SMALL CELL FORUM

RELEASE 9.0

scf.io

DOCUMENT

082.09.05

FAPI and nFAPI specifications

May 2017

www.scf.io/

SMALL CELL FORUM

Solving the HetNet puzzle

www.smallcellforum.org

SMALL CELL FORUM

RELEASE 9.0

scf.io

Small Cell Forum accelerates small cell adoption to drive the wide-scale adoption of small cells and accelerate the delivery of integrated HetNets.

We are not a standards organization but partner with organizations that inform and determine standards development. We are a carrier-led organization. This means our operator members establish requirements that drive the activities and outputs of our technical groups.

We have driven the standardization of key elements of small cell technology including Iuh, FAPI, nFAPI, SON, services APIs, TR-069 evolution and the enhancement of the X2 interface.

Today our members are driving solutions that include small cell/Wi-Fi integration, SON evolution and automation, virtualization of the small cell layer, driving mass adoption via multi-operator neutral host, ensuring a common approach to service APIs to drive commercialization and the integration of small cells into 5G standards evolution.

The Small Cell Forum Release Program has now established business cases and market drivers for all the main use cases, clarifying market needs and addressing barriers to deployment for residential, enterprise, rural & remote, and urban small cells.

The theme of Release 9 is Commercializing Hyperdense HetNets, which looks at practical solutions to help operators transform their network capacity and performance, and which are deployable and cost-effective right now.

The Small Cell Forum Release Program website can be found here: www.scf.io

If you would like more information about Small Cell Forum or would like to be included on our mailing list, please contact:

Email info@smallcellforum.org

Post Small Cell Forum, PO Box 23, GL11 5WA UK

Member Services memberservices@smallcellforum.org

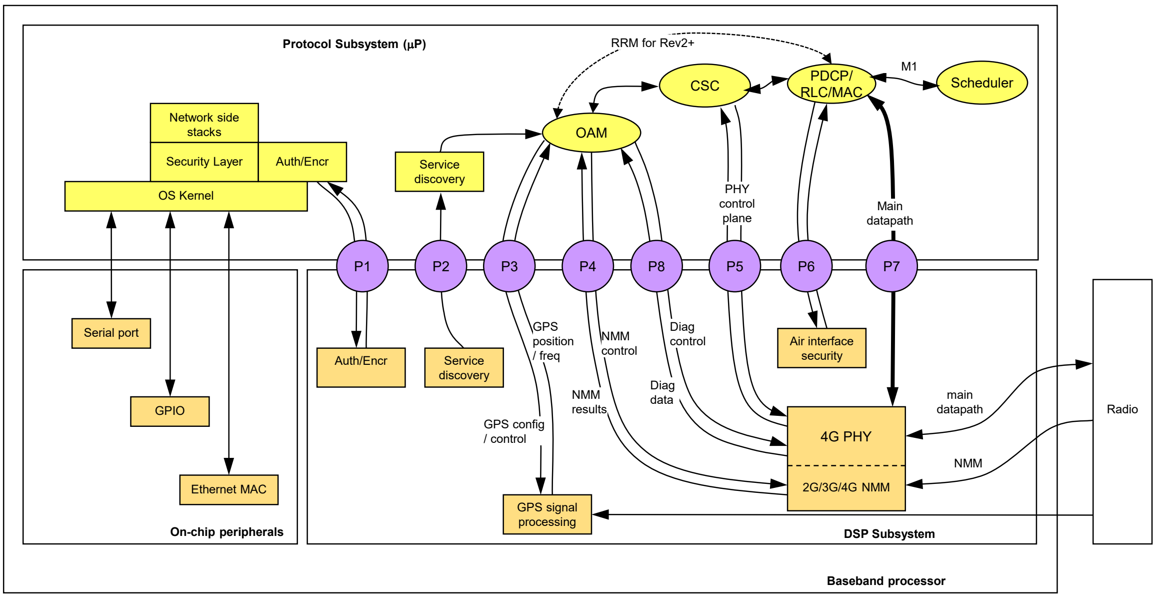
Scope

The functional application platform interface (FAPI) is an initiative within the small cell industry to encourage competition and innovation among suppliers of platform hardware, platform software and application software by providing a common API around which suppliers of each component can compete. In doing this, we imitate a long and distinguished engineering tradition of providing an 'interchangeability of parts' to ensure that the systems vendors can take advantage of the latest innovations in silicon and software with minimum barriers to entry, and the least amount of custom re-engineering.

The FAPI is defined via a reference architecture shown below, which is generic to 3G or 4G small cells. Several APIs are defined, as follows:

- P1 – the security coprocessor interface
- P2 – the service discovery interface
- P3 – the GPS interface
- P4 – the network listen results interface
- P5 – the PHY mode control interface
- P6 – the ciphering coprocessor interface
- P7 – the main data path interface
- M1 – the scheduler interface

This document defines both the P5 and P7 interfaces for 4G cells.



CSC= Cell and Subframe Control

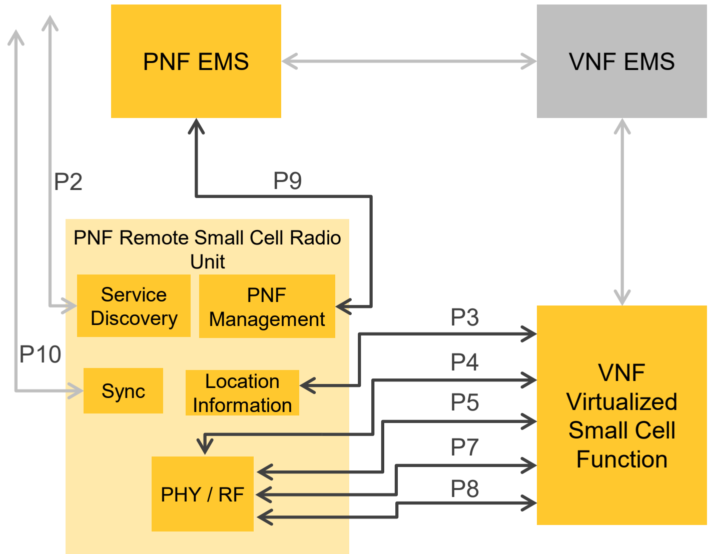
The network functional application platform interface (nFAPI) is an initiative following the virtualization study undertaken by the Small Cell Forum [1] [2]. This virtualization study examined different functional splits where an upper part of the eNB is implemented in a virtual network function (VNF) and a lower part of the eNB resides in a physical network function (PNF).

The MAC-PHY interface (L1 API), defined in FAPI, was identified in the virtualization study as a suitable split point. This has motivated the Forum to define a transportable interface for the split small cell, to establish a scalable ecosystem with a converged approach to virtualization across multiple suppliers.

The nFAPI is defined via a reference architecture shown below, which is specific to 4G small cells. Several APIs are defined, as follows:

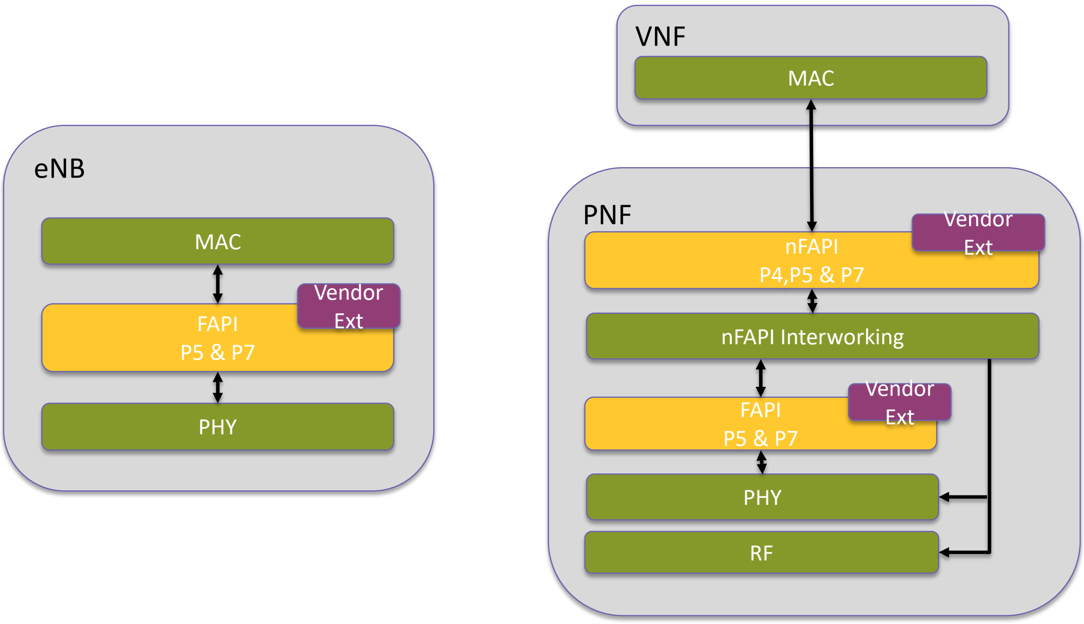
- P2 – the service discovery interface
- P3 – the positioning interface
- P4 – the network monitor mode interface
- P5 – the PHY mode control interface
- P7 – the main data path interface
- P8 – the PHY diagnostics interface
- P9 – the OAM interface
- P10 – the synchronization, frequency, phase and time

This document defines the P4, P5 and P7 interfaces.



Executive Summary

This document provides the specification for two application platform interfaces (APIs) between the MAC and PHY protocol layers within the small cell ecosystem. The first functional application platform interface (FAPI) is an internal interface within a small cell, and the second network functional application platform interface (nFAPI) is an extension where the small cell functionality now resides partially in a virtual network function (VNF) and partially in a physical network function (PNF). The relationship between FAPI and nFAPI is shown below.



The FAPI and nFAPI specifications define a control interface (P5) and a data path interface (P7). It is expected that readers are interested in implementing nFAPI, or FAPI, or both, and to aid implementors the document is structured as follows:

nFAPI (P5 & P7) – This section details procedures to configure and control the PNF entity, and to configure and control RF and PHY(s) within the PNF. It also defines the nFAPI specific messages and structures to implement these procedures.

FAPI (P5 & P7) – This section details procedures to configure and control a PHY entity. In addition, it defines the FAPI specific messages and structures to implement these procedures.

Common Messages (P7) – This section defines messages and structures which are common to both FAPI and nFAPI. The relationship between key FAPI and nFAPI procedures are captured in the Signalling Appendix.

Network Monitor Mode (P4) - In addition to the P5 and P7 interfaces an important function for the PNF is network monitor mode (NMM), which may be used to scan neighbouring LTE, UTRAN and GERAN cells. This section defines the procedures, messages and structures to implement NMM within a PNF.

Contents

Tables

SMALL CELL FORUM
Solving the HetNet puzzle

Figures

1. Introduction

This document describes the FAPI and nFAPI interfaces. The two interfaces are presented in a single document as FAPI can be viewed as a subset of nFAPI.

The Functional Application Platform Interface (FAPI) resides between LTE L2/L3 software and the L1 PHY. Specifically, this FAPI defines both P5 and P7 of the Small Cell Forum LTE FAPI, shown in Figure 1-1.

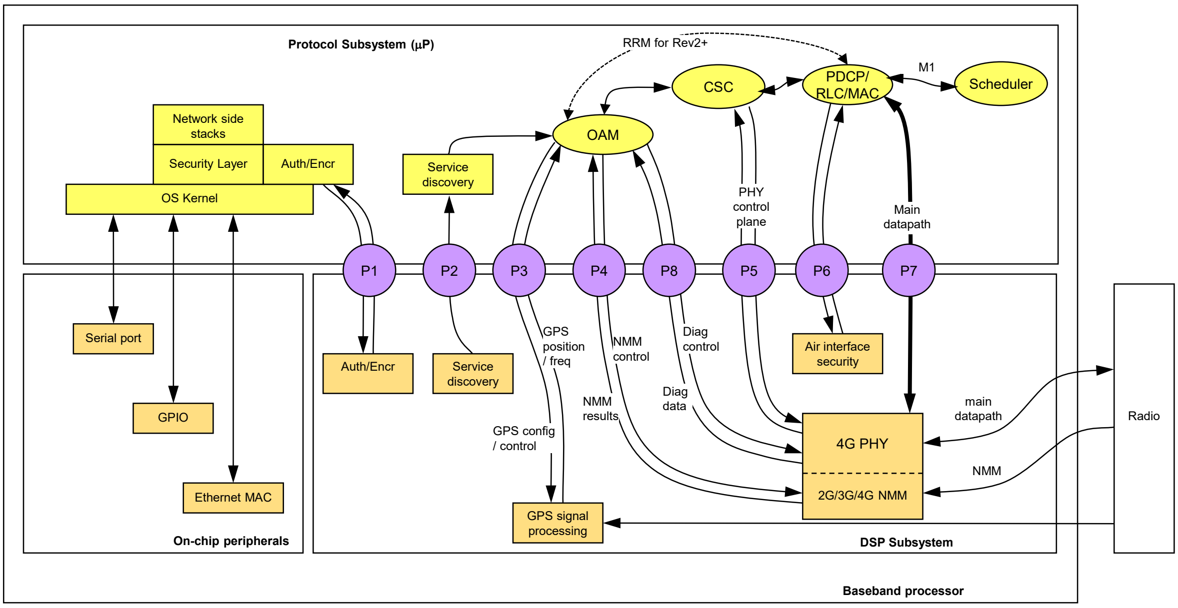


Figure 1-1 FAPI Interface

Network FAPI (nFAPI) resides between a Virtual Network Function (VNF) including LTE L2/L3 software and a separate Physical Network Function (PNF) including the L1 PHY. Specifically, this nFAPI defines the P4, P5 and P7 of the Small Cell forum LTE nFAPI, shown in Figure 1-2. While the remaining interfaces of nFAPI may be required for operation, they are out of scope here, but are described for reference in an appendix below.

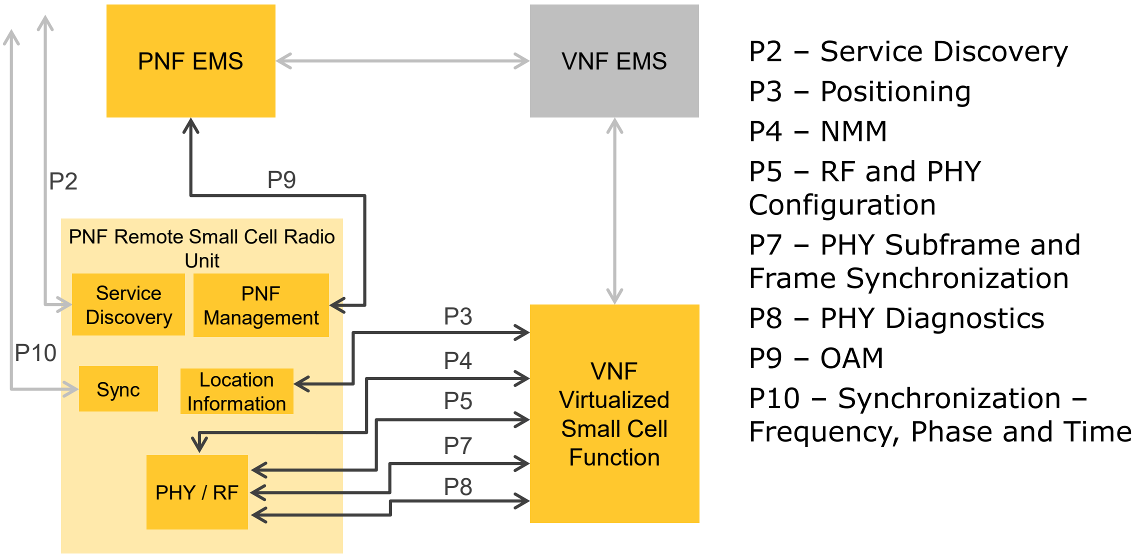


Figure 1-2 nFAPI Interfaces

Importantly, compared with FAPI that assumes a co-location of L2/L3 software and L1 PHY functionality, e.g. using a shared memory system to support the interface implementation, nFAPI has been defined assuming that a packet switched IP network is used to support communication between VNF and PNF components.

This document is divided into four sections. The first section provides a description of nFAPI, its typical procedures and messages specific to nFAPI.

The next section provides a description of FAPI, its typical procedures and messages specific to FAPI.

The next section contains messages which are common to both nFAPI and FAPI.

The final section provides a description of the LTE NMM P4 interfaces, its typical procedures and message definitions.

1.1 LTE

LTE is standardized by 3GPP (http://www.3gpp.org) and designed as an evolution to the current WCDMA wireless network. A critical requirement of LTE is the capability of supporting high data rates (up to 1Gbps), and many aspects of the LTE network have been designed specifically to support high data rates and low latency.

Figure 1-3 shows the architecture of an LTE network. It consists of only two elements; the Evolved Packet Core (EPC) and the E-UTRAN Node B (eNB). The LTE nFAPI and FAPI APIs reside within the eNB element. The two standardized interfaces in an LTE network are called S1 and X2. The L1 is not involved in either of these interfaces, and both are out of scope for this document.

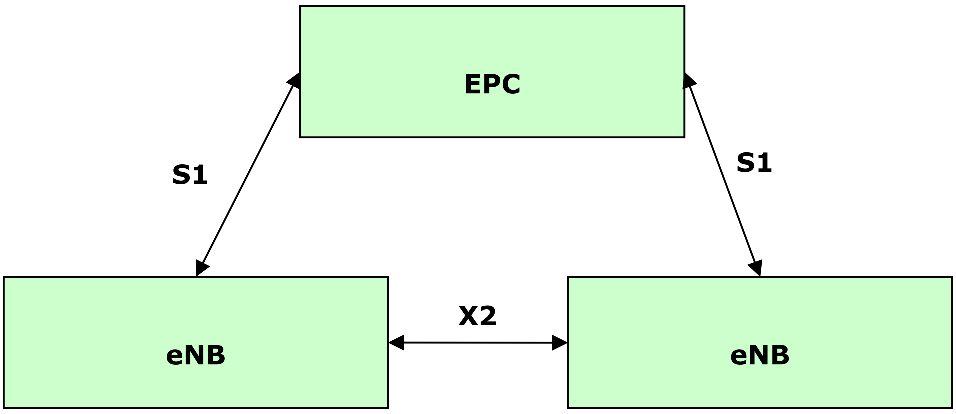


Figure 1-3 LTE architecture

For FAPI the eNB is a single element resulting in FAPI existing as an internal interface within the eNB, as shown in Figure 1-4.

For nFAPI the eNB functionality is split between a VNF and PNF, with nFAPI as an external interface between these two components, as shown in Figure 1-4.

Figure 1-4 also highlights the expected relationship between nFAPI and FAPI where nFAPI exists between the VNF and PNF, with FAPI internal to the PNF.

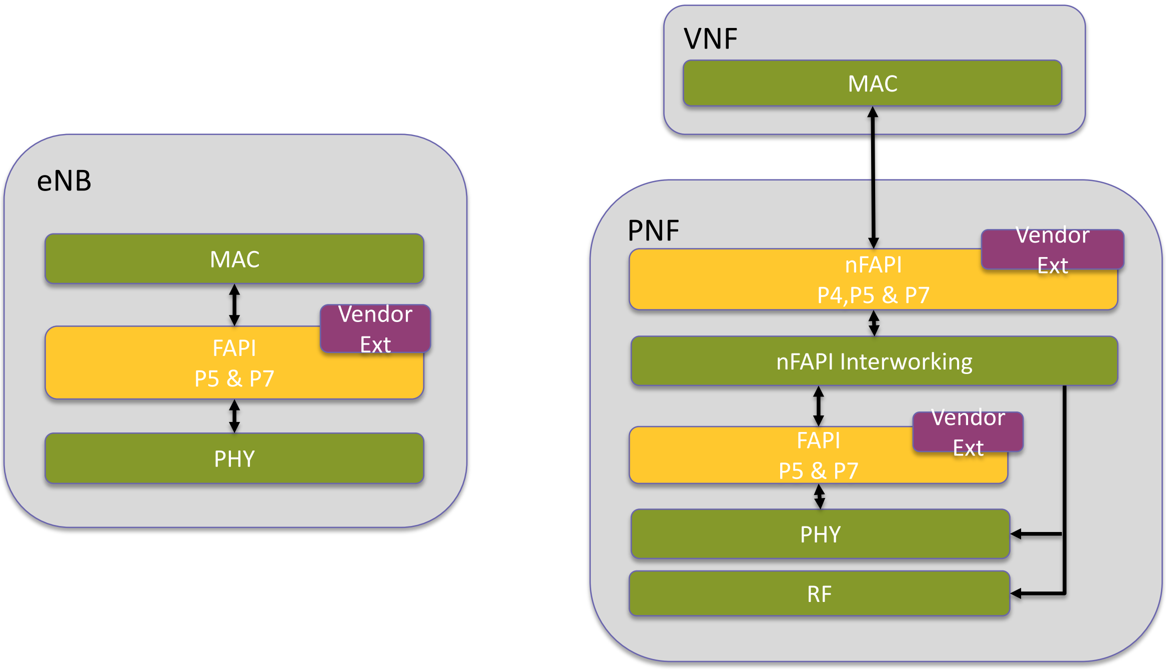


Figure 1-4 FAPI vs nFAPI architecture

2. nFAPI (P5 & P7)

2.1 Introduction

The nFAPI, defined within this document, resides between the virtual network function (VNF) and physical network function (PNF).

Within the context of nFAPI, the PNF device is the physical radio unit, containing a number of PHY instances with associated RF chains within the PNF device. While this document uses this context, the physical representation of the PNF device is not prescribed. The VNF is considered to be the nFAPI peer element controlling and operating the PNF device. The structure and distribution of the VNF is considered to be the eNB functionality to operate the PNF device and the PHY(s), but is not prescribed.

The scope of this section is the P5 and P7 interfaces of nFAPI.

Figure 2-1 shows the functional decomposition of small cell protocol elements and their relationship between the VNF and PNF, and nFAPI and FAPI. The mapping of functional elements and PHY instances is not prescribed, and shown here as an example for context.

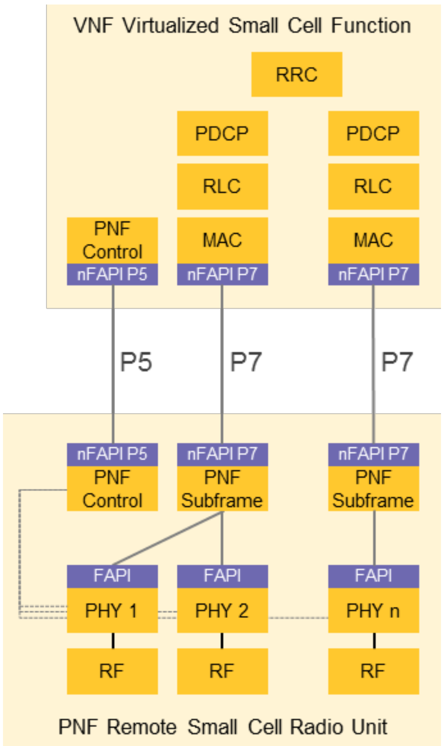


Figure 2-1 nFAPI interactions and context with FAPI

The PNF device is configurable over the P9 and the P5 interfaces to varying levels. There is overlap in configuration capability between P9 and P5 to permit P9 out-of-band configuration and option constraining, with the VNF over P5 further configuring the PNF device for operation. The details of the P9 interface are out of scope of this document.

2.2 nFAPI Procedures

The nFAPI P5 interface configures the PNF device at the PNF device level and at the static PHY level. The nFAPI P7 interface operates the PHY for the subframe procedures.

2.2.1 PNF Configuration Procedures

The PNF device is initially configured through the P9 OAM interface for foundation operation and connectivity with the VNF: this procedure is outside the scope of this specification. This permits the transport layer connection between the PNF and VNF, whereby further configuration of the PNF over the nFAPI interface occurs.

The PNF PARAM and PNF CONFIG exchanges share the PNF capability with the VNF and permits the VNF to configure the PNF within the scope of the PNF's capability. The VNF can then instruct the PNF to start, enabling the PHY operations to occur or stop, disabling the PHY operations from occurring.

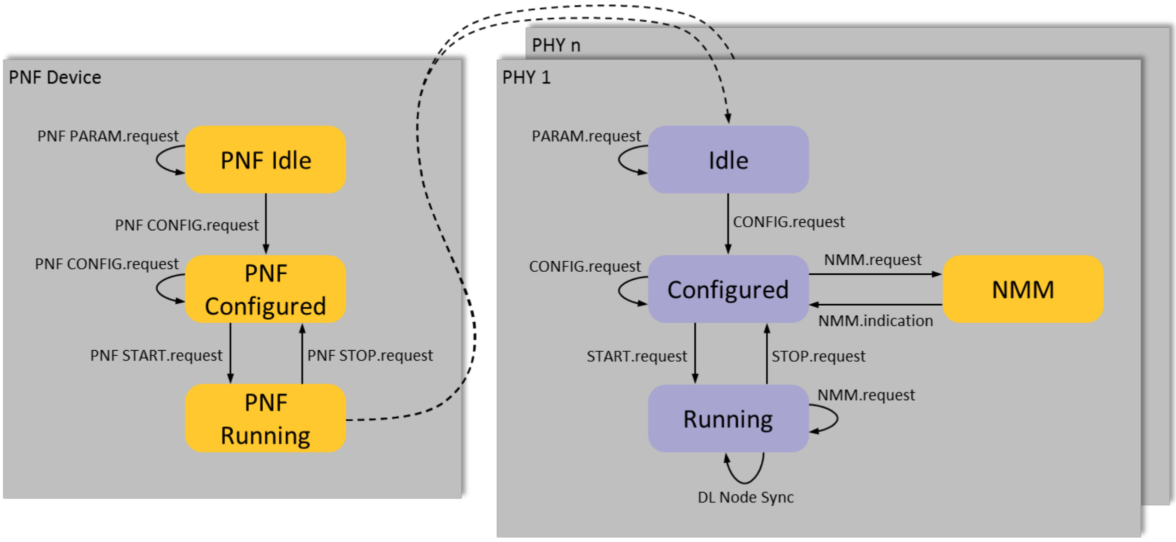


Figure 2-2 Relationship between PNF and PHY state machines

2.2.1.1 Initialization

The initialization procedure moves the PNF from the PNF IDLE state to the PNF RUNNING state, via the PNF CONFIGURED state. An overview of this procedure is given in Figure 2-3, the different stages are:

- The PNF PARAM message exchange procedure
- The PNF CONFIG message exchange procedure
- The PNF START message exchange procedure

The PNF initialization procedure is completed when the PNF sends the VNF a `PNF_START.response` message.

The remainder of this section describes the PNF_PARAM, PNF_CONFIG and PNF_START message exchange procedures.

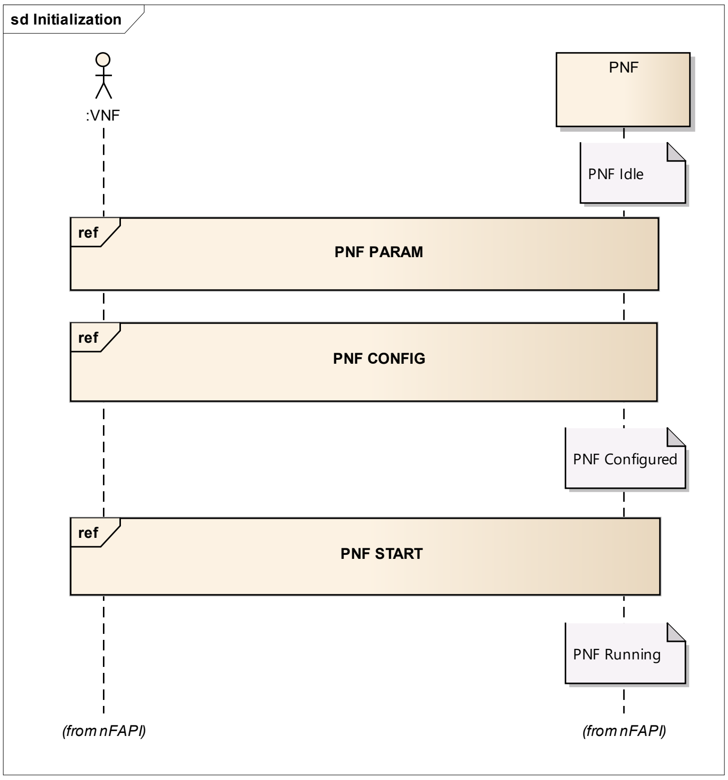


Figure 2-3 PNF Initialization procedure

The PNF_PARAM message exchange procedure is shown in Figure 2-4. Its purpose is to allow the VNF software to collect information about the PNF capability.

From Figure 2-4 it can be seen that the PNF_PARAM message exchange procedure is initiated by the VNF sending a `PNF_PARAM.request` message to the PNF. If the PNF is operating correctly it will return a `PNF_PARAM.response` message. If no response is received, the action taken by the VNF is anticipated to be to trigger a P9 management procedure, and is out of scope of this document.

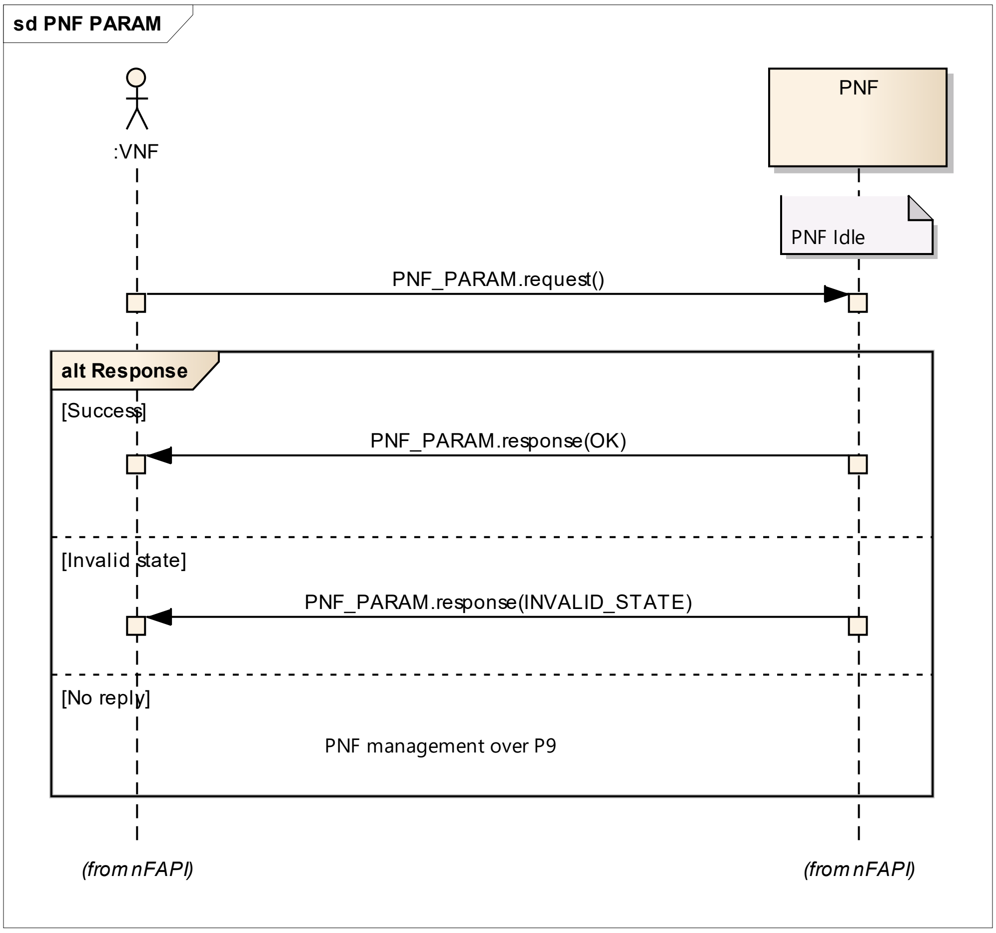


Figure 2-4 PNF_PARAM message exchange

The PNF_CONFIG message exchange procedure is shown in Figure 2-5. Its purpose is to allow the VNF to configure the PNF. The `PNF_CONFIG.request` message can be sent when the PNF is in the PNF IDLE state or the PNF CONFIGURED state. If the PNF is in the PNF IDLE state, this procedure will move the PNF into the PNF CONFIGURED state. If the PNF is in the PNF CONFIGURED state, the PNF will be re-configured with the supplied configuration and will remain in the PNF CONFIGURED state. The PNF will respond with the `PNF_CONFIG.response` message indicating success or rejection of the supplied configuration. A rejection will result in the PNF retaining its previous configuration and state. If no response is received, the action taken by the VNF is anticipated to be to trigger a P9 management procedure, and is out of scope of this document.

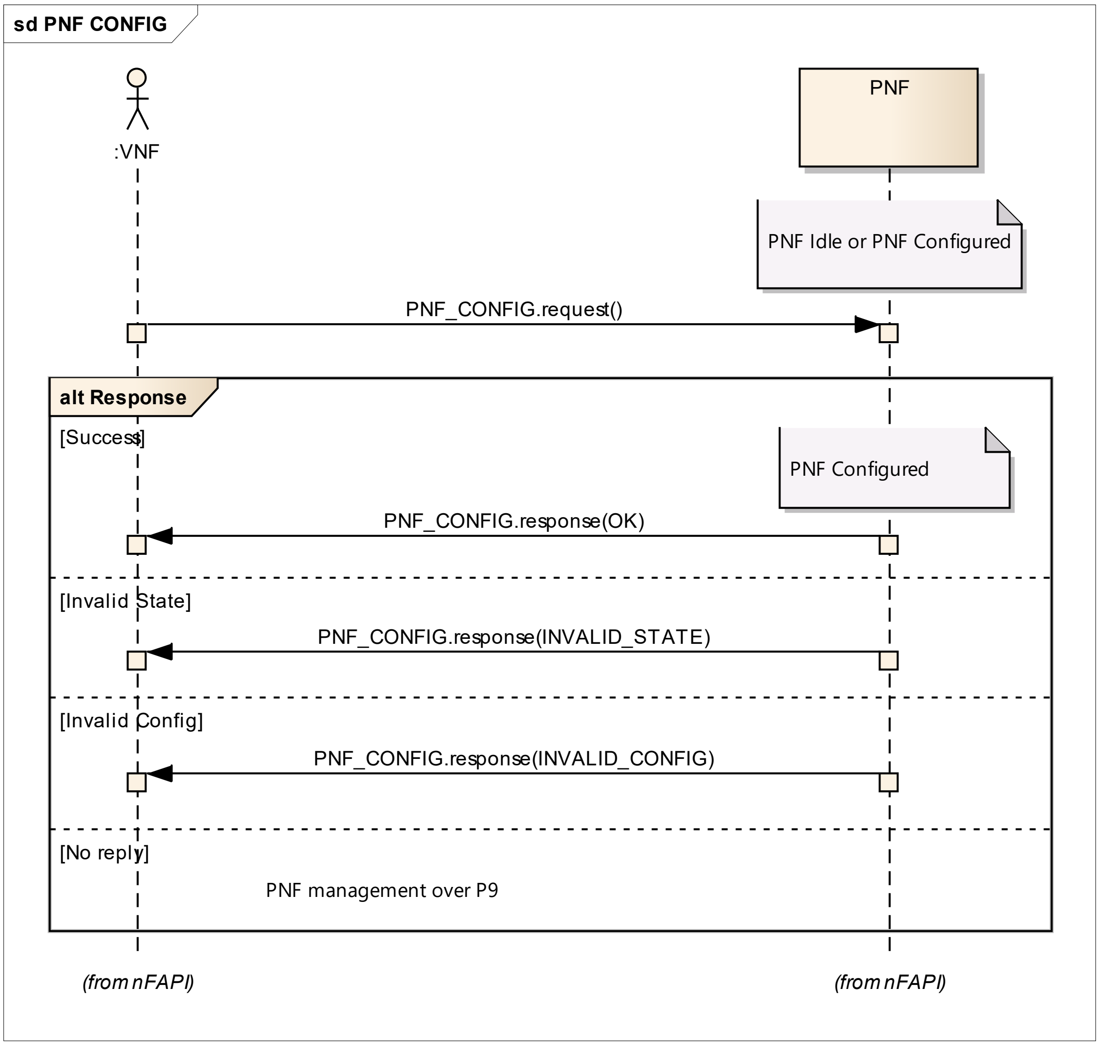


Figure 2-5 PNF_CONFIG message exchange

The PNF_START message exchange procedure is shown in Figure 2-6. Its purpose is to instruct a configured PNF to create the PHY instances. The VNF initiates this procedure by sending a `PNF_START.request` message to the PNF. If the PNF is in the CONFIGURED state, it will create the PHY instances, move to the PNF RUNNING state and respond with a `PNF_START.response` message.

If the PNF receives a `PNF_START.request` message in either the IDLE or RUNNING state it will return a `PNF_START.response` message including an INVALID_STATE error.

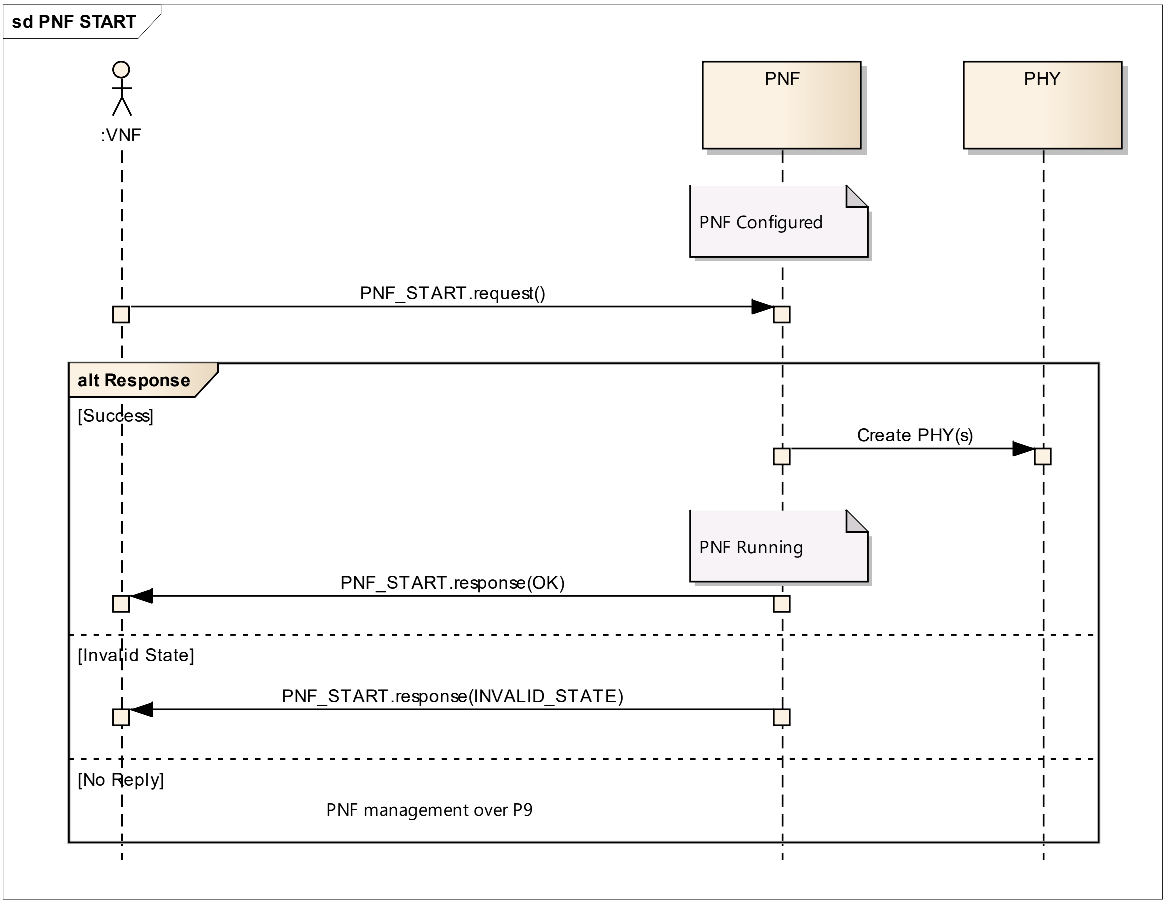


Figure 2-6 PNF_START message exchange

2.2.1.2 Reset

The Reset procedure is used to move the PNF from the PNF RUNNING state to the PNF CONFIGURED state. This also terminates and destroys the PHY instances. The termination procedure is shown in Figure 2-7 and initiated by the VNF sending a `PNF_STOP.request` message.

If the PNF_STOP.request message is received by the PNF while operating in the PNF RUNNING state, it will stop all PHY operations irrespective of the PHY states and return to the PNF_CONFIGURED state. When the PNF has completed its stop procedure a `PNF_STOP.response` message is sent to the VNF.

If the `PNF_STOP.request` message was received by the PNF while not in the PNF RUNNING state, it will return a `PNF_STOP.response` message including an INVALID_STATE error.

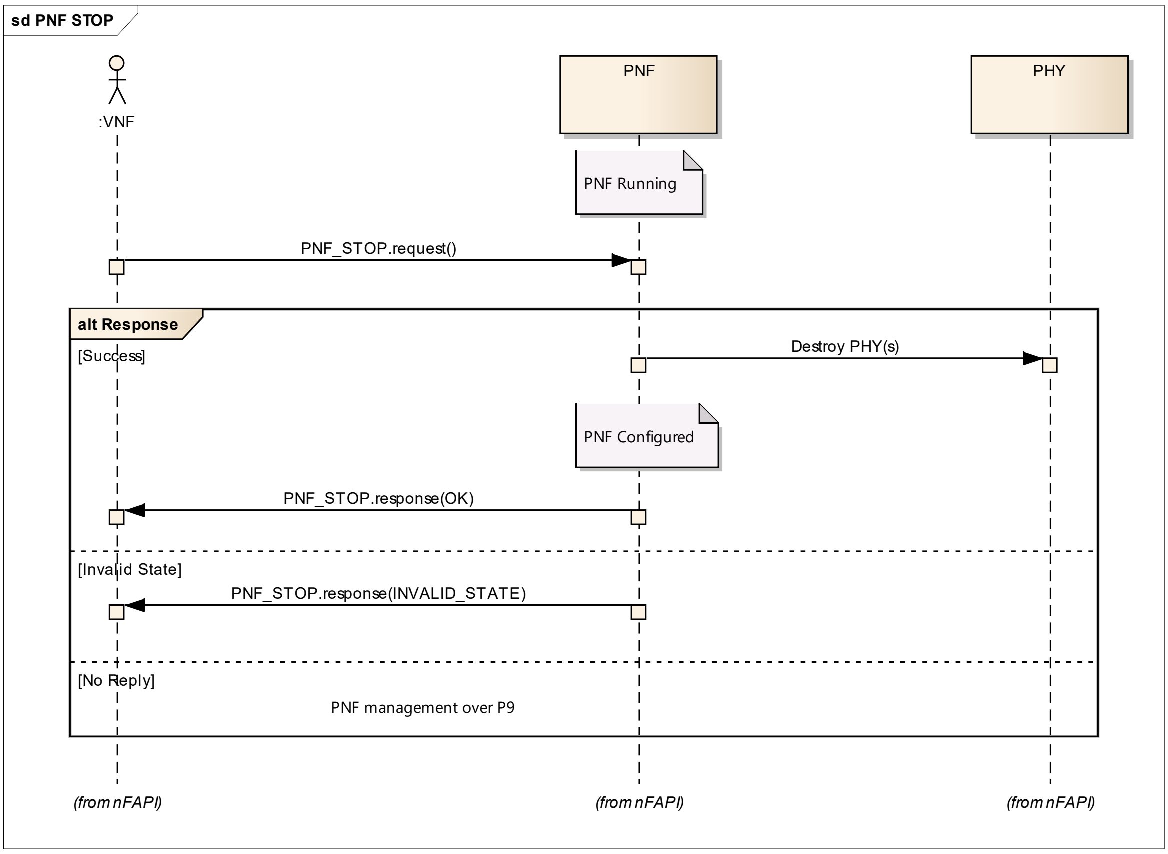


Figure 2-7 PNF_STOP message exchange

2.2.1.3 Restart

The Restart procedure is used to move the PNF from the PNF RUNNING state back to the PNF CONFIGURED state and then to return to the PNF RUNNING state. This procedure is defined as a sequence of the Stop (Figure 2-7) and the Start (Figure 2-6) procedures.

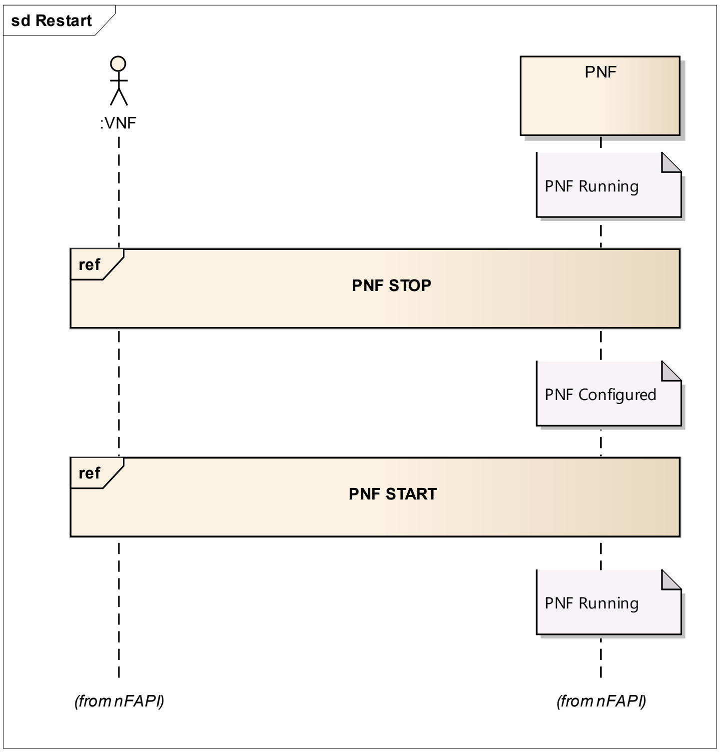


Figure 2-8 PNF Restart procedure

2.2.1.4 Reconfigure

The Reconfigure procedure is used to (optionally) move the PNF from the PNF RUNNING state back to the PNF CONFIGURED state, so the PNF can then be reconfigured with a new configuration, and then returned to the PNF RUNNING state. This procedure is defined as a sequence of the Stop (Figure 2-7), Config (Figure 2-5) and the Start (Figure 2-6) procedures.

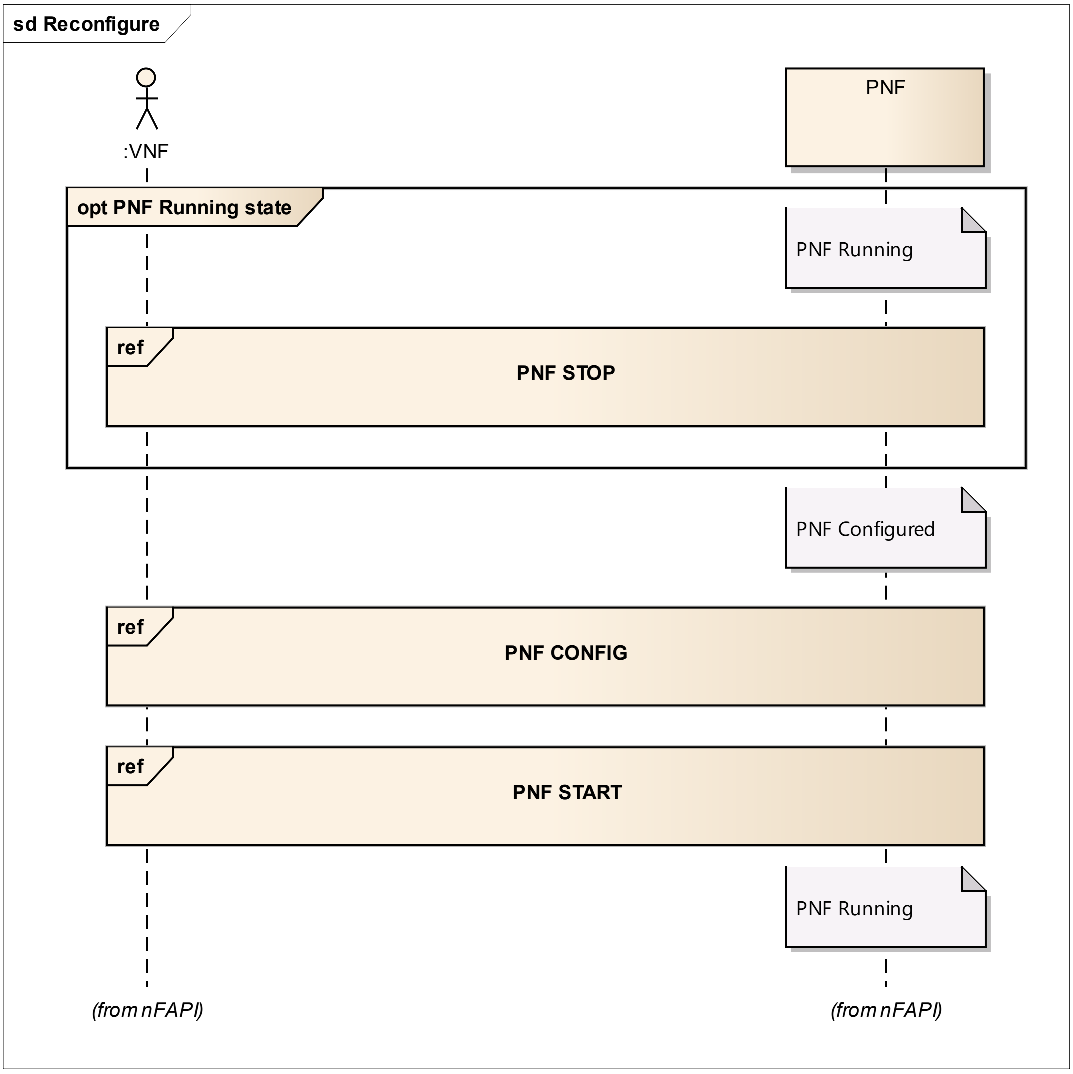


Figure 2-9 PNF Reconfigure procedure

2.2.2 P5 PHY Configuration Procedures

The nFAPI P5 interface includes procedures to configure the PHY, and is based on the FAPI P5 interface detailed in Section 3.2.1. This section only describes nFAPI behaviour where it differs from FAPI.

It should be noted that the format of the P5 PHY messages are different for nFAPI. The definition of nFAPI PARAM, CONFIG, START and STOP messages are specified in Sections 2.3.3.5 to 2.3.3.8.

2.2.2.1 Initialization

The initialization procedure described for FAPI also applies in nFAPI. However, instead of the `SUBFRAME.indication` message, the Node Sync procedures follow the Start message, as shown in Figure 2-10.

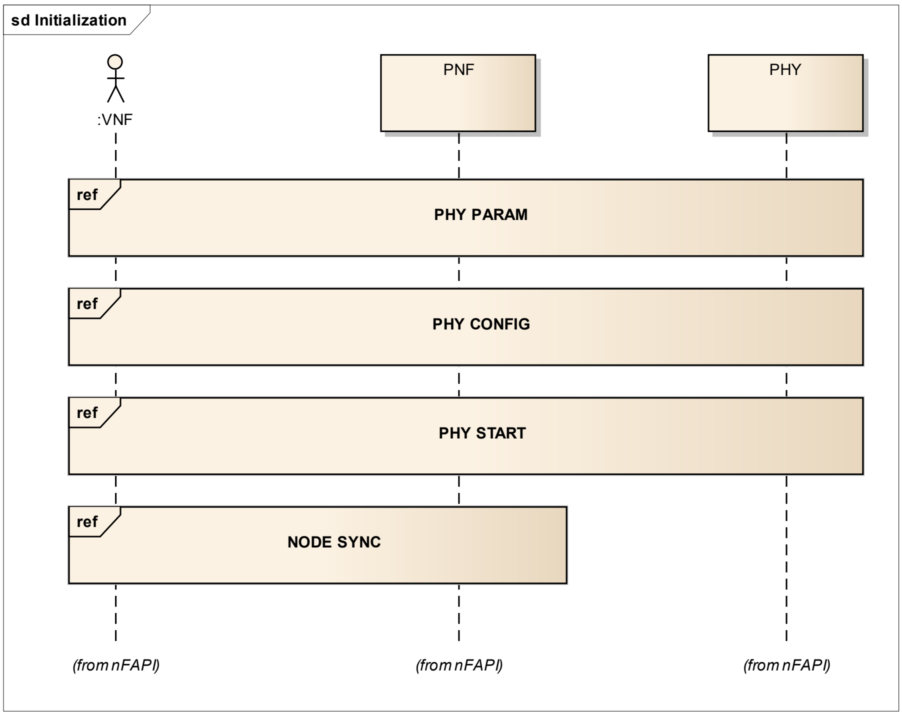


Figure 2-10 PHY Initialization procedure

2.2.2.2 Termination

The termination procedure described for FAPI also applies in nFAPI.

2.2.2.3 Restart

The restart procedure described for FAPI also applies in nFAPI.

2.2.2.4 Reset

The restart procedure described for FAPI also applies in nFAPI. It is part of the PNF reset procedure described in Section 2.2.1.2.

2.2.2.5 Reconfigure

The reconfigure procedure described for FAPI also applies in nFAPI.

2.2.2.6 Query

The query procedure described for FAPI also applies in nFAPI.

2.2.2.7 Notification

The notification procedure described for FAPI also applies in nFAPI.

2.2.2.8 Measurements

The measurements procedure is used by the VNF to request cell-wide measurements from a PHY, and is shown in Figure 2-11. The procedure is only valid when the specified PHY is in the running state, and is initiated by the VNF sending the `MEASUREMENT.request` message configured to request one or more measurements. If successful the `MEASUREMENT.response` message is returned when all the measurements have completed. If an error occurs `MEASUREMENT.response` message is also used to return an error.

The procedure used by the PNF, PHY and RF to perform the measurements is outside the scope of nFAPI.

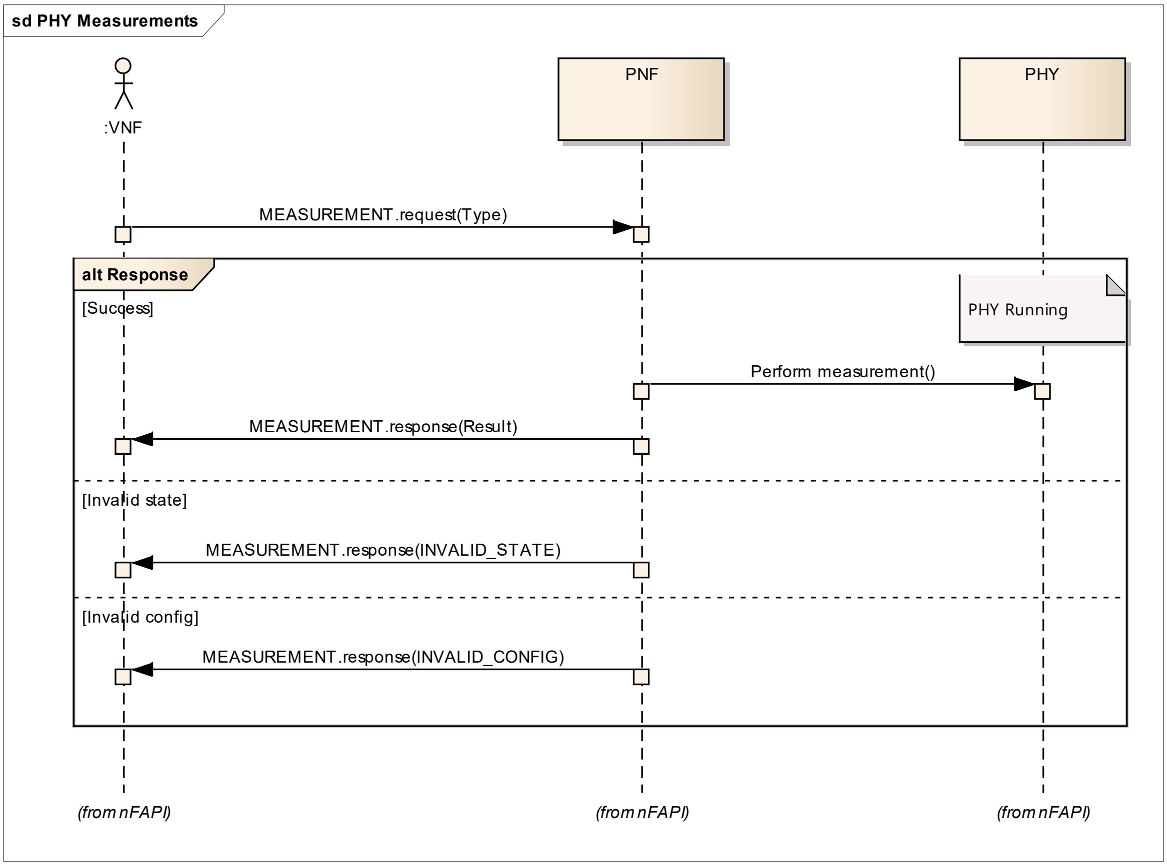


Figure 2-11 Measurement procedure

2.2.3 P7 Subframe Procedures

The nFAPI P7 interface configures the PNF PHYs every subframe, and is based on the FAPI P7 interface detailed in Section 3.2.2. This section only describes nFAPI behaviour where it differs from FAPI.

2.2.3.1 SUBFRAME signal

In FAPI a `SUBFRAME.indication` message is sent from the PHY at the start of every 1ms subframe. In nFAPI this message is not used, since the expected jitter on the fronthaul connection between VNF and PNF prevents this message from being a suitable mechanism for signalling a 1ms interval.

Instead, the combination of the PHY sync procedure and API message timing procedure ensures the alignment of MAC commands to air-interface subframes.

2.2.3.2 SFN/SF synchronization

The SFN/SF synchronization mechanism described for FAPI does not apply in nFAPI. Instead, the PHY synchronization mechanism described next is used to maintain sync between the VNF and PNF.

2.2.3.3 PHY Synchronization

The nFAPI synchronization procedure relates to the estimation and compensation of timing differences between the VNF and the PHY instance. By nFAPI synchronization is generally meant the achievement of a common timing reference between the VNF and the PHY instances. The PHY instances within a PNF or between PNFs may or may not be subframe aligned; therefore this procedure permits the VNF to synchronize to each

PHY instance. For interference and PHY coordination, it may be preferable for the PHY instances to align their time bases. This capability is exchanged between the PNF and the VNF through the PNF PARAM procedure.

The nFAPI synchronization between VNF and PHY instance can be used to find out the timing reference differences between them as well as optionally permitting the VNF to instruct the PHY instance to update its subframe number based on the offset defined by the VNF. The use is mainly for determining subframe timing flows in order to permit P7 subframe procedures to occur in a timely fashion. The internal subframe timing requirements for the P7 procedures are exchanged through the PNF PARAM procedure.

If an accurate reference timing signal is used within the PNF, the frequency deviation between PNF PHY instances will be low, but could occur. If no accurate reference timing signal is available, the PNF reference oscillator must be relied upon. Then the nFAPI synchronization procedure can be used as a process to find out the frequency deviation between nodes.

In the nFAPI synchronization procedure (as illustrated in Figure 2-12), the VNF sends a DL Node Sync message to the PHY instance containing the parameter T1. Upon reception of a DL Node Sync message, the PHY instance shall respond with UL Node Sync message, indicating T2 and T3, as well as T1 which was indicated in the initiating DL Node Sync control message.

As configured by the VNF, the PNF PHY Instance can send either periodic or aperiodic Timing Info messages informing the VNF of the P7 subframe message time of arrival. If configured to aperiodic messages, the Timing Info message is sent when a subframe completes and the set of subframe messages either arrives too late or too early (depending on window configuration) - see Figure 2-12. The actions taken by the VNF in response to the Timing Info is out of scope for this specification.

Note, this exact timing definition associated with what constitutes too early or too late is implementation specific and is outside the scope of this document.

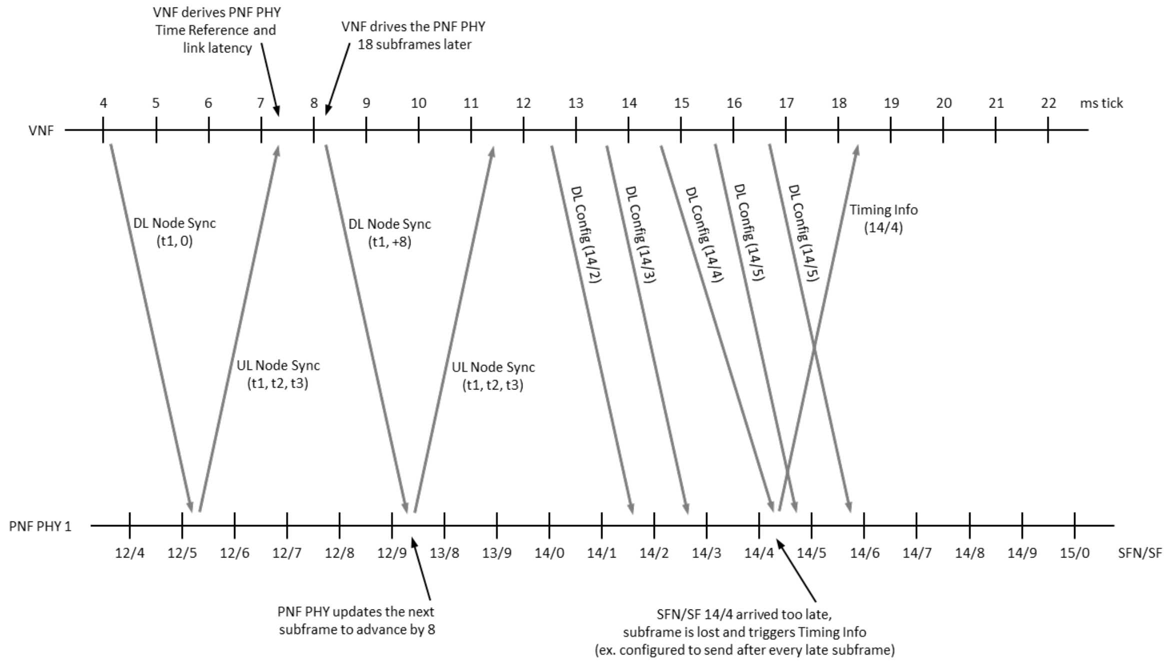


Figure 2-12 Synchronization timing sequence with DL Config example

2.2.3.4 API message order

The API message order described for FAPI also applies in nFAPI.

2.2.3.5 API message timing

The nFAPI P7 messages sent from the VNF to PNF must arrive at the PNF a minimum time before the subframe they configure starts transmission on the air interface. These messages are `DL_CONFIG.request`, `TX.request`, `UL_CONFIG.request` and `HI_DCI0.request`, and the latest time of arrival is specified in microseconds (µs) in the `PNF_PARAM.response` message.

If the message arrives in the allowed timing window then the PNF continues normal operation for the subframe. If the message is late then the PNF follows the procedure shown in Figure 2-13:

- If `DL_CONFIG.request` is received too late
- Return a `Timing.indication` message including the SFN, the lateness of the message in µs and indicating this error applies to `DL_CONFIG.request`.

The same procedure is used for `DL_CONFIG.request`, `TX.request`, `UL_CONFIG.request` and `HI_DCI0.request`. The VNF should react to each error messages as follows:

- If the error is for `DL_CONFIG.request` lateness the VNF shall consider the DL subframe lost
- If the error is for `TX.request` lateness the VNF shall consider the DL subframe lost
- If the error is for `UL_CONFIG.request` lateness the VNF shall consider the UL subframe lost
- If the error is for `HI_DCI0_CONFIG.request` lateness the VNF shall consider the UL subframe being scheduled to be lost

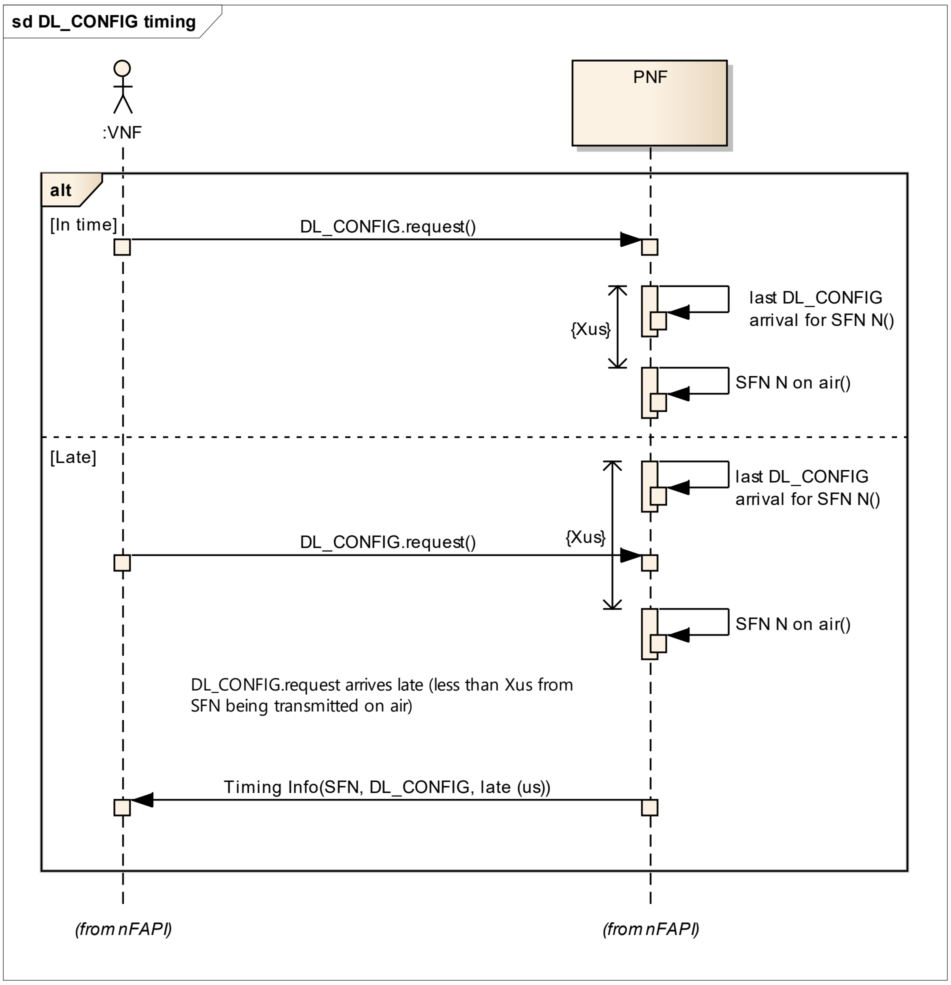


Figure 2-13 DL_CONFIG.request lateness procedure

2.2.3.6 Semi-static information

In nFAPI all cell-specific and UE-specific semi-static information is stored in the MAC.

2.2.3.7 Uplink HARQ signalling

In nFAPI the calculation of the uplink HARQ location is performed in the MAC.

2.2.3.8 Downlink

The downlink subframe procedures for FAPI also apply in nFAPI, with semi-static information storage and uplink HARQ signalling calculation performed in the MAC.

2.2.3.9 Uplink

The uplink subframe procedures for FAPI also apply in nFAPI, with semi-static information storage and uplink HARQ signalling calculation performed in the MAC.

2.2.3.10 RNTI Measurements

UE-specific measurements are reported using the subframe indication messages, and are associated to a specific RNTI. Currently, nFAPI supports the same subframe messages as FAPI:

- SNR
- Timing advance
- Doppler estimation

2.2.4 nFAPI Error Procedures

2.2.4.1 General

For each of the specified events below, the receiver of the nFAPI message shall discard the entire message and send a response with invaid content error (P5) or error indication (P7).

- Invalid message length
- Invalid information element length
- Invalid information element value
- Unknown or unforeseen message type
- Missing mandatory information element
- Unknown information elements
- Out of sequence information elements
- Repeated information elements
- Conditional information element error

2.3 nFAPI messages

2.3.1 Transport Layer

2.3.1.1 nFAPI P5

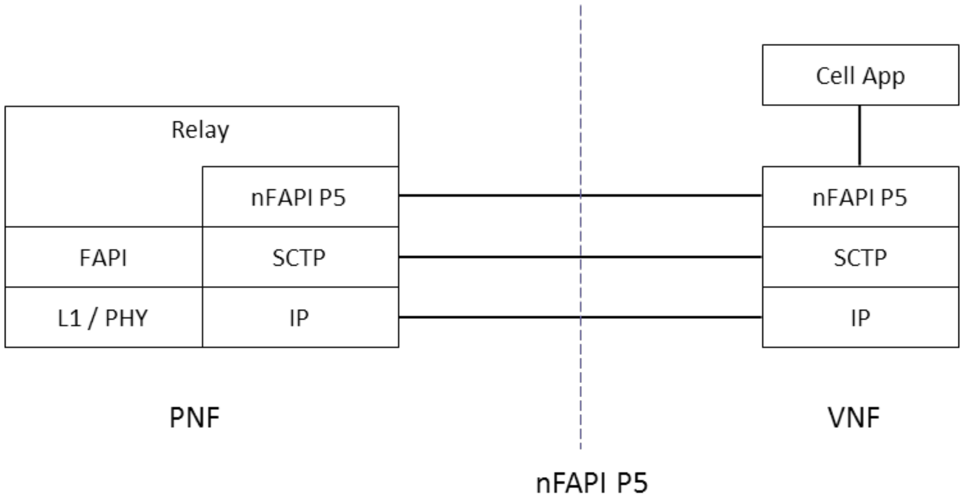


Figure 2-14 nFAPI P5 Protocol Stack

The transport layer for nFAPI P5 is required to ensure reliable message transfer, so SCTP is used as the transport protocol. Therefore there are no extra methods to ensure reliable transmission of messages at the application level (such as timers, packet validation and packet loss).

The payload protocol identifier (PPI) field in SCTP is set to the value 7701, as assigned by IANA[1].

There is one SCTP association between the PNF and the VNF, established by the PNF based on P9 OAM configuration of VNF address and port. There is a single SCTP stream for each nFAPI PHY instance to transport P5 messages.

[1] http://www.iana.org/assignments/service-names-port-numbers

Transport network redundancy may be achieved by SCTP multi-homing between two end-points, of which one or both is assigned multiple IP addresses. SCTP end-points support a multi-homed remote SCTP end-point. For SCTP endpoint redundancy an INIT may be sent from the VNF or PNF at any time for an already established SCTP association, which shall be handled as defined in IETF RFC 4960 [18].

2.3.1.2 nFAPI P7

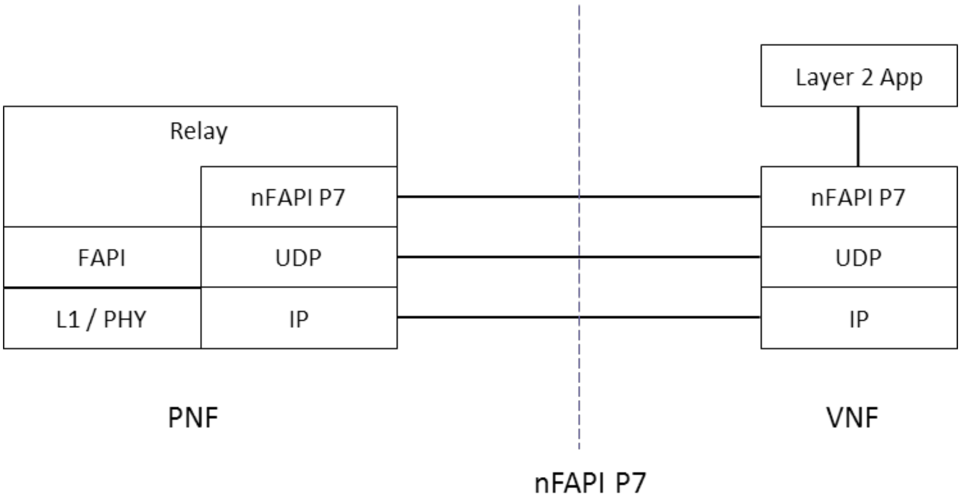


Figure 2-15 nFAPI P7 Protocol Stack

The transport layer for nFAPI P7 does not require reliable transport due to the timing constraints of retransmission, so UDP is used as the transport protocol. There are application level elements (sequence numbers) in the P7 protocol to ensure that lost packets can be identified and handled accordingly.

The address and port to use at the VNF and PNF are exchanged over the P5 interface through PNF PHY configuration. The PHY ID within the P7 message header defines which PNF PHY instance / VNF interface the message is directed towards. These configuration options permit either a single UDP stream for a PNF with multiple PHY instances, or, alternatively, a UDP stream per PHY instance.

2.3.1.3 Securing the nFAPI interface

An IPSec tunnel between the PNF and VNF may be used to protect the P5 and P7 communications. The details associated with configuring the Security Gateway address in the PNF and establishing/maintaining the IPSec tunnel are out of scope of this document.

2.3.2 General Message Format

Each nFAPI message contains the header (Figure 2-16) followed by a single message body. The message body contains the sequence of values (V) and type / length / value (TLV) structures as defined in section 2.3.3.

The headers in Figure 2-16 contain the following parameters:

PHY ID	Within the PNF Device, the unique identity of the PHY instance as assigned through the PNF_Config.request. This value is reserved (set to zero on transmission and ignored on reception) for PNF Device messages: PNF PARAM

	PNF CONFIG PNF START PNF STOP
Message ID	The nFAPI message identity as defined in section 2.3.2.1
Message Length	The length in bytes of the message including the header
Spare	Reserved field to be populated with zeros on transmission and ignored on reception
Length	The length in bytes of the message segment including the header
M	A More flag indicating there are more segments to follow to complete the entire message
Segment Number	The segment number starting at zero and incrementing by one between each segment
Sequence Number	The incrementing sequence number for all complete messages over the P7 nFAPI interface per PHY instance
Checksum	The checksum of the whole message segment (including header) as calculated using the CRC32c algorithm following the same method as the SCTP protocol – defined in IETF RFC 4960 [18] The Checksum is optional to populate and must be filled with zero's when not used
Transmit Timestamp	The offset from VNF SFN/SF 0/0 time reference of the message transmission at the transport layer, in microseconds, with a range of 0 to 10239999 This timestamp shall be used to derive the jitter values in the Timing Info message. It is mandatory to populate this field for the following messages: DL_CONFIG.request UL_CONFIG.request HI_DCI0.request Tx.request This value is reserved (set to zero on transmission and ignored on reception) for all other messages.

The P7 message header permits segmentation of the messages if required by transport mechanisms. The Sequence Number increments across complete messages for a per PHY instance while the Segment Number defines the segment order within a message. The More flag defines if more segments will follow.

Techniques for determining whether segmentation is required are outside the scope of this document.

Header format for P4/P5

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>20</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>30</td><td>1</td></tr>
<tr><td colspan="16">PHY ID</td><td colspan="16">Message ID</td></tr>
<tr><td colspan="16">Message Length</td><td colspan="16">Spare</td></tr>
</table>

Header format for P7

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>20</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>30</td><td>1</td></tr>
<tr><td colspan="16">PHY ID</td><td colspan="16">Message ID</td></tr>
<tr><td colspan="16">Length</td><td colspan="1">M</td><td colspan="7">Segment Number</td><td colspan="8">Sequence Number</td></tr>
<tr><td colspan="32">Checksum</td></tr>
<tr><td colspan="32">Transmit Timestamp</td></tr>
</table>

Figure 2-16 Message header formats

The TLV parameters are defined as the tag specified in the subsequent sections and re-stated in section 2.3.5, the Length field, defined as the length of the subsequent value, and the value field which contains the defined value or structure.

Fields marked as reserved are to be populated as zeros on transmission and ignored on reception. Fields marked as not present for nFAPI are not present within the message structure.

All multi-octet fields representing integers are laid out in big endian order (also known as "most significant byte first", or "network byte order"). All fields as defined in the message structures shall be transmitted in the sequence defined and are packed with no padding. Bit fields are defined by their bit number with bit 0 being the most significant bit (therefore the left most bit).

2.3.2.1 nFAPI Message ID Values

The nFAPI P7 messages which are common with the corresponding FAPI P7 messages as defined in section 4 use the identical Message ID values as defined in Table 3-4, (with the upper 8 bits set to zero).

The nFAPI P5 and P7 messages which are specific to nFAPI use Message ID values defined in Table 2-1.

