Category is applicable to limit the network error contribution.

Note 3: The corresponding Timing Category column is shown since later sections refer to these categories as guidance.

For 5G features:

Table 9-2 : 5G features with time alignment error requirement

5G Features	|TAE| relative or absolute	Corresponding Timing Category in 802.1CM/eCPRI standard (Informative)
TDD	Relative |TAE| ≤ 3µs (note 1) TS38.133	Timing Category C
Dual connectivity	Relative |TAE| ≤ 3µs (note 1) TS38.133	Timing Category C
MIMO and Tx Diversity	Relative |TAE| ≤ 65ns TS38.104 [12]	Timing Category A+ (note 2)

CA (intraband contiguous per base station type)	Relative |TAE| ≤ 130ns (CWV in 3GPP) TS38.104 [12]	Timing Category A (note 2)
CA (intraband contiguous per base station type)	Relative |TAE| ≤ 260ns (CWV in 3GPP) TS38.104 [12]	Timing Category B (note 2)
CA (interband or intraband non-contiguous)	Relative |TAE| ≤ 3µs TS-38.104 [12]	Timing Category C (note 2)
OTDOA	Absolute |TAE| at eRE antenna << 1.5us (not defined by 3GPP)	Not covered since it is not defined by 3GPP

Note 1: For TDD and dual connectivity features, relative |TAE| requirement is applied to any pair of O-RUs within RAN without any cluster boundary limit. Hence relative |TAE| spec indirectly leads to a per O-RU requirement of absolute |TAE| at O-RU antenna = 1.5µs.

Note 2: When these features are supported within 1 O-RU, relative |TAE| is impacted by O-RU internal |TE| only. When these features are supported by multiple cooperating O-RUs, relative |TAE| is also impacted by network relative |TE| where 802.1CM Timing Category is applicable to limit the network error contribution.

Note 3: The corresponding Timing Category column is shown since later sections refer to these categories as guidance.

9.3 Time and Frequency Synchronization Requirements

9.3.1 Allowed PTP and SyncE clock types and clock classes

A network element (NE) may use the following clock types and classes to support PTP and SyncE, and can be used among other such NEs to build an O-RAN-compliant fronthaul network meeting end-to-end frequency synchronization requirements as well as time synchronization requirements at the air interface.

- EEC (per ITU-T G.8262)
- PRTC (per ITU-T G.8272: Class A already specified, class B under development)
- ePRTC (per ITU-T G.8272.1)
- T-GM (per upcoming ITU-T G.8273.1)
- T-BC (per ITU-T G.8273.2: class A, B already specified)
- T-TSC (per ITU-T G.8273.2: class A, B already specified)
- T-TC* (per ITU-T G.8273.3)

*According to ITU-T G.8271.1 and IEEE 802.1CM, T-TC is allowed as T-BC replacement and T-TC/T-BC accuracy performance is equivalent. O-RAN allows T-TC as accepted clock type. However, using T-TC has potential issue with broadcast storms in a bridging network as stated in G.8271.1 and hence guidance by Appendix I in G.8275.1 should be followed.
The time error analysis in Annex H only covers T-BC deployments, and not the T-TC one (which is For Further Study)

O-DU and O-RU are considered end points in an O-RAN-compliant fronthaul network. O-DU and O-RU can support specific PTP clock and specific classes based on O-DU and O-RU synchronization capability options.

O-DU synchronization capability options:

- For configurations C1 and C2: SYNCE and PTP Master.
    - In case of local PRC/PRTC-traceable time source (typically a GNSS receiver) used as frequency and time source: O-DU acts a SyncE and PTP master. The performance requirements are specified in section 9.3.2.
    - In case of a remote PRC/PRTC receiver used as frequency and time source via a Full Timing Support packet network (defined by ITU-T G.8271.1): O-DU shall act as an embedded end application with better wander and jitter filtering capability, as per Appendix IV of ITU-T G.8273.2 (thus acting more

like a combined PTP Slave/Master than a T-BC). Note that acting as a true ITU-G G.8273.2T-BC does not guarantee the frequency accuracy required in 9.3.2, and is therefore For Further Study.
In the for-further-study case of a remote PRC/PRTC receiver used as frequency and time source via a Partial Timing Support packet network (defined by ITU-T G.8271.2): O-DU may act as an InterWorking Function.

- For configuration C3 and C4: O-DU collects its synchronization from any possible time source (either local or remote PRC/PRTC-traceable one). O-DU does not need to meet the 3GPP frequency and TAE target specification as required in the O-RU. However, a more relaxed phase alignment between O-DU and O-RU timing should be kept to avoid data buffer overflow/underflow (impact to delay management topic), and are implementation specific.

O-RU synchronization capability options:

- For configuration C1, C2, and C3 with full timing support: (T-TSC is either embedded or external, as per ITU-T G.8271.1 and IEEE 802.1 CM). In the case where O-RU includes a T-TSC, this one is "embedded in end application" as specified in G.8273.2 Appendix IV and include additional necessary filtering function to ensure 3GPP air interface compliance. Under investigation is what is needed for Partial Timing Support because there is not yet any available ITU recommendation.

- For configuration C4, O-RU is simply synchronized from the local PRC/PRTC using proprietary timing interface.

O-RU shall support network timing as mandatory synchronization capability to cover configuration C1, C2 and C3. Local PRC/PRTC (typically a GNSS receiver) shall be optional synchronization capability to cover configuration C4.

9.3.2 Frequency and Time Synchronization Requirements across fronthaul network elements

9.3.2.1 Configurations C1 and C2

Based on IEEE 802.1CM and ITU-T G.8271.1 guidance, the following table summarizes the frequency and time error budgets across different elements of a O-RAN-compliant fronthaul network.

- O-DU : shall not exceed allocated frequency error budget and time error budget (for chosen air interface target)
- O-RU : shall not exceed allocated frequency error budget and time error budget (for chosen air interface target)
- O-RAN fronthaul network: shall not exceed network limit to satisfy both frequency error budget and time error budget (for chosen air interface target). Allowed number of switches in a deployment can be derived based on allowed network limit vs chosen switch specification. Annex H shows the analysis of number of switches based on T-BC Class B switches.

The following table covers budget allocation for configuration C1 and C2 (Refer to **Figure 9-4** for reference point definition). **Requirement is in BOLD** :

Table 9-3 : Frequency and time error budget allocation (for topology configuration C1 and C2)

Frequency error budget allocation			
Timing Reference	**O-RAN fronthaul network contribution limit**	**O-RU**	**Air interface target**

O-DU PTP/SyncE master class A, with |freq error| ≤ 15ppb • @ O-DU UNI (see Note 1)	**total |dTE_{L+H}| ≤ 45ns** • Between O-DU UNI and O-RU UNI • For C2: Allowed number of hops (see Note 2) • For C1: single hop by definition	**O-RU |freq error| ≤ 35ppb** including both • FFO after O-RU filtering of dTE_{L+H} @ O-RU UNI • O-RU internal additive frequency error (see Note 3)	±50ppb per 3GPP spec
O-DU PTP/SyncE master class B, with |freq error| ≤ 5ppb • @ O-DU UNI (see Note 1)	**total |dTE_{L+H}| ≤ 57ns** • Between O-DU UNI and O-RU UNI • For C2: Allowed number of hops (see Note 2) • For C1: single hop by definition	**O-RU |freq error| ≤ 45ppb** including both • FFO after O-RU filtering of dTE_{L+H} @ O-RU UNI • O-RU internal additive frequency error (see Note 3)	±50ppb per 3GPP spec
Time error budget allocation			
Timing Reference	**O-RAN fronthaul network contribution limit**	**O-RU (All allowed class options are shown) (see Note 7)**	**Air interface target**
No relative |TE| contribution by O-DU since O-DU is common PTP andSyncE master to all co-operated O-RU (Note 1)	**Relative |TE| ≤ 60ns** • Between 2 O-RUs UNI • per IEEE 802.1CM • For C2: Allowed number of hops (see Note 4) • For C1: single hop by definition	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Enhanced T-TSC |TE| ≤ 15ns**	Per IEEE 802.1CM Category A
	Relative |TE| ≤ 100ns (using Class B T-TSC in O-RU) **Relative |TE| ≤ 190ns (using enhanced T-TSC in O-RU)** • Between 2 O-RUs UNI • per IEEE 802.1CM • For C2: Allowed number of hops (see Note 4) • For C1: single hop by definition	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC |TE| ≤ 15ns**	Per IEEE 802.1CM Category B
For C2: absolute |TE| ≤ 1.325us **For C1: absolute |TE| ≤ 1.420us** • @ O-DU UNI • includes holdover budget	**Network |TE| ≤ 95ns (using Class B T-TSC in O-RU)** **Network |TE| ≤ 140ns (using enhanced T-TSC in O-RU)** • Between O-DU UNI and O-RU UNI • For C2: Allowed number of hops (see Note 4) • For C1: single hop by definition	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5 and 8)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC |TE| ≤ 15ns**	Absolute |TAE| at O-RU antenna = 1.5μs derived from 3GPP (Same as IEEE 802.1CM Category C)
Absolute |TE| ≤ 100ns • @ O-DU UNI • based on PRTC/T-GM spec per G.8272 • Not including holdover budget	**Network |TE| is out of scope** • between O-DU UNI and O-RU UNI • For C2: Allowed number of hops (see Note 6) • For C1: single hop by definition	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC|TE| ≤ 15ns**	Absolute |TAE| at O-RU antenna for OTDOA (out of scope) (see Note 6)

Note 1: O-DU implements PTP and SyncE master function in this use case.
Two frequency error limits are defined in this specification, conservative class A with 15ppb, and more advanced class B with 5 ppb.

They are measured after applying a first-order measurement low-pass filter bandwidth of 0.1Hz to the time samples. Measurement condition is applicable when O-DU Master Clock is either in locked state or holdover, but excluding rare and temporary transients:

- Resynchronization to recovered source after holdover.
- Resynchronization to newly selected source after failure of the previous one, in case of redundancy.

It is considered that all master ports of the O-DU are fully synchronized together, and there is no port-to-port time error (see Annex H).

Note 2: dTE_{L+H} = accumulated dynamic time error (dTE_L and dTE_H) of a T-BC clock chain (excluding O-DU contribution) based on G.8271.1 Appendix IV calculation method. Refer to Annex H for detailed analysis for maximum number of T-BC Class B switches.

Note 3: The O-RU requirement is not specified in IEEE 802.1CM or ITU-T G.8271.1. O-RAN only specifies the total O-RU frequency error budget to allow design flexibility by different solution vendors. The O-RU solution vendor can make a tradeoff between FFO (Fractional Frequency Offset after applying O-RU filtering) and internal additive frequency error as long as the total frequency error budget with network total |dTE| limit is met. Refer to Annex H for analysis.

Note 4: Refer to Annex H for analysis of the number of switches to satisfy the allowed network limit.

Note 5: The T-TSC are considered T-TSC embedded in end application as specified in G.8273.2 Appendix IV and the performance may not behave as a standalone T-TSC described in the normative section of the recommendation. However, to ensure interoperability among O-RAN O-RU vendors, the agreed performance will be based on eCPRI and 802.1CM recommendation: under eCPRI Transport Network V1.1 Requirements Specification and IEEE 802.1CM V1.2, a standard T-TSC (|TE| = 60ns) and a new enhanced T-TSC (|TE| = 15ns) are defined. They are considered as cTE in this document.

Note 6: OTDOA feature accuracy is not defined by 3GPP. Hence absolute |TAE| at O-RU antenna and the corresponding network |TE| are out of scope in O-RAN specification. O-DU |TE| and O-RU |TE| are still stated in O-RAN specification based on existing standards. In a deployed network, operator can choose a target absolute |TAE| and then derive the corresponding network |TE| and the allowed number of hops and required types of T-BC(s)/T-TC(s) based on ITU-T G.8271.1 guidance.

Note 7: O-RU |TE| is the requirement. O-RAN recommends O-RU internal split as shown in the table based on eCPRI and 802.1CM guidance. It is also acceptable for RAN solution vendor to use a different O-RU internal split to meet the same O-RU |TE| requirement.

Note 8: O-RAN recommends meeting the specified budget split between O-DU and O-RU |TE| across all categories based on eCPRI and 802.1CM guidance. A different budget split than what O-RAN recommends may meet the overall radio interface requirements, but must be considered under the operator's responsibility. Typically, configuration C1 and C2 with category C target may allow more flexibility in O-DU |TE|, network |TE| and O-RU |TE| budget split as long as the network limit (network |TE| for time error and $|dTE_{L+H}|$ for frequency error) is still respected. For example, if an operator can guarantee the network |TE| < 75ns which is below the 95ns limit set in this O-RAN specification, 20ns (95ns-75ns) can be moved into the O-DU |TE| and/or O-RU |TE| budget. Inter-operability can be guaranteed by the network operator when all participating O-DU and O-RU vendors design to meet the operator-chosen budget split.

Note 9: With Partial Timing Support, when using non-T-BC switches, network contribution limit requires further investigation.

9.3.2.2 Configuration C3

Based on IEEE 802.1CM and ITU-T G.8271.1 guidance, the following table summarizes the frequency and time error budgets across different elements of a O-RAN-compliant fronthaul network.

- PRTC/T-GM : shall not exceed allocated frequency error budget and time error budget (for chosen air interface target)
- O-RU : shall not exceed allocated frequency error budget and time error budget (for chosen air interface target)
- O-RAN fronthaul network : shall not exceed network limit to satisfy both frequency error budget and time error budget (for chosen air interface target). Allowed number of switches in a deployment can be derived based on allowed network limit vs chosen switch specification. Annex H shows the analysis of number of switches based on T-BC Class B switches.

The following table covers budget allocation for configuration C3 (Refer to **Figure 9-4** for reference point definition).
Requirement is in BOLD :

Table 9-4 : Frequency and time error budget allocation (for topology configuration C3)

Frequency error budget allocation			
Timing Reference	**O-RAN fronthaul network contribution limit**	**O-RU**	**Air interface target**
PRTC/T-GM |freq error|≤ 2ppb • @ PRTC/T-GM UNI (see Note 1)	**total |dTE_{L+H}| ≤ 63ns** • Between PRTC output and O-RU UNI • Include PRTC/T-GM MTIE contribution • Allowed number of hops (see Note 2)	**O-RU |freq error| ≤48ppb** including both • FFO after O-RU filtering of dTE_{L+H} @ O-RU UNI • O-RU internal additive frequency error (see Note 3)	±50ppb per 3GPP spec
Time error budget allocation			
Timing Reference	**O-RAN fronthaul network contribution limit**	**O-RU** **(All allowed class options are shown)** **(see Note 7)**	**Air interface target**
No relative |TE| contribution by PRTC/T-GM since PRTC/T-GM is common PTP and SyncE master to all co-operated O-RU (see Note 1)	**Relative |TE| ≤60ns** • Between 2 O-RUs UNI • per IEEE 802.1CM • Allowed number of hops (see Note 4)	**O-RU |TE|** includes as per IEEE 802.1CM**:** • **RF|TE| ≤ 20ns** • **enhanced T-TSC |TE| ≤ 15ns**	Per IEEE 802.1CM Category A
	Relative |TE| ≤100ns (using Class B T-TSC in O-RU) **Relative |TE| ≤190ns (using enhanced T-TSC in O-RU)** • Between 2 O-RUs UNI • per IEEE 802.1CM • Allowed number of hops (see Note 4)	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC|TE| ≤ 15ns**	Per IEEE 802.1CM Category B
Absolute |TE| ≤ 100ns • @ PRTC/T-GM UNI • PRTC/T-GM spec per G.8272 • Not including holdover budget	**Network |TE| ≤ 1320ns (using Class B T-TSC in O-RU)** **Network |TE| ≤ 1365ns (using enhanced T-TSC in O-RU)** • Between PRTC/T-GM UNI and O-RU UNI • Allowed number of hops (see Note 4)	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC|TE| ≤ 15ns**	Absolute |TAE| at O-RU antenna = 1.5μs derived from 3GPP (Same as IEEE 802.1CM Category C)
Absolute |TE| ≤ 100ns • @ PRTC/T-GM UNI • PRTC/T-GM spec per G.8272 • Not including holdover budget	**Network |TE| is out of scope** • between PRTC/T-GM UNI and O-RU UNI • Allowed number of hops (see Note 6)	**O-RU |TE|** includes as per IEEE 802.1CM (see Note 5)**:** • **RF|TE| ≤ 20ns** • **Either Class B T-TSC |TE| ≤ 60ns** • **Or enhanced T-TSC |TE| ≤ 15ns**	Absolute |TAE| at O-RU antenna for OTDOA (out of scope) (see Note 6)

Note 1: PRTC/T-GM follows G.8272 specification. MTIE specification is considering dynamic error during lock condition and its contribution is covered under dTE_{L+H} in note 2. However, there is possible semi-static frequency error which is not part of MTIE specification. Therefore, it is included here.
In case the PRTC/T-GM is a multiple-port device, it is considered that all master ports are fully synchronized together, and there is no port-to-port time error.
Refer to Annex H for detailed analysis.

Note 2: dTE_{L+H} = accumulated dynamic time error (dTE_L and dTE_H) of a T-BC clock chain, including PRTC/T-GM contribution, to O-RU UNI (or O-DU UNI) based on G.8271.1 Appendix IV calculation method. Refer to Annex H for analysis for maximum number of T-BC Class B switches.

Note 3: The O-RU requirement is not specified in IEEE 802.1CM or ITU-T G.8271.1. O-RAN only specifies the total O-RU frequency error budget to allow design flexibility by different solution vendors. The O-RU solution vendor can make a tradeoff between FFO (Fractional Frequency Offset after applying O-RU filtering) and O-RU internal additive frequency error as long as the total frequency error budget with network total |dTE| limit is met. Refer to Annex H for analysis.

Note 4: Refer to Annex H for analysis of the number of switches to satisfy the allowed network limit.
Note 5: The T-TSC are considered T-TSC embedded in end application as specified in G.8273.2 Appendix IV and the performance may not behave as a standalone T-TSC described in the normative section of the recommendation. However, to ensure interoperability among O-RAN O-RU vendors, the agreed performance will be based on eCPRI and 802.1CM recommendation: under eCPRI Transport Network V1.1 Requirements Specification and IEEE 802.1CM V1.2, a standard T-TSC (|TE| = 60ns) and a new enhanced T-TSC (|TE| = 15ns) are defined. They are considered as cTE in this document.

Note 6: OTDOA feature accuracy is not defined by 3GPP. Hence absolute |TAE| at O-RU antenna and the corresponding network |TE| are out of scope in O-RAN specification. O-DU |TE| and O-RU |TE| are still stated in O-RAN specification based on existing standards. In a deployed network, operator can choose a target absolute |TAE| and then derive the corresponding network |TE| and the allowed number of hops and required types of T-BC(s)/T-TC(s) based on ITU-T G.8271.1 guidance.

Note 7: O-RU |TE| is the requirement. O-RAN recommends O-RU internal split as shown in the table based on eCPRI and 802.1CM guidance. It is also acceptable for RAN solution vendor to use a different O-RU internal split to meet the same O-RU |TE| requirement.

Note 8: With Partial Timing Support, when using non-T-BC switches, network contribution limit requires further investigation.

9.3.2.3 Configuration C4

The following table covers budget allocation for configuration C4:

Table 9-5 : Frequency and time error budget allocation (for topology configuration C4)

Frequency error budget allocation			
Timing Reference in network or O-DU	**O-RAN fronthaul network**	**O-RU**	**Air interface target**
No dependency	No dependency	O-RU |freq error| ≤ 50ppb including both • O-RU filtering on local PRTC noise • O-RU internal additive frequency error (Note 3, 4)	±50ppb per 3GPP spec
Time error budget allocation			
Timing Reference in network or O-DU	**O-RAN fronthaul network**	**O-RU (See note 1,2, 3, 4)**	**Air interface target**
No dependency	No dependency	Local time source |TE| ≤ 100ns O-RU |TE| ≤ 30ns	IEEE 802.1CM Category B

		local time source |TE| ≤ 100ns O-RU |TE| ≤ 1.4us including any holdover budget	Absolute |TAE| at O-RU antenna = 1.5µs derived from 3GPP (Same as IEEE 802.1CM Category C)
		local time source |TE| ≤ 100ns O-RU |TE| is out of scope (see Note 5)	Absolute |TAE| at O-RU antenna for OTDOA (out of scope) (see Note 5)

Note 1: local PRTC-traceable time source (typically GNSS receiver) to O-RU. Therefore, budget is based on ITU-T G.8272 (PRTC class A, with max |TE| = 100ns) on each GNSS receiver. Because there is no requirement in ITU-T G.8272 on the relative time error between two neighbor local PRTC (GNSS receivers), it is not possible to remove any common time error in the budget at the O-RUs side.

Note 2: When feature for relative |TAE| is needed, C4 configuration will include 2x local PRTC |TE| = 200ns total. This disadvantage automatically prevents C4 configuration from meeting certain demanding relative |TAE| feature (as target by 802.1CM Category A+/A). Hence, only 802.1CM Category B/C are shown.

Note 3: O-RU |TE| is not governed by eCPRI/802.1CM or ITU-T standard since C4 configuration is not based on network timing solution. O-RU |TE| budget is basically the remaining budget to satisfy target feature |TAE| after excluding the local PRTC |TE| contribution. O-RU |TE| includes the O-RU clock recovery (i.e. deriving a clean clock from local time source) error and any O-RU internal error.

Note 4: This O-RU option requires extra interface and extra hardware support including local PRTC (typically GNSS receiver and antenna) and likely a more expensive oscillator for noise filtering. Standard O-RU with network timing support (target for configuration C1, C2 and C3) cannot offer this option in general. Specific O-RU design is needed.

Note 5: OTDOA feature accuracy is not defined by 3GPP. Hence absolute |TAE| at O-RU antenna and the corresponding O-RU|TE| are out of scope in O-RAN specification. In a deployed network, operator can choose a target absolute |TAE| and then derive the corresponding O-RU |TE|.

9.4 Node Behavior Guidelines

9.4.1 Configurations C1 and C2

This section covers O-RAN topology configurations C1 and C2 where the O-DU acts as SyncE and PTP master.

The operation of O-DU and O-RU during holdover and other related states is described in **Table 9-6**. O-RU holdover and O-DU holdover are independent events. Likewise, O-RU holdover behavior is optional (not mandatory to be supported by HW or SW).

In addition to synchronization state, the O-DU also considers estimated synchronization accuracy, because synchronization state alone does not necessarily reflect synchronization status – a node in the SYNCED or HOLDOVER mode may have synchronization accuracy outside of a required limit.

Table 9-6 : Node behavior during Holdover and Out-of-Synch

O-DU State	Synch accuracy	O-RU State	M-Plane State	Action
-	-	-	Disconnected	Section 9.4.1.1
-	N/A	FREERUN	Connected	Section 9.4.1.2
FREERUN	-	-	Connected	Section 9.4.1.3
SYNCED/ HOLDOVER	In a limit	SYNCED/ HOLDOVER	Connected	Section 9.4.1.4
SYNCED/ HOLDOVER	Out of a limit	SYNCED/ HOLDOVER	Connected	Section 9.4.1.4

9.4.1.1 M-Plane disconnected

O-RAN allows hybrid M-plane model with M-plane communication between

- O-RU and O-DU
- O-RU and Network Management System (NMS)

As a result, the following M-plane disconnected events must be considered:

- O-DU detects loss communication to O-RU
- NMS detects loss communication to O-RU
- O-RU detects loss communication to O-DU
- O-RU detects loss communication to NMS
- O-RU detects loss communication to both O-DU and NMS

Note: The following behavior is an assumption and expected to be described in M-Plane specification.

O-DU

If the O-DU detects a loss of M-plane communication to a O-RU, the O-DU shall stop sending any IQ data towards the O-RU. The O-DU shall also send an explicit command to the O-RU to disable RF transmission.

The O-DU shall keep RF transmission on the O-RU off, and shall not turn it on, if M-plane communication to the O-RU is broken.

Rationale: The requirement for the O-DU to send an explicit command is intended to prevent unsupervised radio operation, if M-plane communication broken in one direction only, and the fault stays undetected on the O-RU. On the other hand, if the O-RU detects the fault, it disables RF transmission autonomously (see below), and the O-DU shall stop IQ transmission accordingly.

O-RU

If the O-RU detects a loss of M-plane communication to the O-DU or both O-DU and NMS, the O-RU stops RF transmission. The O-RU shall keep RF transmission off, and shall not turn it on, if M-plane communication to the O-DU or NMS is broken.

If the O-RU detects a loss of M-plane communication to the NMS only, the O-RU action shall continue RF transmission. It is expected NMS can eventually detect a loss of M-plane communication to the O-RU based on some round-trip sanity check mechanism. Any NMS action is out of the scope of this specification.

NMS

If the NMS detects a loss of M-plane communication to an O-DU or an O-RU, the NMS action is out of scope of this specification.

9.4.1.2 O-RU in the FREERUN State

O-DU

If synchronization state on a connected O-RU transits to the FREERUN state, the O-DU may either keep RF transmission at that O-RU enabled, or disable it. **Rationale:** The O-RU autonomously stops RF transmission, when it transits into the FREERUN state. If the O-DU does not disable RF transmission, the O-RU may autonomously reenable RF transmission when synchronization recovers. In this way, the recovery time is reduced.

Note: Unlike CPRI/OBSAI, where such procedure could cause transmission of unwanted random IQ data on incorrectly wired links, eCPRI has no such risk, because all IQ data is sent in Ethernet frames addressed to a globally unique MAC address of a specific O-RU, and messages are protected from bitflips in addresses by CRC check at Layer 1.

O-RU

If synchronization state on a O-RU transits to the FREERUN state, the O-RU shall autonomously stop RF transmission, until it is resynchronized. Unless there is another reason to keep RF transmission disabled (explicit command from the O-DU, loss of M-plane communication, etc.), the O-RU may re-enable RF transmission when the O-RU gets locked to an input reference again.

9.4.1.3 O-DU in the FREERUN state

O-DU

If an O-DU transits to the FREERUN state, the O-DU shall disable RF transmission on all connected O-RUs, and keep it turned off until synchronization is reacquired.

Note: The O-DU shall support configuration option that allows O-DU to operate outside of the required synchronization limits, or without any synchronization at all.

O-RU

The O-RU shall only react on a change of Quality Level, received in SyncE SSMs, and Clock Class, received in PTP Announce messages:

- If the received Quality Level and Clock Class are acceptable the O-RU shall continue using the reference signal.
- If the received Quality Level or Clock Class becomes unacceptable the O-RU shall stop using the reference and transit to either the FREERUN state or a HOLDOVER state.
- Otherwise, the O-RU shall rely on O-DU to take care of the changed synchronization state. The O-RU need not react on the FREERUN state at the O-DU in any distinct way.

9.4.1.4 Operation in SYNCED/HOLDOVER state

Whether in "synchronized" or "Holdover" state, it is expected that O-DU monitors the "SYNCED/HOLDOVER" status of the O-RUs under its management.

O-DU

- In configuration C1 and C2: by collecting the O-RUs' "SYNCHRONIZED" or "HOLDOVER" state, as well as the received SyncE and PTP quality status , O-DU in "SYNCHRONIZED" or "HOLDOVER" state is able to detect any self-estimated frequency and/or time accuracy degradation by the O-RUs.
  Note: all O-RUs in the SYNCHRONIZED state and directly connected to the same master clock (typically the O-DU in C1, and the nearest T-BC in C2) preserve optimal relative time error between them, regardless the O-DU's status. As a result, intra-site features may remain enabled using different criteria compared to inter-site ones.
- Note: the port-to-port constant and dynamic time errors between two master ports of the same module (either the O-DU in configuration C1, and the T-BC ones in C2) may also contribute to the total error. This is currently not specified in ITU-T G.826x and ITU-T G.827x recommendations, and shall be taken into account in the upcoming editions.

O-RU

The O-RU shall only react on a change of Quality Level received in SyncE SSMs, and Clock Class received in PTP Announce messages:

- If received Quality Level and Clock Class are acceptable, the O-RU shall keep on using the reference signal.
- If received Quality Level or Clock Class become unacceptable, the O-RU shall stop using the reference and transit to either the FREERUN state or a HOLDOVER state.
  ITU-T G.8271.1 network limits define the notion of "within holdover specification" (clock class values 7 and 135), and "exceeding the holdover specification" (clock class values 140, 150, 160, 165).
  ITU-T G.8271.1 defines this holdover specification as 400ns in the context of category C (as per IEEE 802.1CM). This value is however too high and useless for categories A/B, highlighting that the notion of "acceptable" is dependent on the category of each feature.
  As a consequence, it is mandatory that each O-RU reports to O-DU the received Quality Level and Clock Class, allowing O-DU to enable and disable accordingly the various RF features.
  Besides, the value for the holdover specification may be configurable, thus allowing each network operator to tune it to its own needs.

9.4.2 Configurations C3

This section covers O-RAN topology configuration C3 based on IEEE 802.1CM bridging network. PRTC/GM is provided by the fronthaul network.

The operation of the Fronthaul network elements, O-DU and O-RU during holdover and other related states is described in **Table 9-7**. O-RU holdover and O-DU holdover are independent events. Likewise, O-RU holdover behavior is optional (not mandatory to be supported by HW or SW).

More than one PRTC/GM may be considered as a deployment option to improve redundancy. Should a PRTC/GM fail, then another should be available as a backup time source and the PTP network tree would automatically re-arrange. Only a short holdover (note: duration needs to be defined from ITU-T G.8271.1) shall be supported inside the various network elements (as well as O-DU and O-RU) to provide a safe operation during this rearrangement scenario. Note: the O-DU can also be configured to provide backup SYNCE+PTP like in C2.

In addition to synchronization state, the O-DU also considers estimated synchronization accuracy because the synchronization state alone does not necessarily reflect synchronization status; a node in the SYNCED or HOLDOVER mode may have synchronization accuracy outside of a required limit.

Table 9-7 : Node behavior during Holdover and Out-of-Sync

Network State	Sync Accuracy	O-DU and O-RU State	M-Plane State	Action
-	-	-	Disconnected	Section 9.4.2.1
-	N/A	FREERUN	Connected	Sections 9. 4.2.2 and 9. 4.2.3
SYNCED/ HOLDOVER	In limit	SYNCED/ HOLDOVER	Connected	Section 9. 4.2.4
SYNCED/ HOLDOVER	Out of limit	SYNCED/ HOLDOVER	Connected	Section 9. 4.2.4

9.4.2.1 M-Plane disconnected

This section is same as 9.4.1.1

9.4.2.2 O-RU in the FREERUN State

This section is same as 9.4.1.2

9.4.2.3 O-DU in the FREERUN state

Notes:

- if O-DU and O-RU are synchronized from the same fronthaul network and are connected to neighbor nodes in this network, it is most probable that the event leading to O-RU transiting to the FREERUN state will also lead to the same transition at the O-DU.

- if O-DU has backup frequency and time source, such as local or remote PRTC, it can become a backup Master like in configuration C2.

<u>O-DU</u>

If a O-DU transits to the FREERUN state, because the synchronizing network delivers unacceptable synchronization quality, the O-DU shall disable RF transmission on all connected O-RUs, and keep it turned off until synchronization is reacquired again.

Note: The O-DU shall support configuration option that allows O-DU to operate outside of the required synchronization limits, or without any synchronization at all.

<u>O-RU</u>

The O-RU is not synchronized from the O-DU. It may have no indication of the O-DU synchronization status, and therefore shall only rely on O-DU to take care of the changed synchronization state. The O-RU need not react to the FREERUN state at the O-DU in any distinct way.

9.4.2.4 Operation in SYNCED/HOLDOVER state

Whether in “SYNCED” or “HOLDOVER” state, it is expected that O-DU monitors the “SYNCHRONIZED” or “HOLDOVER” state, as well as the received SyncE and PTP quality status. This is same as configurations C1 and C2 described in earlier section.

9.4.3 Configurations C4

This section covers O-RAN topology configurations C4 where the O-RU is synchronized by local PRTC (typically a GNSS receiver).

The operation of O-DU and O-RU during holdover and other related states is described in

Table 9-8**.** O-RU holdover and O-DU holdover are independent events. Likewise, O-RU holdover behavior is optional (not mandatory to be supported by HW or SW).

In addition to the synchronization state, the O-DU also considers estimated synchronization accuracy because the synchronization state alone does not necessarily reflect synchronization status; a node in the SYNCED or HOLDOVER mode may have synchronization accuracy outside of a required limit.

