O-RAN Acceleration Abstraction Layer General Aspects and Principles

O-RAN ALLIANCE e.V.
Buschkauler Weg 27, 53347 Alfter, Germany
Register of Associations, Bonn VR 11238
VAT ID DE321720189

Revision History

Date	Revision	Author	Description
2020.06.24	0.01.00.01	Niall Power (Intel)	First draft of O-RAN O-Cloud AAL General Aspects and Principles Specification
2020.08.16	0.01.00.02	Various	Updated AAL Definitions and restructured architecture and configuration sections.
2020.10.23	0.01.00.03	Lopamudra Kundu (Nvidia)	5G eMBB PHY Layer channels used for AAL Profile definitions for an O-DU AAL Profile, chapter 5 O-DU AAL Inline profiles
		Niall Power (Intel)	AAL O-DU PUSCH and PDSCH FEC Profiles added to chapter 5 Acceleration Manager Chapter 3
		Arjun Nanjundappa (Mavenir Systems)	O-DU AAL profiles for mMTC (NB-IOT)
2020.10.29	0.01.00.04	Various	Updated from Review

Contents

Table of Figures

Chapter 1. Introduction

1.1 Scope of this document

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

The contents of the present document are subject to continuing work within O-RAN WG6 and may change following formal O-RAN approval. Should the O-RAN.org modify the contents of the present document, it will be re-released by O-RAN ALLIANCE with an identifying change of release date and an increase in version number as follows:

Release x.y.z

where:

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

This document defines O-RAN O-Cloud hardware accelerator interface functions and protocols for the O-RAN AAL interface. The document studies the functions conveyed over the interface, including configuration and management functions, procedures, operations and corresponding solutions, and identifies existing standards and industry work that can serve as a basis for O-RAN work.

1.2 References

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

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

[1] O-RAN WG1 Architecture Description

[2] O-RAN WG1 OAM Architecture

[3] O-RAN WG6 Cloud Architecture and Deployment Scenarios

[4] ETSI GS NFV-IFA 002 v2.4.1 NFV Acceleration Technologies; VNF Interfaces Specification

[5] ETSI GS NFV-IFA 019 NFV Acceleration Technologies; Acceleration Resource Management Interface Specification

[6] 5G; NR; Physical Channels and Modulation 3GPP TS 38.211 v15.2.0 Release 15

[7] 5G; NR; Multiplexing and Channel Coding 3GPP TS 38.212 v15.2.0 Release 15

[8] LTE; E-UTRA Physical Channels and Modulation 3GPP TS 36.211 v15.2.0 Release 15

[9] LTE; E-UTRA Multiplexing and Channel Coding 3GPP TS36.212 v15.2.1 Release 15

[10] ETSI_GS_NFV-IFA_004 NFV Acceleration Technologies; Management Aspects Specification

[11] Vocabulary for 3GPP Specifications (TR21.905)

1.3 Definitions and Abbreviations

1.3.1 Definitions

For the purpose of this document the terms and definitions given in ETSI GS NFV-IFA 002 [4], ETSI GS NFV-IFA 004 [10] and the following apply:

Hardware Accelerator specialized HW implementation that can perform some compute, offloaded from the General-Purpose Processor

NOTE: Examples of Hardware Accelerators include ASIC, FPGA, DSP and GPU.

Acceleration Abstraction Layer Interface (AALI) is a programming API and information models between an application and a Hardware Accelerator within an O-Cloud instance.

AAL Implementation is a realization of an AAL interface including but not limited to the software libraries, drivers and Hardware Accelerator

Accelerated Function is a workload building block that a Hardware Accelerator processes on behalf of an application within an O-RAN Cloudified Network Function

AAL Profile specifies a set of Accelerated Functions that a Hardware Accelerator processes on behalf of an application within an O-RAN Cloudified Network Function. The AAL Profile APIs are a subset of the AALI that supports a specific set of Accelerated Functions defined by the AAL Profile.

An Operation is the action applied to input data which is processed in an AAL device producing output data based on the AAL profile supported by the AAL device.

AAL Device is a logical device which is part of the AAL that maps to a logical instance of a HW Accelerator.

- An AAL Device maps to a single HW Accelerator
- A HW Accelerator may support 1 to N AAL Devices
- Each AAL Device shares the resources of the associated HW Accelerator with other AAL device(s) mapped to the same HW Accelerator.
- Mapping of HW Accelerator resources to AAL Devices shall be configurable from O2 interface
- An AAL Device is associated with a single process address space and is not shared between multiple process address spaces
- An AAL Device contains 1 to M Queue IDs/Compute IDs
- An AAL Device may support more than one AAL profile

AAL Device ID is a unique index used to designate the AAL device in all functions exported by the AAL.

AAL Queue is part of the AAL and is defined as an abstract construct that is used by the AAL to group operations together and may access specific resources (compute, I/O) of an AAL Device.

- From the application point of view, each instance of an AAL device consists of one or more AAL queues
- While an AAL device may support multiple AAL profiles, an AAL queue supports only one type of AAL profile
- AAL Queue optionally also supports priority, allowing the application/network function to schedule jobs of different priorities to the AAL Device

NOTE:

- An AAL queue can be used by an application/network function to share AAL Device resources between threads/cores belonging to the same process address space
- An application/network function may use multiple AAL queues to access different AAL profiles supported by an AAL Device

AAL Queue ID is a unique index used to designate the AAL Queue in function exported by the AAL.

NOTE: An AAL Queue or an AAL Queue ID does not reflect a HW design or an AAL Implementation specification

HW Accelerator Manager is an acceleration management function, that provides management capabilities for the HW Accelerator(s) in the O-Cloud Node. Management capabilities include lifecycle management, configuration, updates/upgrades and failure handling.

1.3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [11] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [11].