Message	Value	Message Body Definition
`PNF PARAM.request`	0x0100	See Section 2.3.3.1
`PNF PARAM.response`	0x0101	See Section 2.3.3.1
`PNF CONFIG.request`	0x0102	See Section 2.3.3.2
`PNF CONFIG.response`	0x0103	See Section 2.3.3.2
`PNF START.request`	0x0104	See Section 2.3.3.3
`PNF START.response`	0x0105	See Section 2.3.3.3
`PNF STOP.request`	0x0106	See Section 2.3.3.4
`PNF STOP.indication`	0x0107	See Section 2.3.3.4

Message	Value	Message Body Definition
PARAM.request	0x0108	See Section 2.3.3.5
PARAM.response	0x0109	See Section 2.3.3.5
CONFIG.request	0x010a	See Section 2.3.3.6
CONFIG.response	0x010b	See Section 2.3.3.6
START.request	0x010c	See Section 2.3.3.7
START.response	0x010d	See Section 2.3.3.7
STOP.request	0x010e	See Section 2.3.3.8
STOP.indication	0x010f	See Section 2.3.3.8
MEASUREMENT.request	0x0110	See Section 2.3.3.9
MEASUREMENT.response	0x0111	See Section 2.3.3.9
RESERVED for P5 messages	0x0112-0x017f	
DL_Node Sync	0x0180	See Section 2.3.4.1
UL_Node Sync	0x0181	See Section 2.3.4.2
Timing Info	0x0182	See Section 2.3.4.3
RESERVED for P7 messages	0x0183-0x01ff	
RESERVED for NMM (P4) messages	0x0200-0x02ff	See Table 5-1
RESERVED for Vendor Extension messages	0x0300-0x03ff	See Section 2.3.2.2
RESERVED	0x0400-0xffff	

Table 2-1 nFAPI-specific P5 & P7 message types

2.3.2.2 Vendor Extensions

There are optional vendor extensions within all nFAPI P4, P5 and P7 messages. These vendor extensions are contained within the message structure and shall immediately follow the standard nFAPI message content. The vendor extension shall contain a Vendor Extension Tag and Length field of the vendor extension following the same rules as the nFAPI structures, with the subsequent content to be the vendor extension payload. The nFAPI header message length shall be inclusive of the vendor extension.

There are also defined vendor extension message identifier ranges that permit entire nFAPI messages to be added for vendor extension procedures. These are defined with a Vendor Extension Message ID and shall follow the nFAPI message header content and rules, with the subsequent content to be the vendor extension payload.

As an example, Figure 2-17 shows an nFAPI message with the nFAPI header, two nFAPI structures and a single nFAPI Vendor Extension. Note that the elements in this example are shown for sequencing within the message without definition of element sizes.

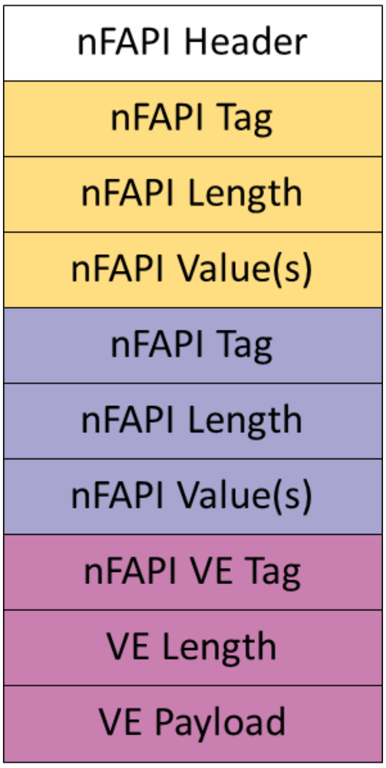


Figure 2-17 Vendor Extension Example

To remain interoperable with VNFs which do not understand the vendor extensions from a specific PNF vendor, a PNF should not rely on any change in behaviour of the VNF as a result of the inclusion of vendor extensions by the PNF, and should not modify its behaviour or procedures away from this specification unless a VNF responds (in a vendor extension for that PNF vendor) in such a way that indicates it can perform vendor-specific behaviour.

The range of nFAPI VE Tags available for use is defined in Table 2-2 and is a reserverd set of Tags that can be used by the individual vendors as required.

Tag (Hex)	Description
0xF000 - FFFF	Range of Tag values available for use as vendor extension Tags for use with nFAPI messages

Table 2-2 nFAPI vendor extension tag values

The vendor of the PNF may be identified from the OUI contained within the PNF PARAM response message. The VNF may use this OUI value in order to determine how to interpret the VE message identifiers and the VE Tags within the standard messages as well as the nFAPI vendor specific messages.

2.3.3 nFAPI P5 Messages

2.3.3.1 PNF PARAM

PNF_PARAM.request

This message is sent by the VNF when the PNF is in the PNF IDLE state. There is no message body in the `PNF_PARAM.request`.

PNF_PARAM.response

The `PNF_PARAM.response` message defines the PNF's overall capability and options for the VNF to follow up with the `PNF_CONFIG.request`.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x1000	PNF Param General	struct	The general PNF capability parameters as defined in Table 2-4	TLV
0x1001	PNF PHY	struct	The PHY capability parameters as defined in Table 2-5	TLV
0x1002	PNF RF	struct	The RF capability parameters as defined in Table 2-6	TLV
0100a	PNF PHY Rel 10	struct	The PHY capability parameters relating to Rel 8/9/10 as defined in Table 2-7	TLV
0x100b	PNF PHY Rel 11	struct	The PHY capability parameters relating to Rel 11 as defined in Table 2-8	TLV
0x100c	PNF PHY Rel 12	struct	The PHY capability parameters relating to Rel 12 as defined in Table 2-9	TLV
0x100d	PNF PHY Rel 13	struct	The PHY capability parameters relating to Rel 13 as defined in Table 2-10	TLV
0x100e	PNF PHY Rel 13 NB-IOT	struct	The PHY capability parameters relating to Rel 13 NB-IOT as defined in Table 2-11	TLV

Table 2-3 PNF_PARAM.response parameters

PNF Param General

Field	Type	Description
Tag	uint16_t	The PNF Param General Tag
Length	uint16_t	The length in bytes of the values
nFAPI Sync Mode	uint8_t	The method of nFAPI Synchronization supported by the PNF. 0 = un-aligned synchronization[2] 1 = internal PNF frame alignment[3] 2 = absolute time aligned synchronization[4] Other values are invalid

[2] The PHY(s) within the PNF device operate to individual frame alignment
[3] The PHY(s) within the PNF device operate to a shared frame alignment
[4] The PHY(s) within the PNF device operate to an absolute time frame alignment

Field	Type	Description
Location Mode	unit8_t	The method of location derivation supported by the PNF. 0 = none 1 = GPS 2 = GLONASS 3 = BeiDou Other values are Reserved
Location Coordinates Length	uint16_t	Length in bytes of the Location Coordinates Array field
Location Coordinates	Array of uint8_t	The Location of the PNF. The value is formatted as the LocationCoordinates IE using BASIC-PER encoding as defined in [17] section 6.4.2. The first bit of the LocationCoordinates IE is in the LSB of the first byte of the array. The MSBs of the last element of the array may be padded with zeros if the ASN.1 element is not an integer number of bytes.
DL Config Timing	uint32_t	The timing offset before the air interface subframe start that the DL_Config.request must be received at the PNF. The value is in microseconds.
TX Timing	uint32_t	The timing offset before the air interface subframe start that the TX.request must be received at the PNF. The value is in microseconds.
UL Config Timing	uint32_t	The timing offset before the air interface subframe start that the UL_CONFIG.request must be received at the PNF. The value is in microseconds.
HI_DCI0 Timing	uint32_t	The timing offset before the air interface subframe start that the HI_DCI0.request must be received at the PNF. The value is in microseconds.
Maximum Number PHYs	uint16_t	The maximum number of operational PHYs supported by the PNF device.
Maximum Total Bandwidth	uint16_t	The total maximum bandwidth (in units of 100kHz) supported by the PNF device.
Maximum Total Number of DL Layers	uint8_t	The maximum total number of downlink layers supported.
Maximum Total Number of UL Layers	uint8_t	The maximum total number of uplink layers supported across all available PHYs.
Shared Bands	uint8_t	Indication that the PNF device shares the list of RF band options available across all available PHYs, so each may only be used with a single PHY.

Field	Type	Description
		0 = False 1 = True Other values are Reserved
Shared PA	uint8_t	Indication that the PNF device shares a single RF PA across all available PHYs, so that the maximum Total Power is shared across all available PHYs. 0 = False 1 = True Other values are Reserved
Maximum Total Power	int16_t	The maximum transmit power of the PNF device summed across all PHYs. Value is in 0.1dB steps with 0 = 0dBm Valid only if Shared PA is True.
OUI[3]	array of uint8_t	The PNF OUI in the format as specified by IEEE

Table 2-4 PNF Param general parameters

PNF PHY

Field	Type	Description
Tag	uint16_t	The PNF PHY Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
Number of RF	uint16_t	The number of RF configurations
Number of RF {		
RF Config Index	uint16_t	The index number defined in the PNF RF struct (Table 2-6) that the PHY can support
}		
Number of RF Exclusion	uint16_t	The number of RF configurations excluded from use by this PHY
Number of RF Exclusion {		
RF Config Index	uint16_t	The index number defined in the PNF RF struct (Table 2-6) that the PHY cannot use
}		

	Field	Type	Description
	Downlink Channel Bandwidth Supported	uint16_t	The downlink channel bandwidth supported in resource blocks as specified in 3GPP TS 36.104. Value: bitX: 0 = not supported, 1= supported. Bit0: 6 Bit1: 15 Bit2: 25 Bit3: 50 Bit4: 75 Bit5: 100 Bit6 – Bit15: Reserved
	Uplink Channel Bandwidth Supported	uint16_t	The uplink channel bandwidth supported in resource blocks as specified in 3GPP TS 36.104. Value: bitX: 0 = not supported, 1= supported. Bit0: 6 Bit1: 15 Bit2: 25 Bit3: 50 Bit4: 75 Bit5: 100 Bit6 – Bit15: Reserved
	Number of DL layers supported	uint8_t	The maximum number of downlink layers supported. Values, 1, 2, 4, 8 Other values are Reserved
	Number of UL layers supported	uint8_t	The maximum number of uplink layers supported. Values, 1, 2, 4 Other values are Reserved
	Maximum 3GPP Release Supported	uint16_t	The major 3GPP releases supported. Value: bitX: 0 = not supported, 1= supported. Bit0: Rel 8 Bit1: Rel 9 Bit2: Rel 10 Bit3: Rel 11 Bit4: Rel 12 Bit5: Rel 13 Bit6 – Bit15: Reserved
	NMM Modes Supported	uint8_t	Network Monitor Modes Supported. 0 = None 1 = NMM in Configured State 2 = NMM in Running State 3 = NMM in Configured and Running State

Field	Type	Description
		Other values are Reserved
}		

Table 2-5 PNF PHY parameters

PNF RF

Field	Type	Description
Tag	uint16_t	The PNF RF Tag
Length	uint16_t	The length in bytes of the values
Number of RF	uint16_t	The number of RF instances
Number of RF {		
RF Config Index	uint16_t	The unique index number for the PHY to associate with this instance of RF configuration
Band	uint16_t	Band number as specified in 3GPP TS36.101
Maximum Transmit Power	int16_t	The maximum transmit power for the RF chain operating at the maximum supported bandwidth as defined in 3GPP TS 36.104. Value is in 0.1dB steps with 0 = 0dBm
Minimum Transmit Power	int16_t	The minimum transmit power for the RF chain operating at the maximum supported bandwidth as defined in 3GPP TS 36.104. Value is in 0.1dB steps with 0 = 0dBm
Number of antennas supported	uint8_t	The maximum number of antennas supported. Values, 1, 2, 4, 8 Other values are Reserved
Minimum Downlink Frequency	uint32_t	The minimum supported downlink frequency in 100kHz units
Maximum Downlink Frequency	uint32_t	The maximum supported downlink frequency in 100kHz units
Minimum Uplink Frequency	uint32_t	The minimum supported uplink frequency in 100kHz units
Maximum Uplink Frequency	uint32_t	The maximum supported uplink frequency in 100kHz units
}		

Table 2-6 PNF RF parameters

PNF PHY Rel 10

Field	Type	Description
Tag	uint16_t	The PNF PHY Rel 10 Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
Transmission mode 7 supported	uint16_t	Indicates if PHY supports TM7 for PDSCH. Value: 0 = not supported, 1= supported.
Transmission mode 8 supported	uint16_t	Indicates if PHY supports TM8 for PDSCH. Value: 0 = not supported, 1= supported.
Two antennas ports for PUCCH	uint16_t	Indicates if PHY supports PUCCH transmit diversity introduced in Release 10. Equivalent to two-AntennaPortsForPUCCH-r10 in [13]. Value: 0 = not supported, 1= supported
Transmission mode 9 supported	uint16_t	Indicates if PHY supports TM9 for PDSCH with 8 antennas and 8 CSI. Equivalent to tm9-With-8Tx-FDD-r10 in [13]. Value: 0 = not supported, 1= supported.
Simultaneous PUCCH PUSCH	uint16_t	Indicates if PHY supports UE sending simultaneous PUCCH and PUSCH introduced in Release 10. Equivalent to simultaneousPUCCH-PUSCH-r10 in [13]. Value: 0 = not supported, 1= supported
Four layer Tx with TM3 and TM4	uint16_t	Indicates if PHY supports four layer transmission for TM3 and TM4. Equivalent to fourLayerTM3-TM4-r10 in [13]. Value: 0 = not supported, 1= supported
}		

Table 2-7 PNF PHY Rel 10 parameters

PNF PHY Rel 11

Field	Type	Description
Tag	uint16_t	The PNF PHY Rel 11 Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
ePDCCH supported	uint16_t	Indicates if PHY supports Enhanced PDCCH. Value: 0 = not supported, 1= supported
Multi ACK CSI reporting	uint16_t	Indicates if PHY supports the multi ACK and CSI reporting required with CA and mixed FDD/TDD carriers. Equivalent to multiACK-CSI-Reporting-r11 in [13]. Value: 0 = not supported, 1= supported.
PUCCH Tx diversity with channel selection	uint16_t	Indicates if PHY supports transmit diversity for PUCCH format 1b with channel selection. Equivalent to txDiv-PUCCH1b-ChSelect in [13]. Value: 0 = not supported, 1= supported.
UL CoMP supported	uint16_t	Indicates if PHY supports UL CoMP. Value: 0 = not supported, 1= supported.
Transmission mode 5 supported	uint16_t	Indicates if PHY supports TM5 for PDSCH. Value: 0 = not supported, 1= supported.
}		

Table 2-8 PNF PHY Rel 11 parameters

PNF PHY Rel 12

Field	Type	Description
Tag	uint16_t	The PNF PHY Rel 12 Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
CSI subframe set	uint16_t	Equivalent to csi-SubframeSet-r12 in [13].

Field	Type	Description
		Value: 0 = not supported, 1= supported.
Enhanced 4TX codebook	uint16_t	Indicates if PHY supports the enhanced 4TX codebook. Equivalent to enhanced-4TxCodebook-r12 in [13]. Value: 0 = not supported, 1= supported.
DRS supported	uint16_t	Indicates if PHY supports the Discovery Reference Signal. Value: 0 = not supported, 1= supported
UL 64QAM supported	uint16_t	Indicates if PHY support 64 QAM in the uplink Value: 0 = not supported, 1= supported.
Transmission mode 10 supported	uint16_t	Indicates if PHY supports TM10 for PDSCH (DL CoMP). Value: 0 = not supported, 1= supported.
Alternative TBS indices	uint16_t	Indicates if PHY supports the alternate TBS indices (256 QAM). Equivalent to alternativeTBS-Indices-r12 in [13]. Value: 0 = not supported, 1= supported.
}		

Table 2-9 PNF PHY Rel 12 parameters

PNF PHY Rel 13

Field	Type	Description
Tag	uint16_t	The PNF PHY Rel 13 Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
PUCCH format 4 supported	uint16_t	Indicates if PHY supports PUCCH format 4. Value: 0 = not supported, 1= supported.
PUCCH format 5 supported	uint16_t	Indicates if PHY supports PUCCH format 5.

Field	Type	Description
		Value: 0 = not supported, 1= supported.
More than 5 CA support	uint16_t	Indicates if PHY supports features required for more than 5 CA support on PUSCH. Equivalent to uci-PUSCH-Ext-r13 in [13]. Value: 0 = not supported, 1= supported.
LAA supported	uint16_t	Indicates if PHY supports DL LAA (subframe format 3). Value: 0 = not supported, 1= supported.
LAA ending in DwPTS supported	uint16_t	Indicates if PHY supports DL LAA ending in a DwPTS subframe. Equivalent to endingDwPTS-r13i n [13]. Value: 0 = not supported, 1= supported.
LAA starting in second slot Supported	uint16_t	Indicates if PHY supports DL LAA starting in the second slot in a subframe. Equivalent to secondSlotStartingPosition-r13 in [13]. Value: 0 = not supported, 1= supported.
Beamforming Supported	uint16_t	Indicates if PHY supports beamforming (FD-MIMO Class B). Equivalent to beamformed-r13 in [13]. Value: 0 = not supported, 1= supported.
CSI-RS enhancements supported	uint16_t	Indicates if PHY supports CSI-RS enhancements (FD-MIMO Class A). Equivalent to csi-RS-EnhancementsTDD-r13 in [13]. Value: 0 = not supported, 1= supported.
DMRS enhancements supported	uint16_t	Indicates if PHY supports DMRS enhancements added in Release 13. Equivalent to dmrs-Enhancements-r13 in [13]. Value: 0 = not supported, 1= supported.
SRS enhancements supported	uint16_t	Indicates if PHY supports SRS enhancements added in Release 13. Equivalent to srs-Enhancements-r13in [13]. Value: 0 = not supported, 1= supported.
}		

Table 2-10 PNF PHY Rel 13 parameters

PNF PHY Rel 13 NB-IOT

Field	Type	Description
Tag	uint16_t	The PNF PHY Rel 13 NB-IOT Tag
Length	uint16_t	The length in bytes of the values
Number of PHY	uint16_t	The number of PHY instances
Number of PHY {		
PHY Config Index	uint16_t	The unique index number of the PHY to permit the PNF to identify the PHY in the PNF_CONFIG.Request
Number of RF	uint16_t	The number of RF configurations
Number of RF {		
RF Config Index	uint16_t	The index number defined in the PNF RF struct (Table 2-6) that the PHY can support
}		
Number of RF Exclusion	uint16_t	The number of RF configurations excluded from use by this PHY
Number of RF Exclusion {		
RF Config Index	uint16_t	The index number defined in the PNF RF struct (Table 2-6) that the PHY cannot use
}		
Number of DL layers supported	uint8_t	The maximum number of downlink layers supported. Values, 1, 2 Other values are Reserved
Number of UL layers supported	uint8_t	The maximum number of uplink layers supported. Values, 1, 2 Other values are Reserved
Maximum 3GPP Release Supported	uint16_t	The major 3GPP releases supported. Value: bitX: 0 = not supported, 1= supported. Bit0 – Bit4: Reserved Bit5: Rel 13 Bit6 – Bit15: Reserved
NMM Modes Supported	uint8_t	Network Monitor Modes Supported. 0 = None 1 = NMM in Configured State 2 = NMM in Running State 3 = NMM in Configured and Running State Other values are Reserved
}		

Table 2-11 PNF PHY Rel 13 NB-IOT parameters

PNF PARAM Errors

The error codes which may be returned in `PNF_PARAM.response` are given in Table 2-12.

PNF PARAM Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The PNF_PARAM.request was received when the PNF was not in the PNF IDLE state.
MSG_INVALID_CONFIG	The PNF_PARAM.request was received with invalid configuration.

Table 2-12 Error codes for PNF_PARAM.response

Note, in this version of nFAPI, no additional information is appended to the invalid config cause. Vendor extensions can be added if specific error causes or details on error reasons are required.

2.3.3.2 PNF CONFIG

PNF_CONFIG.request

The `PNF_CONFIG.request` message configures the PNF for operation and if the PNF is in the PNF IDLE state, instructs the PNF to move into the PNF CONFIGURED state. Subsequent reception of the `PNF_CONFIG.request` in the PNF CONFIGURED state re-configures the PNF for the newly supplied configuration parameters.

Tag	Field	Type	Description	nFAPI
0x1003	PNF PHY_RF Config	struct	The PHY and RF configuration parameters as defined in Table 2-14	TLV

Table 2-13 PNF_CONFIG.request parameters

PNF PHY_RF Config

Field	Type	Description
Tag	uint16_t	The PNF PHY_RF Tag
Length	uint16_t	The length in bytes of the values
Number of PHY_RF Config	uint16_t	The number of PHY-RF Config instances
Number of PHY_RF Config {		
PHY ID	uint16_t	The PHY ID assigned by the VNF for this PHY and RF instance.
PHY Config Index	uint16_t	The index number defined in the PNF PHY struct (Table 2-5) to allocate the PHY ID
RF Config Index	unit16_t	The index number defined in the PNF RF struct (Table 2-6) to allocate the PHY ID to.
}		

Table 2-14 PNF PHY_RF Config parameters

PNF_CONFIG.response

The `PNF_CONFIG.response` message confirms the PNF state and acceptance or rejection of the `PNF_CONFIG.request` configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-15 PNF_CONFIG.response parameters

PNF CONFIG Errors

The error codes which may be returned in `PNF_CONFIG.response` are given in Table 2-16.

PNF CONFIG Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The PNF_CONFIG.request was received when the PNF was not in the PNF IDLE state or the PNF CONFIGURED state.
MSG_INVALID_CONFIG	The configuration provided contains parameters that are invalid or unsupported by the PNF.

Table 2-16 Error codes for PNF_CONFIG.response

2.3.3.3 PNF START

PNF_START.request

The `PNF_START.request` message moves the PNF into PNF RUNNING state from the PNF CONFIGURED state. No message body is defined for `PNF_START.request`.

PNF_START.response

The `PNF_START.response` message confirms the PNF state and acceptance or rejection of the `PNF_START.request` configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-17 PNF_START.response parameters

PNF_START Errors

The error codes which may be returned in `PNF_START.response` are given in Table 2-18.

PNF START Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `PNF_START.request` was received when the PNF was not in the PNF CONFIGURED state.

Table 2-18 Error codes for PNF_START.response

2.3.3.4 PNF STOP

PNF_STOP.request

The PNF_STOP.request message moves the PNF into PNF CONFIGURED state from the PNF RUNNING state. The PHYs are reset (if in the RUNNING state) on reception of the PNF_STOP.request, ready for re-configuration or re-starting as required. No message body is defined for PNF_STOP.request.

PNF_STOP.response

The PNF_STOP.response message confirms the PNF state and acceptance or rejection of the PNF_STOP.request configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-19 PNF_STOP.response parameters

PNF_STOP Errors

The error codes which may be returned in PNF_STOP.response are given in Table 2-20.

PNF STOP Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The PNF_STOP.request was received when the PNF was not in the PNF RUNNING state.

Table 2-20 Error codes for PNF_STOP.response

2.3.3.5 PARAM

PARAM.request

This message is be sent by the VNF when the PNF is in the PNF RUNNING state and the PHY is in the IDLE state. No message body is defined for PARAM.request.

PARAM.response

The PARAM.response message defines the PHY's overall capability and options for the requester to follow up with the CONFIG.request. The message consists of a list of TLVs defining the PNF PHY and RF capabilities and is a superset of the FAPI PARAM.response message.

The PARAM.response message body is defined in Table 3-5. It is the same as FAPI.

The TLVs included in the PARAM.response message in FAPI and nFAPI are defined in Table 3-6 to Table 3-9 and additional nFAPI TLVs are defined below in Table 2-21 and Table 2-22.

Description	Tag
Downlink UEs per Subframe	0x010A
Uplink UEs per Subframe	0x010B
nFAPI RF Bands	0x0114

Description	Tag
P7 PNF Address IPv4	0x0103 (if IPv4 supported)
P7 PNF Address IPv6	0x0104 (if IPv6 supported)
P7 PNF Port	0x0105
NMM GSM Frequency Bands	0x0130
NMM UMTS Frequency Bands	0x0131
NMM LTE Frequency Bands	0x0132
NMM Uplink RSSI supported	0x0133

Table 2-21 Additional nFAPI TLVs included in `PARAM.response` when PHY is in IDLE state

Description	Tag
Downlink UEs per Subframe	0x010A
Uplink UEs per Subframe	0x010B
nFAPI RF Bands	0x0114
P7 PNF Address IPv4	0x0103 (if IPv4 supported)
P7 PNF Address IPv6	0x0104 (if IPv6 supported)
P7 PNF Port	0x0105
P7 VNF Address IPv4	0x0100 (if IPv4 supported)
P7 VNF Address IPv6	0x0101 (if IPv6 supported)
P7 VNF Port	0x0102
Timing window	0x011E
Timing info mode	0x011F
Timing info period	0x0120
Maximum Transmit Power	0x0128
NMM GSM Frequency Bands	0x0130
NMM UMTS Frequency Bands	0x0131
NMM LTE Frequency Bands	0x0132
NMM Uplink RSSI supported	0x0133

Table 2-22 Additional nFAPI TLVs included in `PARAM.response` when PHY is in CONFIGURED state

PARAM Errors

The error codes which may be returned in `PARAM.response` are given in Table 2-23.

PARAM Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `PARAM.request` was received when the

Error Code	Description
	PHY was not in the IDLE state.

Table 2-23 Error codes for PARAM.response

2.3.3.6 CONFIG

CONFIG.request
This message is sent by the VNF when the PNF is in the PNF RUNNING state and the PHY is in the IDLE state. The `CONFIG.request` message configures the PHY for operation and instructs the PHY to move into the CONFIGURED state. Subsequent reception of the `CONFIG.request` in the CONFIGURED state re-configures the PHY for the newly supplied configuration parameters.

The `CONFIG.request` message body is defined in Table 3-13. It is the same as FAPI.

The full list of TLVs is given in Section 4.1.1.1. However, when a PNF PHY is in the IDLE state there is a list of mandatory TLVs that must be included. The FAPI-relevant list is specified in Table 3-14 and Table 3-15, and the additional nFAPI list is provided in Table 2-24.

There is no requirement for the VNF software to provide the TLVs in the order specified in the Tables.

Description	Tag
P7 VNF Address IPv4	0x0100 (if IPv4 supported)
P7 VNF Address IPv6	0x0101 (if IPv6 supported)
P7 VNF Port	0x0102
Timing window	0x011E
Timing info mode	0x011F
Timing info period	0x0120
Maximum Transmit Power	0x0128
EARFCN	0x-129

Table 2-24 Additional nFAPI TLVs included in `CONFIG.request` for IDLE and CONFIGURED states

CONFIG.response
The `CONFIG.response` message confirms the PHY state and acceptance or rejection of the `CONFIG.request` configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-25 `CONFIG.response` message body

CONFIG Errors
The error codes which may be returned in `CONFIG.response` are given in Table 2-26.

CONFIG Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `CONFIG.request` was received when the PHY was not in the IDLE state or the CONFIGURED state.
MSG_INVALID_CONFIG	The configuration provided contains parameters that are invalid or unsupported by the PHY.

Table 2-26 Error codes for CONFIG.response

2.3.3.7 START

START.request

This message is sent by the VNF when the PNF is in the PNF RUNNING state and the PHY is in the CONFIGURED state. The `START.request` message instructs the PHY to move into the RUNNING state. No message body is defined for `START.request`.

START.response

The `START.response` message confirms the PHY state and acceptance or rejection of the `START.request` configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-27 `START.response` message body

START Errors

The error codes which may be returned in `START.response` are given in Table 2-28.

START Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `START.request` was received when the PHY was not in the CONFIGURED state.

Table 2-28 Error codes for START.response

2.3.3.8 STOP

STOP.request

The `STOP.request` message moves the PHY into CONFIGURED state from the RUNNING state. No message body is defined for `STOP.request`.

STOP.response

The `STOP.response` message confirms the PHY state and acceptance or rejection of the `STOP.request` configuration.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 2-29 `STOP.response` message body

STOP Errors

The error codes which may be returned in `STOP.response` are given in Table 2-30.

STOP Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `STOP.request` was received when the PHY was not in the RUNNING state.

Table 2-30 Error codes for STOP.response

2.3.3.9 MEASUREMENT

MEASUREMENT.request

This message is sent by the VNF when the PHY is in the RUNNING state (which requires that the PNF is also in the PNF RUNNING state). The `MEASUREMENT.request` message instructs the PNF to perform one or more cell-wide measurements for the specified PHY. The measurements are requested by including the relevant TLV in the `MEASUREMENT.request` message body.

Tag	Field	Type	Description	nFAPI
0x1004	DL RS Tx power	uint16_t	The DL RS Tx power measurement defined in [15] Section 5.2.1 The value field is reserved.	TLV
0x1005	Received interference power	uint16_t	The Received interference power measurement defined in [15] Section 5.2.2. Measurement period in ms. Note: 3GPP Conformance requires this measurement to be performed over 100ms. Value: 1 → 255	TLV
0x1006	Thermal noise power	uint16_t	The Thermal noise power measurement defined in [15] Section 5.2.3. Measurement period in ms. Note: 3GPP Conformance requires this measurement to be performed over 100ms.	TLV

Tag	Field	Type	Description	nFAPI
			Value: 1 → 255	

Table 2-31 `MEASUREMENT.request` message body

MEASURMENT.response

The `MEASUREMENT.response` message returns an error code, and results for the measurement requested in `MEASUREMENT.request`.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x1007	DL RS TX Power measurement	int16_t	The DL RS Tx power measurement defined in [15] Section 5.2.1. Value is in 0.1dB steps with 0 = 0dBm	TLV
0x1008	Received Interference power measurement	struct	Received interference power measurement. See Table 2-33.	TLV
0x1009	Thermal noise power measurement	int16_t	The Thermal noise power measurement defined in [15] Section 5.2.3. Value is in 0.1dB steps with 0 = 0dBm	TLV

Table 2-32 `MEASUREMENT.response` message body

Received Interference power measurement

Field	Type	Description
Tag	uint16_t	The Received Interference power measurement Tag
Length	uint16_t	The length in bytes of the values
Number of resource blocks	uint16_t	The number of physical resource blocks included in this measurement report. Range 1->100
Received interference power	int16_t	Received interference power for each physical RB. See [15] section 5.2.2 and [16] section 10.1.3. Value is in 0.1dB steps with 0 = 0dBm

Table 2-33 Received interference power measurement parameters

MEASUREMENT Errors

The error codes which may be returned in `MEASUREMENT.response` are given in Table 2-34.

MEASUREMENT Errors

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The `MEASUREMENT.request` was received when the PHY was not in the RUNNING state.
MSG_INVALID_CONFIG	The `MEASUREMENT.request` was received had an invalid configuration.

Table 2-34 Error codes for MEASURMENT.response

2.3.4 nFAPI P7 Sync Messages

2.3.4.1 DL Node Sync

This message is sent by the VNF to the PNF PHY instance initiating a single sync procedure.

Tag	Field	Type	Description	nFAPI
N/A	t1	uint32_t	Offset from VNF SFN/SF 0/0 time reference of the DL Node Sync message transmission at the transport layer, in microseconds. Value: 0 to 10239999 in units of µs Other values are reserved	V
N/A	Delta SFN/SF	int32_t	The delta shift in subframes that the PNF PHY instance must update to on the next subframe boundary	V

Table 2-35 DL Node Sync parameters

2.3.4.2 UL Node Sync

This message is sent by the PNF PHY instance to the VNF on receipt of the DL Node Sync.

Tag	Field	Type	Description	nFAPI
N/A	t1	uint32_t	The supplied t1 field in the DL Node Sync Value: 0 to 10239999 in units of µs Other values are reserved	V
N/A	t2	uint32_t	Offset from PNF SFN/SF 0/0 time reference of the DL Node Sync message reception at the transport layer, in microseconds. Value: 0 to 10239999 in units of µs Other values are reserved	V

Tag	Field	Type	Description	nFAPI
N/A	t3	uint32_t	Offset from PNF SFN/SF 0/0 time reference of the UL Node Sync message transmission at the transport layer, in microseconds. Value: 0 to 10239999 in units of µs Other values are reserved	V

Table 2-36 UL Node Sync parameters

2.3.4.3 Timing Info

This message is sent by the PNF PHY instance to the VNF as configured through the `CONFIG.request` message.

Tag	Field	Type	Description	nFAPI
N/A	Last SFN/SF	uint32_t	The completed SFN/SF at the PNF PHY instance that triggered the Timing Info message Value: 0 to 10239999 Other values are reserved	V
N/A	Time since last Timing Info	uint32_t	The number of ms since the last Timing Info was sent from this PNF PHY instance. Value: 0 to 4294967295	V
N/A	DL Config Jitter	uint32_t	The inter message jitter of the DL Config message reception in microseconds. The method used for jitter calculation is defined in RFC 3550 Section 6.4.1 Value: 0 to 4294967295	V
N/A	Tx Request Jitter	uint32_t	The inter message jitter of the Tx Request message reception in microseconds. The method used for jitter calculation is defined in RFC 3550 Section 6.4.1 Value: 0 to 4294967295	V
N/A	UL Config Jitter	uint32_t	The inter message jitter of the UL Config message reception in microseconds. The method used for jitter calculation is defined in RFC 3550 Section 6.4.1 Value: 0 to 4294967295	V

Tag	Field	Type	Description	nFAPI
N/A	HI_DCI0 Jitter	uint32_t	The inter message jitter of the HI_DCI0 message reception in microseconds. The method used for jitter calculation is defined in RFC 3550 Section 6.4.1 Value: 0 to 4294967295	V
N/A	DL Config Latest Delay	int32_t	The latest delay offset in microseconds from the latest acceptable time for the DL Config as defined in the DL Config Timing in the PNF_PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	Tx Request Latest Delay	int32_t	The latest delay offset in microseconds from the latest acceptable time for the Tx Request as defined in the Tx Config Timing in the PNF_PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	UL Config Latest Delay	int32_t	The latest delay offset in microseconds from the latest acceptable time for the UL Config as defined in the UL Config Timing in the PNF_PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	HI_DCI0 Latest Delay	int32_t	The latest delay offset in microseconds from the latest acceptable time for the HI_DCI0 as defined in the HI_DCI0 Timing in the PNF_PARAM.Response since the last	V

Tag	Field	Type	Description	nFAPI
			transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	
N/A	DL Config Earliest Arrival	int32_t	The earlierst arrival offset in microseconds from the latest time acceptable for the DL Config as defined in the Timing Window in the PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	Tx Request Earliest Arrival	int32_t	The earlierst arrival offset in microseconds from the latest time acceptable for the Tx Request as defined in the Timing Window in the PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	UL Config Earliest Arrival	int32_t	The earlierst arrival offset in microseconds from the latest time acceptable for the UL Config as defined in the Timing Window in the PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	V
N/A	HI_DCI0 Earliest Arrival	int32_t	The earlierst arrival offset in microseconds from the latest time acceptable for the HI_DCI0 as defined in the Timing Window in the	V

Tag	Field	Type	Description	nFAPI
			PARAM.Response since the last transmission of the Timing Info Message. Note: positive value is later than acceptable, negative value is ealier than acceptable Value: -2147483648 to 2147483647	

Table 2-37 Timing Info parameters

2.3.5 nFAPI Tag Values

The tag values used in the TLV sections of nFAPI messages above are given in Table 2-38, except for those used in the PHY Config and Params messages which are shared with FAPI, which are given in Table 4-3.

Tag (Hex)	Description
0x1000	PNF Param General
0x1001	PNF PHY
0x1002	PNF RF
0x1003	PNF PHY_RF Config
0x1004	DL RS Tx power
0x1005	Received interference power
0x1006	Thermal noise power
0x1007	DL RS TX Power measurement
0x1008	Received Interference power measurement
0x1009	Thermal noise power measurement
0x100a	PNF PHY Rel 10
0x100b	PNF PHY Rel 11
0x100c	PNF PHY Rel 12
0x100d	PNF PHY Rel 13
0x100e	PNF PHY Rel 13 NB-IOT

Table 2-38 nFAPI tag values

3. FAPI (P5 & P7)

3.1.1 Introduction

The FAPI L1 API, defined in this document, resides within the eNB component with Figure 3-1 showing the protocol model for the eNB defined in the E-UTRAN architectural standard [7]. It highlights the separation of control- and data-plane information, which is maintained throughout the LTE network. Both control- and data-plane information is passed through the L1 API, however, each API message contains either control- or data-plane information, but never both.

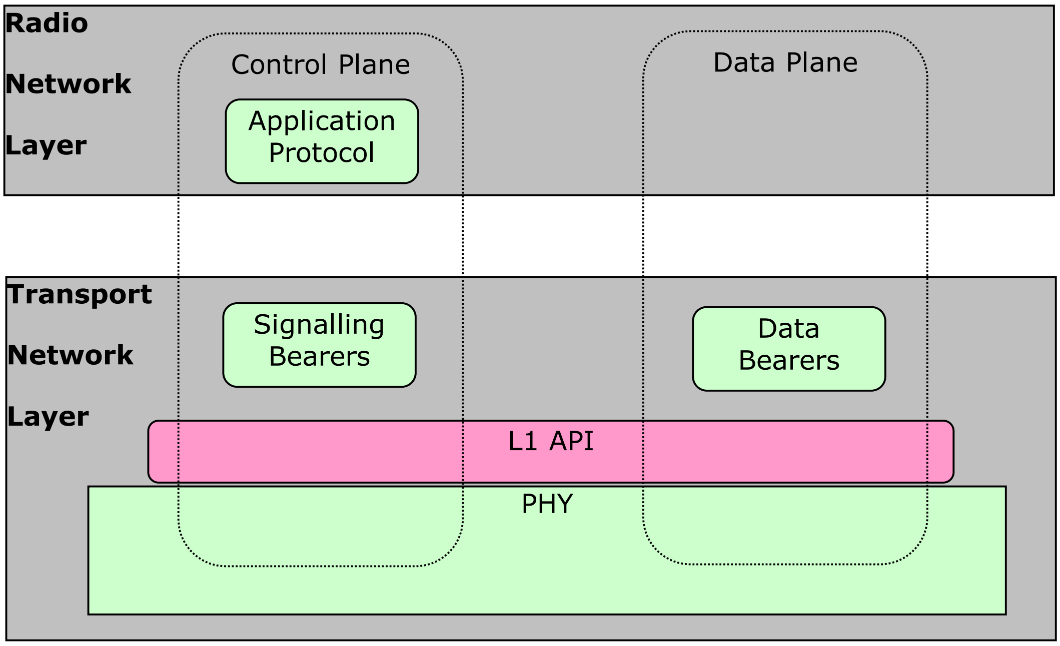


Figure 3-1 E-UTRAN protocol model

Figure 3-2 provides an example of how the different L2/L3 protocol layers will interact with the L1 API. In this example, a PHY control entity is responsible for configuration procedures (P5). The MAC layer is responsible for the exchange of data-plane messages with the PHY (P7). The PHY configuration sent over the P5 interface may be determined using SON techniques, information model parameters sent from the HeMS [14], or a combination of both methods. If carrier aggregation is supported then one instance of the L1 API exists for each carrier.

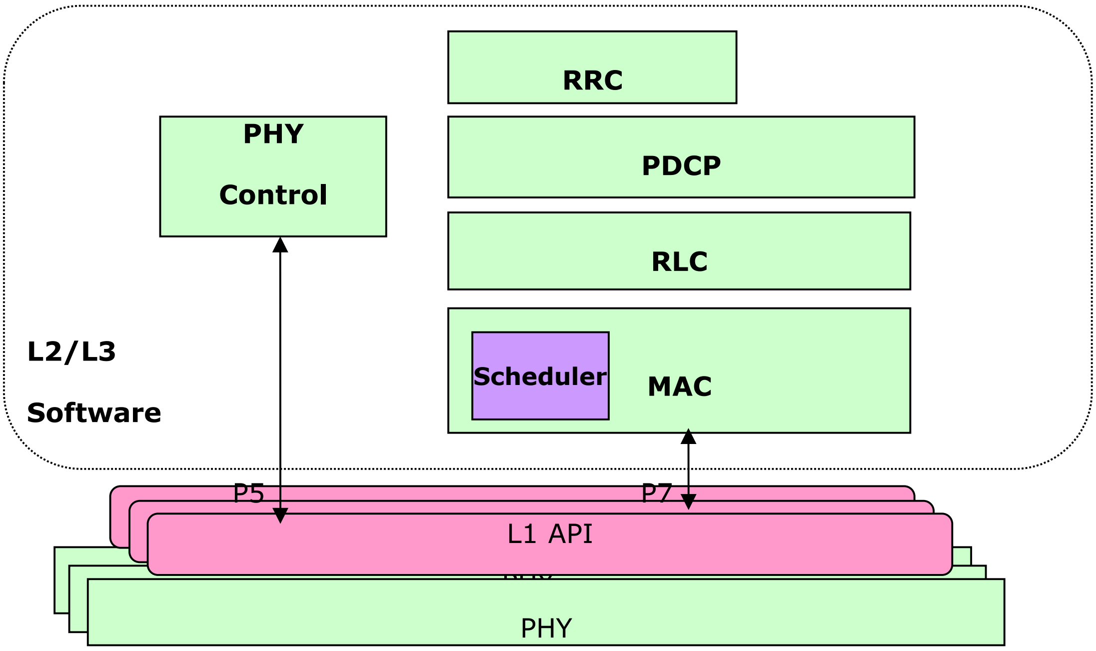


Figure 3-2 L1 API interactions

3.2 FAPI procedures

This section gives an overview of the procedures which use the L1 API. These procedures are split into two groups, namely, configuration procedures and subframe procedures. Configuration procedures handle the management of the PHY layer and are expected to occur infrequently. Subframe procedures determine the structure of each 1ms subframe and operate with a 1ms periodicity.

3.2.1 P5 Configuration procedures

The configuration procedures supported by the L1 API are:

- Initialization
- Termination
- Restart
- Reset
- Error notification

These procedures will move the PHY layer through the IDLE, CONFIGURED and RUNNING states, as shown in Figure 3-3. A list of the L1 API configuration messages which are valid in each state is given in Table 3-1.

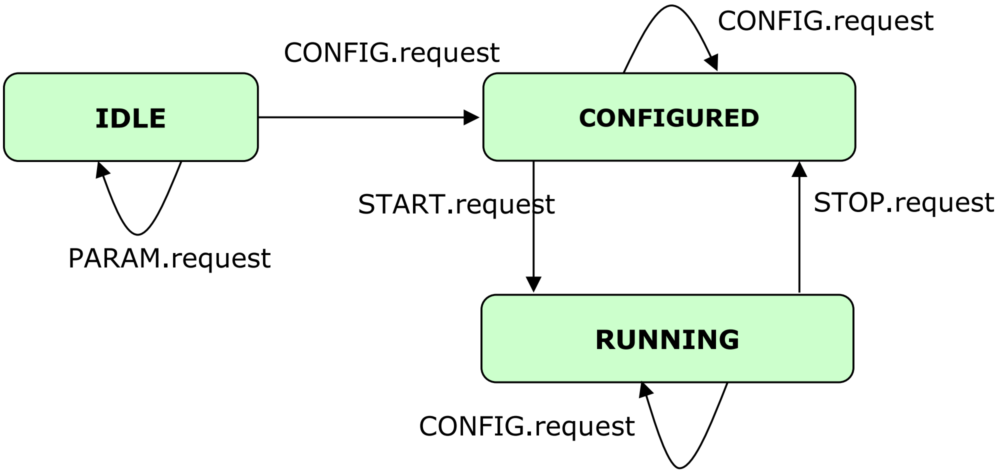


Figure 3-3 PHY layer state transactions on L1 API configuration messages

Idle State	Configured State	Running State
PARAM.request	PARAM.request	CONFIG.request
CONFIG.request	CONFIG.request	STOP.request
	START.request	

Table 3-1 L1 API configuration messages valid in each PHY state

3.2.1.1 Initialization

The initialization procedure moves the PHY from the IDLE state to the RUNNING state, via the CONFIGURED state. An overview of this procedure is given in Figure 3-4, the different stages are:

- The PARAM message exchange procedure
- The CONFIG message exchange procedure
- The START message exchange procedure

The initialization procedure is completed when the PHY sends the L2/L3 software a `SUBFRAME.indication` message.

The remainder of this section describes the PARAM, CONFIG and START message exchange procedures.

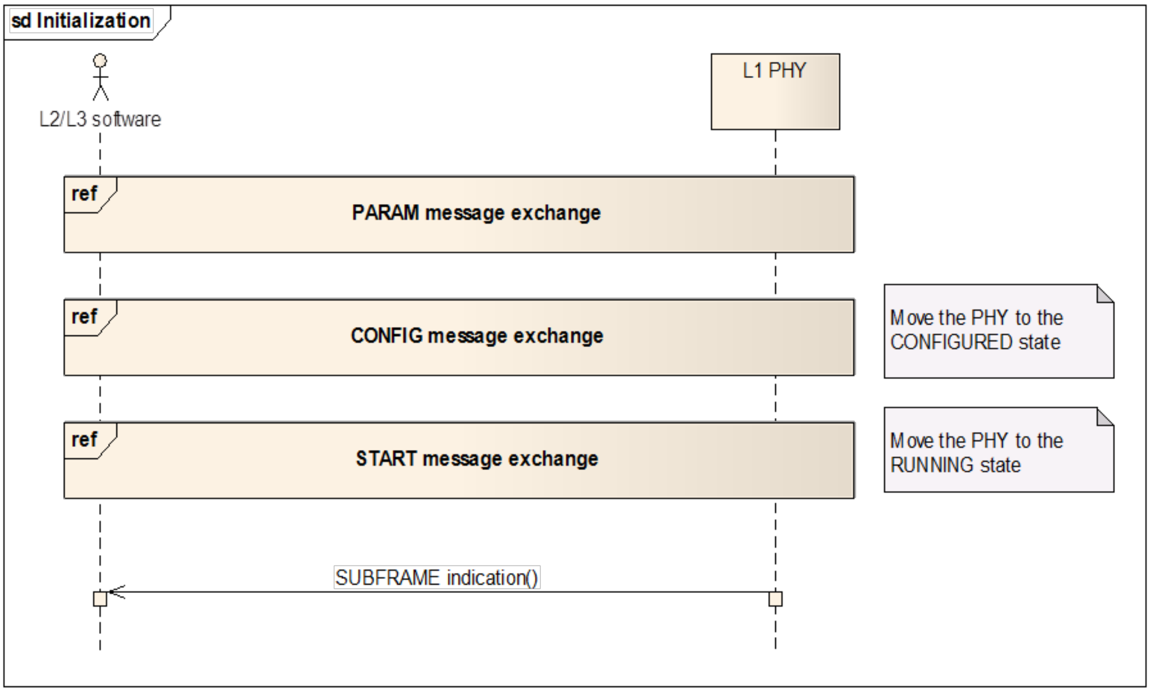


Figure 3-4 Initialization procedure

The PARAM message exchange procedure is shown in Figure 3-5. Its purpose is to allow the L2/L3 software to collect information about the PHY configuration and current state. The information returned by the PHY depends on its state, and is described inTable 3-2. The PARAM message exchange is optional.

PHY state	Information returned by PHY
IDLE	The PHY indicates which capabilities it supports
CONFIGURED	The PHY returns its current configuration
RUNNING	The PHY returns invalid state

Table 3-2 Information returned by the PHY during a PARAM message exchange

From Figure 3-5 it can be seen that the PARAM message exchange procedure is initiated by the L2/L3 software sending a `PARAM.request` message to the PHY. It is recommended that the L2/L3 software starts a guard timer to wait for the response from the PHY. If the PHY is operating correctly it will return a `PARAM.response` message. In the IDLE and CONFIGURED states this message will include the current PHY state and a list of configuration information, as described in Table 3-2. In the RUNNING state this message will indicate an INVALID_STATE error, to determine the PHY capabilities it must be moved to the CONFIGURED state using the termination procedure. If the guard timer expires before the PHY responds this indicates the PHY is not operating correctly. This must be rectified before further L1 API commands are used; the rectification method is outside the scope of this document.

The CONFIG message exchange procedure is shown in Figure 3-6. Its purpose is to allow the L2/L3 software to configure the PHY. It can be used when the PHY is in any state. The procedure has slight differences depending on the PHY state; for clarity each case is described separately.

If the PHY is in the IDLE state the `CONFIG.request` message, sent by the L2/L3 software, must include all mandatory TLVs. The mandatory TLVs are highlighted later in Section 3.3.2.2. If all mandatory TLVs are included, and set to values supported by the PHY, L1 will return a `CONFIG.response` message indicating it is successfully configured and has moved to the CONFIGURED state. If the `CONFIG.request` message has missing mandatory TLVs, invalid TLVs, or unsupported TLVs, the PHY will return a `CONFIG.response` message indicating an incorrect configuration. In this case, it will remain in the IDLE state and all received TLVs will be ignored.

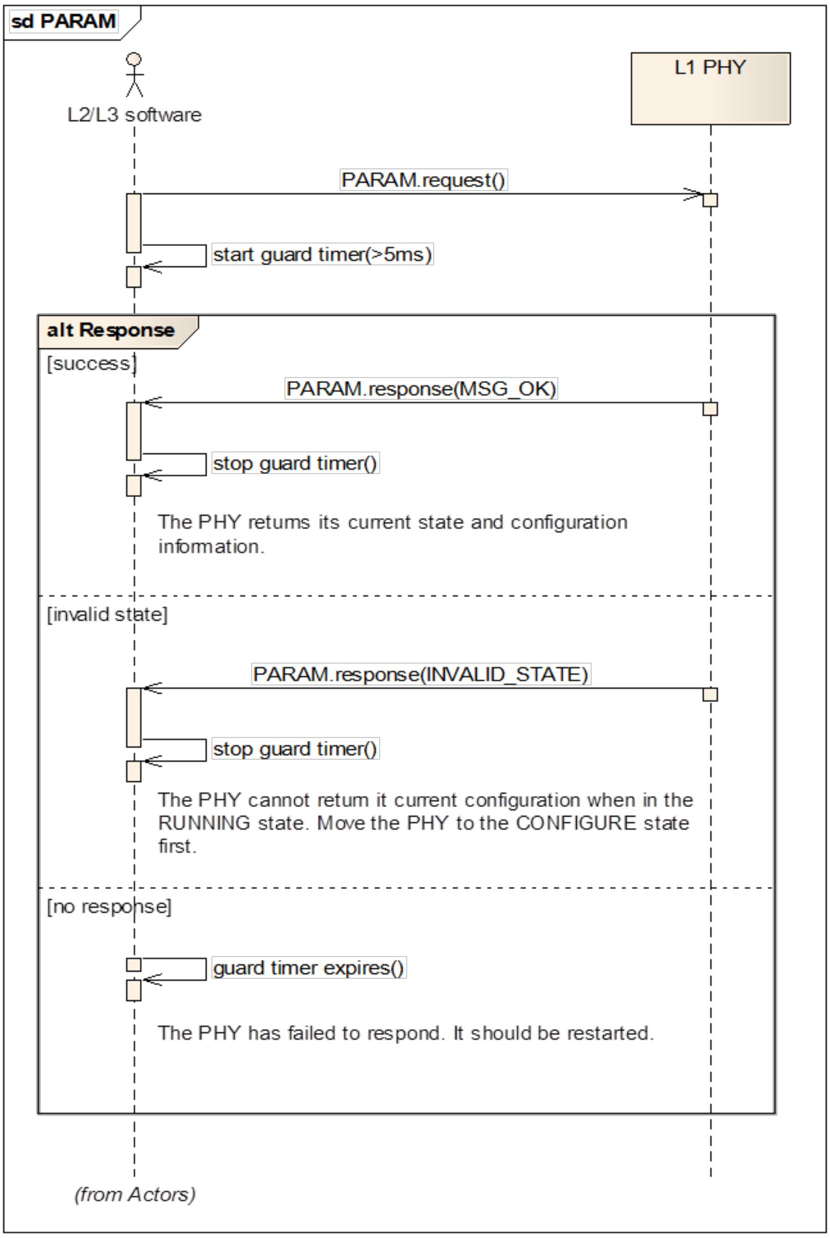


Figure 3-5 PARAM message exchange

If the PHY is in the CONFIGURED state the `CONFIG.request` message, sent by the L2/L3 software, may include only the TLVs that are required to change the PHY to a new configuration. If the PHY supports these new values, it will return a `CONFIG.response` message indicating it has been successfully configured. However, if the `CONFIG.request` message includes invalid TLVs, or unsupported TLVs, the PHY will return a `CONFIG.response` message indicating an incorrect configuration. In this case all received TLVs will be ignored and the PHY will continue with its previous configuration. In both cases, if the PHY receives a `CONFIG.request` while in the CONFIGURED state it will remain in the CONFIGURED state.

If the PHY is in the RUNNING state then a limited subset of CONFIG TLVs may be sent in a `CONFIG.request` message. The permitted TLVs are highlighted later in Section 3.3.2.2. If the `CONFIG.request` message has invalid TLVs, or TLVs which must not be reconfigured in the RUNNING state, the PHY will return a `CONFIG.response` message indicating an incorrect configuration. In this case, it will remain in the RUNNING state and all received TLVs will be ignored.

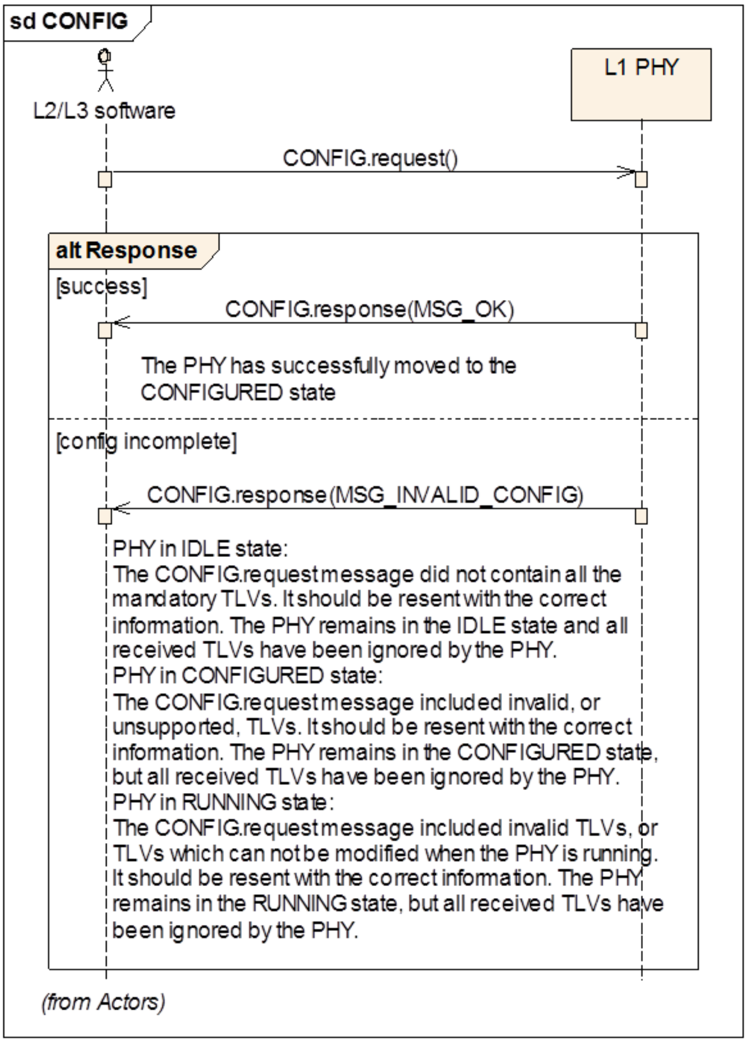


Figure 3-6 CONFIG message exchange

The START message exchange procedure is shown in Figure 3-7. Its purpose is to instruct a configured PHY to start transmitting as an eNB. The L2/L3 software initiates this procedure by sending a `START.request` message to the PHY. If the PHY is in the CONFIGURED state, it will issue a SUBFRAME indication. After the PHY has sent its first `SUBFRAME.indication` message it enters the RUNNING state.

If the PHY receives a `START.request` in either the IDLE or RUNNING state it will return an `ERROR.indication` including an INVALID_STATE error.

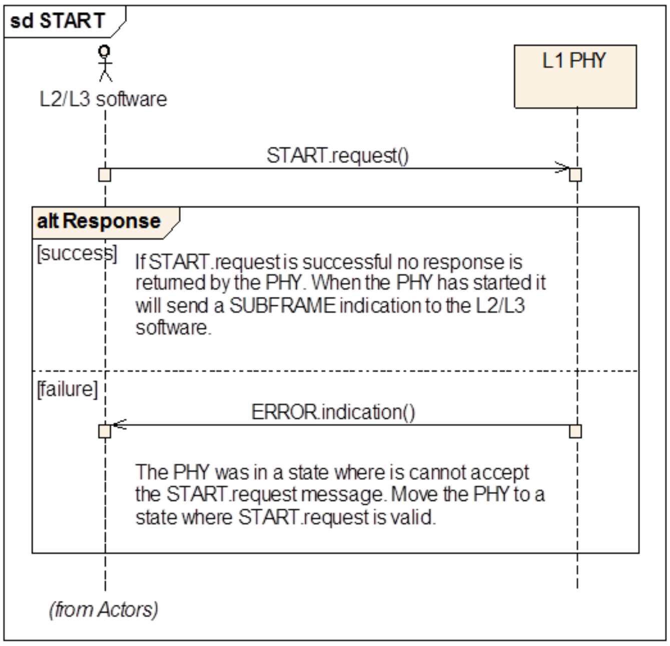


Figure 3-7 START message exchange

3.2.1.2 Termination

The termination procedure is used to move the PHY from the RUNNING state to the CONFIGURED state. This stops the PHY transmitting as an eNB. The termination procedure is shown in Figure 3-8 and initiated by the L2/L3 software sending a `STOP.request` message.

If the `STOP.request` message is received by the PHY while operating in the RUNNING state, it will stop all TX and RX operations and return to the CONFIGURED state. When the PHY has completed its stop procedure a `STOP.indication` message is sent to the L2/L3 software.

If the `STOP.request` message was received by the PHY while in the IDLE or CONFIGURED state, it will return an `ERROR.indication` message including an INVALID_STATE error. However, in this case the PHY was already stopped.

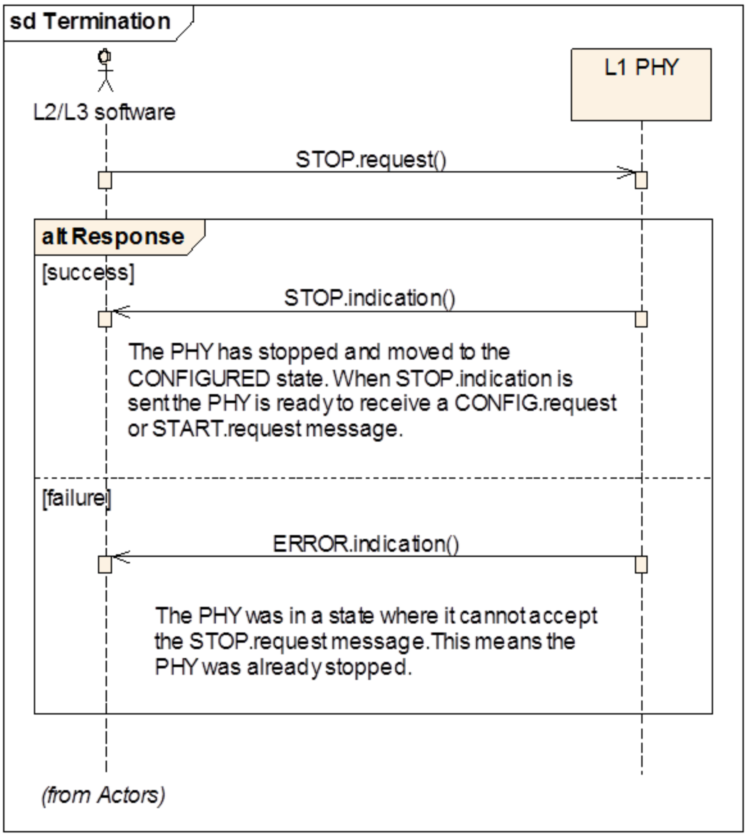


Figure 3-8 Termination procedure

3.2.1.3 Restart

The restart procedure is shown in Figure 3-9. It can be used by the L2/L3 software when it needs to stop transmitting, but later wants to restart transmission using the same configuration. To complete this procedure the L2/L3 software can follow the STOP message exchange shown in Figure 3-8. This moves the PHY to the CONFIGURED state. To restart transmission it should follow the START message exchange, shown in Figure 3-7, moving the PHY back to the RUNNING state.

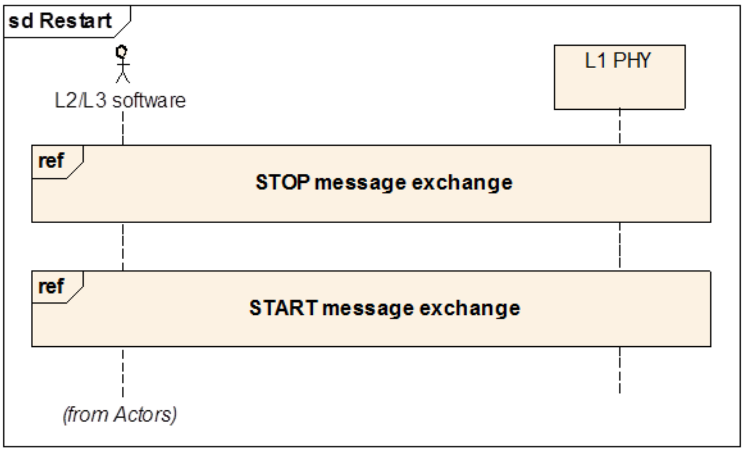


Figure 3-9 Restart procedure

3.2.1.4 Reset

The reset procedure is shown in Figure 3-10. This procedure is used when the L2/L3 software wants to return the PHY to the IDLE state. This can only be achieved by terminating the PHY (as shown in Figure 3-8) and then resetting the PHY. The method for resetting the PHY will be implementation specific and is outside the scope of this document.

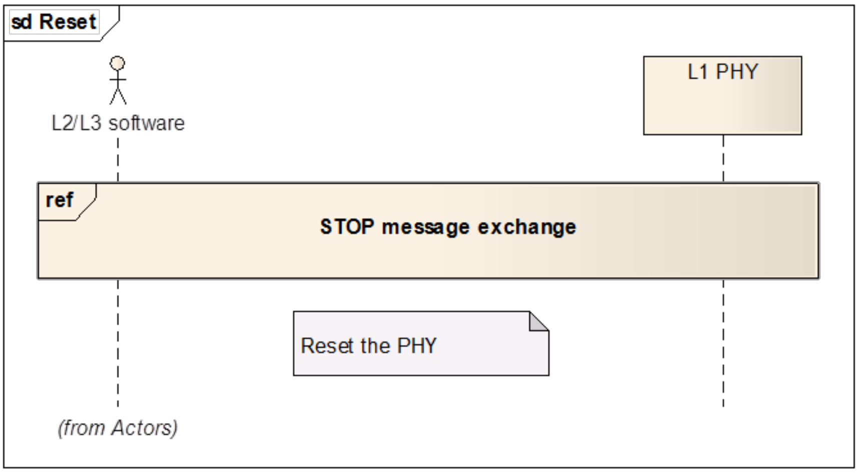


Figure 3-10 Reset procedure

3.2.1.5 Reconfigure

Two methods of reconfiguration are supported by the PHY. A major reconfiguration where the PHY is stopped, and a minor reconfiguration where the PHY continues running.

The major reconfigure procedure is shown in Figure 3-11. It is used when the L2/L3 software wants to make significant changes to the configuration of the PHY. The STOP message exchange, shown in Figure 3-8, is followed to halt the PHY and move it to the CONFIGURED state. The CONFIG message exchange, shown in Figure 3-6, is used to reconfigure the PHY. Finally, the START message exchange, shown in Figure 3-7, is followed to start the PHY and return it to the RUNNING state.

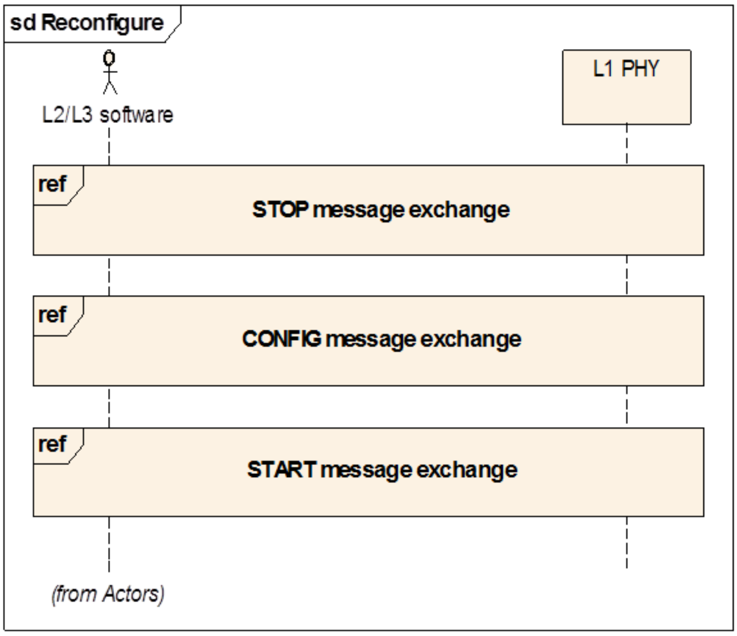


Figure 3-11 Major reconfiguration procedure

The minor reconfiguration procedure is shown in Figure 3-12. It is typically used in conjunction with an RRC system information update.

In the subframe where the L2/L3 software requires the configuration change, it sends the `CONFIG.request` message to the PHY. Only a limited subset of CONFIG TLVs may be sent; these are highlighted later in Section 3.3.2.2. TLVs included in the `CONFIG.request` message for subframe N will be applied at the SFN/SF given in the `CONFIG.request` message. Reconfiguring the PHY while in the RUNNING state has a further restriction, the `CONFIG.request` message must be sent before the `DL_CONFIG.request` and `UL_CONFIG.request` message.

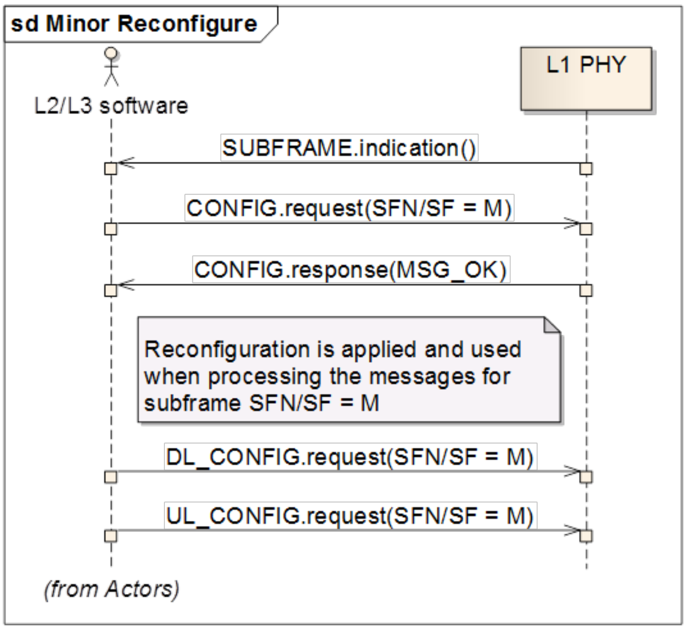


Figure 3-12 Minor reconfigure procedure

3.2.1.6 Query

The query procedure is shown in Figure 3-13. It is used by the L2/L3 software to determine the configuration and operational status of the PHY. The PARAM message exchange, shown in Figure 3-5, is used. This signalling sequence can be followed when the PHY is stopped, in the IDLE state and, optionally, the CONFIGURED state.

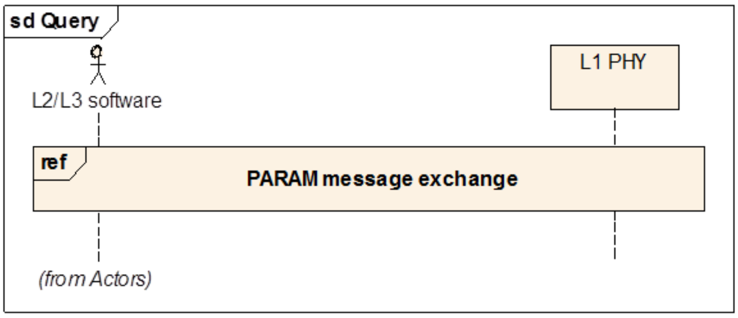


Figure 3-13 Query procedure

3.2.1.7 Notification

The notification procedure is shown in Figure 3-14. The PHY sends a notification message when it has an event of interest for the L2/L3 software. Currently, there is one notification message called `ERROR.indication`.

The `ERROR.indication` message has already been mentioned in multiple procedures. It is used by the PHY to indicate that the L2/L3 software has sent invalid information to the PHY.

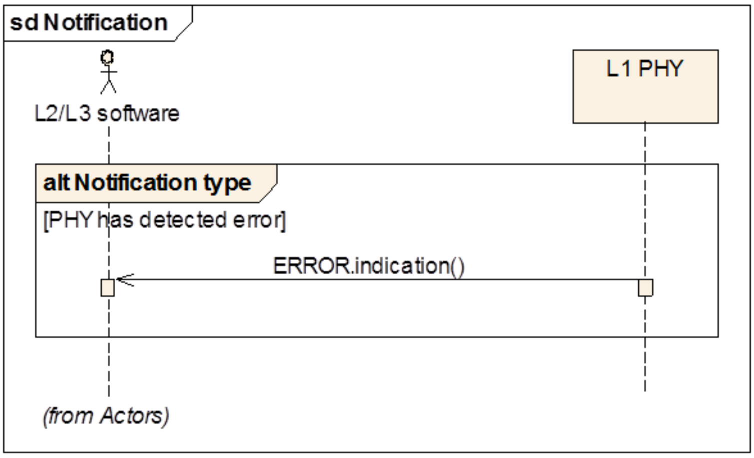


Figure 3-14 Notification procedures

3.2.2 P7 Subframe procedures

The subframe procedures have two purposes. Firstly, they are used to control the DL and UL frame structures. Secondly, they are used to transfer the subframe data between the L2/L3 software and PHY. The subframe procedures supported by the L1 API are:

- Transmission of a 1ms SUBFRAME message
- Synchronization of SFN/SF between the L2/L3 software and PHY
- Transmission of the BCH transport channel
- Transmission of the PCH transport channel

- Transmission of the DLSCH transport channel and reception of ACK/NACK response
- Transmission of the MCH transport channel
- Reception of the RACH transport channel
- Reception of the ULSCH transport channel and transmission of ACK/NACK response
- Reception of the sounding reference signal
- Reception of CQI and RI reporting
- Reception of scheduling request information

3.2.2.1 SUBFRAME signal

A `SUBFRAME.indication` message is sent from the PHY, to the L2/L3 software, indicating the start of a 1ms subframe.

The periodicity of the `SUBFRAME.indication` message for TDD (frame structure 2) is shown in Figure 3-15 and Figure 3-16.

In TDD two frame structures are possible, one with 5ms switch points and one with 10ms switch points [6]. The `SUBFRAME.indication` message is generated for every subframe (DL or UL).

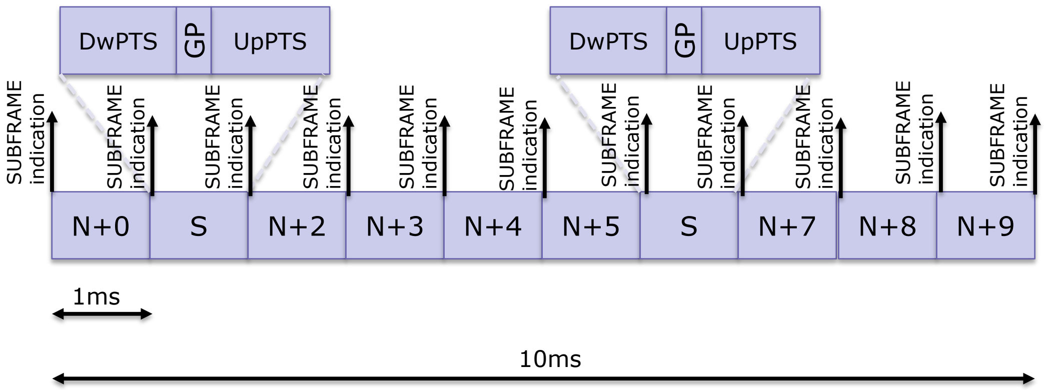


Figure 3-15 SUBFRAME signal for TDD using 5ms switch points

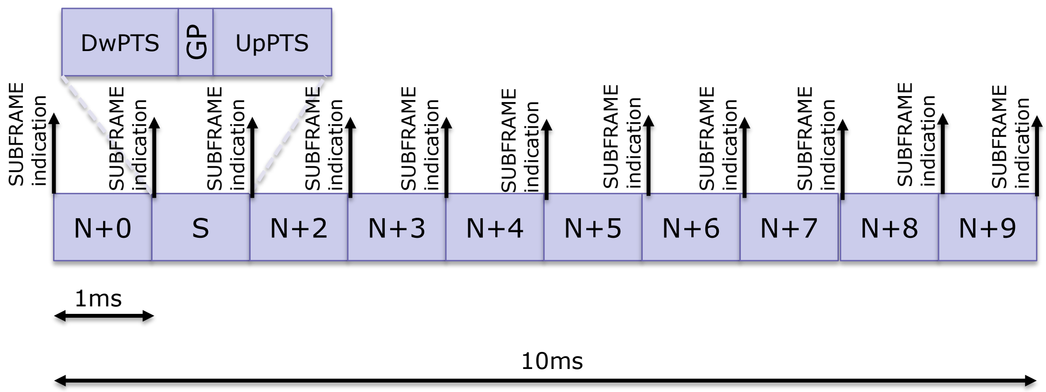


Figure 3-16 SUBFRAME signal for TDD using 10ms switch point

The periodicity of the `SUBFRAME.indication` message for FDD (frame structure 1) is shown in Figure 3-17. The subframe indication is generated for every DL subframe.

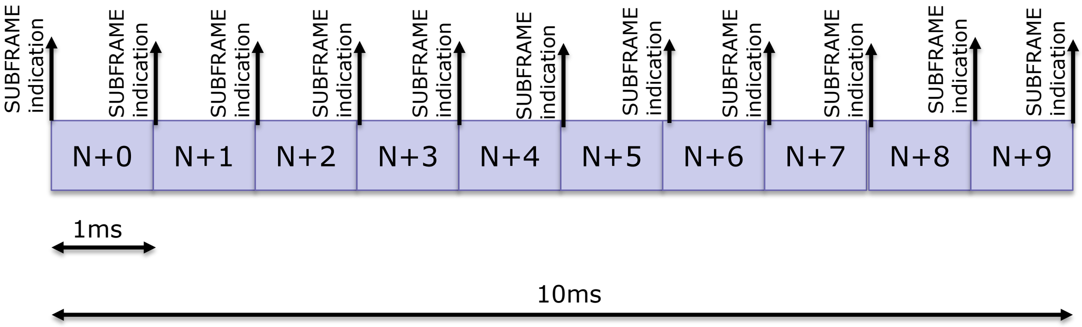


Figure 3-17 SUBFRAME signal for FDD

3.2.2.2 SFN/SF synchronization

The SFN/SF synchronization procedure is used to maintain a consistent SFN/SF value between the L2/L3 software and the PHY. Maintaining this synchronization is important since different subframes have different structures, and in TDD subframes are either downlink or uplink.

Two options are provided by the L1 API; the first option configures the PHY to use the SFN/SF value provided by the L2/L3 software. The second option configures the PHY to initialize the SFN/SF and ensure the L2/L3 software remains synchronous. The synchronization option is selected at compile time. For each option two procedures are described, the initial start-up synchronization and the maintenance of the synchronization.

L2/L3 software is master

The SFN/SF synchronization start-up procedure, where the L2/L3 software is master, is given in Figure 3-18. The start-up procedure followed is:

- After successful configuration the L2/L3 software sends a `START.request` message to move the PHY to the RUNNING state
- When the L2/L3 software is configured as master the initial PHY SFN/SF = M, where M could be any value. In the `SUBFRAME.indication` message, SFN/SF = M
- The L2/L3 software sends a `DL_CONFIG.request` message to the PHY containing the correct SFN/SF = N
- The PHY uses the SFN/SF received from the L2/L3 software. It changes its internal SFN/SF to match the value provided by the L2/L3 software

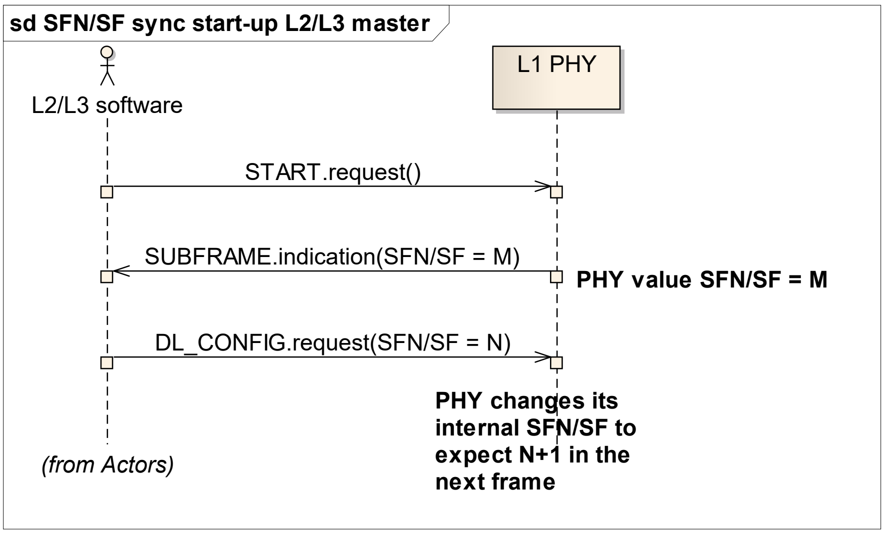


Figure 3-18 SFN/SF synchronization start-up with L2/L3 master

The SFN/SF synchronization maintenance procedure is shown in Figure 3-19. In this example, the L1 PHY is expecting the next `DL_CONFIG.request` to contain information regarding frame M. The procedure followed is:

- The PHY sends the `SUBFRAME.indication` message with SFN/SF = M.
- The L2/L3 software sends a `DL_CONFIG.request` message to the PHY containing SFN/SF = N
- If SFN/SF M = N
    - The PHY received the SFN/SF it was expecting. No SFN/SF synchronization is required
- If SFN/SF M ≠ N
    - The PHY received a different SFN/SF from the expected value. SFN/SF synchronization is required
    - The PHY uses the SFN/SF received from the L2/L3 software. It changes its internal SFN/SF to match the value provided by the L2/L3 software
    - The PHY returns an `ERROR.indication` message indicating the mismatch

This SFN/SF synchronization procedure assumes the L2/L3 software is always correct. However, it is possible that the SFN/SF synchronization was unintended, and due to a L2/L3 software issue. The generation of an `ERROR.indication` message, with expected and received SFN/SF values, should allow the L2/L3 software to perform a correction with a further SFN/SF synchronization.