Table 9-8 : Node behavior during Holdover and Out-of-Sync

O-DU State	Sync Accuracy	O-RU State	M-Plane State	Action
-	-	-	Disconnected	Section 9.4.3.1
-	N/A	FREERUN	Connected	Section 9.4.3.2
FREERUN	-	-	Connected	Section 9.4.3.3
SYNCED/ HOLDOVER	In limit	SYNCED/ HOLDOVER	Connected	Section 9.4.3.4
SYNCED/ HOLDOVER	Out of limit	SYNCED/ HOLDOVER	Connected	Section 9.4.3.4

9.4.3.1 M-Plane disconnected

This section is same as 9.4.1.1

9.4.3.2 O-RU in the FREERUN State

This section is same as 9.4.1.2

9.4.3.3 O-DU in the FREERUN state

O-DU

If a O-DU transits to the FREERUN state, the O-DU shall disable RF transmission on all connected O-RUs and keep it turned off until synchronization is reacquired again.

Note: The O-DU may support a configuration option that allows O-DU to operate outside of the required synchronization limits, or without any synchronization at all.

O-RU

The O-RU is not synchronized from the O-DU. It may have no indication of the O-DU synchronization status, and therefore shall only rely on O-DU to take care of the changed synchronization state. The O-RU need not react to the FREERUN state at the O-DU in any distinct way.

9.4.3.4 Operation in SYNCED/HOLDOVER state

Whether in “SYNCED” or “HOLDOVER” state, it is expected that O-DU monitors the “SYNCHRONIZED” or “HOLDOVER” state. This is same as configurations C1 and C2 described in earlier section.

9.5 S-Plane Handling in Multiple Link Scenarios

Behavior of S-Plane in scenarios with multiple links shall be based on the following principles:

O-DU - Grand Master (configurations C1 & C2)

There must be an input sync reference signal on at least one link to a O-RU. Likewise, it is not prohibited to have input reference signal on multiple or all links to a given O-RU.

O-RU (all configurations)– Slaves

O-RU must be able to adapt at startup to the reception of a sync reference signal on any port from their master. How the ingress signal is detected (usually SSM for SyncE and Announce for PTP), and how the active port is selected (e.g., via round-robin, ITU-T G.8275.1 BMCA, etc.) are implementation-specific.

If the input reference is present on multiple links, the O-RU may, but is not required to, implement redundancy for the input reference signal if only capacity links are present on the module.

9.6 Announce Messages

O-RU shall check the following advertised parameters against a list of acceptable values based on its own design (assumed to be M-Plane configurable):

- Domain Number: Default: 24 (for Full Timing Support per ITU-T G.8275.1) or 44 (fof Partial Timing Support per ITU-T G.8275.2)
- PTP Acceptable Clock Classes:
  - Default: 6, 7
  - Operator configurable: 6, 7, 135, 140, 150, 160, 248

9.7 Elementary Procedures

9.7.1 PTP Time Synchronization procedure

All procedures used to exchange time related information between a time synchronization master and slave shall be compliant to the ITU-T G.8275.1 or G.8275.2 telecom profile, which provides necessary details on utilization of the IEEE 1588-2008 protocol in telecom applications.

9.7.2 System Frame Number Calculation from GPS Time

The general framework for System Frame Number (SFN) calculation from GPS (or GNSS) time is based on the following premises:

- PTP time on the fronthaul interface shall use PTP timescale
- The PTP epoch is 1 January 1970 00:00:00 TAI, which is 31 December 1969 23:59:51.999918 UTC.

- PTP time on the fronthaul interface shall be traceable to a PRTC if a network wide synchronization of O-RUs at the air interface is required (as in TDD 5G)
- From PTP time, the GPS seconds elapsed since GPS epoch (midnight January 6th, 1980) can be calculated, since the difference between PTP and GPS epoch is a constant
- GPS seconds shall be used to calculate the frame number valid at that second according to:
  - FrameNumber = floor[(GPSseconds – β * 0.01 s – α/1.2288e9 Hz)/ framePeriodinSeconds] mod (maxFrameNumber+1)
  - frmePeriodinSeconds = 0.01 seconds; maxFrameNumber = 1023
  - NOTE: α and β are defined as follows:

Table 9-9 : α and β parameter exchange

Direction	Parameter	Range	Resolution
O-DU to O-RU	Offset α (NR) of radio frame timing (NOTE 1)	α : 0~1.2288e7 α * (1/1.2288 ns) : 0.00s~0.01s	1/1.2288 ns
O-DU to O-RU	Offset β (NR) of radio frame timing (NOTE 1)	β: -32768~32767 β * (10ms) : -327680ms~327670ms	10ms
NOTE 1: Parameter data types and values for α and β are provided by the O-RAN M-plane spec. Epoch for α and β (i.e. SFN=0) is set to 1980.1.6 00:00 (UTC)			

The UL to DL radio frame timing offset in TDD systems is described in **Table 9-10** with default values applicable for 5G NR FR1 and FR2 ranges. $N_{TAoffset}$ is the configurable parameter used to control the timing offset. The absolute time offset $T_{TAoffset}$ can be calculated as $N_{TAoffset}$ *1/1.96608GHz.

Table 9-10 : Definition of $T_{TAoffset}$ & $N_{TAoffset}$

Frequency Range	Range	Resolution	Default
FR1	$N_{TAoffset}$: 0~65535 $T_{TAoffset}$: 0µs~33.3µs (NOTE 3)	0.5ns	$N_{TAoffset}$: 25600 (NOTE 2) $T_{TAoffset}$: 25600*Tc= 13.02µs (NOTE 1)
FR2	$N_{TAoffset}$: 0~65535 $T_{TAoffset}$: 0µs~33.3µs (NOTE 3)	0.5ns	$N_{TAoffset}$: 13792 (NOTE 2) $T_{TAoffset}$: 13792*Tc= 7.015µs (NOTE 1)
NOTE 1: Tc=~0.5ns=1/1.96608GHz NOTE 2: based on 3GPP TS38.133 Table 7.1.2-2 NOTE 3: $T_{TAoffset}$=0s for FDD systems			

10 Beamforming Guidelines

10.1 General

This chapter describes terminologies, rules, properties and uses cases related to beamforming and its functionalities. It is the baseline to follow by the O-DU, O-RU and modeling in M-plane.

10.2 Hierarchy of Radiation Structure in O-RU

The hierarchy of radiation structure in O-RU is depicted in **Figure 10-1** and described below:

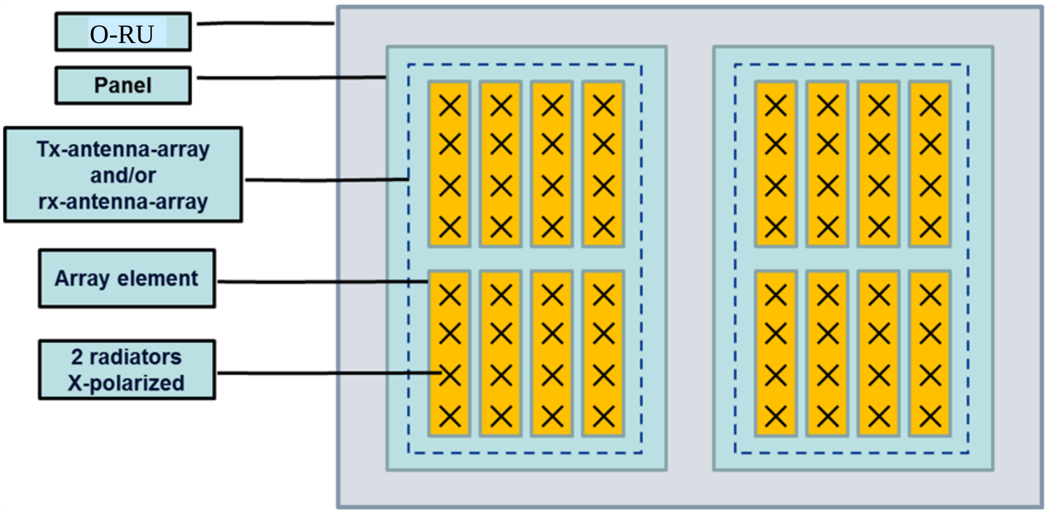


Figure 10-1. Hierarchy of Radiation Structure

- **O-RU:** each O-RU can have 1 or several Panels
- **Panel:** each panel can have 1 or several TX-antenna-arrays/RX-antenna-arrays
- **TX-antenna-array/RX-antenna-array:**
  - TX-antenna-array/RX-antenna-array is a logical construct used for data routing; there is a relationship to physical antennas as defined by the O-RU construction, and eAxC values may be assigned to TX-antenna-arrays/RX-antenna-arrays:
    - Multiple eAxC values may be assigned to a single TX-antenna-array or RX-antenna-array
    - An eAxC value may only be assigned to a single TX-antenna-array and RX-antenna-array
  - Each TX-antenna-array/RX-antenna-array has only 1 polarization for Category A O-RUs.
  - Each TX-antenna-array may include both polarizations for Category B O-RUs.
  - Each TX-antenna-array/RX-antenna-array terminates 1 or several RU_port_IDs.
  - Each TX-antenna-array/RX-antenna-array can have 1 or several array elements (one use case being antenna technologies wherein a single "array element" can support a high degree of beamforming, another use case being the need to address a single element in the array).
  - Each physical array element may be considered as a member of one or multiple TX-antenna-arrays / RX-antenna-arrays.
  - Amplitude and phase of each array element inside one TX-antenna-array/RX-antenna-array can be controlled via the weights pointed to by the beam_ID(s) which are received via the terminated RU_port_ID(s). Alternatively, the beam_IDs may indicate the "state" of an antenna array not explicitly comprising separate radiating elements which is also possible within the O-RAN specification.
  - It is assumed that when executing beamforming, array elements within an array are combined in phase to form directed energy, therefore the array elements belonging to a TX-antenna-array and RX-antenna-array are calibrated together as a group (although such a calibration operation may be hidden from the interface (e.g. executed once upon O-RU manufacture).
- **Array element:** each array element can have 1 or several Radiators (where amplitude and phase relation between the radiators cannot be changed dynamically during real-time)
- **Radiator:** radiating element – see IEEE Std 145-1993 (R2004)

10.3 Calibration

Calibration is the functionality of eliminating/minimizing relative amplitude and phase differences over frequency domain or time delay over the time domain between the array elements belonging to same TX-antenna-array/RX-antenna-array (including effect of front-end analog filters).

Calibration can also be applied between two or many TX-antenna-arrays/RX-antenna-arrays if those TX-antenna-arrays/RX-antenna-arrays belong to the same calibration Group which is part of the O-RU capabilities.

Note: Calibration capability between different TX/RX antenna arrays is expected to be conveyed via M-Plane but is not supported in this version of the specification. When not available, it should be assumed that calibration is not possible between the TX/RX antenna arrays.

10.4 beamId Use for Various Beamforming Methods

There are two main domains in which beamforming is executed, frequency-domain and time-domain; it is also possible to combine both (called “hybrid beamforming”). Frequency-domain beamforming is done between the RE mapping and FFT/iFFT processing stages (in UL and DL respectively) so is inherently a digital operation. Time-domain beamforming may be executed digitally or in the analog domain.

A characteristic of frequency-domain beamforming when used with OFDM is that different users may use the same time slot yet use different beams. In contrast, with time-domain beamforming all the users and signals in a time slot use the same beam. Hybrid beamforming allows different users in the same time slot to use different beams (the frequency-domain part) at the same time as all the users using a shared time-domain beam. An example is the case where the time-domain beam provides directivity in the elevation plane (so all users use the same elevation beam) while the frequency-domain beams provide directivity in the azimuth plane (so different users may use different azimuth beams).

When implementing any kind of beamforming, O-RAN supports the following beamforming methods (see also Annex J for a more mathematical description of the following beamforming methods):

a) Predefined-beam beamforming
b) Weight-based dynamic beamforming (based on real-time-updated weights)
c) Attribute-based dynamic beamforming (based on real-time-updated beam attributes)
d) Channel-information-based beamforming

A: Predefined-beam beamforming

In this case, an index called “beamId” indicates the specific beam pre-defined in the O-RU to use. The beamId could indicate a frequency-domain beam or a time-domain beam or a combination of both (“hybrid” beam) and the O-DU needs to know to ensure the beamId is properly applied e.g. the O-DU could not apply different time-domain beams to the different PRBs in the same OFDM symbol. The method the O-RU uses to generate the beam is otherwise not relevant, it could use the application of gain and phase controls on separate antenna elements, or use multiple shaped-energy antennas, or any other technology. The O-RU is expected to convey to the O-DU via the M-Plane on startup beam characteristics but the O-DU remains ignorant regarding how the beam is actually created by the O-RU.

B: Weight-based dynamic beamforming

Here the O-DU is meant to generate weights that create the beam so the O-DU needs to know the specific antenna characteristics of the O-RU including how many antenna elements are present in the vertical and horizontal directions and the antenna element spacing, among other properties. The weight vector associated with each beam has a beamId value and the interpretation of this beamId value is addressed in this chapter.

C: Attribute-based dynamic beamforming

Like index-based beamforming, attribute-based beamforming allows the O-DU to tell the O-RU to use a specific beamId but in this case that beamId is associated with certain beam attributes as described in chapter 5.4.7.2. How the O-RU achieves the implementation of the beams is not specified, however again the O-DU needs to know whether the beam identified by the beamId is generated as a frequency-domain beam or a time-domain beam to ensure the beamId is properly applied e.g. the O-DU could not apply different time-domain beams to the different PRBs in the same OFDM symbol.

D: Channel-information-based beamforming

In this case the O-DU provides channel information per UE periodically (generally less often than every slot) and then on a slot-by-slot basis the O-DU provides scheduling information which the O-RU uses along with the channel information to calculate the proper beamforming weights for the specific slot with its co-scheduled UEs. Here there is no beamId value associated with the beamforming, instead the ueID is provided associated with each data section. Therefore this sub-chapter regarding beamId usage is not relevant for this beamforming method.

10.4.1 Predefined-beam Beamforming

When implementing index-based beamforming, it is necessary for the O-RU to convey to the O-DU whether the beamforming type is frequency-domain, time-domain, or a mixture of the two ("hybrid beamforming"). In the case of frequency-domain-only or time-domain-only, the beamId is simply an index to the desired beamforming weight vector or other beamforming method. In the case of hybrid beamforming, there are present in the O-RU pre-loaded frequency-domain weight vectors and time-domain weight vectors (these are applied separately). The beamId points to a single combined frequency-domain and time-domain weight vector. However, in reality there will be the application of a frequency-domain beamforming weight vector and the separate application of multiple time-domain beamforming weight vectors, one per frequency-domain weight value.

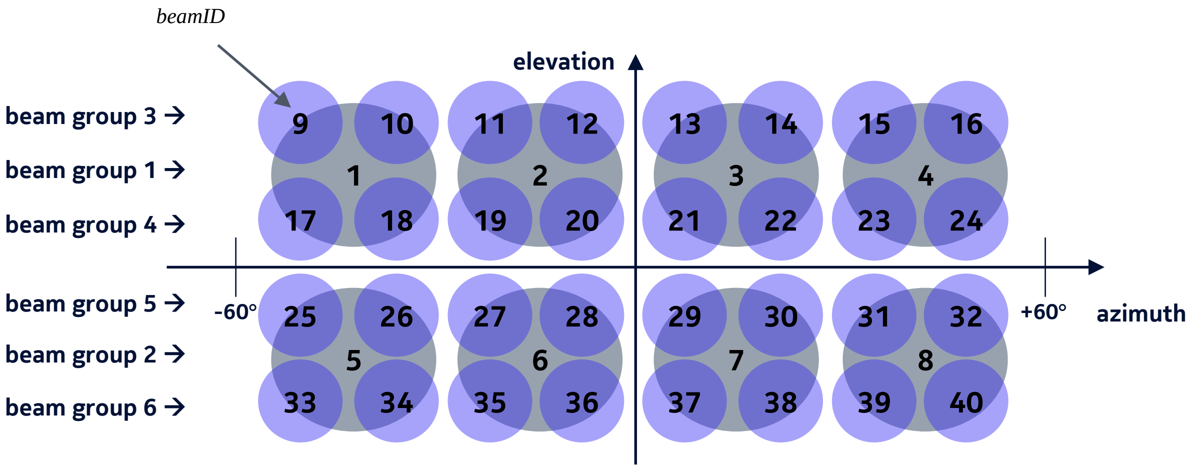
For this case the M-Plane must convey from the O-RU to the O-DU upon start-up as part of the O-RU self-description:

- the list of available beamIds and their characteristics, see chapter "10.4.1.1 Beam Characteristics" below.

Note that there is no requirement that beamIds be in sequential order or that there are no gaps between beamId values. The only constraint is that the beamId value of zero is reserved so cannot refer to any frequency-domain, time-domain or hybrid beam.

10.4.1.1 Beam Characteristics

In order to use predefined-beam beamforming in a standardized way, O-RAN considers beamforming to be defined such that energy (in the DL) or sensitivity (in the UL) is focused into either a "coarse" or "fine" granularity with possible overlaps. In this way "broadcast" beams may be used to cover a wider area with less power or sensitivity, while higher-power or higher-sensitivity beams may be used in e.g. a per-UE fashion. **Figure 10-2** shows an example of the assumed beam arrangement.



Note: ovals represent beams, numbers in ovals represent beamId

Figure 10-2 : Example of updatable-weight frequency-domain plus fixed time-domain beamforming

Figure 10-2 introduces the concept of "beam-group". A "beam-group" is the set of beamIds that may be used simultaneously for some reason. In **Figure 10-2** an assumption can be made that the elevation direction is time-domain while the azimuth direction is frequency-domain; the frequency-domain beams in the same time-domain group may be used at the same time but different time-domain groups must be separated in time.

Figure 10-2 also shows the need to describe beam overlap and beam adjacency. It is necessary, for example, to convey that beamId=1 is a coarse beam overlapping with fine beamIds={9,10,17,18} and that beamId=18 is a fine beam overlapping with coarse beamId=1. It is also necessary to convey that beamId=1 has as neighbors beamId={2,5,11,19,25,26} and that beamId=18 has as neighbors beamIds={2,5,10,17,19,26}. Overlapping beams must not be scheduled together to avoid interference, and neighboring beams should not scheduled together to avoid interference where possible.

The specific beam characteristics are therefore as follows, all per beamId:

1) beam-type for the subject beamId enumerated as COARSE-BEAM or FINE-BEAM;
    - there is no attempt to quantify "coarse" and fine" in terms of beamwidth, this is just a relative relationship
2) beam-group-id for the subject beamId as an integer;
3) coarse-fine-beam-relation as a list: if the subject beamId is coarse, this is a list of the associated fine beams, and if the subject beamId is fine, this is a list of the associated coarse beams;
4) neighbor-beam as a list: a list of all beams that may interfere with the subject beamId

10.4.2 Weight-based dynamic beamforming

Real-time-updated-weight-based beamforming operates the same as index-based beamforming, except that the need for the O-DU to convey actual beam weights to the O-RU introduces additional complexity.

10.4.2.1 Weight-based dynamic frequency-domain or time-domain beamforming (not hybrid)

In the case of either frequency-domain or time-domain beamforming wherein the beamforming weights can be updated in real-time and have a beamId value associated with the weights, the beamId is treated the same: it points to a set of weights that control the array elements' gain and phase and the number of weights equals the number of array elements. In many cases, the magnitude of each complex weight value will equal unity but this is not required; in particular "tapering" may require less-than-unity weight magnitudes for some array elements. The weight values prior to any compression will be fractional hence no I or Q value may exceed positive or negative unity.

The following list describes the information that the M-Plane must carry from the O-RU to the O-DU upon start-up as part of the O-RU self-description, the information listed is per array (so per tx-array or per rx-array):

1) Beamforming type, enumerated as "frequency", "time" or "hybrid"
2) Maximum number of weight-based beamId values supported (could be zero) : "numBeams"
    - O-RUs may have memory limitations that mean the number of beams is limited; zero means no weight-based beamforming is supported by this tx-array or rx-array
3) Initial value of weight-based beamId supported: "initBeamId"
    - Different ranges of beamId may support weight-based beamforming versus e.g. predefined beamforming
4) Frequency granularity of time-domain beamforming, enumerated as "per component carrier" or "per band".
    - Value is only present for time-domain beamforming
5) Time granularity of time-domain beamforming, enumerated as "per-OFDM-symbol" or "per-slot".
    - Value is only present for time-domain beamforming

Because the beams are to be generated by the O-DU the O-RU will not know the beam characteristics so they are not reported.

The actual number of weights K in the frequency-domain or time-domain weight vectors will be clear from the O-RU antenna model, see the Chapter 10.5 on that topic.

10.4.2.2 Weight-based dynamic hybrid beamforming

Here two sub-cases are considered, wherein for one sub-case both the frequency-domain and time-domain weights may be updated in real-time, and for the second sub-case the frequency-domain weights may be updated in real-time but the time-domain beams are fixed.

10.4.2.2.1 Hybrid beamforming with updatable frequency-domain and time-domain weights

For this sub-case the beamforming weight vector is a composite of the frequency-domain weights and the time-domain weights so can be considered as simply a longer weight vector. Where a block-based beam weight compression is employed (block floating point, block scaling or μ-law compression), the block size is a single beamforming weight vector (both frequency-domain and time-domain parts). The actual number of weights in the composite frequency-domain plus time-domain weight vectors (K' + K) will be clear from the O-RU antenna model, see the Chapter 10.5 on that topic.

The following list describes the information that the M-Plane must carry from the O-RU to the O-DU upon start-up as part of the O-RU self-description, the information listed is per array (so per tx-array or per rx-array):

1) Beamforming type, enumerated as "frequency", "time" or "hybrid" – here will be "hybrid"
2) Maximum number of weight-based beamId values supported (could be zero) : "numBeams"
   - O-RUs may have memory limitations that mean the number of beams is limited; zero means no weight-based beamforming is supported by this tx-array or rx-array
3) Initial value of weight-based beamId supported: "initBeamId"
   - Different ranges of beamId may support weight-based beamforming versus e.g. predefined beamforming
4) Frequency granularity of time-domain beamforming, enumerated as "per component carrier" or "per band".
5) Time granularity of time-domain beamforming, enumerated as "per-OFDM-symbol" or "per-slot".

Note that the number of time-domain beam weights associated with a given beamId is the same as the number of array elements which is K, but the number of frequency-domain weights is less, being K'. p' represents the dimensionality of the time-domain beamforming operation, so that K = K' * p'. The total length of the beamforming weight vector, including both the K' frequency-domain weights and the K time-domain weights, is K' + K = K' + (K' * p') = K' * (p'+1). **Figure 10-3** shows an example where K = 16, K' = 4 and p' = 4, and the length of the beamforming weight vector (frequency-domain and time-domain combined) is 20 complex weights.

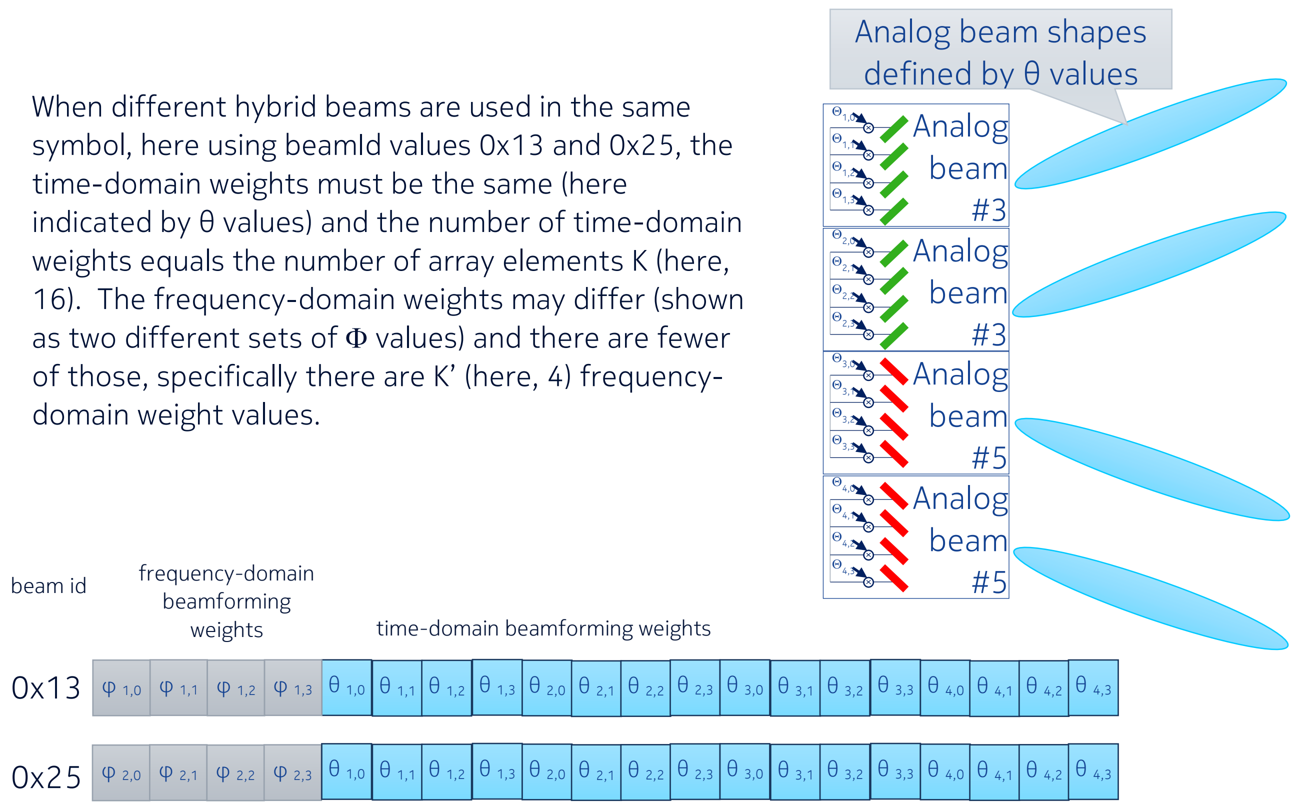


Figure 10-3 : Example of updatable-weight frequency-domain and time-domain beamforming

10.4.2.2.2 Hybrid beamforming with updatable frequency-domain weights and fixed time-domain beams

For this sub-case the beamforming weight vector is a composite of the frequency-domain weights and the time-domain beam numbers with the frequency-domain weights in the first half of the vector and the time-domain beam numbers in the second half of the vector. This vector must not be considered as simply a longer weight vector because the frequency-domain weights may be compressed but the time-domain beam numbers must not be compressed. Where a block-based beam weight compression is employed (block floating point, block scaling or μ-law compression), the block size is a single beamforming weight vector but only that half of the weight vector containing the frequency-domain weights. The remaining half of the vector containing the time-domain beam numbers may not be compressed and contains the integer time-domain beam numbers.

The actual number of weights in the composite frequency-domain weights plus time-domain beam-number vectors (K' and K) will be clear from the O-RU antenna model (see the Chapter 10.5 on that topic) with the number of frequency-domain weights K' indicating which elements in the vector are subject to compression (the first K' complex values in the vector).

The following list describes the information that the M-Plane must carry from the O-RU to the O-DU upon start-up as part of the O-RU self-description, the information listed is per array (so per tx-array or per rx-array):

1) Beamforming type, enumerated as "frequency", "time" or "hybrid" – here will be "hybrid"
2) Maximum number of weight-based beamId values supported (could be zero) : "numBeams"
    a. O-RUs may have memory limitations that mean the number of beams is limited; zero means no weigt-based beamforming is supported by this tx-array or rx-array
3) Initial value of weight-based beamId supported: "initBeamId"
    a. Different ranges of beamId may support weight-based beamforming versus e.g. predefined beamforming
4) Frequency granularity of time-domain beamforming, enumerated as "per component carrier" or "per band".
    a. Value is only present for time-domain beamforming
5) Time granularity of time-domain beamforming, enumerated as "per-OFDM-symbol" or "per-slot".
    a. Value is only present for time-domain beamforming
6) For each of the time-domain beams (the number is known from the O-RU antenna model), beam attributes (see 11.3.1.1)

Note that the number of time-domain beam numbers associated with a given beamId (K') will be the same as the number of frequency-domain beam weights for that beamId; this is because each frequency-domain beamforming weight is applied to a data stream that is subsequently time-domain beamformed using a specific beam number, so if there are e.g. K'=4 frequency-domain weights associated with a given beamId there will be four time-domain beam numbers also associated with that same beamId. See **Figure 10-4** for an example wherein four frequency-domain weights $\Phi_{x,y}$ are applied with four time-domain beamforming numbers (#3 and #5). Here the number of frequency-domain weights K' indicated by the antenna model would be "four" so the first four values in each vector would be complex fractional values and would be compressed, while the second four values would be real integers and not compressed.

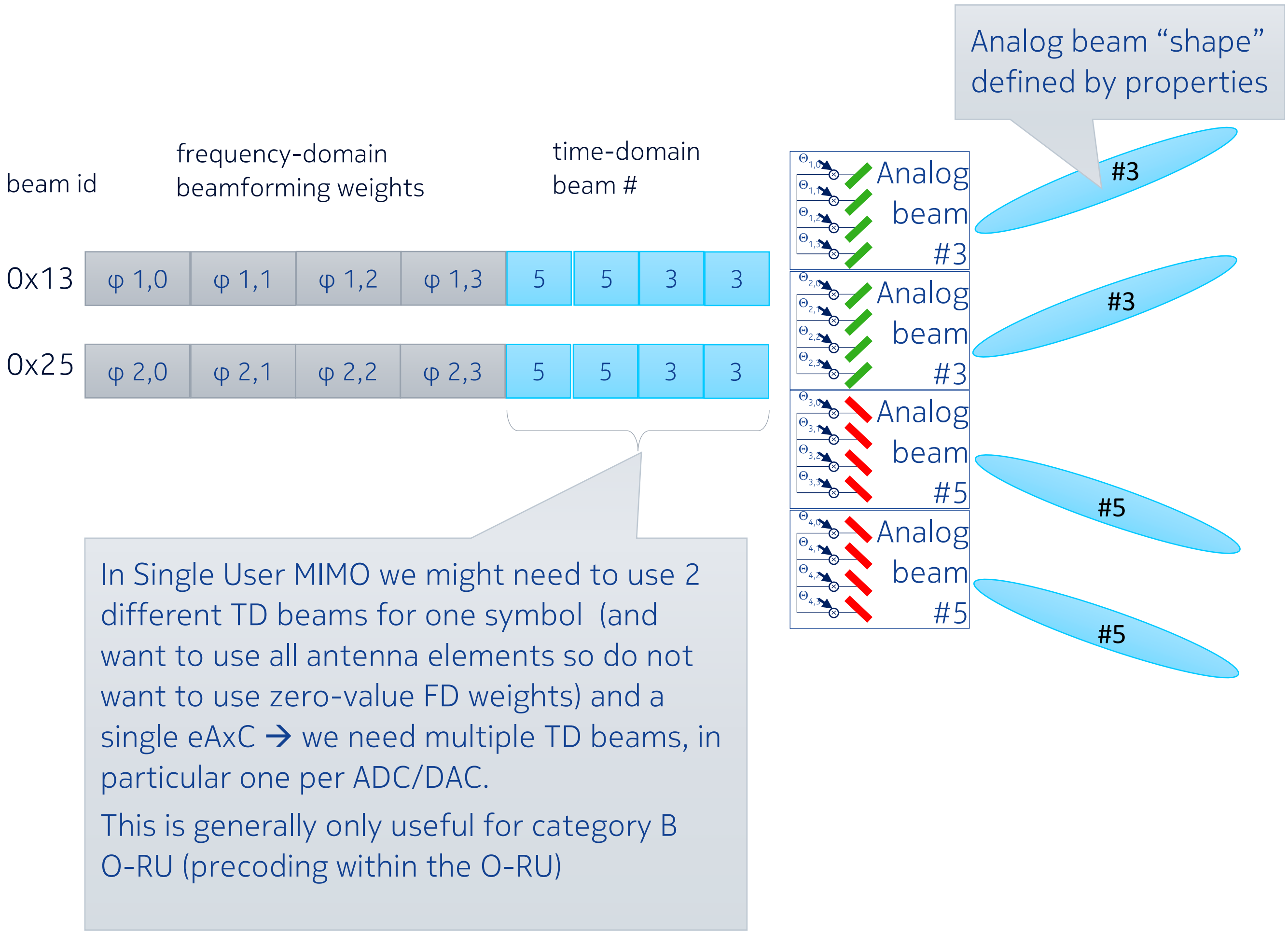


Figure 10-4 : Example of updatable-weight frequency-domain plus fixed time-domain beamforming

10.4.3 Attribute-based dynamic beamforming

Attribute-based dynamic beamforming operates similarly to weight-based dynamic beamforming except that hybrid beamforming is not possible, because the beam attributes are a result of the beamforming operation within the O-RU which will be either frequency-domain or time-domain but not a mixture. Also, instead of beamforming weights associated with a specific beamId being sent from O-DU to O-RU in section extension =1, beam attributes associated with a specific beamId are sent instead from the O-DU to the O-RU in a different section extension =2.