AF	Accelerated Function
AAL	Acceleration Abstraction Layer
AALI	Acceleration Abstraction Layer Interface
CNF	Cloudified Network Function
FCAPS	Fault, Configuration, Accounting, Performance, Security
FEC	Forward Error Correction
MANO	Management and Orchestration
MnS	Management Service
MOC	Managed Object Class
MOI	Managed Object Instance
ONAP	Open Network Automation Platform
O-RAN	Open Radio Access Network
OSM	Open Source Mano
RAN	Radio Access Network
SMO	Service Management and Orchestration
TR	Technical Report
TS	Technical Specification
VNF	Virtual Network Function

Chapter 2. General Aspects

2.1 Hardware Acceleration

In the design of digital computing systems, ranging from general-purpose processors to fully customized hardware, there is a tradeoff between flexibility and efficiency, with efficiency increasing by orders of magnitude when any given application is implemented in hardware. The range of implementation options includes general-purpose processors such as CPUs, more specialized processors such as GPUs, functions implemented on field-programmable gate arrays (FPGAs), and fixed-functions implemented on application-specific integrated circuits (ASICs). Hardware accelerator is a specialized HW implementation that can perform some pre-defined work offloaded from the General-Purpose Processor. Any transformation of data or routine that can be computed, can be calculated purely in software running on a generic CPU, purely in a custom-made hardware, or using a combination of both. The implementation of computing tasks in hardware to reduce latency and to increase throughput is known as hardware acceleration. The hardware acceleration is implemented in the form of lookaside or inline mode where in the former case, the host CPU invokes an accelerator for data processing and receives the result after processing is complete, while in the latter case, the host CPU, after invoking an accelerator for data processing, does not necessarily retrieve the post processed data. The principle of hardware acceleration and functional offloading is illustrated in

Figure 2.1, allowing the application to offload workload to a hardware accelerator and to continue performing other work in parallel- this could be to continue to execute other software tasks in parallel or to sleep and wait for the accelerator hardware to complete. The hardware acceleration boosts application performance in environments with compute-intensive, deeply pipelined, massively parallel operations as shown in Figure 2.1. This model requires the API to support two operations, one for initiating the offload and another for retrieving the operation once complete.

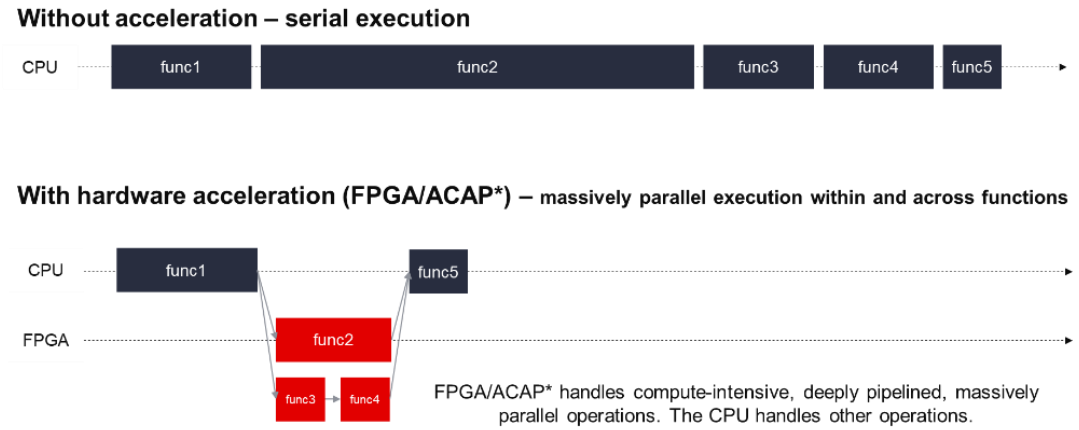


Figure 2.1 Example illustration of the effect of hardware acceleration on functional compute performance

2.2 AAL Architecture

The goal of the AAL is to specify a common and consistent interface for HW Device accelerators to the applications which facilitates decoupling of an application, e.g. O-RAN Cloudified Network Function, from a specific HW Accelerator device implementation. In order to accommodate the many different combinations of HW and SW implementation and also many different network deployment scenarios, the AAL introduces the concept of an AAL profile which is used to distinguish between the different combinations of accelerated functions to be offloaded. The high-level AAL architecture block diagram is shown in Figure 2.2.