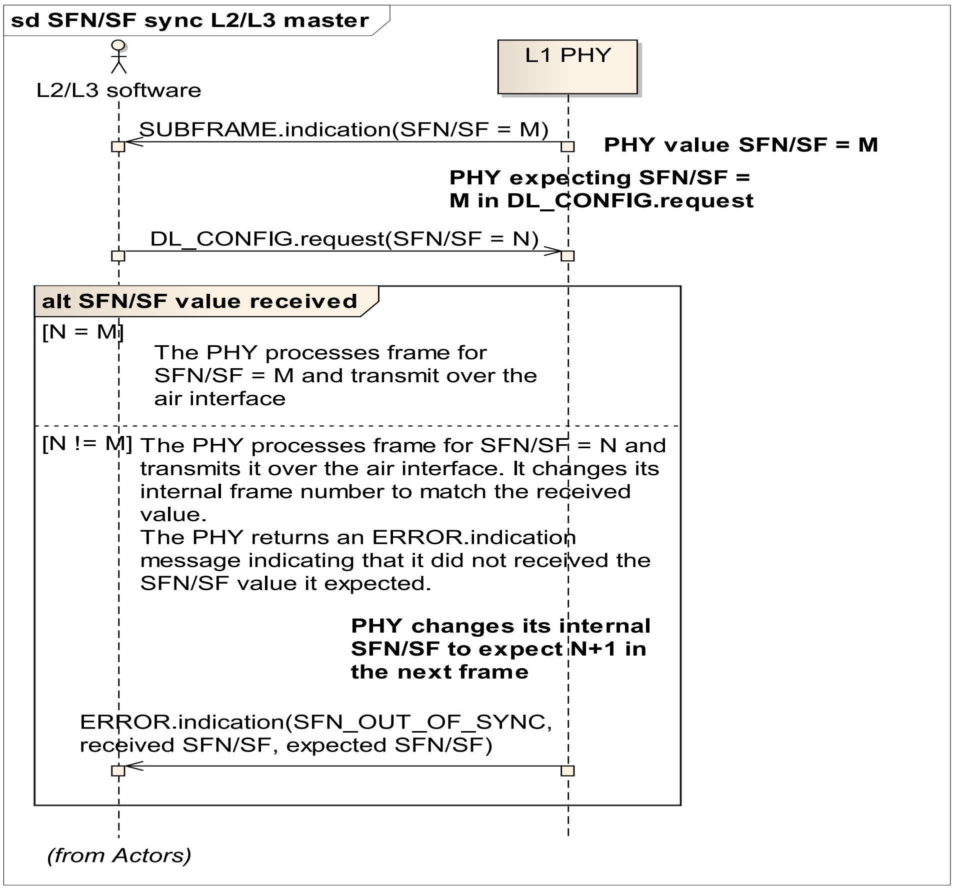


Figure 3-19 SFN/SF synchronization with L2/L3 master

L1 PHY is master

The SFN/SF synchronization start-up procedure, where the L1 software is master, is given in Figure 3-20. The start-up procedure followed is:

- After successful configuration the L2/L3 software sends a `START.request` message to move the PHY to the RUNNING state
- If the L1 software is configured as master the initial PHY SFN/SF = M. The value of M is not deterministic, and could have been set by an external mechanism, such as GPS. The PHY sends a `SUBFRAME.indication` message to the L2/L3 software, with SFN/SF = M. The L2/L3 software uses the SFN/SF received from the PHY. It changes its internal SFN/SF to match the value provided by the PHY
- The L2/L3 software sends a `DL_CONFIG.request` message to the PHY containing SFN/SF = M

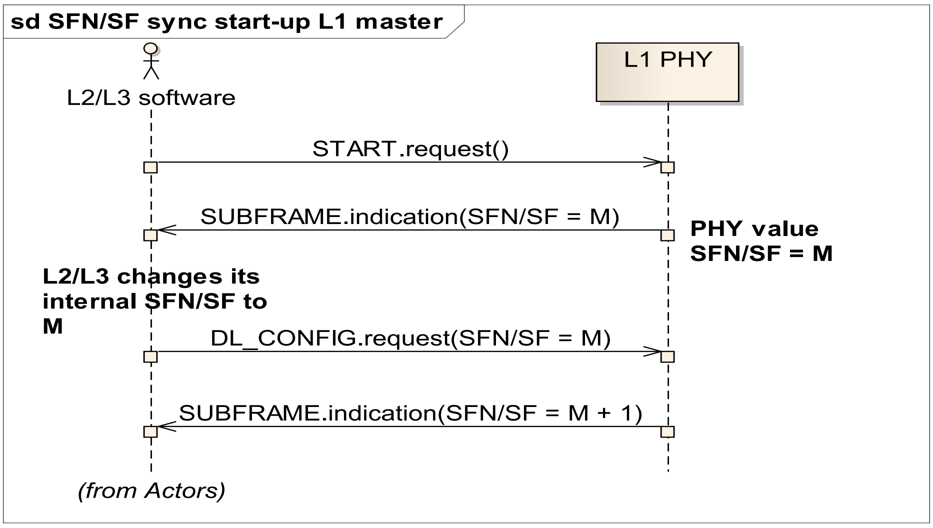


Figure 3-20 SFN/SF synchronization start-up with L1 master

The SFN/SF synchronization maintenance procedure is shown in Figure 3-21. In this example, the L1 PHY is expecting the next `DL_CONFIG.request` to contain information regarding frame M. The procedure followed is:

- The PHY sends a `SUBFRAME.indication` message to the L2/L3 software, with SFN/SF = M
- The L2/L3 software sends a `DL_CONFIG.request` message to the PHY containing SFN/SF = N
- If SFN/SF M = N
    - The PHY received the SFN/SF it was expecting. No SFN/SF synchronization is required
- If SFN/SF M ≠ N
    - The PHY received a different SFN/SF from the expected value. SFN/SF synchronization is required
    - The PHY discards the received `DL_CONFIG.request` message
    - The PHY returns an `ERROR.indication` message indicating the mismatch

This SFN/SF synchronization procedure will continue to discard `DL_CONFIG.request` messages and emit `ERROR.indication` messages until the L2/L3 software corrects its SFN/SF value.

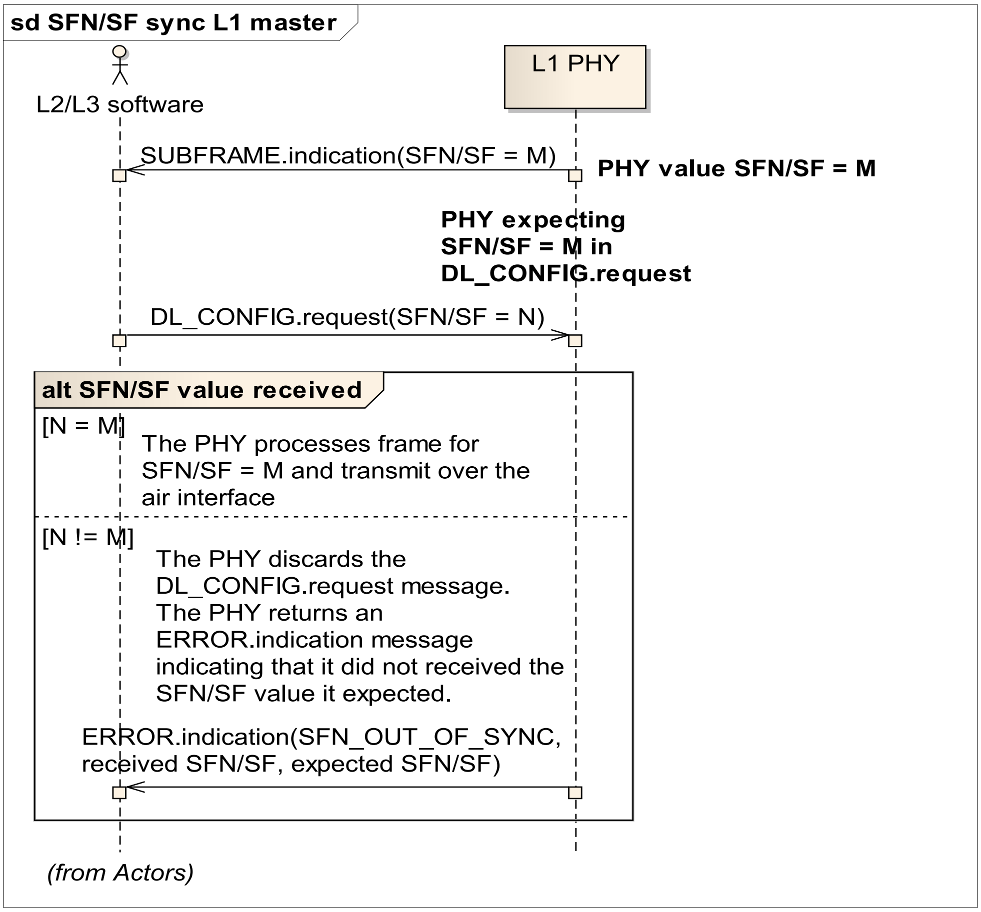


Figure 3-21 SFN/SF synchronization with L1 master

3.2.2.3 API message order

The L1 API has constraints on when certain subframe messages can be sent, or will be received, by the L2/L3 software.

The downlink API message constraints are shown in Figure 3-22:

- The SFN/SF included in the `SUBFRAME.indication` message is expected in the corresponding `DL_CONFIG.request`
- The SFN/SF included in the SUBFRAME.indication message with the offset defined in the LAA config section of the CONFIG.request, is expected in the corresponding LBT_DL_CONFIG.request
- If the PHY is being reconfigured using the `CONFIG.request` message, this must be the first message for the subframe.
- If the PHY is being reconfigured using the `UE_CONFIG.request` message, this must be the next message for the subframe.
- The `DL_CONFIG.request` must be sent for every downlink subframe and must be the next message.
- The `UL_CONFIG.request` must be sent for every uplink subframe and must be the next message.
- The LBT_DL_CONFIG.request is optional and is not a requirement to be sent every subframe.
- The `TX.request` and `HI_DCI0.request` messages are optional. It is not a requirement that they are sent in every downlink subframe.

- There must be only 1 DL_CONFIG.request, 1 UL_CONFIG.request, 1 LBT_DL_CONFIG.request, 1 HI_DCI0.request and 1 TX.request for a subframe.
- There must be only 1 `LBT_DL_CONFIG.request` for an LBT process (an LBT process may contain several subframes)

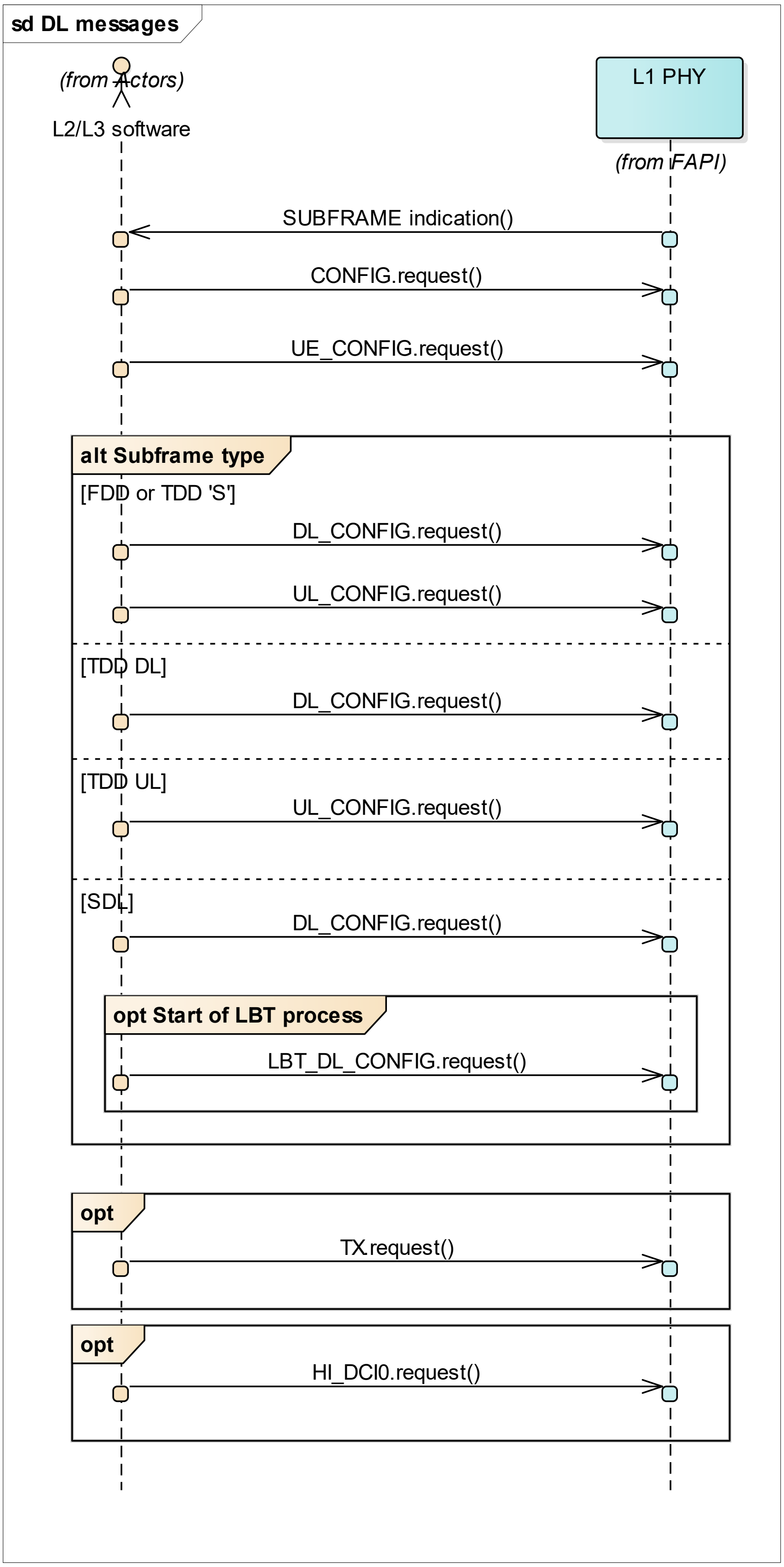


Figure 3-22 DL message order

The uplink API message constraints are shown in Figure 3-23:

- The UL API messages are optional. It is not a requirement that they are sent in every subframe.
- If present, the messages can be in any order
    - The `HARQ.indication` message is included if ACK/NACK responses were expected in the subframe.
    - The `CRC.indication` message is included if uplink data PDUs were expected in the subframe.
    - The `RX_ULSCH.indication` message is included if uplink data PDUs were expected in the subframe.
    - The `RX_SR.indication` message is included if SR PDUs were expected in the subframe.
    - The `RX_CQI.indication` message is included if CQI were expected in the subframe.
    - The `RACH.indication` message is included if any RACH preambles were detected in the subframe
    - The `SRS.indication` message is included if any sounding reference symbol information is expected in the subframe.
    - `The LBT_DL.indication` message is included during the subframes containing the LBT process.
- There will be only 1 `HARQ.indication`, 1 `CRC.indication`, 1 `RX_ULSCH.indication`, 1 `RX_SR.indication`, 1 `RX_CQI.indication`, 1 `RACH.indication`, 1 `SRS.indication` and 1 `LBT_DL.indication` message per subframe

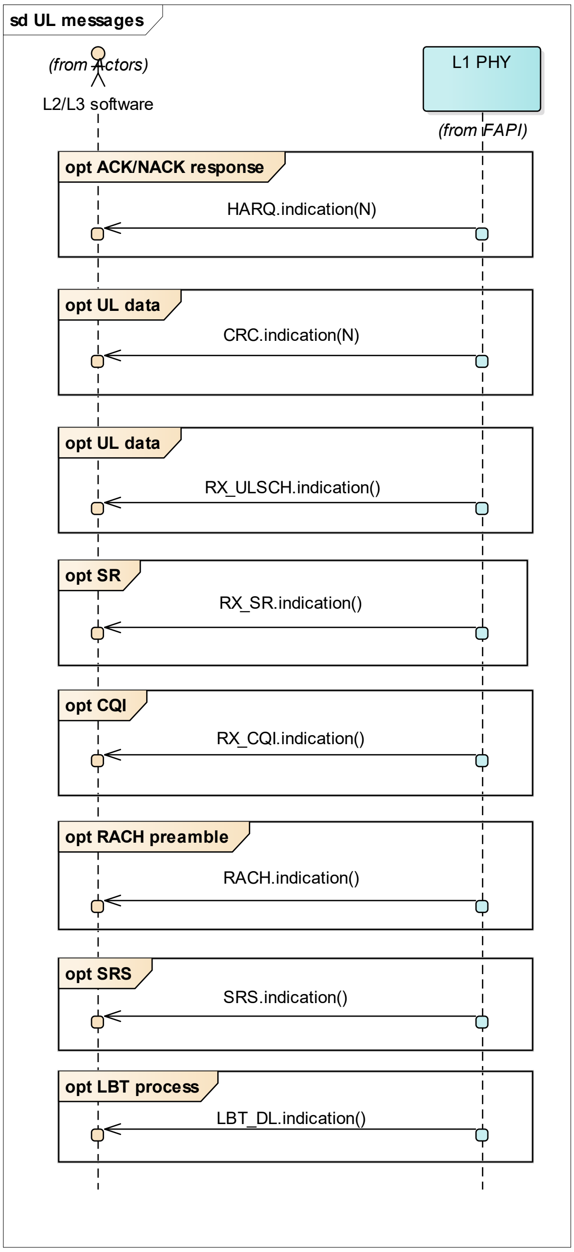


Figure 3-23 UL message order

3.2.2.4 Semi-static information

In LTE the majority of uplink and downlink data is determined by the scheduler on a subframe-by-subframe basis, however, there are several semi-static parameters which create periodic transmission patterns on the uplink. These semi-static parameters are either cell-specific or UE-specific. The cell-specific parameters are RACH and SRS regions which occur regularly and have a predefined pattern advertised on system information messages. The UE-specific parameters are CQI reporting, SR opportunities and SRS reporting. These are sent to the UE in RRC connection messages.

This L1 API supports the storage of the semi-static parameters in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when to allocate RACH and SRS regions, and when to expect the UE to transmit CQI, SR and SRS. If stored in the PHY semi-static parameters are passed from the MAC to PHY with the message exchange shown in Figure 3-24. A UE is released with the message exchange shown in Figure 3-25.

It is not expected that this parameter will be configurable; instead it will be a characteristic of the PHY.

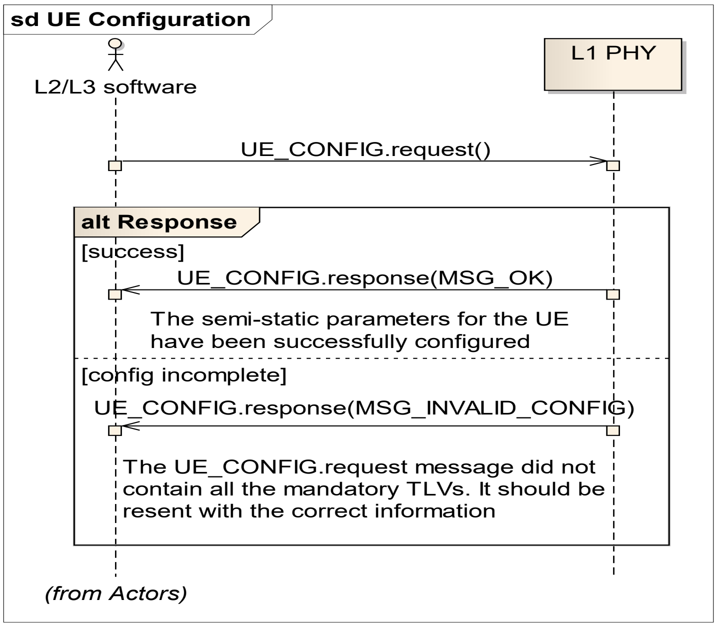


Figure 3-24 UE configuration procedure

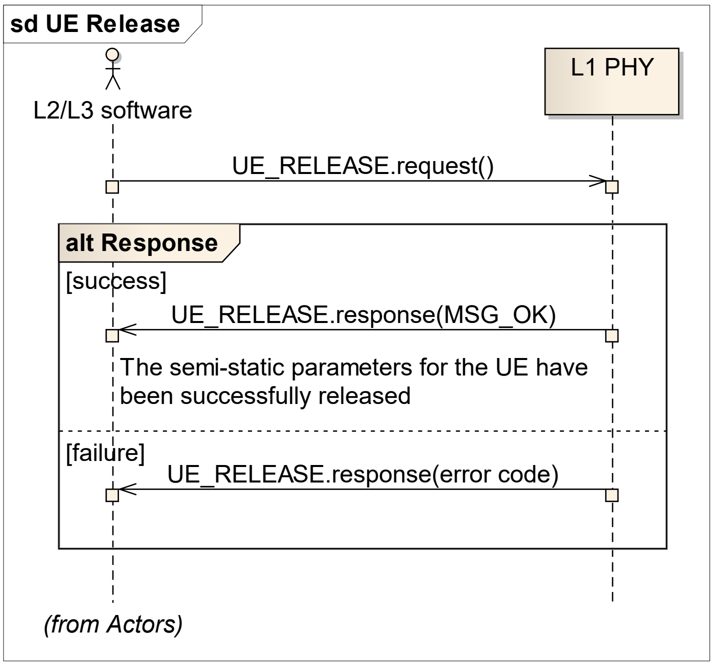


Figure 3-25 UE release procedure

3.2.2.5 Uplink HARQ signalling

Uplink HARQ signalling is used to acknowledge, or negatively acknowledge, downlink data transmissions sent to a UE. For FDD, the PUCCH location of this HARQ signalling is determined by the n_{CCE} value of the DCI which allocated the downlink grant. For

TDD, the PUCCH location of this HARQ signalling is determined by a combination of the n_{CCE} value of the granting DCI and the DL-UL subframe configuration. Both the eNB and the UE need to calculate the HARQ location based on this information. In the eNB this calculation could be performed in either the MAC or PHY.

This L1 API supports the calculation of the uplink HARQ location in either the MAC or PHY. If the calculation is performed in the MAC the L1 API is used to instruct the PHY when and where to receive the HARQ. If stored in the PHY HARQ parameters are passed from the MAC to PHY with the message exchange shown in Figure 3-24.

It is not expected that this parameter will be configurable; instead it will be a characteristic of the PHY.

3.2.2.6 Downlink

The procedures relating to downlink transmission are described in this Section.

BCH

The BCH transport channel is used to transmit the Master Information Block (MIB) information to the UE, unless LAA is configured. The location of the MIB is defined in the LTE standards [4], and shown in Figure 3-26. It is transmitted in subframe 0 of each radio frame. When the radio frame (SFN mod 4) = 0 an updated MIB is transmitted in subframe 0. When the radio frame (SFN mod 4) ≠ 0 the MIB is repeated.

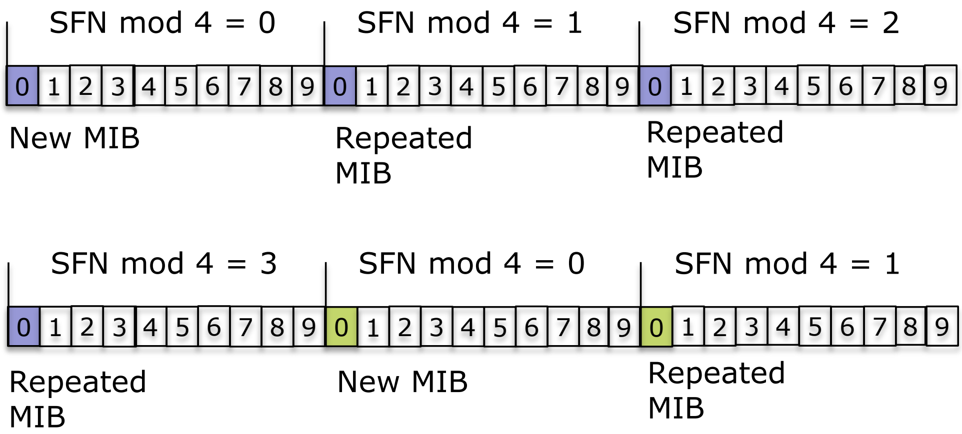


Figure 3-26 MIB scheduling on the BCH transport channel

The BCH procedure is shown in Figure 3-27. The L2/L3 software should provide a BCH PDU to the PHY in subframe SF=0, for each radio frame (SFN mod 4) = 0. This is once every 40ms. The L2/L3 software provides the following information:

- In `DL_CONFIG.request` a BCH PDU is included.
- In `TX.request` a MAC PDU containing the MIB is included.

If the PHY does not receive a BCH PDU in subframe SF=0, where radio frame (SFN mod 4) = 0, then no BCH will be transmitted.

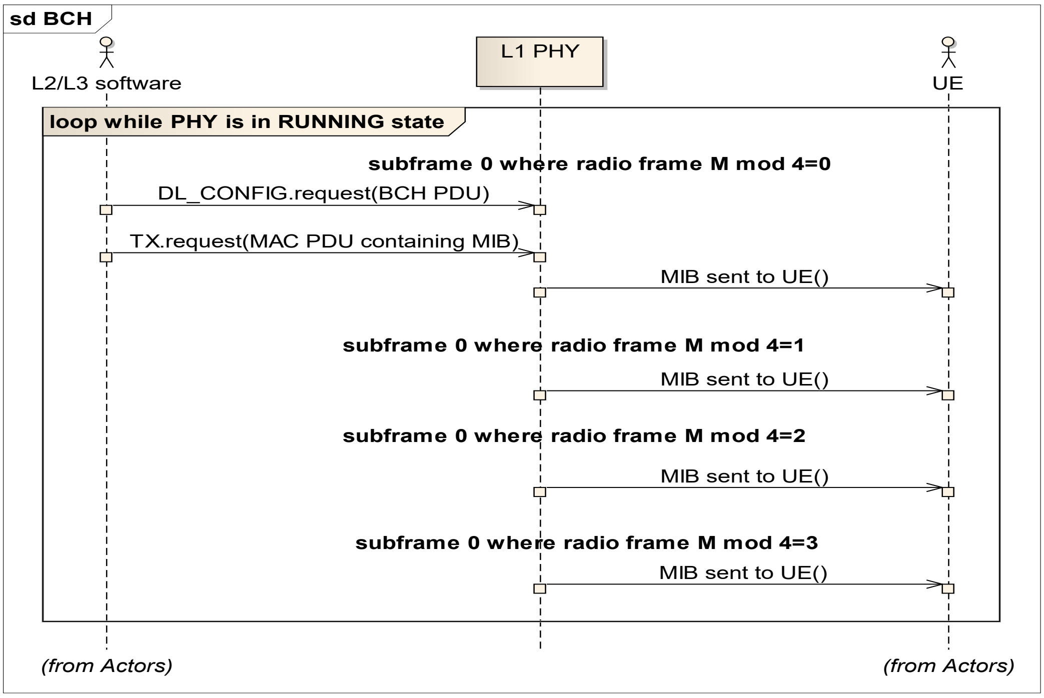


Figure 3-27 BCH procedure

For eMTC introduced in release 13 then the procedure BCH repetition can be enabled. BCH repetition is indicated in the Config TLVs according to the eMTC config "PBCH Repetitions enable R13". The procedure followed is:

- For TDD the L2/L3 software should provide a BCH PDU to the PHY in subframe SF=0, for each radio frame (SFN mod 4) = 0. This is once every 40ms.
- For FDD the L2/L3 software should provide a BCH PDU to the PHY in subframe SF=9, for each radio frame ((SFN-1) mod 4) = 3. This is once every 40ms.
- After receiving a BCH PDU, L1 autonomously transmits the PBCH repetitions (of both legacy and eMTC type) until the next expected BCH PDU SF is reached.

In all the possible PBCH repetition configurations and TDD/FDD, the BCH payload carried via FAPI TX.request message shall match the MIB content according with (SFN mod 4) = 0.

If the PHY does not receive a BCH PDU in subframe SF=0 and if LAA is configured, where radio frame (SFN mod 4) = 0, then no legacy or eMTC BCH will be transmitted.

PCH

The PCH transport channel is used to transmit paging messages to the UE. The UE has specific paging occasions where it listens for paging information [8]. The L2/L3 software is responsible for calculating the correct paging occasion for a UE. The PHY is only responsible for transmitting PCH PDUs when instructed by the `DL_CONFIG.request` message.

The PCH procedure is shown in Figure 3-28. To transmit a PCH PDU the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a PCH PDU and DCI PDU are included.
- In `TX.request` a MAC PDU containing the paging message is included.

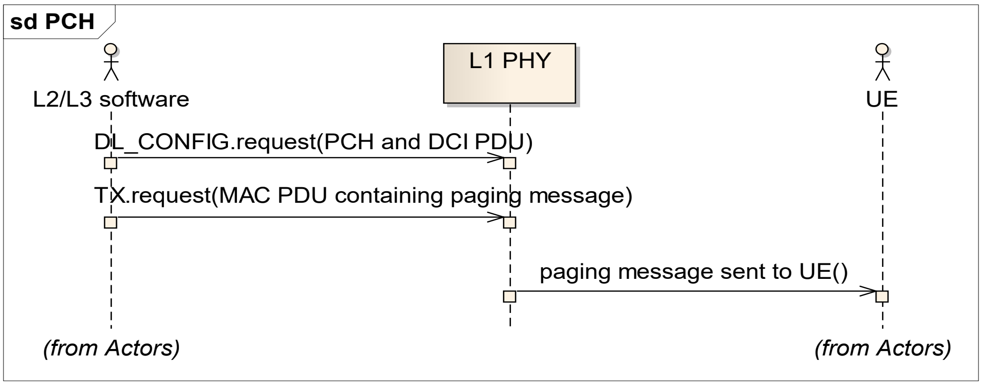


Figure 3-28 PCH procedure

DLSCH

The DLSCH transport channel is used to send data from the eNB to a single UE. This distinguishes the DLSCH from the MCH, where data is sent to multiple UEs. HARQ is always applied on the DLSCH transport channel. Therefore, together with scheduling downlink transmissions the L2/L3 software must schedule uplink bandwidth for the UE to return an ACK/NACK response.

The procedure for the DLSCH transport channel is shown in Figure 3-29.To transmit a DLSCH PDU the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a DLSCH PDU and DCI Format PDU are included. The DCI PDU contains control regarding the DL frame transmission
- In `TX.request` a MAC PDU containing the data is included
- If uplink HARQ signalling is calculated in the MAC a HARQ PDU is included in a later `UL_CONFIG.request`. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration. There are multiple possible HARQ PDUs that can be used to indicate reception of the HARQ response on the uplink:
    - ULSCH_HARQ – is used if the UE is scheduled to transmit data and the ACK/NACK response
    - UCI_HARQ – is used if the UE is just scheduled to transmit the ACK/NACK response
    - ULSCH_UCI_HARQ - is used if the UE is scheduled to transmit data and ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

If the semi-static UE information is held in the MAC the following HARQ PDUs can also be used:

- ULSCH_CQI_HARQ_RI – is used if the UE is scheduled to transmit data, a CQI report and the ACK/NACK response
- UCI_SR_HARQ – is used if the UE has a SR opportunity and is scheduled to transmit the ACK/NACK response

- UCI_CQI_HARQ – is used if the UE is scheduled to transmit a CQI report and the ACK/NACK response
    - UCI_CQI_SR_HARQ - is used if the UE is scheduled to transmit a CQI report, has a SR opportunity and is scheduled to transmit the ACK/NACK response
    - ULSCH_ CSI_UCI_HARQ – is used if the UE is scheduled to transmit data, a CSI report and ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

- If uplink HARQ signalling is calculated in the PHY no information is included in the L1 API regarding HARQ reception.
- The PHY will return the ACK/NACK response information in the `HARQ.indication` message

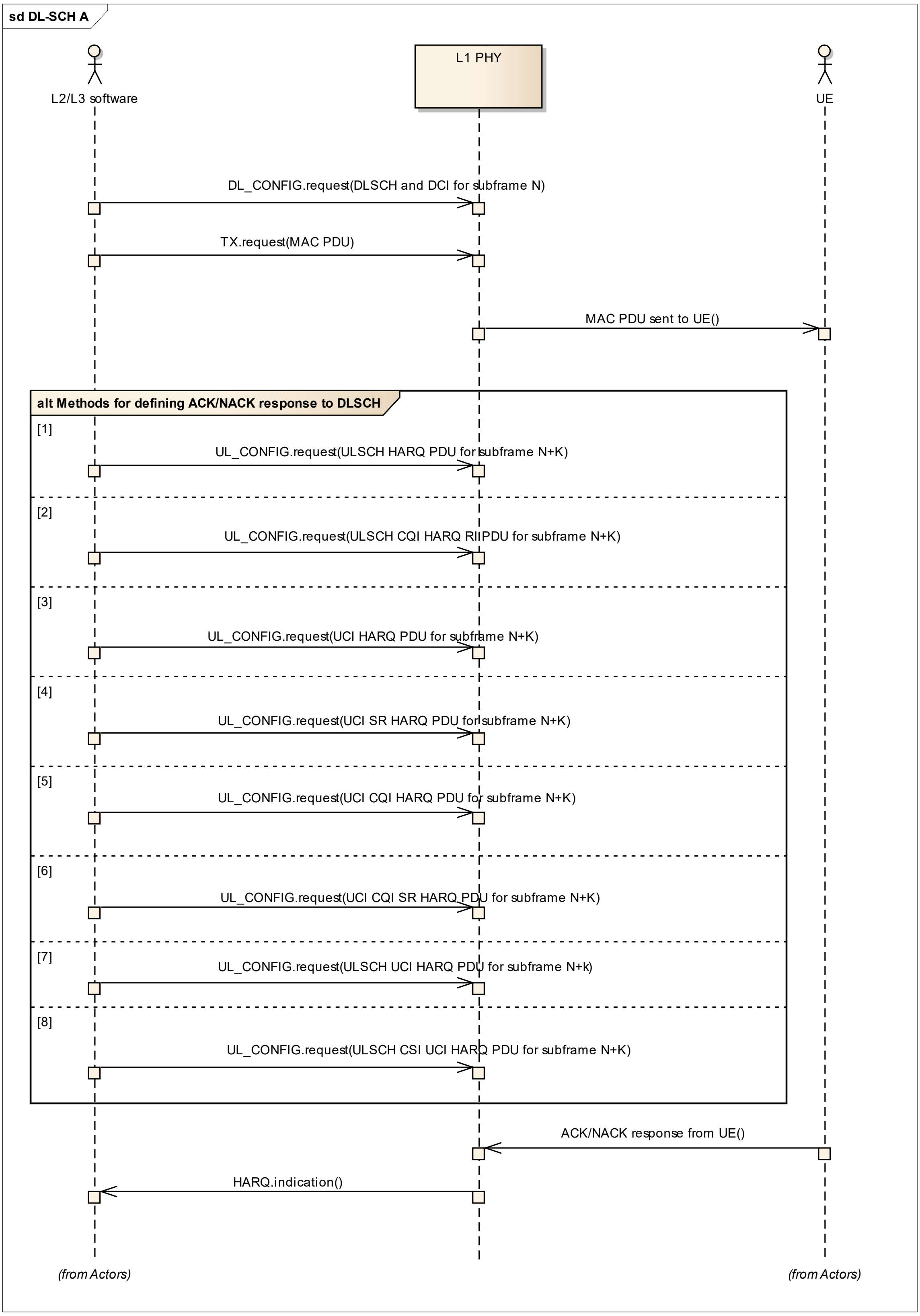


Figure 3-29 DLSCH procedure

With DCI Format 2, 2A, 2B and 2C the DL SCH channel is combined to send two data transport blocks to a UE, this requires a single DCI PDU, but two DLSCH PDUs and two

MAC PDUs. The procedure is shown in Figure 3-30. To initiate a transmission of two data transport blocks the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a DCI Format 2, 2A, 2B or 2C PDU is included. The DCI PDU contains control regarding the DL frame transmission. Two DLSCH PDUs are included, one for each transport block specified in the DCI PDU.
- In `TX.request` two MAC PDUs containing the data are included.

(The remaining behaviour is identical to single-layer transmission.)

- If uplink HARQ signalling is calculated in the MAC a HARQ PDU is included in a later `UL_CONFIG.request`. A single HARQ PDU is required to provide the ACK/NACK response for both transport blocks. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration. There are multiple possible HARQ PDUs that can be used to indicate reception of the HARQ response on the uplink:

    - ULSCH_HARQ – is used if the UE is scheduled to transmit data and the ACK/NACK response
    - UCI_HARQ – is used if the UE is just scheduled to transmit the ACK/NACK response
    - ULSCH_UCI_HARQ - is used if the UE is scheduled to transmit data and ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

If the semi-static UE information is held in the MAC the following HARQ PDUs can also be used:

    - ULSCH_CQI_HARQ_RI – is used if the UE is scheduled to transmit data, a CQI report and the ACK/NACK response
    - UCI_SR_HARQ – is used if the UE is scheduled to transmit a scheduling request and the ACK/NACK response
    - UCI_CQI_HARQ – is used if the UE is scheduled to transmit a CQI report and the ACK/NACK response
    - UCI_CQI_SR_HARQ - is used if the UE is scheduled to transmit a CQI report, has a SR opportunity and is scheduled to transmit the ACK/NACK response
    - ULSCH_ CSI_UCI_HARQ – is used if the UE is scheduled to transmit data, a CSI report and ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

- If uplink HARQ signalling is calculated in the PHY no information is included in the L1 API regarding HARQ reception.
- The PHY will return the ACK/NACK response information in the `HARQ.indication` message

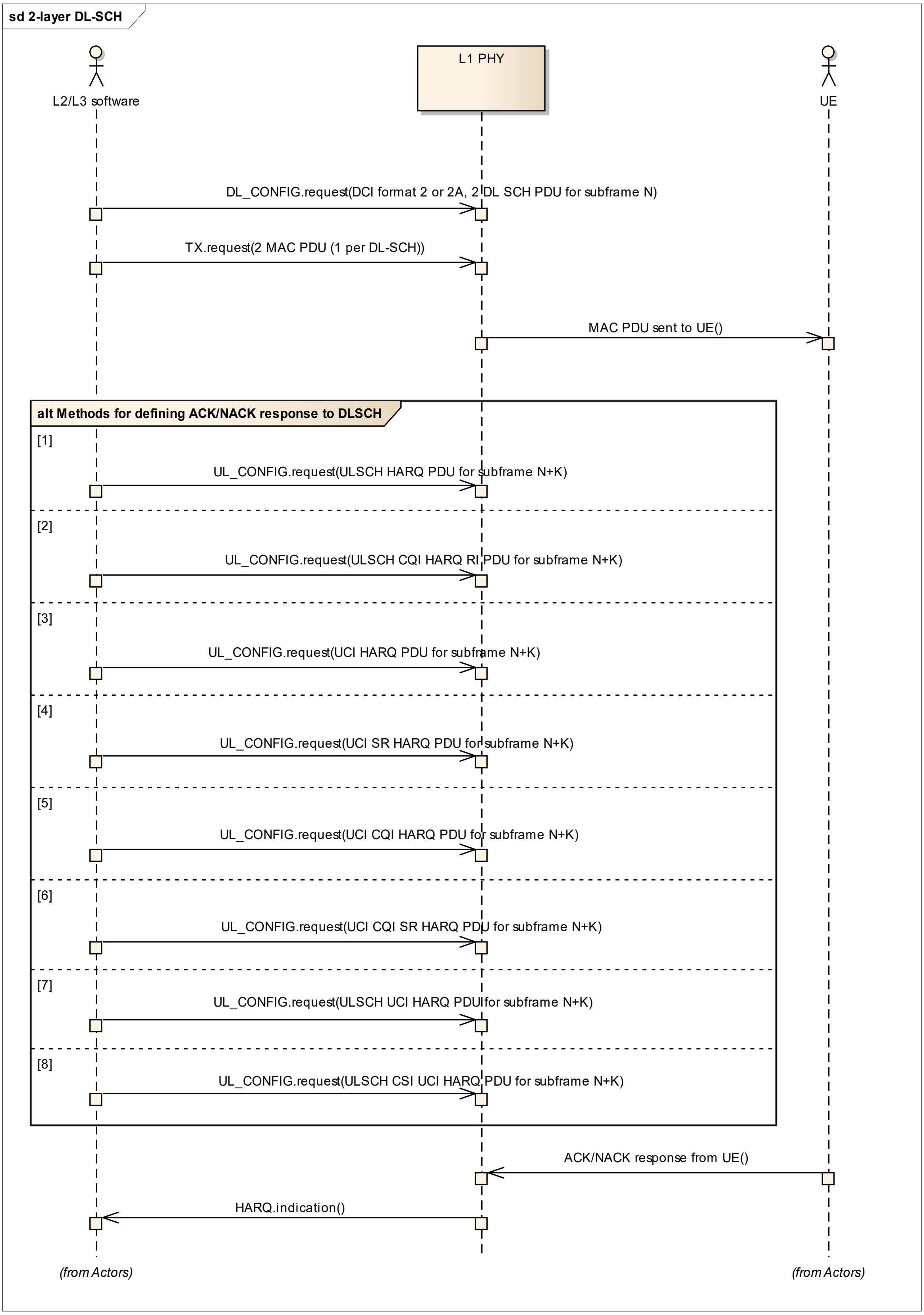


Figure 3-30 2-layer DLSCH procedure

MCH

The MCH transport channel is used to send data, simultaneously, to multiple UEs. The MCH transport channel is transmitted in a subframe configured with a MBSFN region. HARQ is not used.

The MCH procedure is shown in Figure 3-31. To transmit a MCH PDU the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a MCH PDU is included
- In `TX.request` a MAC PDU containing the multicast data is included

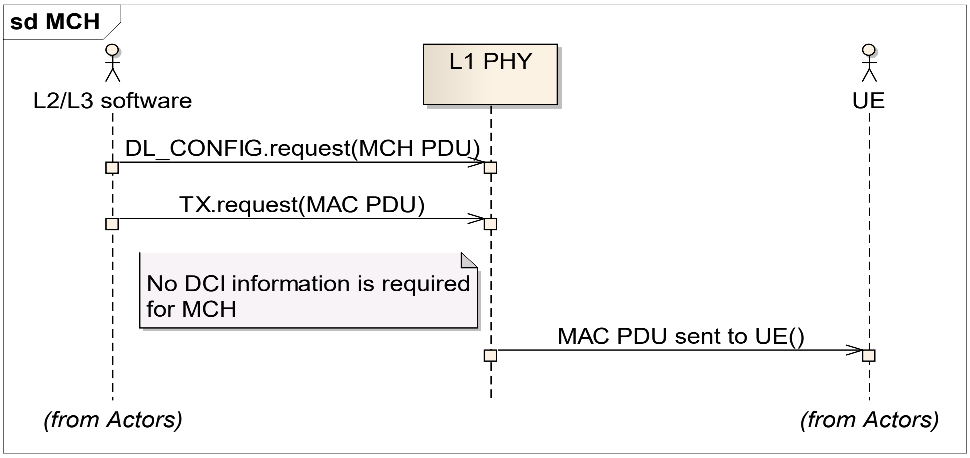


Figure 3-31 MCH procedure

Positioning reference signals

Positioning reference signal (PRS) transmission was introduced in Release 9 to assist LTE positioning at a UE. PRS is transmitted over a PRS bandwidth and periodically in PRS occasions, where each PRS occasion comprises of a sequence of consecutive DL subframes. The L1 API supports the storage of the PRS configuration information in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when it should allocate a PRS transmission, whereas if stored in the PHY there is no need to include this information in the L1 API messages. The procedure for PRS transmission is shown in Figure 3-32.

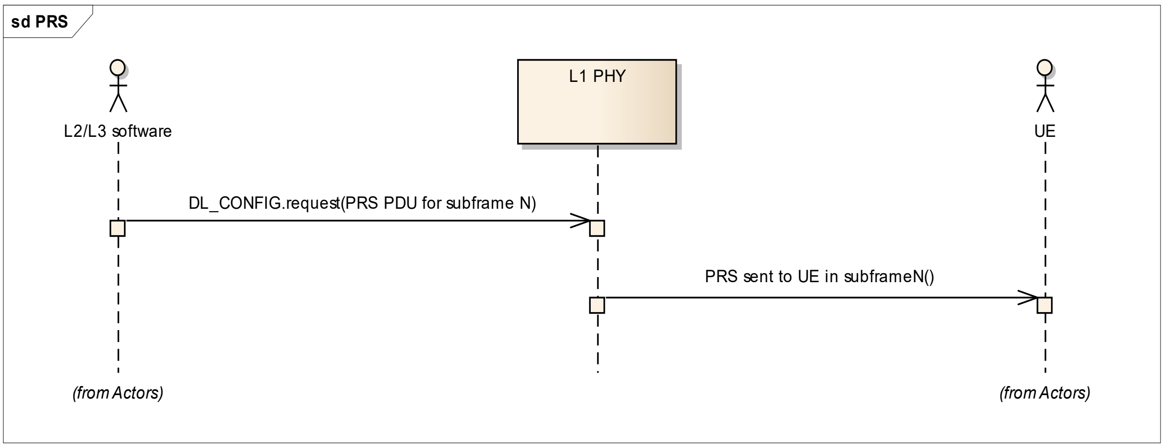


Figure 3-32 PRS procedure

CSI reference signal
The CSI reference signal is added in Release 10. CSI-RS is meant to aid terminals to acquire channel state information (CSI). In the frequency domain CSI-RS is transmitted over the complete cell bandwidth and in the time domain CSI-RS is transmitted over a time period varying from 5ms to 80ms.

3.2.2.7 Uplink

The procedures relating to uplink reception are described in this Section.

RACH
The RACH transport channel is used by the UE to send data to the eNB when it has no scheduled resources. Also, the L2/L3 software can indicate to the UE that it should initiate a RACH procedure. In LTE the occurrence of the RACH follows a pattern advertised on the System Information broadcast messages. The L1 API supports the storage of this information in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when it should allocate a PRACH. If stored in the PHY there is no need to include this information in the L1 API messages.

In the scope of the L1 API, the RACH procedure begins when the PHY receives a `UL_CONFIG.request` message indicating the presence of a RACH.

The RACH procedure is shown in Figure 3-33. To configure a RACH procedure the L2/L3 software must provide the following information:

- If the RACH pattern is stored in the MAC, in `UL_CONFIG.request` the RACH present field must be set. If the RACH pattern is stored in the PHY this step is not required.
- If a UE decides to RACH, and a preamble is detected by the PHY:
    - The PHY will include 1 RACH PDU in the `RACH.indication` message. This RACH PDU includes all detected preambles
- If no RACH preamble is detected by the PHY, then no `RACH.indication` message is sent

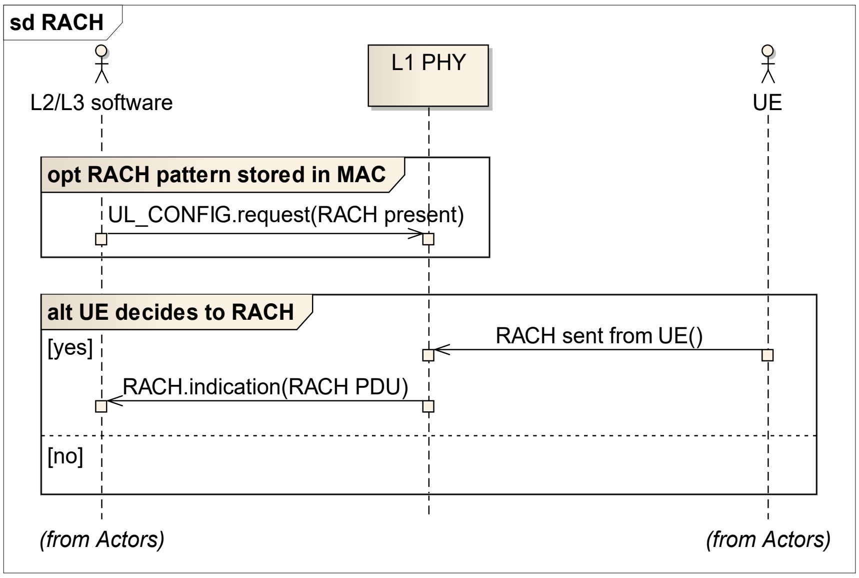


Figure 3-33 RACH procedure

ULSCH

The ULSCH transport channel is used to send data from the UE to the eNB. HARQ is always applied on the ULSCH transport channel. Therefore, together with scheduling uplink transmissions the L2/L3 software must schedule downlink ACK/NACK responses.

The procedure for the ULSCH transport channel is shown in Figure 3-34. To transmit an ULSCH PDU the L2/L3 software must provide the following information:

- Within the `HI_DCI0.request` for subframe N a DCI PDU is included. The DCI Format 0 PDU contains control information regarding the UL frame transmission being scheduled.
- In `UL_CONFIG.request` for subframe N+K1 an ULSCH PDU is included. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration. There are multiple possible ULSCH PDUs that can be used to schedule ULSCH data on the uplink:

    - ULSCH – is used if the UE is scheduled to only transmit data

If the uplink HARQ signalling calculation is performed in the MAC the following ULSCH PDU can also be used:

- ULSCH_HARQ – is used if the UE is scheduled to transmit data and an ACK/NACK response
- ULSCH_UCI_HARQ - is used if the UE is scheduled to transmit data and an ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

If the semi-static UE information is held in the MAC the following ULSCH PDUs can also be used:

- ULSCH_CQI_RI – is used if the UE is scheduled to transmit data and a CQI report
- ULSCH_UCI_CSI - is used if the UE is scheduled to transmit data and a CSI report using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

If both semi-static UE information and uplink HARQ signalling calculation is performed in the MAC the following ULSCH PDU can also be used:

- ULSCH_CQI_HARQ_RI – is used if the UE is scheduled to transmit data, a CQI report and an ACK/NACK response
- ULSCH_CSI_UCI_HARQ – is used if the UE is scheduled to transmit data, CQI report and ACK/NACK response using simultaneous transmission of PUCCH and PUSCH. This is added for Release 10.

- If the Data Report Mode TLV = 0 in the `CONFIG.request` message, then:

    - The PHY will return CRC information for the received data in a the `CRC.indication` message

- The PHY will return the received uplink data in the `RX_ULSCH.indication` message. The `RX_ULSCH.indication` message repeats the CRC information given in the `CRC.indication` message.

- If a CQI was expected in the uplink subframe, the PHY will return the RX_CQI.indication message.
- The ACK/NACK response must be submitted to the PHY using a `HI_DCI0.request` message in subframe N+K1+K2. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration.

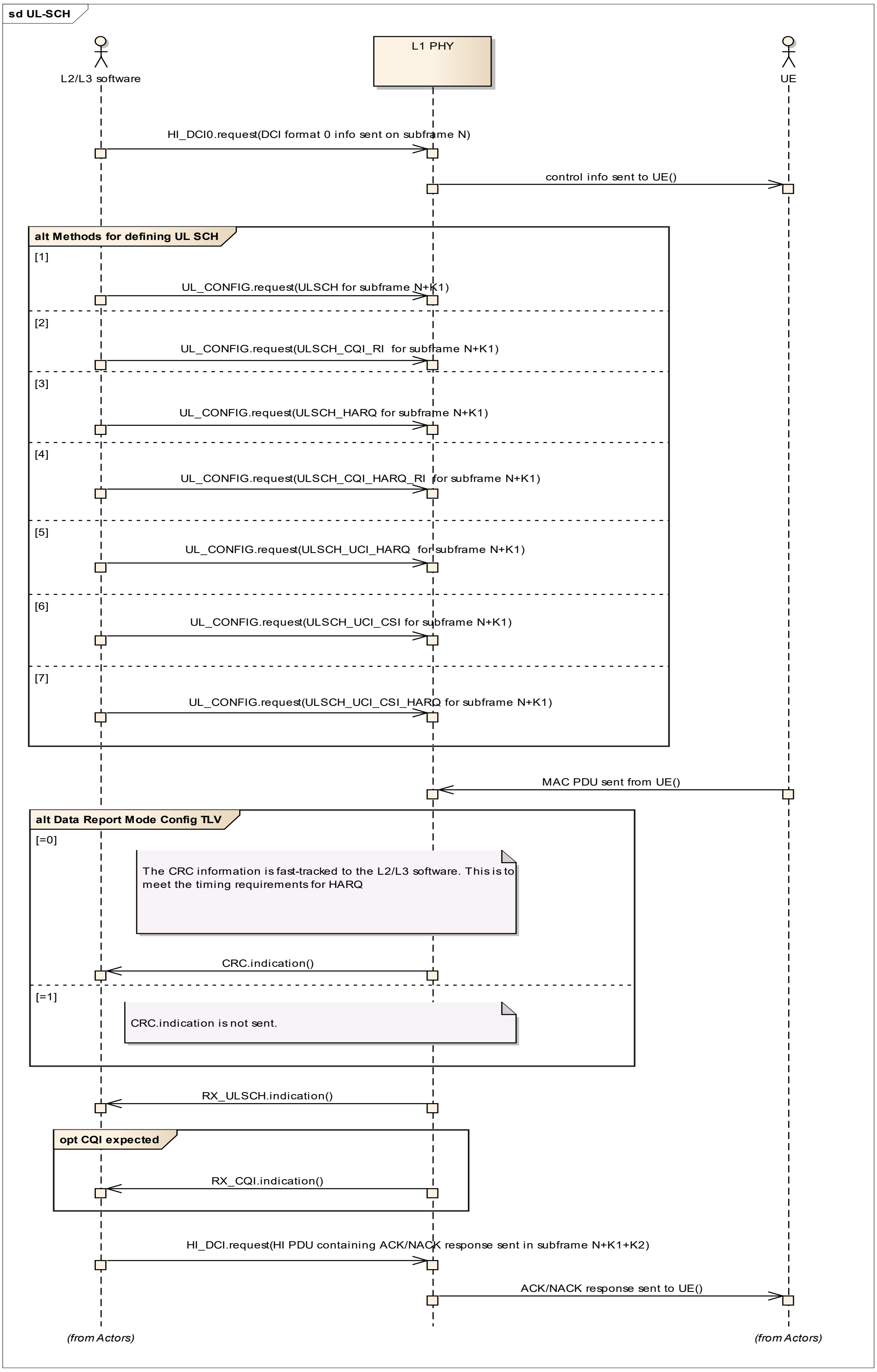


Figure 3-34 ULSCH procedure

With DCI Format 4 the ULSCH channel is used to receive two data transport blocks from a UE. This requires a single DCI PDU, but two ULSCH PDUs. The procedure is shown in Figure 3-35. To initiate a transmission of two data transport blocks the L2/L3 software must provide the following information:

- In `HI_DCI0.request` a DCI Format 4 PDU is included. The DCI PDU contains control regarding the UL frame transmission. Two DLSCH PDUs are included one for each transport block specified in the DCI PDU.
- In `UL_CONFIG.request` a DCI two ULSCH PDUs are included one for each transport block specified in the DCI PDU.
- In `RX_ULSCH.request` two MAC PDUs containing the data are included.

(The remaining behaviour is identical to single-layer transmission.)

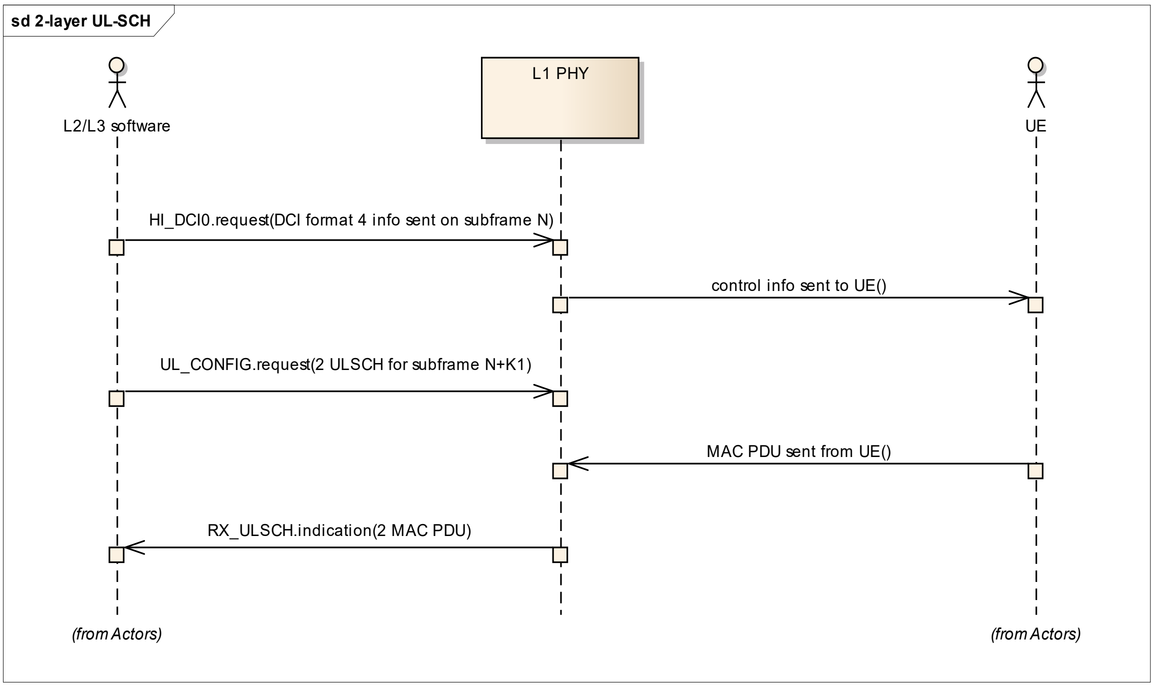


Figure 3-35 2 layer ULSCH procedure

SRS

The sounding reference signal (SRS) is used by L2/L3 software to determine the quality of the uplink channel. In LTE the occurrence of the SRS region follows a pattern advertised on the System Information broadcast messages. The transmission of the SRS by a UE is semi-static information. The L1 API supports the storage of this information in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when to allocate an SRS region and when to expect an SRS transmission from a UE. If stored in the PHY there is no need to include this information in the L1 API messages. In Release 10 aperiodic SRS is introduced where transmission is signalled on the PDCCH.

The SRS procedure is shown in Figure 3-36. To schedule an SRS the L2/L3 software must provide the following information:

- If the SRS information is stored in the MAC, in `UL_CONFIG.request` the SRS present field must be set and one SRS PDU per sounding UE is included. If the SRS information is stored in the PHY this step is not required.

- The PHY will return the SRS response to the L2/L3 software in the `SRS.indication` message

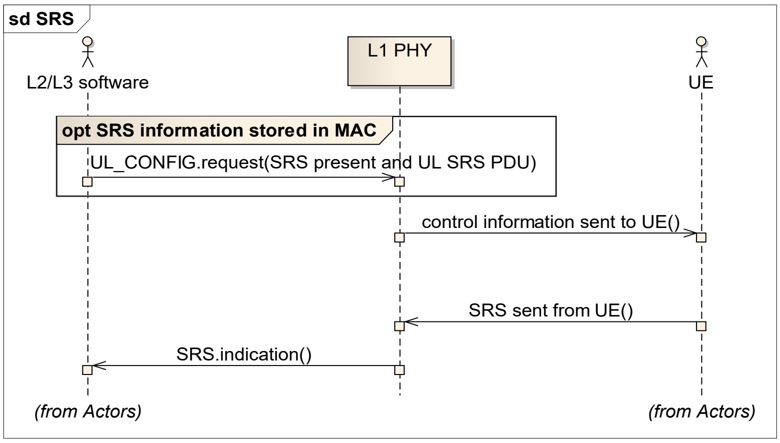


Figure 3-36 SRS procedure

CQI

The CQI reporting mechanism is used by the L2/L3 software to determine the quality of the downlink channel. CQI reporting is initiated through two methods. Firstly, during the RRC connection procedure the L2/L3 software will instruct the UE to transmit periodic CQI reports. Secondly, the L2/L3 software can use the PDCCH to instruct the UE to transmit an aperiodic CQI report.

The L1 API supports the storage of the periodic CQI information in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when to expect a CQI transmission from the UE. If stored in the PHY there is no need to include this information in the L1 API messages.

The CQI reporting procedure is shown in Figure 3-37. To schedule a CQI report the L2/L3 software must provide the following information:

- For an aperiodic report the DCI format 0 PDU is included in the `HI_DCI.request`. This instructs the UE to send a CQI report. For periodic CQI report no explicit DCI information is sent.
- If the CQI information is stored in the MAC:
    - In the `UL_CONFIG.request`, where the L2/L3 software is expecting a CQI report, a CQI PDU is included. There are multiple possible CQI PDUs that can be used to indicate reception of the CQI report on the uplink:
        - ULSCH_CQI_RI – is used if the UE is scheduled to transmit data and a CQI report
        - UCI_CQI – is used if the UE is just scheduled to transmit a CQI report
        - UCI_CQI_SR– is used if the UE is just scheduled to transmit a CQI report and has a SR opportunity
        - ULSCH_UCI_CSI – is used if the UE is scheduled to transmit data and CSI report using simultaneous PUCCH and PUSCH transmission. This is added for Release 10.

If the uplink HARQ signalling calculation is performed in the MAC the following CQI PDUs can also be used:

- ULSCH_CQI_HARQ_RI – is used if the UE is scheduled to transmit data, a CQI report and the ACK/NACK response
    - UCI_CQI_HARQ - is used if the UE is scheduled to transmit a CQI report and the ACK/NACK response
    - UCI_CQI_SR_HARQ – is used if the UE is scheduled to transmit a CQI report, has a SR opportunity and is scheduled to transmit the ACK/NACK response
    - ULSCH_CSI_UCI_HARQ – is used if the UE is scheduled to transmit data, a CSI report and the ACK/NACK response using simultaneous PUCCH and PUSCH transmission. This is added for Release 10.

- If the CQI information is stored in the PHY, the L2/L3 software does not include any information relating to CQI in the `UL_CONFIG.request` message.

    - The PHY will return the CQI report to the L2/L3 software in the `RX_CQI.indication` message

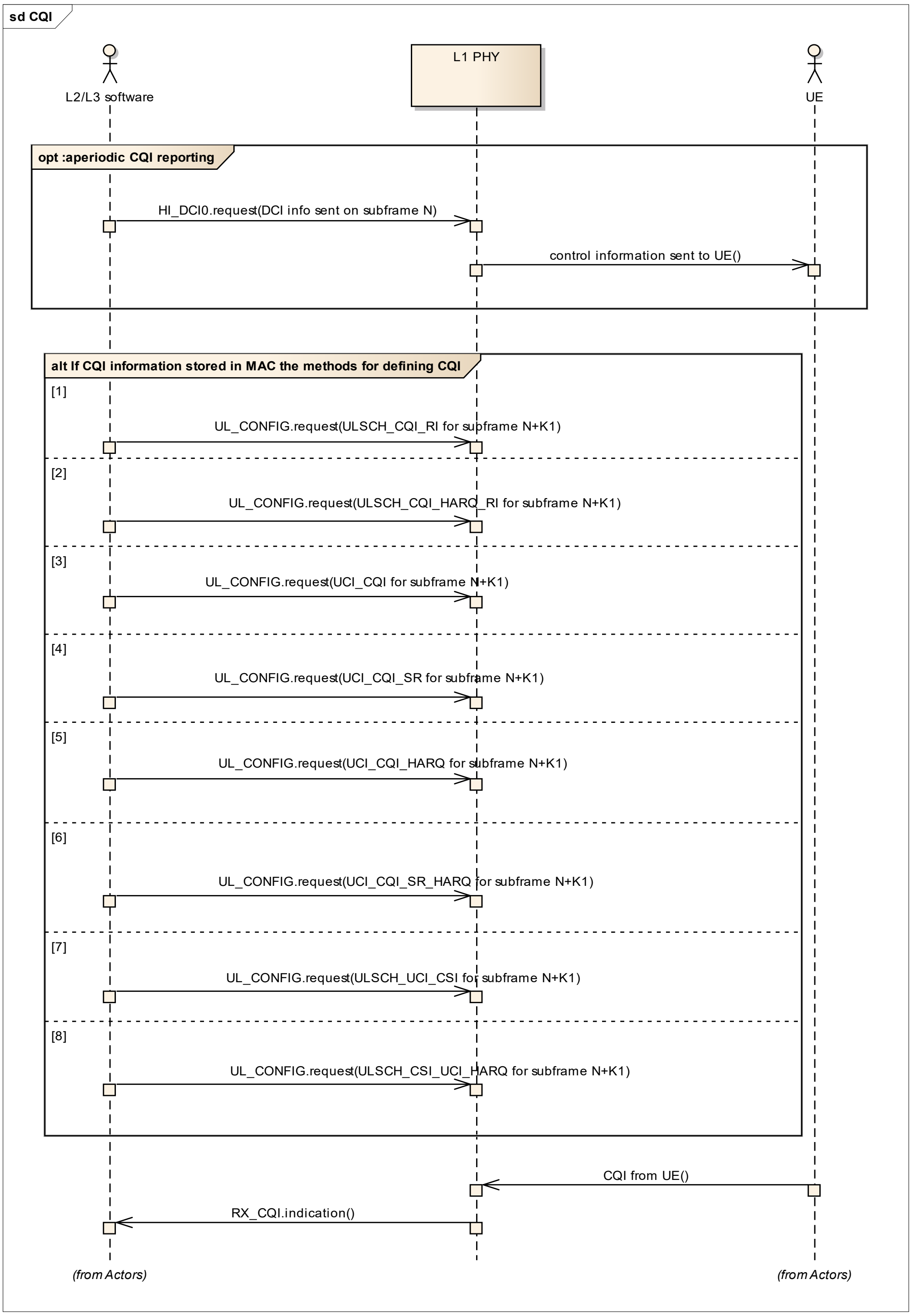


Figure 3-37 CQI procedure

SR

The scheduling request (SR) procedure is used by the UE to request additional uplink bandwidth. The L2/L3 software configures the SR procedure during the RRC connection procedure.

The L1 API supports the storage of the SR information in either the MAC or PHY. If stored in the MAC the L1 API is used to instruct the PHY when to expect an SR transmission from the UE. If stored in the PHY there is no need to include this information in the L1 API messages.

The SR procedure is shown in Figure 3-38. To schedule an SR the L2/L3 software must provide the following information:

- If the SR information is stored in the MAC:
    - In the `UL_CONFIG.request` a SR PDU is included. There are multiple possible SR PDUs that can be used to indicate reception of the SR on the uplink:
        - UCI_SR – is used if the UE has a SR opportunity
        - UCI_CQI_SR– is used if the UE is just scheduled to transmit a CQI report and has a SR opportunity

If the uplink HARQ signalling calculation is performed in the MAC the following SR PDUs can also be used:

- UCI_SR_HARQ – is used if the UE has a SR opportunity and is scheduled to transmit the ACK/NACK response
- UCI_CQI_SR_HARQ – is used if the UE is scheduled to transmit a CQI report, has a SR opportunity and is scheduled to transmit the ACK/NACK response

- If the SR information is stored in the PHY, the L2/L3 software does not include any information relating to SR in the `UL_CONFIG.request` message.
    - The PHY will return the SR to the L2/L3 software in the `RX_SR.indication` message

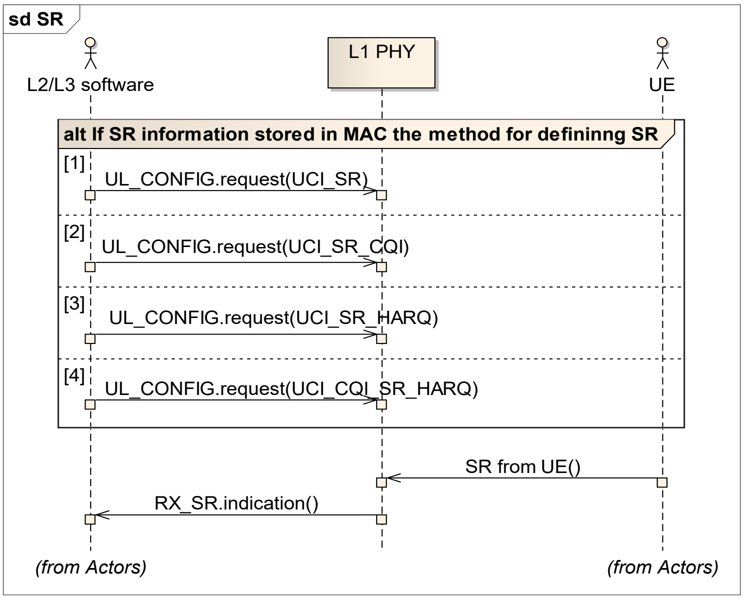


Figure 3-38 SR procedure

3.2.2.8 Cross-carrier scheduling

In Release 10 carrier aggregation was introduced with a UE connected to a primary cell and optionally secondary cells. This inclusion of multiple cells permits the concept of cross-carrier scheduling where a UE is signalled scheduling information on one cell but transmits, or receives, the data is a different cell. The procedure for this for both downlink and uplink is described in this section.

Downlink

In the downlink one cell may transmit scheduling information on a DCI with no corresponding DLSCH PDU. Similarly, a DLSCH PDU may be present with no DCI.

An example procedure for scheduling on the primary cell and data transmission on a secondary cell is given in Figure 3-39, the L2/L3 software must provide the following information:

- In the cell used for scheduling a `DL_CONFIG.request` with a DCI Format PDU is included. The DCI PDU contains control regarding the DL frame transmission.
- In the cell used for transmission a `DL_CONFIG.request` with a DLSCH PDU is included.
- In the cell used for transmission a `TX.request` with a MAC PDU containing the data is included
- The cell used for the uplink HARQ signalling is dependent on whether the UE also has a scheduled ULSCH transmission. In this example no ULSCH transmission is scheduled resulting in the primary cell being used.
- If uplink HARQ signalling is calculated in the MAC a HARQ PDU is included in a later `UL_CONFIG.request`. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration. There are multiple possible HARQ PDUs that can be used to indicate reception of the HARQ response on the uplink. The different HARQ options are explained in Section 3.2.2.7.
- If uplink HARQ signalling is calculated in the PHY an indication is still required from the MAC to the PHY. Since the HARQ signalling can be performed in multiple cells, only the MAC is aware of whether a HARQ is expected.
- The PHY will return the ACK/NACK response information in the `HARQ.indication` message.

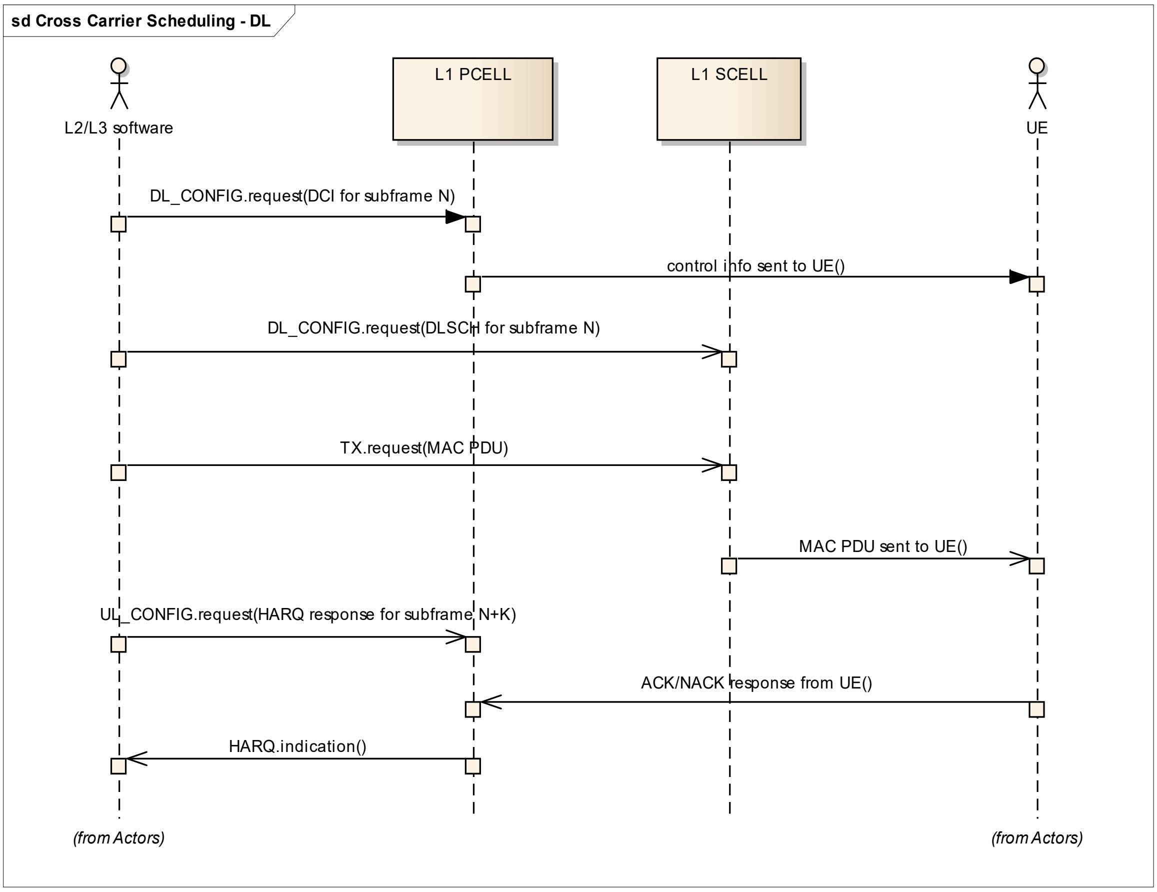


Figure 3-39 Downlink cross carrier scheduling procedure

Uplink

In the uplink one cell may transmit scheduling information on a DCI with no corresponding ULSCH PDU. Similarly, a ULSCH PDU may be present with no DCI.

An example procedure for scheduling on the primary cell and data transmission on a secondary cell is given Figure 3-40, the L2/L3 software must provide the following information:

- In the cell used for scheduling an HI_DCI0.request for subframe N with a DCI PDU is included. The DCI Format 0 or 4 PDU contains control information regarding the UL frame transmission being scheduled.
- In the cell used for transmission an `UL_CONFIG.request` for subframe N+K1 with an ULSCH PDU is included. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration. There are multiple possible ULSCH PDUs that can be used to schedule ULSCH data on the uplink. The different ULSCH PDU options are explained in Section 3.2.2.7.
- If the Data Report Mode TLV = 0 in the `CONFIG.request` message, then:
    - The PHY will return CRC information for the received data in a the `CRC.indication` message
- The PHY will return the received uplink data in the `RX_ULSCH.indication` message. The `RX_ULSCH.indication` message repeats the CRC information given in the CRC.indication message.

- In the cell used for scheduling the ACK/NACK response must be submitted to the PHY using a HI_DCI0.request message in subframe N+K1+K2. The timing of this message is dependent on whether a TDD or FDD system is in operation. For TDD, it is also dependent on the DL/UL subframe configuration.