The following list describes the information that the M-Plane must carry from the O-RU to the O-DU upon start-up as part of the O-RU self-description:

1) Beamforming type, enumerated as "frequency" or "time" (not "hybrid")
2) Maximum number of beamId values supported (could be zero) : "numBeams"
    a. O-RUs may have memory limitations that mean the number of beams is limited; zero means no weight-based dynamic beamforming is supported by this tx-array or rx-array
3) Initial value of beamId supported: "initBeamId"
    a. Different ranges of beamId may support generated beam beamforming versus e.g. predefined beamforming
4) Valid range of bfAzPt (see chapter 5.4.7.2.2)
5) Valid range of bfZePt (see chapter 5.4.7.2.3)
6) Valid range of bfAz3dd (see chapter 5.4.7.2.4)
7) Valid range of bfZe3dd (see chapter 5.4.7.2.5)
8) Valid range of bfAzSl (see chapter 5.4.7.2.6)
9) Valid range of bfZeSl (see chapter 5.4.7.2.7)

10.4.4 Channel-information-based beamforming

As stated earlier, beamId is irrelevant and unused in the case of channel-information-based beamforming.

10.5 O-RU Antenna Model supported by O-RAN

Knowledge of O-RU antenna model is critical for certain types of beamforming. The following model is applicable for O-RU with one or more antennas, where each antenna has array of elements that are

- uniform (all elements have same properties) and
- organized into rectangular array (with rows and columns) that is planar (flat).

O-RU exposes via M-plane logical model of O-RU consisting of one or more arrays compose of one or more array elements. Array element represents independently controllable entity including one or more radiating elements and related RF processing elements. Note RX and TX are in general independently controllable for that in the model TX and RX arrays are described as separate entities. If O-RU supports beamforming, then beamforming is realized within each array separately i.e. beamforming weight vector is applicable within one array. One or more arrays can occupy same physical location e.g. RX array and TX array that use same set of radiators.

Beamforming methods that use dynamic beamforming with beamforming weights conveyed in C-plane message (in contrast to predefined beams) require the O-DU to know antenna properties. Different beamforming methods require knowledge of different subsets of antenna properties.

10.5.1 Coordinate Systems

Some of parameters describing model of antenna related to coordinate system that defines three axes and three angles.

There are two coordinate systems defined:

- array coordinate system
- O-RU coordinate system

The array coordinate system is presented below:

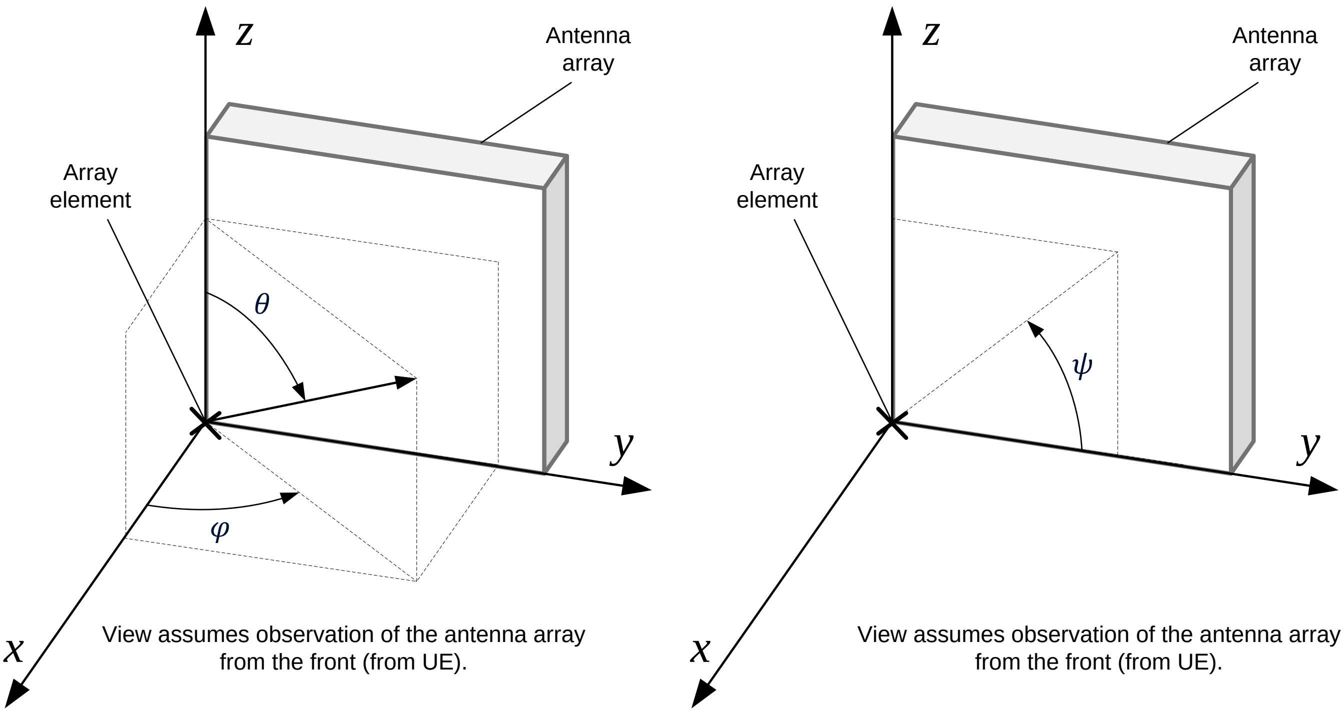


Figure 10-5 : Array Coordinate System

The diagram presents view from the front of array-panel (from UE). Arrows indicate increasing values of coordinates.

x points towards broad-side
y increases to left, with antenna-array's columns
z points towards zenith
φ (phi) is azimuth angle, counter-clockwise rotation around z-axis, 0° points to broad-side, 90° points to y-axis
θ (theta) is zenith angle, counter-clockwise rotation around y-axis, 0° points to zenith, 90° points to horizon
ψ (psi) is angle, counter-clockwise rotation around x-axis, 0° points to horizon, 90° points to zenith

The **array coordinate system** is centered on centre of the leftmost, bottom element of array. The array coordinate system is applicable within one array.

The **O-RU coordinate system** is the selected array coordinate system of an O-RU. The selection is fixed by O-RU design. The O-RU coordinate system is applicable within one O-RU.

10.5.2 O-RU Antenna Model Parameters

The O-RU antenna model can be described with following parameters:

- **K** – number of array elements in array (note that K = M*N*P*Q)
- **M** – number of rows of array elements in array. M>0; value 0 is reserved for future use.
- **N** – number of columns of array elements in array. N>0; value 0 is reserved for future use.
- **P** – number of polarizations in array. P>0, value 0 is reserved for future use.
- **Q** – number of overlapping array elements (array-layers) in array. Each array-layer has M rows, N columns and P polarizations. Q>0; value 0 is reserved for future use. See Annex K for more information on array-layers.
- **x**, **y**, **z** – position of centre of the leftmost, bottom element of array in O-RU coordinate system
- **φ**, **θ** direction of normal vector perpendicular to array's surface in O-RU coordinate system (array's normal vector corresponds to x axis in array's coordinate system)
- **dy** – mean distance between centres of nearby array elements in y direction in array coordinate system (distance between columns); value 0 is reserved for future use.
- **dz** – mean distance between centres of nearby array elements in z direction in array coordinate system (distance between rows); value 0 is reserved for future use.
- **list of polarizations in array** (this list has P elements, each representing p-th polarization); values ordered in ascending order of angle. Example: cross-polarized array having elements of one of two linear polarizations can be described by list: (-45°, +45°) indicating that array element with polarization index p=0 has linear polarization -45°, and array element with polarization index p=0 has linear polarization +45°.
- **independent power budget** per layer - in case of an array with multiple layers, corresponding elements (located in same row and column and same polarization) of different layers may have a shared power budget or have independent power budgets.

For an array supporting hybrid beamforming (see section 10.4.2) there is a need for additional parameters:

- **K'** – number of frequency domain beamforming weights $\varphi_{k'}$ that can be applied within the array. 0<K'≤K; value 0 is reserved for array not supporting hybrid beamforming.
- **h**(k) – mapping of array element k to frequency domain beamforming weight $\varphi_{k'}$ where k'=h(k).
  The mapping is represented as a list of lists: for every 0<k'<K' a list of K/K' numbers identifying array elements where frequency domain beamforming weight $\varphi_{k'}$ is applied. k'=h(k) if number k is in the list corresponding to k'.
  Section 10.5.3 Identification and Ordering of Array Elements describes how numbers are assigned to array elements.

The model assumes the number of array elements corresponding to frequency domain beamforming weight $\varphi_{k'}$ is the same for every k' (0<k'<K') and the elements corresponding to beamforming weights form a rectangular shape without overlapping i.e. every array element is linked with exactly one frequency domain beamforming weight $\varphi_{k'}$.

In addition, the O-RU antenna model provides parameters describing key capabilities of array elements. The model assumes the array is uniform and all elements have the same properties. Each single value is applicable to all elements within the array.

Parameters describing array elements applicable to TX and RX arrays:

- horizonal plane half power (-3 dB) beam width of array element's radiation pattern
- vertical plane half power (-3 dB) beam width of array element's radiation pattern
- horizonal plane quarter power (-6 dB) beam width of array element's radiation pattern
- vertical plane quarter power (-6 dB) beam width of array element's radiation pattern

Beam widths above are angles (expressed in degrees) between half-power (-3 dB) points or quarter-power (-6 dB) points respectively of the main lobe with reference to peak radiated power of main lobe. Horizontal and vertical plane correspond to the xy-plane and xz-plane respectively of the array in the array coordinate system.

The parameter describing array elements specific for TX array:

- $\boldsymbol{m_{a,k}}$ - max rms power rating of array element of the array. Usage of max rms power rating is described in section 6.1.3.3.

10.5.3 Identification and Ordering of Array Elements

In many applications there is a need to assign to array element a number k such that 0≤k<K. One example is mapping position (represented by k, such that 0≤k<K) of beamforming weight in beamforming vector to array element. Other example is identification of array elements in antenna model.

For purpose of identification and ordering a number k is assigned to each element of array by the function f(m,n,p.q):

$$f(m,n,p,q) = q \cdot Q \cdot M \cdot N + p \cdot M \cdot N + m \cdot N + n$$

where:

m - row (bottom to top), 0≤m<M
n - column (left to right, view from the front of array), 0≤n<N
p - polarization index, 0≤p<P; polarization value of polarization index p is ψ_p
q – array-layer, 0≤q<Q

Note that for a rectangular array, the function f(m,n,p,q) can be inverted allowing to specify a "tuple" (m,n,p,q) of k-th element.

10.5.4 Relations Between Array Elements

Beamforming methods that use dynamic beamforming with beamforming weights conveyed in C-plane messages (in contrast to predefined beams) require the O-DU to know that specific elements of one array is co-located with elements of another array e.g. RX array and TX array that use same set of radiators. In addition, one or more TX arrays may share elements and parts of RF processing paths (e.g. a power amplifier) resulting in a shared power budget described by a maximum rms power rating.

If element k_a of array A and element k_b of array B are in same position (same physical row and column) then k_a and k_b are co-located.

If element k_a of array A and element k_b of array B are co-located and share a power budget, then k_a and k_b are shared.

Relation of co-location is symmetric: k_a and k_b are co-located if and only if k_b and k_a are co-located.

Relation of co-location is transitive: if k_a and k_b are co-located and k_b and k_c are co-located then k_a and k_c are co-located.

Relation of sharing implies co-location: if k_a and k_b are shared then k_a and k_b are co-located.

O-RU shall report via M-plane relations between array elements. O-RU shall avoid reporting redundant relations that can be derived from other relations by symmetric and transitive properties and implication of co-location relation by sharing relation. In addition, the O-RU shall provide a concise representation of the common case of two arrays that have all elements in relation (e.g. RX array of -45° polarization and corresponding RX array of +45° polarization).

10.5.5 Model Usage

The O-RU antenna model reported by the O-RU consists of RX arrays and TX arrays. RX arrays and TX arrays represent a capability for transmitting/receiving RF signal related to an eAxC and - if beamforming is supported by O-

RU on given array - beamforming capability. In this section examples are presented: red and green bars represent array elements of different polarizations, grey box represents physical device, white rectangles represent arrays reported by O-RU.

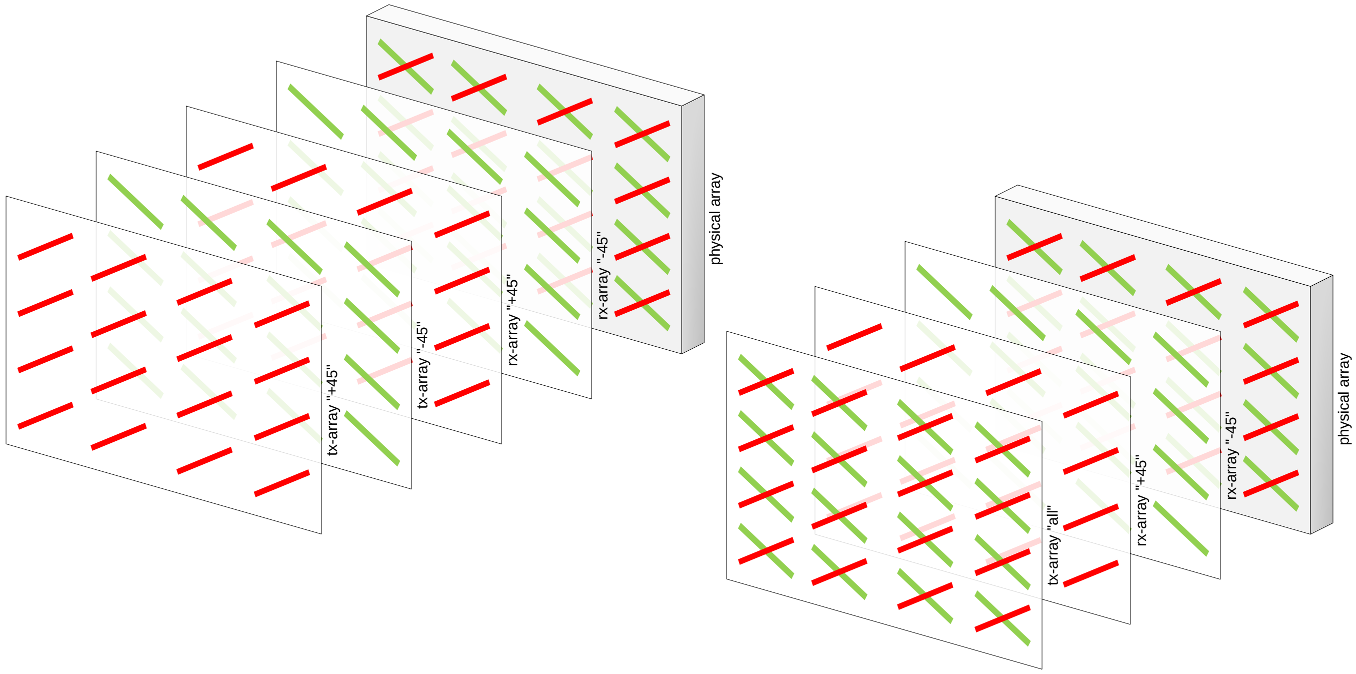


Figure 10-6 : Examples of Model Usage – TX as two single-polarization arrays or one cross-polarized array

As an example **Figure 10-6** presents two O-RU designs: an O-RU with two TX arrays each of one polarization and an O-RU with one TX array of two polarizations (note number and dimension of TX array has an impact on the size of beamforming vectors). Of course, an O-RU that combines both above designs is possible as presented in **Figure 10-7**.

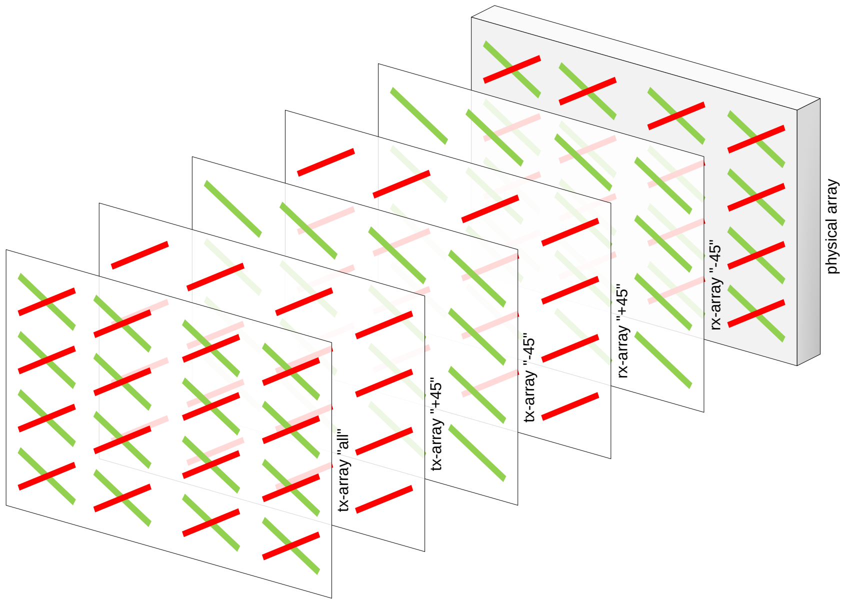


Figure 10-7 : Examples of Model Usage – TX as two single-polarization arrays and one cross-polarized array

Similarly, in RX – if an O-RU does not support the collecting raw SRS by use of beams designed to mute signal from all elements except one then that O-RU – in addition to full RX array – may expose RX arrays with single elements as presented in **Figure 10-8**

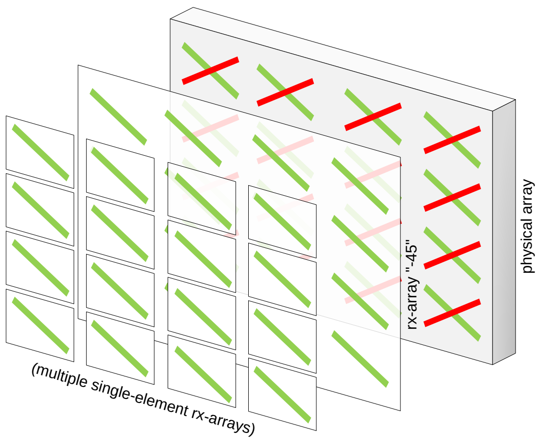


Figure 10-8 : Example of Model Usage – RX with multi-element and multiple single-element arrays

An O-RU that does not support beamforming can be represented with arrays having one element each. Note that a category B O-RU could be represented with an array with two elements to accommodate two polarizations while a category A O-RU would generally be represented with a TX array with only one polarization (P=1). **Figure 10-9** presents the two design examples of a non-beamforming O-RU: an O-RU with two TX arrays each of one polarization and an O-RU with one TX array of two polarizations. Of course, an O-RU that combines both designs is also possible.

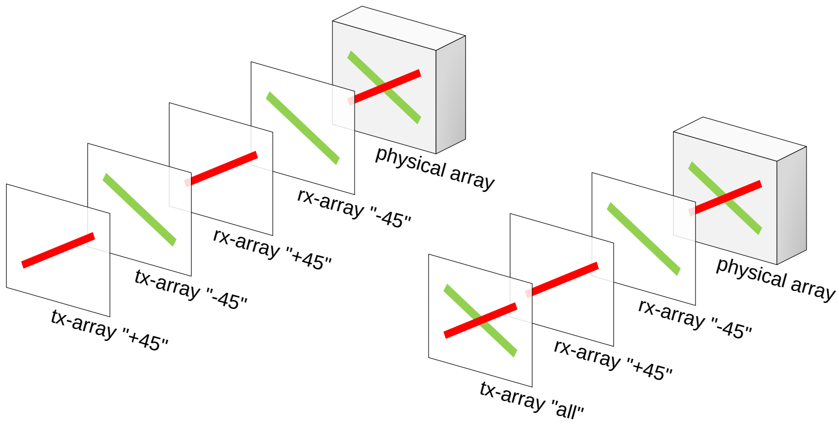


Figure 10-9 : Examples of Model Usage – Non-beamforming O-RU with TX as two single-polarization arrays or one cross-polarized array

Annex A Compression Methods

A.1 Block Floating Point Compression

The compressed data representation is as follows. For each PRB, In-phase (I) and Quadrature (Q) samples are converted into floating point format. Subsequently, the samples are presented as a compressed bit sign and a mantissa (e.g., 9-bit) and a shared exponent (see **Figure A-1**). The compression procedure receives 12 subcarriers with 24 uncompressed I and Q samples. The I and Q samples are subsequently compressed to a bit signed mantissa and unsigned exponent. Further, the exponent is included for each compression block to be sent per PRB (see **Figure A-2**).

Note: Mantissa bitwidths are specified in the compHdr field of the relevant U-Plane or C-Plane message (range 1-16 bitwidth). However, support on O-RU HW for particular mantissa bitwidths is to be defined on individual basis. That is, there is no requirement on O-RU HW to support all possible mantissa bitwidths.

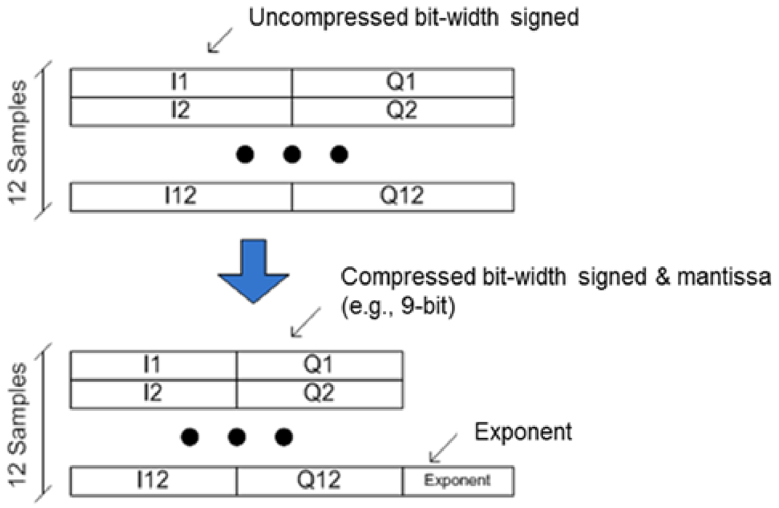


Figure A-1 : Block Floating Point Compression data representation

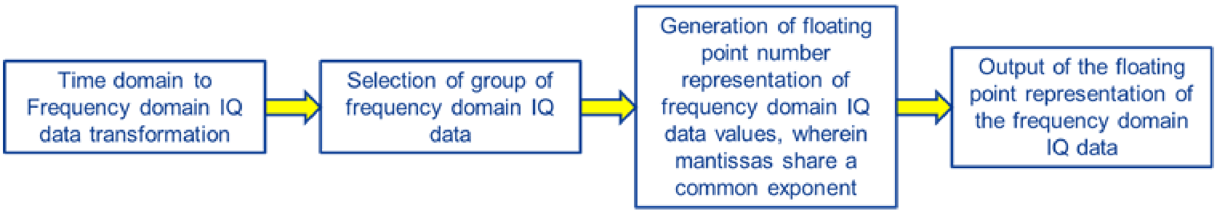


Figure A-2 : Block Floating Point Compression process overview

A.1.1 Block Floating Point Compression Algorithm

The following pseudo code depicts an example reference implementation of the compression algorithm.

Inputs:

- fPRB - Original physical resource block (PRB), 12 complex resource elements with "native" word length of the implementation e.g. 24 bits (UL should use as accurate value as possible from FFT & beamforming)
- iqWidth - Word length after compression (includes sign bit)

Outputs:

- cPRB - Compressed PRB, 12 complex resource elements with word length iqWidth
- exponent - Common exponent for compressed PRB

// Find max and min
maxV = max(Re(fPRB), Im(fPRB)), *minV* = min(Re(fPRB), Im(fPRB))

// Determine max absolute value
maxValue = max(*maxV*, |*minV*|-1) (msb of negative value can be one higher)

// Calculate exponent
raw_exp = [floor($\log_2$(*maxValue*) +1)] (msb of *maxValue*)

// Calculate shift value and limit to positive
exponent = max(*raw_exp* – iqWidth + 1, 0)

```
// Determine right shift value
scaler = 2^-exponent

For iRe = 1:length(fPRB)
    //Scale and round:
    Re(cPRB(iRE)) =  Quantize (scaler × Re(fPRB(iRE)))  /* mult. could be bit-shift, Quantize could be or-round */
    Im(cPRB(iRE)) = Quantize (scaler × Im(fPRB(iRE)))  /* mult. could be bit-shift, Quantize could be or-round */
End
```

A.1.2 Block Floating Point Decompression Algorithm

The following pseudo code depicts an example reference implementation of the decompression algorithm.

Inputs:

- cPRB - Compressed PRB, 12 complex resource elements with word length WL
- exponent - Common exponent for compressed PRB

Outputs:

- fPRB - Decompressed physical resource block (PRB), 12 complex resource elements with “native” word length for further processing. For example, 24 bits or 32 bits

```
//Determine scaler
scaler = 2^exponent

For iRe = 1:length(cPRB)
    //Scale
    Re(fPRB(iRE)) = scaler × Re(cPRB(iRE))  /* this could be replaced with a bit-shift operation */
    Im(fPRB(iRE)) = scaler × Im(cPRB(iRE))  /* this could be replaced with a bit-shift operation */
End
```

A.2 Block Scaling Compression

A block scaling algorithm is proposed which is similar in concept to the Block Floating Point representation except that instead of data being represented by mantissa values and exponent shared within the block, data is instead represented by post-scaled values and a multiplicative scale value shared within the block. It is proposed in the specification that the data block size for this function is a single PRB, same as for the proposed Block Floating Point representation. The following **Figure A-3** shows the algorithm in principle (assuming an 8-bit scaler value).

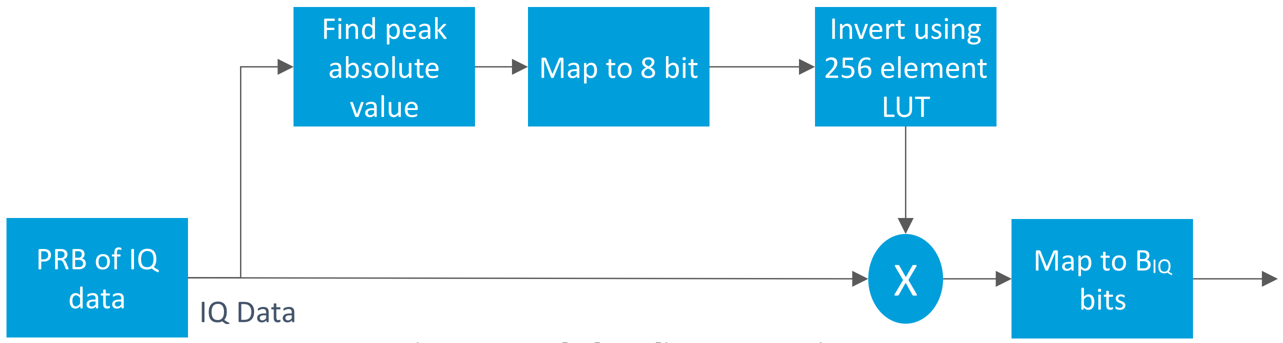


Figure A-3 : Block Scaling Process Diagram

A.2.1 Block Scaling Compression Algorithm

The following pseudo code depicts an example implementation of the compression algorithm.

Inputs:

- fPRB - Original physical resource block (PRB), 12 complex resource elements with “native” word length of the implementation e.g. 24 bits (UL should use as accurate value as possible from FFT & beamforming)
- iqWidth - Word length after compression (includes sign bit)

Outputs:

- cPRB - Compressed PRB, 12 complex resource elements with word length iqWidth
- blockScaler- Common scaler for compressed PRB

// Find max and min
maxV = max(Re(fPRB), Im(fPRB)), *minV* = min(Re(fPRB), Im(fPRB))

// Determine max absolute value
maxValue = max(*maxV*, |*minV*|-1) (msb of negative value can be one higher)

// Calculate scaler
blockScaler = Inverse(*maxValue) /* blockScaler can be chosen to be smaller than Inverse(maxValue). Inverse can be implemented via look up table or other methods. */*

For iRe = 1:length(fPRB)

    //Scale and round:

    *Re(*cPRB(iRE)) = Quantize (*blockScaler* × Re(fPRB(iRE))) /* Quantize could be truncate or round */

    *Im(*cPRB(iRE)) = Quantize (*blockScaler*× Im(fPRB(iRE))) /* Quantize could be truncate or round */

End

A.2.2 Block Scaling Decompression Algorithm

The following pseudo-code depicts an example reference implementation of the block scaling decompression algorithm.

Inputs:

- cPRB - Compressed PRB, 12 complex resource elements with word length WL
- blockScaler - Common scaler for compressed PRB

Outputs:

- fPRB - Decompressed physical resource block (PRB), 12 complex resource elements with “native” word length for further processing. For example 24 bits or 32 bits

For iRe = 1:length(cPRB)

    //Scale

    *Re(*fPRB(iRE)) = *blockScaler* × Re(cPRB(iRE)) /* this could be replaced with a bit-shift operation */

    *Im(*fPRB(iRE)) = *blockScaler* × Im(cPRB(iRE)) /* this could be replaced with a bit-shift operation */

End

A.3 μ-Law Compression

A.3.1 μ-Law Compression Algorithm

Inputs:

- prbI & prbQ – Original physical resource block (PRB), 12 complex resource elements with a word length of 16-bits I and 16-bits Q. The input bit width is fixed to 16-bits.
- compBitWidth – the length of I bits and the length of Q bits after compression over the entire PRB. Note that this means that the μ-law compression is really considered as a “block” compression with the block size being one PRB (same as for block floating point and block scaling).

Outputs:

- compI & compQ – compressed PRB, 12 complex resource elements with word length compBitWidth, including sign, exponent and mantissa.
- compShift – the shift applied to the entire PRB.

The O-RAN μ-law compression method combines a simple bit shift operation (for dynamic range) with a nonlinear piece wise approximation of μ-law compression where for implementation efficiency, μ=8 and the sign & mantissa are 1 and 2-bits respectively.