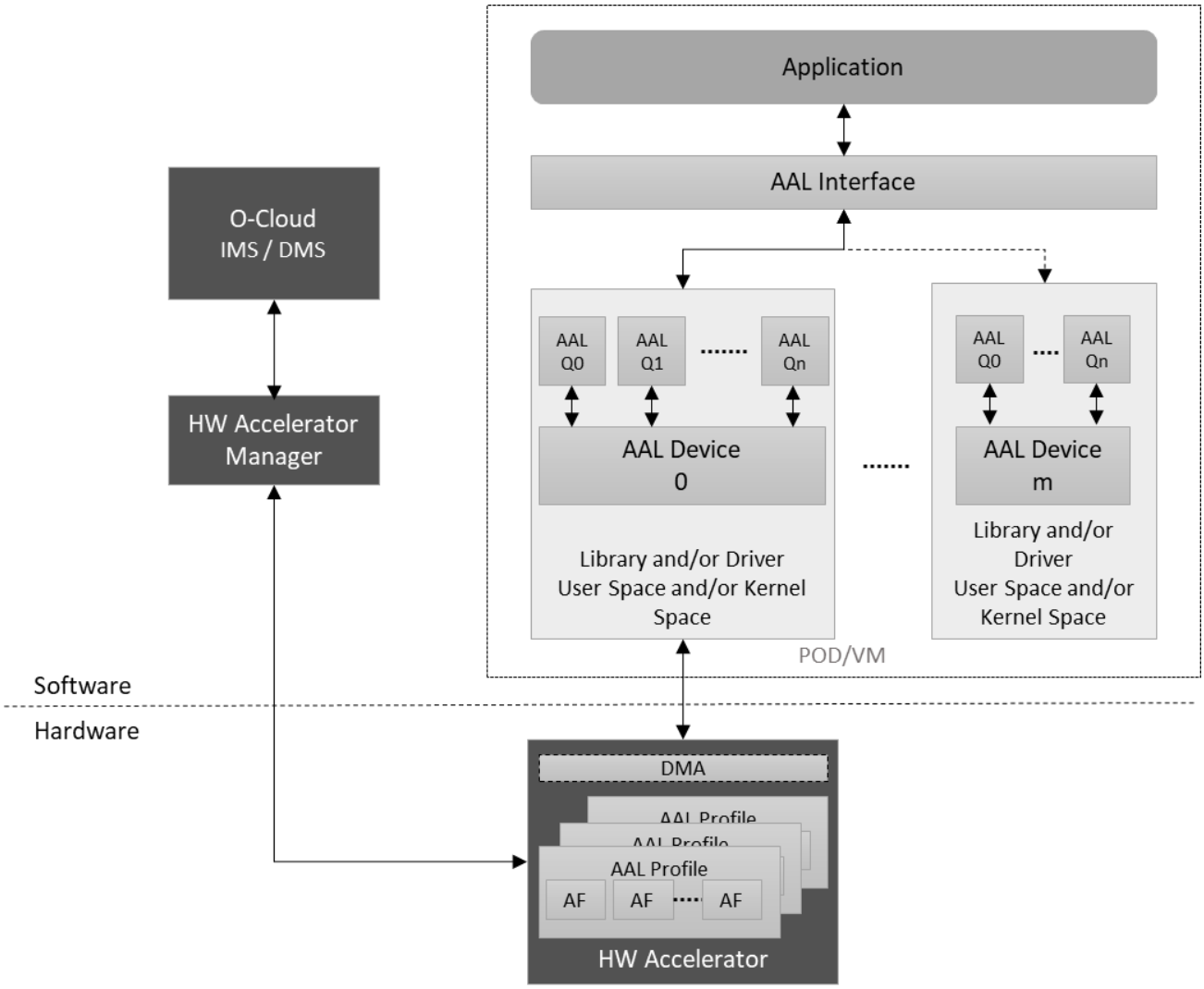


Figure 2.2 High Level AAL Architecture Diagram

Figure 2.3 AAL Component Relationship and Cardinality shows the relationship and cardinality between the components that constitute the AAL architecture.

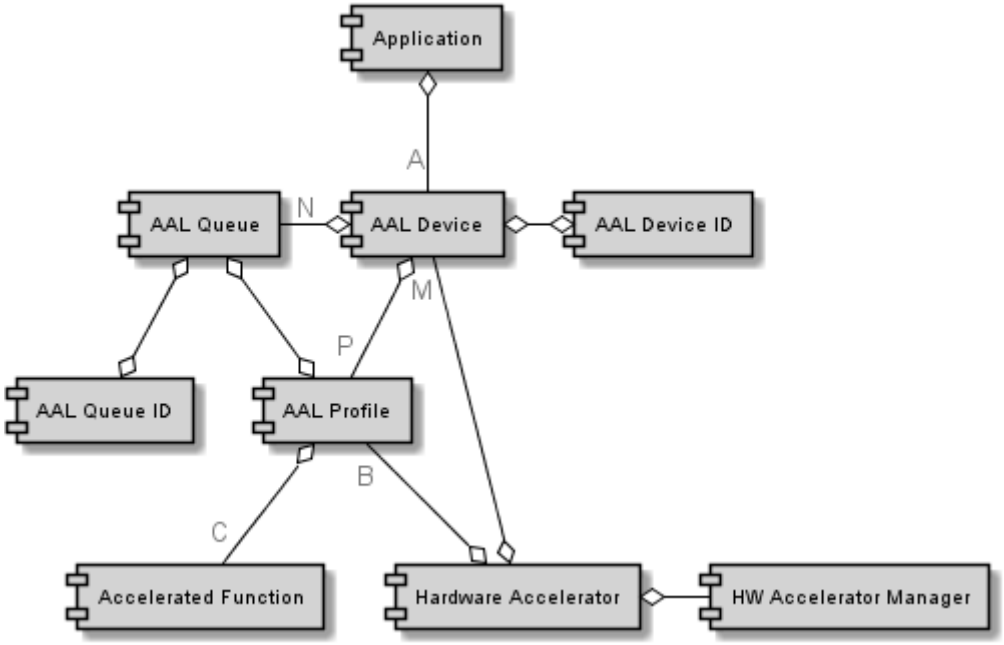


Figure 2.3 AAL Component Relationship and Cardinality

2.2.1 AAL Deployed in Cloud environments

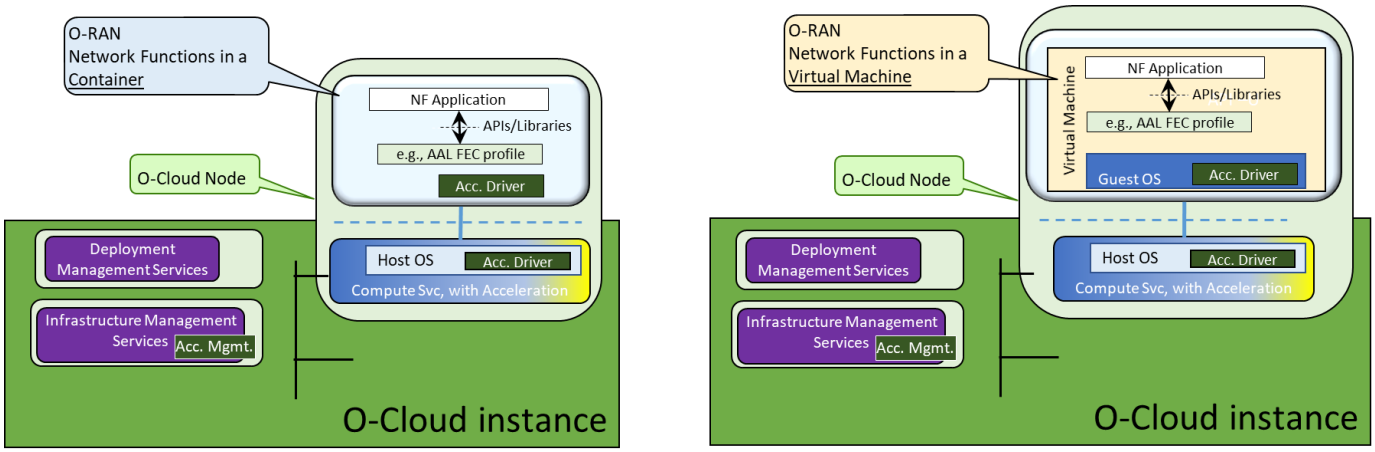


Figure 2.4 Accelerator APIs/Libraries in Container and Virtual Machine Implementations

The AAL specification shall define the AAL interface and the profiles that may be supported by that interface. The AAL interface is the application interface the O-RAN Cloudified Network Functions shall use to access the underlying HW Device Accelerator. In Figure 2.4 Accelerator APIs/Libraries in Container and Virtual Machine Implementations two deployment scenarios are shown, one with Containers and the other with Virtual Machines. In both instances the AALI is the interface between the Network Function application and the HW Accelerators that are exposed to the Network Function application.