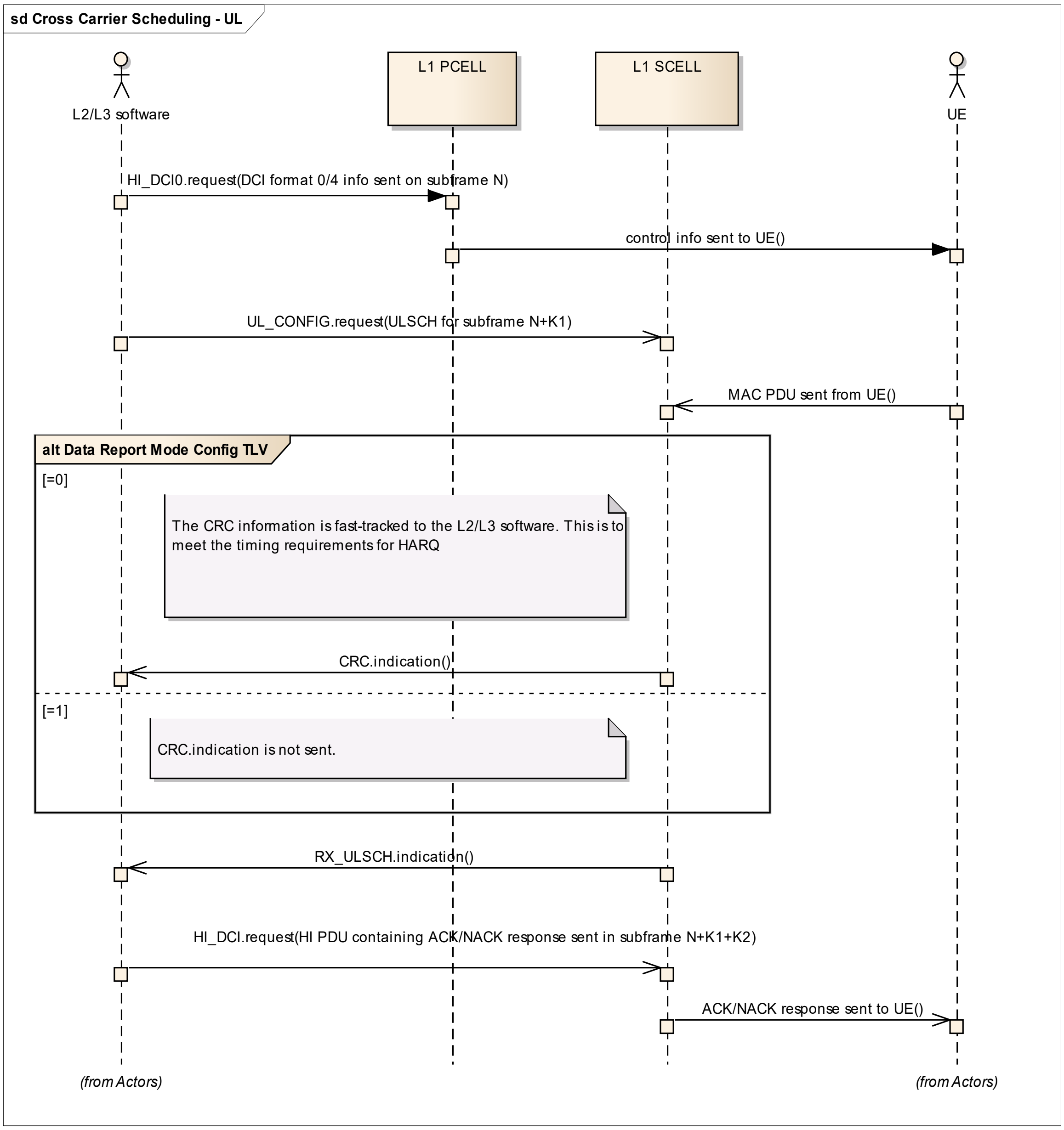


Figure 3-40 Uplink cross carrier scheduling procedure

3.2.2.9 Error sequences

The error sequences used for each subframe procedure are shown in Figure 3-41 to Figure 3-44. In all subframe procedures errors that are detected by the PHY are reported using the `ERROR.indication` message. In Section 3.3, the L1 API message definitions include a list of error codes applicable for each message.

The `DL_CONFIG.request`, `UL_CONFIG.request`, `HI_DCI0.request` and `TX.request` messages include information destined for multiple UEs. An error in information destined for one UE can affect a transmission destined for a different UE. For each message the `ERROR.indication` sent by the PHY will return the first error it encountered.

If the L2/L3 software receives an `ERROR.indication` message for `DL_CONFIG.request`, `UL_CONFIG.request`, `HI_DCI0.request` or `TX.request`, it should assume that the UE did not receive data and control sent in this subframe. This is similar to the UE experiencing interference on the air-interface and LTE mechanisms, such as, HARQ and ARQ, will enable the L2/L3 software to recover.

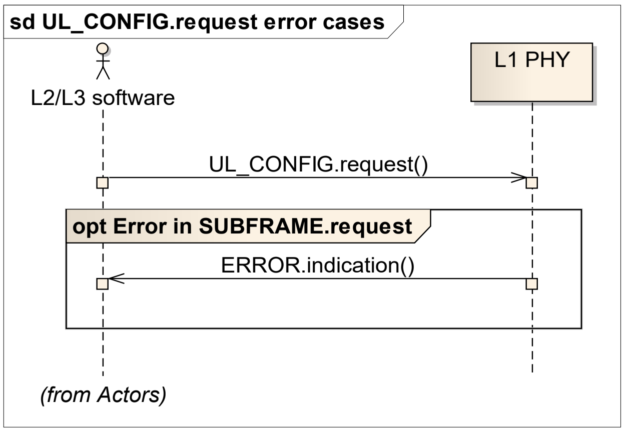


Figure 3-41 UL_CONFIG.request error sequence

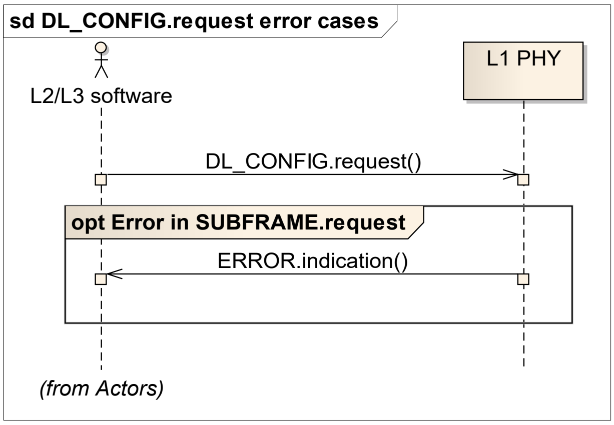


Figure 3-42 DL_CONFIG.request error sequence

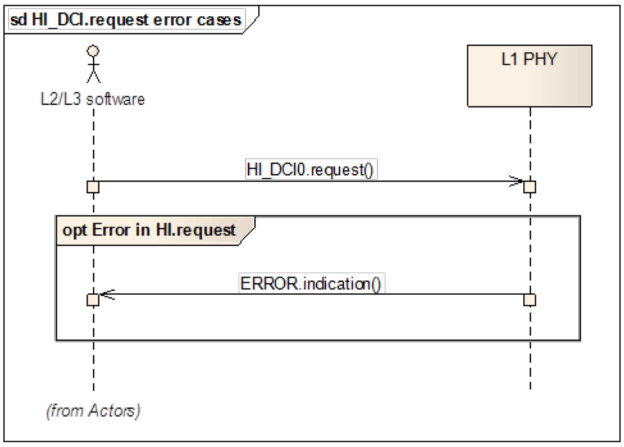


Figure 3-43 HI_DCI0.request error sequence

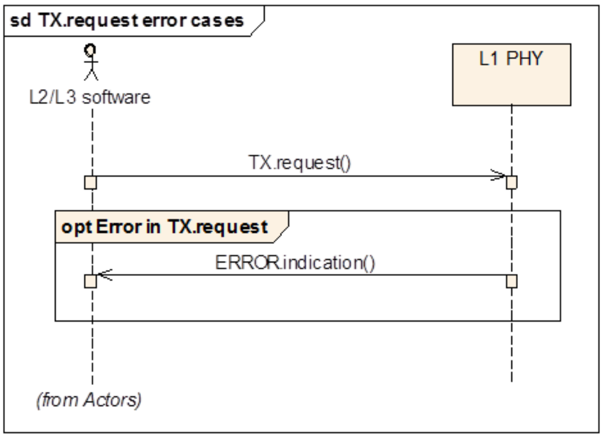


Figure 3-44 TX.request error sequence

3.2.3 LBT Procedures

3GPP Release 13 License Assisted Access (LAA) introduces a new frame structure, frame structure type 3 which in this API is referred to as supplementary downlink (SDL), for use by unlicensed secondary cells. Release 13's definition of frame structure type 3 is very similar to that of frame structure type 1 (or FDD), with the following differences:

- Uplink transmissions are not allowed.
- Downlink transmissions may start anywhere within a subframe and occupy one or more consecutive subframes.
- The last subframe in a downlink transmission burst may end at the subframe boundary or any one of the DwPTS durations defined for frame structure type 2.
- When configured with capability of initial partial subframe, the LAA SCell shall begin transmission of PDSCH/PDCCH/EPDCCH in the second slot of the subframe.

In unlicensed SCells residing in a shared specturm, where coexistence with other Wi-Fi and LAA devices in a fair manner must be maintained, the eNB must perform LBT (Listen Before Talk) procedure prior to transmitting any signal.

LBT messages are used both by the L2/L3 software to control and configure the PHY to performing “listening” (measuring the received signal energy of a channel comparing it to a threshold) and “capturing the channel” (transmitting), and by the PHY to indicate to L2/L3 the result.

The “listening” is defined in 3GPP to carry on a time base which is completely different than the TTI based time scale of LTE technology, instead it is based on defer periods and CCA slot duration (9 μsec). Therefore, the LBT procedure is treated as an asynchronous entity to LBT (not SF procedures) in the following manner:

- The LBT_DL_CONFIG.request message is used to trigger the LBT procedure. This is sent in subframe N-i, where i can be any value ≥ 0, and the value range will be dependent on the PHY implementation.
- The LBT_DL_CONFIG.request message will include a transmission opportunity (TXOP) given in subframe. For each of the TXOP subframes
    - A DL_CONFIG.request message is sent which includes the relevant DCI and DLSCH PDUs
    - A TX.request message is sent including data to transmit
    - If the LBT procedure is successful and the channel is acquired then
        - The LBT_DL.indication message is returned indicating success
        - The downlink data is transmitted, and the corresponding HARQ is returned to the PCell
    - If the LBT procedure is not successful then
        - The LBT_DL.indication message is returned indicating failure
        - The downlink data is not transmitted

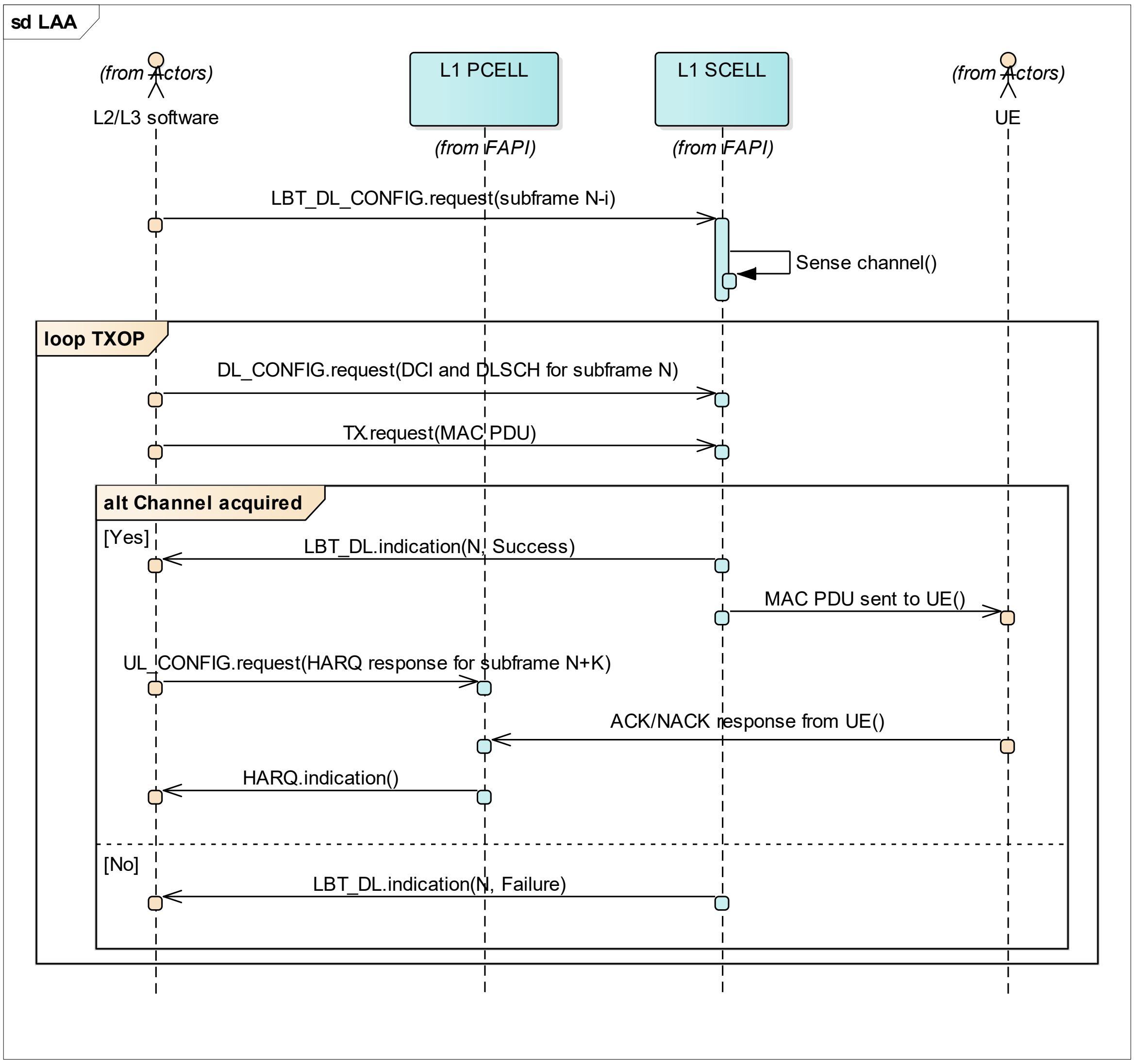


Figure 3-45 LBT procedure

3.2.4 NB-IOT Procedures

3GPP Release 13 introduced the NB-IOT carrier to developed to support the Internet of Things (IoT) market. NB-IOT has three different operating modes:

- **In-band**: the NB-IOT cell is allocated within the LTE cell, utilizing a single or several (if single or multi carrier NB-IOT) PRBs in the system BW of the LTE cell. LTE signals (CRS, PSS, SSS, PBCH, PDCCH, SRS…) of the LTE cell are contained in the PRBs of the NB-IOT cell. In-band can be either when the same PCI indicator is true / false (PCID =/≠ NPCID)
- **Guardband**: the NB-IOT cell is allocated at the guardband of the LTE cell, utilizing a single or several (if single or multi carrier NB-IOT) PRBs in the guardband of the system BW of the LTE cell. NB-IOT cell contains only NB-IOT signals
- **Standalone**: the NB-IOT cell is allocated irrespective of LTE cell, utilizing a single or several (if single or multi carrier NB-IOT) PRBs

The NB-IOT cell is a “logically” parallel cell to the LTE (for all operating modes) and each NB-IOT cell has its own Cell-ID, synchronization and an enhanced SFN method (Hyper SFN) and the following physical signals:

- NPSS
- NSSS
- NRS

And channels:

- NPBCH,
- NPDCCH
- NPDSCH
- NPUSCH
- NPRACH

Therefore, each NB-IOT carrier will have it’s own instance of FAPI. For the in-band mode there will be two instances of FAPI one supporting the LTE carrier and one supporting the NB-IOT carrier.

NB-IOT cells can also be anchor cells or non-anchor cells. Both anchor and non-anchor cells transmit the following channels:

- NPDCCH
- NPDSCH
- NPUSCH

Anchor cells additionally transmit:

- NPSS
- NSSS
- NRS
- NPBCH
- NRACH

3.2.4.1 Subframes

The NB-IOT downlink subframe structure (for an anchor cell) is given in Figure 3-46 and shows:

- A subframe contains one either a NB-IOT signal or a NB-IOT channel
- NRS is transmitted in every NB-IOT DL subframe
- NPSS is transmitted in subframe 5
- NSSS is transmitted in subframe 9 of even radioframes
- NPBCH is transmitted in subframe 0
- The remaining subframes can be used to transmit either NPDCCH or NPDSCH
- The downlink is scheduled via the NPDCCH with the data (PDSCH) transmitted in a later subframe
- The NPDCCH DCI can be transmitted over multiple subframes
- The NPDSCH MAC PDU can be transmitted over multiple subframes

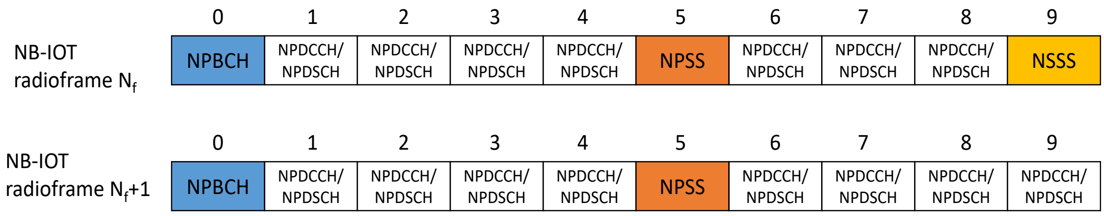


Figure 3-46 NB-IOT downlink subframes

The NB-IOT uplink channels are the NRACH and NULSCH. In NB-IOT the uplink control information (UCI) is transmitted on NULSCH.

NB-IOT uses asynchronous HARQ for both uplink and downlink transmission. NB-IOT is designed to allow simpler UE implementations and this results in a HARQ which is longer than the LTE 8ms HARQ cycle. However, for the eNB the latency can still be the same as LTE, namely 3ms between reception of the NULSCH and transmission of DCI N0 to schedule the next uplink transmission.

3.2.4.2 Downlink

BCH

The BCH transport channel is used to transmit the Master Information Block (MIB) information to the UE. For NB-IOT it is transmitted in subframe 0 of each radio frame in an anchor carrier. When the radio frame (SFN mod 64) = 0 an updated MIB is transmitted in subframe 0. The MIB is divided into 8 sub-blocks, when the radio frame (SFN mod 8) = 0 a new MIB sub-block is transmitted, when the radio frame (SFN mod 8) ≠ 0 the MIB is repeated.

The MIB is only transmitted in anchor carriers.

The BCH procedure is shown in Figure 3-47. The L2/L3 software should provide a BCH PDU to the PHY in subframe SF=0, for each radio frame (SFN mod 64) = 0. This is once every 640ms. The L2/L3 software provides the following information:

- In `DL_CONFIG.request` a N-BCH PDU is included.
- In `TX.request` a MAC PDU containing the MIB is included.

If the PHY does not receive a N-BCH PDU in subframe SF=0, where radio frame (SFN mod 64) = 0, then no BCH will be transmitted.

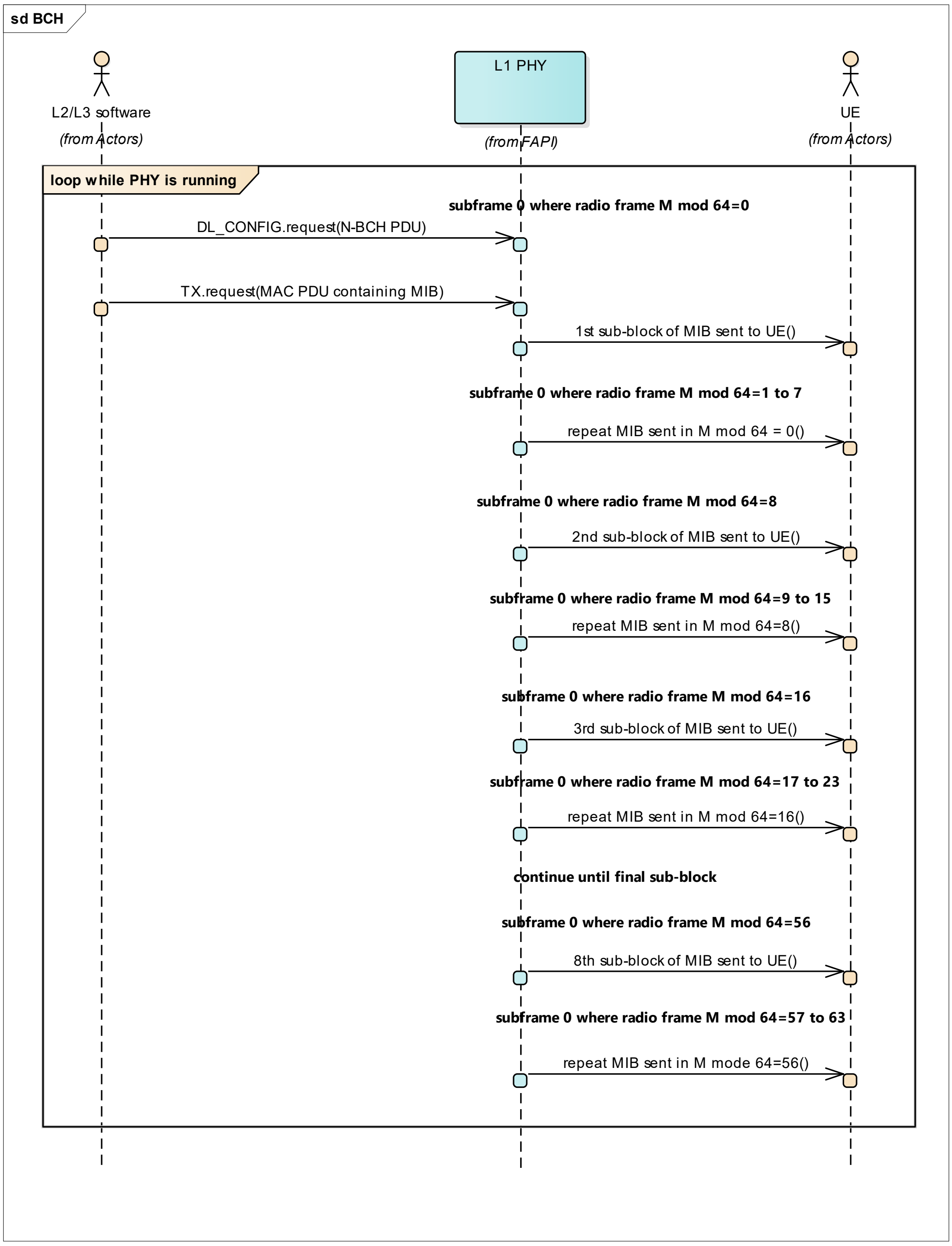


Figure 3-47 NB-IOT BCH procedures

PCH

The PCH transport channel is used to transmit paging messages to the UE. The UE has specific paging occasions where it listens for paging information [8]. The L2/L3 software is responsible for calculating the correct paging occasion for a UE. The PHY is only responsible for transmitting PCH PDUs when instructed by the `DL_CONFIG.request` message.

The PCH procedure is shown in Figure 3-48. To transmit a PCH PDU the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a NPDCCH DCI DL PDU is included. This specifies a repetition number indicating the number of times the NPDCCH DCI is repeated and a scheduling delay between the final transmission on the DCI DL PDU and the MAC PDU containing the paging message.

At the subframe where the paging MAC PDU is transmitted the L2/L3 software provides the following information:

- In `DL_CONFIG.request` a NDLSCH PDU is included. This specifies a repetition number indicating the number of times the NDLSCH is repeated.
- In `TX.request` a MAC PDU containing the paging message is included.

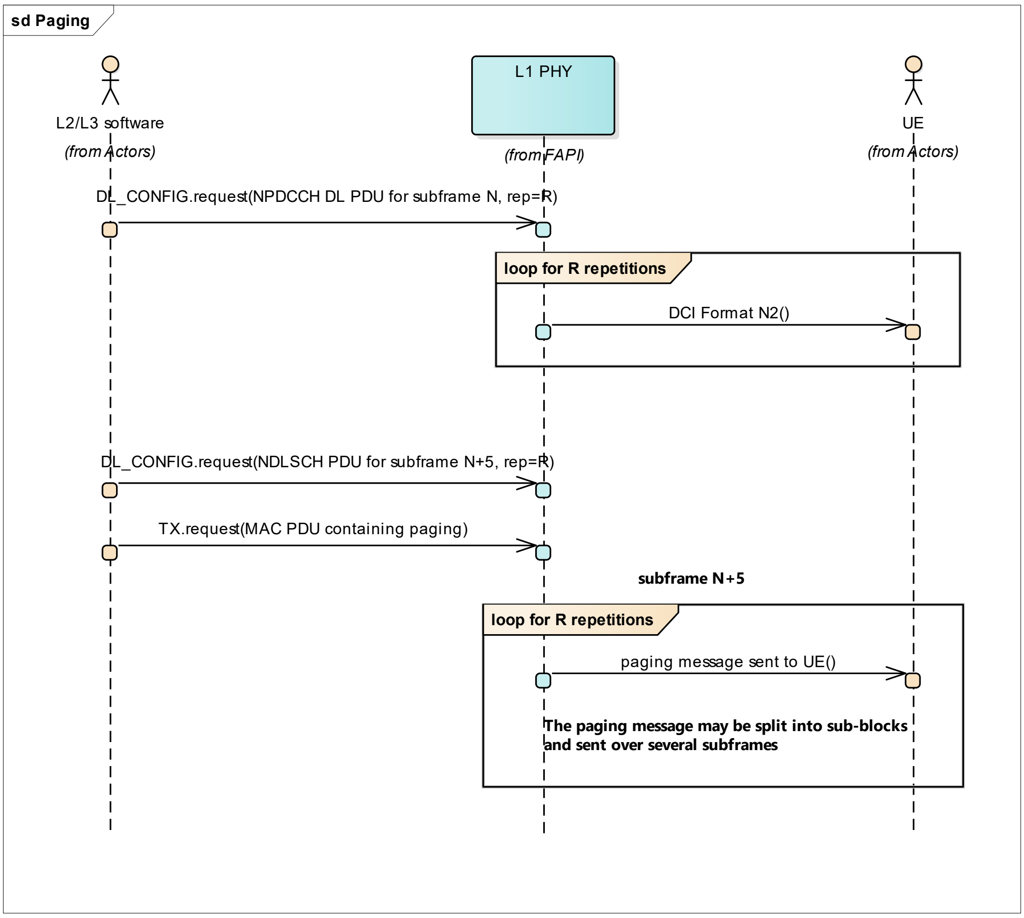


Figure 3-48 NB-IOT PCH procedure

DLSCH

The DLSCH transport channel is used to send data from the eNB to a single UE. HARQ is optional on the DLSCH transport channel. If HARQ is enabled the L2/L3 software must schedule uplink bandwidth for the UE to return an ACK/NACK response on the NULSCH.

The procedure for the DLSCH transport channel is shown in Figure 3-49. To transmit a DLSCH PDU the L2/L3 software must provide the following information:

- In `DL_CONFIG.request` a NPDCCH DCI Format PDU is included. The DCI PDU contains control regarding the DL frame transmission, in addition it also specifies the number of times the DCI is repeated and the delay between the final DCI transmission and start of NDLSCH transmission.

At the required subframe the MAC PDU is transmitted with the L2/L3 software providing the following information:

- In `DL_CONFIG.request` a NDLSCH PDU is included which includes the number of times the MAC PDU is repeated.
- In `TX.request` a MAC PDU containing the data is included

If uplink HARQ is enabled for this PDU then at the relevant subframe the L2/L3 software provides the following information:

- A NB-HARQ PDU is included in an `UL_CONFIG.request`. The timing of this message is configured by an ACK delay value specified in the earlier NPDCCH PDU.
- The PHY will return the ACK/NACK response information in the `NB-HARQ.indication` message

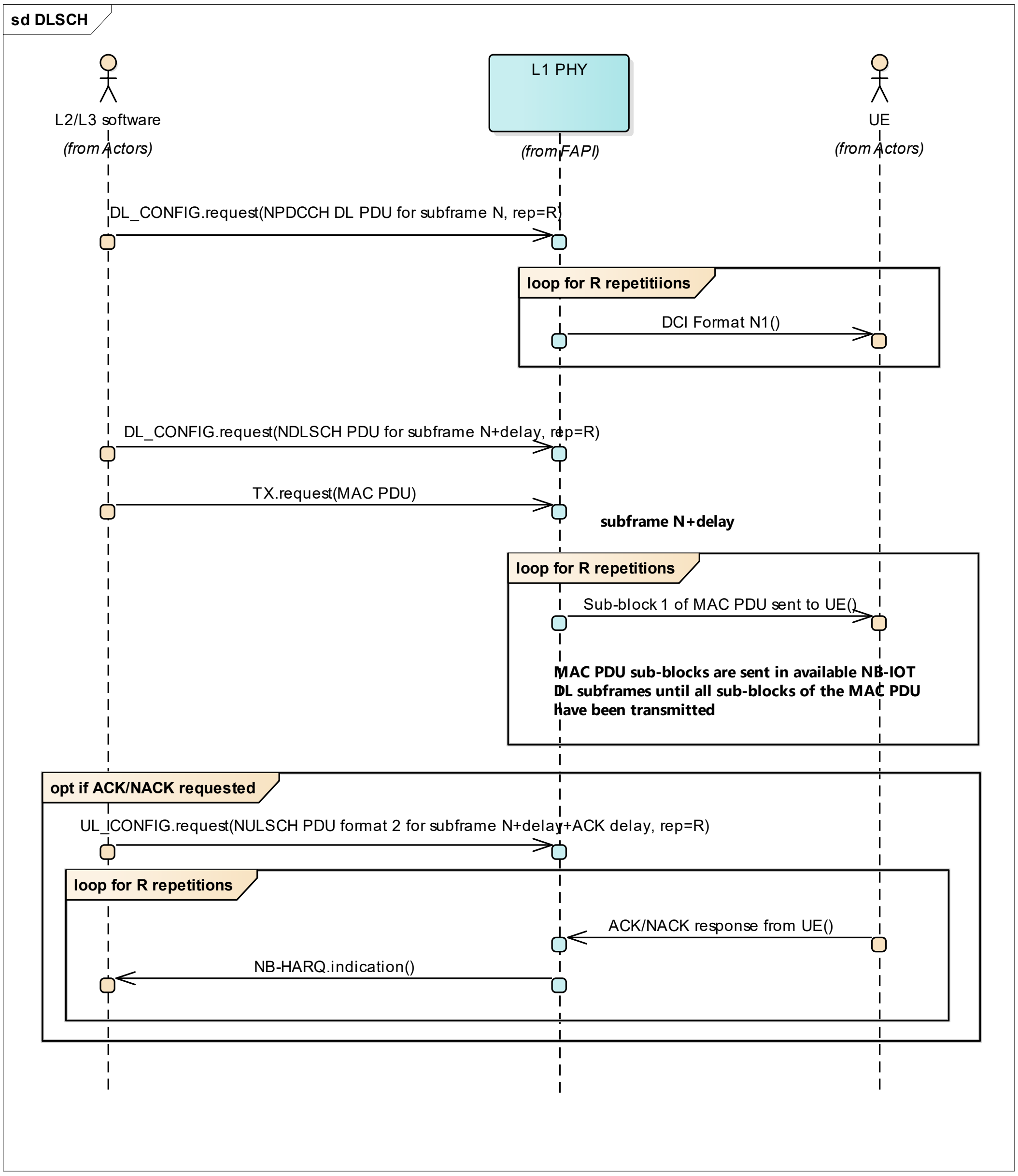


Figure 3-49 NB-IOT DLSCH procedure

PSS

The NB-IOT primary synchronization signal (NPSS) is transmitted in subframe 5 of every radioframe. The NPSS procedure is shown in Figure 3-50 and the information required from the L2/L3 software is as follows:

- The NPSS configuration is stored in the PHY during the initialization procedure.
- In every subframe 5 the PHY transmits the NPSS.

The NPSS is only transmitted in anchor carriers.

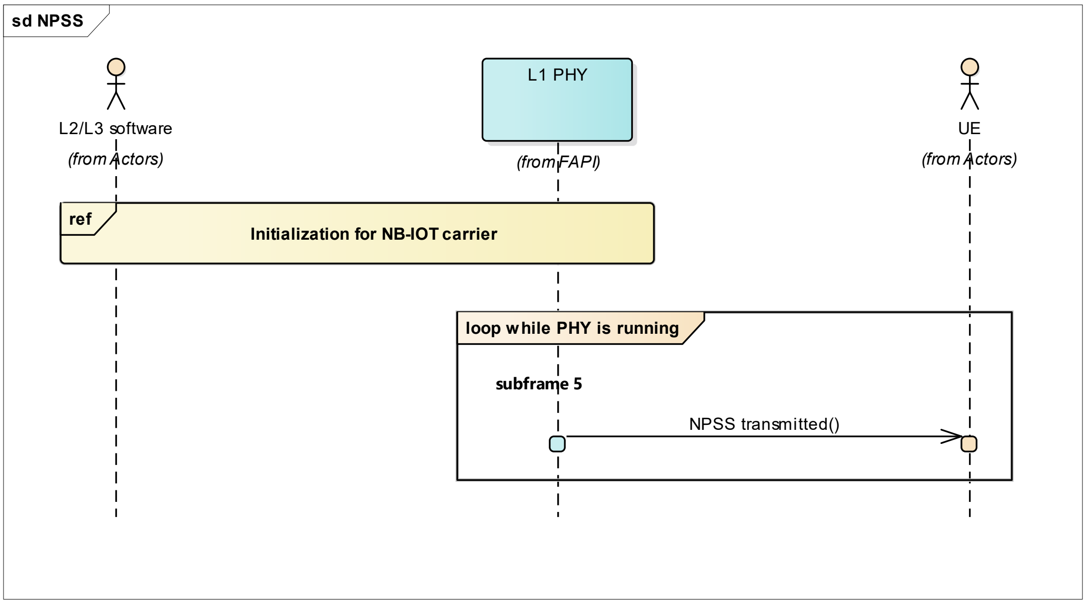


Figure 3-50 NPSS procedure

SSS

The NB-IOT secondary synchronization signal (NSSS) is transmitted in subframe 9 of every even radioframe. The NSSS procedure is shown in Figure 3-51 and the information required from the L2/L3 software is as follows:

- The NSSS configuration is stored in the PHY during the initialization procedure.
- In subframe 9 of every radioframe where M mod 2 =0 the PHY transmits the NSSS.

The NPSS is only transmitted in anchor carriers.

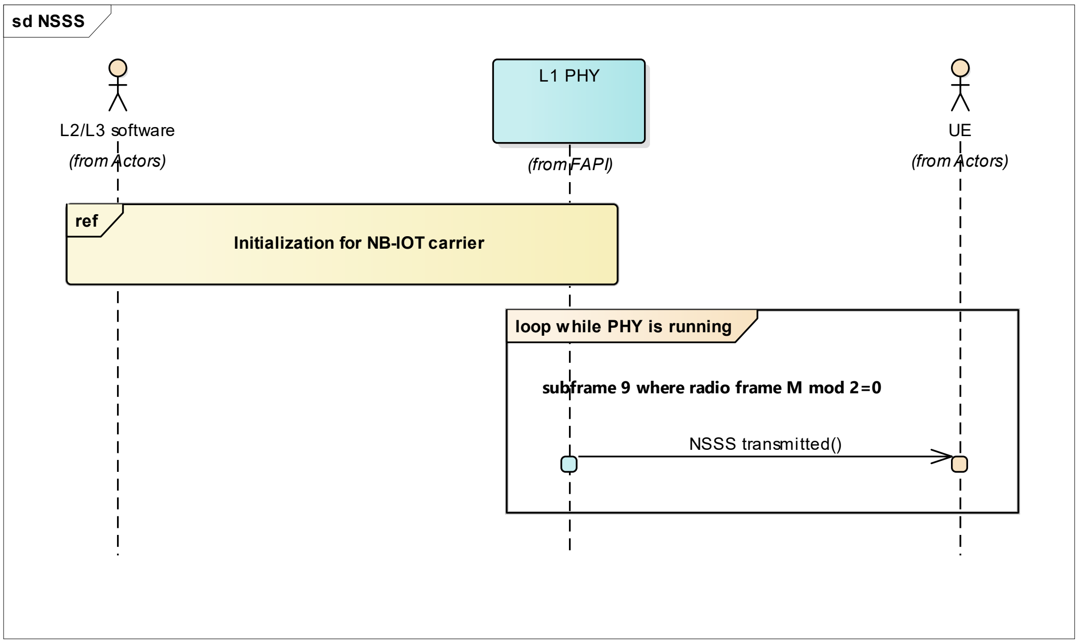


Figure 3-51 NSSS procedure

3.2.4.3 Uplink

RACH

The RACH transport channel is used by the UE to send data to the eNB when it has no scheduled resources. Also, the L2/L3 software can indicate to the UE that it should initiate a RACH procedure. In LTE the occurrence of the RACH follows a pattern advertised on the System Information broadcast messages. For NB-IOT the storage of this information in held in the PHY.

The RACH procedure is shown in Figure 3-52. To configure a RACH procedure the L2/L3 software must provide the following information:

- The RACH pattern is stored in the PHY during the initialization procedure.
- If semi-static information is held in the MAC then a RACH subframe is indicated by including a NRACH PDU in the `UL_CONFIG.request` message
- If a UE decides to RACH, and a preamble is detected by the PHY:
  - The PHY will include 1 RACH PDU in the `NRACH.indication` message. This RACH PDU includes all detected preambles
- If no RACH preamble is detected by the PHY, then no `NRACH.indication` message is sent
- For NB-IOT the RACH is configured with 1 or more RACH preamble repetitions. This is in addition to the 1 or more RACH preamble transmissions also configured for LTE.

The NRACH is only transmitted in anchor carriers

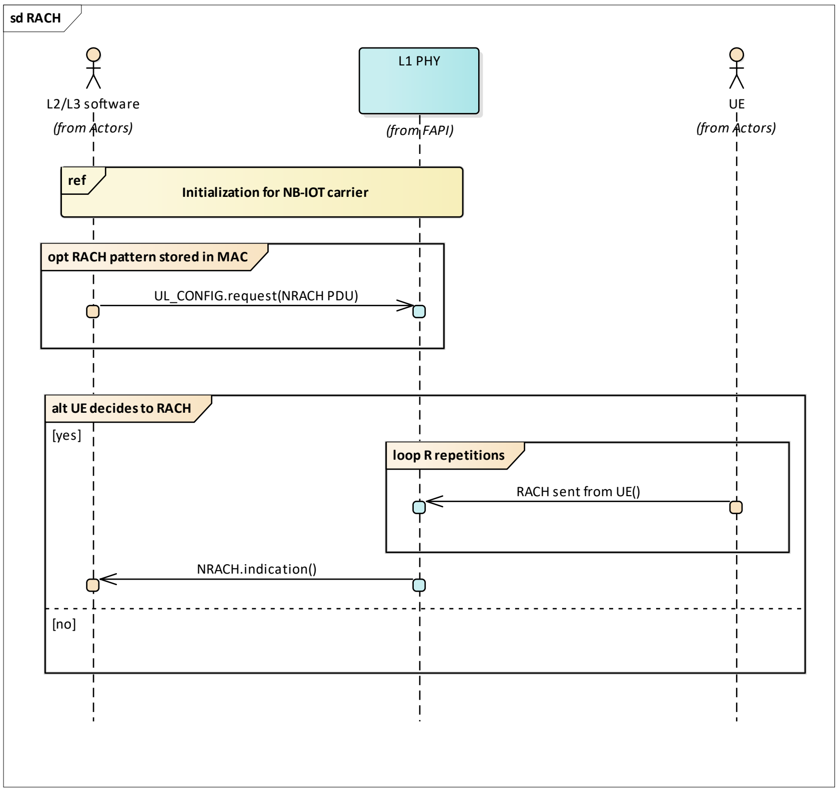


Figure 3-52 NB-IOT RACH procedure

ULSCH

The ULSCH transport channel is used to send data from the UE to the eNB. HARQ is always applied on the ULSCH transport channel.

The procedure for the ULSCH transport channel is shown in Figure 3-53. To transmit an ULSCH PDU the L2/L3 software must provide the following information:

- Within the `HI_DCI0.request` for subframe N a NPDCCH DCI UL PDU is included. The DCI Format N0 PDU contains control information regarding the UL frame transmission being scheduled. The DCI PDU is transmitted one or more times depending on the DCI subframe repetition value provided by the MAC.
- In `UL_CONFIG.request` for subframe N+delay an ULSCH PDU is included. The timing of this message is dependent on the scheduling delay value specified in the DCI Format N0 PDU. There are multiple possible ULSCH PDUs that can be used to schedule ULSCH data on the uplink.
- If the Data Report Mode TLV = 0 in the `CONFIG.request` message, then:
    - The PHY will return CRC information for the received data in a the `CRC.indication` message
- The PHY will return the received uplink data in the `RX_ULSCH.indication` message. The `RX_ULSCH.indication` message repeats the CRC information given in the `CRC.indication` message.

For NB-IOT the UL HARQ cycle is asynchronous, therefore, the ACK/NACK response is submitted to the PHY using a `HI_DCI0.request` message in subframe after a minimum of 3ms following the final uplink subframe of the ULSCH PDU, including repetitions.

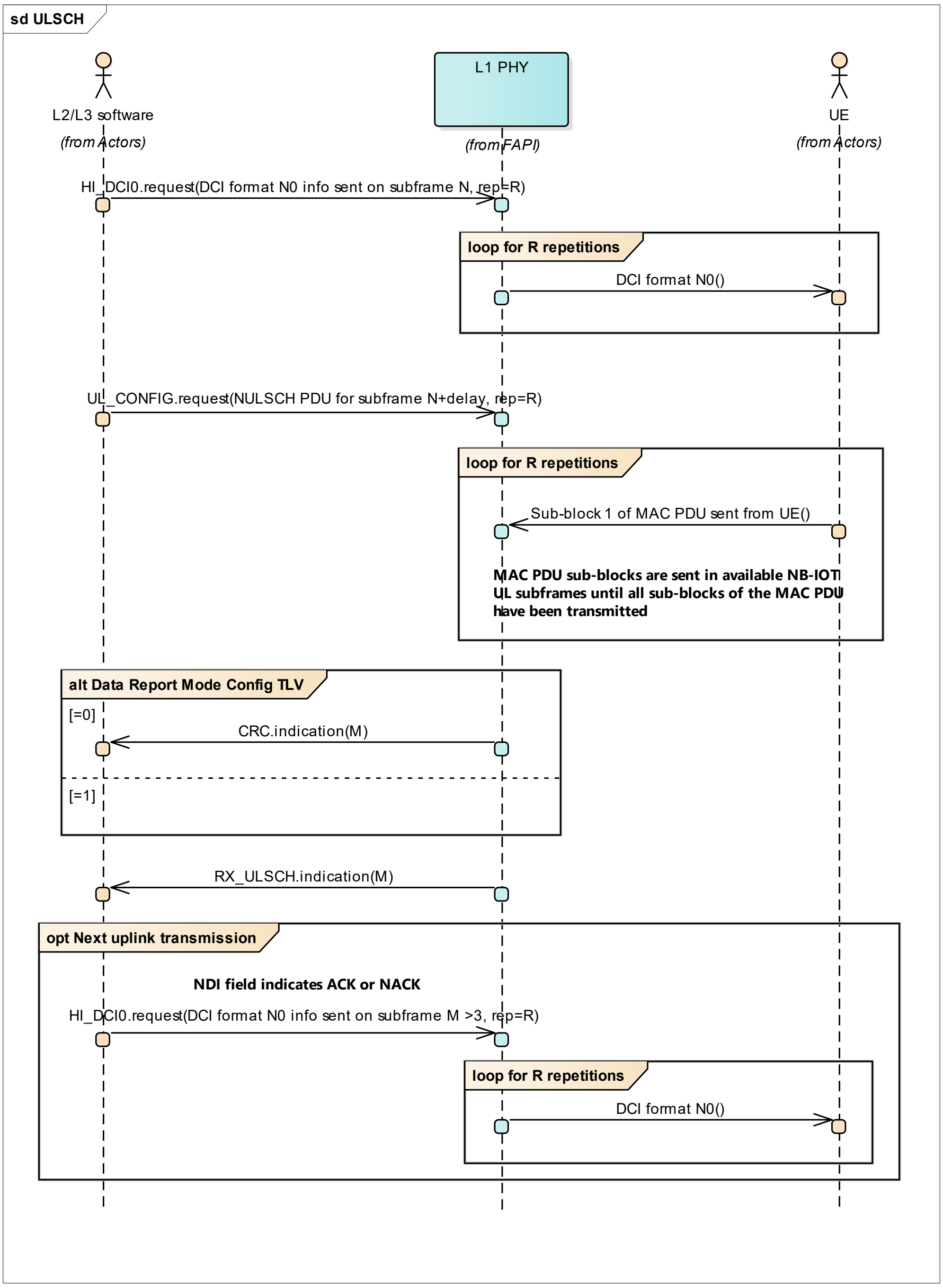


Figure 3-53 NB-IOT ULSCH procedure

3.3 FAPI messages

This section provides a description of the L1 API message formats. It defines the L1 API message header, the message bodies and the error codes associated with the L1 API.

3.3.1 General message format

The general message format of the L1 API is shown in Table 3-3, where it can be seen that each L1 API message consists of a header followed by a message body.

The generic header consists of a message type ID, a message body length and a vendor-specific body length. The current list of message types is given in Table 3-4. The L1 API messages follow a standard naming convention where:

- All `.request` messages are sent from the L2/L3 software to the PHY.
- All `.response` messages are sent from the PHY to the L2/L3 software. These are sent in response to a .request.
- All `.indication` messages are sent from the PHY to the L2/L3 software. These are sent asynchronously.

The message body is different for each message type; however, each message body obeys the following rules:

- The first field in each response message is an error code. For each message it is indicated which error codes can be returned. A full list of error codes is given in Section 3.3.3.

A full description of each message body is given in the remainder of Section 3.3.

The API mechanism can use either little-endian byte order, or big-endian byte order. The selection of byte ordering is implementation specific. However, the LSB of the bit stream shall be the LSB of the field defined in the API. The location of a 12-bit bitstream within a 16-bit value would be:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
msb					bitstream										lsb

This document assumes that the API messages are transferred using a reliable in-order delivery mechanism.

Type	Description
uint8_t	Message type ID
uint8_t	Length of vendor-specific message body (bytes)
uint16_t	Length of message body (bytes)
Message body	
Vendor-specific message body	

Table 3-3 General L1 API message format

Message	Value	Message Body Definition
`PARAM.request`	0x00	See Section 3.3.2.1
`PARAM.response`	0x01	See Section 3.3.2.1
`CONFIG.request`	0x02	See Section 3.3.2.2
`CONFIG.response`	0x03	See Section 3.3.2.2
`START.request`	0x04	See Section 3.3.2.3
`STOP.request`	0x05	See Section 3.3.2.4

Message	Value	Message Body Definition
STOP.indication	0x06	See Section 3.3.2.4
UE_CONFIG.request	0x07	See Section 3.3.2.5
UE_CONFIG.response	0x08	See Section 3.3.2.5
ERROR.indication	0x09	See Section 3.3.2.7
UE_RELEASE.request	0x0a	See Section 3.3.2.6
UE_RELEASE.response	0x0b	See Section 3.3.2.6
RESERVED	0x0c-0x7f	
DL_CONFIG.request	0x80	See Section 4.2.1
UL_CONFIG.request	0x81	See Section 4.2.2
SUBFRAME.indication	0x82	See Section 3.3.2.9
HI_DCI0.request	0x83	See Section 4.2.3
TX.request	0x84	See Section 4.2.5.1
HARQ.indication	0x85	See Section 4.2.6.2
CRC.indication	0x86	See Section 4.2.6.3
RX_ULSCH.indication	0x87	See Section 4.2.6.1
RACH.indication	0x88	See Section 4.2.6.6
SRS.indication	0x89	See Section 4.2.6.7
RX_SR.indication	0x8a	See Section 4.2.6.4
RX_CQI.indication	0x8b	See Section 4.2.6.5
LBT_DL_CONFIG.request	0x8c	See Section 4.2.7
LBT_DL.indication	0x8d	See Section 4.2.8
NB_HARQ.indication	0x8e	See Section 4.2.6.10
NRACH.indication	0x8f	See Section 4.2.6.11
RESERVED	0x9a-0xff	

Table 3-4 FAPI P5 & P7 message types

3.3.2 P5 Configuration messages

The configuration messages are used by the L2/L3 software to control and configure the PHY.

3.3.2.1 PARAM

The PARAM message exchange was described in Figure 3-5.

PARAM.request

This message can be sent by the L2/L3 when the PHY is in the IDLE state and, optionally, the CONFIGURED state. If it is sent when the PHY is in the RUNNING state, a MSG_INVALID_STATE error is returned in `PARAM.response`. No message body is defined for `PARAM.request`. The message length in the generic header = 0.

PARAM.response

The `PARAM.response` message is given in Table 3-5. From this table it can be seen that `PARAM.response` contains a list of TLVs providing information about the PHY. When the PHY is in the IDLE state this information relates to the PHY's overall capability. When the PHY is in the CONFIGURED state this information relates to the current configuration.

The full list of TLVs is given in Section 4.1.1.1. However, the set of TLVs which will be returned in the `PARAM.response` message depends on whether the PHY is TDD, FDD (including SDL frame type 3 introduced for LAA in Release 13) or NB-IOT, and on the current operational state of the PHY. Table 3-6 to Table 3-11 provide clarification on when a TLV will be included. Note: There is no requirement for the PHY to return the TLVs in the order specified in the Table.

Field	Type	Description
Error Code	uint8_t	See Table 3-39.
Number of TLVs	uint8_t	Number of TLVs contained in the message body.
TLVs	Variable	See Table 3-6 to Table 3-11.

Table 3-5 **`PARAM.response` message body**

Description	Tag
PHY State	0xFA
Downlink Bandwidth Support	0xC8
Uplink Bandwidth Support	0xC9
Downlink Modulation Support	0xCA
Uplink Modulation Support	0xCB
PHY Antenna Capability	0xCC
Release Capability	0xCD
MBSFN Capability	0xCE (included if release 9 supported)

Table 3-6 **TLVs included in `PARAM.response` for TDD when PHY is in IDLE state**

Description	Tag
PHY State	0xFA
Downlink Bandwidth Support	0xC8
Uplink Bandwidth Support	0xC9
Downlink Modulation Support	0xCA
Uplink Modulation Support	0xCB
PHY Antenna Capability	0xCC
Release Capability	0xCD
MBSFN Capability	0xCE (included if release 9 supported)

Description	Tag
Duplexing Mode	0x01
PCFICH Power Offset	0x02
P-B	0x03
DL Cyclic Prefix Type	0x04
UL Cyclic Prefix Type	0x05
RF Config	All TLVs in this grouping
PHICH Config	All TLVs in this grouping
SCH Config	All TLVs in this grouping
PRACH Config	All TLVs in this grouping
PUSCH Config	All TLVs in this grouping
PUCCH Config	All TLVs in this grouping
SRS Config	All TLVs in this grouping
Uplink Reference Signal Config	All TLVs in this grouping
TDD Frame Structure Config	All TLVs in this grouping
Data Report Mode	0xF0

Table 3-7 TLVs included in `PARAM.response` for TDD when PHY is in CONFIGURED state

Description	Tag
PHY State	0xFA
Downlink Bandwidth Support	0xC8
Uplink Bandwidth Support	0xC9
Downlink Modulation Support	0xCA
Uplink Modulation Support	0xCB
PHY Antenna Capability	0xCC
Release Capability	0xCD
MBSFN Capability	0xCE (included if release 9 supported)
LAA Capability	Included if release 13 supported. If LAA is not supported only LAA support TLV is included. If LAA support all TLVs in this group are included.
NB-IOT Capability	Included if release 13 supported. If NB-IOT is not supported only NB-IOT support TLV is included. If NB-IOT is supported all TLVs in this group are included. These TLVs can be used if a PHY can be configured as either LTE or NB-IOT, however, a PHY shall not be simultaneously configured as both.

Table 3-8 TLVs included in `PARAM.response` for FDD and SDL when PHY is in IDLE state

Description	Tag
PHY State	0xFA
Downlink Bandwidth Support	0xC8
Uplink Bandwidth Support	0xC9
Downlink Modulation Support	0xCA
Uplink Modulation Support	0xCB
PHY Antenna Capability	0xCC
Release Capability	0xCD
MBSFN Capability	0xCE (included if release 9 supported)
LAA Capability	Included if release 13 supported. If LAA is not supported only LAA support TLV is included. If LAA support all TLVs in this group are included.
Duplexing Mode	0x01
PCFICH Power Offset	0x02
P-B	0x03
DL Cyclic Prefix Type	0x04
UL Cyclic Prefix Type	0x05
RF Config	All TLVs in this grouping
PHICH Config	All TLVs in this grouping
SCH Config	All TLVs in this grouping
PRACH Config	All TLVs in this grouping
PUSCH Config	All TLVs in this grouping
PUCCH Config	All TLVs in this grouping
SRS Config	All TLVs in this grouping
Uplink Reference Signal Config	All TLVs in this grouping
Data Report Mode	0xF0

Table 3-9 TLVs included in `PARAM.response` for FDD and SDL when PHY is in CONFIGURED state

Description	Tag
PHY State	0xFA
PHY Antenna Capability	0xCC
Release Capability	0xCD
NB-IOT Capability	Included if release 13 supported.

Table 3-10 TLVs included in `PARAM.response` for NB-IOT when PHY is in IDLE state

Description	Tag
PHY State	0xFA
PHY Antenna Capability	0xCC
Release Capability	0xCD
NB-IOT Capability	All TLVs in this grouping
Reference signal power	0x0C
Tx antenna ports	0x0D
Rx Antenna ports	0x0E
SCH Config	All TLVs in this grouping
NB-IOT config	All TLVs in this grouping
Data Report Mode	0xF0

Table 3-11 TLVs included in `PARAM.response` for NB-IOT when PHY is in CONFIGURED state

PARAM errors

The error codes which may be returned in `PARAM.response` are given in Table 3-12.

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The PARAM.request was received when the PHY was in the RUNNING state.

Table 3-12 Error codes for `PARAM.response`

3.3.2.2 CONFIG

The CONFIG message exchange was described in Figure 3-6.

CONFIG.request

The `CONFIG.request` message is given in Table 3-13. From this table it can be seen that `CONFIG.request` contains a list of TLVs describing how the PHY should be configured. This message may be sent by the L2/L3 software when the PHY is in any state.

The full list of TLVs is given in Section 4.1.1.1. However, when the PHY is in the IDLE state there is a list of mandatory TLVs that must be included. For clarification Table 3-14 to Table 3-16 are provided. These indicate mandatory TLVs, which must be sent when the PHY is in IDLE state, and may be sent when the PHY is in the CONFIGURED state. The tables, also, indicate optional TLVs which may be sent when the PHY is in either the IDLE or CONFIGURED state. There is no requirement for the L2/L3 software to provide the TLVs in the order specified in the Tables.

When the PHY is in the RUNNING state a limited subset of the TLVs may be included. These TLVs are indicated in Table 3-17. Again, there is no requirement for the L2/L3 software to provide the TLVs in the order specified in the table.

Field	Type	Description
Number of TLVs	uint8_t	Number of TLVs contained in the message body.
TLVs	Variable	See Table 3-14 to Table 3-17.

Table 3-13 `CONFIG.request` **message body**

Description	Tag
Mandatory TLVs – These must be included when the PHY is in IDLE state, they may also be included when the PHY is in CONFIGURED state.	
Duplexing Mode	0x01
PCFICH Power Offset	0x02
P-B	0x03
DL Cyclic Prefix Type	0x04
UL Cyclic Prefix Type	0x05
RF Config	All TLVs in this grouping
PHICH Config	All TLVs in this grouping
SCH Config	All TLVs in this grouping
PRACH Config	All TLVs in this grouping
PUSCH Config	All TLVs in this grouping
PUCCH Config	All TLVs in this grouping
SRS Config	All TLVs in this grouping
Uplink Reference Signal Config	All TLVs in this grouping
TDD Frame Structure Config	All TLVs in this grouping
Data Report Mode	0xF0

Table 3-14 TLVs included in `CONFIG.request` **for TDD for IDLE and CONFIGURED states**

Description	Tag
Mandatory TLVs – These must be included when the PHY is in IDLE state, they may also be included when the PHY is in CONFIGURED state.	
Duplexing Mode	0x01
PCFICH Power Offset	0x02
P-B	0x03
DL Cyclic Prefix Type	0x04
UL Cyclic Prefix Type	0x05
RF Config	All TLVs in this grouping
PHICH Config	All TLVs in this grouping
SCH Config	All TLVs in this grouping

Description	Tag
PRACH Config	All TLVs in this grouping
PUSCH Config	All TLVs in this grouping
PUCCH Config	All TLVs in this grouping
SRS Config	All TLVs in this grouping
Uplink Reference Signal Config	All TLVs in this grouping
LAA config	All TLVs in this grouping if LAA support is required by L2
eMTC config	All TLVs in this grouping if eMTC support is required by L2
Data Report Mode	0xF0

Table 3-15 TLVs included in `CONFIG.request` for FDD and SDL for IDLE and CONFIGURED states

Description	Tag
Mandatory TLVs – These must be included when the PHY is in IDLE state, they may also be included when the PHY is in CONFIGURED state.	
Reference signal power	0x0C
Tx antenna ports	0x0D
Rx Antenna ports	0x0E
SCH Config	All TLVs in this grouping
NB-IOT config	All TLVs in this grouping
Data Report Mode	0xF0

Table 3-16 TLVs included in `CONFIG.request` for NB-IOT for IDLE and CONFIGURED states

Description	Tag
SFN/SF	0xF1
Other TLVs are FFS	

Table 3-17 TLVs permitted in `CONFIG.request` in the RUNNING state

CONFIG.response

The `CONFIG.response` message is given in Table 3-18. If the configuration procedure was successful then the error code returned will be MSG_OK and no TLV tags will be included. If the configuration procedure was unsuccessful then MSG_INVALID_CONFIG will be returned, together with a list of TLVs identifying the problem.

Field	Type	Description
Error Code	uint8_t	See Table 3-39.
Number of Invalid or Unsupported TLVs	uint8_t	Number of invalid or unsupported TLVs contained in the message body.

Field	Type	Description
Number of Missing TLVs	uint8_t	Number of missing TLVs contained in the message body. If the PHY is in the CONFIGURED state this will always be 0.
A list of invalid or unsupported TLVs – each TLV is presented in its entirety.		
TLV	Variable	Complete TLVs
A list of missing TLVs – each TLV is presented in its entirety		
TLV	Variable	Complete TLVs

Table 3-18 `CONFIG.response` **message body**

CONFIG errors
The error codes that can be returned in `CONFIG.response` are given in Table 3-19.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_CONFIG	The configuration provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this state.

Table 3-19 **Error codes for** `CONFIG.response`

3.3.2.3 START
The START message exchange was described in Figure 3-7.

START.request
This message can be sent by the L2/L3 when the PHY is in the CONFIGURED state. If it is sent when the PHY is in the IDLE, or RUNNING, state an `ERROR.indication` message will be sent by the PHY. No message body is defined for `START.request`. The message length in the generic header = 0.

START errors
The error codes returned in an `ERROR.indication` generated by the `START.request` message are given in Table 3-20.

Error code	Description
MSG_INVALID_STATE	The START.request was received when the PHY was in the IDLE or RUNNING state.

Table 3-20 **Error codes for** `ERROR.indication`

3.3.2.4 STOP
The STOP message exchange was described in Figure 3-8.

STOP.request
This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE, or CONFIGURED, state an `ERROR.indication` message will be sent by the PHY. No message body is defined for `STOP.request`. The message length in the generic header = 0.

STOP.indication

This message is sent by the PHY to indicate that it has successfully stopped and returned to the CONFIGURED state. No message body is defined for `STOP.indication`. The message length in the generic header = 0.

STOP errors

The error codes returned in an `ERROR.indication` generated by the `STOP.request` message are given in Table 3-21.

Error code	Description
MSG_INVALID_STATE	The STOP.request was received when the PHY was in the IDLE or CONFIGURED state.

Table 3-21 Error codes for `ERROR.indication`

3.3.2.5 UE CONFIG

The UE CONFIG message exchange was described in Figure 3-24

UE_CONFIG.request

The `UE_CONFIG.request` message is given in Table 3-22. From this table it can be seen that `UE_CONFIG.request` contains a list of TLVs describing how the PHY should be configured with UE-specific parameters. This message may be sent by the L2/L3 software when the PHY is in the RUNNING state. The message is only valid if semi-static configuration is kept in the PHY.

Field	Type	Description
Number of TLVs	uint8_t	Number of TLVs contained in the message body.
TLVs	Variable	See Table 3-23

Table 3-22 `UE CONFIG.request` message body

Description	Tag
Handle	0x01
RNTI	0x02
CQI Config	All TLVs in this grouping
ACK/NACK Config	All TLVs in this grouping
SRS Config	All TLVs in this grouping
SR Config	All TLVs in this grouping
SPS Config	All TLVs in this grouping

Table 3-23 TLVs included in `UE_CONFIG.request`

UE_CONFIG.response

The `UE_CONFIG.response` message is given in Table 3-24. If the configuration procedure was successful then the error code returned will be MSG_OK and no TLV tags will be included. If the configuration procedure was unsuccessful then MSG_INVALID_CONFIG will be returned, together with a list of TLVs identifying the problem.

Only valid if semi-static configuration is stored in the PHY.

Field	Type	Description
Error Code	uint8_t	See Table 3-39.
Number of Invalid or Unsupported TLVs	uint8_t	Number of invalid or unsupported TLVs contained in the message body.
Number of Missing TLVs	uint8_t	Number of missing TLVs contained in the message body. If the PHY is in the CONFIGURED state this will always be 0.
A list of invalid or unsupported TLVs – each TLV is presented in its entirety.		
TLV	Variable	Complete TLVs
A list of missing TLVs – each TLV is presented in its entirety		
TLV	Variable	Complete TLVs

Table 3-24 UE_`CONFIG.response` message body

UE_CONFIG errors

The error codes that can be returned in `UE_CONFIG.response` are given in Table 3-25.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The UE_CONFIG.request was received when the PHY was in the IDLE or CONFIGURED state.
MSG_INVALID_CONFIG	The configuration provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this state.

Table 3-25 Error codes for `CONFIG.response`

UE configuration TLVs

The configuration TLVs that are used in the `UE_CONFIG` message exchanges follow the format given in Table 3-26. Each TLV consists of; a Tag parameter of 1 byte, a Length parameter of 1 byte and a Value parameter. The length of the Value parameter ensures the complete TLV is a multiple of 4-bytes (32-bits).

The individual TLVs are defined in Table 3-27.

Type	Description
uint8_t	Tag
uint8_t	Length (in bytes)
variable	Value

Table 3-26 TLV format

Description	Tag	Type	Value
Handle	*0x01*	*uint32_t*	*An opaque handling to associate the received*

Description	Tag	Type	Value
			information in RX.indication
RNTI	*0x02*	*uint16_t*	The RNTI used for identifying the UE when receiving the PDU See [5] section 7.1 Value: 1 → 65535.
simultaneous PUCCH-PUSCH R10	*0x03*	*uint8_t*	Indicates if UE is configured for simultaneous transmission of PUCCH and PUSCH. Value: 0 : Not configured for simultaneous transmission 1 : Configured for simultaneous transmission
CQI config			
CQI PUCCH resource index	*0x0A*	*uint16_t*	The PUCCH resource for periodic CQI reporting. Value: 0 → 1184.
CQI PMI config index	*0x0B*	*uint16_t*	The periodic PMI reporting configuration. Value: 0 → 1023.
CQI RI config index	*0x0C*	*uint16_t*	The periodic RI reporting configuration. Value: 0 → 1023.
CQI simultaneous ACK/NACK and CQI	*0x0D*	*uint8_t*	Indicates if simultaneous transmission of CQI and ACK/NACK is allowed. Value: 0: no PUCCH Format 2a/2b 1: PUCCH Format 2a/2b can be used
CQI PUCCH resource index-P1	*0x0E*	*uint16_t*	The CQI PUCCH resource for antenna port P1. If present, the UE is configured for periodic CQI reporting on two antenna ports Value: 0 → 1184.
ACK/NACK config			
AN Repetition factor	*0x14*	*uint8_t*	The ACK/NACK repetition factor Value: 2,4,6
AN n1PUCCH-ANRep	*0x15*	*uint16_t*	The ACK/NACK repetition PUCCH resource index Value: 0→ 2047
TDD ack/nack feedback mode	*0x16*	*uint8_t*	The TDD ACK/NACK Feedback Mode Value: 0: bundling 1: multiplexing 2: Format 1b with channel selection, added in Release 10 3: Format 3, added in Release 10
Number of n3PUCCH AN List R10	*0x17*	*uint8_t*	Number of PUCCH resource index values for port P0. TPC command for PUCCH is used to index into the possible resource values.

Description	Tag	Type	Value
			Value: 1→ 4 Added in Release 10
for Number of n3PUCCH AN List R10 {			
n3PUCCH AN list R10	*0x18*	*uint16_t*	$N_{\mathrm{PUCCH}}^{(3,p)}$ for antenna port P0 defined in [9] section 10.1.2.2.2 Value: 0 →549 Added in Release 10
}			
Number of n3PUCCH AN ListP1 R10	*0x19*	*uint8_t*	Number of PUCCH resource index values for port P1. This parameter is valid only if both transmission ports (P0 and P1) have been activated. Value: 1→ 4 Added in Release 10
for number of n3PUCCH AN ListP1 R10{			
n3PUCCH AN ListP1 R10	*0x1A*	*uint16_t*	$N_{\mathrm{PUCCH}}^{(3,p)}$ for antenna port P1 defined in [9] section 10.1.2.2.2 Value: 0 →549 Added in Release 10
}			
Number of n1PUCCH AN CS List R10	*0x1B*	*uint8_t*	Value: 1,2 Added in Release 10
for number of n1PUCCH AN CS List R10{			
Number of n1PUCCH AN CS R10	*0x1C*	*uint8_t*	Number of PUCCH resource index values for PUCCH format 1b with channel selection. Value: 1→ 4 Added in Release 10
for number of n1PUCCH AN CS R10{			
n1PUCCH AN CS R10	*0x1D*	*uint16_t*	PUCCH resource index $n_{\mathrm{PUCCH},j}^{(1)}$ defined in [9] section 10.1.2.2.1 Value: 0 →2047

Description	Tag	Type	Value
			Added in Release 10
}			
}			
two Antenna Port Activated PUCCH Format 1A1B R10	*0x1E*	*uint8_t*	Indicates if UE is configured for two antenna ports for PUCCH format 1a/1b Value: 1 : True Added in Release 10
n1PUCCH AN RepP1 R10	*0x1F*	*uint16_t*	ACK/NACK repetition resource index for antenna port P1 defined in [9] section 10.1.4 Value: 0 →2047 Added in Release 10
SRS Config			
SRS bandwidth	*0x28*	*uint8_t*	SRS Bandwidth. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2 *Value: 0 → 3*
SRS hopping bandwidth	*0x29*	*uint8_t*	Configures the frequency hopping on the SRS. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2. *Value 0 → 3*
Frequency domain position	*0x2A*	*uint8_t*	Frequency-domain position, N_{RRC} This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2 Value: 0 → 23
SRS duration	*0x2B*	*uint8_t*	*The duration of the SRS configuration* Value: 0: once 1: indefinite
I_{SRS} / SRS-ConfigIndex	*0x2C*	*uint16_t*	Defines the periodicity and subframe location of the SRS. SRS Configuration Index. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.2. *Value: 0 → 1023.*
Transmission comb	*0x2D*	*uint8_t*	Configures the frequency location of the SRS. This value is fixed for a UE and allocated in RRC

Description	Tag	Type	Value
			connection setup. Value: 0 → 1
Sounding reference cyclic shift	*0x2E*	uint8_t	Configures the SRS sequence generation. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.1. Value: 0 → 7
SR Config			
SR PUCCH resource index	0x32	uint16_t	The scheduling request PUCCH resource index. Value: 0 → 2047.
SR config index	0x33	uint8_t	The scheduling request configuration index. Value: 0 → 155.
SR PUCCH resource index-P1	0x34	uint16_t	The scheduling request PUCCH resource index for antenna port P1. If present, the UE is configured for SR transmission on two antenna ports. Value: 0 → 2047. Added in Release 10.
SPS Config			
SPS DL config scheduling interval	0x3C	uint16_t	SPS Configuration Interval Value: 10,20,32,40,64,80,128,160,320,640
SPS DL n1PUCCH AN Persistent	0x3D	uint16_t	The SPS PUCCH AN Resource configuration. The TLV can be repeated four times. Value: 0 → 2047.
SPS DL n1PUCCH AN Persistent-P1	0x3E	uint16_t	The SPS PUCCH AN Resource configuration for antenna port P1. The TLV can be repeated four times. [Valid only when two Antenna Port Activated PUCCH Format 1A1BR10 is TRUE.] Value: 0 → 2047 Added in Release 10.

Table 3-27 Configuration TLVs for UE_CONFIG

3.3.2.6 UE RELEASE

The UE RELEASE message exchange was described in Figure 3-25.

UE_RELEASE.request

The `UE_RELEASE.request` message is given in Table 3-28. From this table it can be seen that `UE_RELEASE.request` contains a list of TLVs describing how the PHY should be configured with UE-specific parameters. This message may be sent by the L2/L3 software when the PHY is in the RUNNING state.

This message is used to release the semi-static information in the PHY if it is kept in the PHY.

Field	Type	Description
Number of TLVs	uint8_t	Number of TLVs contained in the message body.
TLVs	Variable	See Table 3-29

Table 3-28 `UE RELEASE.request` **message body**

Description	Tag
Handle	1
RNTI	2

Table 3-29 **TLVs included in** `UE_RELEASE.request`

UE_RELEASE.response

The `UE_RELEASE.response` message is given in Table 3-30. If the configuration procedure was successful then the error code returned will be MSG_OK and no TLV tags will be included. If the configuration procedure was unsuccessful then MSG_INVALID_CONFIG will be returned, together with a list of TLVs identifying the problem.

Field	Type	Description
Error Code	uint8_t	See Table 3-31
Number of Invalid or Unsupported TLVs	uint8_t	Number of invalid or unsupported TLVs contained in the message body.
Number of Missing TLVs	uint8_t	Number of missing TLVs contained in the message body. If the PHY is in the CONFIGURED state this will always be 0.
A list of invalid or unsupported TLVs – each TLV is presented in its entirety.		
TLV	Variable	Complete TLVs
A list of missing TLVs – each TLV is presented in its entirety		
TLV	Variable	Complete TLVs

Table 3-30 `UE_RELEASE.response` **message body**

UE_RELEASE Errors

The error codes that can be returned in `UE_RELEASE.response` are given in Table 3-31.

Error Code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The UE_RELEASE.request was received when the PHY was in the IDLE or CONFIGURED state.
MSG_INVALID_CONFIG	The configuration provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this state.

Table 3-31 Error codes for `UE_RELEASE.response`

3.3.2.7 PHY notifications

The PHY notification messages are used by the PHY to inform the L2/L3 software of an event which occurred.

ERROR.indication

This message is used to report an error to the L2/L3 software. These errors all relate to API message exchanges. The format of `ERROR.indication` is given in Table 3-32.

Field	Type	Description
Message ID	uint8_t	Indicate which message received by the PHY has an error. Values taken from Table 3-4.
Error code	uint8_t	The error code, see Section Table 3-39 for value. If the value is MSG_PDU_ERR then more detailed error information is included.
Error code dependent values	struct	The format of these bytes is dependent on the error code. See Table 3-33 to Table 3-37.

Table 3-32 `ERROR.indication` message body

Field	Type	Description
		Not used

Table 3-33 MSG_INVALID_STATE and MSG_BCH_MISSING

Field	Type	Description
Received SFN/SF	uint16_t	The SFN/SF value received in the message
Expected SFN/SF	uint16_t	The SFN/SF value the PHY was expecting to receive in the message

Table 3-34 SFN_OUT_OF_SYNC and MSG_INVALID_SFN

Field	Type	Description
Sub Error Code	uint8_t	The Sub Error Code for this message, see Section Table 3-40.
Direction	uint8_t	Indicates if this error was in a DL subframe configuration or an UL subframe configuration. 0 = DL, 1 = UL
RNTI	uint16_t	The RNTI in the received PDU. If the error occurred in a MCH, or BCH, PDU this value is set to 0
PDU Type	uint8_t	The PDU Type parameter specified in both DL subframe configuration and UL subframe configuration

Table 3-35 MSG_PDU_ERR

Field	Type	Description

Field	Type	Description
Sub Error Code	uint8_t	The Sub Error Code for this message, see Section Table 3-40.
PHICH Lowest UL RB Index	uint8_t	The PHICH RB Index parameters specified in each HI PDU

Table 3-36 MSG_HI_ERR

Field	Type	Description
Sub Error Code	uint8_t	The Sub Error Code for this message, see Section Table 3-40.
PDU Index	uint16_t	The PDU index parameter specified for each PDU

Table 3-37 MSG_TX_ERR

3.3.2.8 Subframe messages

The subframe messages are used by the L2/L3 software to control the data transmitted, or received, every 1ms subframe.

3.3.2.9 SUBFRAME

SUBFRAME.indication

The `SUBFRAME.indication` message is given in Table 3-38. It is sent from the PHY every 1ms.

Field	Type	Description
SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9

Table 3-38 `SUBFRAME.indication` message body

3.3.3 Error codes

The list of possible error codes returned in either `.response` messages or the `ERROR.indication` message is given in Table 3-39.

Value	Error code	Description
0x0	MSG_OK	Message is OK.
0x1	MSG_INVALID_STATE	The received message is not valid in the PHY's current state.
0x2	MSG_INVALID_CONFIG	The configuration provided in therequest message was invalid
0x3	SFN_OUT_OF_SYNC	The DL_CONFIG.request was received with a different SFN than the PHY expected.
0x4	MSG_SUBFRAME_ERR	An error was received in DL_CONFIG.request or UL_CONFIG.request. The sub-error code should be analyzed
0x5	MSG_BCH_MISSING	A BCH PDU was expected in the

Value	Error code	Description
		DL_CONFIG.request message for this subframe. However, it was not present.
0x6	MSG_INVALID_SFN	The received HI_DCI0.request or TX.request message included a SFN/SF value which was not expected. The message has been ignored.
0x7	MSG_HI_ERR	An error was received in HI_DCI.request. The sub-error code should be analyzed
0x8	MSG_TX_ERR	An error was received in TX.request. The sub-error code should be analyzed
0x9	LBT_NO_PDU_IN_DL_REQ	No PDU was received in the LBT_DL_CONFIG.request message. Message was empty
0xa	LBT_NO_VALID_CONFIG_REQ_RECIEVED	LBT_DL_CONFIG.request was sent before a valid CONFIG.request was received
0xb	FAPI_E_LBT_SF_SFN_PASSED_END_SF_SFN	Start SF/SFN passed End SF/SFN
0xc	FAPI_E_LBT_OVERLAP	The scheduled LBT ovarlaps with previous LBT or TXOP.
0xd	MSG_BCH_PRESENT	A BCH PDU was not expected in the DL_CONFIG.request for the LAA cell for this subframe. However, it was present.
0xe	NBIOT_UNEXPECTED_REQ	A request message received for PHY to transmit or receive a PDU. However, transmission or reception of the previous PDU is still ongoing.

Table 3-39 L1 API error codes

3.3.3.1 Sub error codes

If the ERROR code is MSG_ SUBFRAME _ERR then subframe error codes are provided. These sub error codes are given in Table 3-40.

Value	Error code	Description
FFS		

Table 3-40 L1 API subframe error codes

4. Common Messages (P5 & P7)

4.1 P5 PHY Configuration Messages

4.1.1.1 Configuration TLVs

Configuration TLVs are used in the PARAM and CONFIG message exchanges with the PHY.

For nFAPI the TLV format is given in Table 4-1. Each TLV consists of; a Tag parameter of 2 bytes, a Length parameter of 2 bytes and a Value parameter.

For FAPI TLV the TLV format is given in Table 4-2. Each TLV consists of; a Tag parameter of 1 byte, a Length parameter of 1 byte and a Value parameter. The length of the Value parameter ensures the complete TLV is a multiple of 4-bytes (32-bits).

The TLVs are defined in Table 4-3.