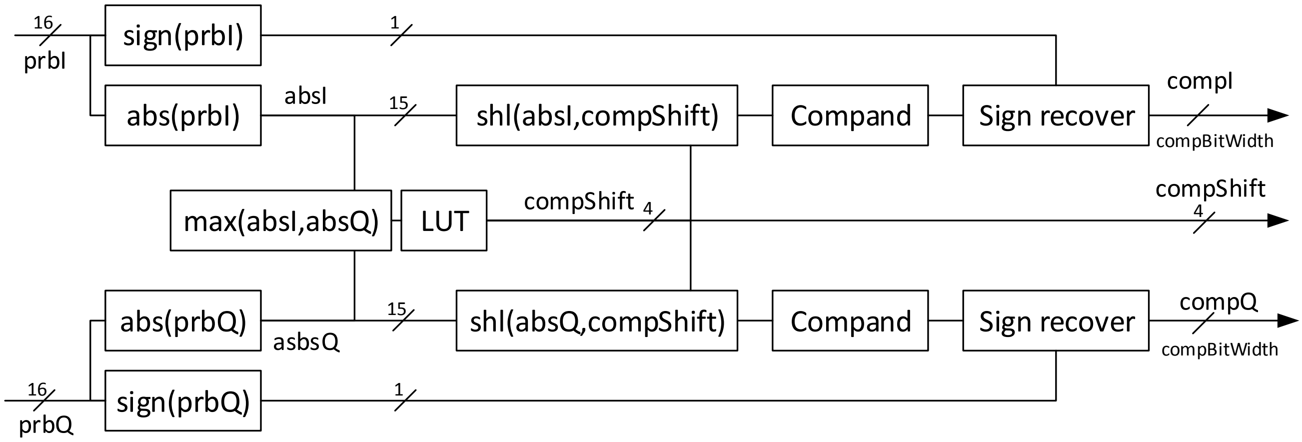


Figure A-4 : μ-Law Compression Algorithm

```
// extract the sign bit and absolute value for the PRB

signI=sign(prbI), signQ=sign(prbQ), absI=abs(prbI ), absQ=abs(prbQ)

// Find the maximum in absI and absQ

maxVal=max(absI,absQ)

// Determine compShift, the shift to be applied to the entire PRB

if maxVal>=2^14 then compShift=0
if maxVal<2^14 then compShift=1
if maxVal<2^13 then compShift=2
if maxVal<2^12 then compShift=3
if maxVal<2^11 then compShift=4
if maxVal<2^10 then compShift=5
if maxVal<2^9   then compShift=6
if maxVal<2^8   then compShift=7

// Apply round and shift right (shr – make greater)
absI = shr(absI,compShift)
absQ= shr(absQ,compShift)

// compand each sample, absBitWidth=15

if absI(i) >   (2^absBitWidth-1) then compI(i) = (2^absBitWidth-1) // saturate
if absI(i) <= 2^(absBitWidth-2) then
     compI(i) = absI(i)/2^(absBitWidth- compBitWidth )
```

```
elseif absI(i <= 2^(absBitWidth-1) then
    compI(i) = absI(i)/2^(absBitWidth- compBitWidth +1) + 2^( compBitWidth -3)
else
    compI(i) = absI(i)/2^(absBitWidth- compBitWidth +2) + 2^( compBitWidth -2)
end

if absQ(i) >   (2^absBitWidth-1) then compQ (i) = (2^absBitWidth-1)
if absQ (i) <= 2^(absBitWidth-2) then
    compQ(i) = absQ (i)/2^(absBitWidth- compBitWidth )
elseif absQ (i <= 2^(absBitWidth-1) then
    compQ (i) = absQ (i)/2^(absBitWidth- compBitWidth +1) + 2^( compBitWidth -3)
else
    compQ (i) = absQ (i)/2^(absBitWidth- compBitWidth +2) + 2^( compBitWidth -2)
end

// re-apply sign

compI = round(signI .* compI);
compQ = round(signQ .* compQ);
```

A.3.2 μ-Law Decompression Algorithm

Inputs:

- compI & compQ– compressed physical resource block (PRB), 12 complex resource elements with a word length of compBitWidth bits I and compBitWidth bits Q.
- compShift – the shift applied to the entire PRB.

Outputs:

- decompI & decompQ – the decompressed PRB, 12 complex resource elements with word length fixed to 16-bits I and 16-bits Q, including sign, exponent and mantissa.

The O-RAN μ-law decompression method is a logical reverse function of the compression method.

```
// extract the sign bit and absolute value for the PRB

signI=sign(compI), signQ=sign(compQ), absI=abs(compI), absQ=abs(compQ)

// decompand each sample, absBitWidth=15

if absI(i) > (2^( compBitWidth -1)-1) then absI (i) = (2^( compBitWidth -1)-1)
if absI (i) <= 2^( compBitWidth -2) then
    decompI(i) = absI (i)*2^(absBitWidth- compBitWidth )
elseif absI (i) <= (2^( compBitWidth -2) + 2^( compBitWidth -3)) then
    decompI(i) = absI (i)*2^(absBitWidth- compBitWidth +1) - 2^13
else
     decompI(i) = absI (i)*2^(absBitWidth- compBitWidth +2) - 2^15
end

if absQ(i) > (2^( compBitWidth -1)-1) then
    absQ (i) = (2^( compBitWidth -1)-1)
if absQ (i) <= 2^( compBitWidth -2) then
     decompQ(i) = absQ (i)*2^(absBitWidth- compBitWidth )
elseif absQ (i) <= (2^( compBitWidth -2) + 2^( compBitWidth -3)) then
    decompQ(i) = absQ (i)*2^(absBitWidth- compBitWidth +1) - 2^13
else
     decompQ(i) = absQ (i)*2^(absBitWidth- compBitWidth +2) - 2^15
end;

//Apply sign and shift
decompI = signI * decompI
```

```
decompQ = signQ .* decompQ
decompI = decompI/2^compShift
decompQ = decompQ/2^compShift
```

A.3.3 μ-Law udCompParam and IQ data format

PRB fields are populated as follows:

- udCompParam (8 bits)
  - compBitWidth, 4 bits, (MSB)
  - compShift, 4-bits, (LSB)
- IQ samples, total bits = 12x 2x compBitWidth
  - 1st sample I, compBitWidth-bits
  - 1st sample Q, compBitWidth-bits
  - 2nd sample I, compBitWidth-bits
  - 2nd sample Q, compBitWidth-bits
  - …
  - 12th sample I, compBitWidth-bits
  - 12th sample Q, compBitWidth-bits

A.4 Beamspace Compression and Decompression

This compression algorithm is specific to beamforming weights and is not suitable for user or control IQ data. Hence this compression method will only be used as part of the bfwCompMeth in the C-Plane.

A.4.1 Beamspace Compression Algorithm

The following pseudo code depicts an example reference implementation of the compression algorithm.

Inputs:

- fBV - Original beamforming vector of N complex elements. N is the number of digital antenna ports supported by the O-RU and is communicated to the DU during startup by the OAM subsystem. Each element is a complex number with a native bitwidth e.g. 16-bit I, and 16-bit Q.
- iqWidth - Word length of each I and Q value after compression (includes sign bit)

Outputs:

- cBV - Compressed beamforming coefficients
- blockScaler- Common scaler for compressed beamforming coefficients
- activeBeamspaceCoefficientMask – active beamspace coefficient indices associated with the compressed beamforming vector

```
// Generate DFT basis matrix
for k = 1 to N
        for l = 1 to NW (k,l) = exp(i*2*pi*k*l/(N))  // W is a N x N complex matrix
    end for
end for
// Transform into beamspace
cBV = W*fBV // multiplication of a NxN complex matrix with a Nx1 complex vector yields another complex vector.

/* The algorithm is initialized to assume that all Beamspace Coefficients are transmitted across the fronthaul link. */
for k = 1 to N
```

```
    activeBeamspaceCoefficientMask(k) = 1
end for

/* At this stage some of the beamspace coefficients may be removed from the vector of coefficients to transmit across the fronthaul. In this example implementation, if the absolute value of a beamspace coefficient is less than 'threshold', it is deemed inactive, i.e. the activeBeamspaceCoefficientMask is '0' at that coefficient index and this index is not sent across the fronthaul. The decompression algorithm will assume a value of 0 for that coefficient. The value of threshold can be chosen by the implementer. Other methods to determine active or inactive beamspace coefficients are also allowed and do not violate the specification. */

t = 0
for k = 1 to N
    if abs(cBV(k)) < threshold
        activeIndex(k) = 0
        cBV(k) = null          // remove the element from the vector
    else
        activeIndex(k) = 1
        t = t + 1
    end if
end for
T = t
// Calculate scaler
maxValue = max(abs(Re(cBV)),abs(Im(cBV)))

blockScaler = Inverse(maxValue) /* scaler can be chosen to be smaller than Inverse(maxValue). Inverse can be implemented via look up table or other methods. */
For iRe = 1 to T
    //Scale and round:
    Re(cBV(iRE)) =  Quantize (blockScaler × Re(cBV(iRE)))  /* Quantize could be truncate or round */
    Im(cBV(iRE)) = Quantize (blockScaler × Im(fBV(iRE)))  /* Quantize could be truncate or round */
End
```

A.4.2 Beamspace Decompression Algorithm

The following pseudo-code depicts an example reference implementation of the block scaling decompression algorithm.

Inputs:

- cBV - Compressed beamforming coefficients
- blockScaler- Common scaler for compressed beamforming coefficients
- activeBeamspaceCoefficientMask – active beamspace indices associated with the compressed beamforming vector

Outputs:

- fBV – Decompressed beamforming vector of N complex elements.

```
m = 0
```

```
for k = 1 to N
    if activeBeamspaceCoefficientMask (k) = 1
    //Scale
        fBSC(k) = blockScaler × cBV(m)
        m = m + 1
    else
        fBSC(k) = 0
    end if
end for
// Generate DFT basis matrix
for k = 1 to N
    for l = 1 to N
        W (k,l) = exp(-i*2*pi*k*l/(N))  // W is a N x N complex matrix
    end for
end for
fBV = W * fBSC
```

A.5 Modulation Compression

Modulation compression is an IQ data compression method that may be applied to DL data only and depends on the observation that modulated data symbols may be represented by a very limited number of I and Q bits. For example, a QPSK modulated symbol has only two potential states of I and two potential states of Q, so such a symbol may be represented with no loss of information with a single bit of I and a single bit of Q. Likewise, a 64QAM constellation point (16x16 constellation) may be represented by at most 4 bits of I and 4 bits of Q. This allows a dramatic reduction in DL throughput and approximates that achieved by a 7-3 DL split. See the two figures below for a description of this concept.

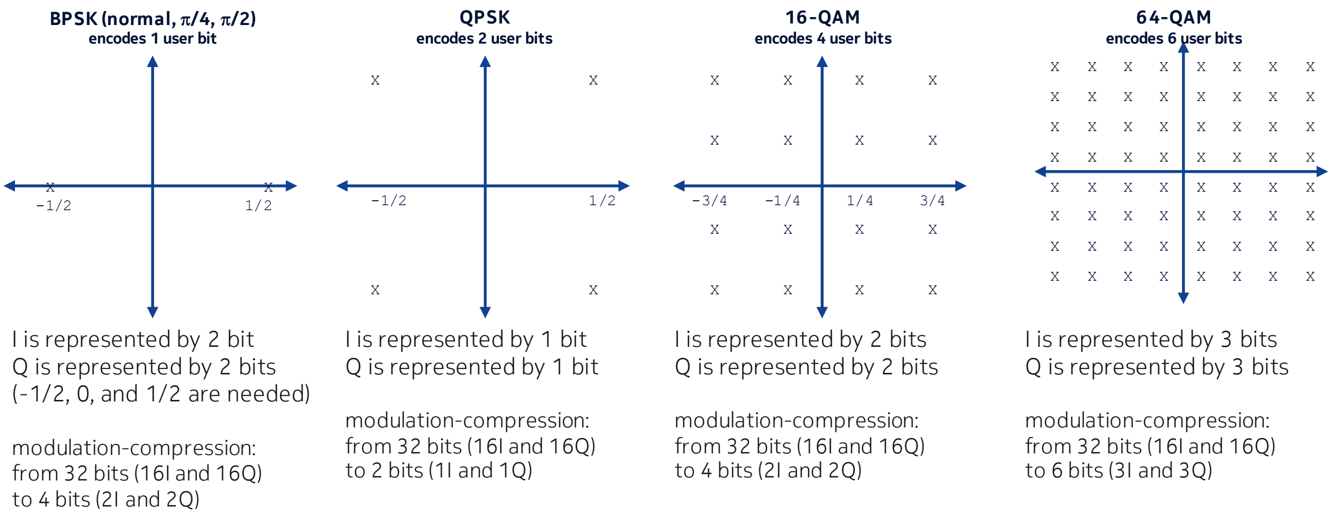


Figure A-5 : Several Constellation types

To represent the constellation points as I and Q values that also overlap allowing multiple constellation sizes to be represented by a single word-width, the constellations are "shifted" to allow a twos-complement I and Q value to represent any constellation point. The figure below shows the same constellations after shifting.

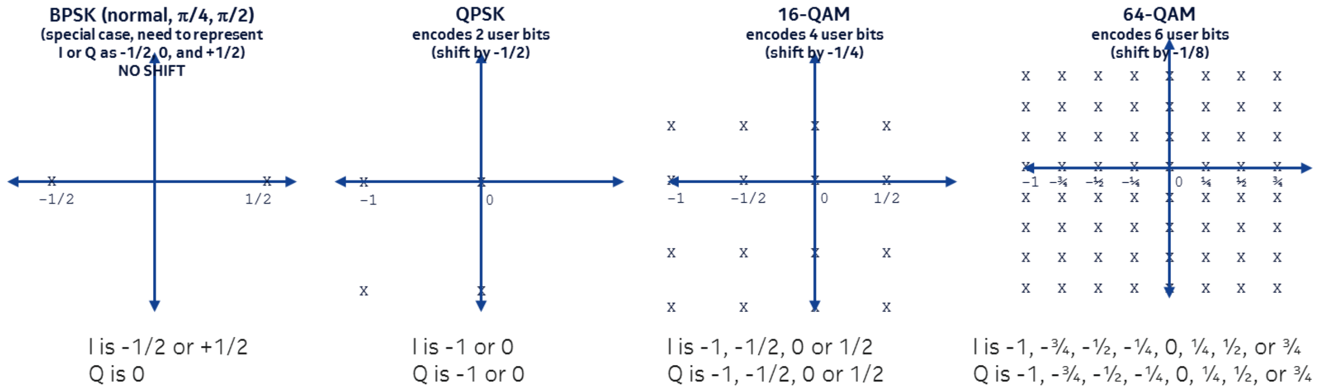


Figure A-6 : Shifted Constellation Points

Once the constellations are shifted, the I and Q values may be encoded in a limited number of bits, being the larger number needed to represent the largest constellation possible in the compression block (the data section). This means that if some data in the section use 64QAM and others use QPSK (e.g. a reference RE) all REs would use the largest needed representation which in this example is 4 bits for I and 4 bits for Q (for 64QAM). This spoils the compression efficiency a little bit but because reference REs are a small fraction of the total number of REs, the efficiency degradation is small. Note that in general, every user will have its own data section (and own beamforming index) so users with high-order modulation may need and use more bits of I and Q while users with lower-order modulation may need and use fewer bits of I and Q.

Note that some constellations should not be shifted. For example, BPSK needs I and Q data to take the values -1/2, zero, and +1/2 (different varieties of BPSK can rotate these as with π/4 BPSK and π/2 BPSK). For this reason, BPSK would use two bits for I and 2 bits for Q; while this seems counterintuitive (BPSK using more bits than QPSK) this is a small penalty given the rarity of BPSK as a modulation type. Here, BPSK would not be shifted. Likewise, PHICH constellations encode 3 states for each I and Q: -1, zero and 1. For this constellation the representation would be -½, 0 and ½ with no constellation shift needed. However, all "QAM" modulations do need a constellation shift. The presence of absence of the need for a constellation shift is indicated by the "csf" field, where for every "one" in the reMask "csf" indicates whether to shift (csf=1) or not (csf=0) the associated RE.

When decompressing, the O-RU must "unshift" the constellation (or perhaps not, depending on "csf") and also apply a scale factor for the constellation types represented in the section. There are expected to be either one or two modulation types in the section, no more. The modulation type is inferred from the reMask bits, where each "one" bit indicates the shift command ("csf") and scale factor ("modCompScaler" when using extension type 4, and "mcScaleOffset" when using extension type 5) for the REs in the subject PRB. The scale factor allows not only for correcting for different constellation scaling (e.g. for multiplexed channel data in a PRB inclusing QPSK and 16QAM, QPSK involves a 2/sqrt(2) factor while 16QAM involves a 4/sqrt(10) factor), but also allows different channel power scaling which is permitted as a 3GPP option.

Note that this compression method is essentially lossless, except that the scale factors, being 15 bits, impose a limit on the accuracy of representation. 15 bits is considered sufficient for all LTE and NR data representations.

Here is the specific decompression algorithm intended by this approach:

1. Read $iqSample$ as a N bit vector in the U-plane message [this is all the IQ data in the data section]

2. Map $iqSample\ [0,2^{N-1}]$ to $iqSampleFx\ [-1,1)$ assuming that the N bits are represented as Q1.(N-1) [this is the normal twos-complement representation of the I and Q samples represented in fractional notation].

3X. For each RE in the PRB (using section extention =4):

 3Xa: fetch the *"csf"* and *"modCompScaler"* values for which this RE has a "1" in the reMask

 3Xb. If "csf" $== 1$ then $iqSampleFx = iqSampleFx + 2^{-N}$ [this is "unshifting" the constellation point].

 3Xc. $iqSampleScaled = \text{modCompScaler} \times iqSampleFx \times \sqrt{2}$ [this scales the constellation point]

3Y. For each RE in the PRB (using section extention =5):

3Ya: fetch the *"csf"* and *"mcScaleOffset"* values for which this RE has a "1" in the relevant mcScaleReMask

3Yb. If "csf" $==$ 1 then $iqSampleFx = iqSampleFx + 2^{-N}$ [this is "unshifting" the constellation point].

3Yc. $iqSampleScaled =$ mcScaleOffset $\times$ $iqSampleFx \times \sqrt{2}$ [this scales the constellation point]

After decompression, | *iqSampleScaled* | must be ≤ 1 and a value of | *iqSampleScaled* | = 1.0 matches 0 dBFS.

Annex B Delay Management Use Cases

Intra-PHY lower layer fronthaul split has characteristic of a stringent bandwidth and tight latency requirement. This implies use of a special "Fronthaul Service Profile" to be supported by the transport network, and which may differ depending on the operating environment, topology and target use cases. The general concept and latency boundaries is based on eCPRI reference points for delay management definitions (See **Figure B-1**). However, this specification additionally differentiates between DL (**Figure B-2**) and UL (**Figure B-3**) latency boundaries. The parameters and how these are determined for a pre-defined latency configuration are explained below. (An actual example of the parameters for a specific use case are presented in Annex B.1.) Pre-defined latency is necessary when actual latency measurements are not provided; both the use of pre-defined latency value and use of a method for measuring actual network latency in the DL and UL are supported in this specification.

The following assumptions are considered for the delay boundaries definitions:

- Tcp_adv_dl: Smallest time advance to receive Downlink Data C-Plane message before the first IQ data can be processed
- The fronthaul transmission delay behavior for C-Plane for DL data, C-Plane for UL data, and U-Plane for DL data is equal. Thus, there is common usage of T12_min and T12_max parameters.
- The transmission window (T1a_max – T1a_min) for C-Plane for DL data, C-Plane for UL data, and U-Plane for DL data all have the same length.
- The reception window (T2a_max – T2a_min) for C-Plane for DL data, C-Plane for UL data, and U-Plane for DL data all have the same length.
- T2a_min_cp_ul: Latest availability at O-RU of C-Plane for UL data message before reception of the first IQ data sample of the respective user's U-Plane UL data packet is received over the air interface.

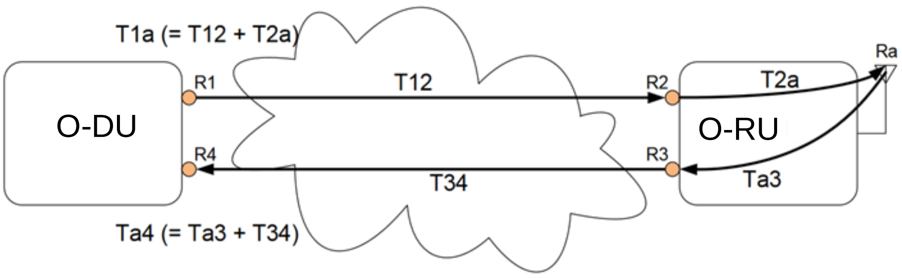


Figure B-1 : Definition of reference points for delay management (adapted from [3])

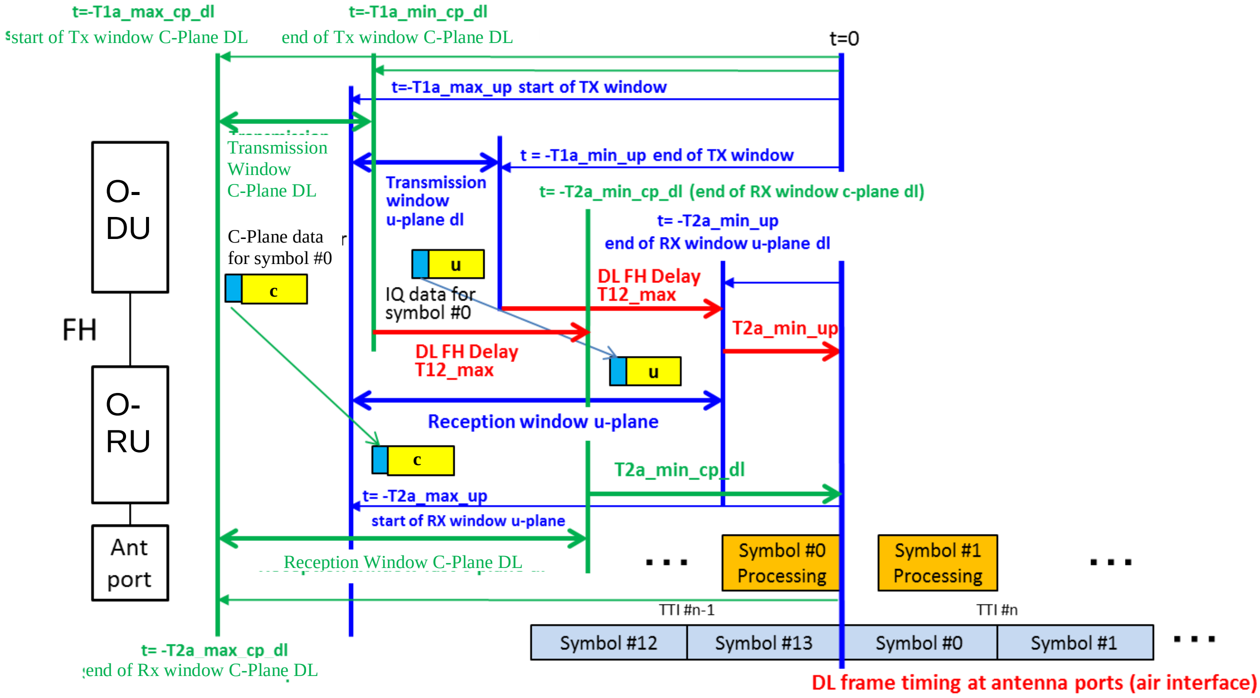


Figure B-2 : Timing relations per symbol IQ in DL direction (U-Plane and C-Plane)

Figure B-2 is based on the eCPRI delay measurement model on timing relations in DL direction. More detail is added to illustrate the following data transfer timing relations:

- **U-plane DL data : Blue path**
- **C-plane for DL data : Green path**

To understand this timing diagram, it is easier to work backward in timeline.

For U-plane DL data flow (use symbol #0 transmission as an example):

- t = 0 : time for symbol #0 air interface transmission
- t = - T2a_min_up : O-RU has a fixed data processing delay (T2a_min_up). In order to meet air interface time for symbol#0 transmission at t = 0, symbol#0 data must be presented on time for the processing unit (yellow block).
  - For symbol#0, start of processing time t = 0 – T2a_min_up = - T2a_min_up
- End of reception window : The latest time that O-RU can accept U-plane DL data for a specific symbol prior to start of data processing. If U-plane DL data arrives earlier than this time and is within reception window range, DL data may wait inside reception window buffer until the start of processing time, hence - the end of reception window is the same time point as the start of processing time as described immediately above.
  - For symbol#0, end of reception window time t = 0 – T2a_min_up = - T2a_min_up.
- Start of reception window : This earliest time that O-RU can accept U-plane DL data for a specific symbol prior to start of data processing. If U-plane DL data arrives later than this time and is within reception window range, DL data will wait inside reception window buffer.
  - For symbol#0, start of reception window time t = 0 - T2a_max_up = - T2a_max_up
- O-RU reception window range = T2a_max_up – T2a_min_up
- End of transmission window : The latest time that O-DU can send U-plane DL data for a specific symbol out to transport interface.
  - For symbol#0, end of transmission window time t = 0 – T1a_min_up = - T1a_min_up.
- Start of transmission window : The earliest time that O-DU can send U-plane DL for a specific symbol out to transport interface.

- For symbol#0, start of transmission window time t = 0 – T1a_max_up = - T1a_max_up.

- O-DU transmission window range = T1a_max_up – T1a_min_up
- DL FH transport delay : T12_min and T12_max is the min and max transport delay. When FH transport is behaving properly, it should guarantee each packet transport (carrying C-plane or U-plane DL packet) delay is within this min and max boundary. **Figure B-2** illustrates case where T12_min = 0.
- O-DU transmission window, FH transport delay and O-RU reception window must satisfy the following criteria for proper behavior:
  - Start of reception window can accept a packet sent at start of transmission window AND experienced min FH transport delay => - T2a_max_up < -T1a_max_up + T12_min ---- (eq 1)
  - End of reception window can accept a packet sent at end of transmission window AND experienced max FH transport delay => - T2a_min_up > -T1a_min_up + T12_max ---- (eq 2)
  - Combining both criteria (-eq1 + eq2), the logical conclusion is as follows:
    - ➢ **O-RU reception window range > O-DU transmission window + FH DL transport max-min**
    - ➢ **(T2a_max_up – T2a_min_up) > (T1a_max_up – T1a_min_up) + (T12_max – T12_min)**

In other words, the reception window at O-RU MUST be greater than or equal to the total of the O-DU transmission variability and the Transport time variability.

For C-plane to support DL data flow (use symbol #0 transmission as an example):

- Focus on the green path, same principle is applied to relate reception window, transmission window and FH transport delay.
- the logical conclusion is as follows:
  - ➢ **Reception window range > Transmission window + FH DL transport max-min**
  - ➢ **(T2a_max_cp_dl – T2a_min_cp_dl) > (T1a_max_cp_dl – T1a_min_cp_dl) + (T12_max – T12_min)**

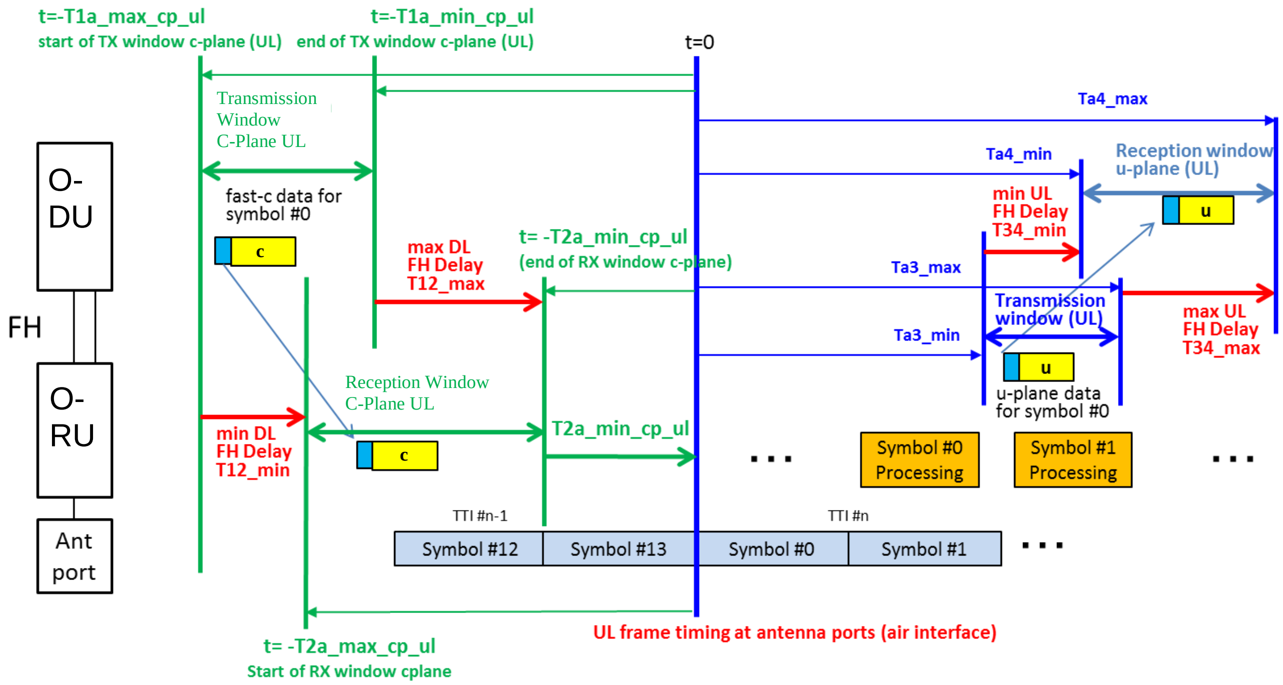


Figure B-3 : Timing relations per symbol IQ in UL direction (U-Plane and C-Plane)

Figure B-3 is based on the eCPRI delay measurement model on timing relations in UL direction. More detail is added to illustrate the following data transfer timing relations:

- **U-plane UL data : Blue path**
- **C-plane for UL data : Green path**

To understand this timing diagram, it is easier to work forward in timeline for the U-plane UL data and work backward in timeline for the C-plane to support UL data flow.

For U-plane UL data flow (use symbol #0 transmission as an example):

- t = 0 : time for symbol #0 air interface UL reception
- t = 0 + Ta3_min_up : O-RU has a fixed data processing delay (Ta3_min_up). Air interface data is immediately presented to data processing unit (yellow block).
  - For symbol#0, end of processing = 0 + Ta3_min_up = Ta3_min_up
- Start of transmission window : The earliest time that O-RU can send U-plane UL for a specific symbol out to transport interface. The earliest time is immediately after data processing.
  - For symbol#0, start of transmission window = 0 + Ta3_min_up = Ta3_min_up.
- End of transmission window : The latest time that O-RU can send U-plane UL data for a specific symbol out to transport interface.
  - For symbol#0, end of transmission window = 0 + Ta3_max_up = Ta3_max_up.
- O-RU transmission window range = Ta3_max_up – Ta3_min_up
- Start of Reception window : This earliest time that O-DU can accept U-plane UL data for a specific symbol prior to start of O-DU data processing. If U-plane DL data arrives later than this time and is within reception window range, UL data will wait inside reception window buffer.
  - For symbol#0, start of reception window = 0 + Ta4_min = Ta4_min
- End of Reception window : The latest time that O-DU can accept U-plane UL data for a specific symbol prior to start of O-DU data processing.
  - For symbol#0, end of reception window = 0 + Ta4_max = Ta4_max.
- O-DU reception window range = Ta4_max – Ta4_min
- UL FH Transport delay : T34_min and T34_max is the min and max transport delay. When FH transport is behaving properly, it should guarantee each packet transport (carrying U-plane UL packet) delay is within this min and max boundary.
- Transmission window, FH transport delay and reception window must satisfy the following criteria for proper behavior:
  - Start of reception window can accept a packet sent at start of transmission window AND experienced min FH transport delay => Ta4_min < Ta3_min_up + T34_min ---- (eq 1)
  - End of reception window can accept a packet sent at end of transmission window AND experienced max FH transport delay => Ta4_max > Ta3_max_up + T34_max---- (eq 2)
  - Combining both criteria (-eq1 + eq2), the logical conclusion is as follows:
    - **O-DU reception window range > O-RU transmission window + FH UL transport max-min**
    - **(Ta4_max – Ta4_min) > (Ta3_max_up – Ta3_min_up) + (T34_max – T34_min)**

In other words, the reception window at O-DU MUST be greater than or equal to the total of the O-RU transmission variability and the Transport time variability.

For C-plane to support UL data flow (use symbol #0 transmission as an example):

- Focus on the green path, same principle to C-plane to support DL data is applied to relate reception window, transmission window and FH transport delay.
- the logical conclusion is as follows:

- **Reception window range > Transmission window + FH transport min-max difference**
- **(T2a_max_cp_ul – T2a_min_cp_ul) > (T1a_max_cp_ul – T1a_min_cp_ul) + (T12_max – T12_min)**

It is useful to consider several different delay management use cases because delay management, in particular the need to buffer large amounts of data within O-RUs, can have a significant cost and complexity impact on O-RU design. More specifically, accommodating long network delays (allowing long fiber lengths and/or many switch hops) can impose a significant buffering requirement on the O-RU while use cases involving short network delays e.g. an in-building application may allow much smaller in-O-RU buffering thereby allowing a lower-cost and lower-power design. Additionally, the SCS may affect the transmission windows described above, and thus use cases may be SCS specific as well. Note that here it is assumed the DL buffering will be done in the O-RU, otherwise the O-DU would have to always implement extensive buffering to accommodate any O-RU use case which would increase overall costs.

In the case of using pre-defined network latency values per use case, the following must be pre-defined (for measured-network-latency cases these same values are determined via the measurement process):

T12_min, T12_max, T34_min and T34_max must be determined based on the desired network configuration. Determination of T12_min and T34_min values must include shortest transmission paths, both fiber and minimum switching delays. In addition, longest fibers, switching delays, and PDV introduced due to the variable delay nature of ethernet must be accounted for in T12_max and T34_max.

T2a_min_up, T2a_min_cp_ul, and Ta3_min must be determined across all O-RU equipment to be used in the system. The determined values must be greater than or equal to the largest of these values across all supported equipment. Equipment with lower values need to have additional delay added to align with the determined values.

Similarly, the maximum transmission windows must be determined across all O-DU and O-RU equipment. As stated above, the transmission windows for DL UP, DL CP and UL CP are all assumed to be the same at current. However, this is the maximum transmission window. Equipment may use less time for transmission of any symbol.