Figure 2.4 Accelerator APIs/Libraries in Container and Virtual Machine Implementationsalso shows the O-Cloud Infrastructure Management Services and Accelerator management. The AAL Specification shall define the requirements for managing the hardware accelerator in the O-Cloud instance. The orchestration of the HW Accelerator Manager is outside the scope of the AAL and shall be specified in the ORAN WG6 O2 specification.

2.3 AAL Interface Specification Objectives

The AAL interface specification facilitates the following:

Deployment of O-RAN Cloudified Network Functions with O-Cloud HW Accelerators from different manufacturers.

2.4 Scope of the AALI

The AAL specification shall define the AALI – the interface between the application and hardware accelerator in the O-Cloud instance. This include the APIs and information models used by the application to interface with the AAL implementation. The AALI implementation itself shall not be defined by the AAL specification and is up to individual vendors to realize and can be implementation specific. ETSI GS NFV-IFA 002 [5] defines several abstraction models including pass through and abstracted models that can be used to realize an AAL implementation.

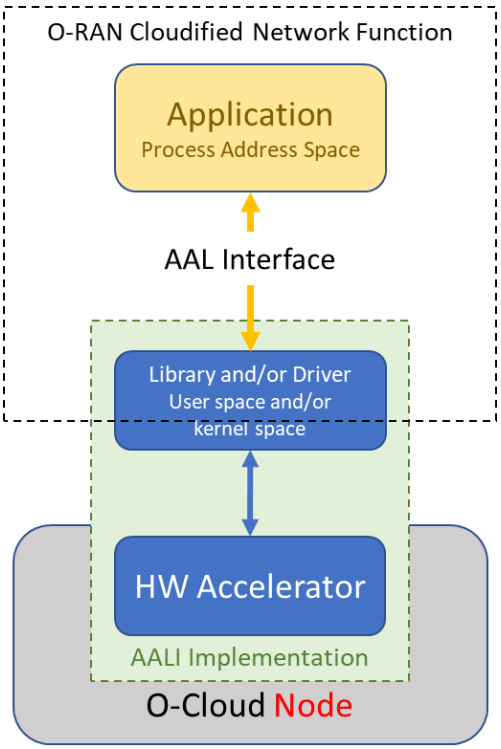


Figure 2.5 AAL Specification Scope

2.5 General Interface Principles

2.5.1 Generic Principles

2.5.1.1 Extensibility

O-RAN has defined the functions that can be accelerated by the cloud platform based on 3GPP specifications and O-RAN deployment scenarios. However, the AAL API should not limit innovation of future implementations and should evolve as the specification requires. In this way, the API shall be extensible to accommodate future revisions of the specification.

2.5.1.2 HW Independence

An AAL profile API should be independent of the underlying HW.

2.5.1.3 Interrupt and Poll Mode

The API shall allow multiple design choices for application vendors and shall not preclude a vendor from adopting an interrupt-driven design or poll-mode design or any combination of both. As such, the API shall support both interrupt mode, poll mode and any combination of interrupt and poll modes for the data-path application interface.

2.5.1.4 Discovery and Configuration

The API shall support application software to discover and configure the AAL Device. The API shall allow an application to discover what physical resources have been assigned to it from the upper layers and then to configure said resources for offload operations.

2.5.1.5 Multiple Device Support

There may be scenarios where multiple AAL Devices (either implementing the same AAL profile or different) are assigned to a single application, which uses one or more of these AAL Devices as needed. The AAL shall support an application using one or more AAL Devices at the same time, as shown in Figure 2.6

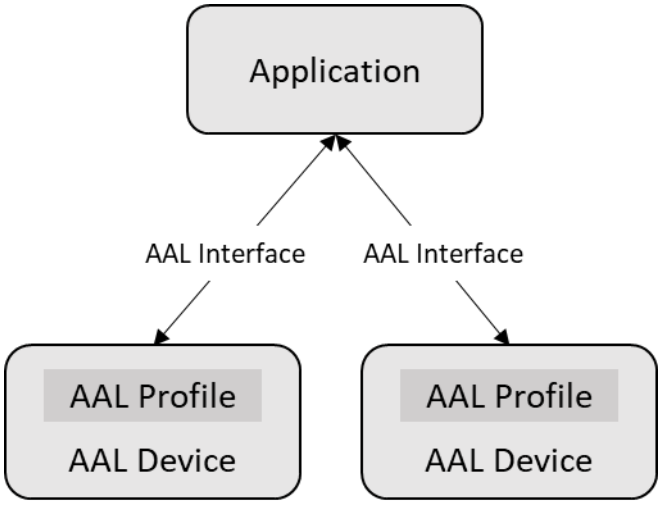


Figure 2.6. Logical Representation of AAL support for multiple devices

2.5.1.6 AAL offload capabilities

The AAL in supporting different implementations shall support different offload architectures including look−aside, inline, and any combination of both.

2.5.1.7 Look-aside Acceleration Model

The AALI shall support look-aside acceleration model where the host CPU invokes an accelerator for data processing and receives the result after processing is complete. A look-aside architecture, illustrated in Figure 2.7, allows the application to offload work to a hardware accelerator and continue to perform other work in parallel—this could be to continue to execute other software tasks in parallel or to sleep and wait for the accelerator hardware to complete. This model requires the AALI to support two operations, one for initiating the offload and another for retrieving the operation once complete.

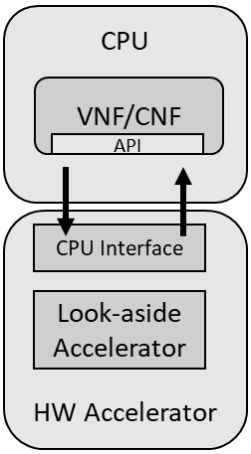


Figure 2.7. AAL look-aside acceleration model

2.5.1.8 Inline Acceleration Model

The AALI shall support inline acceleration model where acceleration by function and I/O-based acceleration are performed on the physical interface as the packet ingresses/egresses the platform. Figure 2.8. AAL inline acceleration model shows one possible implementation of an inline acceleration model.