Type	Description
uint16_t	Tag. In Table 4-3 some Tags are defined as 0xzz. For nFAPI these tags become 0x00zz.
uint16_t	Length (in bytes)
variable	Value

Table 4-1 nFAPI TLV format

Type	Description
uint8_t	Tag.
uint8_t	Length (in bytes)
uint16_t	Value

Table 4-2 FAPI TLV format

Tag	Description	Type	Value	FAPI	nFAPI
These TLVs are used by the L2/L3 software to configure a physical parameter in L1. The Tag values in this Table are given in hexadecimal format.					
	Subframe config			TLV	TLV
0x01 (FAPI) 0x0001 (nFAPI)	Duplexing mode	uint16_t	Type of duplexing mode Value : 0 : TDD, 1 : FDD, 2 : HD_FDD	TLV	TLV
0x02	PCFICH power offset	uint16_t	The power per antenna of the PCFICH with respect to the reference signal. Value: 0-> 10000, represents -6dB to 4dB in steps 0.001dB	TLV	TLV
0x03	P-B	uint16_t	Refers to downlink power allocation. See [9] section 5.2	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value is an index into the referenced table. Value: 0 → 3		
0x04	DL cyclic prefix type	uint16_t	Cyclic prefix type, used for DL See [11] section 5.2.1 0: CP_NORMAL, 1: CP_EXTENDED.	TLV	TLV
0x05	UL cyclic prefix type	uint16_t	Cyclic prefix type, used for UL See [11] section 5.2.1 0: CP_NORMAL, 1: CP_EXTENDED.	TLV	TLV
	RF config				
0x0A	Downlink channel bandwidth	uint16_t	Downlink channel bandwidth in resource blocks. See [10] section 5.6. Value: 6,15, 25, 50, 75, 100	TLV	TLV
0x0B	Uplink channel bandwidth	uint16_t	Uplink channel bandwidth in resource blocks. See [10] section 5.6 Value: 6,15, 25, 50, 75,100	TLV	TLV
0x0C	Reference signal power	uint16_t	Normalized value levels (relative) to accommodate different absolute Tx Power used by eNb. Value: 0 → 255 Representing 0dB to -63.75dB in -0.25dB steps	TLV	TLV
0x0D	Tx antenna ports	uint16_t	The number of cell specific or NB transmit antenna ports. See [11] section 6.2.1 for LTE and section 10.2.6 for NB-IOT. Value:1,2,4,8,16 for LTE 1,2 for NB-IOT	TLV	TLV
0x0E	Rx Antenna ports	uint16_t	The number of cell specific or NB receive	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			antenna ports. See [11] section 6.2.1 for LTE and section 10.2.6 for NB-IOT Value: 1,2,4,8,16 for LTE 1,2 for NB-IOT		
	PHICH config				
0x14	PHICH resource	uint16_t	The number of resource element groups used for PHICH. See [11] section 6.9. 0: PHICH_R_ONE_SIXTH 1: PHICH_R_HALF 2: PHICH_R_ONE 3: PHICH_R_TWO	TLV	TLV
0x15	PHICH duration	uint16_t	The PHICH duration for MBSFN and non-MBSFN sub-frames. See [11] section 6.9 0: PHICH_D_NORMAL 1: PHICH_D_EXTENDED	TLV	TLV
0x16	PHICH power offset	uint16_t	The power per antenna of the PHICH with respect to the reference signal. Value: 0-> 10000, represents -6dB to 4dB in steps 0.001dB	TLV	TLV
	SCH config				
0x1E	Primary synchronization signal EPRE/EPRERS	uint16_t	The power of synchronization signal with respect to the reference signal, (PSS for LTE cell, NPSS for NB-IOT cell) Value: 0 → 10000 represents -6dB to 4dB in step 0.001dB	TLV	TLV
0x1F	Secondary synchronization signal EPRE/EPRERS	uint16_t	The power of synchronization signal with respect to the reference signal, (SSS for LTE cell, NSSS for NB-IOT cell) Value: 0 → 10000 represents -6dB to 4dB in step 0.001dB	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
0x20	Physical cell ID	uint16_t	The Cell ID sent with the synchronization signal. . (N_{ID}^{cell} for LTE cell, N_{ID}^{Ncell} for NB-IOT cell) See [11] section 6.11 for LTE cell, section 10.2.7 for NB-IOT cell. Value: 0 → 503	TLV	TLV
	PRACH config				
0x28	Configuration index	uint16_t	Provides information about the location and format of the PRACH. See [11] section 5.7. Table 5.7.1-2 for FDD, Table 5.7.1-3 for TDD The value is an index into the referenced tables. Value: 0 → 63	TLV	TLV
0x29	Root sequence index	uint16_t	PRACH Root sequence index. See [11] section 5.7.2. Value: 0 → 837	TLV	TLV
0x2A	Zero correlation zone configuration	uint16_t	Equivalent to N_{cs}, see [11] section 5.7.2. TDD: 0 → 6 FDD: 0 → 15	TLV	TLV
0x2B	High speed flag	uint16_t	Indicates if unrestricted, or restricted, set of preambles is used. See [11] section 5.7.2. 0: HS_UNRESTRICTED_SET 1: HS_RESTRICTED_SET	TLV	TLV
0x2C	Frequency offset	uint16_t	The first physical resource block available for PRACH. see [11] section 5.7.1 Value: 0 → UL_channel_bandwidth – 6	TLV	TLV
	PUSCH config				
0x32	Hopping mode	uint16_t	If hopping is enabled indicates the type of hopping used.	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			See [11] section 5.3.4 0: HM_INTER_SF 1: HM_INTRA_INTER_SF		
0x33	Hopping offset	uint16_t	The offset used if hopping is enabled. See [11] section 5.3.4 Value: 0 → 98	TLV	TLV
0x34	Number of sub-bands	uint16_t	The number of sub-bands used for hopping. See [11] section 5.3.4. Value: 1 → 4	TLV	TLV
	PUCCH config				
0x3C	Delta PUCCH Shift	uint16_t	The cyclic shift difference. See [11] section 5.4.1. Value: 1 → 3	TLV	TLV
0x3D	N_CQI RB	uint16_t	The bandwidth, in units of resource blocks, that is available for use by PUCCH formats 2/2a/2b transmission in each slot. See Section 5.4 in [11]. Value: 0 → 98	TLV	TLV
0x3E	N_AN CS	uint16_t	The number of cyclic shifts used for PUCCH formats 1/1a/1b in a resource block with a mix of formats 1/a/1/ab and 2/2a/2b. See Section 5.4 in [11]. Value: 0 → 7	TLV	TLV
0x3F	N1Pucch-AN	uint16_t	$N_{\mathrm{PUCCH}}^{(1)}$, see [9] section 10.1 Value: 0 → 2047	TLV	TLV
	SRS config				
0x46	Bandwidth configuration	uint16_t	The available SRS bandwidth of the cell. See [11] section 5.5.3 The value is an index into the referenced table. Value: 0 → 7	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
0x47	MaxUpPTS	uint16_t	Used for TDD only and indicates how SRS operates in UpPTS subframes. See [11] section 5.5.3.2 and [9] section 8.2 0: Disabled 1: Enabled	TLV	TLV
0x48	SRS subframe configuration	uint16	The subframe configuration. Needed if semi-static configuration is held in PHY. Value: 0 → 15	TLV	TLV
0x49	SRS AckNack SRS simultaneous transmission	uint8	Indicates if SRS and ACK/NACK can be received in the same subframe. Needed if semi-static configuration is held in PHY. 0: no simultaneous transmission 1: simultaneous transmission	TLV	TLV
	Uplink reference signal config				
0x50	Uplink RS hopping	uint16_t	Indicates the type of hopping to use. See [11] section 5.5.1. 0: RS_NO_HOPPING 1: RS_GROUP_HOPPING 2: RS_SEQUENCE_HOPPING	TLV	TLV
0x51	Group assignment (delta sequence-shift pattern)	uint16_t	The sequence shift pattern used if group hopping is enabled. See [11] section 5.5.1 Values: 0 → 29	TLV	TLV
0x52	Cyclic Shift 1 for DMRS	uint16_t	Specifies the cyclic shift for the reference signal used in the cell. See [11] section 5.5.1. The value is an index into the referenced table. Value: 0 → 7	TLV	TLV
	TDD frame structure config				
0x5A	Subframe assignment	uint16_t	For TDD mode only,	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			indicates the DL/UL subframe structure. See [11] section 4.2. Value: 0 → 6		
0x5B	Special sub-frame patterns	uint16_t	For TDD mode only. Length of fields DwPTS, GP and UpPTS. See [11] section 4.2. Value: 0 → 9	TLV	TLV
	LAA config				
0x64	ED Threshold for LBT for PDSCH	uint16_t	Indicates the energy detection threshold in dBm for LBT for PDSCH. $X_{\text{Thresh\_max}}$ defined in [9] section 15.1.4 Values: 0 → 70 representing -100dBm to -30dBm in 1dBm steps.	TLV	TLV
0x65	ED Threshold for LBT for DRS	uint16_t	Indicates the energy detection threshold in dBm for LBT for DRS as defined in [9] section 15.1.4 Values: 0 → 70 representing -100dBm to -30dBm in 1dBm steps.	TLV	TLV
0x66	PD Threshold	uint16_t	Indicates the preamble detection threshold in dBm, if the L1 capabilities support PD. If L2 doesn't intend L1 to use PD for LBT, the value shall be all "1s" (65535) Values: 0 → 70 representing -100dBm to -30dBm in 1dBm steps.	TLV	TLV
0x67	Multi carrier type	uint16_t	Indicates multi carrier type configuration of L1 (according to L1 capabilities and L2 scheduler requirements as defined in [9] section 15.1.5) 0 – no multi carrier support 1 – Mode A1 2 – Mode A2 3 – Mode B1 4 – Mode B2	TLV	TLV
0x68	Multi carrier TX	uint16_t	Indicates multi carrier	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			transmission configuration of L1 (according to type if supporting multi carrier). Self-deferral is described as part of the state machine in [9] section 15.1.1 0 – Mutual transmission (self-deferral support for current carrier) 1 – transmit on channel access win (no self-deferral)		
0x69	Multi carrier freeze	uint16_t	Indicates multi carrier freeze, configuration of L1 (applicable only to type A type if supporting multi carrier) as described in [9] section 15.1.5.1.1 0 –absence of other technology isn't guaranteed 1 – absence of other technology is guaranteed	TLV	TLV
0x6A	Tx antenna ports for DRS	uint16_t	The number of cell specific transmit antenna ports within the DRS occasions as described in [11] section 6.11A. Value:1,2,4	TLV	TLV
0x6B	Transmission power for DRS	uint16_t	Offset of cell specific Reference signals power within DRS occasions to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.	TLV	TLV
	eMTC config				
0x78	*PBCH Repetitions enable R13*	uint16_t	*Enable / Disable PBCH repetitions as defined in release 13 of* [11] *section 6.6.4* *Value: 0 = disable, 1= enable* ***Note:*** *PBCH repetition behaviour is explained in section 3.2.2.6 BCH PDU*	TLV	TLV
0x79	PRACH CAT-M Root sequence	uint16_t	PRACH Root sequence	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
	index		index. See [11] section 5.7.2. Value: 0 → 837		
0x7A	PRACH CAT-M Zero correlation zone configuration	uint16_t	Equivalent to N_{cs}, see [11] section 5.7.2. TDD: 0 → 6 FDD: 0 → 15	TLV	TLV
0x7B	PRACH CAT-M High speed flag	uint16_t	Indicates if unrestricted, or restricted, set of preambles is used. See [11] section 5.7.2. 0: HS_UNRESTRICTED_SET 1: HS_RESTRICTED_SET	TLV	TLV
0x7C	PRACH CE level #0 Enable	uint16_t	Enable \ Disable CE level #0. 0 = Disable. 1 = Enable.	TLV	TLV
0x7D	PRACH CE level #0 Configuration index	uint16_t	Provides information about the location and format of the PRACH. See [8] section 5.7. Table 5.7.1-2 for FDD, Table 5.7.1-3 for TDD The value is an index into the referenced tables. Value: 0 → 63	TLV	TLV
0x7E	PRACH CE level #0 Frequency offset	uint16_t	The first physical resource block available for PRACH for each CE. See [11] section 5.7.1 Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x7F	PRACH CE level #0 Number of repetitions per attempt	uint16_t	Number of PRACH repetitions per attempt for each CE level. See [11]. Value: 1, 2, 4, 8, 16, 32, 64, 128.	TLV	TLV
0x80	CE level #0 Starting subframe periodicity	uint16_t	Optional. PRACH starting subframe periodicity, expressed in number of slots available for preamble transmission (PRACH opportunities), see [11].	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Starting subframe periodicity should be larger than or equal to the Number of PRACH repetitions per attempt for each CE level. Value: 2, 4, 8, 16, 32, 64, 128, 256. Value 0xFFFF indicates Prach start parameter should be ignored		
0x81	PRACH CE level #0 Hopping Enable	uint16_t	Enable \ Disable PRACH frequency hopping for each CE level. Value: 0 = Disable. 1 = Enable.	TLV	TLV
0x82	PRACH CE level #0 Hopping Offset	uint16_t	Valid in case "PRACH Hopping Enable" is enabled. Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x83	PRACH CE level #1 Enable	uint16_t	Enable \ Disable CE level #1. 0 = Disable. 1 = Enable.	TLV	TLV
0x84	PRACH CE level #1 Configuration index	uint16_t	Provides information about the location and format of the PRACH. See [8] section 5.7. Table 5.7.1-2 for FDD, Table 5.7.1-3 for TDD The value is an index into the referenced tables. Value: 0 → 63	TLV	TLV
0x85	PRACH CE level #1 Frequency offset	uint16_t	The first physical resource block available for PRACH for each CE. See [11] section 5.7.1 Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x86	PRACH CE level #1 Number of repetitions per attempt	uint16_t	Number of PRACH repetitions per attempt for each CE level. See [11]. Value: 1, 2, 4, 8, 16, 32, 64, 128.	TLV	TLV
0x87	PRACH CE level #1 Starting	uint16_t	Optional.	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
	subframe periodicity		PRACH starting subframe periodicity, expressed in number of slots available for preamble transmission (PRACH opportunities), see [11]. Starting subframe periodicity should be larger than or equal to the Number of PRACH repetitions per attempt for each CE level. Value: 2, 4, 8, 16, 32, 64, 128, 256. Value 0xFFFF indicates Prach start parameter should be ignored		
0x88	PRACH CE level #1 Hopping Enable	uint16_t	Enable \ Disable PRACH frequency hopping for each CE level. Value: 0 = Disable. 1 = Enable.	TLV	TLV
0x89	PRACH CE level #1 Hopping Offset	uint16_t	Valid in case "PRACH Hopping Enable" is enabled. Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x8A	CE level #2 Enable	uint16_t	Enable \ Disable CE level #2. 0 = Disable. 1 = Enable.	TLV	TLV
0x8B	PRACH CE level #2 Configuration index	uint16_t	Provides information about the location and format of the PRACH. See [8] section 5.7. Table 5.7.1-2 for FDD, Table 5.7.1-3 for TDD The value is an index into the referenced tables. Value: 0 → 63	TLV	TLV
0x8C	PRACH CE level #2 Frequency offset	uint16_t	The first physical resource block available for PRACH for each CE. See [11] section 5.7.1 Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x8D	PRACH CE level #2 Number of repetitions per attempt	uint16_t	Number of PRACH repetitions per attempt	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			for each CE level. See [11]. Value: 1, 2, 4, 8, 16, 32, 64, 128.		
0x8E	PRACH CE level #2 Starting subframe periodicity	uint16_t	Optional. PRACH starting subframe periodicity, expressed in number of slots available for preamble transmission (PRACH opportunities), see [11]. Starting subframe periodicity should be larger than or equal to the Number of PRACH repetitions per attempt for each CE level. Value: 2, 4, 8, 16, 32, 64, 128, 256. Value 0xFFFF indicates Prach start parameter should be ignored	TLV	TLV
0x8F	PRACH CE level #2 Hopping Enable	uint16_t	Enable \ Disable PRACH frequency hopping for each CE level. Value: 0 = Disable. 1 = Enable.	TLV	TLV
0x90	PRACH CE level #2 Hopping Offset	uint16_t	Valid in case "PRACH Hopping Enable" is enabled. Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x91	PRACH CE level #3 Enable	uint16_t	Enable \ Disable CE level #3. 0 = Disable. 1 = Enable.	TLV	TLV
0x92	PRACH CE level #3 Configuration index	uint16_t	Provides information about the location and format of the PRACH. See [8] section 5.7. Table 5.7.1-2 for FDD, Table 5.7.1-3 for TDD The value is an index into the referenced tables.	TLV	TLV
0x93	PRACH CE level #3 Frequency offset	uint16_t	The first physical resource block available for PRACH for each CE. See [11] section 5.7.1	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value: 0 → 94 (UL_channel_bandwidth – 6)		
0x94	PRACH CE level #3 Number of repetitions per attempt	uint16_t	Number of PRACH repetitions per attempt for each CE level. See [11]. Value: 1, 2, 4, 8, 16, 32, 64, 128.	TLV	TLV
0x95	PRACH CE level #3 Starting subframe periodicity	uint16_t	Optional. PRACH starting subframe periodicity, expressed in number of slots available for preamble transmission (PRACH opportunities), see [11]. Starting subframe periodicity should be larger than or equal to the Number of PRACH repetitions per attempt for each CE level. Value: 2, 4, 8, 16, 32, 64, 128, 256. Value 0xFFFF indicates Prach start parameter should be ignored	TLV	TLV
0x96	PRACH CE level #3 Hopping Enable	uint16_t	Enable \ Disable PRACH frequency hopping for each CE level. Value: 0 = Disable. 1 = Enable.	TLV	TLV
0x97	PRACH CE level #3 Hopping Offset	uint16_t	Valid in case “PRACH Hopping Enable” is enabled. Value: 0 → 94 (UL_channel_bandwidth – 6)	TLV	TLV
0x98	PUCCH Interval-ULHoppingConfigCommonModeA	uint16_t	$N_{NB}^{ch,UL}$ for UEModeA. See [11] section 5.4.3 Value: FDD: 1, 2, 4, 8; TDD: 1, 5, 10, 20	TLV	TLV
0x99	PUCCH Interval-ULHoppingConfigCommonModeB	uint16_t	$N_{NB}^{ch,UL}$ for UEModeB. See [11] section 5.4.3 Value: FDD: 2, 4, 8, 16; TDD: 5, 10, 20, 40	TLV	TLV
	NB-IOT config				

Tag	Description	Type	Value	FAPI	nFAPI
0xA5	Operating Mode	uint16_t	NB-IOT operating mode 0: In-band with same PCI 1: In-band with different PCI 2: Guard band mode 3: Stand-alone mode	TLV	TLV
0xA6	Anchor	uint16_t	Indicates whether the NB-IOT carrier is an "Anchor" (containing NPSS, NSSS, NPBCH, NPRACH) or a "non-anchor" NB-IOT carrier (containing only NPDCCH, NPDSCH, NPUSCH) 0: non anchor NB-IOT carrier 1: anchor NB-IOT carrier	TLV	TLV
0xA7	PRB index	uint16_t	Applicable for a guard-band and in-band operating mode (0-2). Indicates PRB index n'_{PRB} with regard to the middle PRB in the LTE system (-7.5, +7.5, -2.5, +2.5 KHz). This information (not same values) is delivered in NPBCH Value - For operating mode 0 & 1 (in-band) 5 bits: 0-31 according to Table 16.8-1 in TR36.213 (Same as eutra-CRS-SequenceInfo-r13 in [4]) - For operating mode 2 (guard-band) 4 bits indicating PRBs: 0-4 indicating lower 5 PRBs 5-9 indicate upper 5 PRBs 10-15 reserved	TLV	TLV
0xA8	Control Region Size	uint16_t	Applicable for Operating mode = 0 (in-band). Indicates the control region size of the LTE cell. This field is delivered in NPBCH	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			value: 1-4: control region size		
0xA9	Assumed CRS #APs	uint16_t	Applicable for Operating mode = 1 (in-band with different PCI). Used in rate matching and RE mapping NPDSCH to UEs, using their assumption on CRS REs (not necessarily the CRS REs) This field is delivered in NPBCH. Value: 0 indicates same CRS APs as NRS APs 1: indicates four CRS APs	TLV	TLV
0xAA	NPRACH Config #0 Enable	uint16_t	Enable \ Disable Config #0 0 = Disable. 1 = Enable.	TLV	TLV
0xAB	NPRACH Config #0 SF Periodicity	uint16_t	Periodicity of an NPRACH resource. See [11] section 10.1.6 Value: 0→7 for 40, 80, 160, 240, 320, 640, 1280, 2560	TLV	TLV
0xAC	NPRACH Config #0 StartTime	uint16_t	NPRACH resource starting time after period. See [11] section 10.1.6 Value: 0→7 for 8, 16, 32, 64, 128, 256, 512, 1024 FFFF to ignore	TLV	TLV
0xAD	NPRACH Config #0 Subcarrier offset	uint16_t	Frequency location of an NPRACH resource within a PRB. See [11] section 10.1.6 Value: 0→6 for 0, 12, 24, 36, 2, 18, 34	TLV	TLV
0xAE	NPRACH Config #0 number of subcarriers	uint16_t	Number of subcarriers in an NPRACH resource. See [11] section 10.1.6 Value: 0→3 for 12, 24, 36, 48	TLV	TLV
0xAF	NPRACH Config #0 CP length	uint16_t	Cyclic prefix length for NPRACH transmission. See [11] section 10.1.6	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value: 0 for 66.7usec 1 for 266.7usec		
0xB0	NPRACH Config #0 Number of repetitions per attempt	uint16_t	Number of repetitions for NPRACH transmission. See [11] section 10.1.6 and [9] section 16.3 Value: 0 → 7 for 1, 2, 4, 8, 16, 32, 64, 128	TLV	TLV
0xB1	NPRACH Config #1 Enable	uint16_t	Enable \ Disable Config #1 0 = Disable. 1 = Enable.	TLV	TLV
0xB2	NPRACH Config #1 SF Periodicity	uint16_t	Periodicity of an NPRACH resource. See [11] section 10.1.6 Value: 0→7 for 40, 80, 160, 240, 320, 640, 1280, 2560	TLV	TLV
0xB3	NPRACH Config #1 StartTime	uint16_t	NPRACH resource starting time after period. See [11] section 10.1.6 Value: 0→7 for 8, 16, 32, 64, 128, 256, 512, 1024 FFFF to ignore	TLV	TLV
0xB4	NPRACH Config #1 SubCarrier offset	uint16_t	Frequency location of an NPRACH resource within a PRB. See [11] section 10.1.6 Value: 0→6 for 0, 12, 24, 36, 2, 18, 34	TLV	TLV
0xB5	NPRACH Config #1 number of subarriers	uint16_t	Number of subcarriers in an NPRACH resource. See [11] section 10.1.6 Value: 0→3 for 12, 24, 36, 48	TLV	TLV
0xB6	NPRACH Config #1 CP length	uint16_t	Cyclic prefix length for NPRACH transmission. See [11] section 10.1.6 Value: 0 for 66.7usec 1 for 266.7usec	TLV	TLV
0xB7	NPRACH Config #1 Number of repetitions per attempt	uint16_t	Number of repetitions for NPRACH transmission. See [11] section 10.1.6 and [9] section 16.3	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value: 0 → 7 for 1, 2, 4, 8, 16, 32, 64, 128		
0xB8	NPRACH Config #2 Enable	uint16_t	Enable \ Disable Config #0 0 = Disable. 1 = Enable.	TLV	TLV
0xB9	NPRACH Config #2 SF Periodicity	uint16_t	Periodicity of an NPRACH resource. See [11] section 10.1.6 Value: 0→7 for 40, 80, 160, 240, 320, 640, 1280, 2560	TLV	TLV
0xBA	NPRACH Config #2 StartTime	uint16_t	NPRACH resource starting time after period. See [11] section 10.1.6 Value: 0→7 for 8, 16, 32, 64, 128, 256, 512, 1024 FFFF to ignore	TLV	TLV
0xBB	NPRACH Config #2 SubCarrier offset	uint16_t	Frequency location of an NPRACH resource within a PRB. See [11] section 10.1.6 Value: 0→6 for 0, 12, 24, 36, 2, 18, 34	TLV	TLV
0xBC	NPRACH Config #2 number of subarriers	uint16_t	Number of subcarriers in an NPRACH resource. See [11] section 10.1.6 Value: 0→3 for 12, 24, 36, 48	TLV	TLV
0xBD	NPRACH Config #2 CP length	uint16_t	Cyclic prefix length for NPRACH transmission. See [11] section 10.1.6 Value: 0 for 66.7usec 1 for 266.7usec	TLV	TLV
0xBE	NPRACH Config #2 Number of repetitions per attempt	uint16_t	Number of repetitions for NPRACH transmission. See [11] section 10.1.6 and [9] section 16.3 Value: 0 → 7 for 1, 2, 4, 8, 16, 32, 64, 128	TLV	TLV
0xBF	threeTone-BaseSequence	uint16_t	DM-RS base sequence used for 3-tone NPUSCH transmission. See [11] section 10.1.4.1.2	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value: 4 bits		
0xC0	sixTone-BaseSequence	uint16_t	DM-RS base sequence used for 6-tone NPUSCH transmission. See [11] section 10.1.4.1.2 Value: 2 bits	TLV	TLV
0xC1	twelveTone-BaseSequence	uint16_t	DM-RS base sequence used for 12-tone NPUSCH transmission. See [11] section 10.1.4.1.2 Value: 5 bits	TLV	TLV
0xC2	threeTone-CyclicShift	uint16_t	Cyclic shift applied to the DM-RS base sequence used for 3-tone NPUSCH transmission. See [11] section 10.1.4.1.2	TLV	TLV
0xC3	sixTone-CyclicShift	uint16_t	Cyclic shift applied to the DM-RS base sequence used for 6-tone NPUSCH transmission. See [11] section 10.1.4.1.2	TLV	TLV
0xC4	DL Gap Config Enable	uint16_t	Enable \ Disable DL Gap 0 = Disable. 1 = Enable.	TLV	TLV
0xC5	DL Gap Threshold	uint16_t	Threshold on the maximum number of repetitions configured for NPDCCH before application of DL transmission gap configuration. See [11] Section 10.2.3.4. Value: 0 → 3 for 32, 64, 128, 256	TLV	TLV
0xC6	DL Gap Periodicity	uint16_t	Periodicity of a DL transmission gap in number of subframes. See [11] Section 10.2.3.4 Value: 0 → 3 for 64, 128, 256, 512	TLV	TLV
0xC7	DL Gap Duration Coefficient	uint16_t	Coefficient to calculate the gap duration of a DL transmission, in number of subframes. See [11] Section 10.2.3.4 Value: 0 → 3 for	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			oneEighth, oneFourth, threeEighth, oneHalf		
	These TLVs are used by L1 to report its physical capabilities to the L2/L3 software.				
0xC8	Downlink bandwidth support	uint16_t	The PHY downlink channel bandwidth capability (in resource blocks). See [10] section 5.6 Value: bitX :0 = no support, 1= support. Bit0: 6 Bit1: 15 Bit2: 25 Bit3: 50 Bit4: 75 Bit5: 100	TLV	TLV
0xC9	Uplink bandwidth support	uint16_t	The PHY uplink channel bandwidth capability (in resource blocks). See [10] section 5.6 Value: bitX :0 = no support, 1= support. Bit0: 6 Bit1: 15 Bit2: 25 Bit3: 50 Bit4: 75 Bit5: 100	TLV	TLV
0xCA	Downlink modulation support	uint16_t	The PHY downlink modulation capability. Value: bitX :0 = no support, 1= support. Bit0: QPSK Bit1: 16QAM Bit2: 64QAM Bit3: 256QAM	TLV	TLV
0xCB	Uplink modulation support	uint16_t	The PHY uplink modulation capability. Value: bitX :0 = no support, 1= support. Bit0: QPSK Bit1: 16QAM Bit2: 64QAM	TLV	TLV
0xCC	PHY antenna capability	uint16_t	Number of antennas supported.	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			Value: 1, 2, 4, 8, 16		
0xCD	Release capability	uint16_t	Indicates which release the PHY supports Value: bitX: 0 = no support, 1 = support. Bit0: Release 8 Bit1: Release 9 Bit2: Release 10 Bit3: Release 11 Bit4: Release 12 Bit5: Release 13	TLV	TLV
0xCE	MBSFN capability	uint16_t	Indicates support for MBSFN features Bit0: 0 = no support, 1= support	TLV	TLV
	LAA capability				
0xD1	LAA support	uint16_t	Indicates support for LAA features Bit0: 0 = no support, 1= support	TLV	TLV
0xD2	PD sensing LBT support	uint16_t	Indicates support for PD sensing in L1 0: PD sensing not supported 1: PD sensing supported	TLV	TLV
0xD3	Multi carrier LBT support	uint16_t	Bit: 0 = Multi carrier Mode A1 1 = Multi carrier Mode A2 2 = Multi carrier Mode B1 3 = Multi carrier Mode B2	TLV	TLV
0xD4	Partial SF support	uint16_t	Indicates support for Partial SF in L1 Bit 0 = start partial SF support Bit 1 = end partial SF support	TLV	TLV
	NB-IOT capability				
0xD5	NB-IOT support	uint16_t	Indicates support for NB-IOT Bit0: 0 = no support, 1= support	TLV	TLV

Tag	Description	Type	Value	FAPI	nFAPI
0xD6	NB-IOT Operating mode capability	uint16_t	Indicates operating mode support for NB-IOT, whether in-band NB-IOT (implying integrated NB-IOT and regular cells) or standalone & guard-band NB-IOT Value: 0 = standalone NB-IOT capability 1 = guard-band NB- IOT capability 2 = in-band NB-IOT Capability	TLV	TLV
	These TLVs are used by the L2/L3 software to configure the interaction between L2/L3 and L1.				
0xF0	Data report mode	uint16_t	The data report mode for the uplink data. 0: A CRC.indication message is sent in every subframe. If ULSCH data has been processed, the CRC.indication contains CRC results for the subframe. The CRC results are, also, given in the RX.indication message. 1: The CRC.indication message is not sent. The CRC results are given in the RX.indication message.	TLV	TLV
0xF1	SFN/SF	uint16_t	The future SFN/SF subframe where the TLVs included in the message should be applied. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	TLV	TLV
	These TLVs are used by L1 to report its current status.				
0xFA	PHY state	uint16_t	Indicates the current operational state of the PHY. 0 = IDLE 1 = CONFIGURED 2 = RUNNING	TLV	TLV
	These TLVs are used exclusively by nFAPI				
	These TLVs are used to setup the transport connection between VNF				

Tag	Description	Type	Value	FAPI	nFAPI
	and PNF				
0x0100	P7 VNF Address IPv4	Array of uint8_t	The IPv4 address of the VNF to be used by the PNF for this P7 PHY instance Note: address is network byte order. If both IPv4 and IPv6 addresses supplied, dual stack implementation is required.	NA	TLV
0x0101	P7 VNF Address IPv6	Array of uint8_t	The IPv6 address of the VNF to be used by the PNF for this P7 PHY instance Note: address is network byte order. If both IPv4 and IPv6 addresses supplied, dual stack implementation is required.	NA	TLV
0x0102	P7 VNF Port	uint16_t	The port of the VNF to be used by the PNF for this P7 PHY instance	NA	TLV
0x0103	P7 PNF Address IPv4	Array of uint8_t	The IPv4 address of the PNF PHY instance to be used by the VNF for this PNF PHY instance Note: address is network byte order. If both IPv4 and IPv6 addresses supplied, dual stack implementation is required.	NA	TLV
0x0104	P7 PNF Address IPv6	Array of uint8_t	The IPv6 address of the PNF PHY instance to be used by the VNF for this PNF PHY instance Note: address is network byte order. If both IPv4 and IPv6 addresses supplied, dual stack implementation is required.	NA	TLV
0x0105	P7 PNF Port	uint16_t	The port of the PNF PHY instance to be used by the VNF for this PNF PHY instance	NA	TLV
	These TLVs are used by the PNF to report its PHY capabilities to the VNF software				
0x010A	Downlink UEs per Subframe	uint8_t	The maximum number of	NA	TLV

Tag	Description	Type	Value	FAPI	nFAPI
			downlink UEs per subframe supported. This is the maximum number of downlink UEs that can be scheduled per subframe, non-inclusive of broadcast, paging and common channels.		
0x010B	Uplink UEs per Subframe	uint8_t	The maximum number of uplink UEs per subframe supported. This is the maximum number of uplink UEs that can be scheduled per subframe, non-inclusive of common channels.	NA	TLV
	These TLVs are used by PNF to report its RF capabilities to the VNF software				
0x0114	nFAPI RF Bands	struct	See Table 4-4	NA	TLV
	These TLVs are used by the VNF to configure the synchronization with the PNF.				
0x011E	Timing window	uint8_t	The window in milliseconds that the PHY must receive and queue the P7 messages. Value: 0 → 30 Other values are reserved	NA	TLV
0x011F	Timing info mode	uint8_t	The configured mode of operation for the timing info message to be sent to the VNF from the PHY Value: bitX :0 = disabled, 1= enabled. Bit0: Periodic Bit1: Aperiodic Bit2 – Bit7: Reserved	NA	TLV
0x0120	Timing info period	uint8_t	If Periodic timing mode is enabled, this defines the periodicity in subframes. This field is ignored if periodic timing mode is disabled. Value: 1 → 255 Other values are reserved	NA	TLV

Tag	Description	Type	Value	FAPI	nFAPI
	These TLVs are used by the VNF to configure the RF in the PNF				
0x0128	Maximum Transmit Power	uint16_t	The maximum transmit power for the PHY and RF operating at the configured bandwidth as defined in 3GPP TS 36.104. 0 to 700 Representing -10dBm to 60dBm in 0.1dB steps Other values are Reserved	NA	TLV
0x0129	EARFCN	uint16_t	The EARFCN to be used for the PHY and RF	NA	TLV
	These TLVs are used by PNF to report its NMM capabilities to the VNF software				
0x0130	NMM GSM Frequency Bands	struct	See Table 4-6	NA	TLV
0x0131	NMM UMTS Frequency Bands	struct	See Table 4-5	NA	TLV
0x0132	NMM LTE Frequency Bands	struct	See Table 4-4	NA	TLV
0x0133	NMM Uplink RSSI supported	uint8_t	Indicates if the uplink RSSI meausremnts are supported by NMM. 0 = Uplink RSSI not supported 1 = Uplink RSSI supported	NA	TLV

Table 4-3 Configuration TLVs

Field	Type	Description
Number of RF Bands	uint16_t	The number of RF Band instances
Number of RF Bands {		
Band	uint16_t	Band number as specified in 3GPP TS36.101
}		

Table 4-4 nFAPI LTE RF bands parameters

Field	Type	Description
Number of RF Bands	uint16_t	The number of RF Band instances
Number of RF Bands {		
Band	uint16_t	Band number as specified in 3GPP TS25.101
}		

Table 4-5 nFAPI UMTS RF bands parameters

Field	Type	Description
Number of RF Bands	uint16_t	The number of RF Band instances
Number of RF Bands {		
Band	uint16_t	Band number as specified in 3GPP TS45.005 –see Table 5-5.
}		

Table 4-6 nFAPI GSM RF bands parameters

4.2 P7 Subframe Messages

4.2.1 DL_CONFIG.request

The format of the `DL_CONFIG.request` message is shown in Table 4-7. A `DL_CONFIG.request` message indicates the SFN/SF subframe it contains information for. This control information is for a downlink subframe.

This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE or CONFIGURED state an `ERROR.indication` message will be sent by the PHY.

The following combinations of PDUs are required:

- A BCH PDU does not have an associated DCI PDU
- A PCH PDU requires an associated DCI PDU
- A DLSCH allocated with Semi-Persistent Scheduling may not have an associated DCI PDU
- A DLSCH for a unique RNTI requires an associated DCI PDU. Therefore, 2 DLSCH for the same RNTI only require 1 DCI PDU
- If present a MCH PDU indicates the subframe includes a MBSFN region, and there is no associated DCI PDU.
- A PRS PDU does not have an associated DCI PDU
- A CSI-RS PDU does not have an associated DCI PDU
- A EPDCCH PDU does not have an associated DCI PDU
- A MPDCCH PDU does not have an associated DCI PDU
- A N-BCH does not have an associated DCI PDU
- A NPDCCH DL PDU does not have an associated DCI PDU
- A NDLSCH PDU does not have an associated NPDCCH DL PDU (this would have been sent in a previous subframe)

The PDUs included in this structure have no ordering requirements.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
N/A	Length	uint16_t	The length of the downlink subframe configuration. Range 0 → 65535.	V	Not present

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2000	DL Config request body	struct	See Table 4-8.	V	TLV

Table 4-7 `DL_CONFIG.request` **message body**

Field		Type	Description
Number of PDCCH OFDM symbols		uint8_t	The number of OFDM symbols for the PDCCH. See [11] section 6.7. Value:0 → 4
Number of DCIs		uint8_t	The number of DCI PDUs included in this message. Range: 0 → 255
Number of PDUs		uint16_t	Number of PDUs that are included in this message. Range 0 → 514
Number of PDSCH RNTIs		uint8_t	Number of unique RNTIs sent on the PDSCH. - a PCH PDU will have an unique RNTI and should be included in this value - a DLSCH PDU can be one transport block sent to a UE with an unique RNTI. This RNTI should be included in this value - a DLSCH PDU can be one of two transport blocks sent to a UE. In this case the two DLSCH PDUs will share the same RNTI. Only one RNTI should be included in this value. - EPDCCH PDUs can be associated to the same RNTI. Only one RNTI should be included in this value (In case DLSCH already include the RNTI value it shouldn't be added) - MPDCCH PDUs can be associated to the same RNTI. Only one RNTI should be included in this value (In case DLSCH already include the RNTI value it shouldn't be added) - if a N-BCH PDU is included it is the only PDU in the message - a NPDCCH PDU will have an unique RNTI. If included only NPDCCH PDUs will be in the message - a NDLSCH PDU will have an unique RNTI. If included the NDLSCH PDU will be the only PDU in the message
Transmission power for PCFICH		uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
For number of PDUs			
	PDU Type	uint8_t	0: DCI DL PDU, see Section 4.2.1.1 1: BCH PDU, see Section 4.2.1.2 2: MCH PDU, see Section 4.2.1.3 3: DLSCH PDU, see Section 4.2.1.4 4: PCH PDU, see Section 4.2.1.5

Field	Type	Description
		5: PRS PDU, see Section 4.2.1.6. Added in Release 9 6: CSI-RS PDU, see Section 4.2.1.7. Added in Release 10 7: EPDCCH DL DCI, see section 4.2.1.8. Added in Release 11 8: MPDCCH, see section 4.2.1.9. Added in Release 13 9: N-BCH, see section 4.2.1.10. Added in Release 13 10: NPDCCH DL PDU, see section 4.2.1.11. Added in Release 13 11: NDLSCH PDU, see section 4.2.1.12
PDU Size	uint8_t	Size of the PDU control information (in bytes). This length value includes the 2 bytes required for the PDU type and PDU size parameters.
DL PDU Configuration	Struct	See sections 4.2.1.1 to 4.2.1.12.

Table 4-8 DL config request body

4.2.1.1 DCI DL PDU

The format of a DCI DL PDU is shown in Table 4-9. The DCI DL PDU contains the information that the L2/L3 software must provide the PHY so it can create the DCI formats, related to the downlink, described in [12] section 5.3.3.1.

DCI format 2B was introduced in Release 9.
DCI format 2C was introduced in Release 10.
DCI format 2D was introduced in Release 11.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2001	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-10 should be included.	V	TLV
0x2002	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-11 should be included.	V	TLV
0x2003	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-12 should be included.	V	TLV
0x2039	Release 11 parameters	struct	If the PHY supports Release 11 the parameters described in Table 4-13 should be included.	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
0x203a	Release 12 parameters	struct	If the PHY supports Release 12 the parameters described in Table 4-14 should be included.	V	TLV
0x203b	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-15 should be included.	V	TLV

Table 4-9 DCI DL PDU

Field	Type	Description
DCI format	uint8_t	Format of the DCI 0 = 1 1 = 1A 2 = 1B 3 = 1C 4 = 1D 5 = 2 6 = 2A 7 = 2B 8 = 2C (release 9) 9 = 2D (release 11)
CCE index	uint8_t	CCE index used to send the DCI. In case transmitted over EPDCCH/ MPDCCH this parameter will be the ECCE index Value: 0 → 88 for PDCCH Value: 0 → 31 for EPDCCH
Aggregation level	uint8_t	The aggregation level used Value: 1,2,4,8,16*,32* * possible only in case transmitted over EPDCCH
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Valid for all DCI formats Value: 1 → 65535.
Resource allocation type	uint8_t	Resource allocation type/header Valid for DCI formats: 1,2,2A,2B,2C,2D 0=type 0 1=type 1
Virtual resource block assignment flag	uint8_t	Type of virtual resource block used Valid for DCI formats: 1A,1B,1D 0 = localized

Field	Type	Description
		1 = distributed
Resource block coding	uint32_t	The encoding for the resource blocks. The coding is dependent on whether resource allocation type 0, 1, 2 is in use. Resource allocation type 0 is explicitly signalled for DCI formats 1, 2, 2A,2B,2C,2D Resource allocation type 1 is explicitly signalled for DCI formats 1, 2, 2A,2B,2C,2D Resource allocation type 2 is implicit for DCI formats 1A, 1B, 1C, 1D See [9] section 7.1.6 for the encoding used for each format. Valid for DCI formats: 1,1A,1B,1C,1D,2,2A,2B,2C,2D See [9] section 7.1.6 for the encoding used for each format and a variable-length bitstring generated. Further information on the location of this bitstring within the resource block coding 32-bit parameter is implementation-specific
MCS_1	uint8_t	The modulation and coding scheme for 1st transport block Valid for DCI formats: 1,1A,1B,1C,1D ,2,2A,2B,2C,2D Value: 0 → 31
Redundancy version_1	uint8_t	The redundancy version for 1st transport block. Valid for DCI formats: 1,1A,1B,1C,1D ,2,2A,2B,2C,2D Value: 0 → 3
New data indicator_1	uint8_t	The new data indicator for 1st transport block. Valid for DCI formats: 1,1A,1B,1C,1D ,2,2A,2B,2C,2D
Transport block to codeword swap flag	uint8_t	Indicates the mapping of transport block to codewords Valid for DCI formats: 2,2A 0 = no swapping 1 = swapped
MCS_2	uint8_t	The modulation and coding scheme for 2nd transport block. Valid for DCI formats: 2,2A,2B,2C,2D Value: 0 → 31
Redundancy version_2	uint8_t	The redundancy version for 2nd transport block. Valid for DCI formats: 2,2A,2B,2C,2D Value: 0 → 3

Field	Type	Description
New Data indicator_2	uint8_t	The new data indicator for 2nd transport block. Valid for DCI formats: 2,2A,2B,2C,2D
HARQ process	uint8_t	HARQ process number Valid for DCI formats: 1,1A,1B,1D,2,2A,2B,2C,2D Value: 0 →15
TPMI	uint8_t	The codebook index to be used for precoding Valid for DCI formats: 1B,1D 2 antenna_ports: 0 → 3 4 antenna_ports: 0 → 15
PMI	uint8_t	Confirmation for precoding Valid for DCI formats: 1B 0 = use precoding indicated in TPMI field 1 = use precoding indicated in last PMI report on PUSCH 2= use precoding indicated in TPM field (release 13 addition)
Precoding information	uint8_t	Precoding information Valid for DCI formats: 2,2A 2 antenna_ports: 0 → 7 4 antenna_ports: 0 → 63
TPC	uint8_t	Tx power control command for PUCCH. Valid for DCI formats: 1,1A,1B,1D,2,2A,2B,2C,2D Value: 0,1,2,3 In case of DCI format 1A and RNTI=SI-RNTI,RA-RNTI or P-RNTI the TPC values are either 0,1 representing N_PRB=2 and N_PRB =3 respectively. In case of SPS-C-RNTI it represents the PUCCH resource index.
Downlink assignment index	uint8_t	The downlink assignment index. In release 8-11 this is only used in TDD mode, value ignored for FDD. In release 12 or later a field indicating the structure type of the primary cell is used to determine if this is valid with size 2 bits. In release 13 or later a field indicating codebooksizeDetermination-r13=0 is used to determine is this field is valid with size 4 bits. Valid for DCI formats: 1,1A,1B,1D,2,2A,2B,2C,2D Value: 1,2,3,4 if 2 bits Value: 0-15 if 4 bits
N_{GAP}	uint8_t	Used in virtual resource block distribution Valid for DCI formats: 1A,1B,1C,1D

Field	Type	Description
		0= N_{GAP1} 1= N_{GAP2}
Transport block size index	uint8_t	The transport block size Valid for DCI formats: 1C in release 8 only (In release 9 this value was deprecated and MCS_1 should be used instead. Using MCS_1 will not affect release 8 UE) Value: 0 → 31
Downlink power offset	uint8_t	Indicates the DL power offset type for multi-user MIMO transmission Valid for DCI formats: 1D Value: 0 → 1
Allocate PRACH flag	uint8_t	Indicates that PRACH procedure is initiated Valid for DCI formats: 1A 0 = false 1=true
Preamble index	uint8_t	The preamble index to be used on the PRACH Valid for DCI formats: 1A Value: 0 → 63
PRACH mask index	uint8_t	The mask index to be used on the PRACH Valid for DCI formats: 1A Value: 0 → 15
RNTI type	uint8_t	RNTI type Valid for DCI format 1,1A,2,2A,2B,2C,2D 1 = C-RNTI 2 = RA-RNTI, P-RNTI, SI-RNTI , SC-RNTI or G-RNTI. 3 = SPS-CRNTI
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.

Table 4-10 DCI DL PDU Release 8 parameters

Field	Type	Description
MCCH flag	uint8_t	Indicates if format 1C is being used to signal a MCCH or SC-MCCH change notification Valid for DCI formats 1C 0 = MCCH or SC-MCCH change notification field is not valid

Field	Type	Description
		1 = MCCH or SC-MCCH change notification field is valid
MCCH change notification	uint8_t	MCCH or SC-MCCH Change Notification Valid for DCI format 1C Value: 0->255
Scrambling identity	uint8_t	Indicates the scrambling identity value N_{SCID} Valid for DCI format 2B Value: 0,1

Table 4-11 DCI DL PDU Release 9 parameters

Field	Type	Description
Cross carrier scheduling flag	uint8_t	Indicates if cross carrier scheduling has been enabled for the UE receiving this DCI Valid for DCI formats 1,1A,1B, 1D,2,2A,2B,2C,2D 0 = Carrier indicator field is not valid 1 = Carrier indicator field is valid
Carrier indicator	uint8_t	Serving Cell Index Valid for DCI formats 1,1A,1B,1D, 2,2A,2B,2C,2D if the Cross-Carrier Scheduling flag is enabled Value: 0->7
SRS flag	uint8_t	Indicates if the SRS request parameter is valid Valid for DCI formats 1A ,2B,2C,2D 0 = SRS request field is not valid 1 = SRS request field is valid
SRS request	uint8_t	SRS request flag Valid for DCI formats 1A,2B,2C,2D if the SRS flag is enabled. 0 = SRS not requested 1= SRS requested
Antenna ports, scrambling and layers	uint8_t	Indicates the Antenna port, scrambling identity value N_{SCID} and number of layers Valid for DCI format 2C,2D Value: 0->7 for Rel 10, 11, 12 or if the field *Rel-13-DMRS-table* is 0 Value: 0->15 for Rel 13 if the field *Rel-13-DMRS-table* is 1
Total DCI length including padding	uint8_t	The total DCI length including padding bits
N_DL_RB	uint8_t	BW of serving cell for which the DCI was

Field	Type	Description
		scheduled for. This is valid for the case of cross carrier scheduling, for the case of a self-scheduling (cross carrier scheduling is not valid or Carrier indicator has value '0', the BW is the "DL BW support" as configured in configuration phase (params) Downlink channel bandwidth in resource blocks. See [10] section 5.6. Value: 6,15, 25, 50, 75, 100

Table 4-12 DCI DL PDU Release 10 parameters

Field	Type	Description
HARQ-ACK resource offset	uint8_t	HARQ-ACK resource offset field is present only when this format is carried by EPDCCH. Valid for DCI formats 1, 1A, 1B, 1D, 1, 2A, 2, 2B, 2C, 2D Value: 0->3
PDSCH RE Mapping and Quasi-Co-Location Indicator	uint8_t	Indicates the parameter set configured by the higher layers which the UE should use. Valid for DCI formats 2D Value: 0->3

Table 4-13 DCI DL PDU Release 11 parameters

Field	Type	Description
Primary cell type	uint8_t	Indicates the type of the primary cell. Valid for DCI formats: 1,1A,1B,1D,2,2A,2B,2C 0 = TDD 1 = FDD 2 = HD_FDD
UL/DL configuration flag	uint8_t	Indicates if format 1C is being used to signal UL/DL configuration Valid for DCI formats 1C 0 = UL/DL configuration field is not valid 1 = UL/DL configuration field is valid
Number of UL/DL configurations	uint8_t	Number of UL/DL configurations, I, as defined by [12] section 5.3.3.1.4
For number of UL/DL configurations		
UL/DL configuration indication	uint8_t	UL/DL configuration index Value: 1->5

Table 4-14 DCI DL PDU Release 12 parameters

Field	Type	Description
LAA end partial SF flag	uint8_t	Indicates if DCI format 1C is being used to signal LAA end partial SF (valid if end partial SF support configuraton is set) 0 = LAA end partial SF configuration field is not valid 1 = LAA end partial SF configuration field is valid
LAA end partial SF configuration	uint8_t	If DCI format 1C scrambled by CC-RNTI is used to signal end partial SF, this field contains LAA common information (4 bits used in [9] Table 13A-1 for configuration of occupied OFDM symbols for current and next SF)
Initial LBT SF	uint8_t	Indicates if the DCI PDU is prepared for full SF (regular) or for initial partial SF (2^{nd} slot) according to [11] section 6.2.4 (if PDCCH) or 6.2.4A (if ePDCCH) 0 = prepared for the case of a regular SF (higher layer parameter " *subframeStartPosition*"='s0') 1 = prepared for the case of a partial SF (higher layer parameter " *subframeStartPosition*"='s07') Valid if initial partial SF in LAA
codebooksizeDetermination-r13	uint8_t	Indicates if the downlink assignment index parameter (DAI) is 4 bits 0 = DAI is 4 bits 1= DAI maybe 0 or 2 bits depending on other type of primary cell. Valid for DCI formats: 1,1A,1B,1D,2,2A,2B,2C
Rel-13-DMRS-tabe flag	uint8_t	Indicates if Release 13 DMRS table for be used. 0 = not used 1 = used
TPM struct flag	uint8_t	Indicates if the TPM structure is present. If this flag is set then also the PMI field (release 8) =2. 0 = not present 1 = present
TPM	struct	If PMI field (release 8) =2 and the PHY is TDD the parameters described in Table 4-16 will be included. Not valid for FDD

Table 4-15 DCI DL PDU Release 13 parameters

Field	Type	Description
numPRBperSubband	uint8_t	Number of PRBs that are treated as one

Field	Type	Description
		subband
number of subbands	uint8_t	Defines the number of subbands used for channel feedback. Value 0 -> 13
numAntennas	uint8_t	Number of physical antennas
For (number of subbands)		
subbandIndex	uint8_t	Index of subband for which the following precoding matrix is applied
scheduledUEs	unit8_t	The number of scheduled UE's per-subband of UE's Index for which the following precoding coefficient is applied
For (number of physical antennas)		
For (number of scheduledUEs)		
precodingValue	unit16_t	Precoding coefficient in a subband for physical antenna #i, real 8 bits followed by imaginary 8 bits

Table 4-16 TPM Structure

4.2.1.2 BCH PDU

The format of the BCH PDU is shown in Table 4-17.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2004	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-18 should be included.	V	TLV

Table 4-17 BCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in TX.request. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH or DLSCH PDU is included in the DL_CONFIG.request message. This value is repeated in `TX.request` and associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
Transmission power	uint16_t	Offset to the reference signal power.

Field	Type	Description
		Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.

Table 4-18 BCH PDU Release 8 parameters

4.2.1.3 MCH PDU

The format of the MCH PDU is shown in Table 4-19. The contents of the MCH PDU are preliminary.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2005	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-20 should be included.	V	TLV

Table 4-19 MCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in `TX.request`. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits. A length of zero indicates that no MCH PDU is present and there is an empty MBSFN region in this subframe. In this case the remaining parameters for this PDU can be ignored.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH or DLSCH PDU is included in the `DL_CONFIG.request` message. This value is repeated in `TX.request` and associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
RNTI	uint16_t	The RNTI associated with the MCH See [5] section 7.1 Value: FFFD
Resource allocation type	uint8_t	This field is not used.
Resource block coding	uint32_t	This field is not used.
Modulation	uint8_t	2: QPSK 4: 16QAM 6: 64QAM 8: 256QAM
Transmission power	uint16_t	Offset of MCH data and MBSFN-RS to the CRS reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.

Field	Type	Description
MBSFN area ID	uint16_t	Indicates MBSFN area ID, see [11] section 6.10.2 Value: 0 → 255

Table 4-20 MCH PDU Release 8 parameters

4.2.1.4 DLSCH PDU

The format of the DLSCH PDU is shown in Table 4-21.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2006	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-22 should be included.	V	TLV
0x2007	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-24 should be included.	V	TLV
0x2008	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-25 should be included.	V	TLV
0x203c	Release 11 parameters	struct	If the PHY supports Release 11 the parameters described in Table 4-26 should be included.	V	TLV
0x203d	Release 12 parameters	struct	If the PHY supports Release 12 the parameters described in Table 4-27 should be included.	V	TLV
0x203e	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-28 should be included.	V	TLV

Table 4-21 DLSCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in `TX.request`. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH or DLSCH PDU is included in the DL_CONFIG.request message. This value is repeated in `TX.request` and

Field	Type	Description
		associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
RNTI	uint16_t	The RNTI associated with the UE See [5] section 7.1 Value: 1 → 65535.
Resource allocation type	uint8_t	Resource allocation type See [9] section 7.1.6 0 = type 0 1 = type 1 2 = type 2 (allocated by DCI format 1A, 1B or 1D) 3 = type 2 (allocated by DCI format 1C) 4 = type 2 (allocated by DCI format 6-1A) 5 = type UEModeB (allocated by DCI format 6-1B)
Virtual resource block assignment flag	uint8_t	Type of virtual resource block used. This should match the value sent in the DCI Format 1A, 1B, 1D PDU which allocated this grant. See [9] section 7.1.6.3 0 = localized 1 = distributed
Resource block coding	uint32_t	The encoding for the resource blocks. It's coding is dependent on whether resource allocation type 0, 1, 2 is in use. This should match the value sent in the DCI Format PDU which allocated this grant. See [9] section 7.1.6 for the encoding used for each format.
Modulation	uint8_t	2: QPSK 4: 16QAM 6: 64QAM 8: 256QAM
Redundancy version	uint8_t	HARQ redundancy version. This should match the value sent in the DCI Format PDU which allocated this grant. Value: 0 → 3.
Transport blocks	uint8_t	The transport block transmitted to this RNTI. A value of 2 indicates this is the second transport block for DCI format 2, 2A, 2B or 2C. For other DCI values this field will always be 1. Value: 1 → 2
Transport block to codeword swap	uint8_t	Indicates the mapping of transport block to codewords. This should match the value sent in

Field	Type	Description
flag		the DCI Format 2, 2A PDU which allocated this grant. 0 = no swapping 1 = swapped
Transmission scheme	uint8_t	The MIMO mode used in the PDU See [9] section 7.1. 0: SINGLE_ANTENNA_PORT_0, 1: TX_DIVERSITY, 2: LARGE_DELAY_CDD, 3: CLOSED_LOOP_SPATIAL_MULTIPLEXING, 4: MULTI_USER_MIMO, 5: CLOSED_LOOP_RANK_1_PRECODING, 6: SINGLE_ANTENNA_PORT_5. 7: SINGLE_ANTENNA_PORT_7, added in Release 9 8: SINGLE_ANTENNA_PORT_8, added in Release 9 9: DUAL_LAYER_TX_PORT_7_AND_8, added in Release 9 10: UP_TO_8_LAYER_TX, added in Release 10 11: SINGLE_ANTENNA_PORT_11, added in Release 13 12: SINGLE_ANTENNA_PORT_13, added in Release 13 13: DUAL_LAYER_TX_PORT_11_AND_13, added in Release 13
Number of layers	uint8_t	The number of layers used in transmission See [11] section 6.3.3 Value: 1 → 8
Number of subbands	uint8_t	Only valid when transmission scheme = 3, 4, 5. Defines the number of subbands and codebooks used for PMI. If value=1 then a single PMI value is supplied which should be used over all RB Value 0 -> 13
Number of subband entries {		
Codebook index	uint8_t	Only valid when transmission scheme = 3, 4, 5. Defines the codebook used. When antenna port = 1: NA When antenna port = 2: 0..3 When antenna port = 4: 0..15
}		
UE category capacity	uint8_t	The UE capabilities category See [13] section 4.1. Value:0 → 14

Field	Type	Description
P-A	uint8_t	The ratio of PDSCH EPRE to cell-specific RS EPRE among PDSCH REs in all the OFDM symbols not containing cell-specific RS in dB. See [9], section 5.2. 0: -6dB 1: -4.77dB 2: -3dB 3: -1.77dB 4: 0dB 5: 1dB 6: 2dB 7: 3dB
Delta power offset index	uint8_t	Delta power offset, value: 0..1, Refer to: Table 7.1.5-1 in [9] for Multi-user MIMO mode. Takes value zero for all other modes.
N_{GAP}	uint8_t	Used in virtual resource block distribution 0= N_{GAP1} 1= N_{GAP2}
N_{PRB}	uint8_t	Used with DCI format 1A and RNTI=SI-RNTI or RA-RNTI. This should match the value sent in the TPC field of the DCI 1A PDU which allocated this grant. 0: N_{PRB} = 2 1: N_{PRB} = 3
Transmission mode	uint8_t	The transmission mode associated with the UE See [9] section 7.1. 1: Mode 1 2: Mode 2 3: Mode 3 4: Mode 4 5: Mode 5 6: Mode 6 7: Mode 7 8: Mode 8 9: Mode 9 10: Mode 10
numBfPRBperSubband	uint8_t	Number of PRBs that are treated as one subband
numBfVector	uint8_t	Number of beam forming vectors One beam forming vector is specified for each subband
bfVector	BfVectorType[numBfVector]	Beam forming vectors, see Table 4-23.

Table 4-22 DLSCH PDU Release 8 parameters

Field		Type	Description
subbandIndex		uint8_t	Index of subband for which the following beam forming vector is applied
numAntennas		uint8_t	Number of physical antennas
For each physical antenna			
	bfValue	uint16_t	Beam forming vector element for physical antenna #i real 8 bits followed by imaginary 8 bits

Table 4-23 BfVectorType Structure

Field	Type	Description
N_{SCID}	uint8_t	Used with DCI format 2B and 2C. 0: N_{SCID}=0 1: N_{SCID}=1

Table 4-24 DLSCH PDU Release 9 parameters

Field		Type	Description
CSI-RS flag		uint8_t	Indicates if parameters related to CSI-RS are valid or not. Value: 0: CSI-RS parameters are not valid 1: CSI-RS parameters are valid
CSI-RS resource config R10		uint8_t	This value is deprecated
CSI-RS zero Tx power resource config bitmap R10		uint16_t	Bitmap of 16 bits. Encoding format of bitmap follows section 6.10.5.2 of [11].
CSI-RS Number of NZP configuration		uint8_t	Indicates the number of Non-Zero power CSI-RS configurations. Value: 0 → 1 for release 10 0 → 3 for release 11
For each CSI-RS configuration			
	CSI-RS resource config	uint8_t	Indicates reference signal configuration for CSI-RS. See [11] table 6.10.5.2-1 and 6.10.5.2-2. Value : 0 → 31
PDSCH_start		uint8_t	Per UE starting OFDM symbol for the PDSCH, impacts the mapping of PDSCH to REs See [9] section 7.1.6.4 Value: 0: field is ignored and PDSCH starts according

Field	Type	Description
		to the CFI 1-4 (4 only in 1.4 MHz BW): value overriding the CFI

Table 4-25 DLSCH PDU Release 10 parameters

Field	Type	Description
DMRS Config flag	uint8_t	Indicates if the DMRS Config parameter is valid. 0 = not used, 1= enabled
DMRS-Scrambling	uint16_t	The scrambling identity for UE specific reference signals. See [11] section 6.10.3 Value: 0 → 503
CSI Config flag	uint8_t	Indicates if the CSI Config parameter is valid. 0 = not used, 1= enabled
CSI- Scrambling	uint16_t	The scrambling identity for CSI. See [11] section 6.10.5 Value: 0 → 503
PDSCH RE mapping flag	uint8_t	Indicates if the PDSCH RE parameters are valid. 0 = not used, 1= enabled
PDSCH RE mapping antenna ports	Uint8_t	Indicates number of antennas used for PDSCH RE mapping. Value: 1, 2, 4
PDSCH RE mapping freq shift	uint8_t	Indicates the frequency shift used for PDSCH RE mapping. Value: 0 → 5

Table 4-26 DLSCH PDU Release 11 parameters

Field	Type	Description
altCQI-Table-r12	uint8_t	altCQI-Table-r12 is indicative of using an alternative MCS table for UEs supporting 256QAM. This is taken into account for calculation of soft buffer size for the transport block See [12] section 5.1.4.1.2 Value: 0-1
MaxLayers	uint8_t	Maximal number of negotiated / configured layers for a UE, used for the calculation of soft buffer size for the transport block See [12] section 5.1.4.1.2

Field	Type	Description
		Value:1-8
N_DL_HARQ	uint8_t	M_{DL_HARQ} in section 5.1.4.1.1 in [12]

Table 4-27 DLSCH PDU Release 12 parameters

Field	Type	Description
DwPTS Symbols	uint8_t	Valid if DCI format 1C is being used to signal LAA end partial SF. Indicates the number of starting symbols according to [9] Table 13-A-1 Values: 3, 6, 9, 10, 11, 12, 14
Initial LBT SF	uint8_t	Indicates if DLSCH is prepared for full SF (regular) or for initial partial SF (2nd slot) according to [11] Section 6.4 0 = prepared for the case of a regular SF (higher layer parameter " *subframeStartPosition*"='s0') 1 = prepared for the case of a partial SF (higher layer parameter " *subframeStartPosition*"='s07') Valid if initial partial SF support configuration is set
UE Type	uint8_t	Value: 0: non LC/CE UE 1: LC/CE CEModeA UE 2: LC/CE CEModeB UE
PDSCH Payload Type	uint8_t	Value: 0: PDSCH carrying SIB1-BR 1: PDSCH carrying SI message (except for SIB1-BR or PCH) 2: PDSCH carrying other Valid if UE Type is 1 or 2
Initial transmission SF (io)	Uint16_t	Absolute Sub-Frame of the initial transmission Value: 0 → 10239 Value: 0xFFFF current absolute SF Valid if UE Type is 1 or 2
Rel-13-DMRS-tabe flag	uint8_t	Indicates if Release 13 DMRS table is used. 0 = not used 1 = used

Table 4-28 DLSCH PDU Release 13 parameters

4.2.1.5 PCH PDU

The format of the PCH PDU is shown in Table 4-29.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2009	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-30 should be included.	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
0x203f	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-31 should be included.	V	TLV

Table 4-29 PCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in `TX.request`. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH or DLSCH PDU is included in the DL_CONFIG.request message. This value is repeated in `TX.request` and associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
P-RNTI	uint16_t	The P-RNTI associated with the paging See [5] section 7.1 Value: 0xFFFE
Resource allocation type	uint8_t	Resource allocation type See [9] section 7.1.6 2 = type 2 (allocated by DCI format 1A, 1B or 1D) 3 = type 2 (allocated by DCI format 1C) 6 = NB index (allocated by DCI 6-2)
Virtual resource block assignment flag	uint8_t	Type of virtual resource block used. This should match the value sent in the DCI Format 1A, 1B, 1D PDU which allocated this grant. See [9] section 7.1.6.3 0 = localized 1 = distributed
Resource block coding	uint32_t	The encoding for the resource blocks. It's coding is dependent on whether resource allocation type 0, 1, 2 is in use. This should match the value sent in the DCI Format PDU which allocated this grant. See [9] section 7.1.6 for the encoding used for each format.
MCS	uint8_t	For PCH PDU only QPSK modulation is allowed. 0: QPSK
Redundancy version	uint8_t	For PCH PDU only redundancy version 0 is allowed

Field	Type	Description
		Value: 0
Number of transport blocks	uint8_t	The number of transport blocks transmitted to this RNTI. Only 1 transport block is sent on the PCH per subframe. Value: 1
Transport block to codeword swap flag	uint8_t	Reserved. This parameter is not used on the PCH transport channel.
Transmission scheme	uint8_t	The MIMO mode used in the PDU See [9] section 7.1. 0: SINGLE_ANTENNA_PORT_0, 1: TX_DIVERSITY, 6: SINGLE_ANTENNA_PORT_5.
Number of layers	uint8_t	The number of layers used in transmission See [11] section 6.3.3 Value: 1 → 4
Codebook index	uint8_t	Reserved. This parameter is not used on the PCH transport channel.
UE category capacity	uint8_t	Reserved. This parameter is not used on the PCH transport channel. Starting Release 12, the field is introduced for Category 0 The UE capabilities category See [13] section 4.1. Value:0 → 14
P-A	uint8_t	The ratio of PDSCH EPRE to cell-specific RS EPRE among PDSCH REs in all the OFDM symbols not containing cell-specific RS in dB. See [9], section 5.2. 0: -6dB 1: -4.77dB 2: -3dB 3: -1.77dB 4: 0dB 5: 1dB 6: 2dB 7: 3dB
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
N_{PRB}	uint8_t	Used with DCI 1A format. This should match the

Field	Type	Description
		value sent in the TPC field of the DCI 1A PDU which allocated this grant. 0: N_{PRB} = 2 1: N_{PRB} = 3
N_{GAP}	uint8_t	Used in virtual resource block distribution 0= N_{GAP1} 1= N_{GAP2}

Table 4-30 PCH PDU Release 8 parameters

Field	Type	Description
UE mode	uint8_t	Value: 0: non LC/CE UE 1: LC/CE UE
Initial transmission SF (io)	uint16_t	Absolute Sub-Frame of the initial transmission Value: 0 → 10239 Value: 0xFFFF current absolute SF

Table 4-31 PCH PDU Release 13 parameters

4.2.1.6 PRS PDU

The format of the PRS PDU is shown in Table 4-32. Transmission of PRS in a subframe is indicated to L1 by the existence of the PRS PDU in DL_CONFIG.request. Otherwise, PRS is not transmitted. PRS PDU is only included if the PHY supports Release 9.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x200a	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-33 should be included.	V	TLV

Table 4-32 PRS PDU

Field	Type	Description
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
PRS bandwidth	uint8_t	PRS bandwidth in resource blocks. Value: 6, 15, 25, 50, 75, 100
PRS cyclic prefix type	uint8_t	The cyclic prefix used for PRS transmission. Value: 0: normal cyclic prefix 1: extended cyclic prefix
PRS muting	uint8_t	PRS muting dictates if PRS REs are vacant (prsMutingInfo-r9 indicates the SF occasions) Value – 0: no muting (PRS REs contain PRS with Transmission power)

Field	Type	Description
		Value – 1: muting (PRS REs with no signal)

Table 4-33 PRS PDU Release 9 parameters

4.2.1.7 CSI-RS PDU

The format of the CSI-RS PDU is shown in Table 4-34. CSI-RS PDU is only included if the PHY supports Release 10.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x200b	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-35 should be included.	V	TLV
0x2040	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-36 should be included.	V	TLV

Table 4-34 CSI-RS PDU

Field	Type	Description
CSI-RS antenna port count R10	uint8_t	Indicates number of antennas used for transmission of CSI reference signal. Value: 1, 2, 4, 8, 12, 16
CSI-RS resource config R10	uint8_t	~~Indicates reference signal configuration for CSI-RS. See [11] table 6.10.5.2-1 and 6.10.5.2-2.~~ ~~Value : 0 → 31~~ This value is deprecated
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
CSI-RS zero Tx power resource config bitmap R10	uint16_t	Bitmap of 16 bits. Encoding format of bitmap follows section 6.10.5.2 of [11].
CSI-RS Number of NZP configuration	uint8_t	Indicates the number of Non-Zero power CSI-RS configurations. Value: 0 → 1 for release 10 0 → 3 for release 11 0 → 8 for release 13
For each CSI-RS configuration		
CSI-RS resource	uint8_t	Indicates reference signal configuration for CSI-

Field	Type	Description
config		RS. See [11] table 6.10.5.2-1 and 6.10.5.2-2. Value : 0 → 31

Table 4-35 CSI-RS PDU Release 10 parameters

Field	Type	Description
Class	uint8_t	Indicates CSI-RS class 0 = not used 1 = Class A (non-precoded) 2 = Class B (beamformed)
cdmType	uint8_t	Indicates CDM type for CSI-RS. See [36.211] section 6.10.5.2. Valid for Class A 0 = cdm2 1 = cdm4
numBfVector	uint8_t	Number of beam forming vectors With Class B one beam forming vector is specified for each CSI-RS. Value : 0 for Class A Value : 0 → 8 for Class B
For each BF vector		
CSI-RS resource index	uint8_t	Index of the CSI-RS resource. This is included to link bfValues to CSI-RS resources included in Release 10 parameters. Value : 0 → 7
For each antenna port count		
bfValue	uint16_t	Beam forming vector element for physical antenna #i real 8 bits followed by imaginary 8 bits

Table 4-36 CSI-RS PDU Release 13 parameters

4.2.1.8 EPDCCH PDU

The format of the EPDCCH DL PDU is shown in Table 4-37. Transmission of EPDCCH in a subframe is indicated to L1 by the existence of the EPDCCH PDU in DL_CONFIG.request. Otherwise, EPDCCH is not transmitted. EPDCCH DL PDU is only included if the PHY supports Release 11.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2001	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-10 should be included.	V	TLV
0x2002	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-11	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
			should be included.		
0x2003	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-12 should be included.	V	TLV
0x2039	Release 11 parameters	struct	If the PHY supports Release 11 the parameters described in Table 4-13 should be included.	V	TLV
0x203a	Release 12 parameters	struct	If the PHY supports Release 12 the parameters described in Table 4-14 should be included.	V	TLV
0x203b	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-15 should be included.	V	TLV
0x2041	Release 11 EPDCCH parameters	struct	If the PHY support EPDCCH- Release 11, the parameters described in Table 4-38 should be included.	V	TLV
0x2042	Release 13 EPDCCH parameters	struct	If the PHY support EPDCCH- Release 13, the parameters described in Table 4-39 should be included.	V	TLV

Table 4-37 EPDCCH DL PDU

As indicated in Table 4-37 the EPDCCH PDU actually carries DCI structure of Release 8-13 with the addition of the EPDCCH parameters from Table 4-38.

Field	Type	Description
EPDCCH Resource assignment flag	uint8_t	Type of virtual resource block used 0 = localized 1 = distributed
EPDCCH ID	uint16_t	EPDCCH index- used for the scrambler initiation The DMRS scrambling sequence initialization parameter $n_{\mathrm{ID},i}^{\mathrm{EPDCCH}}$ defined in [11] section 6.10.3A.1 Value: 0 → 503
EPDCCH Start Symbol	uint8_t	Indicates the OFDM starting symbol for any EPDCCH and PDSCH

Field		Type	Description
			Values: 1 → 4.
EPDCCH NumPRB		uint8_t	Number of PRBs allocated for EPDCCH Value: 2, 4, 8.
For each PRB (Valid for EPDCCH PDU type)			
	EPDCCH PRB index	Uint8_t	PRB index Value: 0 → 99.
bfVector		BfVectorType	Beam forming vectors, see Table 4-23.

Table 4-38 EPDCCH DL PDU Release 11 parameters

Field	Type	Description
DwPTS Symbols	uint8_t	Valid if DCI format 1C is being used to signal LAA end partial SF. Indicates the number of starting symbols according to [9] table 13-A-1 Values: 3, 6, 9, 10, 11, 12, 14
Initial LBT SF	uint8_t	Indicates if DLSCH is prepared for full SF (regular) or for initial partial SF (2nd slot) according to [11] section 6.8A 0 = prepared for the case of a regular SF (higher layer parameter " *subframeStartPosition*"='s0') 1 = prepared for the case of a partial SF (higher layer parameter " *subframeStartPosition*"='s07') Valid if initial partial SF support configuration is set

Table 4-39 EPDCCH DL PDU Release 13 parameters

4.2.1.9 MPDCCH DL PDU

The format of the MPDCCH PDU is shown in Table 4-40. The MPDCCH DCI UL PDU contains the information which the L2/L3 software must provide the PHY so it can create the DCI formats 6-1A, 61B and 6-2 described in [12] sections 5.3.3.1. Transmission of MPDCCH in a subframe is indicated to L1 by the existence of the MPDCCH PDU in DL_CONFIG.request. Otherwise, MPDCCH is not transmitted. MPDCCH PDU is only included if the PHY supports Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205b	Release 13 parameters	struct	If the PHY supports MPDCCH- Release 13, the parameters described in Table 4-41 should be included.	V	TLV

Table 4-40 MPDCCH DL PDU

Field	Type	Description
MPDCCH Narrowband	uint8_t	Narrowband for MPDCCH Value: 0 → 15
Number of PRB pairs	uint8_t	Number of PRB-pairs constituting the MPDCCH-PRB-pair set Value: 2,4,6 (2+4) *See Note1 below
Resource Block Assignment	uint8_t	Combinational index r as defined in [9] section 9.1.4.4 Value: 0 → 14 *See Note1 below
MPDCCH transmission type	uint8_t	0 = Localized 1 = Distributed
Start symbol	uint8_t	Value: 1 → 4
ECCE index	uint8_t	CCE index used to send the DCI. Value: 0 → 22
Aggregation level	uint8_t	The aggregation level used Value: 2,4,8,16,24
RNTI type	uint8_t	0 = temporary C-RNTI 2 = RA-RNTI 3 = P-RNTI 4 = other
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Valid for all DCI formats Value: 1 → 65535.
CEMode	uint8_t	1 = CEModeA 2 = CEModeB
DMRS scrambling init	Uint16_t	The DMRS scrambling sequence initialization parameter $n_{\mathrm{ID},i}^{\mathrm{EPDCCH}}$ defined in [11] section 6.10.3A.1 Value: 0 → 503
Initial transmission SF (io)	Uint16_t	Absolute Sub-Frame of the initial transmission Value: 0 → 10239 Value: 0xFFFF current absolute SF
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in

Field	Type	Description
		0.001 dB steps.
DCI format	uint8_t	Format of the DCI 10 = 6-1A 11 = 6-1B 12 = 6-2
Resource block coding	uint16_t	The encoding for the resource blocks Valid for DCI formats: 6-1A,6-1B,6-2 Further information on the location of this bitstring within the resource block coding 16-bit parameter is implementation-specific
MCS	uint8_t	The modulation and coding scheme for the transport block Valid for DCI formats: 6-1A, 6-1B, 62 Value: 0 → 15
PDSCH repetition levels	uint8_t	Valid for DCI formats: 6-1A, 6-1B, 6-2 Value: 1 → 8 1 → 4 for DCI 6-1A 1 → 8 for DCI 6-1B and DCI 6-2
Redundancy version	uint8_t	The redundancy version for the transport block. Valid for DCI formats: 6-1A Value: 0 → 3
New data indicator	uint8_t	The new data indicator for the transport block. Valid for DCI formats: 6-1A, 6-1B
HARQ process	uint8_t	HARQ process number Valid for DCI formats: 6-1A, 6-1B Value: 0 →15
TPMI length	uint8_t	Length of "TPMI" field in units of bits. See Table 5.3.3.1.3A-1 in [12]. Valid for DCI formats: 6-1A Value: 0,2,4
TPMI	uint8_t	The codebook index to be used for precoding Valid for DCI formats: 6-1A 2 antenna_ports: 0 → 3 4 antenna_ports: 0 → 15

Field	Type	Description
PMI flag	uint8_t	Indicates if "PMI" field is present. Valid for DCI format 6-1A 0 = Not present. 1 = Present.
PMI	uint8_t	Confirmation for precoding Valid for DCI formats: 6-1A 0 = use precoding indicated in TPMI field 1 = use precoding indicated in last PMI report on PUSCH
HARQ resource offset	uint8_t	HARQ-ACK resource offset used for Δ_{ARO} determination. See [6] 10.1 Value: 0 →3
DCI subframe repetition number	uint8_t	Indicates the number of MPDCCH repetitions (r1, r2, r3, r4) See [9] section 9.1.5 Value: 1 →4
TPC	uint8_t	Tx power control command for PUCCH. Valid for DCI formats: 6-1A Value: 0,1,2,3
Downlink assignment index Length	uint8_t	Length of Downlink assignment index field in units of bits. See Table 5.3.3.1.2-2 of [9] Valid for DCI formats: 6-1A Value: 0, 2, 4
Downlink assignment index	uint8_t	The downlink assignment index. Only used in TDD mode, value ignored for FDD. Valid for DCI formats: 6-1A Value: 0 →15
Allocate PRACH flag	uint8_t	Indicates that PRACH procedure is initiated Valid for DCI formats: 6-1A, 6-1B 0 = false 1=true
Preamble index	uint8_t	The preamble index to be used on the PRACH

Field	Type	Description
		Valid for DCI formats: 6-1A Value: 0 → 63
PRACH mask index	uint8_t	The mask index to be used on the PRACH Valid for DCI formats: 6-1A Value: 0 → 15
Starting CE Level	uint8_t	2 bits provide the PRACH starting CE level Valid for DCI formats: 6-1A Value: 0 → 3
SRS request	uint8_t	SRS request flag Valid for DCI formats 6-1A. 0 = SRS not requested 1= SRS requested
Antenna ports and scrambling identity flag	uint8_t	Indicates if "Antenna ports and scrambling identity" field is present. Valid for DCI format 6-1A 0 = Not present. 1 = Present.
Antenna ports and scrambling identity	uint8_t	Indicates the Antenna port and, scrambling identity value Valid for DCI format 6-1A Value: 0->3
Frequency hopping enabled flag	uint8_t	Indicates if hopping is being used. Valid for DCI format 6-1A 0 = no hopping, 1= hopping enabled
Paging/Direct indication differentiation flag	uint8_t	Valid for DCI format 6-2 0 = Direct Information, 1= paging
Direct indication	uint8_t	Valid for DCI format 6-2 Value: 0->255
Total DCI length including padding	uint8_t	The total DCI length including padding bits
Number of TX Antenna ports	uint8_t	Number of TX physical antenna ports

Field		Type	Description
For Number of TX Antenna ports			
	Precoding value	uint16_t	Precoding element for physical antenna #i real 8 bits followed by imaginary 8 bits

Table 4-41 MPDCCH DL PDU Release 13 parameters

Note1: Clarification regarding MPDCCH "Number of PRB pairs" and "Resource Block Assignment" fields.