Additionally, the advance between DL UP/ CP must be determined. (Tcp_adv_dl)

Once the above parameters are determined, the remaining values may be calculated as shown in the following tables:

<u>**Downlink Data Direction**</u>

Table B-1 : U-Plane DL delay boundaries

Downlink	Method
T1a_max_up	≤ T12_min + T2a_max_up
T1a_min_up	≥ T12_max + T2a_min_up
T2a_max_up	≥ T2a_min_up + (T12_max – T12_min) + O-DU Transmission Window
T2a_min_up	Specified per Use Case
T12_max	Specified per Use Case
T12_min	Specified per Use Case
O-DU Transmission Window	Specified per Use Case

Table B-2 : C-Plane DL delay boundaries

Downlink	Method
Tcp_adv_dl	Specified per Use Case
T1a_max_cp_dl	T1a_max_up + Tcp_adv_dl
T1a_min_cp_dl	T1a_min_up + Tcp_adv_dl
T2a_max_cp_dl	T2a_max_up + Tcp_adv_dl
T2a_min_cp_dl	T2a_min_up + Tcp_adv_dl
T12_max	Same as U-plane DL
T12_min	Same as U-plane DL

<u>**Uplink Data Direction** (need not be the same as the Downlink values)</u>

Table B-3 : U-Plane Uplink delay boundaries

Uplink	Method
Ta3 max	≤ Ta3_min + O-RU Transmission Window
Ta3 min	Specified per Use Case
Ta4 max	≤ Ta3_max + T34_max
Ta4 min	≥ Ta3_min + T34_min

T34 max	Specified per Use Case
T34 min	Specified per Use Case
O-RU Transmission Window	Specified per Use Case

Table B-4 : C-Plane Uplink delay boundaries

Downlink	Method
T1a_max_cp_ul	≤ T12_min + T2a_max_cp_ul
T1a_min_cp_ul	≥ T12_max + T2a_min_cp_ul
T2a_max_cp_ul	≥ T2a_min_cp_ul + (T12_max – T12_min) + O-DU Transmission Window
T2a_min_cp_ul	Specified per Use Case
T12_max	Specified per Use Case
T12_min	Specified per Use Case
O-DU Transmission Window	Specified per Use Case

B.1 Latency categories and sub-categories

The O-DU and O-RU are defined to be in “delay categories” and “delay sub-categories” for the purposes of allowing a matching of O-DU and O-RU units that will operate properly together from the point of view of accommodating a customer’s network delay. Network delay comprises the “time-of-flight” of signals through (typically) a fiber-optic cable (so can be known from the fiber length) added to the signal traversal latency through any switches in the network.

The delay category and delay sub-category values depend in part on the processing latency within the O-DU and O-RU. It may be expected especially for an O-RU that the processing latency may depend on the specific frequency band and sub-carrier spacing that is used. Further, a multi-band radio may experience different processing latencies for its different bands. Therefore, it may be expected that an O-RU (and perhaps more rarely an O-DU) will have different delay category and delay sub-category ratings for different bands.

O-DU categories are defined as A.## (per direction, i.e. O-DU UL and O-DU DL are different categories)

- Prefix(blank/f): The prefix distinguishes the type of O-DU with dynamic/fixed timing advance/retard
- A: Category from [A-N] per table below
- .##: sub-category from [.00 - .40] per tables below

O-RU categories are defined as X.## (per direction, i.e. O-RU UL and O-RU DL are different categories)

- X: Category from [O-Z] per table below
- .##: sub-category from [.00-.40] per tables below

The following tables are used by equipment vendors to assign categories to their equipment based on design characteristics.

Table B-5 : O-DU and O-RU Delay Categories

O-DU Category	
Category	• T1a_max_up$_{IIs\text{-}CU}$ - Txmax$_{IIs\text{-}CU}$ OR • Ta4_max_up$_{IIs\text{-}CU}$ [µsec]
A	≥ 400
B	380 to 399
C	360 to 379
D	340 to 359
E	320 to 339
F	300 to 319
G	280 to 299
H	260 to 279
I	240 to 259
J	220 to 239
K	200 to 219
L	180 to 199
M	160 to 179
N	0 to 159

O-RU Category	
Category	• T2a_min_up OR • Ta3_max [µsec]
O	0 to 50
P	51 to 70
Q	71 to 90
R	91 to 110
S	111 to 130
T	131 to 150
U	151 to 170
V	171 to 190
W	191 to 210
X	211 to 230
Y	231 to 250
Z	251 to 270

Note: Categories are defined to group endpoints with similar delay characteristics for easy evaluation relative to use cases. The calculated value for O-DU or O-RU falling anywhere within the range for the category indicates that the endpoint is classified as that category. It is NOT required that the endpoint be able to support the full range of the category.

Table B-6 and B-7 are intended for use by network providers to determine the best and worst case T12_max/ T34_max values that can be supported by a given equipment combination. Alternatively, network providers may locate the desired T12_max/ T34_max and select from the equipment combinations meeting that criteria. Common criteria are identified by different colors on diagonals through the tables

Table B-6 : Latency_min (Minimum supported T12_max/ T34_max in µsec)

	O-DU													
RU	A	B	C	D	E	F	G	H	I	J	K	L	M	N
O	350	330	310	290	270	250	230	210	190	170	150	130	110	0
P	330	310	290	270	250	230	210	190	170	150	130	110	90	0
Q	310	290	270	250	230	210	190	170	150	130	110	90	70	0
R	290	270	250	230	210	190	170	150	130	110	90	70	50	0
S	270	250	230	210	190	170	150	130	110	90	70	50	30	0
T	250	230	210	190	170	150	130	110	90	70	50	30	10	0
U	230	210	190	170	150	130	110	90	70	50	30	10	0	0
V	210	190	170	150	130	110	90	70	50	30	10	0	0	0
W	190	170	150	130	110	90	70	50	30	10	0	0	0	0
X	170	150	130	110	90	70	50	30	10	0	0	0	0	0
Y	150	130	110	90	70	50	30	10	0	0	0	0	0	0
Z	130	110	90	70	50	30	10	0	0	0	0	0	0	0

Table B-7 : Latency_max (Maximum supported T12_max/ T34_max in µsec)

	O-DU													
RU	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**
O	3000	399	379	359	339	319	299	279	259	239	219	199	179	159
P	2949	348	328	308	288	268	248	228	208	188	168	148	128	108
Q	2929	328	308	288	268	248	228	208	188	168	148	128	108	88
R	2909	308	288	268	248	228	208	188	168	148	128	108	88	68
S	2889	288	268	248	228	208	188	168	148	128	108	88	68	48
T	2869	268	248	228	208	188	168	148	128	108	88	68	48	28
U	2849	248	228	208	188	168	148	128	108	88	68	48	28	8
V	2829	228	208	188	168	148	128	108	88	68	48	28	8	0
W	2809	208	188	168	148	128	108	88	68	48	28	8	0	0
X	2789	188	168	148	128	108	88	68	48	28	8	0	0	0
Y	2769	168	148	128	108	88	68	48	28	8	0	0	0	0
Z	2749	148	128	108	88	68	48	28	8	0	0	0	0	0

Table B-8 : Use Case Mapping (Based on Latency_min)

Range (KM)	**T12max T34max** (µsec)	**Minimum T12Max Guaranteed for Category Combinations** (Combinations may support larger T12max)												
50	250	AO - AT	BO - BS	CO - CR	DO - DQ	EO - EP	FO							
45	225	AU	BT	CS	DR	EQ	FP	GO						
40	200	AV	BU	CT	DS	ER	FQ	GP	HO					
35	175	AW	BV	CU	DT	ES	FR	GQ	HP	IO				
30	150	AX - AY	BW - BX	CV - CW	DU - DV	ET - EU	FS - FT	GR - GS	HQ - HR	IP - IQ	JO - JP	KO		
25	125	AZ	BY	CX	DW	EV	FU	GT	HS	IR	JQ	KP	LO	
20	100		BZ	CY	DX	EW	FV	GU	HT	IS	JR	KQ	LP	MO
15	75			CZ	DY	EX	FW	GV	HU	IT	JS	KR	LQ	MP
10	50				DZ	EY - EZ	FX - FY	GW - GX	HV - HW	IU - IV	JT - JU	KS - KT	LR - LS	MQ - MR
5	25						FZ	GY	HX	IW	JV	KU	LT	MS
0	0							GZ	HY	IX	JW	KV	LU	MT

Note: The Use Case Mapping table can be used to determine the O-DU/ O-RU delay category combinations which are guaranteed to support a given latency use case. This is defined based on Latency_min for the category. It may be possible for equipment from a lower category to support a higher use case, but cannot be guaranteed. To determine the exact maximum latency for an O-DU/ O-RU pair, it is necessary to calculate based on the delay parameter values for the paired equipment.

Table B-9 : O-DU and O-RU Tx/Rx Window Sub-Categories

Receive Window Sub-Category	
Sub-Category	• T2a_max_up - T2a_min_up • $RXmax_{lls\text{-}CU}$ [in usec]
.40	≥ 400
.38	380 to 399
.36	360 to 379
.34	340 to 359
.32	320 to 339
.30	300 to 319
.28	280 to 299
.26	260 to 279
.24	240 to 259
.22	220 to 239
.20	200 to 219
.18	180 to 199
.16	160 to 179
.14	140 to 159
.12	120 to 139
.10	100 to 119
.08	80 to 99
.06	60 to 79
.04	40 to 59
.02	20 to 39
.01	10 to 19
.00	0 to 9

Transmit Window Sub-Category	
Sub-Category	• $TXmax_{lls\text{-}CU}$ • Ta3_max - Ta3_min [in usec]
.20	≥ 200
.19	190 to 199
.18	180 to 189
.17	170 to 179
.16	160 to 169
.15	150 to 159
.14	140 to 149
.13	130 to 139
.12	120 to 129
.11	110 to 119
.10	100 to 109
.09	90 to 99
.08	80 to 89
.07	70 to 79
.06	60 to 69
.05	50 to 59
.04	40 to 49
.03	30 to 39
.02	20 to 29
.01	10 to 19
.00	0 to 9

Note: Sub-categories are defined to group endpoints with similar delay characteristics for easy evaluation relative to use cases. The calculated value for O-DU or O-RU falling anywhere within the range for the category indicates that the endpoint is classified as that sub-category. It is NOT required that the endpoint be able to support the full range of the sub-category.

Table B-10 is used by service providers to identify equipment sub-category combinations which meet the desired network variability. Variability is shown in KM in the table. This range is based on 5 usec per Km.

Table B-10 : Dynamic Range (in Km) for sub-category pairs

	Receive Window																				
TxMax	.40	.38	.36	.34	.32	.30	.28	.26	.24	.22	.20	.18	.16	.14	.12	.10	.08	.06	.04	.02	.01
.20	40	36	32	28	24	20	16	12	8	4											
.19	42	38	34	30	26	22	18	14	10	6	2										
.18	44	40	36	32	28	24	20	16	12	8	4										
.17	46	42	38	34	30	26	22	18	14	10	6	2									
.16	48	44	40	36	32	28	24	20	16	12	8	4									
.15	50	46	42	38	34	30	26	22	18	14	10	6	2								
.14	52	48	44	40	36	32	28	24	20	16	12	8	4								
.13	54	50	46	42	38	34	30	26	22	18	14	10	6	2							
.12	56	52	48	44	40	36	32	28	24	20	16	12	8	4							
.11	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2						
.10	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4						
.09	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2					
.08	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4					
.07	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2				
.06	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4				
.05	70	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2			
.04	72	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4			
.03	74	70	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2		
.02	76	72	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4		
.01	78	74	70	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2	
.00	80	76	72	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4	2

B.2 Example Case: Evaluating O-DU / O-RU Combinations

This example illustrates an O-RU and an O-DU, each supporting multiple SCS. It illustrates how the actual O-DU/ O-RU delay profiles map to categories, and then how the corresponding categories are used to evaluate the resulting transport network limitations. The values represented are exemplary only. The example is provided to illustrate:

- Interoperability across multiple SCS
  - O-RU and O-DU will have multiple delay profiles
  - Delay profiles are different for uplink and downlink
- applicability service provider use case

Table B-11 : Delay profiles for the example O-RU

	O-RU Parameters	30 KHz	120 KHz		15 KHz
Downlink	T2a_min_up,	**50**	**50**		**205**
	T2a_max_up,	250	180		435
	T2a_min_cp_dl,	175	175		330
	T2a_max_cp_dl,	375	205		460
	Tcp_adv_dl	125	125		125
	Category	**O.20** (250-50)=**200**	**O.12**		**W.22**
Uplink	Ta3_min,	50	50		70
	Ta3_max	**100**	**70**		**235**
	T2a_min_cp_ul,	125	125		125
	T2a_max_cp_ul	325	255		360
	Category	**R.05**	**P.02**		**Y.16**

Table B-12 : Delay profiles for the example O-DU.

	O-DU Parameters	30 KHz	120 KHz	15 KHz
Downlink	T1a_max_up$_{O\text{-}DU}$,	**250**	**180**	**435**
	TXmax$_{O\text{-}DU}$	**40**	**20**	**70**
	Category	**K.04**	**M.02**	**C.07**
Uplink	Ta4_max$_{O\text{-}DU}$,	250	180	395
	RXmax$_{O\text{-}DU}$	**200**	130	325
	Category	**I.20**	**L.12**	**B.32**

This results in 6 different category combinations:

Table B-13 : Resulting 6 different category combinations

	30 KHz	120 KHz	15 KHz
Downlink	KO	MO	CW
Uplink	IR	LP	BY

The respective minimum and maximum T12max values are highlighted in the tables below. Using the 30 KHz as an example, it can be seen that the range on the transport is limited by the uplink (IR) to between 130 usec and 168 usec. This means that this combination can be guaranteed to be able to support at least 130 usec of delay, and may be capable of supporting up to 168 usec of delay. Using the actual delay values for the combination at 30 KHz, the uplink is limited to Ta4_max – Ta3_max = 250 – 100 = 150 usec. (The downlink value is higher, so uplink becomes the limiting factor for this combination.)

Table B-14 : Delay Category O-DU and O-RU with highlighted valid options for this example, minimum T12max

	lls-DU													
RU	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**
O	350	330	310	290	270	250	230	210	190	170	**150**	130	**110**	0
P	330	310	290	270	250	230	210	190	170	150	130	**110**	90	0
Q	310	290	270	250	230	210	190	170	150	130	110	90	70	0
R	290	270	250	230	210	190	170	150	**130**	110	90	70	50	0
S	270	250	230	210	190	170	150	130	110	90	70	50	30	0
T	250	230	210	190	170	150	130	110	90	70	50	30	10	0
U	230	210	190	170	150	130	110	90	70	50	30	10	0	0
V	210	190	170	150	130	110	90	70	50	30	10	0	0	0
W	190	170	**150**	130	110	90	70	50	30	10	0	0	0	0
X	170	150	130	110	90	70	50	30	10	0	0	0	0	0
Y	150	**130**	110	90	70	50	30	10	0	0	0	0	0	0
Z	130	110	90	70	50	30	10	0	0	0	0	0	0	0

Table B-15 : Delay Category O-DU and O-RU with highlighted valid options for this example, maximum T12max

	lls-DU													
RU	A	B	C	D	E	F	G	H	I	J	K	L	M	N
O	3000	399	379	359	339	319	299	279	259	239	**219**	199	**179**	159
P	2949	348	328	308	288	268	248	228	208	188	168	**148**	128	108
Q	2929	328	308	288	268	248	228	208	188	168	148	128	108	88
R	2909	308	288	268	248	228	208	188	**168**	148	128	108	88	68
S	2889	288	268	248	228	208	188	168	148	128	108	88	68	48
T	2869	268	248	228	208	188	168	148	128	108	88	68	48	28
U	2849	248	228	208	188	168	148	128	108	88	68	48	28	8
V	2829	228	208	188	168	148	128	108	88	68	48	28	8	0
W	2809	208	**188**	168	148	128	108	88	68	48	28	8	0	0
X	2789	188	168	148	128	108	88	68	48	28	8	0	0	0
Y	2769	**168**	148	128	108	88	68	48	28	8	0	0	0	0
Z	2749	148	128	108	88	68	48	28	8	0	0	0	0	0

Evaluating the overall combination across all SCS, it can be seen that the maximum possible range for this combination if all 3 SCS are to be used is 148 usec (limited by the 120 KHz uplink) and the minimum guaranteed range is 110 usec. (In actuality, the limit is 110 usec for this combination.)

Similarly, there are 6 combinations of sub-categories to evaluate. (In this case, the sub-category combinations happen to be the same for 120 KHz uplink and downlink.) In this case the different between T12max and T12min is limited by the 120 KHz delay profile, with a maximum range of ~20 KM (~100 usec).

Table B-16 : Delay Sub-Category O-DU and O-RU with highlighted (in red) valid options for this example

	Receive Window																				
TxMax	.40	.38	.36	.34	.32	.30	.28	.26	.24	.22	.20	.18	.16	.14	.12	.10	.08	.06	.04	.02	.01
.20	40	36	32	28	24	20	16	12	8	4											
.19	42	38	34	30	26	22	18	14	10	6	2										
.18	44	40	36	32	28	24	20	16	12	8	4										
.17	46	42	38	34	30	26	22	18	14	10	6	2									
.16	48	44	40	36	**32**	28	24	20	16	12	8	4									
.15	50	46	42	38	34	30	26	22	18	14	10	6	2								
.14	52	48	44	40	36	32	28	24	20	16	12	8	4								
.13	54	50	46	42	38	34	30	26	22	18	14	10	6	2							
.12	56	52	48	44	40	36	32	28	24	20	16	12	8	4							
.11	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2						
.10	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4						
.09	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2					
.08	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4					
.07	66	62	58	54	50	46	42	38	34	**30**	26	22	18	14	10	6	2				
.06	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4				
.05	70	66	62	58	54	50	46	42	38	34	**30**	26	22	18	14	10	6	2			
.04	72	68	64	60	56	52	48	44	40	36	**32**	28	24	20	16	12	8	4			
.03	74	70	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2		
.02	76	72	68	64	60	56	52	48	44	40	36	32	28	24	**20**	16	12	8	4		
.01	78	74	70	66	62	58	54	50	46	42	38	34	30	26	22	18	14	10	6	2	
.00	80	76	72	68	64	60	56	52	48	44	40	36	32	28	24	20	16	12	8	4	2

Using only the O-DU/ O-RU category/ sub-category combinations, if the corresponding O-DU/ O-RU is to be used across all 3 SCS, the resulting delay constraints limit the implementation to:

- 110 usec ≤ T12max ≤ 148 usec
- T12min = T12max – 100 usec

The service provider can then use this analysis to determine if the resulting combination suits their target use case.

Annex C M-Plane Impacts

The CUS-Plane makes certain demands on the M-Plane as listed below:

1. **Generic static O-RU configuration:** there are many parameters e.g. frequency band, number of antennas, power level, etc. that will need to be gathered from the O-RU and configured by the O-DU. These are expected to be much the same as is currently experienced with existing radio modules.
2. **O-RU management:** this includes status monitoring, KPI measurements (PM counters), alarm collection and software download. This is expected to be handled much the same as with existing radio modules.
3. **Compression:** command to use static method (compression and IQ bit width) or use dynamic method using udCompHdr.
4. **Rtcid**: M-Plane provides the bit-widths for the four defined fields (must sum to 16 bits total).
5. **Synch state**: it is expected the O-RU will report its sync state to the EMS via the M-Plane.
6. **Synch**: it is expected the EMS via the M-Plane will convey to the O-RU the clock quality being received (or does this come directly from the GM?)
7. **BeamId format**: to accommodate hybrid BF, and maybe even for other cases, the O-RU must convey via M-Plane characteristics of the O-RU so that the O-DU can "know" how to generate BF weights.
8. **PRB raster and offset-setting**: The M-Plane needs to convey the minimum PRB raster (based on the minimum SCS, for LTE generally 15 KHz) to allow PRB counting across multiple SCS values (to support mixed-numerology channels). In addition, the M-Plane must convey the offset to the zeroth PRB.
9. **Beam-weights / beam attributes**: It is intended that the M-Plane can download beam weights or beam attributes to an O-RU when weight updating does not need to be real-time. The number of weights or types of attributes applicable to the O-RU is meant to be conveyed from the O-RU to the O-DU at start-up via M-plane messaging.
10. **Power-Efficiency**: the M-Plane will very likely include commands to enable O-RU power-saving techniques which may be vendor-dependent.
11. **Delay-Management and Transport Priority**: for each eAxC (Pcid), whether the UL data is delay-managed or not, and what the transport priority should be.

Annex D IQ Sample and Exponent Packetization for Different Bitwidths

Bit-ordering and packetization for I and Q samples and compression parameters follows patterns that repeat after every transmission of 12 resource elements. At this point, the pattern repeats starting from the udCompParam information element, and is followed by the I and Q samples for the next 12 resource elements.

The cells in the following Tables indicate the bit ordering for the following IEs

- udCompParam (assumed to be one byte in the tables)
- I samples denoted by $I_{btwidth-1}\ldots I_0$
- Q samples denoted by $Q_{bitwidth-1}\ldots Q_0$

Table D-1. IQ data samples bit-ordering (6-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_5	I_4	I_3	I_2	I_1	I_0	Q_5	Q_4	1	N+1
Q_3	Q_2	Q_1	Q_0	…				1	N+2
…									
			…	I_5	I_4	I_3	I_2	1	N+17
I_1	I_0	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+18
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_5	I_4	I_3	I_2	I_1	I_0	Q_5	Q_4	1	M+1
Q_3	Q_2	Q_1	Q_0	…				1	M+2
…									
			…	I_5	I_4	I_3	I_2	1	M+17
I_1	I_0	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+18
…									

Table D-2. IQ data samples bit-ordering (7-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_6	1	N+1
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	N+2
…									
	…	I_6	I_5	I_4	I_3	I_2	I_1	1	N+20
I_0	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+21
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_6	1	M+1
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	M+2
…									
	…	I_6	I_5	I_4	I_3	I_2	I_1	1	M+20
I_0	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+21
…									

Table D-3. IQ data samples bit-ordering (8-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	N+1
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+2
…									
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	N+23
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+24
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	M+1
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+2
…									
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	M+23
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+24
…									

Table D-4. IQ data samples bit-ordering (9-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_8	I_7	I_6	I_5	I_4	I_3	I_2	I_1	1	N+1
I_0	Q_8	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	1	N+2
Q_1	Q_0	…						1	N+3
…									
					…	I_8	I_7	1	N+25
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_8	1	N+26
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+27
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_8	I_7	I_6	I_5	I_4	I_3	I_2	I_1	1	M+1
I_0	Q_8	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	1	M+2
Q_1	Q_0	…						1	
…									
					…	I_8	I_7	1	M+25
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_8	1	M+26
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+27
…									

Table D-5. IQ data samples bit-ordering (10-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_9	I_8	I_7	I_6	I_5	I_4	I_3	I_2	1	N+1
I_1	I_0	Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	1	N+2
Q_3	Q_2	Q_1	Q_0	…				1	N+3
…									
			…	I_9	I_8	I_7	I_6	1	N+28
I_5	I_4	I_3	I_2	I_1	I_0	Q_9	Q_8	1	N+29
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+30
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_9	I_8	I_7	I_6	I_5	I_4	I_3	I_2	1	M+1
I_1	I_0	Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	1	M+2
Q_3	Q_2	Q_1	Q_0	…				1	M+3
…									
			…	I_9	I_8	I_7	I_6	1	M+28
I_5	I_4	I_3	I_2	I_1	I_0	Q_9	Q_8	1	M+29
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+30
…									

Table D-6. IQ data samples bit-ordering (11-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	I_3	1	N+1
I_2	I_1	I_0	Q_{10}	Q_9	Q_8	Q_7	Q_6	1	N+2
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	N+3
…									
	…	I_{10}	I_9	I_8	I_7	I_6	I_5	1	N+31
I_4	I_3	I_2	I_1	I_0	Q_{10}	Q_9	Q_8	1	N+32
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+33
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M

I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	I_3	1	M+1
I_2	I_1	I_0	Q_{10}	Q_9	Q_8	Q_7	Q_6	1	M+2
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	M+3
…									
	…	I_{10}	I_9	I_8	I_7	I_6	I_5	1	M+31
I_4	I_3	I_2	I_1	I_0	Q_{10}	Q_9	Q_8	1	M+32
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+33
…									

Table D-7. IQ data samples bit-ordering (12-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	1	N+1
I_3	I_2	I_1	I_0	Q_{11}	Q_{10}	Q_9	Q_8	1	N+2
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+3
…									
I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	1	N+34
I_3	I_2	I_1	I_0	Q_{11}	Q_{10}	Q_9	Q_8	1	N+35
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+36
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	1	M+1
I_3	I_2	I_1	I_0	Q_{11}	Q_{10}	Q_9	Q_8	1	M+2
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+3
…									
I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	1	M+34
I_3	I_2	I_1	I_0	Q_{11}	Q_{10}	Q_9	Q_8	1	M+35
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+36
…									

Table D-8. IQ data samples bit-ordering (13-bit bitwidth mantissa example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	1	N+1
I_4	I_3	I_2	I_1	I_0	Q_{12}	Q_{11}	Q_{10}	1	N+2
Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	1	N+3
Q_1	Q_0	…						1	N+4
…									
					…	I_{12}	I_{11}	1	N+36
I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	I_3	1	N+37
I_2	I_1	I_0	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	N+38
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+39
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	1	M+1
I_4	I_3	I_2	I_1	I_0	Q_{12}	Q_{11}	Q_{10}	1	M+2
Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	1	M+3
Q_1	Q_0	…						1	M+4
…									
					…	I_{12}	I_{11}	1	M+36
I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	I_3	1	M+37
I_2	I_1	I_0	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	M+38
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+39
…									

Table D-9. IQ data samples bit-ordering (14-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	I_6	1	N+1
I_5	I_4	I_3	I_2	I_1	I_0	Q_{13}	Q_{12}	1	N+2
Q_{11}	Q_{10}	Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	1	N+3
Q_3	Q_2	Q_1	Q_0	…				1	N+4
…									
…				I_{13}	I_{12}	I_{11}	I_{10}	1	N+39
I_9	I_8	I_7	I_6	I_5	I_4	I_3	I_2	1	N+40
I_1	I_0	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	N+41
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+42
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	I_6	1	M+1
I_5	I_4	I_3	I_2	I_1	I_0	Q_{13}	Q_{12}	1	M+2
Q_{11}	Q_{10}	Q_9	Q_8	Q_7	Q_6	Q_5	Q_4	1	M+3
Q_3	Q_2	Q_1	Q_0	…				1	M+4
…									
…				I_{13}	I_{12}	I_{11}	I_{10}	1	M+39
I_9	I_8	I_7	I_6	I_5	I_4	I_3	I_2	1	M+40
I_1	I_0	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	M+41
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+42
…									

Table D-10. IQ data samples bit-ordering (15-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	1	N+1
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_{14}	1	N+2
Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	Q_7	Q_6	1	N+3
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	N+4
…									
…		I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	1	N+42
I_8	I_7	I_6	I_5	I_4	I_3	I_2	I_1	1	N+43
I_0	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	N+44
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+45
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	1	M+1
I_6	I_5	I_4	I_3	I_2	I_1	I_0	Q_{14}	1	M+2
Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	Q_7	Q_6	1	M+3
Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	…		1	M+4
…									
…		I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	1	M+42
I_8	I_7	I_6	I_5	I_4	I_3	I_2	I_1	1	M+43
I_0	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	M+44
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+45
…									

Table D-11. IQ data samples bit-ordering (16-bit IQ bitwidth example)

0 (msb)	1	2	3	4	5	6	7 (lsb)	Number of Octets	
…									
udCompParam								1	Octet N
I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	1	N+1

I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	N+2
Q_{15}	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	N+3
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+4
…									
I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	1	N+45
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	N+46
Q_{15}	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	N+47
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	N+48
0	0	0	0	E_3	E_2	E_1	E_0	1	Octet M
I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	1	M+1
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	M+2
Q_{15}	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	M+3
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+4
…									
I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	1	M+45
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	1	M+46
Q_{15}	Q_{14}	Q_{13}	Q_{12}	Q_{11}	Q_{10}	Q_9	Q_8	1	M+47
Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	1	M+48
…									

Annex E OFDM Phase Compensation

Consider the time interval $t^{\mu}_{start,l} \le t < t^{\mu}_{start,l} + \left(N^{\mu}_{u} + N^{\mu}_{CP,l}\right)T_c$ as defined in clause 5.3.1 of [6], which corresponds to the transmission of the lth OFDM symbol, with baseband waveform $s_l^{(p,\mu)}(t)$.

Suppose that the transmitter performs up-conversion of the signal $s_l^{(p,\mu)}(t)$ to RF centre frequency f_0, and the receiver performs down-conversion from RF centre frequency f_1. Unlike LTE, it is possible that the transmitter and receiver have different centre frequencies, $f_0 \neq f_1$, which motivates the introduction of OFDM phase compensation in NR.

Ignoring channel distortions and noise for simplicity, the baseband signal at the receiver in the same time interval can be written

$$s_l^{(p,\mu)}(t) \cdot e^{j2\pi(f_0-f_1)t} = s_l^{(p,\mu)}(t) \cdot e^{j\theta_l} \cdot e^{j2\pi(f_0-f_1)\left(t-t^{\mu}_{CP,l}\right)}$$

where $\theta_l = 2\pi(f_0 - f_1)t^{\mu}_{CP,l}$ and $t^{\mu}_{CP,l} = t^{\mu}_{start,l} + N^{\mu}_{CP,l}T_c$ is the time at the end of the cyclic prefix of OFDM symbol l.

We can see from the equation above that each OFDM symbol is subjected to a phase shift θ_l, which can result in large phase changes between successive OFDM symbols. This may cause problems for receiver processes such as channel estimation and frequency error tracking, which expect the channel to vary smoothly between symbols.

The solution adopted by 3GPP in RAN WG1 meeting AH#18-01 [7][8] is to apply a phase pre-compensation term $e^{-j2\pi f_0 t^{\mu}_{CP,l}}$ at the transmitter and a phase post-compensation term $e^{j2\pi f_1 t^{\mu}_{CP,l}}$ at the receiver. These two terms together provide the required correction $e^{-j\theta_l}$. The phase pre-compensation requirement for the transmitter is captured in clause 5.4 of [1]. Note that the phase compensation depends only on the starting time of each OFDM symbol, and is common for all OFDM symbols transmitted using a given numerology μ, regardless of which NR physical channels they belong to, except for PRACH.

Annex F Beamforming Attributes Frame of Reference

The beamforming attributes involving pointing angle (bfAzPt and bfZePt) are defined following 3GPP 38.901, section 7.1. Two coordinate systems are defined, the Global Coordinate System (GCS) and local coordinate system (LCS). The GCS applies across multiple BS and UT locations, while the LCS applies to a single array antenna. The GCS is defined as shown below in **Figure F-1**. In the GCS, $\boldsymbol{\phi}$ refers to the azimuth angle and $\boldsymbol{\theta}$ refers to the zenith angle.

The LCS is defined by a 3-parameter rotation of the GCS. The rotation with parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ is shown in **Figure F-2** (left). Note that the rotation parameters are defined as follows:

- α is defined as the bearing angle (sector pointing angle)
- β is defined as the downtilt angle
- γ is defined as the slant angle

A two-dimensional array antenna is defined such that the x'-axis is broadside to the array antenna. If an antenna architecture such as in 3GPP 38.901 section 7.3 is used, the horizontal direction is defined as y' and the vertical direction is defined as z'.

All parameters which relate to "peak" refer to the principal beam pointing in a particular direction. For a given beam configuration, the peak direction is the angle corresponding to maximum gain. Thus, bfAzPt specifies the pointing angle of maximum gain.

The pointing parameter bfAzPt is defined as ϕ'. The pointing parameter bfZePt is defined as θ'. The angles are shown in **Figure F-2** (right).

The beamwidth parameters bfAz3dd and bfZe3dd are defined as the as the angular widths at which the beam falls to 3 dB below the peak beam gain on both sides of the peak direction, in azimuth and zenith, respectively.

The sidelobe parameters bfAzSl and bfZeSl are defined as the suppression level of the highest sidelobe relative to the peak gain in the azimuth and zenith principal planes, respectively.