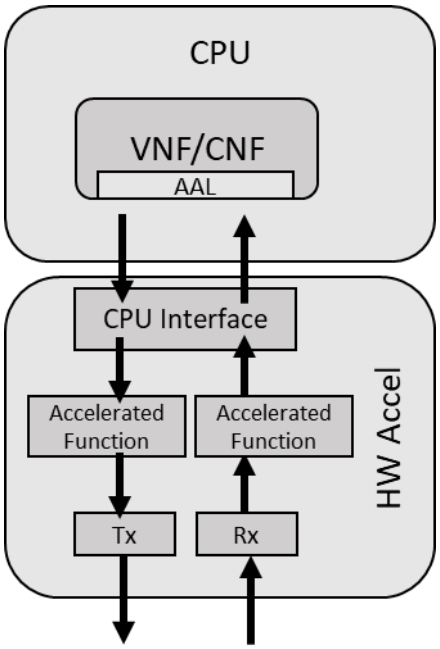


Figure 2.8. AAL inline acceleration model

In Figure 2.8. AAL inline acceleration model, “Tx” refers to the transmission of the data from the acceleration device to an Ethernet interface, while “Rx” refers to the reception of data from the Ethernet interface to the acceleration device.

While the look-aside architecture (in DL) shall support dataflow from the CPU to the accelerator and back to the CPU before being sent to the front-haul interface, the inline architecture (in DL) shall support data flow from the CPU to the accelerator and directly from the accelerator to the front-haul interface, instead of being sent back to the CPU. The typical user plane data flows for accelerating the O-DU high-PHY functions for the look-aside and inline architectures are as follows.

Look-aside architecture user plane dataflow

CPU ↔ accelerator ↔ CPU ↔ front-haul: for a set of consecutive PHY functions offload (e.g., FEC)

CPU ↔ accelerator ↔ CPU ↔ accelerator ↔…↔ CPU ↔ front-haul: for a set of non-consecutive PHY functions offload

Inline architecture user plane dataflow

CPU ↔ accelerator ↔ front-haul: for a set of consecutive PHY functions offload (up to the end of the PHY pipeline)

Figure 2.9 illustrates one possible implementation of the look-aside and inline architectures. While a set of PHY-layer functions are offloaded to the accelerator hardware for look-aside acceleration, the entire end-to-end high PHY pipeline is offloaded to the accelerator for inline acceleration.

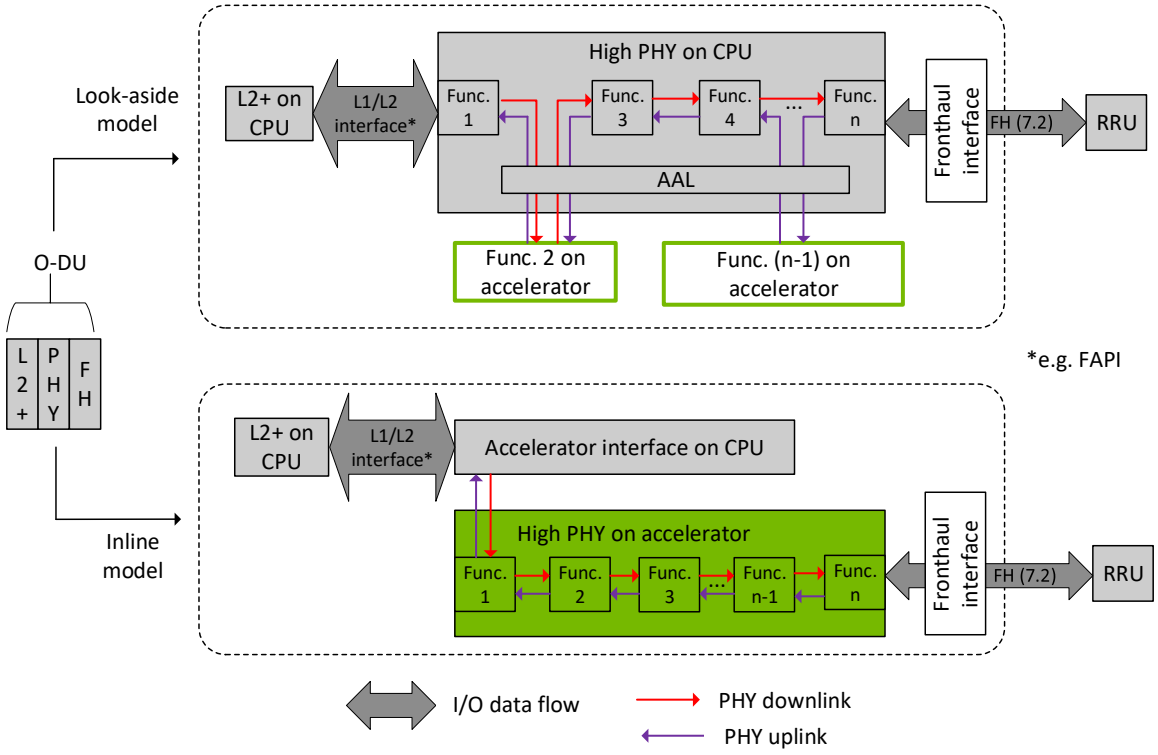


Figure 2.9. User plane dataflow paths in look-aside and inline acceleration architectures.

2.5.1.9 API Concurrency and Parallelism

To enable greater flexibility and design choice by application vendors, the AAL shall support multi-threading environment allowing an application to offload acceleration requests in parallel from several threads.

2.6 Relationship with Standards

The O-RAN AAL interface shall leverage existing standards wherever possible.

2.6.1 Relationship with ETSI

In [4,5,10], ETSI has specified a generic acceleration and abstraction model as well as acceleration resource management that have served as the basis of this specification. This specification consistently complements the aforementioned ETSI specifications and provides guidance on practical implementation of the concepts introduced in ETSI specifications on NFV acceleration interfaces.