- In case of a single 2PRB set or a single 4PRB set is configured, the FAPI messagefields (number of PRB pairs, resource Block assignment) will be the same as numberPRB-Pairs-r11 and resourceBlockAssignment-r11 as configured in RRC message.
- In case of 2+4 PRB set is configured, depending on the candidate PRB set used at this TTI, the FAPI message fields (number of PRB pairs, resource Block assignment) shall be determined as follows: (note it may not be the same as in RRC message)

  - If candidate 2PRB set is used, number of PRB pairs set to 2, and resource Block assignment is set to the same value as in RRC message.
  - If candidate 4PRB set is used, number of PRB pairs set to 4, and resource Block assignment is derived to align with the PRB location of these PRBs.
  - If candidate 2+4 PRB set is used, number of PRB pairs set to 6, and the resource Block assignment corresponds to the value that all 6 RBs are allocated

4.2.1.10 N-BCH PDU

The format of the NBCH PDU is shown in Table 4-42. The PHY shall receive an N-BCH PDU once every 640ms.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205c	Release 13 parameters	struct	The PHY shall support Release 13 N-BCH if an NB-IOT cell and the parameters described in Table 4-43 should be included.	V	TLV

Table 4-42 NBCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in TX.request. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH, DLSCH or NBCH, NPDCCH, NDLSCH PDU is included in the

Field	Type	Description
		DL_CONFIG.request message. This value is repeated in TX.request and associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
Transmission power	uint16_t	Offset to the NRS power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
Hyper SFN 2lsbs	uint16_t	Hyper SFN 2 lsbs. Value: 0 → 3

Table 4-43 NBCH PDU Release 13 parameters

4.2.1.11 NPDCCH DL PDU

The format of the NPDCCH PDU is shown in Table 4-44.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205d	Release 13 parameters	struct	The PHY shall support Release 13 N-PDCCH if an NB-IOT cell and the Table 4-45 parameters described in should be included.	V	TLV

Table 4-44 NPDCCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in TX.request. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH, DLSCH or NBCH, NPDCCH, NDLSCH PDU is included in the DL_CONFIG.request message. This value is repeated in `TX.request` and associates the control information to the data. It is reset to 0 for every subframe Range 0 → 65535
NCCE index	uint8_t	NCCE index used to send the DCI. Value: 0 → 1
Aggregation level	uint8_t	The aggregation level used

Field	Type	Description
		Value: 1,2
Start symbol	uint8_t	The index l in the first slot in a subframe fulfils $l \geq l_{\mathrm{DataStart}}$ where $l_{\mathrm{DataStart}}$ is given by clause 16.6.1 of [9] Value: 0 → 4 (0 for guard-band and standalone operating modes)
RNTI type	uint8_t	0 = temporary C-RNTI 1 = RA-RNTI 2 = P-RNTI 3 = other
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Valid for all DCI formats Value: 1 → 65535.
Scrambling re-initialization batch index	uint8_t	The current SF's offset from scrambling re-initialization (scrambling sequence is re-initialized at start of search space and every 4 NPDCCH transmissions) Value: 1-4
NRS antenna ports assumed by the UE	uint8_t	The number of cell specific transmit antenna ports assumed by the UE. Refer to [11] section 10.2.5.5. Value:1,2
DCI format	uint8_t	Format of the DCI 0 = N1 1 = N2
Scheduling delay	unit8_t	The scheduling delay field (I_{Delay}, 3 bits) determines the scheduling delay (k_0) for NPDSCH. Refer to [9] section 16.4.1 (Table 16.4.1-1). Value: 0 → 7
Resource assignment	uint8_t	The resource assignment field (I_{SF}, 3 bits) determines the number of SFs (I_{SF}, 3bits) of the NPDSCH transport block. Refer to [9] section 16.4.1.3. Valid for DCI formats: N1, N2 (Paging = Paging/Direct indication field = 1) Value: 0 → 7
Repetition number	uint8_t	The repetition number field (I_{REP}, 4 bits) determines the number of repetitions (I_{REP}) for NPDSCH transport block. Refer to [9] section 16.4.1.3. Valid for DCI formats: N1, N2 (Paging =

Field	Type	Description
		Paging/Direct indication field = 1) Value: 0 → 15
MCS	uint8_t	The modulation and coding scheme field (4 bits) for the NPDSCH transport block Refer to [9] section 16.4.1.5 Valid for DCI formats: N1, N2 (Paging = Paging/Direct indication field = 1) Value: 0 → 13
New data indicator	uint8_t	The new data indicator for the transport block. Valid for DCI formats: N1 Value: 0-1 for NDI reserved for RA-RNTI
HARQ-ACK resource	uint8_t	The HARQ-ACK resource field (4 bits) determines the ACK/NACK subcarrier and delay (k_0). Refer to [9] section 16.4.2 (Table 16.4.2-1 for $\Delta f = 3.75\,\mathrm{kHz}$ subcarrier spacing and 16.4.2-2 for $\Delta f = 15\,\mathrm{kHz}$ subcarrier spacing) Valid for DCI formats: N1 Value: 0 → 15 reserved for RA-RNTI
NPDCCH order indication	uint8_t	Indicates that PRACH procedure is initiated by a "PDCCH Order" sent in DCI format N1 scrambled with C-RNTI. PDCCH order is described in section 16.3.2 in [9] 0 = false 1=true
Starting number of NPRACH repetitions	uint8_t	Repetition number field (I_{REP}) determines the PRACH repetition number according to Table 16.3.2-1 in [9] Valid for NPDCCH order indication (if flag=true) Value: 0: indicating $N_{REP} = R_1$ 1: indicating $N_{REP} = R_2$ 2: indicating $N_{REP} = R_3$ 3: Reserved
Subcarrier indication of NPRACH	uint8_t	Subcarrier indication field (I_{SC}) determines the allocated 3.75 KHz subcarrier for NPRACH. Valid for NPDCCH order indication (if flag=true) Value: 0→47 for SCs 0-47 48→63 are reserved
Paging/Direct	uint8_t	Valid for DCI format N2

Field	Type	Description
indication differentiation flag		0 = Direct Information, 1= paging
Direct indication	uint8_t	Valid for DCI format N2 with Direct indication, containing 8 bits provide direct indication of system information update and other fields, refer to [4] Section 6.6. Value: 0->255
DCI subframe repetition number	uint8_t	The DCI subframe repetition number field (2 bits) is used by the UE (together with max repetition number provided by higher layers) to determine the repetition number and allocation of NPDCCH. Refer to [9] section 16.6. Valid for DCI formats: N1 and N2 (paging = Paging/Direct indication field = 1) Value: 0 → 3 (Format N1) 0 → 7 (Format N2)
Total DCI length including padding	uint8_t	The total DCI length including padding bits. Zeros are appended to format N1 until the payload size is equal to that of format N0 Zeros (“reserved information”) are appended to format N2 with direct indication until size equal to format N2 paging

Table 4-45 NPDCCH PDU Release 13 parameters

4.2.1.12 NDLSCH PDU

The format of the NDLSCH PDU is shown in Table 4-46.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205e	Release 13 parameters	struct	The PHY shall support Release 13 N-DLSCH if an NB-IOT cell and the parameters described in Table 4-47 should be included.	V	TLV

Table 4-46 NDLSCH PDU

Field	Type	Description
Length	uint16_t	The length (in bytes) of the associated MAC PDU, delivered in TX.request. This should be the actual length of the MAC PDU, which may not be a multiple of 32-bits.
PDU index	uint16_t	This is a count value which is incremented every time a BCH, MCH, PCH, DLSCH or NBCH, NPDCCH, NDLSCH PDU is included in the DL_CONFIG.request message. This value is repeated in `TX.request` and associates the control

		information to the data. It is reset to 0 for every subframe Range 0 → 65535
Start symbol	uint8_t	The index l in the first slot in a subframe fulfils $l \geq l_{\text{DataStart}}$ where $l_{\text{DataStart}}$ is given by clause 16.6.1 of [9] Value: 0 → 4 (0 for guard-band and standalone operating modes)
RNTI type	uint8_t	0 = contains BCCH 1 = doesn't contain BCCH
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Value: 1 → 65535.
Resource assignment	uint8_t	The resource assignment field (I_{SF}, 3 bits) determines the number of SFs (N_{SF}) of the NPDSCH transport block. Refer to [9] section 16.4.1.3. This should match the value sent in the DCI Format PDU which allocated this grant. Value: 0-7 (mapped to 1,2,3,4,5,6,8,10 according to Table 16.4.1.3-1 in [9])
Repetition number	uint8_t	The repetition number field (I_{REP}, 4 bits) determines the number of repetitions (N_{REP}) for NPDSCH transport block. Refer to [9] section 16.4.1.3. This should match the value sent in the DCI Format PDU which allocated this grant. Value: 0-15 (mapped to 1,2,4,…2048 according to Table 16.4.1.3-2 in [9]) for PDSCH that doesn't contain BCCH (RNTI type = 1) 0-15 (mapped to 4,8,16 according to Table 16.4.1.3-3 in [9]) for PDSCH that doesn't contain BCCH (RNTI type = 1)
Modulation	uint8_t	2: QPSK
Number of subframes for resource assignment	uint8_t	The number of N_{SF} SFs the CW is split across. This should match the value sent in the DCI Format PDU which allocated this grant. Value: 1-10 for NPDSCH not containing BCCH 1-8 for NPDSCH containing BCCH)
Scrambling sequence initialization C_{init} [S_F,S_N]	uint16_t	The C_{init} of the scrambling sequence is based on SFN (S_F) and a slot (S_N) Value: [15] reserved [14:5] SFN, range 0 → 1023 [4:0] slot, range 0 → 19
SF_{idx}	uint16_t	The current SF in the current N_{SF}*N_{REP} SFs Value: 1-10240
NRS antenna ports	uint8_t	The number of cell specific transmit antenna ports assumed by

assumed by the UE		the UE. Refer to [11] section 10.2.5.5. Value:1,2

Table 4-47 NDLSCH PDU Release 13 parameters

4.2.2 UL_CONFIG.request

The format of the `UL_CONFIG.request` message is shown in Table 4-48. An `UL_CONFIG.request` message indicates the SFN/SF subframe it contains information for. This control information is for an uplink subframe.

This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE or CONFIGURED state an `ERROR.indication` message will be sent by the PHY.

For nFAPI the semi-static information and HARQ signalling calculation are located in the MAC. For FAPI the supported PDUs are dependent on whether the semi-static information and uplink HARQ signalling calculation is held in the MAC or PHY.

If the semi-static information and uplink HARQ signalling calculation is held in the MAC, the following combinations of PDUs are required:

- In order to support RACH in the subframe, the RACH present field must be true
- In order to support SRS in the subframe, the SRS present field must be true
- If the SRS present field is true, there can be 0 SRS PDU, or ≥1 SRS PDU present.
- The ULSCH PDU is present when a UE has been instructed to only send uplink data
- The ULSCH_CQI_RI, ULSCH_HARQ and ULSCH_CQI_HARQ_RI PDUs are present when a UE has been instructed to send uplink data and control
- The UCI_CQI, UCI_SR, UCI_SR_HARQ, UCI_CQI_HARQ, UCI_CQI_SR and UCI_CQI_SR_HARQ PDUs are present when a UE has been only been instructed to transmit control
- The ULSCH_UCI_CSI, ULSCH_UCI_HARQ and ULSCH_CSI_UCI_HARQ PDUs are present when UE has been instructed to transmit both data and control over PUSCH and PUCCH simultaneously. This was added in Release 10
- The following combinations can have the same RNTI values:
    - UCI_x + SRS
    - ULSCH_x + SRS

If the semi-static information and uplink HARQ signalling calculation is held in the PHY, the following combinations of PDUs are required:

- The ULSCH PDU is present when a UE has been instructed to send uplink data
- If the SRS present field is true, there can be ≥1 SRS PDU present

If the semi-static information is held in the MAC, and the uplink HARQ signalling calculation is held in the PHY, the following combinations of PDUs are required:

- In order to support RACH in the subframe, the RACH present field must be true
- In order to support SRS in the subframe, the SRS present field must be true
- If the SRS present field is true, there can be 0 SRS PDU, or ≥1 SRS PDU present.
- The ULSCH PDU is present when a UE has been instructed to only send uplink data
- The ULSCH_CQI_RI is present when a UE has been instructed to send uplink data and control
- The UCI_CQI, UCI_SR and UCI_CQI_SR PDUs are present when a UE has been only been instructed to transmit control
- The ULSCH_UCI_CSI PDU is present when UE has been instructed to transmit both data and control over PUSCH and PUCCH simultaneously. This was added in Release 10.
- The following combinations can have the same RNTI values:
  - ULSCH_x + SRS

If the semi-static information is held in the PHY, and the uplink HARQ signalling calculation is held in the MAC, the following combinations of PDUs are required:

- The ULSCH PDU is present when a UE has been instructed to send uplink data
- The ULSCH HARQ_PDU is present when a UE has been instructed to send uplink data, and an ACK/NACK response is expected
- The UCI_HARQ PDU is present when an ACK/NACK response is expected from a UE not transmitting uplink data
- The ULSCH_UCI_HARQ PDU is present when UE has been instructed to send data and control using simultaneous transmission of PUSCH and PUCCH channel. This was added in Release 10.

If NB-IOT is supported then the following PDUs are required:

- The NULSCH PDU is present when a UE has been instructed to send uplink data
- The NRACH PDU is present to indicate if NB-RACH is supported in the subframe

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
N/A	Length	uint16_t	The length of the downlink subframe configuration. Range 0 → 65535.	V	Not present
0x200c	UL config request body	struct	See Table 4-49.	V	TLV

Table 4-48 `UL_CONFIG.request` **message body**

Field	Type	Description
Number of PDUs	uint8_t	The number of UL SCHs PDUs included in this message.

Field	Type	Description
RACH/PRACH frequency resources	uint8_t	If semi-static information is held in the MAC For FDD: 0: No RACH in this subframe 1: RACH present in this subframe For TDD: Bits 0:5 indicate which RACH frequency resources are used in this subframe, see [11] section 5.7.1. 0: This RACH frequency index is not used 1: This RACH frequency index is used If semi-static information is held in the PHY this parameter is ignored.
SRS present	uint8_t	If semi-static information is held in the MAC 0: No SRS in this subframe 1: SRS present in this subframe
For number of PDUs		
PDU Type	uint8_t	0: ULSCH, see Section 4.2.2.1. 1: ULSCH_CQI_RI, see Section 4.2.2.2. 2: ULSCH_HARQ, see Section 4.2.2.3. 3: ULSCH_CQI_HARQ_RI, see Section 4.2.2.4. 4: UCI_CQI, see Section 4.2.2.8. 5: UCI_SR, see Section 4.2.2.9. 6: UCI_HARQ, see Section 4.2.2.10. 7: UCI_SR_HARQ, see Section 4.2.2.11. 8: UCI_CQI_HARQ, see Section 4.2.2.12. 9 : UCI_CQI_SR, see Section 4.2.2.13 10 : UCI_CQI_SR_HARQ, see Section 4.2.2.14 11: SRS, see Section 4.2.2.19. 12: HARQ_BUFFER, see Section 4.2.2.20 13: ULSCH_UCI_CSI, see Section 4.2.2.21. Added in Release 10 14: ULSCH_UCI_HARQ, sec Section 4.2.2.22. Added in Release 10 15: ULSCH_CSI_UCI_HARQ, see Section 4.2.2.23. Added in Release 10 16: NULSCH, see Section 4.2.2.24. Added in Release 13 17: NRACH, see Section 4.2.2.25. Added in Release 13.
PDU Size	uint8_t	Size of the PDU control information (in bytes). This length value includes the 2 bytes required for the PDU type and PDU size parameters.
UL PDU configuration	Struct	See sections 4.2.2.1 to 4.2.2.25.

Table 4-49 UL config request body

4.2.2.1 ULSCH PDU

The format of the ULSCH PDU is given in Table 4-50.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x200d	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-51 should be included.	V	TLV
0x200e	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-52 should be included	V	TLV
0x2043	Release 11 parameters	struct	If the PHY supports Release 11 the parameters described in Table 4-53 should be included	V	TLV
0x2044	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-54 should be included	V	TLV

Table 4-50 ULSCH PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in the `RX.indication`
Size	uint16_t	The size of the ULSCH PDU in bytes as defined by the relevant UL grant. The size can be 0 if UCI over ULSCH without data is configured. The size of CQI/RI/HARQ are not added to this element.
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU See [5] section 7.1 Value: 1 → 65535.
Resource block start	uint8_t	The starting resource block for this ULSCH allocation. This should match the value sent in the DCI Format 0 PDU which allocated this grant. Value: 0 → 99
Number of resource blocks	uint8_t	The number of resource blocks allocated to this ULSCH grant. This should match the value sent in the DCI Format 0 PDU which allocated this grant. Value: 1 → 100
Modulation type	uint8_t	2: QPSK 4: 16QAM

Field	Type	Description
		6: 64QAM
Cyclic Shift 2 for DMRS	uint8_t	The 2nd cyclic shift for DMRS assigned to the UE in the ULSCH grant. This should match the value sent in the DCI Format 0/4 PDU which allocated this grant. Value: 0 → 7
Frequency hopping enabled flag	uint8_t	Indicates if hopping is being used. This should match the value sent in the DCI Format 0 PDU which allocated this grant. See [9] Section 8.4. 0 = no hopping, 1= hopping enabled
Frequency hopping bits	uint8_t	The frequency hopping bits. This should match the value sent in the DCI Format 0 PDU which allocated this grant. See [9] Section 8.4 Value: 0 → 3
New data indication	uint8_t	Specify whether this received transport block is a new transmission from UE. This should match the value sent in the DCI Format 0 PDU which allocated this grant. If SPS is used then the MAC should calculate the value it would have transmitted in the NDI field of a DCI Format 0 and use this.
Redundancy version	uint8_t	Redundancy version Value: 0 → 3
HARQ process number	uint8_t	HARQ Process number. TDD 0 → 13 FDD 0 → 15
UL Tx mode	uint8_t	0 = SISO/SIMO 1 = MIMO
Current TX NB	uint8_t	The current HARQ transmission count of this transport block. Valid if frequency hopping enabled.
N srs	uint8_t	Indicates if the resource blocks allocated for this grant overlap with the SRS configuration. 0 = no overlap 1 = overlap

Table 4-51 ULSCH PDU Release 8 parameters

Field	Type	Description
Resource allocation type	uint8_t	Resource allocation type See [9] section 8.1

Field	Type	Description
		0 = type 0 1 = type 1
Resource block coding	uint32_t	Used for Resource Allocation type 1 The encoding for the resource blocks when resource allocation type is selected. This should match the value sent in the DCI Format PDU which allocated this grant. See [9] section 8.1.2 for the encoding format.
Transport blocks	uint8_t	The transport block transmitted from this RNTI. Added in Release 10 A value of 2 indicates this is the second transport block for DCI format 4. For other DCI values this field will always be 1. Value: 1 → 2
Transmission scheme	uint8_t	The MIMO mode used in the PDU See [9] section 8.0. 0: SINGLE_ANTENNA_PORT_10, 1: CLOSED_LOOP_SPATIAL_MULTIPLEXING, added in Release 10
Number Of layers	uint8_t	The number of layers used in transmission See [11] section 5.3.2A.2 Value: 1 → 4
Codebook Index	uint8_t	The codebook used for precoding Only valid when transmission scheme = 1 Defines the codebook used. When antenna port = 1: NA When antenna port = 2: 0..5 When antenna port = 4: 0..23
Disable sequence hopping flag	uint8_t	Indicates if any configured group hopping should be disabled for this UE. 0 = Any configured sequence hopping not disabled 1 = Any configured sequence hopping disabled

Table 4-52 ULSCH PDU Release 10 parameters

Field	Type	Description
Virtual cell ID enabled flag	uint8_t	Indicates if virtual cell is being used and nPUSCH identity is valid. 0 = not used, 1= enabled
nPUSCH Identity	uint16_t	Virtual cell ID for initialization of group hopping, sequence hopping and sequence shift pattern of PUSCH DMRS.

Field	Type	Description
		See [11] section 5.5.1.5 Valid if virtual cell ID is enabled. Value: 0 → 509
DMRS Config flag	uint8_t	Indicates if the PUSC DMRS config is valid. 0 = not used, 1= enabled
nDMRS-CSH Identity	uint16_t	Virtual cell ID for initialization of cyclic shift hopping of PUSCH DMRS. See [11] section 5.5.1.5 Valid if DMRS Config flag is enabled. Value: 0 → 509

Table 4-53 ULSCH PDU Release 11 parameters

Field	Type	Description
UE Type	uint8_t	Value: 0: non LC/CE UE 1: LC/CE CEModeA UE 2: LC/CE CEModeB UE
Total Number of repetitions	Uint16_t	Value: 1 → 2048 Value 1 indicates single allocation
Repetition Number	Uint16_t	Current transmission number Value: 1 → 2048
Initial transmission SF (io)	Uint16_t	Absolute Sub-Frame of the initial transmission Value: 0 → 10239 Value: 0xFFFF current absolute SF
Empy symbols due to re-tunning	Uint8_t	Indicates the symbols that are left empty due to eMTC retuning. Value: Bitmap of 4 bits. A bit value of 1 indicates the symbol is empty A bit value of 0 indicates the symbol is not empty bit[0] configures symbol 13 bit[1] configures symbol 12 bit[2] configures symbol 1 bit[3] configures symbol 0

Table 4-54 ULSCH PDU Release 13 parameters

4.2.2.2 ULSCH_CQI_RI PDU

The format of the ULSCH_CQI_RI PDU is given in Table 4-55. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
CQI_RI	struct	Description of contents given in Table 4-60

Field	Type	Description
information		
Initial transmission parameters	struct	Description of contents given in Table 4-58

Table 4-55 ULSCH_CQI_RI PDU

4.2.2.3 ULSCH_HARQ PDU

The format of the ULSCH_HARQ PDU is given in Table 4-56. This PDU is only valid if the uplink HARQ signalling calculation is held in the MAC.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
HARQ information	struct	Description of contents given in Table 4-67
Initial transmission parameters	struct	Description of contents given in Table 4-58

Table 4-56 ULSCH_HARQ PDU

4.2.2.4 ULSCH_CQI_HARQ_RI PDU

The format of the ULSCH_CQI_HARQ_RI PDU is given in Table 4-57. This PDU is only valid if both semi-static information and the uplink HARQ signalling calculation are held in the MAC.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
CQI_RI information	struct	Description of contents given in Table 4-60
HARQ information	struct	Description of contents given in Table 4-67
Initial transmission parameters	struct	Description of contents given in Table 4-58

Table 4-57 ULSCH_CQI_HARQ_RI PDU

4.2.2.5 Initial transmission parameters

The format of the initial transmission parameters used in ULSCH PDUs is given in Table 4-58.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x200f	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-59 should be included.	V	TLV

Table 4-58 Initial transmission information

Solving the HetNet puzzle

Field	Type	Description
N srs initial	uint8_t	0 = last OFDM symbol is not punctured 1 = last OFDM symbol is punctured.
Initial number of resource blocks	uint8_t	The number of resource blocks used in the initial transmission of this transport block. Value: 1 → 100

Table 4-59 Initial transmission information Release 8 parameters

4.2.2.6 CQI RI Information (ULSCH)

The format of the CQI RI information used in ULSCH PDUs is given in Table 4-60.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2010	Release 8 parameters	struct	If the PHY only supports Release 8 then the parameters described in Table 4-61 should be included.*	V	TLV
0x2011	Release 9 or later parameters	struct	If the PHY supports Release 9 or later then the parameters described in Table 4-62 should be included.*	V	TLV
0x2045	Release 13 parameters	struct	If the PHY supports Release 13 then in addition to Release 9 parameters the parameters described in Table 4-64 should be included.	V	TLV

Table 4-60 CQI_RI information

*Either Rel8 or Rel9 parameters are included, not both.

The format of the CQI_RI information is given in Table 4-61 for Release 8 versions of this API. The aperiodic CQI demultiplexing depends on the RI value. Depending on transmission mode, report mode and RI value the CQI sizes might be different, therefore, in order to allow correct decoding CQI sizes for both rank=1 and rank>1 should be provided.

Field	Type	Description
DL CQI/PMI Size Rank = 1	uint8_t	The size of the DL CQI/PMI in bits in case of rank 1 report. Value: 0 → 255
DL CQI/PMI Size Rank>1	uint8_t	The size of the DL CQI/PMI in bits in case of rank>1 report. Value: 0 → 255 In case rank is not reported, CQI/PMI size for

Field	Type	Description
		rank=1 and rank>1 size should be the same
RI Size	uint8_t	The size of RI in bits Value:0 → 2 0 indicates that RI is not reported
Delta Offset CQI	uint8_t	Delta offset for CQI. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.6.3 Value: 0 → 15
Delta Offset RI	uint8_t	Delta offset for RI. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.6.3 Value: 0 → 15

Table 4-61 CQI_RI information for Release 8

The format of the CSI information is given in Table 4-62 for Release 9 and later versions of this API. The aperiodic CQI demultiplexing depends on the RI values reported for each CC. For each reported CC depending on transmission mode, report mode and RI value the CQI sizes might be different. Therefore, in order to allow correct decoding the RI sizes and CQI sizes for each possible rank value should be provided for each CC.

Field	Type	Description
Report type	uint8_t	Type of CSI report 0 = periodic report 1 = aperiodic report
Delta offset CQI	uint8_t	Delta offset for CQI. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.6.3 Value: 0 → 15
Delta offset RI	uint8_t	Delta offset for RI. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.6.3 Value: 0 → 15
CQI/PMI/RI	struct	For periodic report, see Table 4-63 For aperiodic report, see Table 4-65

Table 4-62 CQI_RI information for Release 9 and later

Field	Type	Description
DL CQI/PMI/RI size	uint8_t	The size of the DL CQI/PMI/RI/CRI in bits. Value: 0 → 255
Control Type	uint8_t	0 – CQI/PMI 1 – RI/CRI

Table 4-63 CQI/PMI/RI periodic report for Release 9 and later

Field	Type	Description
CQI/PMI/RI	struct	For periodic report, see Table 4-66 Aperiodic report no Release 13 parameters are defined

Table 4-64 CQI_RI information for Release 13

Field	Type	Description
Number of CC	uint8_t	The number of CC in the aperiodic report Value: 1 → 32
For each CC		
RI Size	uint8_t	The size of RI or CRI in bits Value:0 → 3 0 indicates that RI /CRI is not reported
For rank value 1 to 2^{RI_size}		
DL CQI/PMI size	uint8_t	The size of the DL CQI/PMI in bits in case of this RI value. The size of the DL CQI/PMI/RI in bits in case of this CRI value. Value: 0 → 255

Table 4-65 CQI/PMI/RI a periodic report for Release 9 and later

Field	Type	Description
DL CQI/PMI/RI size 2	uint16_t	The size of the DL CQI/PMI/RI in bits. If the CQI/PMI/RI size exceeds 255 (8-bits) then the Release 9 size value = 0, and this field is used instead. Value: ≥ 255

Table 4-66 CQI/PMI/RI periodic report for Release 13

4.2.2.7 HARQ Information (ULSCH)

The format of the HARQ information for ULSCH PDUs is given in Table 4-67.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2012	Release 10 parameters	struct	The PHY shall support Release 8, 9 or 10 and the parameters described in	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
			Table 4-68 should be included.		
0x2046	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-69 should be included.	V	TLV

Table 4-67 ULSCH_HARQ information

Field	Type	Description
HARQ Size	uint8_t	The size of the ACK/NACK in bits. Value: 1 → 21 for Format 1a/1b/3 Value: 0 for Format 4/5 Values greater than 2 were added in Release 10, see [9] Section 10.1.1
Delta Offset HARQ	uint8_t	Delta offset for HARQ. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.6.3 Value: 0 → 15
ACK_NACK mode	uint8_t	The format of the ACK/NACK response expected. For TDD only. 0 = BUNDLING 1 = MULTIPLEXING 2 = Format 1b with channel selection 3 = Format 3 4 = Format 4 5 = Format 5

Table 4-68 HARQ information Release 10 parameters

Field	Type	Description
HARQ Size 2	uint16_t	The size of the ACK/NACK in bits. Valid for ACK_NACK mode = 4 or 5
Delta Offset HARQ 2	uint8_t	Delta offset 2 for HARQ. This value is fixed for a UE, allocated in RRC connection setup and used for ACK_NACK mode = 4 or 5. See [9] section 8.6.3 Value: 0 → 15

Table 4-69 HARQ information Release 13 parameters

4.2.2.8 UCI_CQI PDU

The format of the UCI_CQI PDU is given in Table 4-70. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_CQI.indication.
CQI Information	struct	Description of contents given in Table 4-81

Table 4-70 UCI_CQI PDU

4.2.2.9 UCI_SR PDU

The format of the UCI_SR PDU is given in Table 4-71. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_SR.indication
SR Information	struct	Description of contents given in Table 4-85.

Table 4-71 UCI_SR PDU

4.2.2.10 UCI_HARQ PDU

The format of the UCI_HARQ PDU is given in Table 4-72. This PDU is only valid if the uplink HARQ signalling calculation is held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in HARQ.indication
HARQ Information	struct	Description of contents given in Table 4-88

Table 4-72 UCI_HARQ PDU

4.2.2.11 UCI_SR_HARQ PDU

The format of the UCI_SR_HARQ PDU is given in Table 4-73. This PDU is only valid if both semi-static information and the uplink HARQ signalling calculation are held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_SR.indication and HARQ.indication
SR information	struct	Description of contents given in Table 4-85
HARQ information	struct	Description of contents given in Table 4-88

Table 4-73 UCI_SR_HARQ PDU

4.2.2.12 UCI_CQI_HARQ PDU

The format of the UCI_CQI_HARQ PDU is given in Table 4-74. This PDU is only valid if both semi-static information and the uplink HARQ signalling calculation are held in the MAC.

For TDD when both HARQ and CQI, or HARQ and SR, are transmitted on PUCCH, multiple HARQ ACK/NACK responses are bundled according to table 7.3-1 of [9]. This is referred to in Table 4-136 and Table 4-137 as 'Special Bundling' and implies a unique interpretation of the message fields.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_CQI.indication and HARQ.indication
CQI Information	struct	Description of contents given in Table 4-81
HARQ Information	struct	Description of contents given in Table 4-88

Table 4-74 UCI_CQI_HARQ PDU

4.2.2.13 UCI_CQI_SR_PDU

The format of the UCI_CQI_SR PDU is given in Table 4-75. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_CQI.indication and RX_SR.indication
CQI Information	struct	Description of contents given in Table 4-81
SR Information	struct	Description of contents given in Table 4-85

Table 4-75 UCI_CQI_SR PDU

4.2.2.14 UCI_CQI_SR_HARQ_PDU

The format of the UCI_CQI_SR HARQ_PDU is given in Table 4-76. This PDU is only valid if both semi-static information and the uplink HARQ signalling calculation are held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77. Handle returned in RX_CQI.indication, RX_SR.indication and HARQ.indication
CQI Information	struct	Description of contents given in Table 4-81
SR Information	struct	Description of contents given in Table 4-85
HARQ Information	struct	Description of contents given in Table 4-88

Table 4-76 UCI_CQI_SR_HARQ PDU

4.2.2.15 UE Information

The format of the UE information used in UCI PDUs is given in Table 4-77.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2013	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-78 should be included.	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2047	Release 11 parameters	struct	The PHY shall support Release 11 and the parameters described in Table 4-79 should be included.	V	TLV
0x2048	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-80 should be included.	V	TLV

Table 4-77 UE information

Field	Type	Description
Handle	uint32_t	An opaque handling returned in the relevant indication message.
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU See [5] section 7.1 Value: 1 → 65535.

Table 4-78 UE information Release 8 parameters

Field	Type	Description
Virtual cell ID enabled flag	uint8_t	Indicates if virtual cell is being used and nPUCCH identity is valid. 0 = not used, 1= enabled
nPUCCH Identity	uint16_t	Virtual cell ID for initialization of base sequence and cyclic shift hopping of PUCCH. See [11] section 5.5.1.5 Valid if virtual cell ID is enabled. Value: 0 → 503

Table 4-79 UE information Release 11 parameters

Field	Type	Description
UE Type	uint8_t	Value: 0: non LC/CE UE 1: LC/CE CEModeA UE 2: LC/CE CEModeB UE
Empty symbols	uint8_t	Indicates the symbols that are left empty due to eMTC retuning. Value: Bitmap of 2 bits. A bit value of 1 indicates the symbol is empty A bit value of 0 indicates the symbol is not empty

Field	Type	Description
		bit[0] configures symbol 13 bit[1] configures symbol 0
Total Number of repetitions	Uint16_t	Value: 1 → 32 Value 1 indicates single allocation
Repetition Number	Uint16_t	Current transmission number Value: 1 → 32

Table 4-80 UE information Release 13 parameters

4.2.2.16 CQI information (UCI)

The format of the CQI information used in UCI PDUs is given in Table 4-81.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2014	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-82 should be included.	V	TLV
0x2015	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-83 should be included.	V	TLV
0x2049	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-84 should be included.	V	TLV

Table 4-81 CQI information

Field	Type	Description
PUCCH index	uint16_t	The PUCCH index value $n_{\mathrm{PUCCH}}^{(2)}$ Value: 0 → 1184
DL CQI/PMI Size	uint8_t	The size of the DL CQI/PMI in bits. Value: 0 → 255

Table 4-82 CQI information Release 8 parameters

Field	Type	Description
Number of PUCCH Resources	uint8_t	A value of 2 indicates that the UE is configured to transmit on two antenna ports Value: 1 → 2
PUCCH Index P1	uint16_t	The PUCCH index value $n_{\mathrm{PUCCH}}^{(2)}$ for antenna port P1 Only valid when Number of PUCCH resources is 2. Value: 0 → 1184

Table 4-83 CQI information Release 10 parameters

Field	Type	Description
CSI_mode	uint8_t	0 = PUCCH format 2/2a/2b/3 1 = PUCCH format 4 2 = PUCCH format 5
DL CQI/PMI Size 2	uint16_t	The size of the DL CQI/PMI in bits. Valid for CSI mode 1 or 2, where the field DL CQI/PMI Size will be set to 0.
Starting PRB	uint8_t	The starting PRB for the PUCCH Value: 0->109 CSI mode = 1 or 2) See [36.331] Section PUCCH config information element
n_PRB	uint8_t	The number of PRB in PUCCH Value: 0 → 7 (CSI mode = 1) See [36.331] Section PUCCH config information element
cdm_Index	uint8_t	Selected CDM option Value: 0 → 1 (CSI mode = 2) See [36.331] Section PUCCH config information element
N srs	uint8_t	Indicates if the resource blocks allocated for this grant overlap with the SRS configuration. 0 = no overlap 1 = overlap Valid for CSI mode = 2

Table 4-84 CQI information Release 13 parameters

4.2.2.17 SR information

The format of the SR Information used in UCI PDUs is given in Table 4-85.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2016	Release 8 parameters	struct	The PHY shall support Release 8 and the	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
			parameters described in Table 4-86 should be included.		
0x2017	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-87 should be included.	V	TLV

Table 4-85 SR information

Field	Type	Description
PUCCH Index	uint16_t	The PUCCH Index value $n^{(1)}_{\mathrm{PUCCH}}$ Value: 0 → 2047

Table 4-86 SR information Release 8 parameters

Field	Type	Description
Number of PUCCH Resources	uint8_t	A value of 2 indicates that the UE is configured to transmit SR on two antenna ports Value: 1 → 2
PUCCH Index P1	uint16_t	The PUCCH Index value $n^{(1)}_{\mathrm{PUCCH}}$ for antenna port P1. Only valid when Two Antenna Port Activated is True. Value: 0 → 2047

Table 4-87 SR information Release 10 parameters

4.2.2.18 HARQ information (UCI)

The format of the HARQ information used in UCI PDUs is given in Table 4-88 and dependent on whether TDD or FDD mode is used.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2018	Release 10 TDD parameters	struct	The PHY shall support Release 8, 9 or 10 and if it is configured to TDD mode the parameters described in Table 4-89 should be included. [1]	V	TLV
0x2019	Release 8 FDD parameters	struct	If the PHY only supports Release 8 and is configured for FDD mode the parameters described in Table 4-90 should be included. [12]	V	TLV
0x201a	Release 9 or later FDD	struct	If the PHY supports	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
	parameters		Release 9 or 10 and is configured for FDD mode the parameters described in Table 4-91 should be included. [12]		
0x204a	Release 11 FDD/TDD parameters	struct	If the PHY supports Release 11 and is configured for either FDD or TDD mode the parameters described in TTable 4-92 should be included.[3]	V	TLV
0x204b	Release 13 FDD/TDD parameters	struct	If the PHY supports Release 13 and is configured for either FDD or TDD mode the parameters described in Table 4-93 should be included. [4]	V	TLV

Table 4-88 HARQ information

[1]Only FDD or TDD parameters are included.
[2]For FDD either Rel8 or Rel9 parameters are included, not both.
[3]If Release 11 is supported then these fields are also included.
[4]If Release 13 is supported then these fields are also included

Field	Type	Description
HARQ size	uint8_t	Size of the ACK/NACK in bits. Values greater than 4 were added in Release 10, see [9] Section 10.1.1 Value 1..21 for Format 1a/1B/3 Value: 0 for Format 4/5
ACK_NACK mode	uint8_t	The format of the ACK/NACK response expected. 0 = Format 1a/1b BUNDLING 1 = Format 1a/1b MULTIPLEXING 2 = Format 1b with channel selection, added in Release 10 3 = Format 3, added in Release 10 4 = Format 4, added in Release 13 5 = Format 5, added in Release 13
Number of PUCCH resources	uint8_t	The number of PUCCH resources associated with the ACK/NACK response on antenna port 0. See [9] section 10.1 Value: 1 → 4 for Format 1a/1B/3 Value: 0 for Format 4/5 Resources for antenna port 1 are in n_PUCCH_2_x.
n_PUCCH_1_0	uint16_t	HARQ resource 0

Field	Type	Description
		Value: 0 → 2047 (ACK_NACK mode = All Format 1a/1b) Value: 0 → 549 (ACK_NACK mode = Format 3)
n_PUCCH_1_1	uint16_t	HARQ resource 1 Value: 0 → 2047 (ACK_NACK mode =All Format 1a/1b)
n_PUCCH_1_2	uint16_t	HARQ resource 2, value: 0 → 2047 (ACK_NACK mode = Format 1a/1b with multiplexing or channel selection)
n_PUCCH_1_3	uint16_t	HARQ resource 3, value: 0 → 2047 (ACK_NACK mode = Format 1a/1b with multiplexing or channel selection)

Table 4-89 HARQ information for TDD for Release 10

For FDD Release 8 only the PUCCH index value is required.

Field	Type	Description
n_PUCCH_1_0	uint16_t	The PUCCH Index value for ACK/NACK Value: 0 → 2047 (ACK_NACK mode = 0 or 1) Value: 0 → 549 (ACK_NACK mode = 2)
HARQ Size	uint8_t	The size of the ACK/NACK in bits. Value: 1 → 2

Table 4-90 HARQ information for FDD for Release 8

For Release 10 FDD three HARQ ACK/NACK modes are defined, namely PUCCH Format 1a/1b, Format 1b with channel selection, and PUCCH Format 3

Field	Type	Description
HARQ Size	uint8_t	The size of the ACK/NACK in bits. Value: 1 → 10 for Format 1a/1B/3 Value: 0 for Format 4/5 Values greater than 2 were added in Release 10
ACK_NAK mode	uint8_t	The format of the ACK/NACK response expected. 0 = Format 1a/1b 1 = Channel Selection, added in Release 10 2 = Format 3, added in Release 10 3 = Format 4, added in Release 13 4 = Format 5, added in Release 13
Number of PUCCH Resources	uint8_t	The number of PUCCH resources associated with the ACK/NACK response on antenna port 0 See [9] section 10.1 Value: 1 → 4 for Format 1a/1b/3 Value: 0 for Format 4/5

Field	Type	Description
		Resources for antenna port 1 are in n_PUCCH_2_x.
n_PUCCH_1_0	uint16_t	The PUCCH Index value for ACK/NACK HARQ resource 0 Value: 0 → 2047 (ACK_NACK mode = Format 1a/1b or channel selection) Value: 0 → 549 (ACK_NACK mode = Format 3)
n_PUCCH_1_1	uint16_t	HARQ resource 1 Value: 0 → 2047 (ACK_NACK mode = Format 1a/1b or channel selection)
n_PUCCH_1_2	uint16_t	HARQ resource 2 Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection)
n_PUCCH_1_3	uint16_t	HARQ resource 3 Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection)

Table 4-91 HARQ information for FDD for Release 9 and 10

Field	Type	Description
Num_ant_ports	uint8_t	The number of antenna ports used by the UE transmit Value: 1->2 See [9] section 10.1.2.2.1 for two antenna port transmission for format 1b with channel selection, and Section 10.1.2.2.2 for two antenna port transmission for format 3.
n_PUCCH_2_0	uint16_t	The PUCCH Index value for ACK/NACK HARQ resource 4 on antenna port Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection) Value: 0 → 549 (ACK_NACK mode = Format 3)
n_PUCCH_2_1	uint16_t	HARQ resource 5 Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection)
n_PUCCH_2_2	uint16_t	HARQ resource 6 Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection)
n_PUCCH_2_3	uint16_t	HARQ resource 7 Value: 0 → 2047 (ACK_NACK mode = Format 1b with channel selection)

Table 4-92 HARQ information for FDD and TDD Release 11 parameters

Field	Type	Description
HARQ Size 2	uint16_t	The size of the ACK/NACK in bits. Valid for ACK_NACK mode = Format 4 or 5

Field	Type	Description
Starting PRB	uint8_t	The starting PRB for the PUCCH Value: 0->109 (ACK_NACK mode = Format 4 or 5) See [4] Section 6.2.2 PUCCH config information element
n_PRB	uint8_t	The number of PRB in PUCCH Value: 0 → 7 (ACK_NACK mode = Format 4) See [4] Section 6.2.2 PUCCH config information element
cdm_Index	uint8_t	Orthogonal sequence Index Value: 0 → 1 (ACK_NACK mode = Format 5) See [4] Section 6.2.2 PUCCH config information element
N srs	uint8_t	Indicates if the resource blocks allocated for this grant overlap with the SRS configuration. 0 = no overlap 1 = overlap Valid for ACK_NACK mode = Format 5

Table 4-93 HARQ information for FDD and TDD for Release 13 parameters

4.2.2.19 SRS

The format of the SRS PDU is given in Table 4-94. This PDU is only valid if semi-static information is held in the MAC.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x201b	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-95 should be included.	V	TLV
0x201c	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-96 should be included	V	TLV
0x204c	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-97 should be included	V	TLV

Table 4-94 SRS PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in the SRS.indication

Field	Type	Description
Size	uint16_t	The size of the PDU in bytes.
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU See [5] section 7.1 Value: 1 → 65535.
SRS Bandwidth	uint8_t	SRS Bandwidth. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2 Value: 0 → 3
Frequency Domain Position	uint8_t	Frequency-domain position, N_{RRC} This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2 Value: 0 → 23
SRS Hopping Bandwidth	uint8_t	Configures the frequency hopping on the SRS. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.2. Value 0 → 3
Transmission Comb	uint8_t	Configures the frequency location of the SRS. This value is fixed for a UE and allocated in RRC connection setup. Value: 0 → 3
I_{SRS} / SRS-ConfigIndex	uint16_t	Defines the periodicity and subframe location of the SRS. SRS Configuration Index. This value is fixed for a UE and allocated in RRC connection setup. See [9] section 8.2. Value: 0 → 1023.
Sounding Reference Cyclic Shift	uint8_t	Configures the SRS sequence generation. This value is fixed for a UE and allocated in RRC connection setup. See [11] section 5.5.3.1. Value: 0 → 11

Table 4-95 SRS PDU Release 8 parameters

Field	Type	Description
Antenna Port	uint8_t	Defines the number of antenna ports used by the UE for the SRS. This value is fixed for a UE and allocated in RRC connection setup. 0 = 1 antenna port

Field	Type	Description
		1 = 2 antenna ports 2 = 4 antenna ports

Table 4-96 SRS PDU Release 10 parameters

Field	Type	Description
Number of Combs	uint8_t	Defines the maximum number of transmission combs (TC). See [11] section 5.5.3.1. 0 = 2 TC 1 = 4 TC

Table 4-97 SRS PDU Release 13 parameters

4.2.2.20 HARQ_BUFFER PDU

The HARQ Buffer PDU indicates for which UE the HARQ buffer should be released. The format of the HARQ_BUFFER PDU is given in Table 4-98. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
UE information	struct	Description of contents given in Table 4-77.

Table 4-98 HARQ buffer release PDU

4.2.2.21 ULSCH_UCI_CSI PDU

This PDU is used when data and CSI information is transmitted using simultaneous PUSCH and PUCCH. This PDU is only valid if semi-static information is held in the MAC.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
CSI information	struct	Description of contents given in Table 4-81

Table 4-99 ULSCH_UCI_CSI PDU

4.2.2.22 ULSCH_UCI_HARQ PDU

This PDU is used when data and ACK/NACK is transmitted using simultaneous PUSCH and PUCCH. This PDU is only valid if uplink HARQ calculation is done MAC layer.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
HARQ information	struct	Description of contents given in Table 4-88

Table 4-100 ULSCH_UCI_HARQ PDU

4.2.2.23 ULSCH_CSI_UCI_HARQ PDU

This PDU is used when data, ACK/NACK and CSI information is transmitted using simultaneous PUSCH and PUCCH. This PDU is only valid if semi-static information and the uplink HARQ calculation are held in the MAC.

Field	Type	Description
ULSCH PDU	struct	Description of contents given in Table 4-50
CSI information	struct	Description of contents given in Table 4-60
HARQ information	struct	Description of contents given in Table 4-88

Table 4-101 ULSCH_CSI_UCI_HARQ PDU

4.2.2.24 NULSCH PDU

The NULSCH PDU format is shown in Table 4-102.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205f	Release 13 parameters	struct	The PHY shall support Release 13 N-ULSCH if an NB-IOT cell and the parameters described in Table 4-103 should be included.	V	TLV

Table 4-102 NULSCH PDU

Field	Type	Description
NPUSCH format	uint8_t	NPUSCH format: Value: 0 = NPUSCH format 1 (data) 1 = NPUSCH format 2 (ACK/NACK)
Handle	uint32_t	An opaque handling returned in the `RX.indication`
Size	uint16_t	The size of the NULSCH PDU in bytes as defined by the relevant UL grant.
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Value: 1 → 65535.
Subcarrier indication	uint8_t	Subcarrier indication field (I_{SC}) determines the allocated NPUSCH subcarriers either for 15 KHz or 3.75 KHz. - For NPUSCH format 1, I_{SC}=6 bits, indicating #SCs for every SC spacing - For NPUSCH format 2, I_{SC}= 2 bits for 15KHz, 3 bits for 3.75KHz, indicating #SC allocation Refer to [9] section 16.5.1. This should match the value sent in the DCI Format N0 PDU which allocated this grant Value: For PUSCH format 1 0 → 18 for 15 KHz (12,4,2,1 options for 1,3,6,12 SCs) 0 → 47 for 3.75 KHz (0-47 for 1 SC) For PUSCH format 2 0 → 3 for 15 KHz (Offset value {0,1,2,3}) 0 → 7 for 3.75 KHz (Offset values {0, -1, -2,

		-3, -4, -5, -6, -7})
Resource assignment	uint8_t	Valid for NPUSCH format 1. The resource assignment field (I_{RU}, 3 bits) determines the Transport block size of NPUSCH (TBS) of the NPUSCH transport block (together with the MCS index). Refer to [9] section 16.5.1.2. This should match the value sent in the DCI Format N0 PDU which allocated this grant Value: 0 → 7 mapped to one of predefined set of N_{RU} {1,2,3,4,5,6,8,10)
MCS	uint8_t	Valid for NPUSCH format 1. The modulation and coding scheme field (4 bits) for the NPUSCH transport block, determines the modulation (QPSK for multi tone, p2/2 BPSK or pi/4 QPSK for single tone) and the size of NPUSCH (I_{TBS}) Refer to [9] section 16.5.1.2 This should match the value sent in the DCI Format N0 PDU which allocated this grant Value: 0 → 10 for single tone NPUSCH 0 → 12 for multi tone NPUSCH
Redundancy version	uint8_t	Valid for NPUSCH format 1. The redundancy version (RV_{idx}) for the NPUSCH transport block Refer to [9] section 16.5.1.2 Value: 0 for RV=0 1 for RV=2
Repetition number	uint8_t	Valid for both NPUSCH formats. The repetition number field (I_{REP} for NPUSCH format 1 in DCI0, N_{Rep}^{AN} for NPUSCH format 2 in *ack-NACK-NumRepetitions /ack-NACK-NumRepetitions-Msg4*) determines the number of repetitions (N_{REP}) for NPUSCH transport block or for an HARQ-ACK report. Refer to [9] section 16.5.1.2 for NUPSCH format 1 and [9] section 16.4.2.for NUPSCH format 2. Value: 0 → 7 mapped to one of predefined set of N_{REP} {1,2,4,8,16,32,64,128}
New data indication	uint8_t	Valid for NPUSCH format 1. Specify whether this received transport block is a new transmission from UE. This should match the value sent in the DCI Format N0 PDU which allocated this grant.
N srs	uint8_t	Valid for both NPUSCH formats for an in-band operating mode. Indicates if the resource blocks allocated for this grant overlap with the SRS configuration. 0 = no overlap 1 = overlap
Scrambling sequence initialization C_{init} [S_F,S_N]	uint8_t	Valid for both NPUSCH formats. The C_{init} of the scrambling sequence is based on SFN (S_F) and a slot (S_N) Value: [15] reserved [14:5] SFN, range 0 → 1023

		[4:0] slot, range 0 → 19
SF_{idx}	Uint16_t	Valid for both NPUSCH formats. The current SF. For NPUSCH format 1: - Value: 1-10240 (N_{RU}*N_{REP}*N^{UL}_{slots} /2 SFs for 15KHz) - Value: 1-40960 (N_{RU}*N_{REP}*N^{UL}_{slots} *2 SFs for 3.75KHz) For NPUSCH format 2: - Value: 1-256 (N_{REP}*N^{UL}_{slots} /2 SFs for 15KHz) - Value: 1-1024 (N_{REP}*N^{UL}_{slots} *2 SFs for 3.75KHz)
UE information	struct	Valid for NPUSCH format 2. Description of contents given in Table 4-77. Handle returned in NB-HARQ.indication
NB-HARQ information	struct	Valid for NPUSCH format 2. Description of contents given in Table 4-104

Table 4-103 NULSCH PDU Release 13 parameters

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2061	Release 13 FDD parameters	struct	If the PHY only supports Release 13 and is configured for NB-IOT the parameters described in Table 4-105 should be included.	V	TLV

Table 4-104 NB-HARQ information

Field	Type	Description
HARQ-ACK resource	uint8_t	The HARQ-ACK resource field (4 bits) determines the ACK/NACK subcarrier Refer to [9] section 16.4.2 (Table 16.4.2-1 for $\Delta f = 3.75\,\text{kHz}$ subcarrier spacing and 16.4.2-2 for $\Delta f = 15\,\text{kHz}$ subcarrier spacing) This should match the value sent in the DCI Format N1 PDU which allocated the equivalent DL for this ACK Value: 0 → 15 for 15 KHz (SCs 0-3 with different k0) 0 → 15 for 3.75 KHz (SCs 38-45 with different k0)

Table 4-105 NB-HARQ information for Release 13

4.2.2.25 NRACH PDU

The NRACH PDU format is shown in Table 4-106. If semi-static information is held in the MAC then this PDU is included to indicate that this is an NRACH subframe. If there is no NRACH then this PDU is not included.

A new PDU is included for NB-IOT to allow signalling for the the different NPRACH configurations.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2067	Release 13 parameters	struct	The PHY shall support Release 13 NRACH in an NB-IOT cell and the parameters described in Table 4-107 should be included.	V	TLV

Table 4-106 NRACH PDU

Field	Type	Description
NPRACH config#0	uint8_t	0: No RACH in this subframe 1: RACH present in this subframe
NPRACH config#1	uint8_t	0: No RACH in this subframe 1: RACH present in this subframe
NPRACH config#2	uint8_t	0: No RACH in this subframe 1: RACH present in this subframe

Table 4-107 NRACH PDU Release 13 parameters

4.2.3 HI_DCI0.request

The format of the `HI_DCI0.request` message is given in Table 4-108. This message contains two types of control information relating to the uplink. Firstly, it is used for the L2/L3 to send the ACK/NACK response for MAC PDUs received on the ULSCH. LTE has strict timing requirements for returning this information to the UE. Secondly, it is used for DCI control format information relating to the uplink which is broadcast on the PDCCH.

Section 3.2.2.3 contains a detailed description on when this message should be sent, and the correct value of SFN/SF.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x201d	HI DCI0 body	struct	See Table 4-109.	V	TLV

Table 4-108 `HI_DCI0.request` **message body**

Field	Type	Description
SFN/SF	uint16_t	The SFN/SF in this message should be the same as the corresponding DL_CONFIG.request message. A 2-byte value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9
Number of DCI	uint8_t	Number of DCI PDUs included in this message

Field	Type	Description
Number of HI	uint8_t	Number of HI PDUs included in this message
For Number of DCI + HI PDUs		
PDU Type	uint8_t	0: HI PDU, see Section 4.2.3.1. 1: DCI UL PDU, see Section 4.2.3.2. 2: EPDCCH DCI UL PDU, see Section 4.2.3.3. 3: MPDCCH DCI UL PDU. Added in Release 13, see Section 4.2.3.4. 4: NPDDCH DCI UL PDU. Added in Release 13, see Section 4.2.3.5
PDU Size	uint8_t	Size of the PDU control information (in bytes). This length value includes the 2 bytes required for the PDU type and PDU size parameters.
HI/DCI PDU Configuration	Struct	See Sections 4.2.3.1 to 4.2.3.5.

Table 4-109 HI DCI0 body

4.2.3.1 HI PDU

The format of a HI PDU is shown in Table 4-110 **Error! Reference source not found.**. The HI PDU contains the ACK/NACK response for a ULSCH transmission.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x201e	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-111 should be included.	V	TLV
0x201f	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-112 should be included.	V	TLV

Table 4-110 HI PDU

Field	Type	Description
Resource block start	uint8_t	This value is the starting resource block assigned to the ULSCH grant associated with this HI response. It should match the value sent in the DCI format 0 which allocated the ULSCH grant See [9] section 9.1.2 Value: 0 → 100
Cyclic Shift 2 for DMRS	uint8_t	This value is the 2nd cyclic shift for DMRS assigned to the ULSCH grant associated with this HI response. It should match the value sent in the DCI format 0 which allocated the ULSCH grant See [9] section 9.1.2

Field	Type	Description
		Value: 0 → 7
HI value	uint8_t	The PHICH value which is sent on the resource. 0: HI_NACK 1: HI_ACK
I_PHICH	uint8_t	Is used in the calculation of the PHICH location. For TDD only. See [9] section 9.1.2 1 = TDD subframe configuration 0 is used and the ULSCH grant associated with this HI was received in subframe 4 or 9 0 = in all other cases
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.

Table 4-111 HI PDU Release 8 parameters

Field	Type	Description
Flag TB2	uint8_t	Indicates is HI is present for a second transport block 0 = HI is not present 1 = HI is present
HI Value 2	uint8_t	The PHICH value for a second transport block. 0: HI_NACK 1: HI_ACK

Table 4-112 HI PDU Release 10 parameters

4.2.3.2 DCI UL PDU

The format of a DCI UL PDU is shown in Table 4-113. The DCI UL PDU contains the information which the L2/L3 software must provide the PHY so it can create the DCI format 0 or format 3/3A described in [12] section 5.3.3.1.

DCI format 4 was introduced in Release 10.
DCI format 5 was introduced in Release 12.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2020	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-114 should be included.	V	TLV
0x2021	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-115 should be	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
			included.		
0x204d	Release 12 parameters	struct	If the PHY supports Release 12 the parameters described in Table 4-116 should be included.	V	TLV

Table 4-113 DCI UL PDU

Field	Type	Description
DCI format	uint8_t	Format of the DCI 0 = 0 1 = 3 2 = 3A 3 = 4 4 = 5
CCE index	uint8_t	CCE index used to send the DCI. Value: 0 → 88
Aggregation level	uint8_t	The aggregation level used Value: 1,2,4,8
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Valid for all DCI formats Value: 1 → 65535.
Resource block start	uint8_t	The starting resource block for this ULSCH allocation. Valid for DCI format 0. In Release 10 this is only valid when resource allocation type 0 is signalled. Value: 0 → 100
Number of resource blocks	uint8_t	The number of resource blocks allocated to this ULSCH grant. Valid for DCI format 0. In Release 10 this is only valid when resource allocation type 0 is signalled. Value: 0 → 100
MCS_1	uint8_t	The modulation and redundancy version for the first transport block. See [9] section 8.6. Valid for DCI formats 0,4 Value: 0 → 31
Cyclic Shift 2 for	uint8_t	The 2nd cyclic shift for DMRS assigned to the UE

Field	Type	Description
DMRS		in the ULSCH grant. Valid for DCI formats 0,4 Value: 0 → 7
Frequency hopping enabled flag	uint8_t	Indicates if hopping is being used. See [9] Section 8.4. Valid for DCI format 0,5 In Release 10 this is only valid when resource allocation type 0 is signalled. In Release 12 this is used for DCI format 5/SCI format 0. 0 = no hopping, 1= hopping enabled
Frequency hopping bits	uint8_t	The frequency hopping bits See [9] Section 8.4 Valid for DCI format 0. In Release 10 this is only valid when resource allocation type 0 is signalled. Value: 0 → 3
New Data indication_1	uint8_t	The new data indicator for the transport block. Valid for DCI formats 0,4
UE TX antenna selection	uint8_t	Indicates how the CRC is calculated on the PDCCH. See [12] section 5.3.2.2 Valid for DCI format 0 0 = Not configured; 1 = Configured and using UE port 0; 2 = Configured and using UE port 1.
TPC	uint8_t	Tx power control command for PUSCH. Valid for DCI formats 0,4 Value: 0,1,2,3 Tx power control command for PSSCH and PSCCH, according to [9] Section 14.1.1.5 and 14.2.1.3. Valid for DCI formats 5 Value: 0,1
CQI/CSI request	uint8_t	Aperiodic CQI request flag Valid for DCI formats 0,4 In Release 10 the size of this field became flexible and depends on the Size of CQI/CSI field in the Release 10 parameter structure. If this field is 1 bit 0 = Aperiodic CQI/CSI not requested 1 = Aperiodic CQI/CSI requested

Field	Type	Description
		Else if this field is 2 bits Value: 0,1,2,3 Else if this field is 3 bits Value: 0 -> 7
UL index	uint8_t	UL index. Valid for TDD mode only. Valid for DCI formats 0,4 Value: 0,1,2,3
DL assignment index	uint8_t	DL assignment index. Valid for TDD mode only. Valid for DCI formats 0,4 Value: 1,2,3,4
TPC bitmap	uint32_t	TPC commands for PUCCH and PUSCH Valid for DCI formats: 3,3A The encoding follows [12] section 5.3.3.1.6
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.

Table 4-114 DCI UL PDU Release 8 parameters

Field	Type	Description
Cross carrier scheduling flag	uint8_t	Indicates if cross carrier scheduling has been enabled for the UE receiving this DCI. Valid for DCI formats 0,4 0 = Carrier indicator field is not valid 1 = Carrier indicator field is valid
Carrier indicator	uint8_t	Serving Cell Index Valid for DCI formats 0,4 if the Cross-Carrier Scheduling flag is enabled Value: 0->7
Size of CQI/CSI field	uint8_t	Indicates the size of the CQI/CSI request field Valid for DCI format 0,4 0 = 1 bit 1 = 2 bits 2 = 3 bits
SRS flag	uint8_t	Indicates if the SRS request parameter is present. Valid for DCI format 0 0 = SRS request field is not present 1 = SRS request field is present
SRS request	uint8_t	SRS request

Field	Type	Description
		Valid for DCI formats 0,4 under the following conditions DCI format 0: If the SRS flag field indicates this parameter is valid 0 = SRS not requested 1= SRS requested DCI format 4: The encoding follows [9] section 8.2
Resource allocation flag	uint8_t	Indicates if the Resource Allocation Type parameter is valid. Valid for DCI format 0 0 = Resource Allocation Type field is not valid and resource allocation type 0 is used 1 = Resource Allocation Type field is valid
Resource allocation type	uint8_t	Resource allocation type/header. Valid for DCI formats 0,4 under the following conditions DCI format 0: If the Resource Allocation flag field indicates this parameter is valid. DCI format 4: Always valid 0=type 0 1=type 1
Resource block coding	uint32_t	The encoding for the resource blocks. It's coding is dependent on whether resource allocation type 0, 1 is in use and signalled for DCI formats 0,4,5 For DCI format 0 this field is valid when resource allocation type 1 is signalled. For DCI format 4 this field is always valid. For DCI format 5/SCI format 0 this field is always valid. See [9] section 8.1 for the encoding used for each allocation type.
MCS_2	uint8_t	The modulation and redundancy version for the second transport block See [9] section 8.6. Valid for DCI format 4 Value: 0 → 31
New data indication_2	uint8_t	The new data indicator for the second transport block Valid for DCI format 4
Number of antenna ports	uint8_t	Defines number of antenna ports for this ULSCH allocation Value: 0 = 1 antenna port

Field	Type	Description
		1 = 2 antenna ports 2 = 4 antenna ports
TPMI	uint8_t	The codebook index to be used for precoding Valid for DCI format 4 2 antenna_ports: 0 → 7 4 antenna_ports: 0 → 63
Total DCI length including padding	uint8_t	The total DCI length including padding bits
N_UL_RB	uint8_t	BW of serving cell for which the DCI was scheduled for. This is valid for the case of cross carrier scheduling, for the case of a self-scheduling (cross carrier scheduling is not valid or Carrier indicator has value '0', the BW is the "DL BW support" as configured in configuration phase (params) Uplink channel bandwidth in resource blocks. See [10] section 5.6. Value: 6,15, 25, 50, 75, 100

Table 4-115 DCI UL PDU Release 10 parameters

Field	Type	Description
PSCCH Resource	uint8_t	6-bits describing the resource blocks for transmitting PSCCH, see [9] section 14.2.1 Valid for DCI format 5
Time resource pattern	uint8_t	7-bits describing the time resource pattern index, see [9] section 14.1.1. Valid for DCI format 5/SCI format 0

Table 4-116 DCI UL PDU Release 12 parameters

4.2.3.3 EPDCCH DCI UL PDU

The format of an EPDCCH DCI UL PDU is shown in Table 4-117. The DCI UL PDU contains the information which the L2/L3 software must provide the PHY so it can create the DCI format 0 or format 3/3A described in [12] section 5.3.3.1.

DCI format 4 was introduced in Release 10.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2020	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-114 should be included.	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2021	Release 10 parameters	struct	If the PHY supports Release 10 the parameters described in Table 4-115 should be included.	V	TLV
0x2041	Release 11 EPDCCH parameters	struct	If the PHY support EPDCCH- Release 11, the parameters described in Table 4-38 should be included.	V	TLV

Table 4-117 EPDCCH DCI UL PDU

As indicated in Table 4-117 the EPDCCH PDU actually carries DCI structure of Release 8-10 with the addition of the EPDCCH parameters from Table 4-38.

4.2.3.4 MPDCCH DCI UL PDU

The format of a MPDCCH DCI UL PDU is shown in Table 4-118. The MPDCCH DCI UL PDU contains the information which the L2/L3 software must provide the PHY so it can create the DCI formats 6-0A and 6-0B described in [12] sections 5.3.3.1. Additionally, legacy DCI-3 and DCI-3A may be transmitted via MPDCCH.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x204e	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-119 should be included.	V	TLV

Table 4-118 MPDCCH DCI UL PDU

Field	Type	Description
MPDCCH Narrowband	uint8_t	Narrowband for MPDCCH Value: 0 → 15
Number of PRB pairs	uint8_t	Number of PRB-pairs constituting the MPDCCH-PRB-pair set Value: 2,4,6 (2+4) *See Note1 in MPDCCH DCI DL PDU section
Resource Block Assignment	uint8_t	Combinational index r as defined in [9] section 9.1.4.4 Value: 0 → 14 *See Note1 in MPDCCH DCI DL PDU section
MPDCCH transmission type	uint8_t	0 = Localized 1 = Distributed
Start symbol	uint8_t	Value: 1 → 4
ECCE index	uint8_t	CCE index used to send the DCI. Value: 0 → 22
Aggregation level	uint8_t	The aggregation level used

Field	Type	Description
		Value: 2,4,8,16,24
RNTI type	uint8_t	0 = temporary C-RNTI 4 = other
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Valid for all DCI formats Value: 1 → 65535.
CEMode	uint8_t	1 = CEModeA 2 = CEModeB
DMRS scrambling init	Uint16_t	The DMRS scrambling sequence initialization parameter $n_{\mathrm{ID},i}^{\mathrm{EPDCCH}}$ defined in [11] section 6.10.3A.1 Value: 0 → 503
Initial transmission SF (io)	Uint16_t	Absolute Sub-Frame of the initial transmission Value: 0 → 10239 Value: 0xFFFF current absolute SF
Transmission power	uint16_t	Offset to the reference signal power. Value: 0 → 10000, representing -6 dB to 4 dB in 0.001 dB steps.
DCI format	uint8_t	Format of the DCI 1 = 3 2 = 3A 4 = 6-0A 5 = 6-0B
Resource block start	uint8_t	The starting resource block for this ULSCH allocation. Valid for DCI format 6-0A, 6-0B. Value: 0 → 99
Number of resource blocks	uint8_t	The number of resource blocks allocated to this ULSCH grant. Valid for DCI format 6-0A, 6-0B. Value: 1 → 6
MCS	uint8_t	The modulation and coding scheme. See section 8.6. Valid for DCI formats 6-0A, 6-0B Value: 0 → 15

Field	Type	Description
PUSCH repetition levels	uint8_t	Valid for DCI formats: 6-0A, 6-0B Value: 1 → 4 for DCI 6-0A 1 → 8 for DCI 6-0B
Frequency hopping flag	uint8_t	Indicates if hopping is being used. Valid for DCI format 6-0A 0 = no hopping, 1= hopping enabled
New Data indication	uint8_t	The new data indicator for the transport block. Valid for DCI formats 6-0A, 6-0B Value: 0 → 1
HARQ process	uint8_t	HARQ process number Valid for DCI formats: 6-0A, 6-0B Value: 0 →7
Redundancy version	uint8_t	The redundancy version for the transport block. Valid for DCI formats: 6-0A Value: 0 → 3
TPC	uint8_t	Tx power control command for PUSCH. Valid for DCI formats 6-0A Value: 0,1,2,3
CSI request	uint8_t	Aperiodic CSI request flag Valid for DCI formats 6-0A 0 = Aperiodic CSI not requested 1 = Aperiodic CSI requested
UL index	uint8_t	UL index. Valid for TDD mode only. Valid for DCI formats 6-0A Value: 0,1,2,3
DAI presence flag	uint8_t	Indicates if DL assignment index field is present in the DCI Valid for DCI format 6-0A Value: 0: Not present, 1: Present
DL assignment index	uint8_t	DL assignment index. Valid for TDD mode only. Valid for DCI formats 6-0A

Field	Type	Description
		Value: 1,2,3,4
SRS request	uint8_t	Aperiodic SRS request Valid for DCI format 6-0A 0 = SRS not requested 1= SRS requested
DCI subframe repetition number	uint8_t	Indicates the number of MPDCCH repetitions (r1, r2, r3, r4) Value: 1 →4
TPC bitmap	uint32_t	TPC commands for PUCCH and PUSCH Valid for DCI formats: 3,3A
Total DCI length including padding	uint8_t	The total DCI length including padding bits
Number of TX Antenna ports	uint8_t	Number of TX physical antenna ports
For Number of TX Antenna ports		
Precoding value	uint16_t	Precoding element for physical antenna #i real 8 bits followed by imaginary 8 bits

Table 4-119 MPDCCH DCI UL PDU Release 13 parameters

4.2.3.5 NPDCCH DCI UL PDU

The format of a NPDCCH DCI UL PDU is shown in Table 4-120.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2062	Release 13 parameters	struct	The PHY shall support Release 13 N-PDCCH if an NB-IOT cell and the Table 4-121 parameters described in should be included.	V	TLV

Table 4-120 NPDCCH DCI UL PDU

Field	Type	Description
NCCE index	uint8_t	NCCE index used to send the DCI. Value: 0 → 1
Aggregation level	uint8_t	The aggregation level used

SMALL CELL FORUM
Solving the HetNet puzzle

Field	Type	Description
		Value: 1,2
Start symbol	uint8_t	Start symbol for the transmisison of the NPDCCH containing the DCI N0. the index l in the first slot in a subframe fulfils $l \geq l_{\mathrm{DataStart}}$ where $l_{\mathrm{DataStart}}$ is given by clause 16.6.1 of [9] Value: 0 → 4 (0 for guard-band and standalone operating modes)
RNTI	uint16_t	The RNTI used for identifying the UE when receiving the PDU Value: 1 → 65535.
Scrambling re-initialization batch index	Uint8_t	The current SF's offset from scrambling re-initialization (scrambling sequence is re-initialized at start of search space and every 4 NPDCCH transmissions) Value: 1-4
NRS antenna ports assumed by the UE	Uint8_t	The number of cell specific transmit antenna ports assumed by the UE. Refer to [11] section 10.2.5.5. Value:1,2
Subcarrier indication	uint8_t	Subcarrier indication field (I_{SC}, 6 bits) determines the allocated NPUSCH subcarriers either for 15 KHz or 3.75 KHz. Refer to [9] section 16.5.1.1 Value: 0→18 for 15 KHz (12,4,2,1 options for 1,3,6,12 SCs) 0 → 47 for 3.75 KHz (0-47 for 1 SC) 48→63 are reserved
Resource assignment	uint8_t	The resource assignment field (I_{RU}, 3 bits) determines the Transport block size of NPUSCH (TBS) of the NPDSCH transport block (together with the MCS index) Refer to [9] section 16.5.1.2. Value: 0 → 7
Scheduling delay	Unit8_t	The scheduling delay field (I_{Delay}, 2 bits) determines the scheduling delay (k_0) for NPUSCH. Refer to [9] section 16.5.1 (Table 16.5.1-1). Value: 0 → 3
MCS	uint8_t	The modulation and coding scheme field (4 bits)

Field	Type	Description
		for the NPUSCH transport block Refer to [9] section 16.5.1.2 Value: 0 → 10 for single tone NPUSCH 0 → 12 for multi tone NPUSCH
Redundancy version	uint8_t	The redundancy version (RV_{idx}) for the NPUSCH transport block Refer to [9] section 16.5.1.2 Value: 0 for RV=0 1 for RV=2
Repetition number	uint8_t	The repetition number field (I_{REP}, 3 bits) determines the number of repetitions (N_{REP}) for NPUSCH transport block. Refer to [9] section 16.5.1.2. Value: 0 → 7
New data indicator	uint8_t	The new data indicator for the NPUSCH transport block. Value: 0→1
DCI subframe repetition number	uint8_t	The DCI subframe repetition number field (2 bits) is used by the UE (together with max repetition number provided by higher layers) to determine the repetition number and allocation of NPDCCH. Refer to [9] section 16.6. Value: 0 → 3

Table 4-121 NPDCCH DCI UL PDU Release 13 parameters

4.2.4 SUBFRAME errors

The error codes returned in an `ERROR.indication` generated by the `DL_CONFIG.request` message are given in Table 4-122.

Error code	Description
MSG_INVALID_STATE	The `DL_CONFIG.request` was received when the PHY was in the IDLE or CONFIGURED state.
SFN_OUT_OF_SYNC	The `DL_CONFIG.request` was received with a different SFN/SF than the PHY expected. The PHY has followed the SFN/SF sync process, see Section 3.2.2.2.
MSG_BCH_MISSING	A BCH PDU was expected in the `DL_CONFIG.request` message for this subframe.
MSG_PDU_ERR	An error was received in `DL_CONFIG.request`. The sub-error code should be analyzed.

Table 4-122 Error codes for `ERROR.indication` generated by `DL_CONFIG.request`

The error codes returned in an `ERROR.indication` generate by the `UL_CONFIG.request` message are given in Table 4-123.

Error code	Description
MSG_INVALID_STATE	The `UL_CONFIG.request` was received when the PHY was in the IDLE or CONFIGURED state.
MSG_PDU_ERR	An error was received in `UL_CONFIG.request`. The sub-error code should be analyzed.

Table 4-123 Error codes for `ERROR.indication` generated by `UL_CONFIG.request`

The error codes returned in an `ERROR.indication` generate by the `HI_DCI0.request` message are given in Table 4-124.

Error code	Description
MSG_INVALID_STATE	The `HI_DCI0.request` was received when the PHY was in the IDLE or CONFIGURED state.
MSG_INVALID_SFN	The `HI_DCI0.request` message received in subframe N included a SFN/SF value which was not N-1. The message has been ignored.
MSG_HI_ERR	An error was received in `HI_DCI0.request`. The sub-error code should be analyzed

Table 4-124 Error codes for `ERROR.indication` generated by `HI_DCI0.request`

4.2.5 Downlink data

4.2.5.1 TX.request

The format of the `TX.request` message is described in Table 4-125. This message contains the MAC PDU data for transmission over the air interface. The PDUs described in this message must follow the same order as `DL_CONFIG.request`.

This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE, or CONFIGURED, state an `ERROR.indication` message will be sent by the PHY.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
N/A	TX request body (FAPI)	struct	See Table 4-126.	V	Not present
0x2022	TX request body (nFAPI)	struct	See Table 4-128.	Not present	TLV

Table 4-125 `TX.request` message body

Field	Type	Description
Number of PDUs	uint16_t	Number of PDUs included in this message.
For each PDU		
PDU length	uint16_t	The total length (in bytes) of the PDU description and PDU data, without the padding bytes.
PDU index	uint16_t	This is a count value which starts from 0. It is incremented for each BCH, MCH, PCH or DLSCH PDU. This value was included in DL_CONFIG.request and associates the data to the control information. It is reset to 0 for every subframe Range 0 → 65535
numTLV	uint32_t	The number of TLVs describing the data of the transport block.
TLVs	TLV[numTLV]	Always a multiple of 32-bits. See Table 4-127

Table 4-126 TX request body (FAPI)

Field	Type	Description
tag	uint16_t	Range 0 -> 1 0: Payload is carried directly in the value field 1: Pointer to payload is in the value field
length	uint16_t	Length of the actual payload in bytes, without the padding bytes
value	Variable or uint32_t	Always a multiple of 32-bits. Tag=0: Only the most significant bytes of the size indicated by 'length' field are valid. Remaining bytes are zero padded to the nearest 32-bit boundary Tag=1: Pointer to the payload. Occupies 32-bits

Table 4-127 TLV structure

Field	Type	Description
Number of PDUs	uint16_t	Number of PDUs included in this message.
For each PDU		
PDU length	uint16_t	The total length (in bytes) of the PDU description and PDU data.
PDU index	uint16_t	This is a count value which starts from 0. It is incremented for each BCH, MCH, PCH or DLSCH PDU. This value was included in DL_CONFIG.request and associates the data to the control information. It is reset to 0 for every subframe

Field	Type	Description
		Range 0 → 65535
PDU#N	Variable	Contents of PDU#N. This will be a MAC PDU.

Table 4-128 TX request body (nFAPI)

4.2.5.2 Downlink data errors

The error codes returned in an `ERROR.indication` generate by the `TX.request` message are given in Table 4-129.

Error code	Description
MSG_INVALID_STATE	The `TX.request` was received when the PHY was in the IDLE or CONFIGURED state.
MSG_INVALID_SFN	The `TX.request` message received in subframe N included a SFN/SF value which was not N. The message has been ignored.
MSG_TX_ERR	An error was received in `TX.request`. The sub-error code should be analyzed.

Table 4-129 Error codes for `ERROR.indication`

4.2.6 Uplink data

4.2.6.1 RX_ULSCH.indication

The format of the `RX_ULSCH.indication` message is shown in Table 4-130.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2023	RX indication body	struct	See Table 4-131.	V	TLV

Table 4-130 `RX.indication` message body

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of PDUs	uint16_t	Number of PDUs included in this message.	V	V
	For (Number of PDUs) {				
0x2038	RX UE Information	struct	Described in Table 4-167. Handle was specified in ULSCH PDU.	V	TLV
0x2024	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-132	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
			should be included.		
0x2025	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-133 will be included.	V	TLV
	}				
N/A	PDU#1	Variable	Contents of PDU#1. This will be a MAC PDU.	V	V
N/A	PDU#2	Variable	Contents of PDU#2. This will be a MAC PDU	V	V
	..	...	...		
N/A	PDU#n	Variable	Contents of PDU#n. This will be a MAC PDU	V	V

Table 4-131 RX indication body

Field	Type	Description
Length	uint16_t	Length of PDU in bytes.
Data offset	uint16_t	Gives the PDU#i data address offset from the beginning of the 'Number of PDUs' field. An offset of 0 indicates a CRC or decoding error.
UL_CQI	uint8_t	SNR Value: 0-255, representing -64dB to 63.5dB, with 0.5dB step size.
Timing advance	uint16_t	The timing advance measured for this PDU and UE. Value: T_A from 0 to 63 as defined in [9] section 4.2.3.

Table 4-132 RX Indication Release 8 parameters

Field	Type	Description
Timing advance R9	uint16_t	Timing advance used for positioning. See [15] section 5.2.4 and [16] section 10.3.1. Value: 0 → 2047 (measured in steps of 2Ts) 2048 → 7690 (measured in steps of 8Ts)

Table 4-133 RX Indication Release 9 parameters

4.2.6.2 HARQ.indication

The format of the uplink HARQ control from the UE is dependent on whether a TDD or FDD PHY is used. To accommodate this difference the `HARQ.indication` messages contains either TDD information or FDD information. The format of the `HARQ.indication` message is given in Table 4-134.

The `HARQ.indication` messages provide the following results for each ACK/NACK report.