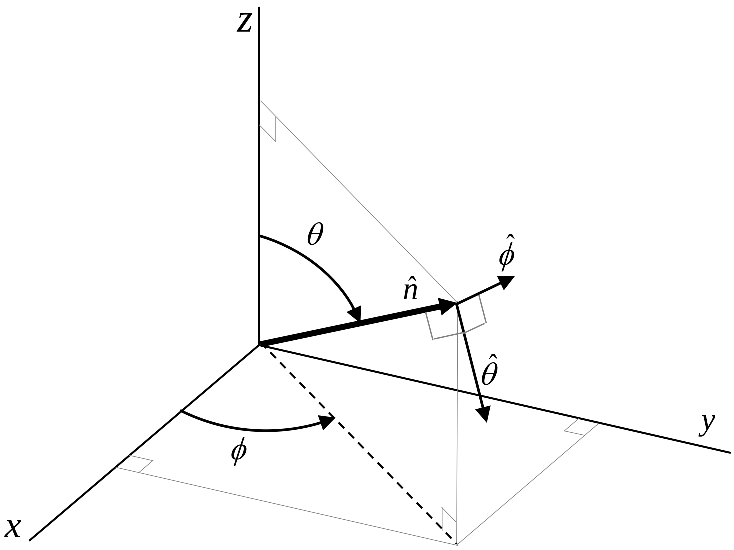


Figure F-1 : Global Coordinate System Definition

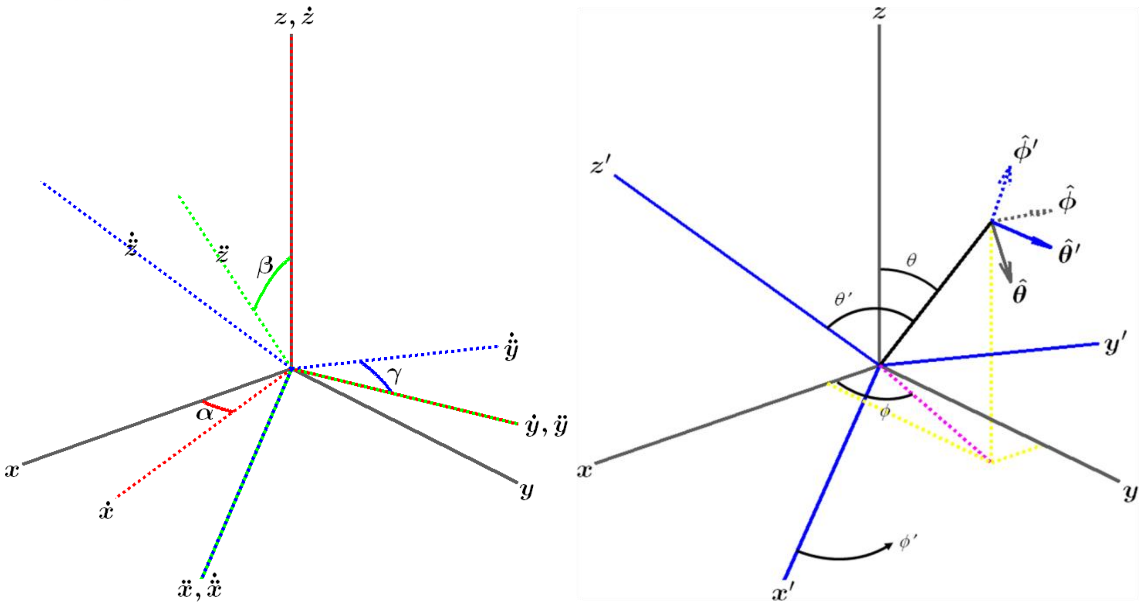


Figure F-2 : Orienting the LCS (blue) with respect to the GCS (gray) by a sequence of 3 rotations (left); Definition of spherical coordinates and unit vectors in both the GCS and LCS (right)

Annex G LAA Algorithms and Examples:

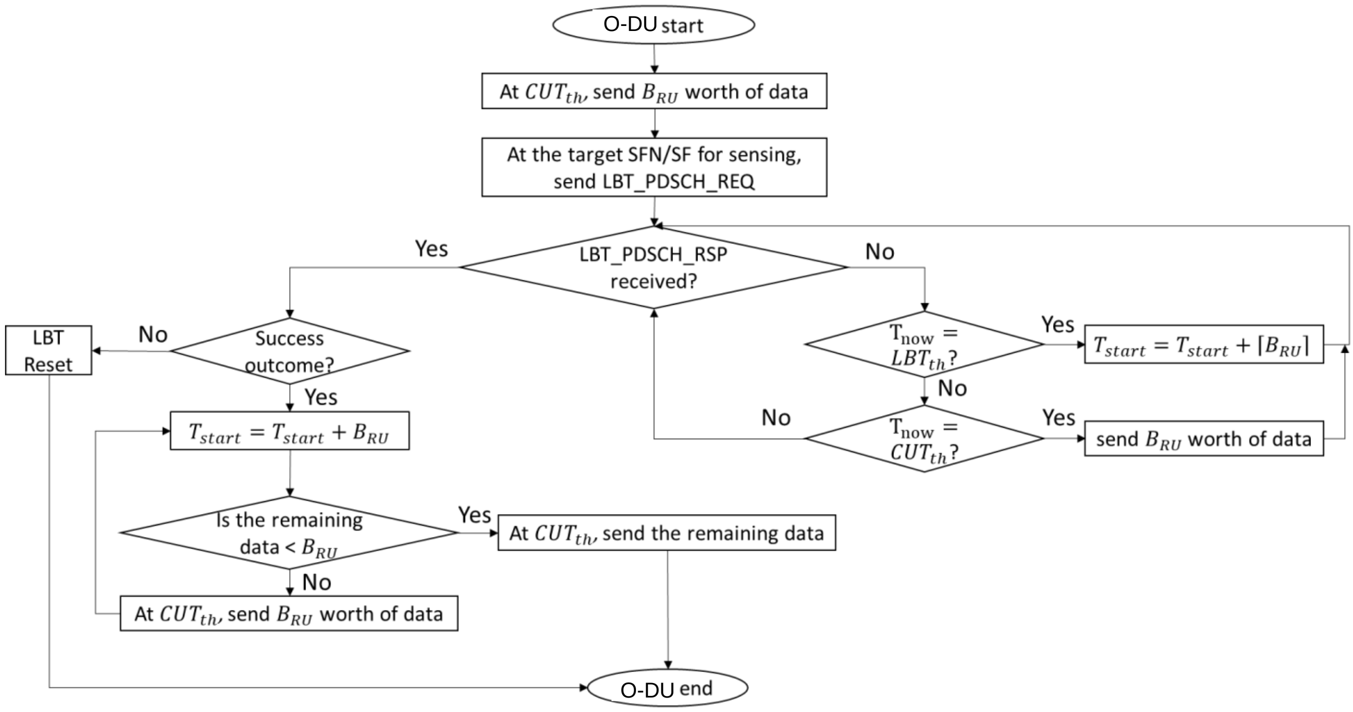
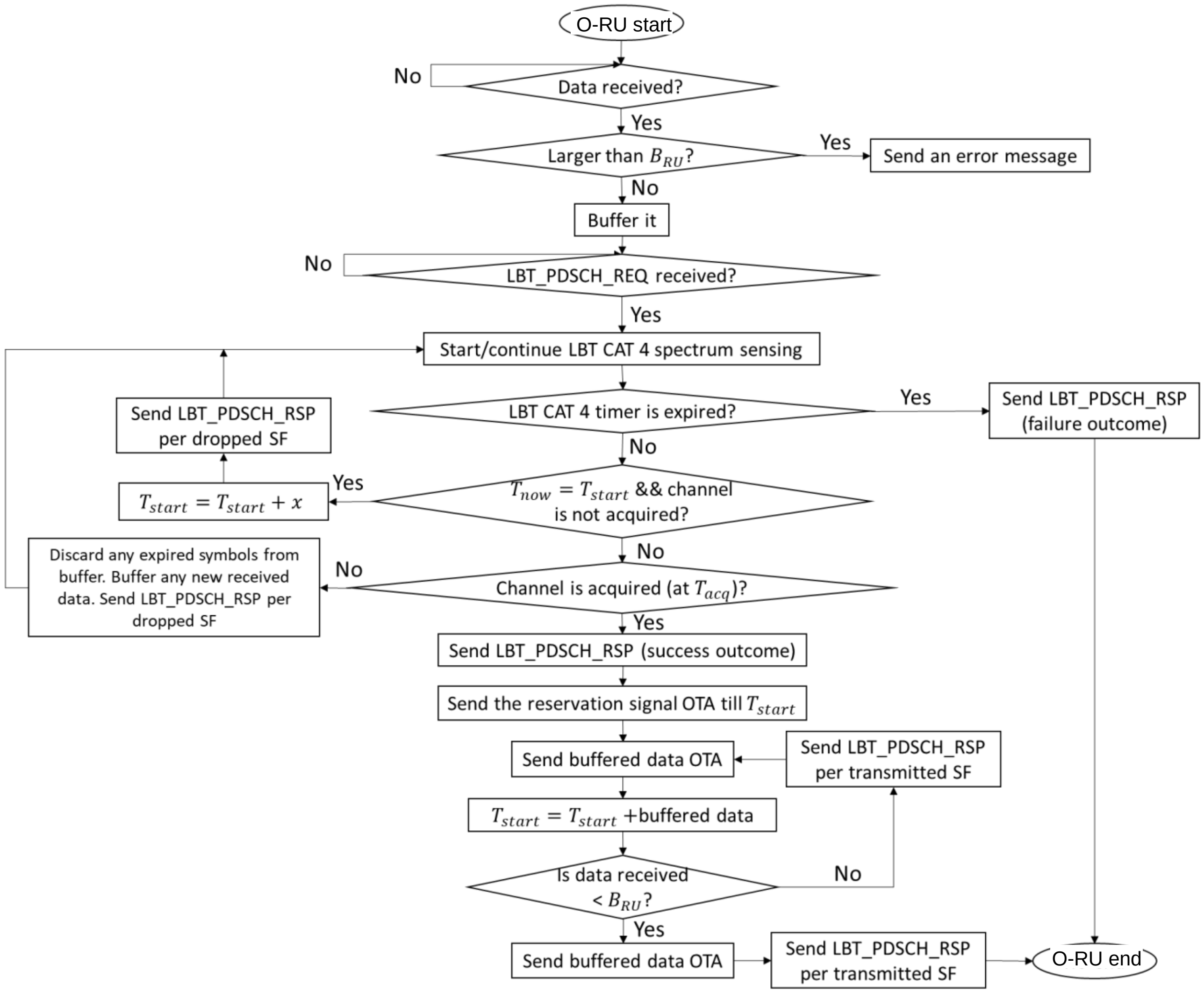


Figure G-1 : PDSCH Transmission Algorithm O-DU flowchart

where $\lceil a \rceil$: is the ceiling of number “a” to the nearest x, where x is 1 ms for normal SF and 0.5 ms for partially-filled SF.

1



2

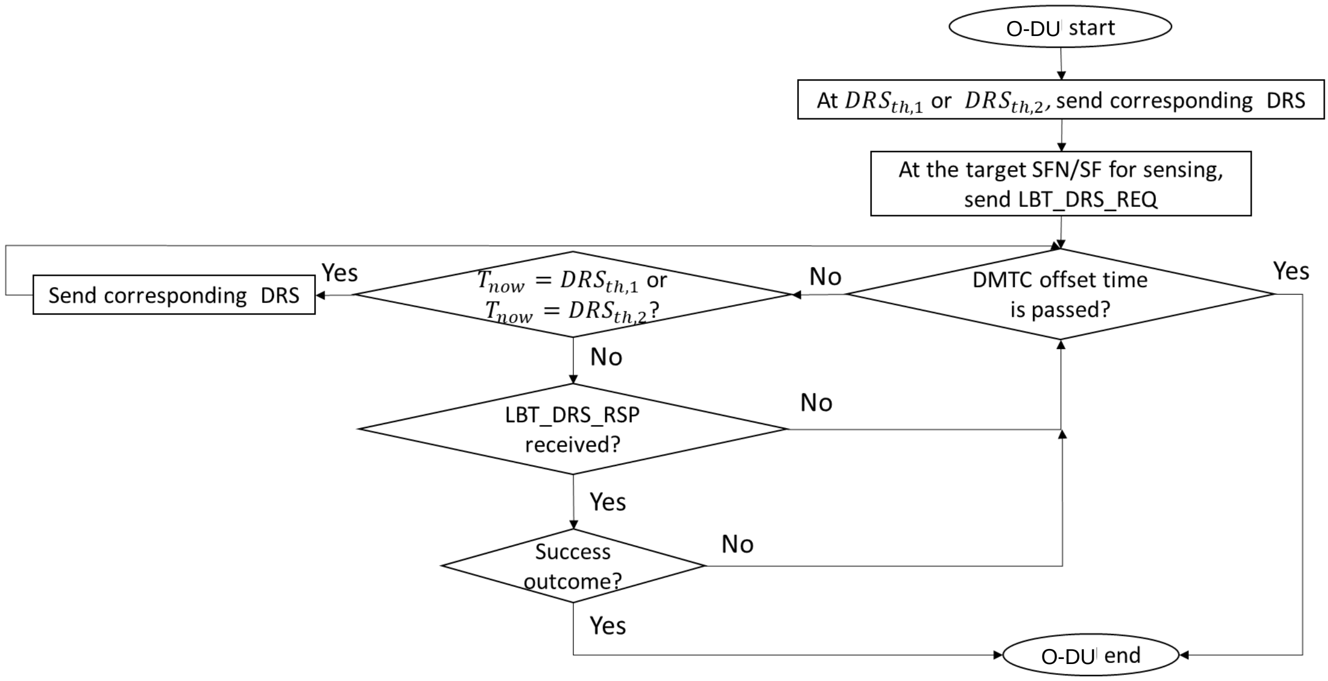
3 **Figure G-2 : PDSCH Transmission Algorithm O-RU flowchart**

4

5 # DRS Transmission Procedure

6 ## O-DU flowchart

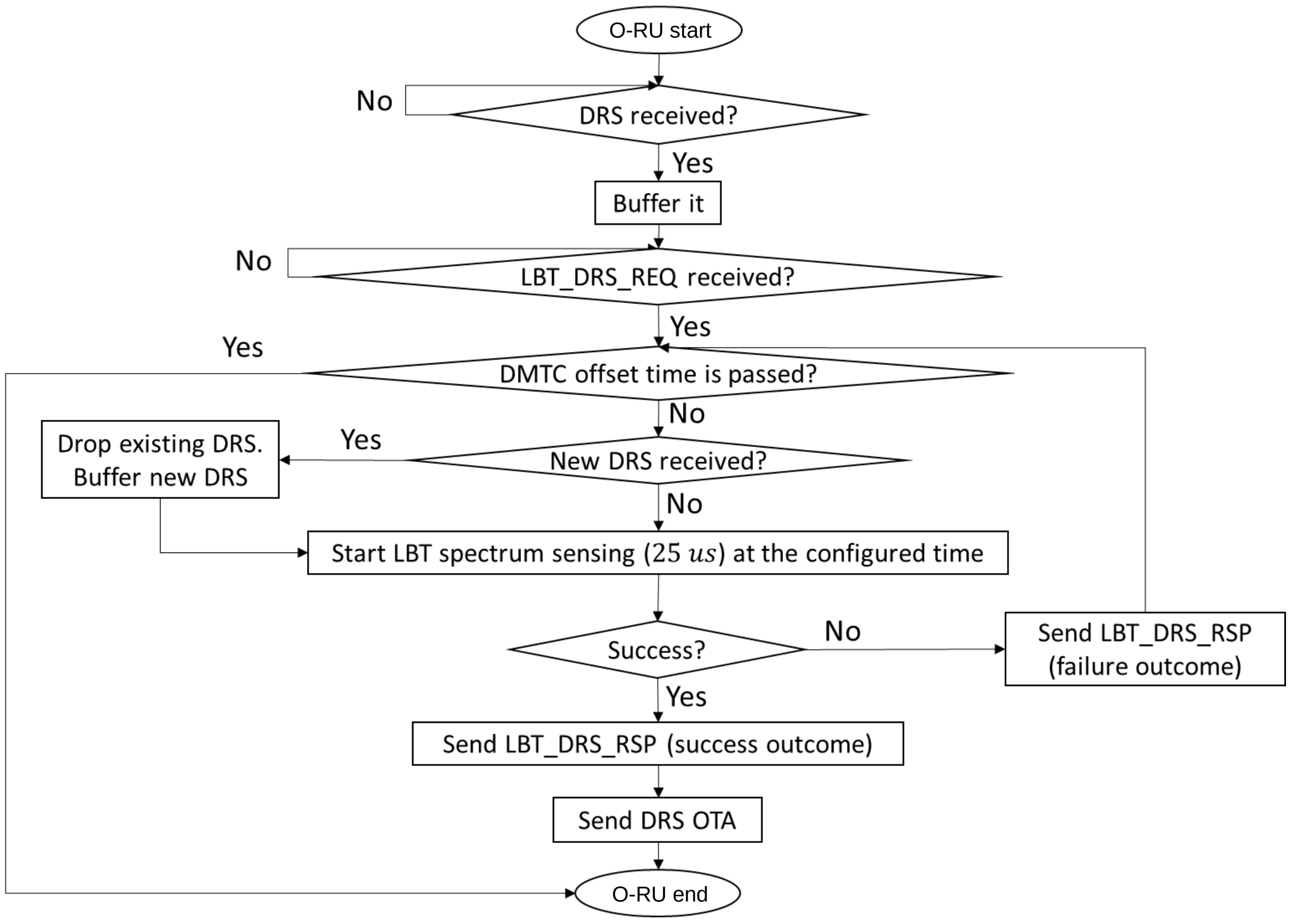
7 The O-DU flowchart covers both cases, where the O-RU does 25 us sensing or LBT CAT 4



1

2 **Figure G-3 : DRS Transmission Procedure O-DU flowchart**

3



4

5 **Figure G-4 : DRS Transmission Procedure O-RU flowchart – 25 µs sensing**

6

7

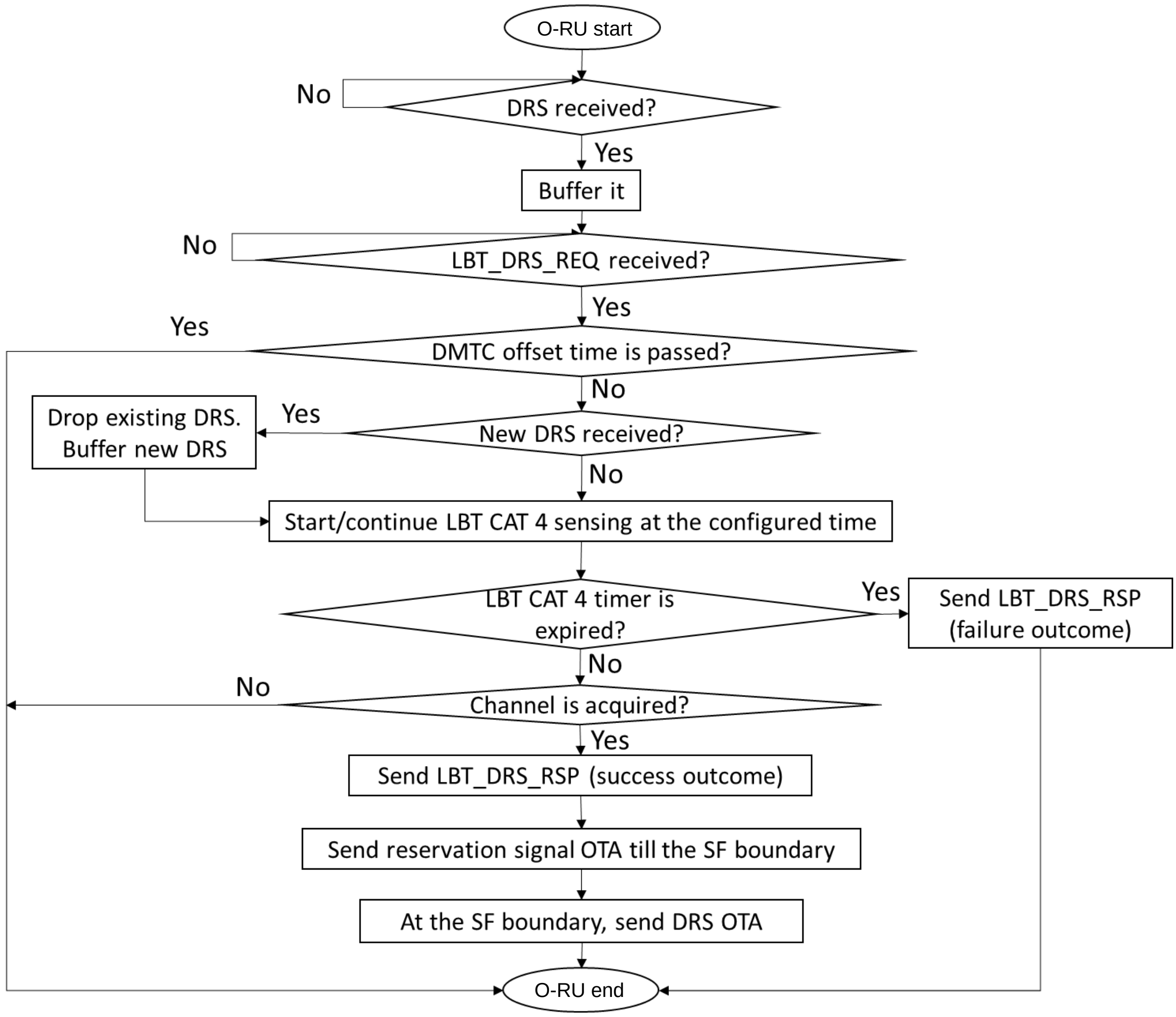


Figure G-5 : DRS Transmission Procedure O-RU flowchart – LBT Cat 4

Annex H S-Plane detailed frequency and phase error budget analysis, and future ITU-T clock types and classes reference

H.1 Reference documents

Section 9.2.1 includes the existing reference documents, and the following ones may be considered when available:

New revisions of ITU-T G.827x specifications are being developed and targeted for the end of 2018 or early 2019. They are expected to introduce higher-performance (more accurate) clock types including ePRTC/eGM, future T-BC and T-TSC classes. New HRM is also expected to cover more air interface use cases related to IEEE P802.1CM Categories.

The following references from different standard organizations are work in progress. These upcoming references cover the future evolution over the above existing standard references to fully meet O-RAN synchronization needs.

- IEEE P802.1CM (committee draft 2.2 made available 7, May 2018 for passing Rev com) Time-Sensitive Networking for Fronthaul
- IEEE 1588 Draft Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

At the time of releasing the O-RAN specification current revision, there is expectation but no binding commitment to include the new content of the above new specifications and revisions. Hence, when these specifications and new revisions are officially published, O-RAN must carefully review all the contents and determine their acceptance into O-RAN specification.

Section 9.2.3 specifies the SyncE clock specification, and the following ones may be considered when available:

- When used for clock synchronization, SyncE shall comply with the following additional ITU-T specifications:
    - ITU-T G.8262.1 (for eEEC)
- The use of eEECs is recommended as they provide less generated noise in normal operation and during network rearrangements, allowing a larger number of hops while meeting a given time error budget.

Section 9.2.4.2 specifies the PTP clock specification, and the following ones may be considered when available:

- A revision of IEEE 1588 is expected to be published by the end of 2018, and to remain backward compatible with its 2008 edition. The use of new features in this revision may be recommended in future editions of this document.
- Also expected by the end of 2018 or in 2019 are updates for Full Timing Support with tighter accuracy requirements to the recommendations specifying the time and phase synchronization aspects of packet networks (ITU-T G.8271), the network limits for time synchronization in packet networks (G.8271.1) and the timing characteristics of clocks (G.8273.1, G.8273.2, G.8273.3). These updates will contribute to meeting the IEEE 802.1CM requirements of its Case 1.2 in its Categories A and B, and of its Case 2 in its Categories A+ and A.

Section 9.2.5 specifies the accuracy specification, and the following ones may be considered when available:

- reference to future ITU-T G.8262.1 (for eEEC) for the SyncE jitter and wander requirements shall be added when available.

Section 9.3.1 specifies the clock types and classes specification, and the following ones may be considered when available:

- eEEC (per upcoming ITU-T G.8262.1)
- T-BC (per ITU-T G.8273.2: class A, B already specified, class C under development)
- T-TSC (per ITU-T G.8273.2: class A, B already specified, class C under development)

H.2 Frequency and time error budget analysis

This section provides the informative analysis to support budget allocation in Section 9.3.2 for a Full Timing Support network (as per ITU-T G.8271.1 for the limits, ITU-T.G.8273.2 for the clock definition). The analysis serves 2 purposes:

Considering T-BC Class B switches (as per ITU-T G. 8273.2) in a ITU-G G.8271.1 compliant network, the number of allowed switches to satisfy the allocated network limit is computed in detail as an example.

Note: the following configurations are outside the scope of this annex, and are therefore For Further Study:

- ITU-T G.8271.1 compliant networks using T-TCs instead of T-BCs.
- Non-ITU-T G.8271.1-compliant networks, such as ITU-T G.8271.2 ones.

Each network element in the fronthaul clock chain generates time error (including constant cTE and dynamic dTE_H, dTE_L), which will accumulate through the entire clock chain and be present at the O-RU UNI, as described in ITU-T G.8271.1 Appendix IV. In particular, accumulated dynamic time error will cause O-RU slave clock FFO (fractional frequency error) after clock recovery and filtering. Given O-RU must meet the 3GPP air interface frequency accuracy target (±50ppb), O-RU filtering is needed to filter the accumulated dynamic time error and reduce the frequency error down to an acceptable level. The allowed network limit (i.e. dynamic time error), reasonable O-RU filter bandwidth and acceptable frequency error after filtering are the result of a compromise exercise as shown in the following analysis.

The value of the O-RU filtering bandwidth is a key compromise, combined with the local oscillator thermal sensitivity:

- The higher filtering bandwidth, the faster frequency correction of the local oscillator thermal sensitivity and therefore the lower temporary accumulated time error under thermal variations, but the poorer efficiency in low pass filtering the dynamic noise seen on the UNI
- The lower filtering bandwidth, the better efficiency in low pass filtering the dynamic noise seen on the UNI, but the poorer frequency correction of the local oscillator thermal sensitivity and therefore the higher temporary accumulated time error under thermal variations.

Frequency error budget for Network limit (C1 and C2) :

Based on the above compromise explanation, a practical expectation of O-RU filtering max BW is set to 75mHz to start the analysis.

Table H-1 : O-DU Frequency Error Budget

O-DU class	A	B
• Consider O-DU PTP/SyncE master frequency error budget = …………….. (refer to note 1 in section 9.3.2.1)	±15 ppb	±5 ppb
• Consider O-RU total frequency error budget based on O-DU frequency error budget taken away from the 3GPP air interface (±50ppb) budget = ………..	±35ppb	±45ppb
• Further split the O-RU total frequency error budget as follows as an example of O-RU design: ○ FFO (O-RU slave clock) =…………………………………………… ○ FFO (O-RU internal additive frequency noise) = ………………………	 ±21ppb ±14ppb.	 ±27ppb ±18ppb
• With FFO (O-RU slave clock) value and filter BW = 75mHz, based on ITU-T SG15 Q13 C1730, Geneva, 5 – 16 December 2011: ***FFO (in ppb) = ±2*π * |dTE_{L+H}|(in ns)*filter BW(in Hz)*** ⇨ FFO (O-RU slave clock) = $2\pi * \|dTE_{L+H}\| *$ filter BW ⇨ $\|dTE_{L+H}\|$ = FFO (O-RU slave clock)/(2π * filter BW) = …………………… which is the max allowed network noise limit (between O-DU UNI and O-RU UNI) guaranteeing FFO at the output of the O-RU filter with 75mHz BW. Note that after this network noise limit is agreed in O-RAN spec, it is up to O-RU vendor implementation to select filter BW (not necessarily 75mHz) to trade off the internal budget split between FFO (O-RU slave clock) and FFO (O-RU internal additive frequency noise) as long as the O-RU total frequency error budget (±35ppb or ±43ppb) is still met.	±45ns	±57ns
• Based on G.8271.1 Appendix IV guidance to calculate accumulated error: ⇨ Total dynamic noise = RMSsum (dTE_{L+H}) ⇨ $\|dTE_{L+H}\|$ = RMSsum ($\|dTE_{L+H}\|$ of all nodes excluding O-RU's T-TSC) Consider the model of clock chain of n T-BC clocks (between O-DU UNI to RU UNI) ITU-T G.8273.2 (class B) switch: $\|dTE_L\| = 20$ns, $\|dTE_H\| = 35$ns ⇨ $\| dTE_{L+H} \|$ limit = sqrt [$n*20^2 + 35^2$] ns = ……………………………………… ⇨ n = ($\|dTE_{L+H}\|^2 - 35^2)/20^2$, ……………………………………………………… the maximum number of class B T-BCs in each chain (excluding O-DU)	±45ns 2	±57ns 5

Time error budget for network limit (C1 and C2) :

Using existing class B T-BCs, and considering no time error contribution by the fiber asymmetry nor from two master ports of the same T-BC, then:

Following G.8271.1 Appendix IV guidance to calculate accumulated error with the following clock chain models:

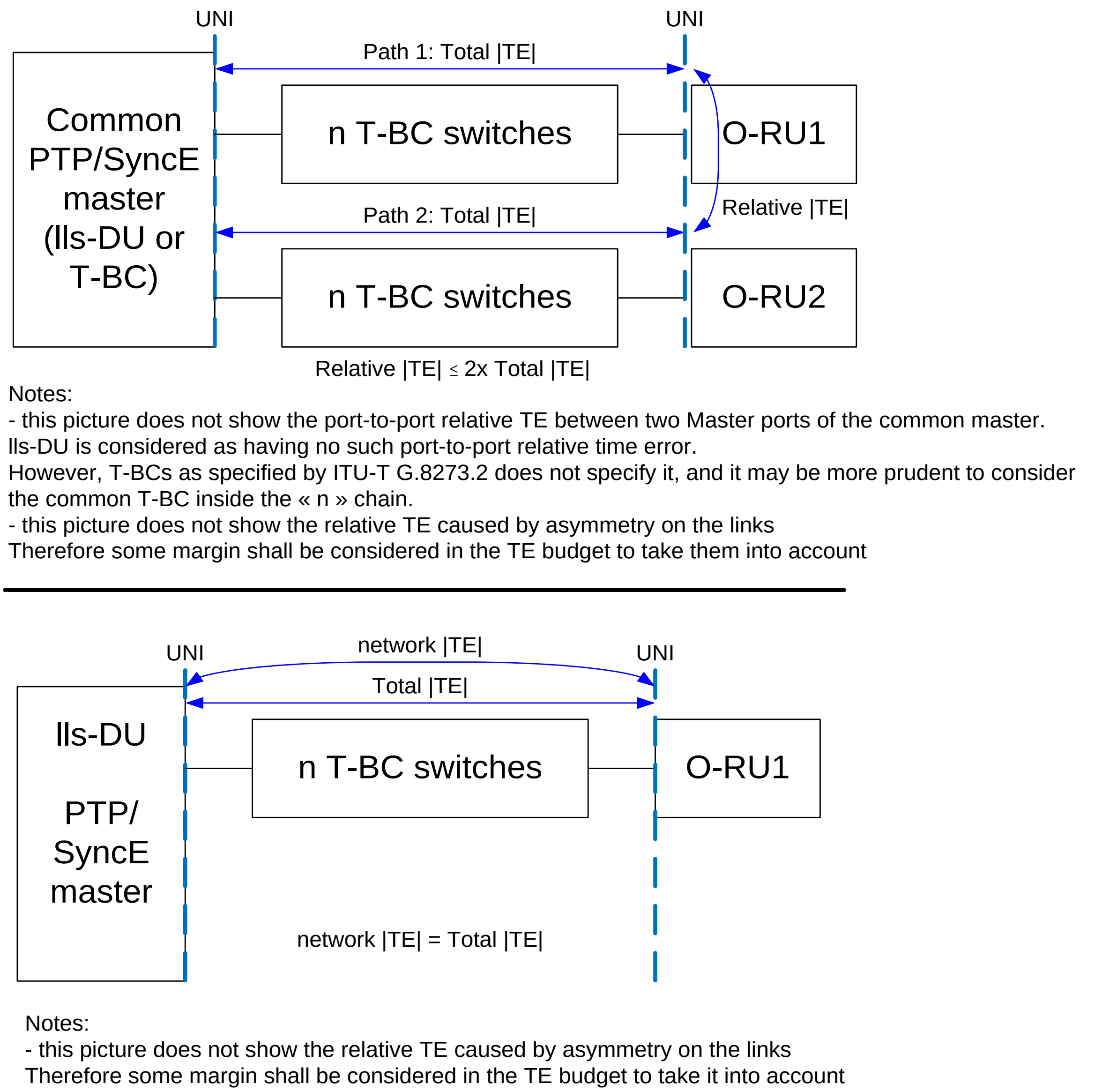


Figure H-1 : clock chain model for analysis

As per ITU-T G.8271.1 Appendix IV:

Total |TE| = sum (|cTE| of n nodes) + RMSsum (|dTE_L| of n nodes and |dTE_H| of last node)

$= n*|cTE| + \text{sqrt}(n*|dTE_L|^2 + |dTE_H|^2)$

However, the O-RU's time error budget (cTE, $|dTE_L|^2 + |dTE_H|^2$) has already been taken into account in the budget (see tables below), so only the T-BCs in the network are included in "n" and their contribution is limited to cTE and dTE_L. As a result, the above general formula can be further simplified for the chain of n T-BCs excluding the O-RU's T-TSC:

Total $|TE| = n*|cTE| + \text{sqrt}(n*|dTE_L|^2)$,

Where a node is based on T-BC Class B switch with the following noise generation specification:

Constant time error = |cTE| = 20ns

Low-band dynamic error = $|dTE_L|$ = 20ns

Table H-2 : O-RU Time Error Budget

For O-RU type= and relative |TE| limit = to meet category	Enhanced 60ns Cat A	Class B 100ns Cat B	Enhanced 190ns Cat B
⇨ n*20 + sqrt (n*20^2) < relative |TE| /2 …………………………….	30ns	50ns	95ns
⇨ maximum n value, number of class B T-BCs on each branch to common network element (either O-DU or T-BC): ……………… (when n=0, this means that only direct connection is supported.)	0	1	2 (note1)
⇨ Remaining relative TE margin to be assigned to fiber asymmetry and relative TE between two ports of the common network element	60ns	20ns	53ns

For O-RU type= and network |TE| limit = to meet category	class B 95ns Cat C	Enhanced 140ns Cat C
⇨ n*20 + sqrt (n*20^2) < absolute |TE|		
⇨ maximum n value, number of class B T-BCs on each branch to O-DU:	2 (note1)	4
⇨ Remaining absolute TE margin to be assigned to fiber asymmetry on each branch:	12ns	20ns

Note 1:
Using n=3 on each branch, the total relative time error per branch is equal to 60+sqrt(3*20²) = 95ns, leaving no margin for fiber asymmetry or relative TE between two ports. It is therefore recommended to limit the number of T-BCs to 2.