Chapter 3. HW Accelerator Manager General Principles and Requirements

The HW Accelerator Manager is part of the O-Cloud platform management, it is responsible for the lifecycle management, configuration, updates/upgrades and error handling of the HW Accelerator(s) that are part of the Cloud Platform Hardware. The purpose of the HW Accelerator Manager is to define the standard common management methods and interfaces that a HW Accelerator(s) in an O-Cloud Node should support.

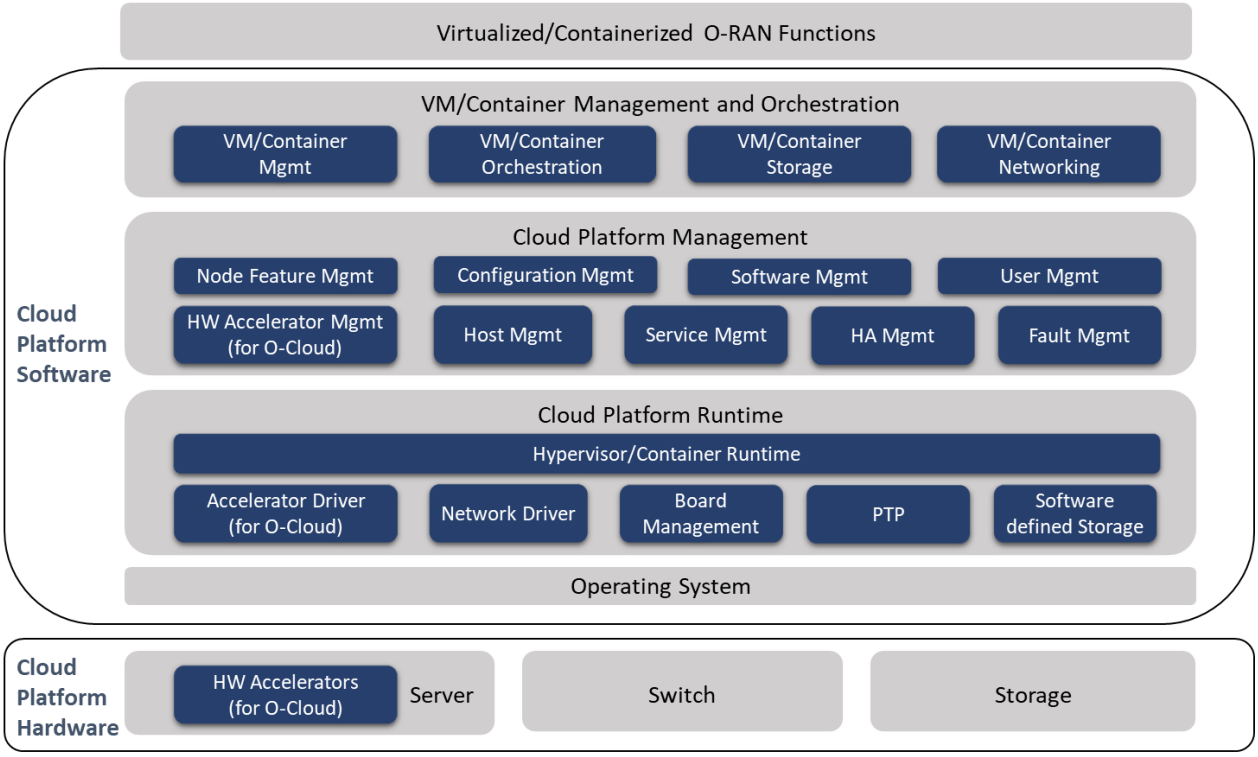


Figure 3.1 Example O-Cloud Platform and Hardware

3.1 HW Accelerator Manager Requirements

3.1.1 Lifecycle Management

The HW Accelerator Manager shall provide lifecycle management as described in ETSI NFV Acceleration Technologies; Management Aspect Specification [6] for the HW Accelerator(s) on the O-Cloud node. The HW Accelerator Manager shall provide local management functions within the O-Cloud Node. The AAL Specification will define what interfaces to use when exposing these management functions to the O-Cloud IMS. In addition, the following functions may be supported by the HW Accelerator Manager.

3.1.1.1 Updates/Upgrades

The HW Accelerator Manager shall allow the update and/or upgrade of a HW Accelerator on the O-Cloud node. An example of this includes the programming or re-programming of an FPGA image or driver updates. Updates/Upgrades can be done locally or remotely.

3.1.1.2 Configuration

The HW Accelerator Manager shall allow the configuration of the HW Accelerator as prescribed by the SMO through the O2 interface. The configuration of the HW Accelerator Manager may include HW Accelerator resource assignment to AAL Devices, AAL Queues and/or Cloudified Network Functions.

3.1.1.3 HW Accelerator Manager Monitoring

3.1.1.3.1 Fault Monitoring

HW Accelerator Manager shall process HW Accelerator Events at O-Cloud node and take appropriate action. For example, an error interrupt event could result in resetting shared hardware accelerator resources.

HW Accelerator Manager should expose relevant events or state changes to the Notification Subscription Framework Section 5.4.5 from [3].

SMO or other external entities can subscribe to HW Accelerator Manager events via the Notification Subscription Framework.

3.1.1.3.2 Performance Monitoring

Analytics such as throughput, latency, etc. may be measured and analyzed at the application layer (user space) in order to monitor the performance of the HW Accelerator. The details of such functions are for further study.

Chapter 4. AALI Configuration and Management Principles

The AAL interface has two distinct parts, the first a set of common methods ('AAL Common') which allows an application to query and identify the available AAL profiles that the underlying cloud platform offers and configure them for application use.

The second is the AAL Profile APIs which are specific to each defined AAL profile. The AAL profile shall be common across the HW Accelerators accelerating the same set of functions. It enables the applications to be able to efficiently offload the accelerated workload process to HW Accelerators in a consistent way without requiring them to know every single detail of the accelerators. Figure 4.1 shows examples of the AALI APIs presented to an application in three different scenarios.