- ACK – The PHY confidently detected an ACK
- NACK – The PHY confidently detected an NACK
- DTX – The PHY confidently detected that the UE did not transmit an ACK/NACK response
- ACK or NACK – The PHY is unsure whether it detected an ACK or NACK.
- ACK or DTX - The PHY is unsure whether it detected an ACK or DTX
- NACK or DTX - The PHY is unsure whether it detected an NACK or DTX
- ACK or NACK or DTX - The PHY is unsure whether it detected an ACK or NACK or DTX

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2026	HARQ indication body	struct	See Table 4-135.	V	TLV

Table 4-134 **`HARQ.indication` message body**

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of HARQs	uint16_t	Number of HARQs included in this message.	V	V
	For (Number of HARQs) {				
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the HARQ instance	Not Present	V
0x2038	RX UE Information	struct	Described in Table 4-167. ULSCH PDU or UCI PDU.	V	TLV
0x2027	Release 8 TDD parameters	struct	If the PHY only supports Release 8 and is configured for TDD mode the parameters described in Table 4-136 should be included.*	V	TLV
0x2028	Release 9 or later TDD parameters	struct	If the PHY supports Release 9,10,12 or 12 or later and is configured for TDD mode the parameters described in Table 4-137 should be included.*	V	TLV
0x204f	Release 13 or later TDD parameters	struct	If the PHY supports Release 13 or later and is configured for TDD mode the parameters described in Table 4-138 should be included.*	V	TLV
0x2029	Release 8 FDD	struct	If the PHY only supports	V	TLV

SMALL CELL FORUM
Solving the HetNet puzzle

Tag Value	Field	Type	Description	FAPI	nFAPI
	parameters		Release 8 and is configured for FDD mode the parameters described in Table 4-143 should be included.*		
0x202a	Release 9 or later FDD parameters	struct	If the PHY supports Release 9,10,12 or 12 or later and is configured for FDD mode the parameters described in Table 4-144 should be included.*	V	TLV
0x2050	Release 13 or later FDD parameters	struct	If the PHY supports Release 13 or later and is configured for FDD mode the parameters described in Table 4-145 should be included.*	V	TLV
0x2052	UL_CQI Information	struct	Described in Table 4-168	V	TLV
	}				

Table 4-135 HARQ indication body

*Only FDD or TDD parameters are included. In addition one only of Rel8, Rel9 or Rel13 parameters are included.

TDD format

The TDD PHY Release 8 parameters are given in Table 4-136, for TDD PHY Release 9, 10, 11 or 12 parameters are given in Table 4-137, for TDD PHY Release 13 or later parameters are given in Table 4-138.

Field	Type	Description
Mode	uint8_t	The format of the ACK/NACK response expected. The bundling and multiplexing options are passed to the PHY in an uplink subframe configuration PDU. If Format 1a/1b or Channel Selection ACK/NACK is combined with either CQI or SR information then a special ACK/NACK encoding is used which reports the number of ACKs, rather than providing specific ACK/NACK values. This is identified separately and called SPECIAL_BUNDLING in this API. (see [9] section 7.3 for more information) 0 = Format 1a/1b BUNDLING 1 = Format 1a/1b MULTIPLEXING 2 = Format 1a/1b SPECIAL BUNDLING 3 = Channel Selection 4 = Format 3
Number of ACK/NACK	uint8_t	The number of ACK/NACK results reported for this UE. See [9] section 10. Value: 1 → 4
HARQ data	struct	The format of the data is dependent on the

Field	Type	Description
		HARQ mode; BUNDLING – Table 4-139 MULTIPLEXING – Table 4-140 SPECIAL BUNDLING - Table 4-141

Table 4-136 HARQ information for TDD Release 8

Field	Type	Description
Mode	uint8_t	The format of the ACK/NACK response expected. The bundling and multiplexing options are passed to the PHY in an uplink subframe configuration PDU. If Format 1a/1b or Channel Selection ACK/NACK is combined with either CQI or SR information then a special ACK/NACK encoding is used which reports the number of ACKs, rather than providing specific ACK/NACK values. This is identified separately and called SPECIAL_BUNDLING in this API. (see [9] section 7.3 for more information) 0 = Format 1a/1b BUNDLING 1 = Format 1a/1b MULTIPLEXING 2 = Format 1a/1b SPECIAL BUNDLING 3 = Channel Selection 4 = Format 3
Number of ACK/NACK	uint8_t	The number of ACK/NACK results reported for this UE. Value 1..4 for Format 1a/1b BUNDLING Value 1..4 for Format 1a/1b MULTIPLEXING Value 1..8 for Channel Selection Value 1..21 for Format 3
for Number of ACK/NACK		
HARQ data	struct	The format of the data is dependent on the HARQ mode; Format 1a/1b BUNDLING – Table 4-142 Format 1a/1b MULTIPLEXING – Table 4-142 Format 1a/1b SPECIAL BUNDLING - Table 4-141 Channel Selection – Table 4-142 Format 3 – Table 4-142

Table 4-137 HARQ information for TDD Release 9, 10, 11 and 12

Field	Type	Description
Mode	uint8_t	The format of the ACK/NACK response expected. The bundling and multiplexing options are passed to the PHY in an uplink subframe configuration PDU. If Format 1a/1b or Channel Selection ACK/NACK is combined with either CQI or SR information then a special ACK/NACK encoding is used which reports the number of ACKs, rather than providing specific ACK/NACK values. This is

Field	Type	Description
		identified separately and called SPECIAL_BUNDLING in this API. (see [9] section 7.3 for more information) 0 = Format 1a/1b BUNDLING 1 = Format 1a/1b MULTIPLEXING 2 = Format 1a/1b SPECIAL BUNDLING 3 = Channel Selection 4 = Format 3 5 = Format 4 6 = Format 5
Number of ACK/NACK	uint16_t	The number of ACK/NACK results reported for this UE. Value 1..4 for Format 1a/1b BUNDLING Value 1..4 for Format 1a/1b MULTIPLEXING Value 1..8 for Channel Selection Value 1..21 for Format 3 Value ≥ 22 for Format 4 Value ≥ 22 for Format 5
for Number of ACK/NACK		
HARQ data	struct	The format of the data is dependent on the HARQ mode; Format 1a/1b BUNDLING – Table 4-142 Format 1a/1b MULTIPLEXING – Table 4-142 Format 1a/1b SPECIAL BUNDLING - Table 4-141 Channel Selection – Table 4-142 Format 3 – Table 4-142 Format 4 – Table 4-142 Format 5 – Table 4-142

Table 4-138 HARQ information for TDD Release 13 and later

Field	Type	Description
Value 0	uint8_t	Indicates HARQ results. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX
Value 1	uint8_t	Indicates HARQ results. Range 1 → 7, see above.

Table 4-139 TDD HARQ data format for mode = BUNDLING

Field	Type	Description
Value 0	uint8_t	Indicates HARQ results. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX
Value 1	uint8_t	Indicates HARQ results. Range 1 → 7, see above.
Value 2	uint8_t	Indicates HARQ results. Range 1 → 7, see above.
Value 3	uint8_t	Indicates HARQ results. Range 1 → 7, see above.

Table 4-140 TDD HARQ data format for mode = MULTIPLEXING

Field	Type	Description
Value 0	uint8_t	Number of ACK among multiple ACK/NACK responses, see [9] table 7.3.-1 0 = 0 or None (UE detect at least one DL assignment is missed) 1 = 1 or 4 or 7 ACKs reported 2 = 2 or 5 or 8 ACKs reported 3 = 3 or 6 or 9 ACKs reported 4 = DTX (UE did not transmit anything)

Table 4-141 TDD HARQ data format for mode = SPECIAL BUNDLING

Field	Type	Description
Value 0	uint8_t	Indicates HARQ results. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX

Table 4-142 TDD HARQ data format for Release 9 and later

FDD format

The FDD PHY Release 8 parameters are given in Table 4-143, for FDD PHY Release 9, 10, 11 or 12 or later parameters are given in Table 4-144, for FDD PHY Release 13 or later parameters are given in Table 4-145.

Field	Type	Description
HARQ TB1	uint8_t	HARQ feedback of 1st TB.

Field	Type	Description
		Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX
HARQ TB2	uint8_t	HARQ feedback of 2nd TB. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX

Table 4-143 HARQ information for FDD Release 8

Field	Type	Description
Mode	uint8_t	The format of the ACK/NACK response expected. 0 = Format 1a/1b 1 = Channel Selection, added in Release 10 2 = Format 3, added in Release 10
Number of ACK/NACK	uint8_t	The number of ACK/NACK results reported for this UE. Value 1..2 for Format 1a/1b Value 1..4 for Channel Selection Value 1..10 for Format 3
for Number of ACK_NACK results		
HARQ TB_N	uint8_t	HARQ feedback of Nth TB. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX

Table 4-144 HARQ information for FDD Release 9, 10, 11 and 12

Field	Type	Description
Mode	uint8_t	The format of the ACK/NACK response expected. 0 = Format 1a/1b 1 = Channel Selection, added in Release 10 2 = Format 3, added in Release 10 3 = Format 4, added in Release 13 4 = Format 5, added in Release 13
Number of ACK/NACK	uint16_t	The number of ACK/NACK results reported for this UE. Value 1..2 for Format 1a/1b Value 1..4 for Channel Selection Value 1..10 for Format 3 Value ≥ 22 for Format 4 Value ≥ 22 for Format 5
for Number of ACK_NACK results		
HARQ TB_N	uint8_t	HARQ feedback of N^{th} TB. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX

Table 4-145 HARQ information for FDD Release 13 and later

4.2.6.3 CRC.indication

The format of the `CRC.indication` message is given in Table 4-146.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x202b	CRC indication body	struct	See Table 4-147	V	TLV

Table 4-146 `CRC.indication` message body

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of CRCs	uint16_t	Number of CRCs included in this message.	V	V
	For (Number of CRCs) {				

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the CRC instance	Not Present	V
0x2038	RX UE Information	struct	Described in Table 4-167. Handle was specified in ULSCH PDU.	V	TLV
0x202c	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-148 should be included.	V	TLV
	}				

Table 4-147 CRC indication body

Field	Type	Description
CRC Flag	uint8_t	A flag indicating if a CRC error was detected. 0: CRC_CORRECT 1:CRC_ERROR

Table 4-148 CRC indication Release 8 parameters

4.2.6.4 RX_SR.indication

The format of the RX_SR.`indication` message is given in Table 4-149.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x202d	RX SR indication body	struct	See Table 4-150.	V	TLV

Table 4-149 `RX_SR.indication` message body

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of SRs	uint16_t	Number of SRs included in this message.	V	V
	For (Number of SRs) {				
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the SR instance	Not Present	V
0x2038	RX UE Information	struct	Described in Table 4-167. Handle was specified in an UCI PDU.	V	TLV
0x2052	UL_CQI	struct	Described in Table 4-168	V	TLV

Tag Value	Field	Type	Description	FAPI	nFAPI
	Information				
	}				

Table 4-150 RX SR indication body

4.2.6.5 RX_CQI.indication

The format of the `RX_CQI.indication` message is given in Table 4-151. The format of DL CQI feedback and reports varies depending upon the channel used for feedback (PUSCH or PUCCH) and the DL transmission mode. This is detailed in [12]. The formats differ in the fields reported and the resultant number of bits required for the report. .

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x202e	RX CQI indication body	struct	See Table 4-152.	V	TLV

Table 4-151 `RX_CQI.indication` message body

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of PDUs	uint16_t	Number of PDUs included in this message.	V	V
	For (Number of PDUs) {				
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the CQI instance	Not Present	V
0x2038	RX UE Information	struct	Described in Table 4-167. Handle was specified in ULSCH PDU or UCI PDU.	V	TLV
0x202f	Release 8 parameters	struct	If the PHY only supports Release 8 the parameters described in Table 4-153 should be included.*	V	TLV
0x2030	Release 9 parameters	struct	If the PHY supports Release 9 or later the parameters described in Table 4-154 will be included.*	V	TLV
0x2052	UL_CQI Information	struct	Described in Table 4-168	V	TLV
	}				
N/A	PDU#1	Variable	Contents of PDU#1. Raw	V	V

Tag Value	Field	Type	Description	FAPI	nFAPI
			format CQI report as defined in [12]. LSB of the CQI report is bit [0] of byte 0.		
N/A	PDU#2	Variable	Contents of PDU#2. Raw format CQI report as defined in [12]. LSB of the CQI report is bit [0] of byte 0.	V	V
	..	…	…		
N/A	PDU#n	Variable	Contents of PDU#n. Raw format CQI report as defined in [12]. LSB of the CQI report is bit [0] of byte 0.	V	V

Table 4-152 RX CQI indication body

*Only Rel8 or Rel9 parameters are included, not both.

Field	Type	Description
Length	uint16_t	Length of PDU in bytes.
Data Offset	uint16_t	Gives the PDU#i data address offset from the beginning of the 'Number of PDUs' field. An offset of 0 indicates a CRC or decoding error.
UL_CQI	uint8_t	This value is deprecated.
RI	uint8_t	The rank indication reported by the UE on PUSCH for aperiodic CSI. Value: 0..8 0 = RI not received 1..4 = RI value
Timing Advance	uint16_t	The timing advance measured for this PDU and UE. Value: T_A from 0 to 63 as defined in [9] section 4.2.3.

Table 4-153 CQI and RI information for Release 8

Field		Type	Description
Length		uint16_t	Length of PDU in bytes.
Data Offset		uint16_t	Gives the PDU#i data address offset from the beginning of the 'Number of PDUs' field. An offset of 0 indicates a CRC or decoding error.
UL_CQI		uint8_t	This value is deprecated.
Number of CC reported		uint8_t	Value: 1..5
For each CC			
	RI	uint8_t	The rank indication reported by the UE on PUSCH for aperiodic CSI.

Field		Type	Description
			Value: 0..8 0 = RI not received 1.. 8 = RI value in Release 10
Timing Advance		uint16_t	The timing advance measured for this PDU and UE. Value: T_A from 0 to 63 as defined in [9] section 4.2.3.
Timing Advance R9		uint16_t	Timing advance used for positioning. See [15] section 5.2.4 and [16] section 10.3.1. Value: 0 → 2047 (measured in steps of 2Ts) 2048 → 7690 (measured in steps of 8Ts)

Table 4-154 CQI and RI information for Release 9 and 10

4.2.6.6 RACH.indication

The format of the `RACH.indication` message is given in Table 4-155.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2031	RACH indication body	struct	See Table 4-156.	V	TLV

Table 4-155 `RACH.indication` message body

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of Preambles	uint8_t	Number of RACH preambles	V	V
	For (Number of Preambles) {				
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the Preamble instance	Not Present	V
0x2032	Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-157 should be included.	V	TLV
0x2033	Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-158 will be included.	V	TLV
0x2051	Release 13 parameters	struct	If the PHY supports Release 13 the parameters described in Table 4-159 will be included.	V	TLV
	}				

Table 4-156 RACH indication body

Field	Type	Description
RNTI	uint16_t	The RA-RNTI value See [5] section 7.1 Value: 1 → 65535.
Preamble	uint8_t	The detected preamble Value: 0 → 63
Timing Advance	uint16_t	The measured timing advance for the preamble. Value: 0 → 1282

Table 4-157 `RACH.indication` Release 8 parameters

Field	Type	Description
Timing Advance R9	uint16_t	Timing advance used for positioning. See [15] section 5.2.4 and [16] section 10.3.1. Value: 0 → 2047 (measured in steps of 2Ts) 2048 → 7690 (measured in steps of 8Ts)

Table 4-158 `RACH.indication` Release 9 parameters

Field	Type	Description
RACH resource type	uint8_t	Indicates if this indication is related to Cat-M UE and in which CE level Value: 0 – non LC/CE RACH 1 – LC/CE RACH CE level 0 2 – LC/CE RACH CE level 1 3 – LC/CE RACH CE level 2 4 – LC/CE RACH CE level 3

Table 4-159 `RACH.indication` Release 13 parameters

4.2.6.7 SRS.indication

The format of the `SRS.indication` message is given in Table 4-160.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2034	SRS indication body	struct	See Table 4-161.	V	TLV

Table 4-160 `SRS.indication` message body

Tag Value	Field		Type	Description	FAPI	nFAPI
N/A	Number of UEs		uint8_t	Number of UEs contributing to the uplink SRS	V	V
	For (Number of UEs) {					
N/A		Instance Length	uint16_t	The Length in bytes of all TLVs within the UE instance	Not Present	V
0x2038		RX UE Information	struct	Described in Table 4-167. Handle was specified in SRS PDU.	V	TLV
0x2035		Release 8 parameters	struct	The PHY shall support Release 8 and the parameters described in Table 4-162 should be included.	V	TLV
0x2036		Release 9 parameters	struct	If the PHY supports Release 9 the parameters described in Table 4-163 will be included.	V	TLV
0x2037		Release 10 TDD parameters	struct	If the PHY is TDD and supports Release 10 the parameters described in Table 4-164 will be included. For FDD no parameters are included.	V	TLV
0x2054		TDD Channel Measurement	struct	If the PHY is TDD the parameters described in Table 4-165 will be included. For FDD no parameters are included.	V	TLV
0x2053		Release 11 parameters	struct	If the PHY supports Release 11 the parameters described in Table 4-166 will be included.	V	TLV
	}					

Table 4-161 SRS indication body

Field		Type	Description
Doppler estimation		uint16_t	FFS. Values: 0 → 255,
Timing Advance		uint16_t	The timing advance measured for the UE. Value: T_A from 0 to 63 as defined in [9] section 4.2.3.
Number of resource blocks		uint8_t	Number of resource blocks to be reported for this UE
RB start		uint8_t	The starting point of the RBs to be reported.
For (Number of RBs) {			
	SNR	uint8_t	Field size dependent on configured bandwidth SNR for RBs, each RBs report one SNR. Value: 0-255, representing -64dB to 63.5dB, with

Field		Type	Description
			0.5dB step size.’
}			

Table 4-162 `SRS.indication` **Release 8 parameters**

Field	Type	Description
Timing Advance R9	uint16_t	Timing advance used for positioning. See [15] section 5.2.4 and [16] section 10.3.1. Value: 0 → 2047 (measured in steps of 2Ts) 2048 → 7690 (measured in steps of 8Ts)

Table 4-163 `SRS.indication` **Release 9 parameters**

Field	Type	Description
UpPTS Symbol	uint8_t	Indicates symbol where SRS was received. Only valid if the SRS was received in subframe 1 or 6. Value: 0 = symbol 0 1 = symbol 1

Table 4-164 `SRS.indication` **Release 10 parameters for TDD**

Field			Type	Description
numPRBperSubband			uint8_t	Number of PRBs that are treated as one subband
Number of subbands			uint8_t	Defines the number of subbands used for channel feedback. Value 0 -> 13
numAntennas			unit8_t	Number of physical antennas
For (Number of subbands) {				
	subbandIndex		unit8_t	Index of subband for which the following channel coefficient is applied
	For (Number of physical antennas) {			
		Channel	unit16_t	Averaged channel coefficient in a subband for physical antenna #i, real 8 bits followed by imaginary 8 bits
	}			
}				

Table 4-165 SRS.indication channel measurement for TDD

Field	Type	Description

Field	Type	Description
UL_RTOA	uint16_t	UL relative time of arrival used for network based positioning. See [15] section 5.2.8 and [22] section 8. Value: 0 → 4800 (measured in steps of 2Ts)

Table 4-166 `SRS.indication` **Release 11 parameters**

4.2.6.8 RX UE Information

The format of the RX UE Information is given in Table 4-167.

Field	Type	Description
Handle	uint32_t	The handle received in the UL_CONFIG.request.
RNTI	uint16_t	The RNTI passed to the PHY in a UL_CONFIG.request See [5] section 7.1 Value: 1 → 65535.

Table 4-167 RX UE Information

4.2.6.9 UL_CQI Information

The format of the UL_CQI Information is given in Table 4-168.

Field	Type	Description
UL_CQI	uint8_t	SNR Value: 0-255, representing -64dB to 63.5dB, with 0.5dB step size.
Channel	uint8_t	The channel to which this measurement refers 0 = PUCCH 1 = PUSCH

Table 4-168 UL_CQI Information

4.2.6.10 NB_HARQ.indication

The format of the `NB_HARQ.indication` message is given in Table 4-169.

NB-IOT was added in Release 13 and for Release 13 a maximum of one HARQ is returned per subframe,

The `NB_HARQ.indication` messages provide the following results for each ACK/NACK report.

- ACK – The PHY confidently detected an ACK
- NACK – The PHY confidently detected an NACK
- DTX – The PHY confidently detected that the UE did not transmit an ACK/NACK response
- ACK or NACK – The PHY is unsure whether it detected an ACK or NACK.
- ACK or DTX - The PHY is unsure whether it detected an ACK or DTX
- NACK or DTX - The PHY is unsure whether it detected an NACK or DTX

- ACK or NACK or DTX - The PHY is unsure whether it detected an ACK or NACK or DTX

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2063	NB_HARQ indication body	struct	See Table 4-170.	V	TLV

Table 4-169 `NB_HARQ.indication` **message body**

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of HARQs	uint16_t	Number of HARQs included in this message.	Not present	V
	For (Number of HARQs) {				
N/A	Instance Length	uint16_t	The Length in bytes of all TLVs within the HARQ instance	Not Present	V
0x2038	RX UE Information	struct	Described in Table 4-167. NULSCH format 2 PDU	V	TLV
0x2064	Release 13 NB-IOT parameters	struct	If the PHY supports Release 13 or later and is configured for FDD mode the parameters described in Table 4-171 should be included.*	V	TLV
0x2052	UL_CQI Information	struct	Described in Table 4-168	V	TLV

Table 4-170 NB_HARQ indication body

Field	Type	Description
HARQ TB1	uint8_t	HARQ feedback of 1st TB. Range 1 → 7 1 = ACK 2 = NACK 3 = ACK or NACK 4 = DTX 5 = ACK or DTX 6 = NACK or DTX 7 = ACK or NACK or DTX

Table 4-171 NB_HARQ information for Release 13

4.2.6.11 NRACH.indication

The format of the `NRACH.indication` message is given in Table 4-172.

NB-IOT was added in Release 13.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	The SFN/SF of the SUBFRAME this information was received in. A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
0x2065	NRACH indication body	struct	See Table 4-173.	V	TLV

Table 4-172 `NRACH.indication` **message body**

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	Number of initial SCs detected	uint8_t	Number of NRACH with initial SCs	V	V
	For (Number of initial SCs) {				
0x2066	Release 13 parameters	struct	If the PHY supports NB-IOT Release 13 the parameters described in Table 4-174 will be included.	V	TLV
	}				

Table 4-173 **NRACH indication body**

Field	Type	Description
RNTI	uint16_t	The RA-RNTI value See [5] section 7.1 Value: 1 → 65535.
Initial SC	uint8_t	The detected initial SC Value: 0 → 47
Timing Advance	uint16_t	The measured timing advance for the preamble. Value: 0 → 3840 (steps of 2Ts indicating max of 250usec)
NRACH CE level	uint8_t	Indicates the NRACH CE level as configured in CONFIG Value: 0 – CE level 0 1 – CE level 1 2 – CE level 2

Table 4-174 `NRACH.indication` **Release 13 parameters**

4.2.7 LBT_DL_CONFIG.request

The format of the `LBT_DL_CONFIG.request` message is given in Table 4-175. An `LBT_DL_CONFIG.request` message indicates the SFN/SF subframe the LBT process starts in. This control information is for an LAA downlink subframe.

This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE or CONFIGURED state an `ERROR.indication` message will be sent by the PHY.

The following combinations of PDUs are required:

- An LBT_PDSCH_REQ PDU
- An LBT_DRS_REQ PDU

The PDUs included in this structure have no ordering requirements.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
N/A	Length	uint16_t	The length of the downlink subframe configuration. Range 0 → 65535.	V	Not present
0x2055	LBT_DL Config request body	struct	See Table 4-176.	V	TLV

Table 4-175 `LBT_DL_CONFIG.request` **message body**

Field	Type	Description
Number of PDUs	Uint16_t	Number of PDUs that are included in this message. Range 1 → 2
For number of PDUs		
PDU Type	uint8_t	0: LBT_PDSCH_REQ PDU 1: LBT_DRS_REQ PDU
PDU Size	PDU Size	uint8_t
DL PDU Configuration	PDU	struct

Table 4-176 LBT_DL config request body

4.2.7.1 LBT_PDSCH_REQ PDU

The format of a LBT_PDSCH PDU is shown in Table 4-177.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2056	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-178 should be included.	V	TLV

Table 4-177 LBT_PDSCH_REQ PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in LBT_PDSCH_RSP PDU in LBT_DL.indication
n_{CCA}	uint32_t	Indicates the value of the defer factor (M_P in [9] section 15.1.1) in slots.
N_{CCA}	uint32_t	Indicates the value of LBT backoff counter (N in [9] section 15.1.1) in slots.
Offset	uint32_t	Indicates the LBT start time in microseconds from the beginning of the subframe scheduled by this message. Values: 0-999
LTE TXOP SF	uint32_t	Indicates the LTE TXOP ($T_{MCOT,P}$ in [9] section 15.1.1) duration in subframes. Note: L1 may adjust the TXOP according "SFN/SF End"
TXOP SFN/SF End	Uint16_t	Indicates the SFN/SF by which the TXOP must end. In worst case, this would be the last TXOP subframe. A 16-bit value where: [15:4] SFN, range 0 → 1023, [3:0] SF, range 0 → 9 Value of all "1s" indicate no TXOP SFN/SF End
LBT mode	uint32_t	Part of multi-carrier support. Indicates whether full LBT process is carried or partial LBT process is carried (multi carrier mode B according to [9] section 15.1.5.2) 0 – full LBT (regular LBT) 1 – Partial LBT (looking back 25usec prior to transmission as indicated in [9] section 15.1.5.2)

Table 4-178 LBT_PDSCH_REQ PDU Release 13 parameters

4.2.7.2 LBT_DRS_REQ PDU

The format of a LBT_DRS PDU is shown in Table 4-179.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2057	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-180 should be included.	V	TLV

Table 4-179 LBT_DRS_REQ PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in LBT_PDSCH_RSP PDU in LBT_DL.indication
Offset	uint32_t	Indicates the LBT start time in microseconds from the beginning of the subframe scheduled by this message. Values: 0-999
SFN/SF End	Uint16_t	Indicates the SFN/SF by which the DRS window (Discovery signal occasion as described in [9] section 6.11A) must end. In worst case, this would be the last TXOP subframe. A 16-bit value where: [15:4] SFN, range 0 → 1023,

Field	Type	Description
		[3:0] SF, range 0 → 9 Value of all "1s" indicate no TXOP SFN/SF End
LBT mode	uint32_t	Part of multi-carrier support. Indicates whether full LBT process is carried or partial LBT process is carried (multi carrier mode B according to [9] section 15.1.5.2) 0 – full LBT (regular LBT) 1 – Partial LBT (looking back 25usec prior to transmission as indicated in [9] section 15.1.5.2)

Table 4-180 LBT_DRS_REQ PDU Release 13 parameters

4.2.8 LBT_DL.indication

The format of the `LBT_DL.indication` message is given in Table 4-181. An `LBT_DL.indication` message, indicating the SFN/SF subframe for the respective LAA subframe in which the LBT acquisition attempt was made, is provided. There could be several LBT_DL.indication messages per a single LBT_DL_Config.request, sent the LBT operation has ended (either succesfully or until LBT operation was expired).

This message can be sent by the L2/L3 when the PHY is in the RUNNING state. If it is sent when the PHY is in the IDLE or CONFIGURED state an `ERROR.indication` message will be sent by the PHY.

The following combinations of PDUs are required:

- An LBT_PDSCH_RSP PDU (indicating response for LBT_PDSCH_REQ PDU)
- An LBT_DRS_RSP PDU (indicating response for LBT_DRS_REQ PDU)

The PDUs included in this structure have no ordering requirements.

Tag Value	Field	Type	Description	FAPI	nFAPI
N/A	SFN/SF	uint16_t	A 16-bit value of upcoming SFN/SF to which LBT result applies where, [15:4] SFN, range 0 → 1023 [3:0] SF, range 0 → 9	V	V
N/A	Length	uint16_t	The length of the downlink subframe configuration. Range 0 → 65535.	V	Not present
0x2058	LBT_DL indication message body	struct	See Table 4-182.	V	TLV

Table 4-181 LBT_DL.indication message body

Field	Type	Description
Number of PDUs	Uint16_t	Number of PDUs that are included in this message. Range 1 → 2
For number of PDUs		

Field	Type	Description
PDU Type	uint8_t	0: LBT_PDSCH_RSP PDU 1: LBT_DRS_RSP PDU
PDU Size	PDU Size	uint8_t
DL PDU Configuration	PDU	struct

Table 4-182 LBT_DL indication body

4.2.8.1 LBT_PDSCH_RSP PDU

The format of a LBT_PDSCH PDU is shown in Table 4-183.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x2059	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-184 should be included.	V	TLV

Table 4-183 LBT_PDSCH_RSP PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in LBT_PDSCH_RSP PDU in LBT_DL.indication
result	uint32_t	Indicates the LBT procedure result of SFN/SF: Values: 0 = SUCCESS – indicates that the channel was successfully acquired 1 = FAILURE – indicates failure to acquire the channel by the end of SFN/SF
LTE TXOP symbols	uint32_t	Actual LTE TXOP in symbols. Valid when LBT result = SUCCESS
Initial Partial SF	uint32_t	Indicates whether the initial SF in the LBT process is full or partial Values: 0 – full SF (two slots, 14 symbols) 1 – partial SF (only 2nd slot, last 7 symbols)

Table 4-184 LBT_PDSCH_RSP PDU Release 13 parameters

4.2.8.2 LBT_DRS_RSP PDU

The format of a LBT_DRS PDU is shown in Table 4-185.

Tag Value	Field	Type	Description	FAPI	nFAPI
0x205a	Release 13 parameters	struct	The PHY shall support Release 13 and the parameters described in Table 4-186 should be included.	V	TLV

Table 4-185 LBT_DRS_RSP PDU

Field	Type	Description
Handle	uint32_t	An opaque handling returned in LBT_PDSCH_RSP PDU in LBT_DL.indication
result	uint32_t	Indicates the LBT procedure result of SFN/SF: Values: 0 = SUCCESS – indicates that DRS is sent 1 = FAILURE – indicates that DRS is not sent

Table 4-186 LBT_DRS_RSP PDU Release 13 parameters

4.2.9 P7 nFAPI Tag Values

The tag values used in the TLV sections of nFAPI P7 messages are given in Table 4-187.

For FAPI this TLV structure is not used.

Tag (Hex)	Description
0x2000	DL Config request body
0x2001	DCI DL PDU release 8 parameters
0x2002	DCI DL PDU release 9 parameters
0x2003	DCI DL PDU release 10 parameters
0x2039	DCI DL PDU release 11 parameters
0x203a	DCI DL PDU release 12 parameters
0x203b	DCI DL PDU release 13 parameters
0x2004	BCH PDU release 8 parameters
0x2005	MCH PDU release 8 parameters
0x2006	DLSCH PDU release 8 parameters
0x2007	DLSCH PDU release 9 parameters
0x2008	DLSCH PDU release 10 parameters
0x203c	DLSCH PDU release 11 parameters
0x203d	DLSCH PDU release 12 parameters
0x203e	DLSCH PDU release 12 parameters
0x2009	PCH PDU release 8 parameters
0x203f	PCH PDU release 13 parameters
0x200a	PRS PDU release 9 parameters
0x200b	CSI-RS PDU release 10 parameters
0x2040	CSI-RS PDU release 13 parameters
0x2041	EPDCCH release 11 parameters

Tag (Hex)	Description
0x2042	EPDCCH release 13 parameters
0x205b	MPDCCH DL PDU release 13 parameters
0x205c	NBCH PDU Release 13 parameters
0x205d	NPDCCH PDU Release 13 parameters
0x205e	NDLSCH PDU Release 13 parameters
0x200c	UL Config request body
0x200d	ULSCH PDU release 8 parameters
0x200e	ULSCH PDU release 10 parameters
0x2043	ULSCH PDU release 11 parameters
0x2044	ULSCH PDU release 13 parameters
0x200f	Initial transmission release 8 parameters
0x2010	CQI RI (ULSCH) release 8 parameters
0x2011	CQI RI (ULSCH) release 9 parameters
0x2045	CQI RI (ULSCH) release 13 parameters
0x2012	HARQ (ULSCH) release 10 parameters
0x2046	HARQ (ULSCH) release 13 parameters
0x2013	UE information release 8 parameters
0x2047	UE information release 11 parameters
0x2048	UE information release 13 parameters
0x2014	CQI (UCI) release 8 parameters
0x2015	CQI (UCI) release 10 parameters
0x2049	CQI (UCI) release 13 parameters
0x2016	SR release 8 parameters
0x2017	SR release 10 parameters
0x2018	HARQ (UCI) TDD release 10 parameters
0x2019	HARQ (UCI) FDD release 8 parameters
0x201a	HARQ (UCI) FDD release 9 parameters
0x204a	HARQ (UCI) FDD/TDD release 11 parameters
0x204b	HARQ (UCI) FDD/TDD release 13 parameters
0x201b	SRS release 8 parameters
0x201c	SRS release 10 parameters
0x204c	SRS release 13 parameters
0x205f	NULSCH PDU Release 13 parameters
0x2067	NRACH PDU Release 13 parameters

Tag (Hex)	Description
0x2061	NB-HARQ information for Release 13
0x201d	HI DCI0 body
0x201e	HI release 8 parameters
0x201f	HI release 10 parameters
0x2020	DCI UL PDU release 8 parameters
0x2021	DCI UL PDU release 10 parameters
0x204d	DCI UL PDU release 12 parameters
0x204e	MPDCCH DCI UL PDU release 13 parameters
0x2062	NPDCCH DCI UL PDU Release 13 parameters
0x2022	TX request body
0x2023	RX indication body
0x2024	RX indication release 8 parameters
0x2025	RX indication release 9 parameters
0x2026	HARQ indication body
0x2027	HARQ indication TDD release 8 parameters
0x2028	HARQ indication TDD release 9 parameters
0x204f	HARQ indication TDD release 13 parameters
0x2029	HARQ indication FDD release 8 parameters
0x202a	HARQ indication FDD release 9 parameters
0x2050	HARQ indication FDD release 13 parameters
0x202b	CRC indication body
0x202c	CRC indication release 8 parameters
0x202d	RX SR indication body
0x202e	RX CQI indication body
0x202f	RX CQI indication release 8 parameters
0x2030	RX CQI indication release 9 parameters
0x2031	RACH indication body
0x2032	RACH indication release 8 parameters
0x2033	RACH indication release 9 parameters
0x2051	RACH indication release 13 parameters
0x2034	SRS indication body
0x2035	SRS indication release 8 parameters
0x2036	SRS indication release 9 parameters
0x2037	SRS indication release 10 parameters

Tag (Hex)	Description
0x2053	SRS indication release 11 parameters
0x2054	SRS indication TDD channel measurement parameters
0x2038	RX UE information
0x2052	UL_CQI information
0x2063	NB_HARQ indication body
0x2064	NB_HARQ information for Release 13
0x2065	NRACH indication body
0x2066	NRACH indication Release 13 parameters
0x2055	LBT_DL Config request body
0x2056	LBT_PDSCH_REQ PDU release 13 parameters
0x2057	LBT_DRS_REQ PDU release 13 parameters
0x2058	LBT_DL indication message body
0x2059	LBT_PDSCH_RSP PDU release 13 parameters
0x205a	LBT_DRS_RSP PDU release 13 parameters
0x2068 to 0x2fff	RESERVED for nFAPI P7 Tags

Table 4-187 nFAPI P7 tag values

5. Network Monitor Mode (P4)

5.1 Introduction

The nFAPI and FAPI P4 interface configures and operates the Network Monitor Mode (NMM) functions within the capability of the PHY(s). This section defines how NMM will operate for nFAPI. It is possible to use the same procedures and messages for FAPI.

NMM is typically a sub-component of SON, and the Small Cell Forum has previously defined a SON API for Small Cells [3].

5.2 NMM Procedures

5.2.1 Generic NMM Procedures

The NMM procedures are a set of procedures which perform an NMM function. All the NMM procedures follow a common generic pattern which is illustrated in Figure 5-1.

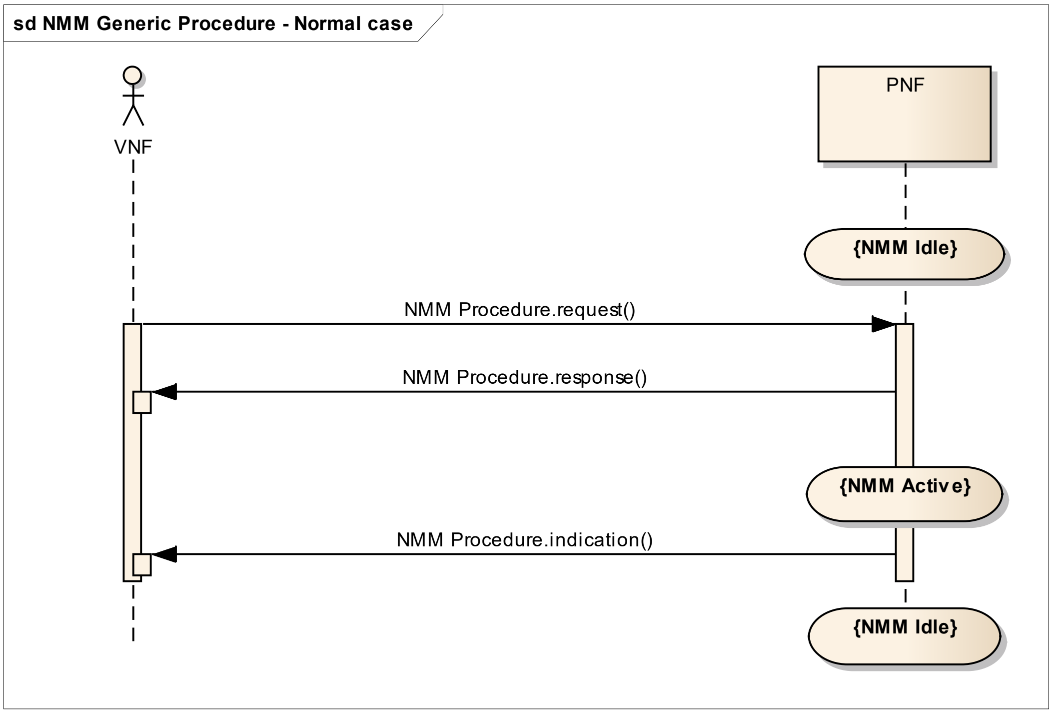


Figure 5-1 General NMM Procedure - Normal case

This generic procedure is used for each of the specific procedures described in the following sections, unless otherwise specified below.

These NMM procedures may be carried out in the Configured state, or depending on the PHY capabilities, may also be carried out in the Running state. There are only 2 NMM states, NMM Idle and NMM Active, which are sub-states of the PHY state. Transitions of the PHY state are only permitted in the NMM Idle sub state.

Any of the NMM procedures may also be terminated by the NMM Stop message exchange as illustrated in Figure 5-2.

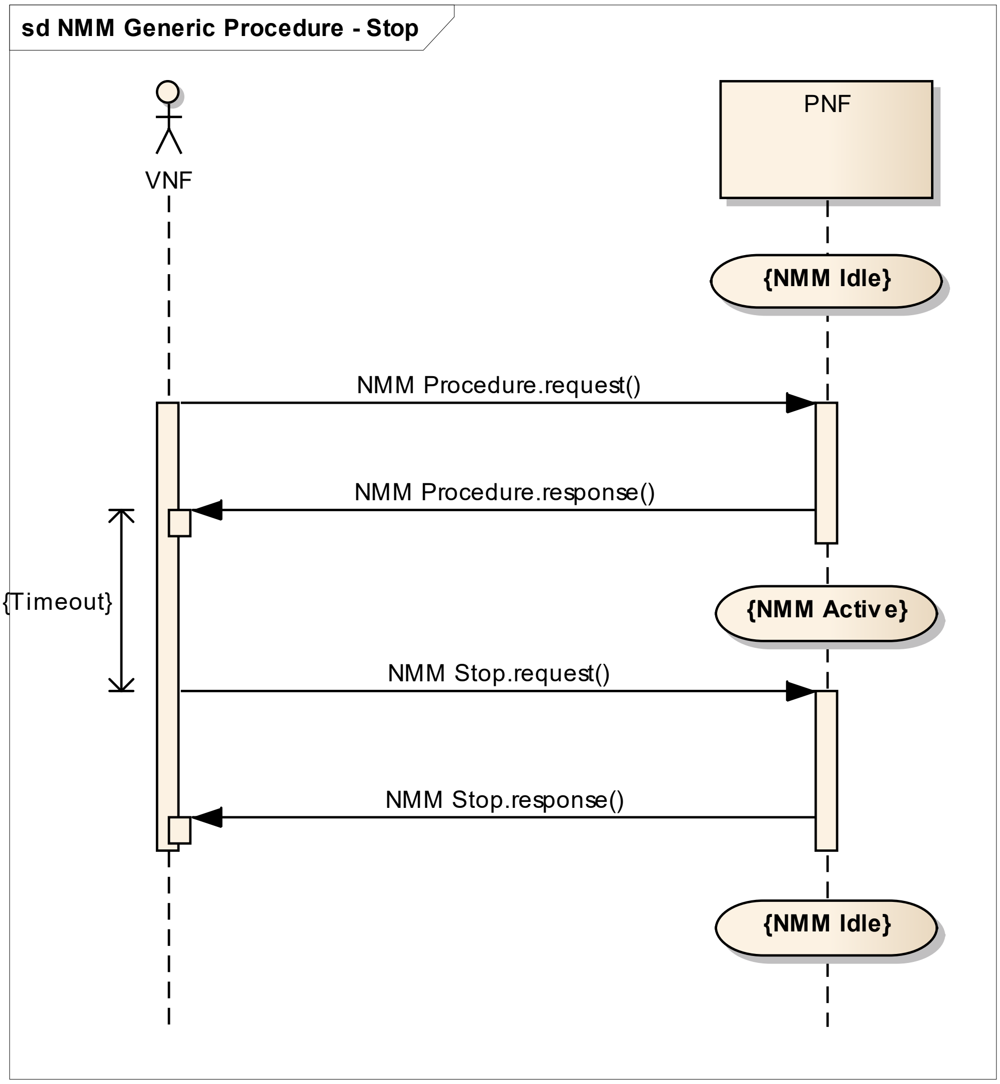


Figure 5-2 General NMM Procedure - Stopped

The PNF may emit the NMM procedure terminating indication in advance of the NMM Stop response, containing partial or full results from the NMM procedure. Since the NMM Stop request from the VNF may cross with a NMM procedure indication from the PNF at the normal termination of the NMM procedure, it is not an error for the NMM Stop request to arrive after completion of the NMM procedure at the PNF in the NMM Idle state, as illustrated in Figure 5-3.

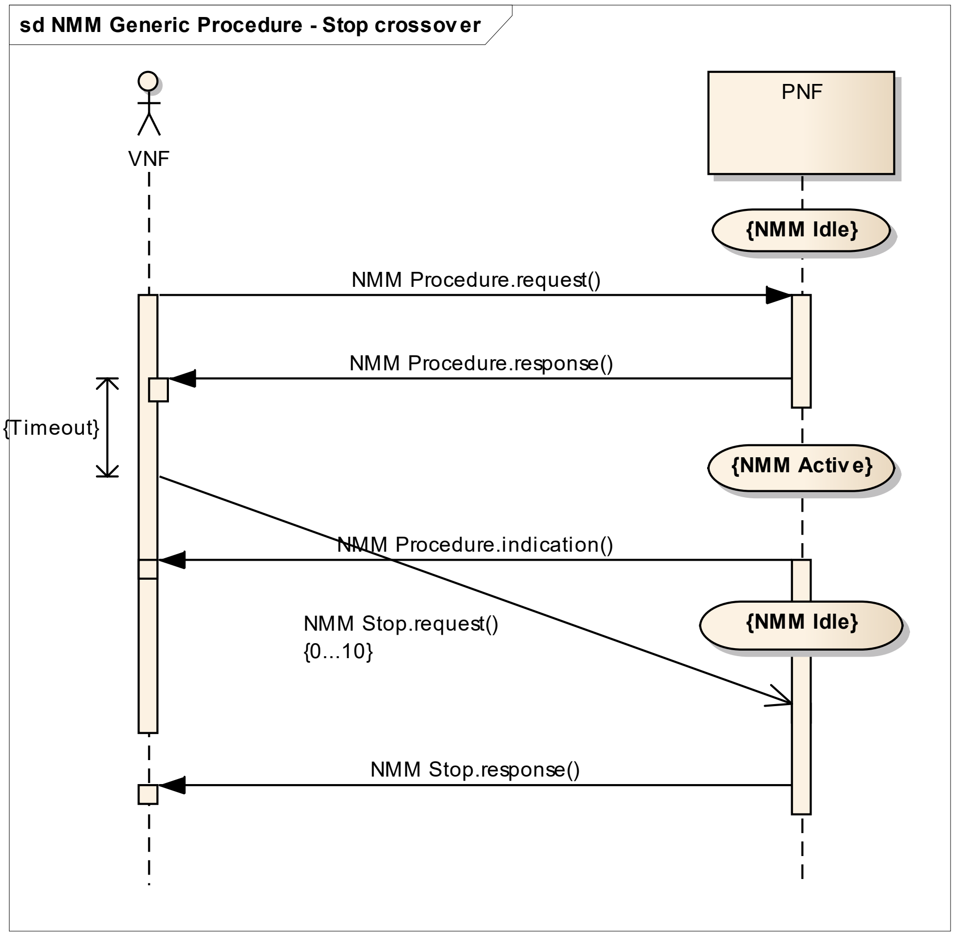


Figure 5-3 General NMM Procedure - Stopped with crossover of normal termination

5.2.2 NMM Procedure Timeout

All NMM procedures also include a Timeout value. This value should be used by the PNF to guide the maximum time it should take to complete the procedure, and thus may control the quality or likely success of the results. If this timeout expires before successful completion of the procedure, the PNF should abort the procedure, and indicate a timeout failure.

5.2.3 NMM Procedure Sequences

The NMM procedures must be carried out in the order they are presented below (except the RSSI procedure, which is optional, and may be carried out in any order with respect to the other procedures), as each typically depends on state information retrieved in a previous procedure.

The basic full NMM sequence for GERAN (which does not use the Broadcast Detect or System Information Schedule procedures) is illustrated in Figure 5-4 below.

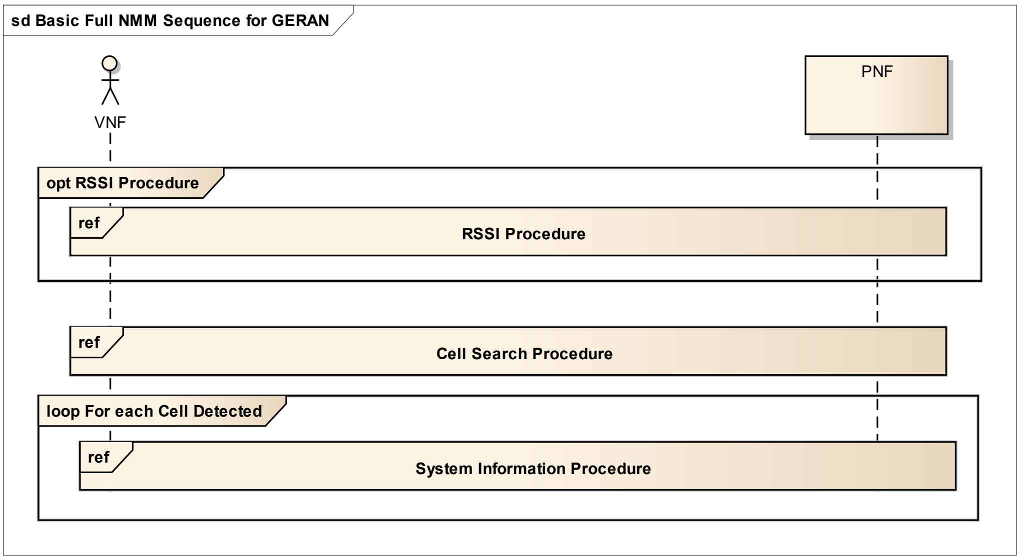


Figure 5-4 Basic full NMM Sequence for GERAN

The basic full NMM sequence for UTRAN (which does not use the System Information Schedule procedure) is illustrated in Figure 5-5 below.

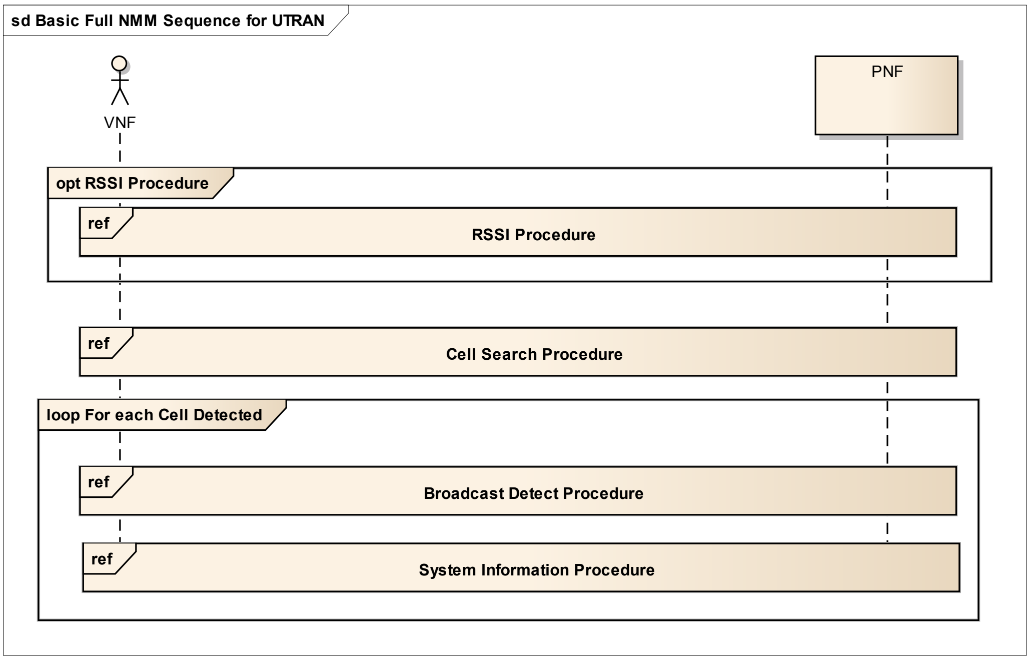


Figure 5-5 Basic full NMM Sequence for UTRAN

The basic full NMM sequence for LTE (which uses the System Information Schedule procedure) is illustrated in Figure 5-6 below.

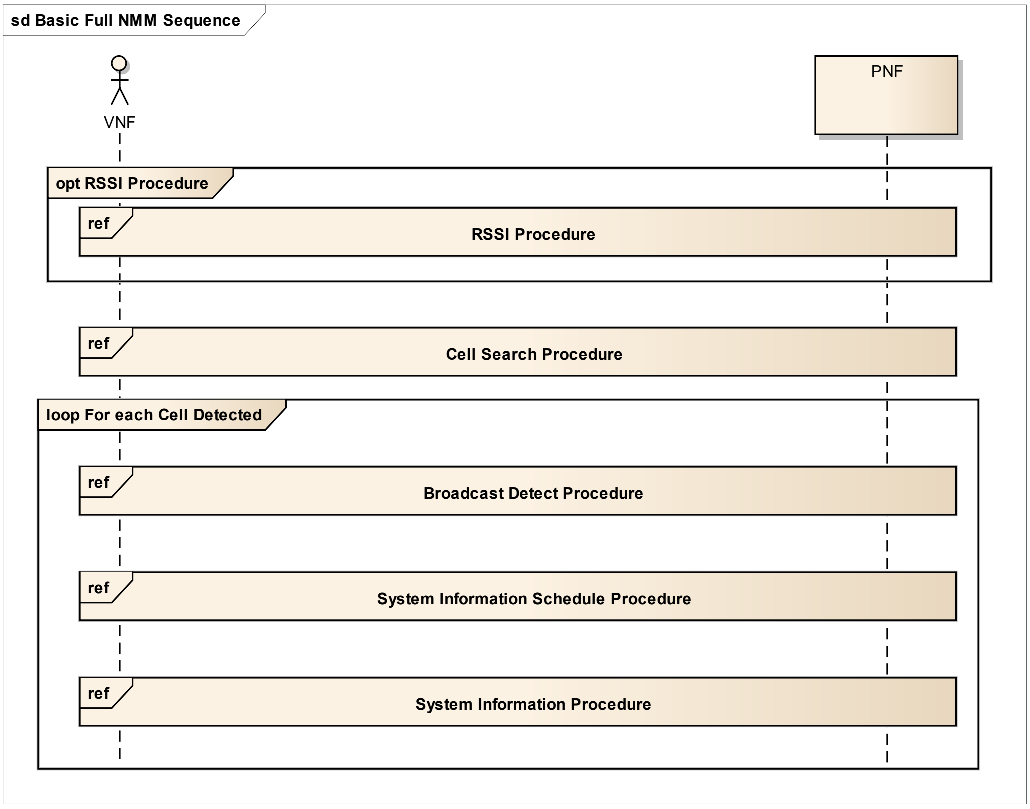


Figure 5-6 Basic full NMM Sequence for LTE

For LTE, which requires the cell bandwidth to have been determined from the Broadcast Detect procedure before performing the Information Schedule and System Information procedures with the correct bandwidth, it may be more efficient to perform the Bandwidth Detect procedures for all detected cells before performing any of the System Information Schedule and System Information procedures, as illustrated in Figure 5-7 below.

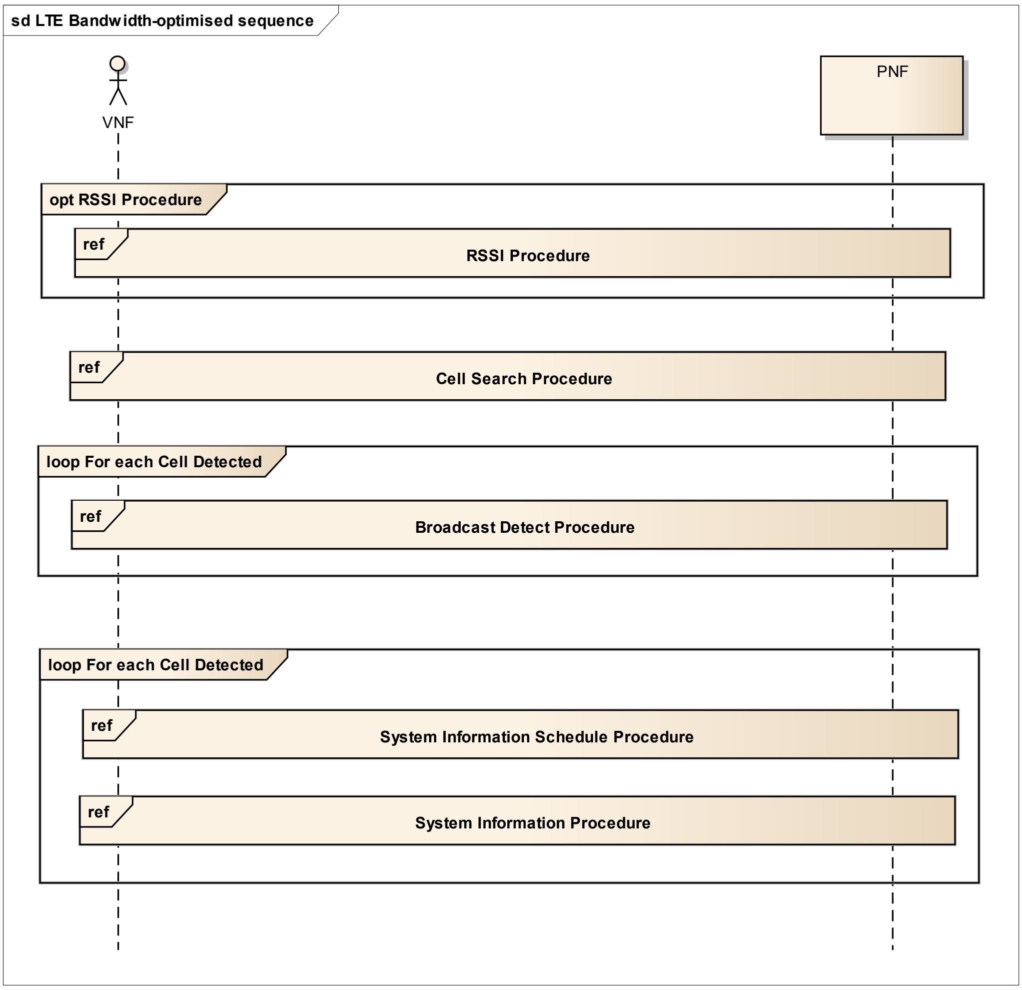


Figure 5-7 Bandwidth-optimised LTE sequence

The basic full NMM sequence for NB-IoT (which uses the System Information Schedule procedure) is illustrated in Figure 5-8 below.

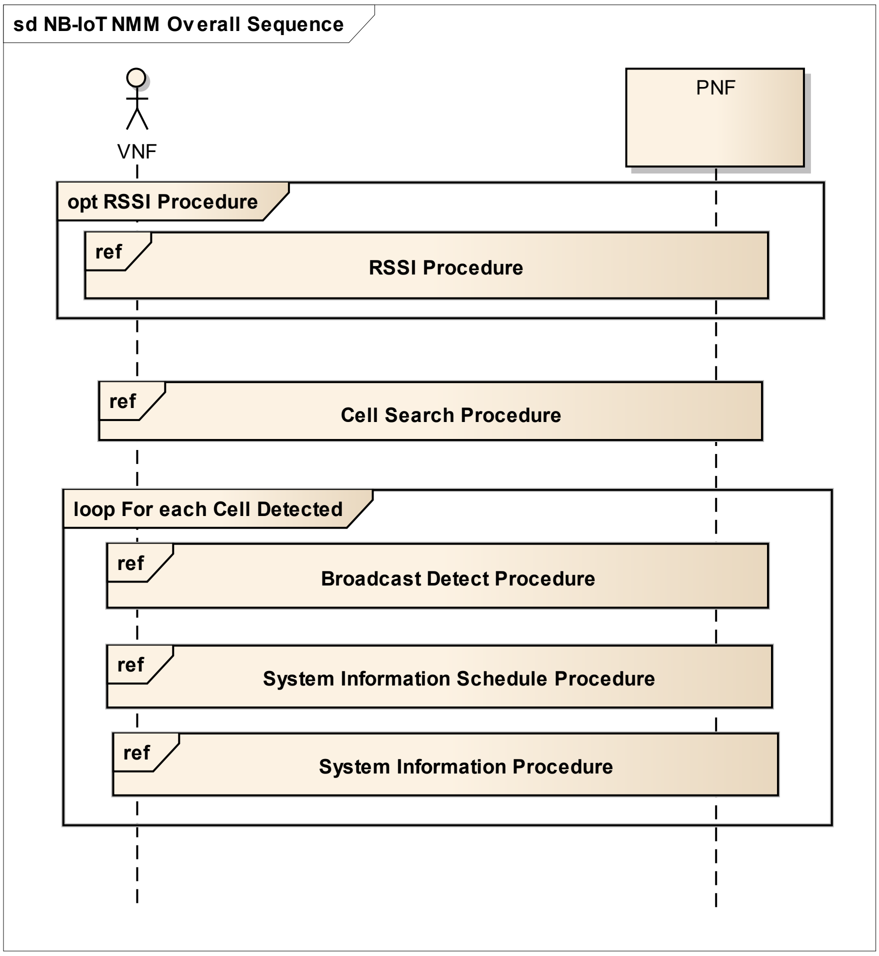


Figure 5-8 Basic full NMM sequence for NB-IoT

The above sequences may be repeated for each LTE EARFCN, UTRAN UARFCN, GERAN band or NB-IoT EARFCN (and Raster offset) for which NMM information is required. All other possible sequences are outside the scope of this specification.

5.2.4 RSSI Procedure

This procedure is used to measure the RSSI of a set of channels at a specified bandwidth. It does not require any signals to be demodulated, and depending on PHY capabilities, may be carried out on channels which are normally uplink or downlink channels.

5.2.5 Cell Search Procedure

This procedure is used to attempt to find one or more cells of the specified technologies which are broadcasting on a set of channels. It only attempts to detect the cells by detecting their fundamental synchronisation channels, and does not decode any information from the broadcast channels.

5.2.6 Broadcast Detect Procedure

This procedure is used to synchronise to the cell and to return the fundamental broadcast information (e.g. MIB) from a detected cell. For GERAN there is no such fundamental broadcast information, so this procedure is not used.

5.2.7 System Information Schedule Procedure

This procedure is used to return the System Information Schedule information, and is only relevant to those technologies which use SI Schedule information as a necessary part of their System Information broadcasts. It is not used for UTRAN or GERAN.

5.2.8 System Information Procedure

This procedure is used to return one or more of the System Information messages from a detected cell. This procedure uses a modified version of the NMM generic procedure, with multiple indications from the PNF to the VNF, as illustrated in Figure 5-9 below.

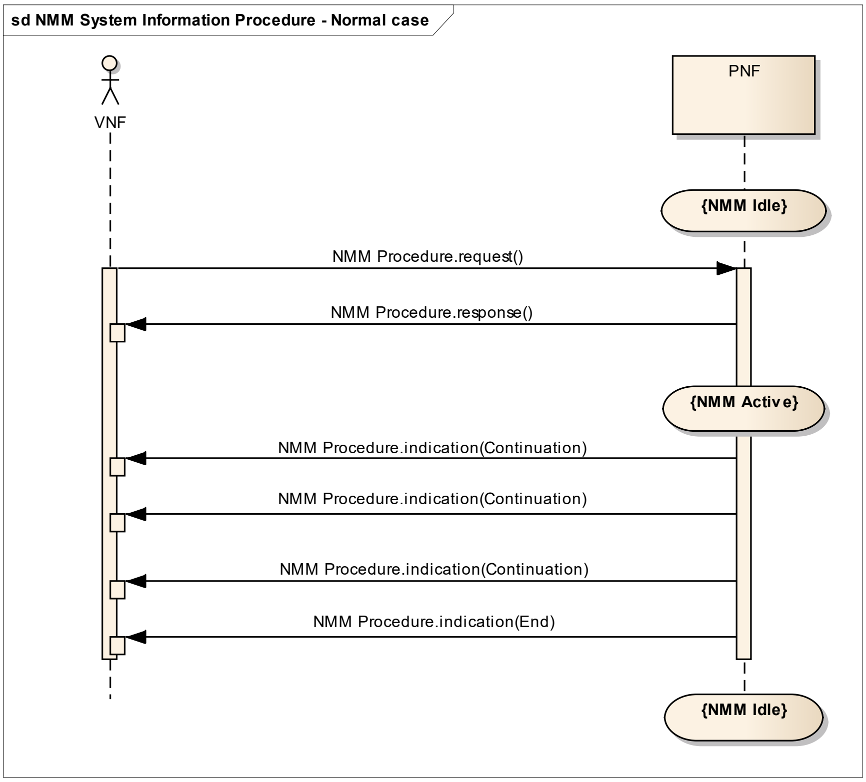


Figure 5-9 NMM System Information Procedure

Each of the indication messages from the PNF to the VNF includes a cause code which states whether it is the final indication or not, and the procedure is terminated by the final indication. Each indication carries a single System Information element except that the final indication may optionally omit any System Information element, and only indicate the termination of the procedure.

5.3 NMM Messages

5.3.1 Transport Layer

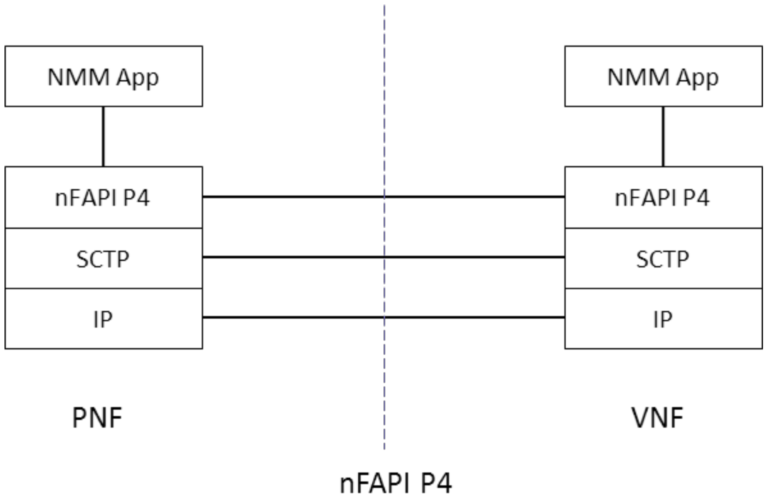


Figure 5-10 nFAPI P4 Protocol Stack

The transport layer for nFAPI P4 is required to ensure reliable message transfer, so SCTP is used as the transport protocol. Therefore there are no extra methods to ensure reliable transmission of messages at the application level (such as timers, packet validation and packet loss).

There is one SCTP association between the PNF and the VNF, established by the PNF based on P9 OAM configuration for VNF address and port. The SCTP stream for each nFAPI P4 instance is integrated with an nFAPI PHY instance, so the same SCTP stream is shared between P4 and P5.

Transport network redundancy may be achieved by SCTP multi-homing between two end-points, of which one or both is assigned with multiple IP addresses. SCTP end-points support a multi-homed remote SCTP end-point. For SCTP endpoint redundancy an INIT may be sent from the VNF or PNF, at any time for an already established SCTP association, which shall be handled as defined in IETF RFC 4960 [18].

5.3.2 General Message Format

The general message format in nFAPI is described in Section 0

The general message format in FAPI is described in Section 3.3.1

Message	Value	Message Body Definition
RSSI.request	0x0200	See Section 5.3.3.1
RSSI.response	0x0201	See Section 5.3.3.1
RSSI.indication	0x0202	See Section 5.3.3.1
CELL_SEARCH.request	0x0203	See Section 5.3.3.2
CELL_SEARCH.response	0x0204	See Section 5.3.3.2

Message	Value	Message Body Definition
CELL_SEARCH.indication	0x0205	See Section 5.3.3.2
BROADCAST_DETECT.request	0x0206	See Section 5.3.3.3
BRAODCAST_DETECT.response	0x0207	See Section 5.3.3.3
BRAODCAST_DETECT.indication	0x0208	See Section 5.3.3.3
SYSTEM_INFORMATION_SCHEDULE.request	0x0209	See Section 5.3.3.4
SYSTEM_INFORMATION_SCHEDULE.response	0x020a	See Section 5.3.3.4
SYSTEM_INFORMATION_SCHEDULE.indication	0x020b	See Section 5.3.3.4
SYSTEM_INFORMATION.request	0x020c	See Section 5.3.3.5
SYSTEM_INFORMATION.response	0x020d	See Section 5.3.3.5
SYSTEM_INFORMATION.indication	0x020e	See Section 5.3.3.5
NMM_STOP.request	0x020f	See Section 5.3.3.6
NMM_STOP.response	0x0210	See Section 5.3.3.6
RESERVED	0x0211 to 0x02ff	Reserved for NMM P4 messages

Table 5-1 P4 message types

5.3.3 P4 Messages

5.3.3.1 RSSI

RSSI.request

The `RSSI.request` message is sent by the VNF to request that the PNF performs an RSSI measurement on the requested channels at the requested bandwidth. Only one of the RAT-specific structures shall be included in the message.

Tag	Field	Type	Description	nFAPI
N/A	RAT Type	uint8_t	RAT Type of this request, which specifies which of the possible TLV structures is included below: 0=LTE, 1=UTRAN, 2=GERAN, 3=NB-IoT	V
0x3000	LTE RSSI Request	struct	The LTE RSSI request information as defined in Table 5-3	TLV
0x3001	UTRAN RSSI Request	struct	The UTRAN RSSI request information as defined in Table 5-4	TLV
0x3002	GERAN RSSI Request	struct	The GERAN RSSI request information as defined in Table 5-5	TLV
0x3020	NB-IoT RSSI Request	struct	The NB-IoT RSSI request information as defined in Table 5-6	TLV

Table 5-2 RSSI.request parameters

LTE RSSI Request

Field		Type	Description
Tag		uint16_t	The RSSI Request Tag
Length		uint16_t	The length in bytes of the values
Frequency Band Indicator		uint8_t	The E-UTRA band for which the carrierList applies.
Measurement Period		uint16_t	The length of time to measure RSSI over, in units of 1ms.
Bandwidth		uint8_t	The bandwidth (in resource blocks) over which the RSSI is measured. Value: 6,15, 25, 50, 75,100
Timeout		uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of EARFCNs		uint8_t	The number of EARFCNs which should be measured. In the case that no EARFCN (value 0) is specified, all valid EARFCNs for the specified bandwidth in the band shall be measured, in order of ascending EARCFN. Range: 0… MAX_CARRIER_LIST.
For (Number of EARFCNs) {			
	EARFCN	uint16_t	The list of EARFCNs to be measured.
}			

Table 5-3 LTE RSSI Request

UTRAN RSSI Request

Field		Type	Description
Tag		uint16_t	The RSSI Request Tag
Length		uint16_t	The length in bytes of the values
Frequency Band Indicator		uint8_t	The UTRA band for which the carrierList applies.
Measurement Period		uint16_t	The length of time to measure RSSI over, in units of 1ms.
Timeout		uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of UARFCNs		uint8_t	The number of UARFCNs which should be measured. In the case that no UARFCN (value 0) is specified, all UARFCNs in the band shall be measured, in order of ascending UARCFN. Range: 0… MAX_CARRIER_LIST.
For (Number of UARFCNs) {			
	UARFCN	uint16_t	The list of UARFCNs to be measured.

Field	Type	Description
}		

Table 5-4 UTRAN RSSI Request

GERAN RSSI Request

Field	Type	Description
Tag	uint16_t	The RSSI Request Tag
Length	uint16_t	The length in bytes of the values
Frequency Band Indicator	uint8_t	The GERAN band for which the carrierList applies. Values, as defined in 3GPP 45.005: 1 = T-GSM 380 2 = T-GSM 410 3 = GSM 450 4 = GSM 480 5 = GSM 710 6 = GSM 750 7 = T-GSM 810 8 = GSM 850 9 = P-GSM 900 10 = E-GSM 900 11 = R-GSM 900 12 (void) 13 = DCS 1800 14 = PCS 1900
Measurement Period	uint16_t	The length of time to measure RSSI over, in units of 1ms.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of ARFCNs	uint8_t	The number of ARFCNs which should be measured. In the case that no ARFCN (value 0) is specified, all ARFCNs in the band shall be measured, in order of ascending ARCFN. Range: 0… MAX_CARRIER_LIST.
For (Number of ARFCNs) {		
ARFCN	uint16_t	The list of ARFCNs to be measured.
Direction	uint8_t	The link direction to be measured. Value: 0=DL, 1=UL
}		

Table 5-5 GERAN RSSI Request

NB-IoT RSSI Request

Field	Type	Description
NB-IoT RSSI RequestTag	uint16_t	The RSSI Request Tag

Length		uint16_t	The length in bytes of the values
Frequency Band Indicator		uint8_t	The band for which the carrierList applies. Band indicator is encoded as specified in 3GPP 36.104.
Measurement Period		uint16_t	The length of time to measure RSSI over, in units of 1ms.
Timeout		uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of EARFCNs		uint8_t	The number of EARFCNs which should be measured. In the case that no EARFCN (value 0) is specified, all valid EARFCNs and raster offsets in the band shall be measured, in order of ascending EARCFN. Range: 0… MAX_CARRIER_LIST.
For (Number of EARFCNs) {			
	EARFCN	uint16_t	The value of EARFCN to be measured.
	Number of RO_{DL}	uint8_t	RO_{DL} is used to specify the NB-IoT raster offset from the EARFCN channel. If no offset is specified then RSSI should be measured on all valid offset values.
	For (Number of RO_{DL}) {		
	RO_{DL}	uint8_t	Raster offset from the EARFCN channel, encoded as below: Value: 0 = -7.5KHz 1 = -2.5KHz 2 = 0 (aligned to carrier raster) 3 = +2.5KHz 4 = +7.5KHz
	}		
}			

Table 5-6 NB-IoT RSSI Request

RSSI.response

The `RSSI.response` message is sent by the PNF on receipt of an RSSI request. If the error code indicates no error, the PNF shall start the requested procedure, and shall send the `RSSI.indication` once the procedure is complete.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-7 RSSI.response parameters

RSSI.response errors

The error codes that may be returned in `RSSI.response` are given in Table 5-8.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.

Table 5-8 Error codes for `RSSI.response`

RSSI.indication

The `RSSI.indication` message is sent by the PNF on completion of the RSSI procedure, and conveys the results of the RSSI procedure to the VNF.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x3003	RSSI Indication	struct	The RSSI indication information as defined in Table 5-10	TLV

Table 5-9 RSSI.indication parameters

RSSI Indication

Field		Type	Description
Tag		uint16_t	The RSSI Indication Tag
Length		uint16_t	The length in bytes of the values
Number of RSSI		uint16_t	The number of RSSI results returned in the following array.
For (Number of RSSI) {			
	RSSI	int16_t	The list of RSSI values of the carriers measured, in the order of the list of the original request. While measuring NB-IoT cell, if Raster offset was not specified then, for every channel 5 RSSI values will be reported in the order of the list of all valid RO_{DL} values as specified in Table 5-6 Value in steps of 0.1dBm, with 0=0dBm.
}			

Table 5-10 RSSI Indication

RSSI.indication errors

The error codes that may be returned in `RSSI.indication` are given in Table 5-11.

Error code	Description
MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an NMM STOP.request and the results so far collected, if any, are included in the indication.
MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-11 Error codes for `RSSI.indication`

5.3.3.2 CELL SEARCH

CELL SEARCH.request

The `CELL SEARCH.request` message is sent by the VNF to request that the PNF performs a CELL SEARCH measurement on the requested channels. Only one of the RAT-specific structures shall be included in the message.

Tag	Field	Type	Description	nFAPI
N/A	RAT Type	Uint_8	RAT Type of this request, which specifies which of the possible TLV structures is included below: 0=LTE, 1=UTRAN, 2=GERAN, 3=NB-IoT	V
0x3004	LTE CELL SEARCH Request	struct	The LTE CELL SEARCH request information as defined in Table 5-13	TLV
0x3005	UTRAN CELL SEARCH Request	struct	The UTRAN CELL SEARCH request information as defined in Table 5-14	TLV
0x3006	GERAN CELL SEARCH Request	struct	The GERAN CELL SEARCH request information as defined in Table 5-15	TLV
0x3021	NB-IoT CELL SEARCH Request	struct	The NB-IoT CELL SEARCH request information as defined in Table 5-16	TLV

Table 5-12 CELL SEARCH.request parameters

LTE CELL SEARCH Request

Field	Type	Description
Tag	uint16_t	The LTE CELL SEARCH Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN for which cells should be measured.
Measurement Bandwidth	uint8_t	The number of resource blocks which should be used

Field		Type	Description
			for measuring RSRP. See [12].
Exhaustive Search		uint8_t	NMM should try to find all cells on the carrier. Value: 0=non-exhaustive search, 1=exhaustive search
Timeout		uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of PCI		uint8_t	The number of cells in the PCI list. If 0 all cells on the carrier should be found. Otherwise, depending on exhaustiveSearch flag, only the given pciList is searched or the pciList is used for indicating a priority list. Range: 0 to MAX_PCI_LIST.
For (Number of PCI) {			
	PCI	uint16_t	The list of PCI for cells which should be searched.
}			

Table 5-13 LTE CELL SEARCH Request

UTRAN CELL SEARCH Request

Field		Type	Description
Tag		uint16_t	The UTRAN CELL SEARCH Request Tag
Length		uint16_t	The length in bytes of the values
UARFCN		uint16_t	The UARFCN for which cells should be measured.
Exhaustive Search		uint8_t	NMM should try to find all cells on the carrier. Value: 0=non-exhaustive search, 1=exhaustive search
Timeout		uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of PSC		uint8_t	The number of cells in the PSC list. If 0 all cells on the carrier should be found. Otherwise, depending on Exhaustive Search flag, only the given PSC list is searched or the PSC list is used for indicating a priority list. Range: 0 to MAX_PSC_LIST.
For (Number of PSC) {			
	PSC	uint16_t	The list of PSC for cells which should be searched.
}			

Table 5-14 UTRAN CELL SEARCH Request

GERAN CELL SEARCH Request

Field	Type	Description
Tag	uint16_t	The GERAN CELL SEARCH Request Tag
Length	uint16_t	The length in bytes of the values
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of ARFCN	uint8_t	The number of cells in the ARFCN List. Range: 0 to MAX_ARFCN_LIST.
For (Number of ARFCN) {		
ARFCN	uint16_t	The list of ARFCN for cells which should be searched.
}		

Table 5-15 GERAN CELL SEARCH Request

NB-IoT CELL SEARCH Request

Field	Type	Description
Tag	uint16_t	The NB-IoT CELL SEARCH Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN for which cells should be measured.
RO_{DL}	uint8_t	Raster offset from the EARFCN channel, encoded as below: Value: 0 = -7.5KHz 1 = -2.5KHz 2 = 0 (aligned to carrier raster) 3 = +2.5KHz 4 = +7.5KHz
Exhaustive Search	uint8_t	NMM should try to find all cells on the carrier. Value: 0=non-exhaustive search, 1=exhaustive search
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.
Number of PCI	uint8_t	The number of cells in the PCI list. If 0 all cells on the carrier should be found. Otherwise, depending on exhaustiveSearch flag, only the given pciList is searched or the pciList is used for indicating a priority list. Range: 0 to MAX_PCI_LIST.
For (Number of PCI) {		
PCI	uint16_t	The list of PCI for cells which should be searched.
}		

Table 5-16 NB-IoT CELL SEARCH Request

CELL SEARCH.response

The `CELL SEARCH.response` message is sent by the PNF on receipt of a CELL SEARCH request. If the error code indicates no error, the PNF shall start the requested procedure, and shall send the `CELL SEARCH.indication` once the procedure is complete.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-17 CELL SEARCH.response parameters

CELL SEARCH.response errors

The error codes that can be returned in `CELL SEARCH.response` are given in Table 5-18.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.