H.3 Frequency error budget for Network limit (C3) :

- Based on the above compromise explanation, a practical expectation of O-RU filtering max BW is set to 75mHz to start the analysis
- Based on G.8272, PRTC/T-GM MTIE (during lock) specification can be used to describe PRTC/T-GM dynamic noise generation:

Table H-3 : Wander Generation (MTIE)

MTIE limit (us)	Observation interval (s)
$0.275 \times 10^{-3}\,\tau + 0.025$	$0.1 < \tau \leq 273$
0.10	$\tau > 273$

⇨ Given O-RU filtering max BW = 75mHz, it corresponds to observation interval τ = 1/(π*75mHz) = 4.2s.
From the above table, MTIE limit (with τ = 4.2s) = 26.2ns pk-pk.
From this MTIE number, the value of dTE_L is computed as 13.1ns.

⇨ Besides MTIE, which can be treated as dynamic noise during lock condition, there is additional consideration of PRTC/T-GM during holdover condition. Potential semi-static frequency drift could happen during holdover, ±2ppb is reserved based on ITU-T G.8271.1 Appendix V PRTC failure scenario (b) which permits 400ns holdover limit for short period of 5 minutes.

Table H-4 : Network (C3) Frequency Error Budget

PRTC class	A
Consider PRTC PTP/SyncE master frequency error budget =……………………………………… (refer to note 1 in section 9.3.2.2)	±2 ppb
Consider O-RU total frequency error budget based on O-DU frequency error budget taken away from the 3GPP air interface (±50ppb) budget = ………………………………………………………	±48ppb
• Further split the O-RU total frequency error budget as follows as an example of O-RU design: ○ FFO (O-RU slave clock) =………………………………………………………. ○ FFO (O-RU internal additive frequency noise) = ……………………………………	 ±30ppb ±18ppb

• With FFO (O-RU slave clock) value and filter BW = 75mHz, based on ITU-T SG15 Q13 C1730, Geneva, 5 – 16 December 2011: ***FFO (in ppb) = ±2*π * |dTE$_{L+H}$|(in ns)*filter BW(in Hz)*** ⇨ FFO (O-RU slave clock) = 2π * |dTE$_{L+H}$| * filter BW ⇨ |dTE$_{L+H}$| = FFO (O-RU slave clock)/(2π * filter BW) = ………………………………… which is the max allowed network noise limit (between O-DU UNI and O-RU UNI) guaranteeing FFO at the output of the O-RU filter with 75mHz BW. Note that after this network noise limit is agreed in O-RAN spec, it is up to O-RU vendor implementation to select filter BW (not necessarily 75mHz) to trade off the internal budget split between FFO (O-RU slave clock) and FFO (O-RU internal additive frequency noise) as long as the O-RU total frequency error budget (±35ppb or ±45ppb) is still met.	±63ns
• Based on G.8271.1 Appendix IV guidance to calculate accumulated error: ⇨ Total dynamic noise = RMSsum (dTE$_{L+H}$) ⇨ |dTE$_{L+H}$| = RMSsum (|dTE$_{L+H}$| of all nodes including PRTC/T-GM but not O-RU's T-TSC) • Consider the model of clock chain of PRTC/T-GM and n T-BC switches (between PRTC input to O-RU UNI) and using PRTC/T-GM MTIE specification and Class B dTE$_{L+H}$ specification. ⇨ PRTC/T-GM dynamic noise = |dTE$_{L+H}$| = MTIE/2 based on max 75mHz O-RU filter BW assumption: ………………………………………………………………………………. ⇨ T-BC Class B switch dynamic noise = |dTE$_L$| = 20ns, |dTE$_H$| = 35ns ⇨ |dTE$_{L+H}$| = sqrt(13^2 + n*20^2 +35^2) ns= ………………………………………………………. ⇨ Maximum n= (|dTE$_{L+H}$|^2 - 35^2 - 13^2) / 20^2 = ………………………………………………… the maximum number of class B T-BCs in each chain (excluding PRTC)	13ns 63ns 6

Time error budget for network limit (C3):

Using current class B T-BCs, and considering no time error contribution by the fiber asymmetry nor from two master ports of the same T-BC, then:

Following G.8271.1 Appendix IV guidance to calculate accumulated error and the clock chain model as shown in **Figure H-1**:

As per ITU-T G.8271.1 Appendix IV:

Total |TE| = sum (|cTE| of n nodes) + RMSsum (|dTE$_L$| of n nodes and |dTE$_H$| of last node)

$$= n*|cTE| + \mathrm{sqrt}\,(n*|dTE_L|^2 + |dTE_H|^2)$$

However, the O-RU's time error budget (cTE, $|dTE_L|^2 + |dTE_H|^2$) has already been taken into account in the budget (see tables below), so only the T-BCs in the network are included in "n" and their contribution is limited to cTE and dTE_L. As a result, the above general formula can be further simplified for the chain of n T-BCs excluding the O-RU's T-TSC:

Total $|TE| = n*|cTE| + \mathrm{sqrt}\,(n*|dTE_L|^2)$,

Where a node is based on T-BC Class B switch with the following noise generation specification:

Constant time error = |cTE| = 20ns

Low-band dynamic error = $|dTE_L|$ = 20ns

- For analysis to meet category A and B, refer to time error budget for network limit (C1 and C2) section. ITU-T G.8271.1 does not specify the port-to-port TE on the common node driving the two branches. A conservative option should consider the input port of this common node as the place where the two branches start, and therefore the common T-BC as part of the n nodes in each branch.

- For analysis to meet category C, refer to ITU-T G.8271.1 Appendix V guidance, a more completed analysis is shown to support PRTC/T-GM with 20 T-BC Class B switches and some margin left for fiber asymmetry, network rearrangement/holdover or end application (i.e. O-RU) holdover. O-RAN can possibly support even a longer clock chain since O-RU |TE| is lower than the allocation in ITU-T analysis. However, since 20 T-BC Class B switches is already a very long chain, there is no need to further analyze longer chain possibility.

Summary of allowed number of switches:

The maximum allowed number of switches shall be determined based on the smallest allowed number constraint by

- Frequency error budget

  AND

- Operator-chosen most constraint time error budget category

Table H-5 : Network (C3) Frequency Error Budget

Frequency Error Network limit	C1 and C2, class A O-DU	C1 and C2, class B O-DU	C3	Comment
Absolute Frequency error budget between time source and O-RU	2	5	6	Any branch must not exceed this number of T-BCs from O-DU or PRTC/T-GM to meet 50ppb frequency accuracy at the air interface.

Table H-6 : Network (C3) Time Error Budget

Time Error Network limit	C1 and C2, class A O-DU	C1 and C2, class B O-DU	C3	Comment
Relative Time error budget (Cat A w/enhanced T-TSC) between O-RUs	0	0	0	Any branch must not exceed this number of T-BCs from common T-BC to meet target relative time error limit at the air interface.
Relative Time error budget (Cat B w/ class B T-TSC) between O-RUs	1	1	1	
Relative Time error budget (Cat B w/enhanced T-TSC) between O-RUs	2	2	2	
Absolute Time error budget (Cat C w/class B T-TSC) between time source and O-RU	2	2	20 (further limit to 6 due to freq. limit)	Any branch must not exceed this number of T-BCs from O-DU or PRTC/T-GM to meet 1500ns absolute time error limit at the air interface.
Absolute Time error budget (Cat C w/enhanced T-TSC) between time source and O-RU	4 (further limit to 2 due to freq. limit)	4	20 (further limit to 6 due to freq. limit)	

Note 1 : More switches are allowed if the same above limits can be met by specific switch equipment (exceeding G8273.2 T-BC Class B performance) or future better performance T-BC class.

Note 2 : The analysis on the number of switches (for time error budget) is meant to rough estimate and excludes both fiber asymmetry factor and relative TE between two ports of the common network element. It is operator’s responsibility to control these two parameters. The analysis for each time error budget (specific category) has some left-over margin that could be used to cover them. If the left-over margin is not enough, the alternative is to reduce the allowed number of switches.

Annex I Precoding and Examples:

Case 1 Tx Diversity 1-CRS Port Ant0, 1 PRB:

At the O-DU

- For single Tx case, $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)}-1)$ M_q_sym modulation bits belonging to PDSCH ANT0 are packed into a PRB. All CRS REs for ANT0 are packed into a PRB for transmission and are unpacked at the O-RU.

At the O-RU

- At the O-RU, for single antenna port single Tx a single layer is used and mapping is defined as $x^{(0)}(i)=d^{(0)}(i)$
- O-RU needs to map CRS REs into antenna ports using crsReMask, crssymbolNumber and crsShift. Follow case 5

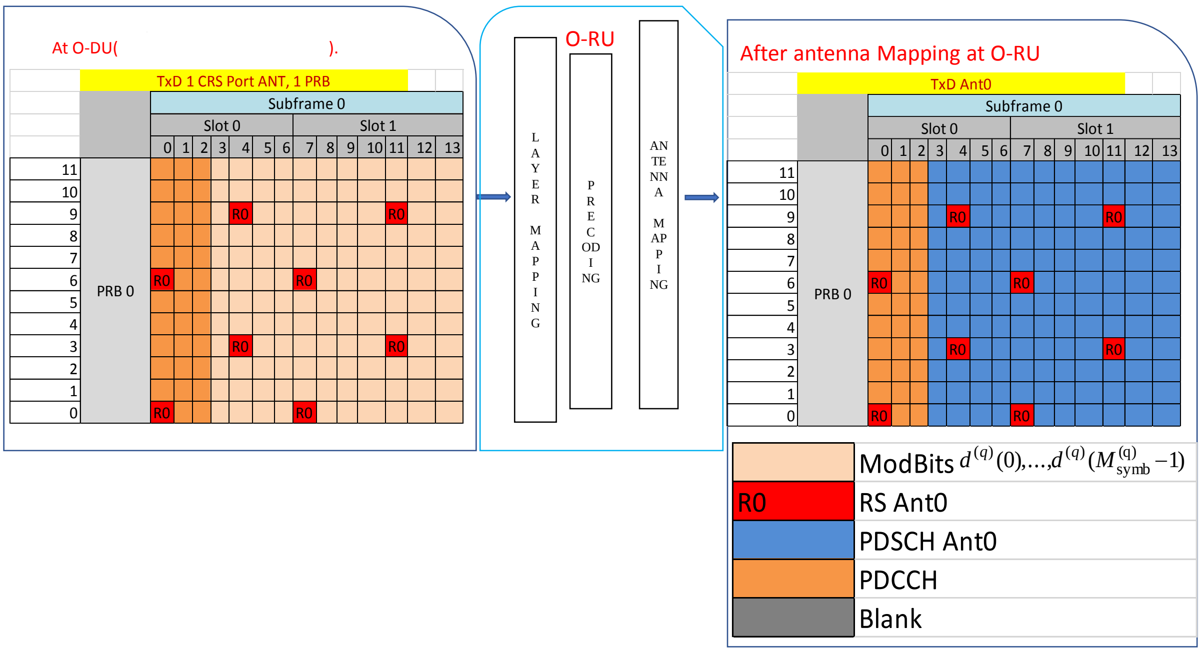


Figure I-1 : Single Tx 1-CRS Port Ant0, 1 PRB

Case 2 Tx Diversity 2-CRS Port Ant0, Ant1, and 1 PRB:

Case 2.1 At the O-DU

- For TxD case, $d^{(q)}(0),\ldots,d^{(q)}(M_{\text{symb}}^{(q)}-1)$ M_q_sym modulation bits belonging to PDSCH are packed into a PRB. All CRS REs for ANT0 and ANT1 are packed into the same PRB for transmission and are unpacked at the O-RU.

Case 2.2 At the O-RU

- At the O-RU, for two antenna port TxD 2 layers are used and mapping is defined as $\begin{array}{l} x^{(0)}(i)=d^{(0)}(2i) \\ x^{(1)}(i)=d^{(0)}(2i+1) \end{array}$

- For TxD, information for 2 layers are packed into a PRB for transmission and are unpacked at the O-RU. At the O-RU, after layer mapping and, precoding, CRS REs for 2 antenna ports are mapped to the appropriate RE positions and rest are left blank (gray-shaded REs) as illustrated below. Follow Case 5.

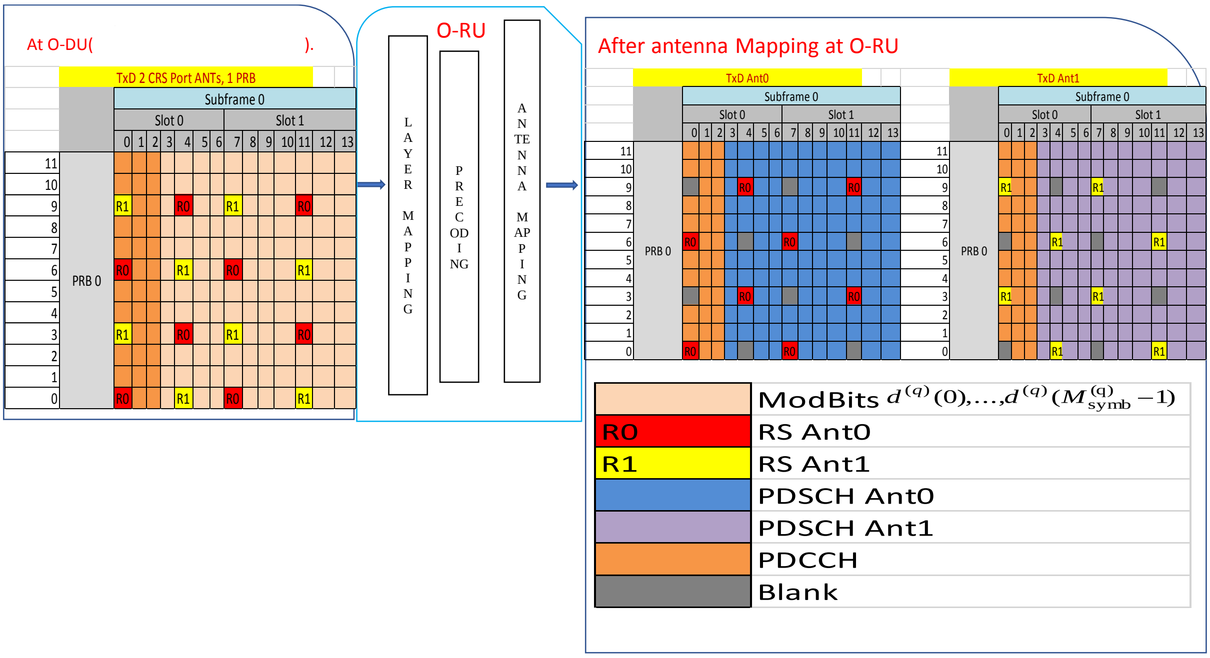


Figure I-2 : Tx Diversity 2-CRS Port Ant0, Ant1, and 1 PRB

Case 3 Tx Diversity 4-CRS Port Ant0,1,2,3 and 1 PRB:

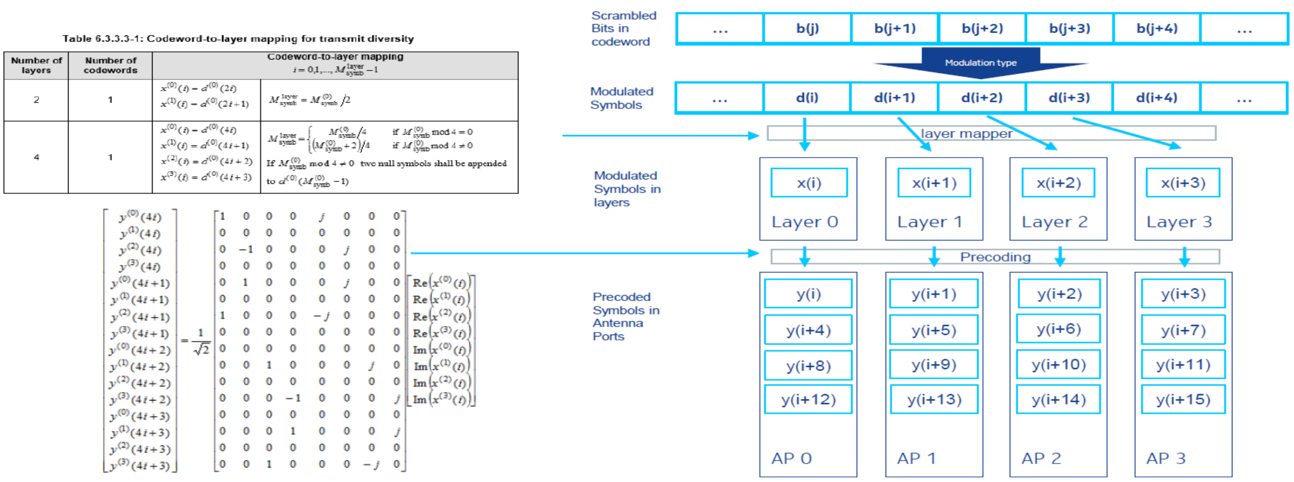


Figure I-3 : Case 3 Layer Mapping

Case 3.1 At the O-DU

- For TxD case, $d^{(q)}(0),...,d^{(q)}(M_{symb}^{(q)}-1)$ M_q_sym modulation bits belongs to PDSCH are packed to into a PRB. All CRS RE's for ANT0, ANT1, ANT2 and ANT3 are packed into that same PRB for transmission and are unpacked at the O-RU.

Case 3.2 At the O-RU

- At the O-RU, for two antenna port TxD 4 layers are used and mapping is defined as
$$x^{(0)}(i) = d^{(0)}(4i)$$
$$x^{(1)}(i) = d^{(0)}(4i+1)$$
$$x^{(2)}(i) = d^{(0)}(4i+2)$$
$$x^{(3)}(i) = d^{(0)}(4i+3)$$

- For TxD, user data RE’s for 4 layers are packed into a PRB for transmission and are unpacked at the O-RU. At the O-RU, after layer mapping and precoding, CRS REs for 4 antenna ports are mapped to the appropriate RE position and rest are left blank as illustrated below.

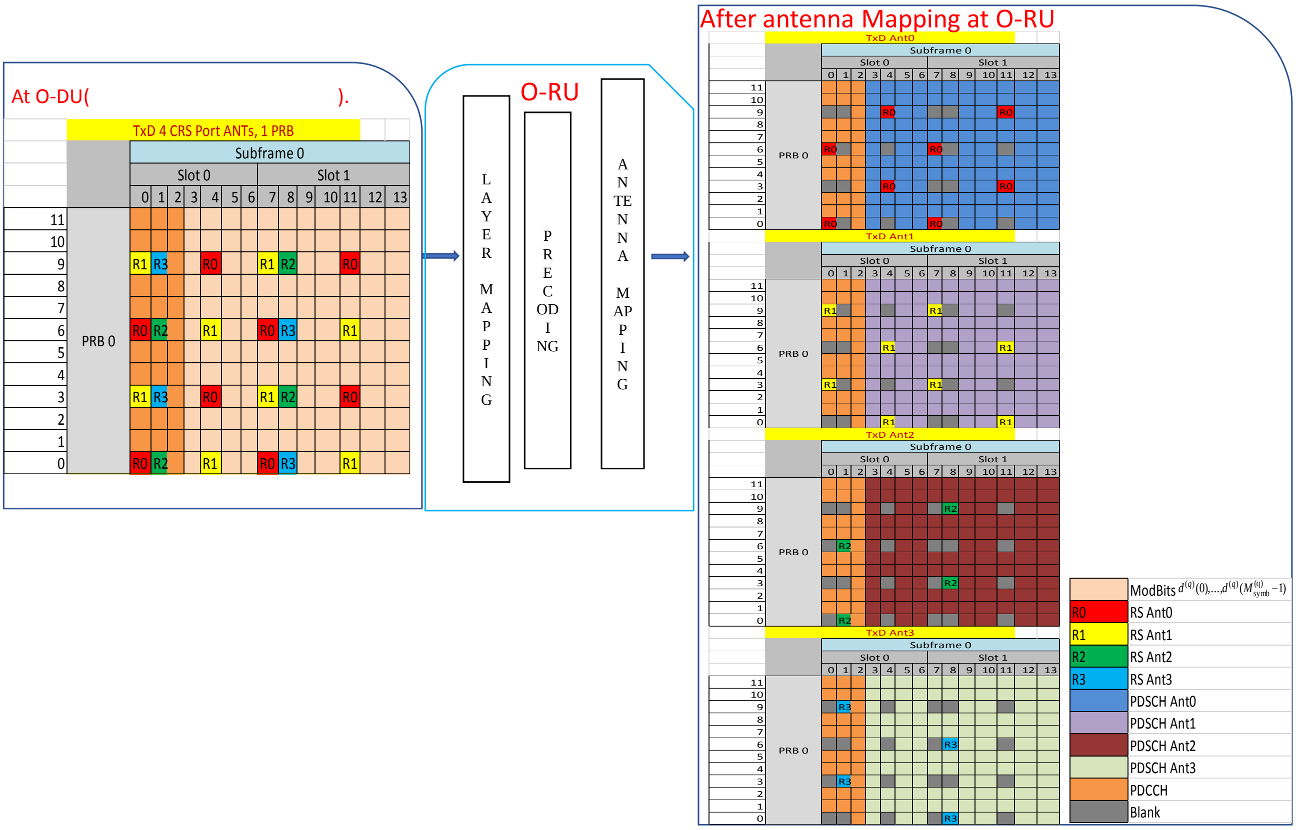


Figure I-4 : Tx Diversity 4-CRS Port Ant0,1,2, 3, and 1 PRB

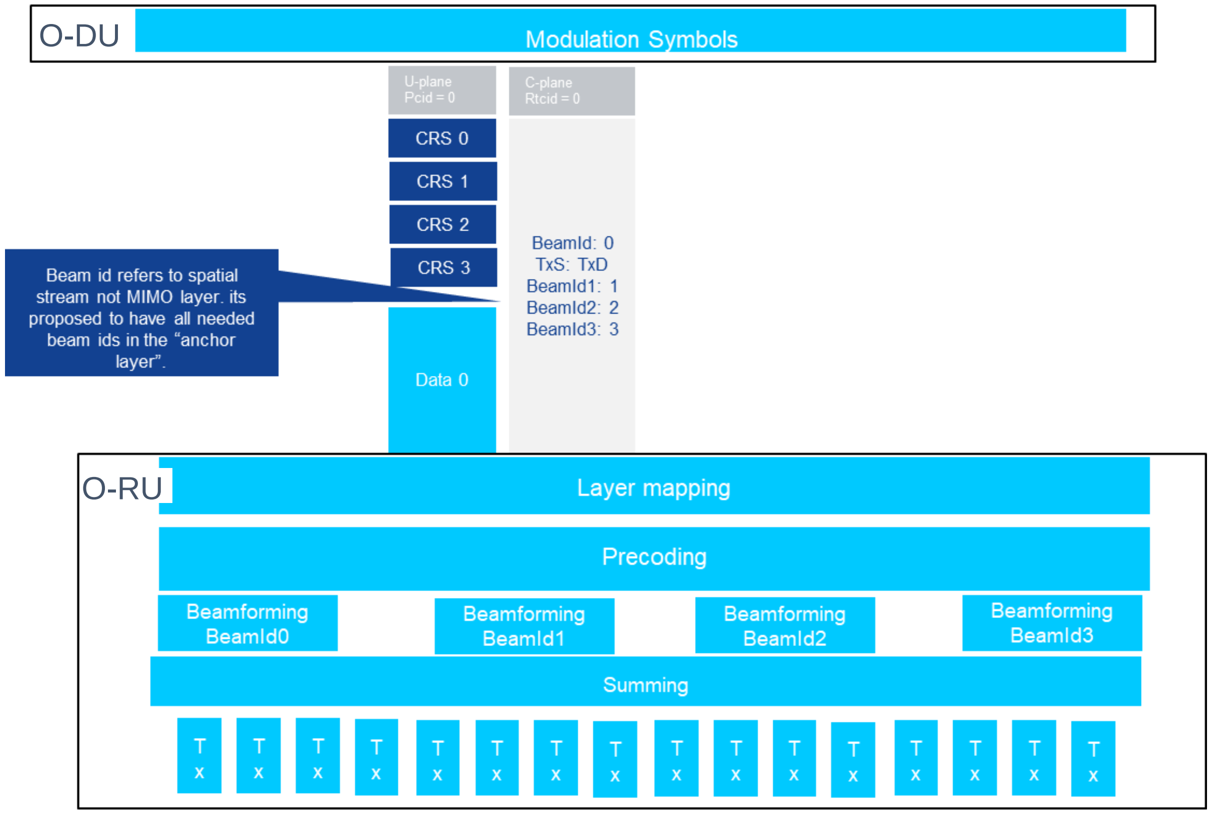


Figure I-5 : TxD – 4 CRS Ports

Case 4 TM3/TM4 3 Layers, 4 Antenna ports:

One symbol from each of layers is linearly mapped to each antenna port.

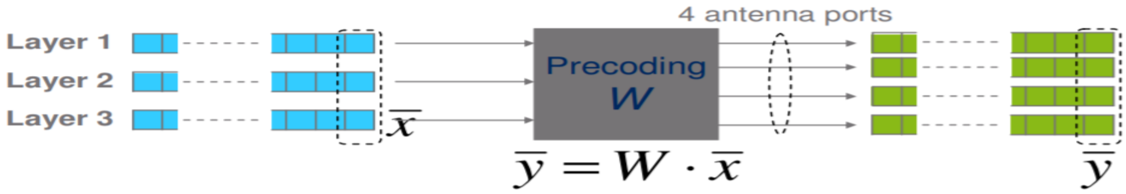


Figure I-6 : Case 4 Layer Mapping

For TM3/TM4, REs belonging to all the antenna ports are mapped to the zeroth layer at the O-DU. The O-RU should consider layer zero's CRS REs for CRS mapping for all the antenna ports using reMask bit field; CRS RE positions in the non-zero layers can be ignored as illustrated below (gray-shared REs in the O-DU represent CRS RE positions that are not populated by the O-DU.

$$
\begin{aligned}
x^{(0)}(i) &= d^{(0)}(3i) \\
x^{(1)}(i) &= d^{(0)}(3i+1) \quad M_{\mathrm{symb}}^{\mathrm{layer}} = M_{\mathrm{symb}}^{(0)} \big/ 3 \\
x^{(2)}(i) &= d^{(0)}(3i+2)
\end{aligned}
$$

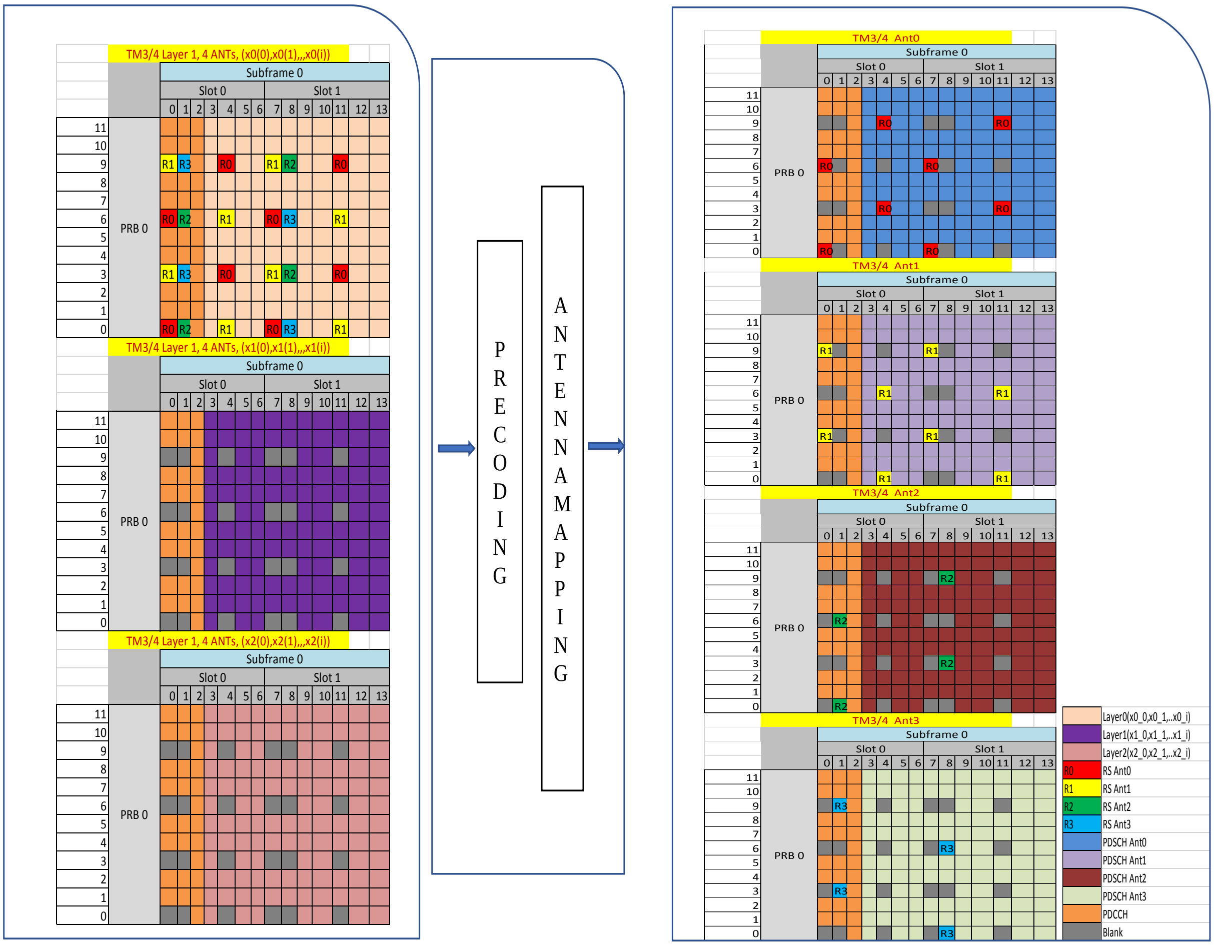


Figure I-7 : TM3/TM4, 3 Layers 4 Antenna Ports

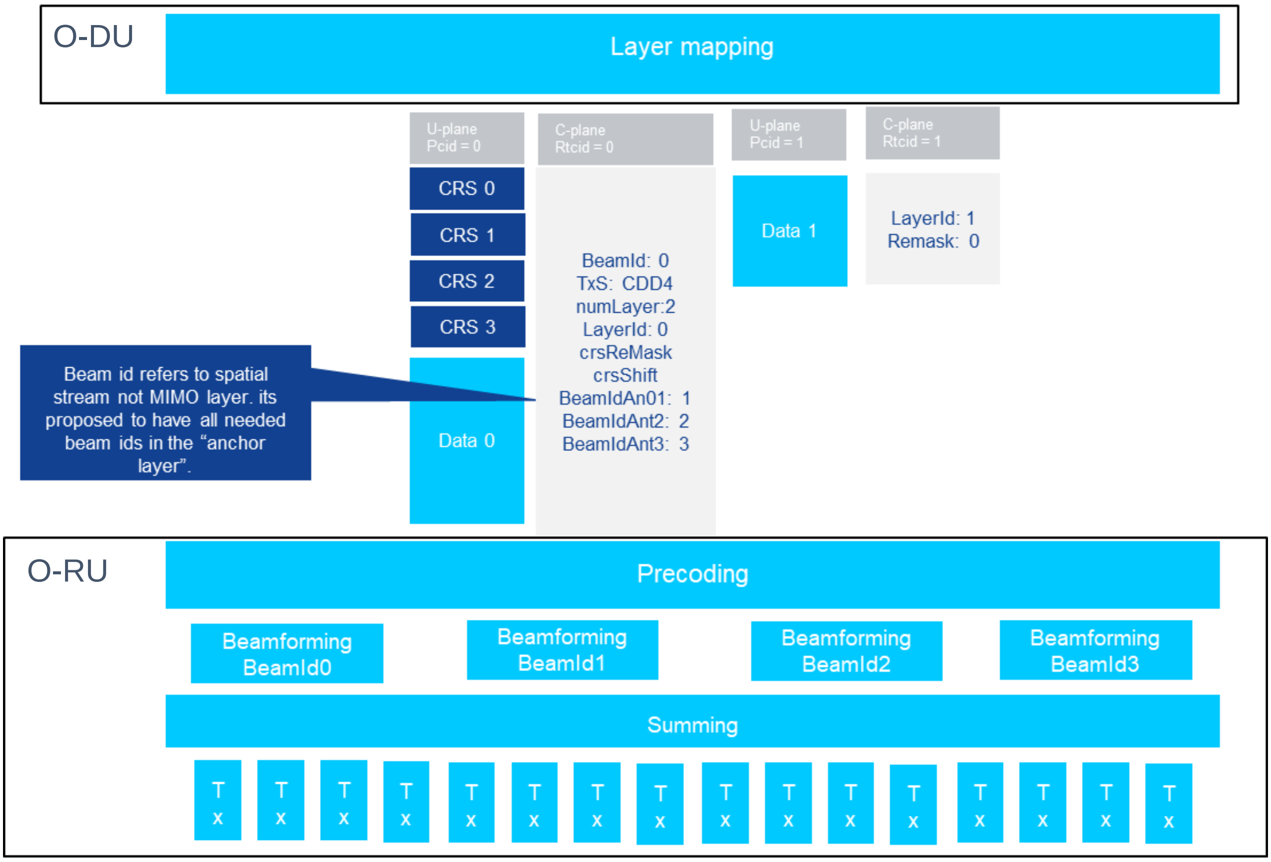


Figure I-8 : TM3 rank 2/4 CRS Ports

CRS location assignment: reMask Bit position for Layer0, 2 and 4 for all possible vShift cases

Note: CRS shift pattern for antenna port 0 when $3 \leq$ vshift ≤ 5 is same as $0 \leq$ vshift ≤ 3 for antenna port 1 resulting in only three possible vshift for a 2-antenna port case and this holds good for 4-antenna port case as well. Hence the crsReMask is differentiated with the crsShift field. If the O-RU identifies crsShift as 1, then shift in CRS belongs to each antenna have to read properly.