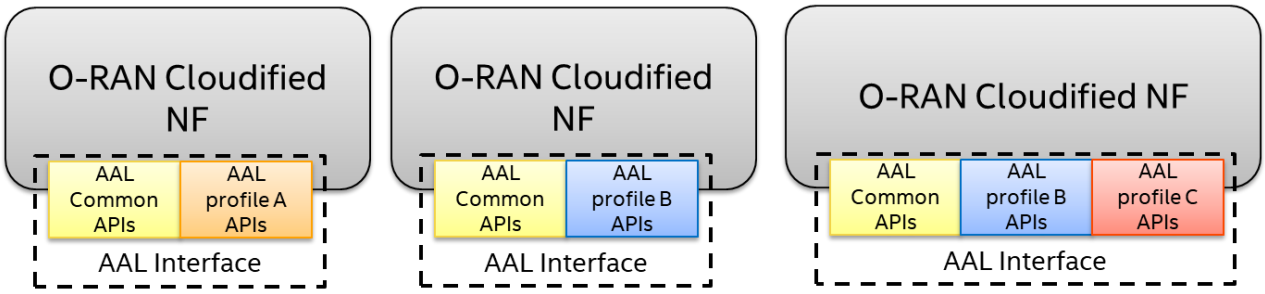


Figure 4.1 AALI Common and profile APIs

4.1 AALI Common Functions

The AAL specification consists of a common initialization and configuration section and multiple profile API specifications.

Note: A Network Function may choose to use one or more AAL profiles as part of its implementation.

The AAL initialization and configuration are done on a per AAL Device basis. That is, for each AAL Device that is assigned to the NF, the NF shall initialize and configure each AAL Device that is assigned to it.

4.1.1 AAL Initialization and Configuration Procedures

4.1.1.1 AAL Device Management

This section discusses the AAL logical and abstract representations presented to applications using the AAL interface. An AAL Device should not be confused with a HW Device.

4.1.1.2 AAL Device Identification & Representation

Within a process address space each AAL Device is uniquely designated by two identifiers:

- A unique AAL Device ID used to designate the AAL Device in all functions exported by the AAL API.
- A device name used to designate the AAL Device in console messages, for administration or debugging purposes.

Depending on HW design and implementation choice, a HW Accelerator may want to accelerate multiple profiles or offer support for sharing HW Accelerator resources between multiple threads, processes, VMs, PODs. For this reason, a second abstract identifier known as AAL Queue ID is required to

- distinguish between multiple supported AAL profiles per AAL Device
- prioritize access to AAL Device resources
- group operation requests

- Allow parallel access through AALI for multiple threads

As an abstract identifier, an AAL Queue or AAL Queue ID does not reflect a HW design specification or requirement but an AAL interface specification.

4.1.1.2.1 Example AAL Device Mapping

The following Section contains example deployments mapping AAL Devices to Applications / O-RAN Cloudified Network Functions

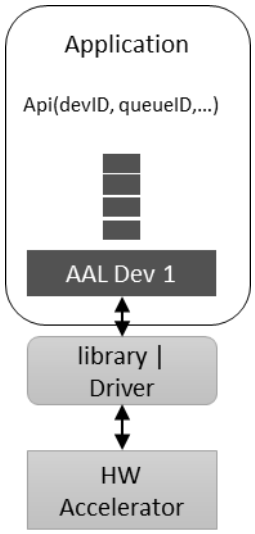


Figure 4.2 Basic mapping of AAL Device to O-RAN Cloudified NF

Figure 4.2 Basic mapping of AAL Device to O-RAN Cloudified NF example shows a simple deployment with a HW Accelerator supporting a single AAL Device which exposes a single AAL Queue for the application to use.

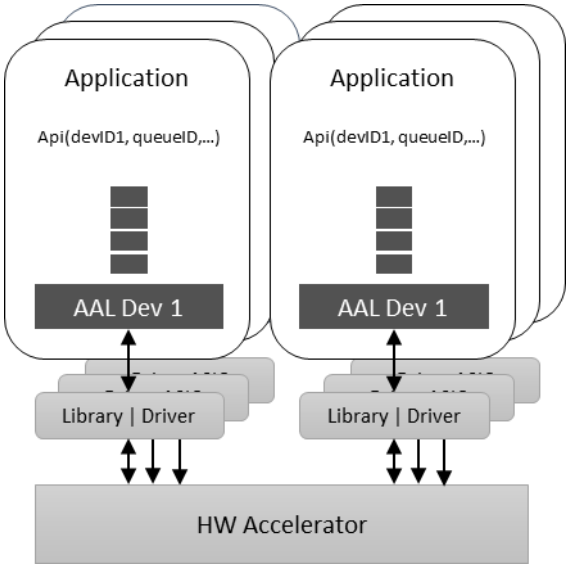


Figure 4.3 AAL Device mapping example showing multiple application support

Figure 4.3 example shows the AAL supporting multiple applications with a single HW Accelerator. The HW Accelerator exposes multiple VFs through SRIOV. Each VF maps to an instance of an AAL Device. Each Application is assigned a single AAL Device (VF) and all AAL Devices share the resources of the underlying HW Accelerator.

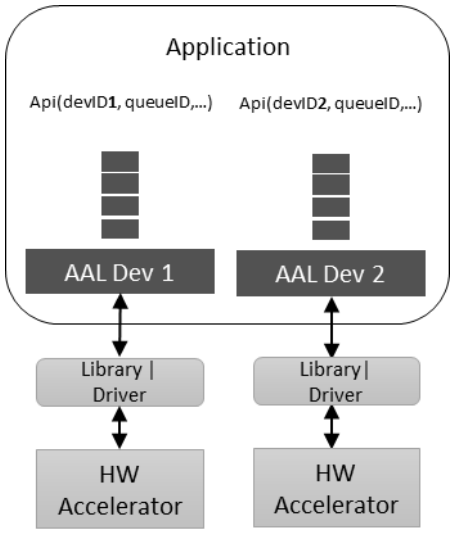


Figure 4.4 Device mapping example showing multiple device support

Figure 4.4 example shows the AAL supporting an Application that is used by multiple HW Accelerators. In this case, a cloud platform contains multiple HW Accelerators, each HW Accelerator being exposed to the application as an AAL Device.