Table 5-18 Error codes for `CELL SEARCH.response`

CELL SEARCH.indication

The `CELL SEARCH.indication` message is sent by the PNF on completion of the CELL SEARCH procedure, and conveys the results of the CELL SEARCH procedure to the VNF.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x3007	LTE CELL SEARCH Indication	Struct	The LTE CELL SEARCH indication information as defined in Table 5-20	TLV
0x3008	UTRAN CELL SEARCH Indication	Struct	The UTRAN CELL SEARCH indication information as defined in Table 5-21	TLV
0x3009	GERAN CELL SEARCH Indication	Struct	The GERAN CELL SEARCH indication information as defined in Table 5-22	TLV
0x300a	PNF CELL SEARCH	Struct	A structure of opaque data optionally sent by the PNF to the VNF. If	TLV

Tag	Field	Type	Description	nFAPI
	STATE		included, the VNF shall include the same structure of data in all subsequent BROADCAST DETECT request messages for any of the found cells.	
0x3022	NB-IoT CELL SEARCH Indication	Struct	The NB-IoT CELL SEARCH indication information as defined in Table 5-23	TLV

Table 5-19 CELL SEARCH.indication parameters

LTE CELL SEARCH Indication

Field		Type	Description
Tag		uint16_t	The LTE CELL SEARCH Indication Tag
Length		uint16_t	The length in bytes of the values
Number of LTE Cells Found		uint16_t	The number of LTE cells indicated in this message.
For (Number of LTE Cells Found) {			
	PCI	uint16_t	The physical cell identity of the cell which the NMM synchronized to.
	RSRP	uint8_t	The measured RSRP value in units of -1dB. See [12].
	RSRQ	uint8_t	The measured RSRQ value in units of -1dB. See [12].
	Frequency Offset	int16_t	The estimated frequency offset (in ppb) of the detected cell compared to the timebase clock of the PNF. A value of 0x8000 shall indicate that no Frequency Offset information is available.
}			

Table 5-20 LTE CELL SEARCH Indication

UTRAN CELL SEARCH Indication

Field		Type	Description
Tag		uint16_t	The UTRAN CELL SEARCH Indication Tag
Length		uint16_t	The length in bytes of the values
Number of UTRAN Cells Found		uint16_t	The number of LTE cells indicated in this message.
For (Number of UTRAN Cells Found) {			
	PSC	uint16_t	The primary scrambling code of the cell which the NMM synchronized to.
	RSCP	uint8_t	The measured RSCP value in units of -1dB. See [20].
	EcN0	uint8_t	The measured Ec/No value in units of -1dB. See

Field		Type	Description
			[20].
	Frequency Offset	int16_t	The estimated frequency offset (in ppb) of the detected cell compared to the timebase clock of the PNF. A value of 0x8000 shall indicate that no Frequency Offset information is available.
}			

Table 5-21 UTRAN CELL SEARCH Indication

GERAN CELL SEARCH Indication

Field		Type	Description
Tag		uint16_t	The GERAN CELL SEARCH Indication Tag
Length		uint16_t	The length in bytes of the values
Number of GSM Cells Found		uint16_t	The number of GSM cells indicated in this message.
For (Number of GSM Cells Found) {			
	ARFCN	uint16_t	The ARFCN of the cell which the NMM synchronized to.
	BSIC	uint8_t	The BSIC of the cell which the NMM synchronized to.
	RxLev	uint8_t	The measured RxLev value. See [19].
	RxQual	uint8_t	The measured RxQual value. See [19].
	Frequency Offset	int16_t	The estimated frequency offset (in ppb) of the detected cell compared to the timebase clock of the PNF. A value of 0x8000 shall indicate that no Frequency Offset information is available.
	SFN Offset	uint32_t	The offset in µs of the start of the current GSM Radio HyperFrame (i.e. FN=0) from the start of the preceding LTE Radio Frame of the PNF for SFN=0. Valid range: 0 to 10239999. If the PNF does not have a SFN/SF reference timebase operational or cannot derive this offset value, it should set this to the value 0xFFFFFFFF.
}			

Table 5-22 GERAN CELL SEARCH Indication

NB-IoT CELL SEARCH Indication

Field		Type	Description
Tag		uint16_t	The NB-IoT CELL SEARCH Indication Tag
Length		uint16_t	The length in bytes of the values
Number of NB-IoT Cells Found		uint16_t	The number of NB-IoT cells indicated in this message.
For (Number of NB-IoT Cells Found) {			
	PCI	uint16_t	The physical cell identity of the cell which the NMM

Field		Type	Description
			synchronized to.
	RSRP	uint8_t	The measured RSRP value in units of -1dB. See [12].
	RSRQ	uint8_t	The measured RSRQ value in units of -1dB. See [12].
	Frequency Offset	int16_t	The estimated frequency offset (in ppb) of the detected cell compared to the timebase clock of the PNF. A value of 0x8000 shall indicate that no Frequency Offset information is available.
}			

Table 5-23 NB-IoT CELL SEARCH Indication

CELL SEARCH.indication errors

The error codes that can be returned in `CELL SEARCH.indication` are given in Table 5-11.

Error code	Description
MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an NMM STOP.request and the results so far collected, if any, are included in the indication.
MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-24 Error codes for `CELL SEARCH.indication`

5.3.3.3 BROADCAST DETECT

BROADCAST DETECT.request

The BROADCAST `DETECT.request` message is sent by the VNF to request that the PNF performs a BROADCAST DETECT measurement on the requested cell. Only one of the RAT-specific structures shall be included in the message.

Tag	Field	Type	Description	nFAPI
N/A	RAT Type	uint_8	RAT Type of this request, which specifies which of the possible TLV structures is included below: 0=LTE, 1=UTRAN, 2=NB-IoT	V
0x300b	LTE BROADCAST DETECT Request	struct	The LTE BROADCAST DETECT request information as defined in Table 5-26	TLV
0x300c	UTRAN BROADCAST DETECT Request	struct	The UTRAN BROADCAST DETECT request information as defined in Table 5-27	TLV

Tag	Field	Type	Description	nFAPI
0x300a	PNF CELL SEARCH STATE	struct	A structure of opaque data conditionally sent by the VNF to the PNF. It must be included if the PNF included it in the CELL SEARCH.Indication.	TLV
0x3023	NB-IoT BROADCAST DETECT Request	struct	The NB-IoT BROADCAST DETECT request information as defined in Table 5-28	TLV

Table 5-25 BROADCAST DETECT.request parameters

LTE BROADCAST DETECT Request

Field	Type	Description
Tag	uint16_t	The BROADCAST DETECT Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN for the cell which the NMM should read the PBCH.
PCI	uint16_t	The physical cell identity of the cell which the NMM should read the PBCH.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-26 LTE BROADCAST DETECT Request

UTRAN BROADCAST DETECT Request

Field	Type	Description
Tag	uint16_t	The BROADCAST DETECT Request Tag
Length	uint16_t	The length in bytes of the values
UARFCN	uint16_t	The UARFCN for the cell which the NMM should read the PBCH.
PSC	uint16_t	The primary scrambling code of the cell which the NMM should read the PBCH.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-27 UTRAN BROADCAST DETECT Request

NB-IoT BROADCAST DETECT Request

Field	Type	Description
Tag	uint16_t	The BROADCAST DETECT Request Tag
Length	uint16_t	The length in bytes of the values

Field	Type	Description
EARFCN	uint16_t	The EARFCN for the NB-IoT carrier for which the NMM should read the NPBCH.
RO_{DL}	uint8_8	Raster offset for the carrier Value: 0 = -7.5KHz 1 = -2.5KHz 2 = 0 (aligned to carrier raster) 3 = +2.5KHz 4 = +7.5KHz
PCI	uint16_t	The physical cell identity of the cell which the NMM should read the NPBCH.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-28 NB-IoT BROADCAST DETECT Request

BROADCAST DETECT.response

The `BROADCAST DETECT.response` message is sent by the PNF on receipt of a `BROADCAST DETECT request`. If the error code indicates no error, the PNF shall start the requested procedure, and shall send the `BROADCAST DETECT.indication` once the procedure is complete.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-29 BROADCAST DETECT.response parameters

BROADCAST DETECT.response errors

The error codes that can be returned in `BROADCAST DETECT.response` are given in Table 5-30.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.

Table 5-30 Error codes for `BROADCAST DETECT.response`

BROADCAST DETECT.indication

The `BROADCAST DETECT.indication` message is sent by the PNF on completion of the BROADCAST DETECT procedure, and conveys the results of the BROADCAST DETECT procedure to the VNF.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x300e	LTE BROADCAST DETECT Indication	struct	The LTE BROADCAST DETECT indication information as defined in Table 5-32	TLV
0x300f	UTRAN BROADCAST DETECT Indication	struct	The UTRAN BROADCAST DETECT indication information as defined in Table 5-33	TLV
0x3010	PNF CELL BROADCAST STATE	struct	A structure of opaque data optionally sent by the PNF to the VNF. If included, the VNF shall include the same structure of data in any subsequent SYSTEM INFORMATION SCHEDULE request or subsequent SYSTEM INFORMATION request message for this cell.	TLV
0x3024	NB-IoT BROADCAST DETECT Indication	struct	The NB-IoT BROADCAST DETECT indication information as defined in Table 5-34	TLV

Table 5-31 BROADCAST DETECT.indication parameters

LTE BROADCAST DETECT Indication

Field	Type	Description
Tag	uint16_t	The LTE BROADCAST DETECT Indication Tag
Length	uint16_t	The length in bytes of the value
Number of Tx Antenna	uint8_t	The number of Tx Antenna detected for the cell: 1, 2 or 4
MIB Length	uint16_t	Length in bytes of the following MIB array
MIB[MIB Length]	array of uint8_t	The MIB read from the specified cell.
SFN Offset	uint32_t	The offset in µs of the first Radio Frame which contained this MIB from the start of the preceding Radio Frame of the PNF for SFN=0 and SF=0. Valid range: 0 to 10239999. If the PNF does not have a SFN/SF reference timebase operational or cannot derive this offset value, it should set this to the value 0xFFFFFFFF.

Table 5-32 LTE BROADCAST DETECT Indication

UTRAN BROADCAST DETECT Indication

Field	Type	Description
Tag	uint16_t	The UTRAN BROADCAST DETECT Indication Tag
Length	uint16_t	The length in bytes of the value
MIB Length	uint16_t	Length in bytes of the following MIB array
MIB[MIB Length]	array of uint8_t	The MIB read from the specified cell.
SFN Offset	uint32_t	The offset in μs of the first UTRAN Radio Frame which contained this MIB from the start of the preceding Radio Frame of the PNF for SFN=0 and SF=0. Valid range: 0 to 10239999. If the PNF does not have a SFN/SF reference timebase operational or cannot derive this offset value, it should set this to the value 0xFFFFFFFF.

Table 5-33 UTRAN BROADCAST DETECT Indication

NB-IoT BROADCAST DETECT Indication

Field	Type	Description
Tag	uint16_t	The NB-IoT BROADCAST DETECT Indication Tag
Length	uint16_t	The length in bytes of the value
Number of Tx Antenna	uint8_t	The number of Tx Antenna detected for the cell: 1 or 2
MIB Length	uint16_t	Length in bytes of the following MIB array
MIB[MIB Length]	array of uint8_t	The MIB read from the specified cell.
SFN Offset	uint32_t	The offset in μs of the first Radio Frame which contained this MIB from the start of the preceding Radio Frame of the PNF for SFN=0 and SF=0. Valid range: 0 to 10239999. If the PNF does not have a SFN/SF reference timebase operational or cannot derive this offset value, it should set this to the value 0xFFFFFFFF.

Table 5-34 NB-IoT BROADCAST DETECT Indication

BROADCAST DETECT.indication errors

The error codes that can be returned in `BROADCAST DETECT.indication` are given in Table 5-35.

Error code	Description
MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an NMM STOP.request and the results so far collected, if any, are included in the indication.

Error code	Description
MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-35 Error codes for BROADCAST DETECT Indication

5.3.3.4 SYSTEM INFORMATION SCHEDULE

SYSTEM INFORMATION SCHEDULE.request

The `SYSTEM INFORMATION SCHEDULE.request` message is sent by the VNF to request that the PNF performs a SYSTEM INFORMATION SCHEDULE measurement on the requested cell. Only one of the RAT-specific structures shall be included in the message

Tag	Field	Type	Description	nFAPI
N/A	RAT Type	Uint_8	RAT Type of this request, which specifies which of the possible TLV structures is included below: 0=LTE, 1=UTRAN, 2=GERAN, 3=NB-IoT	V
0x3011	LTE SYSTEM INFORMATION SCHEDULE Request	struct	The LTE SYSTEM INFORMATION SCHEDULE request information as defined in Table 5-37	TLV
0x3010	PNF CELL BROADCAST STATE	struct	A struct of opaque data conditionally sent by the VNF to the PNF. It must be included if the PNF included it in the `BROADCAST DETECT.Indication` for this cell.	TLV
0x3025	NB-IoT SYSTEM INFORMATION SCHEDULE Request	struct	The NB-IoT SYSTEM INFORMATION SCHEDULE request information as defined in Table 5-38	TLV

Table 5-36 SYSTEM INFORMATION SCHEDULE.request parameters

LTE SYSTEM INFORMATION SCHEDULE Request

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION SCHEDULE Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN of the cell for which the SIB1 is wanted
PCI	uint16_t	PCI of the cell for which the SIB1 is wanted
Downlink channel bandwidth	uint16_t	Downlink channel bandwidth (in resource blocks) of the cell for which SIB1 is wanted. See [10] section 5.6. Value: 6, 15, 25, 50, 75, 100
PHICH Configuration	uint8_t	The PHICH-Config of the cell, mapped as:

Field	Type	Description
		Bits 7-4: Reserved (0) Bit 3: PHICH Duration, encoded as: 0=Normal, 1=Extended Bits 1-0: PHICH Resource, encoded as: 0=1/6; 1=1/2; 2=1, 3=2
Number of Tx Antenna	uint8_t	The number of Tx Antenna for the cell: 1, 2 or 4
retryCount	uint8_t	The number of SIB1 repetition periods for which decoding of SIB1 should be retried.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-37 LTE SYSTEM INFORMATION SCHEDULE Request

NB-IoT SYSTEM INFORMATION SCHEDULE Request

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION SCHEDULE Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN for the NB-IoT carrier for which the NMM should read the NPBCH.
RO_{DL}	uint8_8	Raster offset for the carrier Value: 0 = -7.5KHz 1 = -2.5KHz 2 = 0 (aligned to carrier raster) 3 = +2.5KHz 4 = +7.5KHz
PCI	uint16_t	PCI of the cell for which the SIB1-NB is wanted
SchedulingInfoSIB1-NB	uint8_t	Indicates the number of NPDSCH repetitions and TBS value for SIB1-NB. This field is encoded as specified in [9]. Value: 0 → 15.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-38 NB-IoT SYSTEM INFORMATION SCHEDULE Request

SYSTEM INFORMATION SCHEDULE.response

The `SYSTEM INFORMATION SCHEDULE.response` message is sent by the PNF on receipt of a `SYSTEM INFORMATION SCHEDULE request`. If the error code indicates no error, the PNF shall start the requested procedure, and shall send the `SYSTEM INFORMATION SCHEDULE.indication` once the procedure is complete.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-39 SYSTEM INFORMATION SCHEDULE.response parameters

SYSTEM INFORMATION SCHEDULE.response errors

The error codes that can be returned in `SYSTEM INFORMATION SCHEDULE.response` are given in Table 5-40.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.

Table 5-40 Error codes for SYSTEM INFORMATION SCHEDULE`.response`

SYSTEM INFORMATION SCHEDULE.indication

The `SYSTEM INFORMATION SCHEDULE.indication` message is sent by the PNF on completion of the SYSTEM INFORMATION SCHEDULE procedure, and conveys the results of the SYSTEM INFORMATION SCHEDULE procedure to the VNF.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x3018	LTE SYSTEM INFORMATION Indication	struct	The LTE SYSTEM INFORMATION as defined in Table 5-51, which for this message carries the SIB1	TLV
0x3026	NB-IoT SYSTEM INFORMATION Indication	struct	The NB-IoT SYSTEM INFORMATION Indication as defined in Table 5-54	TLV

Table 5-41 SYSTEM INFORMATION SCHEDULE.indication parameters

SYSTEM INFORMATION SCHEDULE.indication errors

The error codes that can be returned in `SYSTEM INFORMATION SCHEDULE.indication` are given in Table 5-42.

Error code	Description
MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an

Error code	Description
	NMM STOP.request and the results so far collected, if any, are included in the indication.
MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-42 Error codes for SYSTEM INFORMATION SCHEDULE.indication

5.3.3.5 SYSTEM INFORMATION

SYSTEM INFORMATION.request

The `SYSTEM INFORMATION.request message` is sent by the VNF to request that the PNF performs a SYSTEM INFORMATION measurement on the requested cell. Only one of the RAT-specific structures shall be included in the message.

Tag	Field	Type	Description	nFAPI
N/A	RAT Type	uint_8	RAT Type of this request, which specifies which of the possible TLV structures is included below: 0=LTE, 1=UTRAN, 2=GERAN, 3=NB-IoT	V
0x3014	LTE SYSTEM INFORMATION Request	struct	The LTE SYSTEM INFORMATION request information as defined in Table 5-44	TLV
0x3015	UTRAN SYSTEM INFORMATION Request	struct	The UTRAN SYSTEM INFORMATION request information as defined in Table 5-45	TLV
0x3016	GERAN SYSTEM INFORMATION Request	struct	The GERAN SYSTEM INFORMATION request information as defined in Table 5-46	TLV
0x3010	PNF CELL BROADCAST STATE	struct	A structure of opaque data conditionally sent by the VNF to the PNF. It must be included if the PNF included it in the BROADCAST DETECT.Indication for this cell.	TLV
0x3027	NB-IoT SYSTEM INFORMATION Request	struct	The NB-IoT SYSTEM INFORMATION Request information as defined in Table 5-47	TLV

Table 5-43 SYSTEM INFORMATION.request parameters

LTE SYSTEM INFORMATION Request

Field	Type	Description
Tag	uint16_t	The SYSTEM INFORMATION Request Tag
Length	uint16_t	The length in bytes of the values
EARFCN	uint16_t	The EARFCN of the cell for which the SIBs are wanted

Field	Type	Description
PCI	uint16_t	PCI of the cell for which the SIBs are wanted
Downlink channel bandwidth	uint16_t	Downlink channel bandwidth (in resource blocks) of the cell for which SIBs are wanted. See [10] section 5.6. Value: 6, 15, 25, 50, 75, 100
PHICH Configuration	Uint8_t	The PHICH-Config of the cell, mapped as: Bits 7-4: Reserved (0) Bit 3: PHICH Duration, encoded as: 0=Normal, 1=Extended Bits 1-0: PHICH Resource, encoded as: 0=1/6; 1=1/2; 2=1, 3=2
Number of Tx Antenna	uint8_t	The number of Tx Antenna for the cell: 1, 2 or 4
Number of SI Periodicity	uint8_t	The number of System Information periodicity values in the following array
For (Number of SI Periodicity) {		
SI Periodicity	uint8_t	The SI Periodicity of the requested SIBs, with the first element being for SIB2, the next for SIB3, etc, encoded as follows: 0=This SIB not required 1=80ms 2=160ms 3=320ms 4=640ms 5=1280ms 6=2560ms 7=5120ms
SI Index	uint8_t	The index of this SIB in the SIB1 SchedulingInfoList: 0 = This SIB not required 1…32 = index value in SIB1 SchedulingInfoList
}		
SI Window Length	unit8_t	The SI window in units of 1ms, with the following valid values: 1,2,5,10,15,20,40
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-44 LTE SYSTEM INFORMATION Request

UTRAN SYSTEM INFORMATION Request

Field	Type	Description
Tag	uint16_t	The SYSTEM INFORMATION Request Tag
Length	uint16_t	The length in bytes of the values

Field	Type	Description
UARFCN	uint16_t	The UARFCN for the cell for which the SIBs are wanted.
PSC	uint16_t	The primary scrambling code of the cell for which the SIBs are wanted.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-45 UTRAN SYSTEM INFORMATION Request

GERAN SYSTEM INFORMATION Request

Field	Type	Description
Tag	uint16_t	The SYSTEM INFORMATION Request Tag
Length	uint16_t	The length in bytes of the values
ARFCN	uint16_t	The ARFCN of the cell for which the SIs are wanted.
BSIC	uint8_t	The BSIC of the cell for which the SIs are wanted.
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-46 GERAN SYSTEM INFORMATION Request

NB-IoT SYSTEM INFORMATION Request

Field		Type	Description
Tag		uint16_t	The SYSTEM INFORMATION Request Tag
Length		uint16_t	The length in bytes of the values
EARFCN		uint16_t	The EARFCN for the NB-IoT carrier for which the NMM should read the NPBCH.
RO_{DL}		uint8_8	Raster offset for the carrier Value: 0 = -7.5KHz 1 = -2.5KHz 2 = 0 (aligned to carrier raster) 3 = +2.5KHz 4 = +7.5KHz
PCI		uint16_t	PCI of the cell for which the SIBs are wanted
Number of SI Periodicity		uint8_t	The number of System Information periodicity values in the following array
For (Number of SI Periodicity) {			
	SI Periodicity	uint8_t	The SI Periodicity of the requested SIBs, with the first element being for SIB2-NB, the next for SIB3-NB, etc, encoded as follows:

Field	Type	Description
		0=640ms 1=1280ms 2=2560ms 3=5120ms 4=10240ms 5=20480ms 6=40960ms
SI RepetitionPattern	uint8_t	The repetition pattern of the requested SIBs, with the first element being for SIB2-NB, next for SIB3-NB etc, encoded as follows: 0=every2ndRF 1=every4thRF 2=every8thRF 3=every16thRF
SI TB Size	uint8_t	This gives the transport block size of the System Information messages and is encoded as follows: 0=56bits 1=120bits 2=208bits 3=256bits 4=328bits 5=440bits 6=552bits 7=680bits
Number of SI Index	uint8_t	Number of SIBs associated with a particular SI periodicity Value: 0 → 7.
For (Number of SI Index) {		
SI Index	uint8_t	Index of this SIB type in SIB-Mapping-Info-NB, encoded as follows: 0 = SIB3-NB 1 = SIB4-NB 2 = SIB5-NB 3 = SIB14-NB 4 = SIB16-NB
}		
}		
SI Window Length	unit8_t	The SI window encoded as follows: 0=160ms 1=320ms 2=480ms 3=640ms 4=960ms 5=1280ms 6=1600ms

Field	Type	Description
Timeout	uint32_t	The timeout value after which the PNF should abort the procedure in units of 1ms. The value of 0 indicates that the PNF should attempt to complete the procedure without any VNF-imposed timeout.

Table 5-47 NB-IoT SYSTEM INFORMATION Request

SYSTEM INFORMATION.response

The `SYSTEM INFORMATION.response` message is sent by the PNF on receipt of n SYSTEM INFORMATION request. If the error code indicates no error, the PNF shall start the requested procedure, and shall send the `SYSTEM INFORMATION.indication` once the procedure is complete.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-48 SYSTEM INFORMATION.response parameters

SYSTEM INFORMATION.response errors

The error codes that can be returned in `SYSTEM INFORMATION.response` are given in Table 5-49.

Error code	Description
MSG_OK	Message is OK.
MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.

Table 5-49 Error codes for SYSTEM INFORMATION.response

SYSTEM INFORMATION.indication

The `SYSTEM INFORMATION.indication` message is sent by the PNF during and on completion of the SYSTEM INFORMATION procedure, and conveys the results of the SYSTEM INFORMATION procedure to the VNF. The final indication of the procedure may omit the SYSTEM INFORMATION Indication. Only one of the RAT-specific structures shall be included in the message.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V
0x3018	LTE SYSTEM INFORMATION Indication	struct	The LTE SYSTEM INFORMATION indication information as defined in Table 5-51	TLV

Tag	Field	Type	Description	nFAPI
0x3019	UTRAN SYSTEM INFORMATION Indication	struct	The UTRAN SYSTEM INFORMATION indication information as defined in Table 5-52	TLV
0x301a	GERAN SYSTEM INFORMATION Indication	struct	The GERAN SYSTEM INFORMATION indication information as defined in Table 5-53	TLV
0x3026	NB-IoT SYSTEM INFORMATION Indication	struct	The NB-IoT SYSTEM INFORMATION indication information as defined in Table 5-54	TLV

Table 5-50 SYSTEM INFORMATION.indication parameters

LTE SYSTEM INFORMATION Indication

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION Indication Tag
Length	uint16_t	The length in bytes of the values
SIB Type	uint8_t	The SIB type carried in the array below. LTE Values: 0=MIB; 1=SIB1; 2=SIB2 etc
SIB Length	uint16_t	The length in bytes of the following SIB array
SIB[SIB Length]	array of uint8_t	The SIB element read from the specified cell.

Table 5-51 LTE SYSTEM INFORMATION Indication

UTRAN SYSTEM INFORMATION Indication

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION Indication Tag
Length	uint16_t	The length in bytes of the values
SIB Length	uint16_t	The length in bytes of the following SIB array
SIB[SIB Length]	array of uint8_t	The SIB element read from the specified cell.

Table 5-52 UTRAN SYSTEM INFORMATION Indication

GERAN SYSTEM INFORMATION Indication

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION Indication Tag
Length	uint16_t	The length in bytes of the values
SI Length	uint16_t	The length in bytes of the following SI array
SI[SI Length]	array of uint8_t	The SI element read from the specified cell.

Table 5-53 GERAN SYSTEM INFORMATION Indication

NB-IoT SYSTEM INFORMATION Indication

Field	Type	Description
Tag	uint16_t	The LTE SYSTEM INFORMATION Indication Tag
Length	uint16_t	The length in bytes of the values
SIB Type	uint8_t	The SIB type carried in the array below and encoded as follows. Values: 0=MIB-NB 1=SIB1-NB 2=SIB2-NB 3=SIB3-NB 4=SIB4-NB 5-13 = Reserved 14=SIB14-NB 15=Reserved 16=SIB16-NB
SIB Length	uint16_t	The length in bytes of the following SIB array
SIB[SIB Length]	array of uint8_t	The SIB element read from the specified cell.

Table 5-54 NB-IoT SYSTEM INFORMATION Indication

SYSTEM INFORMATION.indication errors

The error codes that can be returned in `SYSTEM INFORMATION.indication` are given in Table 5-55.

Error code	Description
MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
MSG_PARTIAL_RESULTS	The requested procedure has produced the partial results included, but the procedure is continuing and further indications will follow.
MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an NMM STOP.request and the results so far collected, if any, are included in the indication.
MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-55 Error codes for SYSTEM INFORMATION`.indication`

5.3.3.6 NMM Stop

NMM Stop.request

The NMM Stop.request message may be sent by the VNF to stop any ongoing NMM procedure.

No message body is defined for the NMM Stop request.

NMM Stop.response

The `NMM Stop.response` message is sent by the PNF on receipt of an NMM Stop request message. The PNF may emit an NMM procedure indication in advance of the NMM Stop response, containing partial or full results from the NMM procedure. Since the NMM Stop request from the VNF may cross with the NMM procedure indication from the PNF at the normal termination of the NMM procedure, it is not an error for the NMM Stop request to arrive after completion of the NMM procedure at the PNF.

Tag	Field	Type	Description	nFAPI
N/A	Error Code	uint32_t	The error code	V

Table 5-56 NMM Stop.response parameters

NMM STOP.response errors

The error codes that can be returned in `NMM STOP.response` are given in Table 5-57.

Error code	Description
MSG_NMM_STOP_OK	The NMM STOP.request caused an NMM procedure to be stopped.
MSG_NMM_STOP_IGNORED	The NMM STOP.request was received when no NMM procedure was in progress, so was ignored.
MSG_NMM_STOP_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM.

Table 5-57 Error codes for NMM STOP`.response`

5.3.4 NMM nFAPI Tag Values

The tag values used in the TLV sections of nFAPI P7 messages are given in

0x301b – 0x301f	Reserved
0x3020	NB-IoT RSSI Request
0x3021	NB-IoT CELL SEARCH Request
0x3022	NB-IoT CELL SEARCH Indication
0x3023	NB-IoT BROADCAST DETECT Request
0x3024	NB-IoT BROADCAST DETECT Indication
0x3025	NB-IoT SYSTEM INFORMATION SCHEDULE Request
0x3026	NB-IoT SYSTEM INFORMATION Indication
0x3027	NB-IoT SYSTEM INFORMATION Request

Table 5-58.

Tag (Hex)	Description
0x3000	LTE RSSI Request
0x3001	UTRAN RSSI Request

0x3002	GERAN RSSI Request
0x3003	RSSI Indication
0x3004	LTE CELL SEARCH Request
0x3005	UTRAN CELL SEARCH Request
0x3006	GERAN CELL SEARCH Request
0x3007	LTE CELL SEARCH Indication
0x3008	UTRAN CELL SEARCH Indication
0x3009	GERAN CELL SEARCH Indication
0x300a	PNF CELL SEARCH STATE
0x300b	LTE BROADCAST DETECT Request
0x300c	UTRAN BROADCAST DETECT Request
0x300e	LTE BROADCAST DETECT Indication
0x300f	UTRAN BROADCAST DETECT Indication
0x3010	PNF CELL BROADCAST STATE
0x3011	LTE SYSTEM INFORMATION SCHEDULE Request
0x3014	LTE SYSTEM INFORMATION Request
0x3015	UTRAN SYSTEM INFORMATION Request
0x3016	GERAN SYSTEM INFORMATION Request
0x3018	LTE SYSTEM INFORMATION Indication
0x3019	UTRAN SYSTEM INFORMATION Indication
0x301a	GERAN SYSTEM INFORMATION Indication
0x301b – 0x301f	Reserved
0x3020	NB-IoT RSSI Request
0x3021	NB-IoT CELL SEARCH Request
0x3022	NB-IoT CELL SEARCH Indication
0x3023	NB-IoT BROADCAST DETECT Request
0x3024	NB-IoT BROADCAST DETECT Indication
0x3025	NB-IoT SYSTEM INFORMATION SCHEDULE Request
0x3026	NB-IoT SYSTEM INFORMATION Indication
0x3027	NB-IoT SYSTEM INFORMATION Request

Table 5-58 nFAPI NMM (P4) tag values

5.3.5 Error codes

The list of possible error codes returned in either `.response` messages or `.indication` messages is given in Table 5-59.

Value	Error code	Description
100	MSG_OK	Message is OK.
101	MSG_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM, or another NMM procedure was already in progress.
102	MSG_INVALID_CONFIG	The request parameters provided has missing mandatory TLVs, or TLVs that are invalid or unsupported in this PHY state.
103	MSG_RAT_NOT_SUPPORTED	The request was for a RAT not supported by this PNF PHY instance.
200	MSG_NMM_STOP_OK	The NMM STOP.request caused an NMM procedure to be stopped.
201	MSG_NMM_STOP_IGNORED	The NMM STOP.request was received when no NMM procedure was in progress, so was ignored.
202	MSG_NMM_STOP_INVALID_STATE	The request was received when the PHY was in the IDLE state, or in the RUNNING state and this PNF does not support RUNNING state NMM.
300	MSG_PROCEDURE_COMPLETE	The requested procedure completed normally and the results, if any, are included in the indication.
301	MSG_PROCEDURE_STOPPED	The requested procedure was stopped by an NMM STOP.request and the results so far collected, if any, are included in the indication.
302	MSG_PARTIAL_RESULTS	The requested procedure has produced the partial results included, but the procedure is continuing and further indications will follow.
303	MSG_TIMEOUT	The procedure timed out, and partial or no results are included in the indication.

Table 5-59 NMM error codes

5.4 Constants

Constant	Usage	Value
MAX_CARRIER_LIST	Maximum number of carriers present in a single NMM message	32
MAX_PCI_LIST	Maximum number of cell PCIs present in a single NMM message	32
MAX_PSC_LIST	Maximum number of cells PSCs present in a single NMM message	32
MAX_ARFCN_LIST	Maximum number of ARFCN present in a single NMM message	128

Table 5-60 Constants used for NMM message sizing

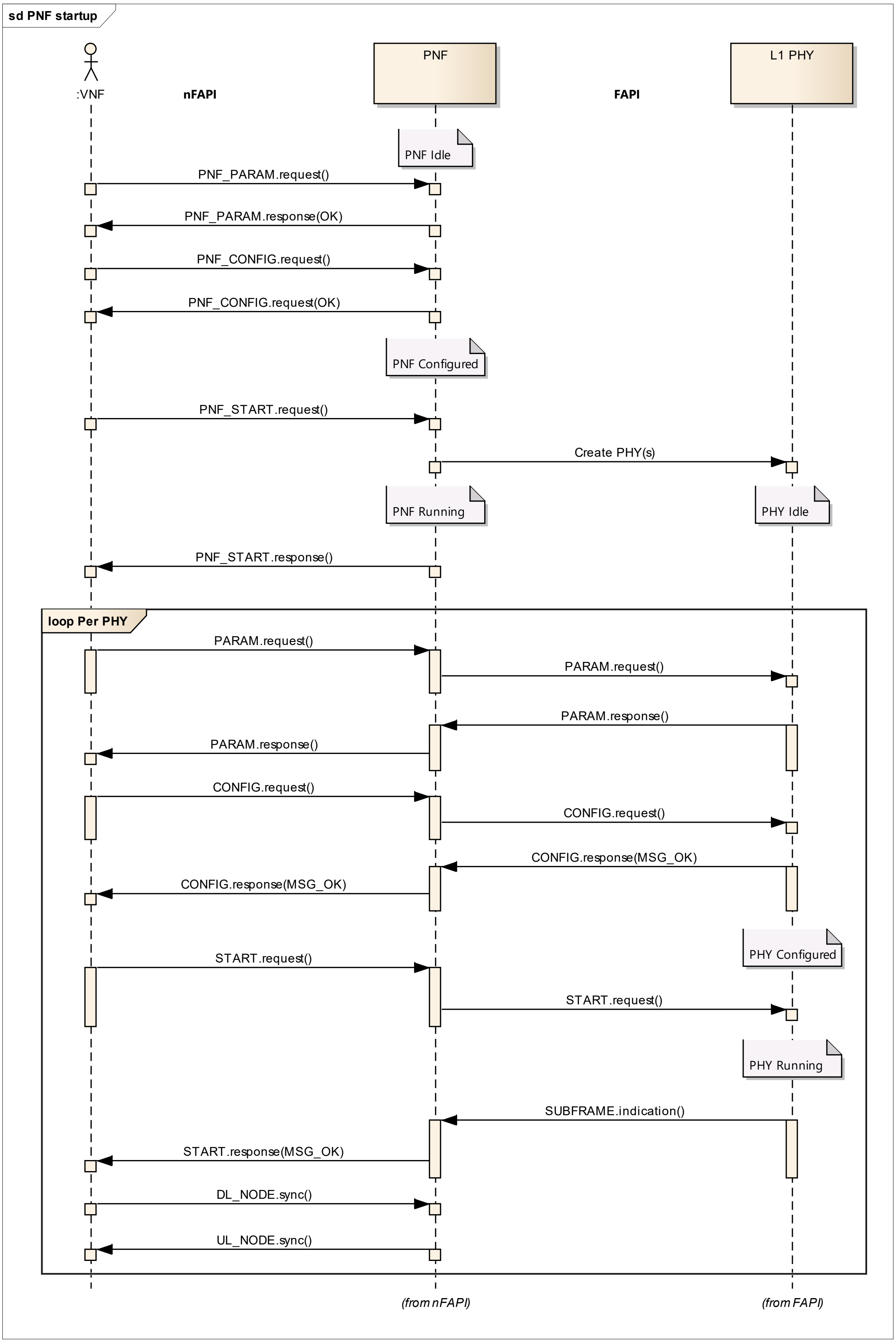
Signalling Appendix

This appendix provides a normative description of how the nFAPI and FAPI interfaces interact for key procedures.

PNF Start

The normative PNF Start procedure is shown below and can be summarized by:

- The nFAPI PNF initiailization procedure is followed.
  - This includes the nFAPI PNF PARAM procedure
  - Followed by the nFAPI PNF CONFIG procedure
  - And the nFAPI PNF START procedure, including (if required by the PNF) any actions to create PHY instances
- The nFAPI PNF initialization procedure completes when the VNF receives the `PNF_START.response` message.
- The VNF follows the nFAPI PHY initialization procedure for each PHY
  - This includes the nFAPI PHY PARAM procedure;
    - Where the nFAPI `PARAM.request` triggers the FAPI `PARAM.request` message.
    - And the FAPI `PARAM.response` triggers the nFAPI `PARAM.response` message.
  - Followed by the nFAPI PHY CONFIG procedure;
    - Where the nFAPI `CONFIG.request` triggers the FAPI `CONFIG.request` message
    - And the FAPI CONFIG.response triggers the nFAPI `CONFIG.response` message
  - And the nFAPI PNF START procedure;
    - Where the nFAPI `START.request` triggers the FAPI `START.request` message
    - The FAPI START procedure completes when the PHY transmits the first `SUBFRAME.indication` message, this triggers the nFAPI `START.response` message
- Following the nFAPI PHY initialization procedure the nFAPI PHY Synchronization procedure is followed.
  - This includes the `DL_NODE.sync` and `UL_NODE.sync` messages

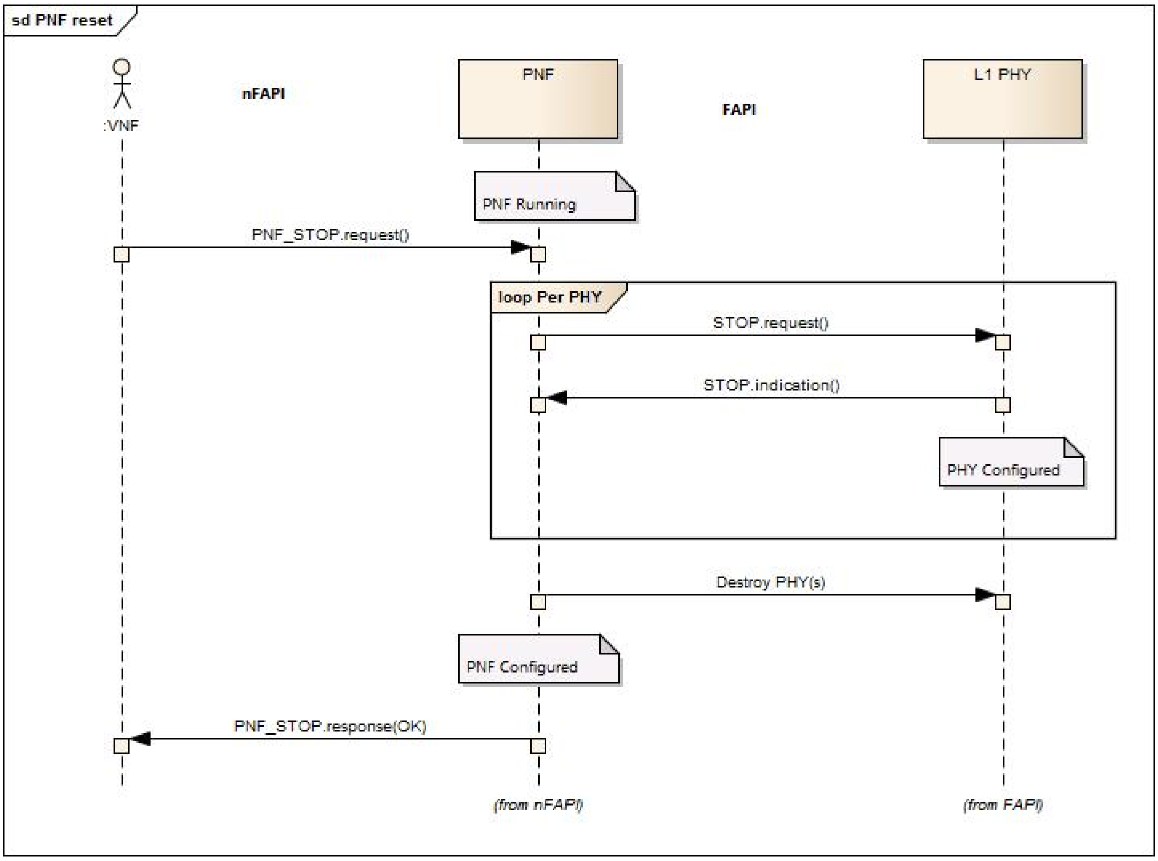


sd PNF startup
:VNF
nFAPI
PNF
FAPI
L1 PHY
PNF Idle
PNF_PARAM.request()
PNF_PARAM.response(OK)
PNF_CONFIG.request()
PNF_CONFIG.request(OK)
PNF Configured
PNF_START.request()
Create PHY(s)
PNF Running
PHY Idle
PNF_START.response()
loop Per PHY
PARAM.request()
PARAM.request()
PARAM.response()
PARAM.response()
CONFIG.request()
CONFIG.request()
CONFIG.response(MSG_OK)
CONFIG.response(MSG_OK)
PHY Configured
START.request()
START.request()
PHY Running
SUBFRAME.indication()
START.response(MSG_OK)
DL_NODE.sync()
UL_NODE.sync()
(from nFAPI)
(from FAPI)

PNF Reset

The normative PNF Reset procedure is shown below and can be summarized by:

- The VNF follows the nFAPI PNF reset procedure
  - On receiving the `PNF_STOP.request` the PNF performs the FAPI PHY reset procedure for each PHY.
  - For each PHY the FAPI PHY reset procedure completes when a `STOP.indication` message is received from the PHY. If required the PNF will destroy the PHY instance.
  - When all PHYs have been reset the PNF returns the `PNF_STOP.response` message to the VNF



Other PNF Interfaces (Informative)

Although the other interfaces defined for the PNF are outside the scope of this nFAPI specification, this section describes the intent of these interfaces, and provides information about some of the possible ways in which they may be implemented.

Service Discovery (P2)

This interface is intended to provide the initial provisioning of the fundamental parameters needed by the PNF in order to obtain the normal OAM capabilities, and this is provisioned to provide service to the VNF(s). This initial provisioning includes parameters such as:

- The IP configuration of the PNF IP interface (i.e. whether provided dynamically through DHCP, or if not, the static IP configuration, such as IP address, Netmask, default gateway, DNS address, etc.)
- The FQDN or IP address of the OAM server for the PNF to contact for service provisioning
- Security related parameters, such as Security Gateway server FQDN or IP address

The methods by which a PNF obtains such information may be varied, depending on the deployment scenario for the PNF, operator choice, vendor capabilities etc. There is not a single method which can satisfy all such scenarios. This diversity of requirements results in a diversity of common solutions for this interface, some of which are listed here for information:

- **Web pages on the PNF.** The PNF may start in a special mode in which it adopts a fixed IP address, and allows a local PC connected to it to access a web server running on the PNF which provides the user with the capability to perform the initial provisioning. It may be possible to re-enter this mode through a "factory restore" procedure on the PNF.
- **Initial Provisioning Server.** The PNF may start in a special mode in which it contacts an Initial Provisioning Server using a pre-provisioned server address, and using its serial number or other pre-provisioning identity information obtains the correct initial provisioning information from the server. It may be possible to re-enter this mode through a "factory restore" procedure on the PNF.
- **SIM, smart card, USB disk or memory card.** The PNF may have a physical interface into which an external memory device can be plugged, and the PNF obtains its initial provisioning data from it.

- **Initial Provisioning Interface.** The PNF may have a network (e.g. UDP port responding to broadcasts) or other radio based interface (e.g. Bluetooth) specifically for initial provisioning, to which an application running on a PC, tablet or smart phone can connect, and perform the initial provisioning.

Positioning (P3)

This interface is intended to provide positioning information of the PNF to the VNF to enable it to perform SON activities, to provide the positioning information onwards to the EPC for use in Location Services, or to provide it in support of emergency service location requirements. How the PNF obtains this position information is outside the scope of this specification, but may include:

- GNSS (e.g. GPS)

- Manual provisioning through OAM (P9)

There are many possible implementations of this interface. A basic implementation is already subsumed into the P5 interface of this nFAPI specification to provide a Location Coordinates parameter to the VNF using LocationCoordinates IE of [17].

A more comprehensive implementation may include the carrying of the standard [17] message set through vendor extensions so that the PNF may report location in exactly the same manner as it is obtained from a UE, including e.g. the options to include assistance data to the PNF for its positioning methods.

PHY Diagnostics (P8)
This interface is intended to allow a PNF vendor to provide additional diagnostic modes and capabilities to enable the internal operation of the PNF to be analysed in detail for diagnostic purposes. Such diagnostic modes may include the need for relatively large amounts or high bandwidth of data, and may need to be continuously operative during normal PNF operation. It is expected that these capabilities could be provided in a number of ways, including e.g.

- Vendor extensions in OAM to provide small volumes of low bandwidth data, or to provision additional proprietary diagnostic interfaces to the PNF
- Vendor extensions to any or all of P4, P5 and P7 of nFAPI to:
    - Provide additional control and information using the Vendor Extensions of existing nFAPI messages
    - Provide additional Vendor-specific messages over the same nFAPI message streams

It is suggested that any such Vendor Extensions should follow the practices and format of the messages and procedures defined for nFAPI.

OAM (P9)
This interface is intended to provide OAM services to the PNF. Due to the similarity in requirements for the OAM of the PNF compared to those of an HeNB, it is expected that the PNF will be managed using TR-069, using a data model based on a subset of TR-196, together with the other standard device data models specified for HeNBs by 3GPP and the BBF.

Synchronization, frequency, phase and time (P10)
This interface is intended to provide the PNF with the frequency, phase and time synchronisation it needs for normal operation. Those requirements may vary considerably depending on the services the PNF is providing, e.g.

- An LTE FDD PNF which only provides Release 9 capabilties may be deployed in a network with its only synchronisation requirement being to meet the 3GPP frequency accuracy requirements for its transmitter.
- An LTE TDD PNF may be deployed in a network which requires all cell to the frame aligned to within a few microseconds at all times.

There are many methods by which such frequency and time synchronisation may be provided, such as:

- GNSS (e.g. GPS)
- IEEE1588 or NTP
- SyncE
- High accuracy internal oscillator

- Synchronisation using NMM over the P4 interface to other cells

The environments in which the PNF is to be deployed may also affect considerably the ease and cost by which any possible such solution is implemented, and as such there is no single solution that is suitable for all PNFs. More detailed information on the requirements and solution options may be found in [21].

Transport and nFAPI Message Loss

P4 / P5

The SCTP association is initialized from the PNF towards the VNF using the end point configuration defined over P9. There is a bidirectional (incoming plus outgoing) stream for the PNF, each PHY and if required the NMM PHY for P4. Once the streams are established, the nFAPI procedures can occur between the P5 and P4 applications of the PNF and VNF.

If either side decides to perform a shutdown of the SCTP association, the messages in flight within each stream are delivered and then all streams are closed and the application layer must handle the recovery. This should result in a cleanup of application contexts associated with the SCTP association and return to the pre-connected state. For the PNF this should return to a re-connection attempt or raise awareness of the status through the P9 interface. For the VNF, the PNF's resources should be cleared and return to a pre-connected state – which is VNF implementation specific and out of scope for this document.

If either side decides to perform an abort of the SCTP association, the procedure for handling at the application layer is the same as the shutdown procedure. The only difference being that the messages pending transmission are discarded, undelivered.

If communication with the end point is lost (e.g. through time out of heartbeat) then the same procedure as abort / shutdown for handling at the application layer is required.

Further details and example application / protocol procedures are defined in RFC 4960 [18].

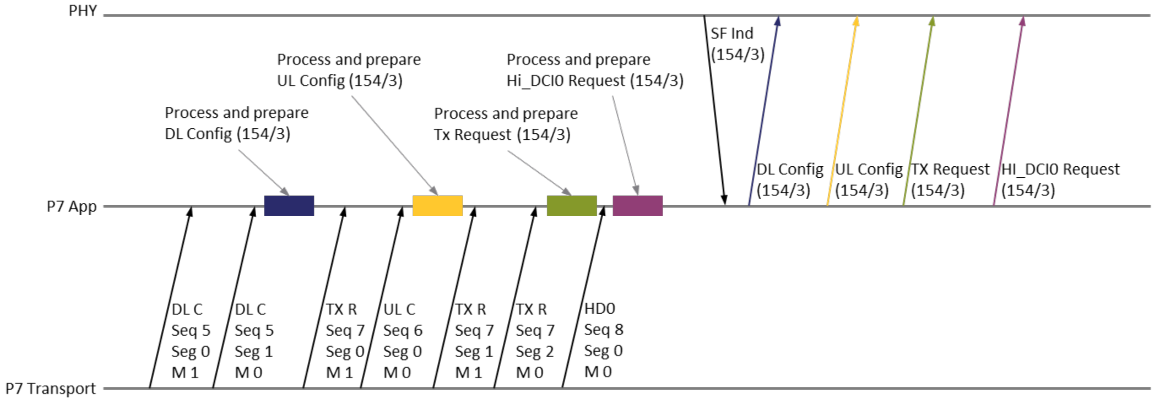
P7

UDP being a connectionless protocol relies on the application layer to handle lost and out of order messages. To this end the P7 protocol header defines segments and sequence numbers on a per PHY instance to permit the nFAPI P7 application to handle such occurances.

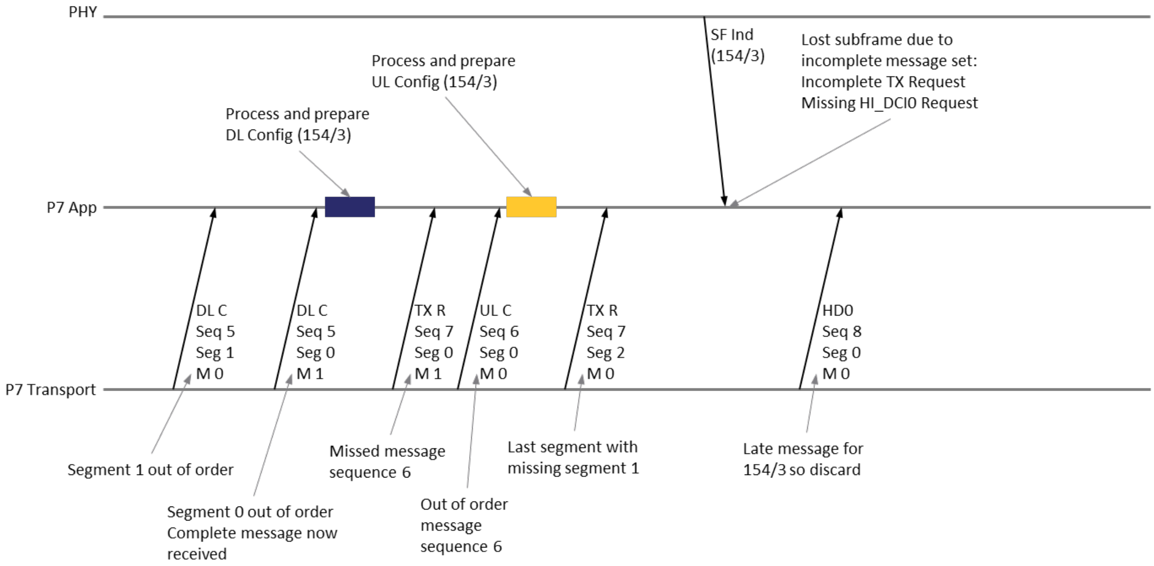
When the receiver on the P7 interface recognises missed or out of order messages, it is the responsibility of the receiving entity to handle such situations. This means for the PNF within subframe procedures, the PNF can buffer and play-out the received messages, handling any out of order or segmented messages that require some extra time to receive and re-constitute. There will be timing constraints associated with this process, which may conclude in the PNF discarding the subframe and resuming on the subsequent subframe.

Examples of message flow between transport, nFAPI P7 and FAPI interface are shown below. First a successful sequence with resultant DL message flow through FAPI. Secondly a failure sequence with out of order messages, segments and missing segments, messages.

P7 message sequence (Success)



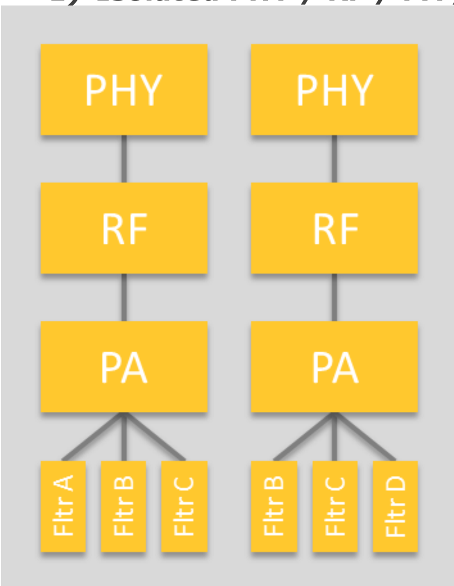
P7 message sequence (Late / Missing Segment)



PNF PHY and RF Configuration Options

This appendix presents examples of PHY and RF configurations together with their respective physical architectures. This is intended to show the relationship between PNF, PHY and RF data sets of the P5 interface and how they interact with each other. The list of examples is not exhaustive, but aims to capture the options available within the interface.

1) *Isolated PHY / RF / PA / Filters*



DL Config Timing			-
TX Timing			-
UL Config Timing			-
HI_DCI0 Timing			-
Maximum Number PHYs			2
Maximum Total Bandwidth			40
Maximum DL Layers			2
Maximum UL Layers			2
Shared Bands			No
Shared PA			No
Maximum Total Power			N/A
Internal PNF frame alignment			Yes
PHY Config List			
	1		
		Bandwidth List	5,10,15,20
		Band List	1,2,3 (idx into RF Cfg List)
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
	2		
		Bandwidth List	5,10,15,20
		Band List	4,5,6 (idx into RF Cfg List)
		Band Exclusion List	-

Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16

RF Config List

1

Band	7
Maximum Power	24
Maximum Antennas	2
Minimum DL Frequency	2620
Maximum DL Frequency	2690
Minimum UL Frequency	2500
Maximum UL Frequency	2570

2

Band	4
Maximum Power	24
Maximum Antennas	2
Minimum DL Frequency	2110
Maximum DL Frequency	2155
Minimum UL Frequency	1710
Maximum UL Frequency	1755

3

Band	41
Maximum Power	20
Maximum Antennas	1
Minimum DL Frequency	2496
Maximum DL Frequency	2690
Minimum UL Frequency	2496
Maximum UL Frequency	2690

4

Band	4
Maximum Power	24
Maximum Antennas	2
Minimum DL Frequency	2110
Maximum DL Frequency	2155
Minimum UL Frequency	1710
Maximum UL Frequency	1755

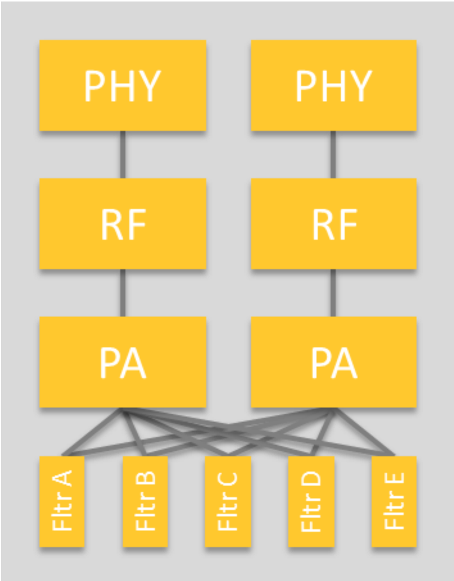
5

Band	41
Maximum Power	20
Maximum Antennas	1
Minimum DL Frequency	2496
Maximum DL Frequency	2690
Minimum UL Frequency	2496
Maximum UL Frequency	2690

6

Band	12
Maximum Power	30
Maximum Antennas	2
Minimum DL Frequency	729
Maximum DL Frequency	746
Minimum UL Frequency	699
Maximum UL Frequency	716

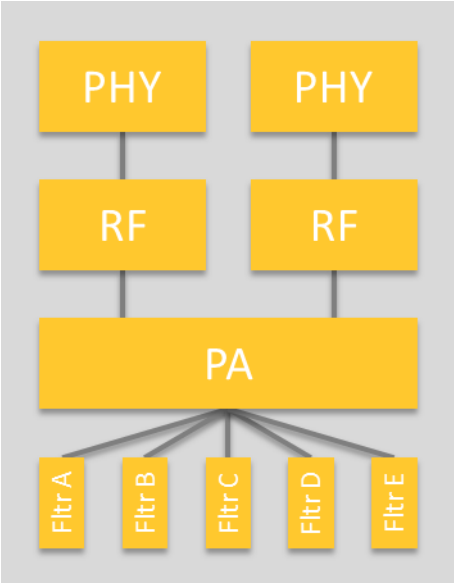
2) *Isolated PHY / RF / PA, Shared Filters*



DL Config Timing			-
TX Timing			-
UL Config Timing			-
HI_DCI0 Timing			-
Maximum Number PHYs			2
Maximum Total Bandwidth			40
Maximum DL Layers			2
Maximum UL Layers			2
Shared Bands			Yes
Shared PA			No
Maximum Total Power			N/A
Internal PNF frame alignment			Yes
PHY Config List			
	1		
		Bandwidth List	5,10,15,20
		Band List	1,2,3,4,5
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
	2		
		Bandwidth List	5,10,15,20
		Band List	1,2,3,4,5
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
RF Config List			
	1		
		Band	7
		Maximum Power	24

	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
2		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
3		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
4		
	Band	12
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716
5		
	Band	3
	Maximum Power	24
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

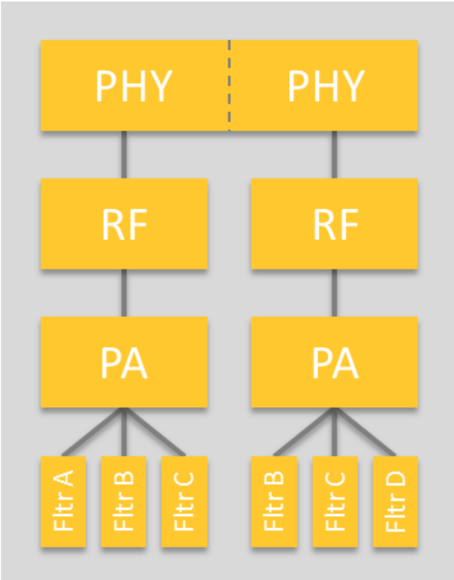
3) Isolated PHY / RF, Shared PA / Filters



DL Config Timing	-
TX Timing	-
UL Config Timing	-
HI_DCI0 Timing	-
Maximum Number PHYs	2
Maximum Total Bandwidth	40
Maximum DL Layers	2
Maximum UL Layers	2
Shared Bands	Yes
Shared PA	Yes
Maximum Total Power	30
Internal PNF frame alignment	Yes
PHY Config List	
1	
Bandwidth List	5,10,15,20
Band List	1,2,3,4,5
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
2	
Bandwidth List	5,10,15,20
Band List	1,2,3,4,5
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
RF Config List	
1	
Band	7
Maximum Power	24

	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
2		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
3		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
4		
	Band	12
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716
5		
	Band	3
	Maximum Power	24
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

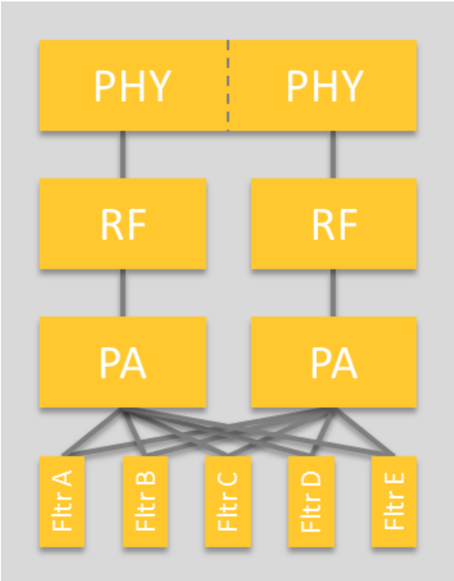
4) Shared PHY, Isolated RF / PA / Filters



DL Config Timing	-
TX Timing	-
UL Config Timing	-
HI_DCI0 Timing	-
Maximum Number PHYs	2
Maximum Total Bandwidth	30
Maximum DL Layers	2
Maximum UL Layers	2
Shared Bands	No
Shared PA	No
Maximum Total Power	N/A
Internal PNF frame alignment	Yes
PHY Config List	
1	
Bandwidth List	5,10,15,20
Band List	1,2,3
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
2	
Bandwidth List	5,10,15,20
Band List	4,5,6
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
RF Config List	
1	
Band	7
Maximum Power	24

	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
2		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
3		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
4		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
5		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
6		
	Band	12
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716

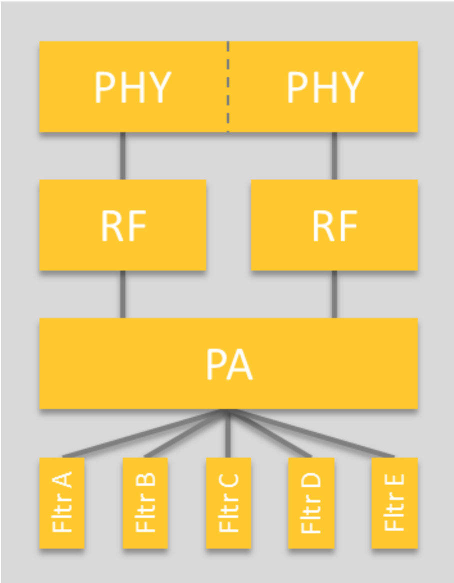
5) Shared PHY / Filters, Isolated RF / PA



DL Config Timing			-
TX Timing			-
UL Config Timing			-
HI_DCI0 Timing			-
Maximum Number PHYs			2
Maximum Total Bandwidth			30
Maximum DL Layers			2
Maximum UL Layers			2
Shared Bands			Yes
Shared PA			No
Maximum Total Power			N/A
Internal PNF frame alignment			Yes
PHY Config List			
	1		
		Bandwidth List	5,10,15,20
		Band List	1,2,3,4,5
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
	2		
		Bandwidth List	5,10,15,20
		Band List	1,2,3,4,5
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
RF Config List			
	1		
		Band	7
		Maximum Power	24

	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
2		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
3		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
4		
	Band	12
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716
5		
	Band	3
	Maximum Power	24
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

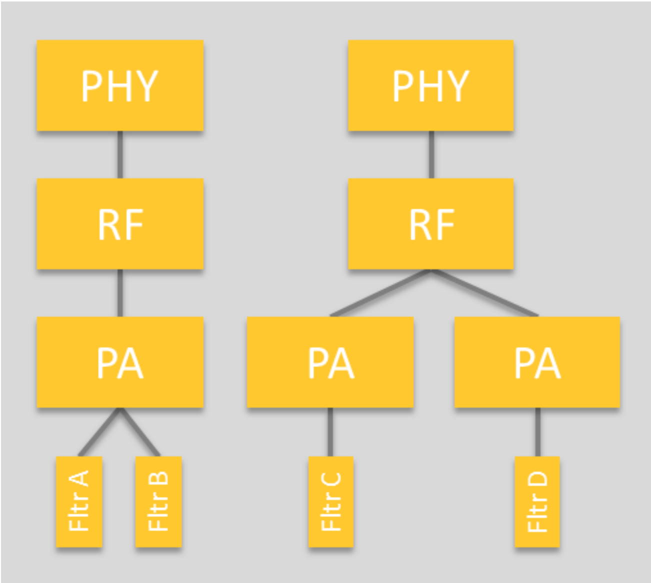
6) Shared PHY / PA / Filters, Isolated RF



DL Config Timing	-
TX Timing	-
UL Config Timing	-
HI_DCI0 Timing	-
Maximum Number PHYs	2
Maximum Total Bandwidth	30
Maximum DL Layers	2
Maximum UL Layers	2
Shared Bands	Yes
Shared PA	Yes
Maximum Total Power	30
Internal PNF frame alignment	Yes
PHY Config List	
1	
Bandwidth List	5,10,15,20
Band List	1,2,3,4,5
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
2	
Bandwidth List	5,10,15,20
Band List	1,2,3,4,5
Band Exclusion List	-
Maximum DL Layers	2
Maximum UL Layers	1
Maximum UE per TTI	16
RF Config List	
1	
Band	7
Maximum Power	24

	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
2		
	Band	4
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2110
	Maximum DL Frequency	2155
	Minimum UL Frequency	1710
	Maximum UL Frequency	1755
3		
	Band	41
	Maximum Power	20
	Maximum Antennas	1
	Minimum DL Frequency	2496
	Maximum DL Frequency	2690
	Minimum UL Frequency	2496
	Maximum UL Frequency	2690
4		
	Band	12
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716
5		
	Band	3
	Maximum Power	24
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

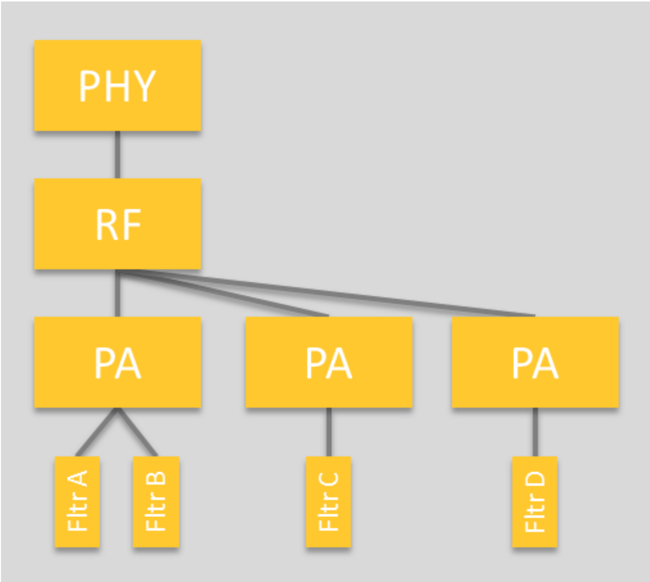
7) *Specific FDD + TDD Example*



DL Config Timing		-
TX Timing		-
UL Config Timing		-
HI_DCI0 Timing		-
Maximum Number PHYs		2
Maximum Total Bandwidth		40
Maximum DL Layers		2
Maximum UL Layers		2
Shared Bands		Yes
Shared PA		No
Maximum Total Power		N/A
Internal PNF frame alignment		Yes
PHY Config List		
1		
	Bandwidth List	5,10,20
	Band List	1,2
	Band Exclusion List	-
	Maximum DL Layers	2
	Maximum UL Layers	1
	Maximum UE per TTI	16
2		
	Bandwidth List	5,10,20
	Band List	3,4
	Band Exclusion List	-
	Maximum DL Layers	2
	Maximum UL Layers	1
	Maximum UE per TTI	16
RF Config List		
1		
	Band	12
	Maximum Power	30

	Maximum Antennas	2
	Minimum DL Frequency	729
	Maximum DL Frequency	746
	Minimum UL Frequency	699
	Maximum UL Frequency	716
2		
	Band	17
	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	734
	Maximum DL Frequency	746
	Minimum UL Frequency	704
	Maximum UL Frequency	716
3		
	Band	7
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
4		
	Band	3
	Maximum Power	21
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

8) Specific TDD or FDD Example



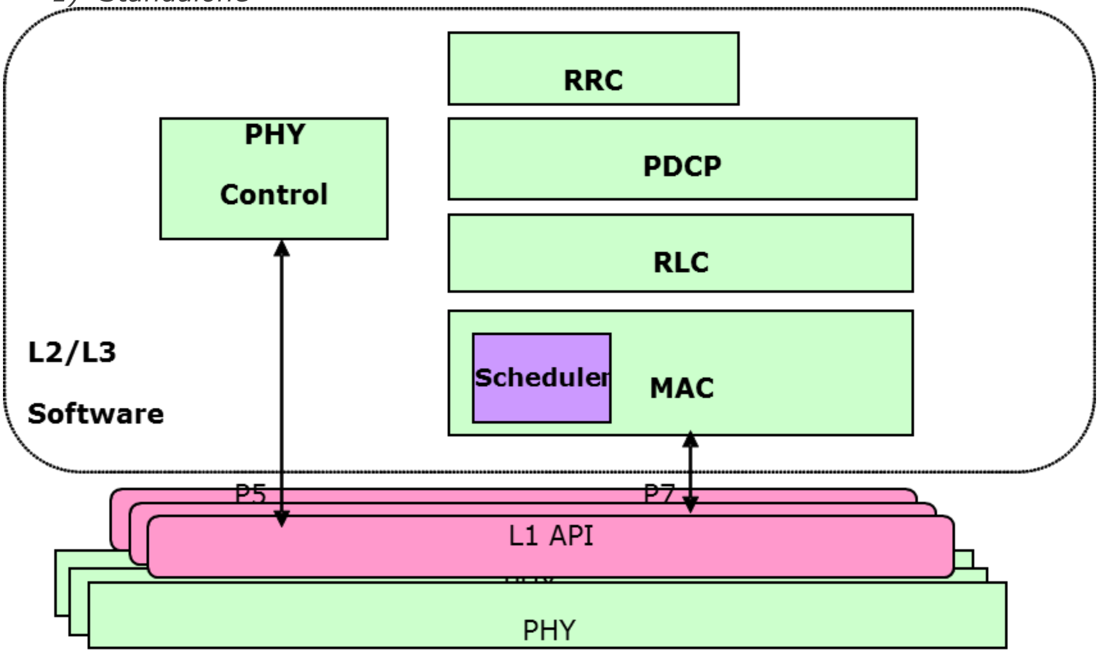
DL Config Timing			-
TX Timing			-
UL Config Timing			-
HI_DCI0 Timing			-
Maximum Number PHYs			1
Maximum Total Bandwidth			20
Maximum DL Layers			2
Maximum UL Layers			2
Shared Bands			Yes
Shared PA			No
Maximum Total Power			N/A
Internal PNF frame alignment			Yes
PHY Config List			
	1		
		Bandwidth List	5,10,20
		Band List	1,2,3,4
		Band Exclusion List	-
		Maximum DL Layers	2
		Maximum UL Layers	1
		Maximum UE per TTI	16
RF Config List			
	1		
		Band	12
		Maximum Power	30
		Maximum Antennas	2
		Minimum DL Frequency	729
		Maximum DL Frequency	746
		Minimum UL Frequency	699
		Maximum UL Frequency	716
	2		
		Band	17

	Maximum Power	30
	Maximum Antennas	2
	Minimum DL Frequency	734
	Maximum DL Frequency	746
	Minimum UL Frequency	704
	Maximum UL Frequency	716
3		
	Band	7
	Maximum Power	24
	Maximum Antennas	2
	Minimum DL Frequency	2620
	Maximum DL Frequency	2690
	Minimum UL Frequency	2500
	Maximum UL Frequency	2570
4		
	Band	3
	Maximum Power	21
	Maximum Antennas	1
	Minimum DL Frequency	1805
	Maximum DL Frequency	1980
	Minimum UL Frequency	1710
	Maximum UL Frequency	1785

NB-IOT Configuration Options

This appendix provides examples of the relationship between MAC, FAPI and PHY for the different NB-IOT modes. It is intended to show examples to aid an implementor and is not exhaustive.

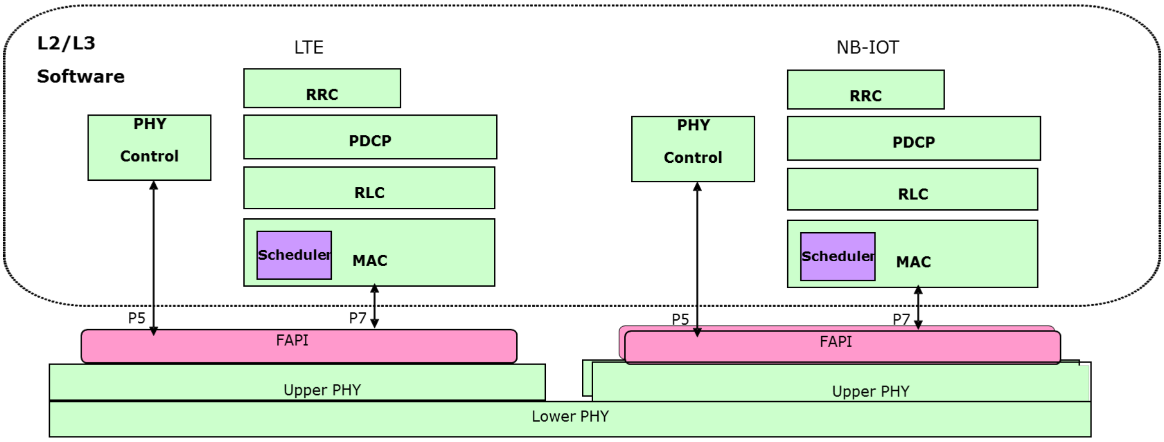
1) *Standalone*



In standalone mode the NB-IOT FAPI is identical to the LTE FAPI in Figure 3-2.

2) *In-band*

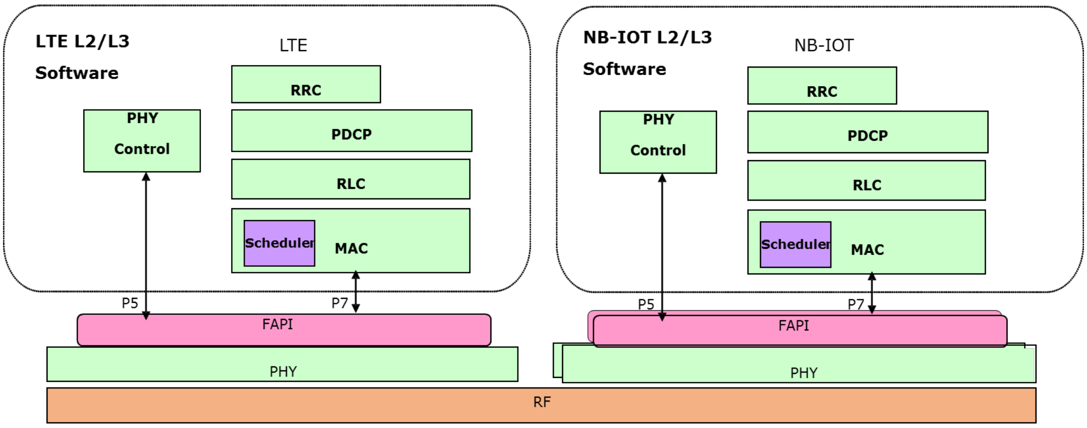
In in-band mode one LTE carrier (with one instance of FAPI) will include one or more NB-IOT carriers (with one instance of FAPI per NB-IOT carrier).



A eNB implementation can include both LTE and NB-IOT carriers and in this scenario there may be joint processing within the PHY (and communication between MAC schedulers). This is shown by splitting the PHY into upper and lower components, however, the actual split between upper and lower is implementation dependent.

3) Guard-band

In guard-band mode one LTE carrier (with one instance of FAPI) will include one or more NB-IOT carriers in it's guard-band (with one instance of FAPI per NB-IOT carrier).



A eNB implementation can include both LTE and NB-IOT carriers and in this scenario there is no requirement for joint processing within the PHY. However, the RF components are likely to be shared.

References

Ref.	Title	Number	Version & Date
[1]	Virtualization for Small Cells: Overview	SCF106	106.05.1.02
[2]	Small Cell Virtualization Functional Splits and Use Cases	SCF159	159.05.1.02
[3]	SON API for Small Cells	SCF083	083.05.01
[4]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control	TS36.331	13.5.0
[5]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control(MAC) protocol specification	TS36.321	13.5.0
[6]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Overall Description; Stage 2	TS36.300	13.7.0
[7]	3GPP Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Architecture description	TS36.401	13.2.0
[8]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode	TS36.304	13.5.0
[9]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer procedures	TS36.213	13.5.0
[10]	3GPP Base Station(BS) Radio Transmission and Reception	TS36.104	13.7.0
[11]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channel and Modulation	TS36.211	13.5.0
[12]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding	TS36.212	13.5.0
[13]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment(UE) radio access capabilities	TS36.306	13.5.0
[14]	3GPP Home eNodeB (HeNB) Operations, Administration, Maintenance and Provisioning (OAM&P); Information model for Type 1 interface HeNB to HeNB Management System (HeMS)	TS32.592	13.1.0
[15]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer; Measurements	TS36.214	13.4.0
[16]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); Requirements for support of radio resource management	TS36.133	13.7.0
[17]	3GPP Evolved Universal Terrestrial Radio Access (E-UTRA); LTE Positioning Protocol (LPP)	TS36.355	13.3.0
[18]	IETF RFC 4960 Stream Control Transmission Protocol	IETF RFC 4960	Sept 2007
[19]	3GPP GSM/EDGE Radio Access Network; Radio subsystem link control	TS45.008	13.5.0

[20]	3GPP Radio Access Network; Physical Layer Measurements (FDD)	TS25.215	10.0.0
[21]	Synchronisation for LTE Small Cells	SCF075	075.06.01
[22]	3GPP Network Based Positioning Systems in E-UTRAN	TS36.111	11.4.0