Table I-1 : CRS Location Assignment For layer1

	Layer1				
		crsSymNum 0 and 7		crsSymNum 4 and 11	
vshift	crsShift	crsReMask	PosInd in Bit Ant 0	crsReMask	PosInd in Bit Ant 0
0	0	0000 0100 0001	0,6	0010 0000 1000	3,9
1	0	0000 1000 0010	1,7	0100 0001 0000	4,10
2	0	0001 0000 0100	2,8	0100 0010 0000	5,11
3	0	0010 0000 1000	3,9	0000 0100 0001	6,0
4	0	0100 0001 0000	4,10	0000 1000 0010	7,1
5	0	1000 0010 0000	5,11	0001 0000 0100	8,2

crsReMask is not repeated for all for vShift combinations, hence crsShift is always indicated as 0.

Table I-2 : CRS Location Assignment For layer2

	Layer2						
		crsSymNum 0 and 7			crsSymNum 4 and 11		
vshift	crsShift	crsReMask	PosInd in Bit Ant 0	PosInd in Bit Ant 1	crsReMask	PosInd in Bit Ant 0	PosInd in Bit Ant 1
0	0	0010 0100 1001	0,6	3,9	0010 0100 1001	3,9	0,6
1	0	0100 1001 0010	1,7	4,10	0100 1001 0010	4,10	1,7
2	0	1001 0010 0100	2,8	5,11	1001 0010 0100	5,11	2,8
3	1	0010 0100 1001	3,9	6,0	0010 0100 1001	6,0	3,9
4	1	0100 1001 0010	4,10	7,1	0100 1001 0010	7,1	4,10
5	1	1001 0010 0100	5,11	8,2	1001 0010 0100	8,2	5,11

crsReMask for vShift 0 is same as for vShift 3 case but RE locations are different for Ant 0 and 1. Hence indicating 1 for 3,4,5 vShift cases, thereby O-RU must pick the positions as interpreted in above table.

Table I-3 : CRS Location Assignment For layer4

	Layer4						
		crsSymNum 0 and 7			crsSymNum 4 and 11		
vshift	crsShift	crsReMask	PosInd in Bit Ant 0	PosInd in Bit Ant 1	crsReMask	PosInd in Bit Ant 0	PosInd in Bit Ant 1
0	0	0010 0100 1001	0,6	3,9	0010 0100 1001	3,9	0,6
1	0	0100 1001 0010	1,7	4,10	0100 1001 0010	4,10	1,7
2	0	1001 0010 0100	2,8	5,11	1001 0010 0100	5,11	2,8
3	1	0010 0100 1001	3,9	6,0	0010 0100 1001	6,0	3,9
4	1	0100 1001 0010	4,10	7,1	0100 1001 0010	7,1	4,10
5	1	1001 0010 0100	5,11	8,2	1001 0010 0100	8,2	5,11

	Layer4						
		crsSymNum 1			crsSymNum 8		
vshift	crsShift	crsReMask	PosInd in Bit Ant 2	PosInd in Bit Ant 3	crsReMask	PosInd in Bit Ant 3	PosInd in Bit Ant2
0	0	0010 0100 1001	0,6	3,9	0010 0100 1001	0,6	3,9
1	0	0100 1001 0010	1,7	4,10	0100 1001 0010	1,7	4,10
2	0	1001 0010 0100	2,8	5,11	1001 0010 0100	2,8	5,11
3	1	0010 0100 1001	3,9	6,0	0010 0100 1001	3,9	6,0
4	1	0100 1001 0010	4,10	7,1	0100 1001 0010	4,10	7,1
5	1	1001 0010 0100	5,11	8,2	1001 0010 0100	5,11	8,2

crsReMask for vShift 0 is same as for vShift 3 case but RE locations are different for Ant 0 and 1. Hence indicating 1 for 3,4,5 vShift cases, thereby O-RU must pick the positions as interpreted in above table.

Pseudo code to determine the CRS belongs to ANT port-N

- Determine vShift = N_CELL_ID % 6
- Below pseudo code for Num Layer = 4

```
Switch(crsSymNum)
{
    Case 0 || case 7:
```

```
                    if( vShift  == 0 )
                    {
                        if (bitSet_0 && bitSet_6)
                        {
                            mapIQ of 0 and 6 to Ant0 //rest blank
                        }
                        if (bitSet_3 && bitSet_9)
                        {
                            mapIQ of 0 and 6 to Ant1 //rest blank
                        }
                    }
                Break;
    Case 1:
                    if( vShift  == 0 )
                    {
                      if (bitSet_0 && bitSet_6)
                      {
                         mapIQ of 0 and 6 to Ant2
                      }
                      if (bitSet_3 && bitSet_9)
                      {
                         mapIQ of 3 and 9 to Ant3
                      }
                    }
                    Break;
Case 4 || case 11:
                    if( vShift  == 0 )
                    {
                      if (bitSet_0 && bitSet_6)
                      {
                         mapIQ of 0 and 6 to Ant1
                      }
                      if (bitSet_3 && bitSet_9)
                      {
                         mapIQ of 0 and 6 to Ant0
                      }
                    }
                    Break;
Case 8:
                if( vShift  == 0 )
                {
                  if (bitSet_0 && bitSet_6)
                  {
                     mapIQ of 0 and 6 to Ant3
                  }
                  if (bitSet_3 && bitSet_9)
                  {
                     mapIQ of 3 and 9 to Ant2
                  }
                }
                Break;
}
```

Annex J Beamforming Methods Description:

Beams are RF energy directed in specific angular directions in space. Beamforming can generate energy lobes (maxima, or "peaks") &/or energy nulls (minima, or "valleys") in the spatial dimension. They can be formed using various methods in the analog domain, the digital domain, or a combination thereof. Beams and beamforming can further be used to re-utilize temporo-spectral (Time-Frequency) resources to achieve Spatial Multiplexing.

O-RAN has four distinct methods supported at the O-RU for beamforming.

1. **Predefined-Beam Beamforming.**

In this method, beam indices ("beamId" values) are conveyed from the O-DU to the O-RU to indicate which beam to apply to the DL or UL data. The beams are expected to be pre-defined within the O-RU and the method of beamforming is not specified. However, the O-DU still needs to know whether the beamforming is frequency-domain, time-domain or "hybrid" so the beamId values can be correctly applied e.g. to not apply different time-domain beamIds to subcarriers in the same OFDM symbol.

2. Weight-based Dynamic Beamforming.

In this method, beamforming weights are transmitted across the interface using C-Plane messages.

For frequency-domain beamforming the operation may be considered as follows:

Let $\mathrm{X} = [\mathrm{x}_0\ \mathrm{x}_1\ \ldots \mathrm{x}_P]$ represent the frequency domain IQ data in one data section, where P is the number of PRBs in the data section and x is a PRB, $x \in \mathbb{C}^{L*1}$, L is the number of streams.

Let $\mathrm{W} = [\mathrm{w}_0\ \mathrm{w}_1\ \ldots \mathrm{w}_K]$ represent the beamforming weights associated with the tx-array, where w is a beamforming weight, $\mathrm{w} \in \mathbb{C}^{K*L}$, K is the number of array elements.

The output after beamforming $\mathrm{Y} = [\mathrm{y}_0\ \mathrm{y}_1\ \ldots \mathrm{y}_P]$ is given by

$$\mathrm{y} = \mathrm{w}.\mathrm{x}$$

where y is a beamforming output for the PRB x , $\mathrm{y} \in \mathbb{C}^{K*1}$

The equation above refers to DL beamforming but the same principle is applied for UL as well.

For time-domain beamforming the operation is the similar:

Here, however, instead of being applied per data section the beamforming is applied for the entire OFDM symbol.

Let $\mathrm{X} = [\mathrm{x}_0\ \mathrm{x}_1\ \ldots \mathrm{x}_P]$ represent the frequency domain IQ data in one OFDM symbol, where P is the number of PRBs in the OFDM symbol and x is a PRB, $x \in \mathbb{C}^{L*1}$, L is the number of streams.

Let $\mathrm{W} = [\mathrm{w}_0\ \mathrm{w}_1\ \ldots \mathrm{w}_P]$ represent the beamforming weights associated with one tx-array, where w is a beamforming weight, $\mathrm{w} \in \mathbb{C}^{K*L}$, K is the number of array elements.

The output after beamforming $\mathrm{Y} = [\mathrm{y}_0\ \mathrm{y}_1\ \ldots \mathrm{y}_P]$ is given by

$$\mathrm{y} = \mathrm{w}.\mathrm{x}$$

where y is a beamforming output for the PRB x , $\mathrm{y} \in \mathbb{C}^{K*1}$

The equation above refers to DL beamforming but the same principle is applied for UL as well.

For hybrid beamforming the operation involves the multiplication by frequency-domain weights and time-domain weights.

There are K number of time-domain weights (the is the same as the number of array elements) and K' number of frequency-domain weights (K' is less than K, see chapter 10.5). The K' frequency-domain weights are applied for the PRBs in a data section, and the K time-domain weights are applied for all the PRBs in the entire OFDM symbol.

Let $\mathrm{X} = [\mathrm{x}_0\ \mathrm{x}_1\ \ldots \mathrm{x}_P]$ represent the frequency domain IQ data in one OFDM symbol, where P is the number of PRBs in the OFDM symbol and x is a PRB, $x \in \mathbb{C}^{L*1}$, L is the number of streams.

Let $\mathrm{W} = [\mathrm{w}_0\ \mathrm{w}_1\ \ldots \mathrm{w}_P]$ represent the beamforming weights in one OFDM symbol, where w is a time-domain beamforming weight, $\mathrm{w} \in \mathbb{C}^{K*L}$, K is the total number of array elements.

Let $\mathrm{W}' = [\mathrm{w}'_0\ \mathrm{w}'_1\ \ldots \mathrm{w}'_{P'},\ \mathrm{w}'_0\ \mathrm{w}'_1\ \ldots \mathrm{w}'_{P'}\ \ldots \mathrm{w}'_0\ \mathrm{w}'_1\ \ldots \mathrm{w}'_{P'}]$ represent the beamforming weights in one data section, where w' is a frequency-domain beamforming weight, $\mathrm{w}' \in \mathbb{C}^{K'*L}$.

Note that $K = K' * p'$ so the W' vector represents K / K' repetitions of the K' frequency-domain weights., The output after beamforming $Y = [y_0\ y_1\ ...\ y_P]$ is given by

$$y = (w * w').x$$

where y is a beamforming output for the PRB x , $y \in \mathbb{C}^{M*1}$

The equation above refers to DL beamforming but the same principle is applied for UL as well.

3. Attribute-Based Dynamic Beamforming.

In this method, beamforming attributes and/or their indices (if already known to the O-RU) are transmitted across the interface.

Whereas a beam *index* provides a pointer to a beamforming vector already known to the O-RU, and beamforming *weights* specify an important method for *how* to form the beam, beamforming *attributes* specify the *what*, an inherently compact characterization of the desired beam pattern itself, to be formed directionally in space.

These beamforming attributes include:

- bfAzPt: the azimuth beamforming pointing angle in degrees
- bfZePt: the zenith beamforming pointing angle in degrees
- bfAz3dd: the azimuth beamforming 3dB down beam width in degrees
- bfZe3dd: the zenith beamforming 3dB down beam width in degrees
- bfAzSl: the azimuth beamforming sidelobe suppression value in dB
- bfZeSl: the zenith beamforming sidelobe suppression value in dB

Multiple methods of forming the beam per the Beam Attributes are possible and are left as O-RU implementation choices. Some potential schemes are: Beamforming phased array weights (Analog or Digital), Holographic Beamforming, Butler Matrices, Lenses, and other known and emerging techniques. These schemes may also be hybridized with method #4 “Channel-Information-Based beamforming”.

4. Channel-Information-Based Beamforming.

In this method, beamforming weights are calculated at the O-RU based on the channel estimates that are transmitted across the interface.

Assuming K users who are jointly scheduled for MU-MIMO, a beamforming matrix $G = [g_1,...,g_K] \in \mathbb{C}^{MxK}$ is applied to the frequency domain IQ data for K users.

Let $H \triangleq [h_1 ..., h_K] \in \mathbb{C}^{KxM}$ be the channel estimates for the K users.

Multiple methods to calculate the beamforming weights are possible and are left as the O-RU implementation choices. Some potential schemes are:

Zero-forcing : $G = \frac{1}{\sqrt{\Psi}} (H^H\ H)^{-1} H^H$

Regularized zero-forcing/ MMSE: $G = \frac{1}{\sqrt{\Psi}} (H^H\ H + \xi I_M)^{-1} H^H$,

ξ is the regularization parameter and, the normalization parameter Ψ can be chosen to satisfy the total power constraint $\{GG^{\mathbf{H}}\} \leq M$ with equality

Note that the O-RAN specification allows configuration of regularization factor per user.

Annex K: Layers of Array Elements

Figure A-1 shows an example of a rectangular array with 12 rows and 4 columns of array elements. There are four data converters (not shown in the figure). Each data converter connects to all the 48 array elements (also known as the full-connection model in 3GPP). The array element contains 4 gain and phase control elements each connecting to one of the data converters. The gain and phase control element is used to apply time domain (TD) beamforming weights. This type of connection creates four overlapping arrays ("array-layers") by reusing the same array elements. On the transmit

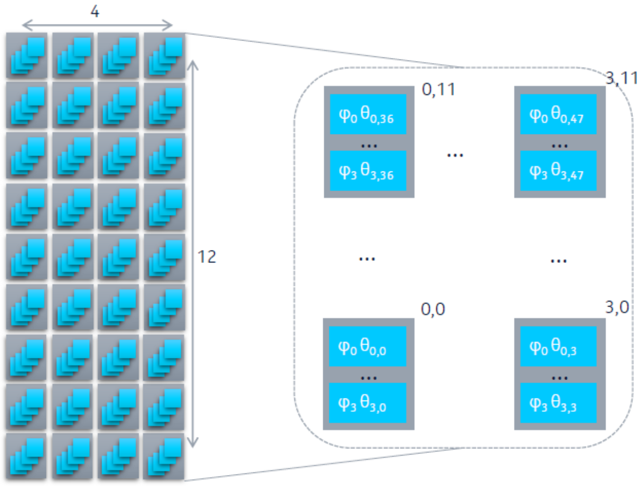


Figure K-1

side each DAC can correspond to one layer, which is then connected to one of the overlapping Tx arrays. Hence the structure of the tx-array so constructed can support 4 layers. The 48 elements corresponding to layer Q = 0…3 receive the same input from DAC Q and this input can be beamformed with a 48 element TD beamforming vector $(\theta_{q,0}, \theta_{q,47})$and a Frequency Domain (FD) beamforming weight φ_q corresponding to layer q. Each array element hence can receive 4 equivalent beamforming weights $\varphi_0\theta_{0,0} to \varphi_3\theta_{3,0}$ corresponding to 4 layers.

K.1 Use Case A

Use case A is to send eAxC via 48 elements. In this case, the eAxC is beamformed in frequency domain (FD) by $\varphi_0$and 48 complex weights corresponding to time domain (TD) using$(\theta_{q,0}, \theta_{q,47})$ see **Figure K-2** . In this case, the O-RU has 4 simple tx-arrays with 48 elements each.

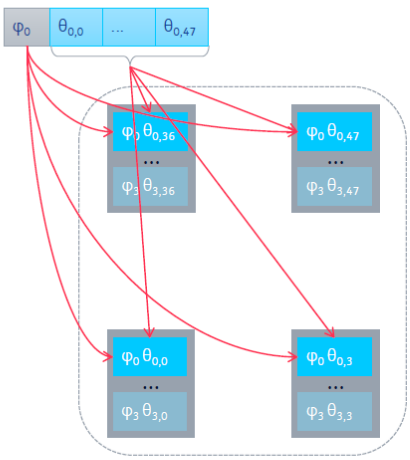


Figure K-2

K.2 Use Case B

In this use case, one eAxC signal is sent over all the 4x48 elements. This can be accomplished by using 4 digital weights in FD and 4x48 TD weights (Figure K-3). In this case, O-RU has one tx-array with 4x48 elements.

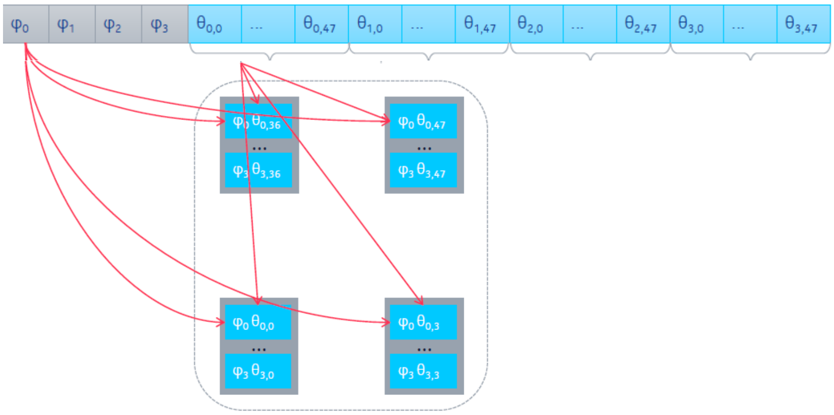


Figure K-3

One should note in the above cases that the 48 array elements each have one power amplifier which is shared across the layers Q = 0…3.

Annex ZZZ : O-RAN Adopter License Agreement

BY DOWNLOADING, USING OR OTHERWISE ACCESSING ANY O-RAN SPECIFICATION, ADOPTER AGREES TO THE TERMS OF THIS AGREEMENT.

This O-RAN Adopter License Agreement (the “Agreement”) is made by and between the O-RAN Alliance and the entity that downloads, uses or otherwise accesses any O-RAN Specification, including its Affiliates (the “Adopter”).

This is a license agreement for entities who wish to adopt any O-RAN Specification.

Section 1: DEFINITIONS

1.1 “Affiliate” means an entity that directly or indirectly controls, is controlled by, or is under common control with another entity, so long as such control exists. For the purpose of this Section, “Control” means beneficial ownership of fifty (50%) percent or more of the voting stock or equity in an entity.

1.2 “Compliant Implementation” means any system, device, method or operation (whether implemented in hardware, software or combinations thereof) that fully conforms to a Final Specification.

1.3 “Adopter(s)” means all entities, who are not Members, Contributors or Academic Contributors, including their Affiliates, who wish to download, use or otherwise access O-RAN Specifications.

1.4 “Minor Update” means an update or revision to an O-RAN Specification published by O-RAN Alliance that does not add any significant new features or functionality and remains interoperable with the prior version of an O-RAN Specification. The term “O-RAN Specifications” includes Minor Updates.

1.5 “Necessary Claims” means those claims of all present and future patents and patent applications, other than design patents and design registrations, throughout the world, which (i) are owned or otherwise licensable by a Member, Contributor or Academic Contributor during the term of its Member, Contributor or Academic Contributorship; (ii) such Member, Contributor or Academic Contributor has the right to grant a license without the payment of consideration to a third party; and (iii) are necessarily infringed by a Compliant Implementation (without considering any Contributions not included in the Final Specification). A claim is necessarily infringed only when it is not possible on technical (but not commercial) grounds, taking into account normal technical practice and the state of the art generally available at the date any Final Specification was published by the O-RAN Alliance or the date the patent claim first came into existence, whichever last occurred, to make, sell, lease, otherwise dispose of, repair, use or operate a Compliant Implementation without infringing that claim. For the avoidance of doubt in exceptional cases where a Final Specification can only be implemented by technical solutions, all of which infringe patent claims, all such patent claims shall be considered Necessary Claims.

1.6 “Defensive Suspension” means for the purposes of any license grant pursuant to Section 3, Member, Contributor, Academic Contributor, Adopter, or any of their Affiliates, may have the discretion to include in their license a term allowing the licensor to suspend the license against a licensee who brings a patent infringement suit against the licensing Member, Contributor, Academic Contributor, Adopter, or any of their Affiliates.

Section 2: COPYRIGHT LICENSE

2.1 Subject to the terms and conditions of this Agreement, O-RAN Alliance hereby grants to Adopter a nonexclusive, nontransferable, irrevocable, non-sublicensable, worldwide copyright license to obtain, use and modify O-RAN Specifications, but not to further distribute such O-RAN Specification in any modified or unmodified way, solely in furtherance of implementations of an ORAN

Specification.

2.2 Adopter shall not use O-RAN Specifications except as expressly set forth in this Agreement or in a separate written agreement with O-RAN Alliance.

Section 3: FRAND LICENSE

3.1 Members, Contributors and Academic Contributors and their Affiliates are prepared to grant based on a separate Patent License Agreement to each Adopter under Fair Reasonable And Non- Discriminatory (FRAND) terms and conditions with or without compensation (royalties) a nonexclusive, non-transferable, irrevocable (but subject to Defensive Suspension), non-sublicensable, worldwide patent license under their Necessary Claims to make, have made, use, import, offer to sell, lease, sell and otherwise distribute Compliant Implementations; provided, however, that such license shall not extend: (a) to any part or function of a product in which a Compliant Implementation is incorporated that is not itself part of the Compliant Implementation; or (b) to any Adopter if that Adopter is not making a reciprocal grant to Members, Contributors and Academic Contributors, as set forth in Section 3.3. For the avoidance of doubt, the foregoing licensing commitment includes the distribution by the Adopter's distributors and the use by the Adopter's customers of such licensed Compliant Implementations.

3.2 Notwithstanding the above, if any Member, Contributor or Academic Contributor, Adopter or their Affiliates has reserved the right to charge a FRAND royalty or other fee for its license of Necessary Claims to Adopter, then Adopter is entitled to charge a FRAND royalty or other fee to such Member, Contributor or Academic Contributor, Adopter and its Affiliates for its license of Necessary Claims to its licensees.

3.3 Adopter, on behalf of itself and its Affiliates, shall be prepared to grant based on a separate Patent License Agreement to each Members, Contributors, Academic Contributors, Adopters and their Affiliates under Fair Reasonable And Non-Discriminatory (FRAND) terms and conditions with or without compensation (royalties) a nonexclusive, non-transferable, irrevocable (but subject to Defensive Suspension), non-sublicensable, worldwide patent license under their Necessary Claims to make, have made, use, import, offer to sell, lease, sell and otherwise distribute Compliant Implementations; provided, however, that such license will not extend: (a) to any part or function of a product in which a Compliant Implementation is incorporated that is not itself part of the Compliant Implementation; or (b) to any Members, Contributors, Academic Contributors, Adopters and their Affiliates that is not making a reciprocal grant to Adopter, as set forth in Section 3.1. For the avoidance of doubt, the foregoing licensing commitment includes the distribution by the Members', Contributors', Academic Contributors', Adopters' and their Affiliates' distributors and the use by the Members', Contributors', Academic Contributors', Adopters' and their Affiliates' customers of such licensed Compliant Implementations.

Section 4: TERM AND TERMINATION

4.1 This Agreement shall remain in force, unless early terminated according to this Section 4.

4.2 O-RAN Alliance on behalf of its Members, Contributors and Academic Contributors may terminate this Agreement if Adopter materially breaches this Agreement and does not cure or is not capable of curing such breach within thirty (30) days after being given notice specifying the breach.

4.3 Sections 1, 3, 5 - 11 of this Agreement shall survive any termination of this Agreement. Under surviving Section 3, after termination of this Agreement, Adopter will continue to grant licenses (a) to entities who become Adopters after the date of termination; and (b) for future versions of ORAN Specifications that are backwards compatible with the version that was current as of the date of termination.

Section 5: CONFIDENTIALITY

Adopter will use the same care and discretion to avoid disclosure, publication, and dissemination of O-RAN Specifications to third parties, as Adopter employs with its own confidential information, but no less than reasonable care. Any disclosure by Adopter to its Affiliates, contractors and consultants should be subject to an obligation of confidentiality at least as restrictive as those contained in this Section. The foregoing obligation shall not apply to any information which is: (1) rightfully known by Adopter without any limitation on use or disclosure prior to disclosure; (2) publicly available through no fault of Adopter; (3) rightfully received without a duty of confidentiality; (4) disclosed by O-RAN Alliance or a Member, Contributor or Academic Contributor to a third party without a duty of confidentiality on such third party; (5) independently developed by Adopter; (6) disclosed pursuant to the order of a court or other authorized governmental body, or as required by law, provided that Adopter provides reasonable prior written notice to O-RAN Alliance, and cooperates with O-RAN Alliance and/or the applicable Member, Contributor or Academic Contributor to have the opportunity to oppose any such order; or (7) disclosed by Adopter with O-RAN Alliance's prior written approval.

Section 6: INDEMNIFICATION

Adopter shall indemnify, defend, and hold harmless the O-RAN Alliance, its Members, Contributors or Academic Contributors, and their employees, and agents and their respective successors, heirs and assigns (the "Indemnitees"), against any liability, damage, loss, or expense (including reasonable attorneys' fees and expenses) incurred by or imposed upon any of the Indemnitees in connection with any claims, suits, investigations, actions, demands or judgments arising out of Adopter's use of the licensed O-RAN Specifications or Adopter's commercialization of products that comply with O-RAN Specifications.

Section 7: LIMITATIONS ON LIABILITY; NO WARRANTY

EXCEPT FOR BREACH OF CONFIDENTIALITY, ADOPTER'S BREACH OF SECTION 3, AND ADOPTER'S INDEMNIFICATION OBLIGATIONS, IN NO EVENT SHALL ANY PARTY BE LIABLE TO ANY OTHER PARTY OR THIRD PARTY FOR ANY INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES RESULTING FROM ITS PERFORMANCE OR NON-PERFORMANCE UNDER THIS AGREEMENT, IN EACH CASE WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, AND WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. O-RAN SPECIFICATIONS ARE PROVIDED "AS IS" WITH NO WARRANTIES OR CONDITIONS WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE. THE O-RAN ALLIANCE AND THE MEMBERS, CONTRIBUTORS OR ACADEMIC CONTRIBUTORS EXPRESSLY DISCLAIM ANY WARRANTY OR CONDITION OF MERCHANTABILITY, SECURITY, SATISFACTORY QUALITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, ERROR-FREE OPERATION, OR ANY WARRANTY OR CONDITION FOR O-RAN SPECIFICATIONS.

Section 8: ASSIGNMENT

Adopter may not assign the Agreement or any of its rights or obligations under this Agreement or make any grants or other sublicenses to this Agreement, except as expressly authorized hereunder, without having first received the prior, written consent of the O-RAN Alliance, which consent may be withheld in O-RAN Alliance's sole discretion. O-RAN Alliance may freely assign this Agreement.

Section 9: THIRD-PARTY BENEFICIARY RIGHTS

Adopter acknowledges and agrees that Members, Contributors and Academic Contributors (including future Members, Contributors and Academic Contributors) are entitled to rights as a third-party beneficiary under this Agreement, including as licensees under Section 3.

Section 10: BINDING ON AFFILIATES

Execution of this Agreement by Adopter in its capacity as a legal entity or association constitutes that legal entity's or association's agreement that its Affiliates are likewise bound to the obligations that are applicable to Adopter hereunder and are also entitled to the benefits of the rights of Adopter hereunder.

Section 11: GENERAL

This Agreement is governed by the laws of Germany without regard to its conflict or choice of law provisions.

This Agreement constitutes the entire agreement between the parties as to its express subject matter and expressly supersedes and replaces any prior or contemporaneous agreements between the parties, whether written or oral, relating to the subject matter of this Agreement.

Adopter, on behalf of itself and its Affiliates, agrees to comply at all times with all applicable laws, rules and regulations with respect to its and its Affiliates' performance under this Agreement, including without limitation, export control and antitrust laws. Without limiting the generality of the foregoing, Adopter acknowledges that this Agreement prohibits any communication that would violate the antitrust laws.

By execution hereof, no form of any partnership, joint venture or other special relationship is created between Adopter, or O-RAN Alliance or its Members, Contributors or Academic Contributors. Except as expressly set forth in this Agreement, no party is authorized to make any commitment on behalf of Adopter, or O-RAN Alliance or its Members, Contributors or Academic Contributors.

In the event that any provision of this Agreement conflicts with governing law or if any provision is held to be null, void or otherwise ineffective or invalid by a court of competent jurisdiction, (i) such provisions will be deemed stricken from the contract, and (ii) the remaining terms, provisions, covenants and restrictions of this Agreement will remain in full force and effect.

Any failure by a party or third party beneficiary to insist upon or enforce performance by another party of any of the provisions of this Agreement or to exercise any rights or remedies under this Agreement or otherwise by law shall not be construed as a waiver or relinquishment to any extent of the other parties' or third party beneficiary's right to assert or rely upon any such provision, right or remedy in that or any other instance; rather the same shall be and remain in full force and effect.