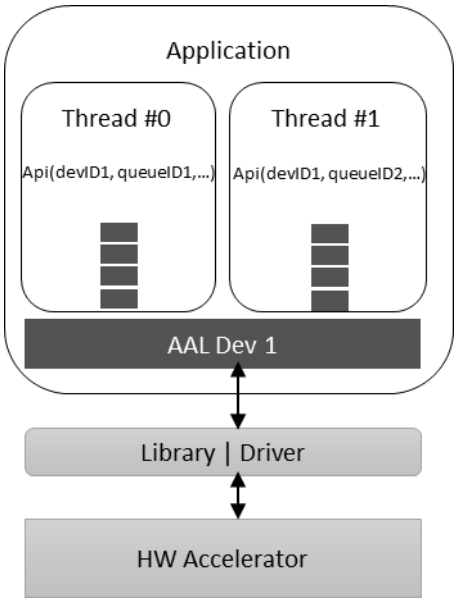


Figure 4.5 AAL Device mapping showing multiple Queue support

Figure 4.5 example shows the AAL Device supporting multiple AAL Queues which are used by an application in multiple threads – allowing the application to avoid locking when exercising the AAL interface.

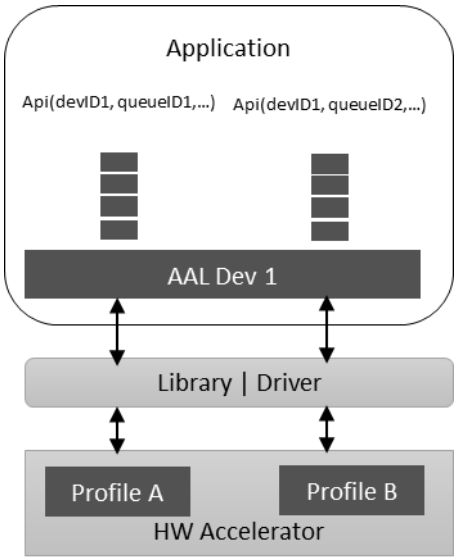


Figure 4.6 AAL Device Mapping example showing multi-function support

Figure 4.6 example shows the AAL accommodating a HW Accelerator that supports multiple AAL profiles. In this case application accesses the different profiles using the AAL Queue ID in the AALI.

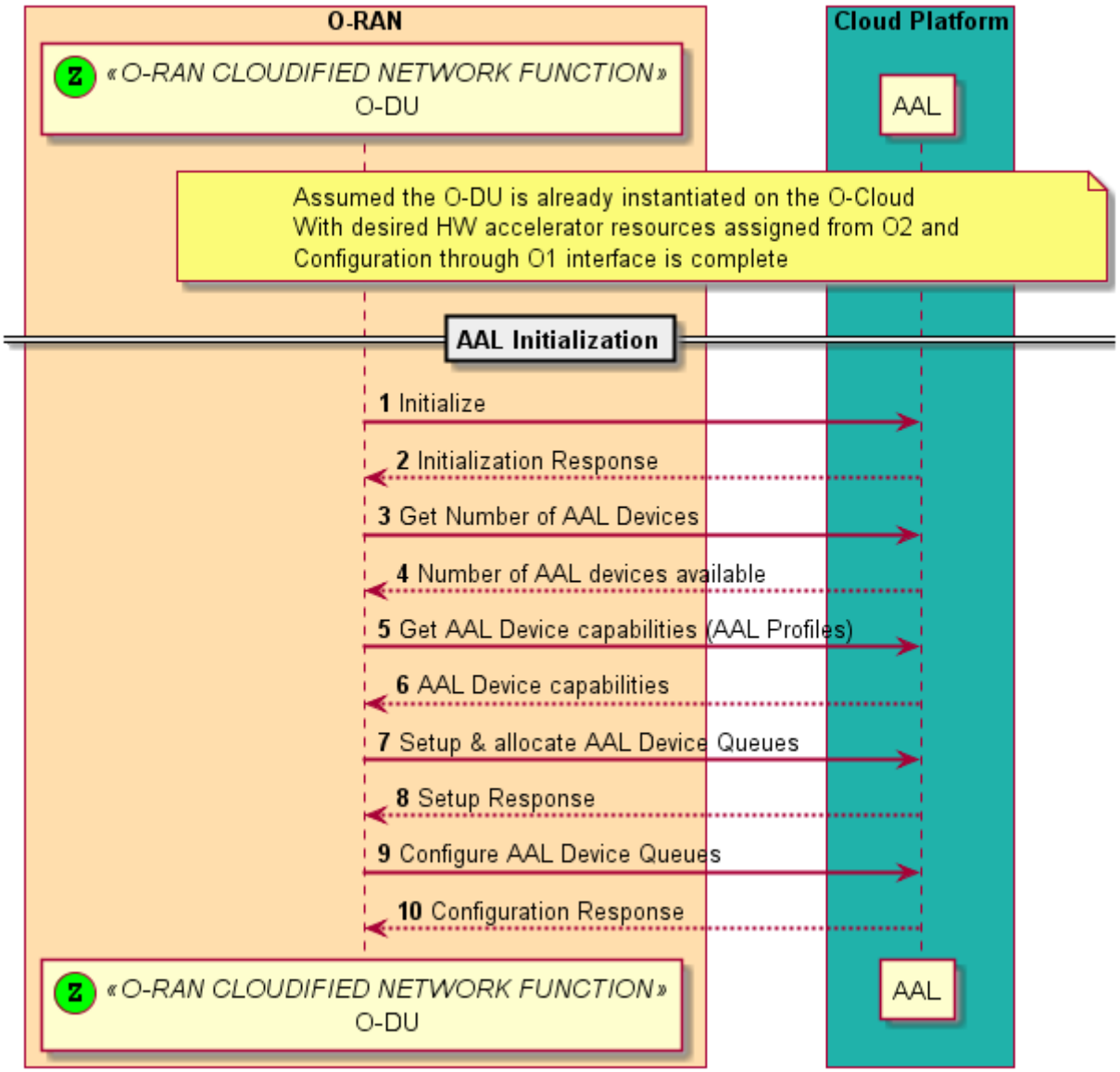
4.1.1.3 AAL Device Configuration

Configuration of an AAL Device has two different levels: configuration that applies to the whole AAL Device, and configuration that applies to a single AAL Queue.

Note that, although all AAL Queues on an AAL Device support same capabilities, they can be configured differently and will then behave differently. This section details the AAL Device configuration which includes the following operations:

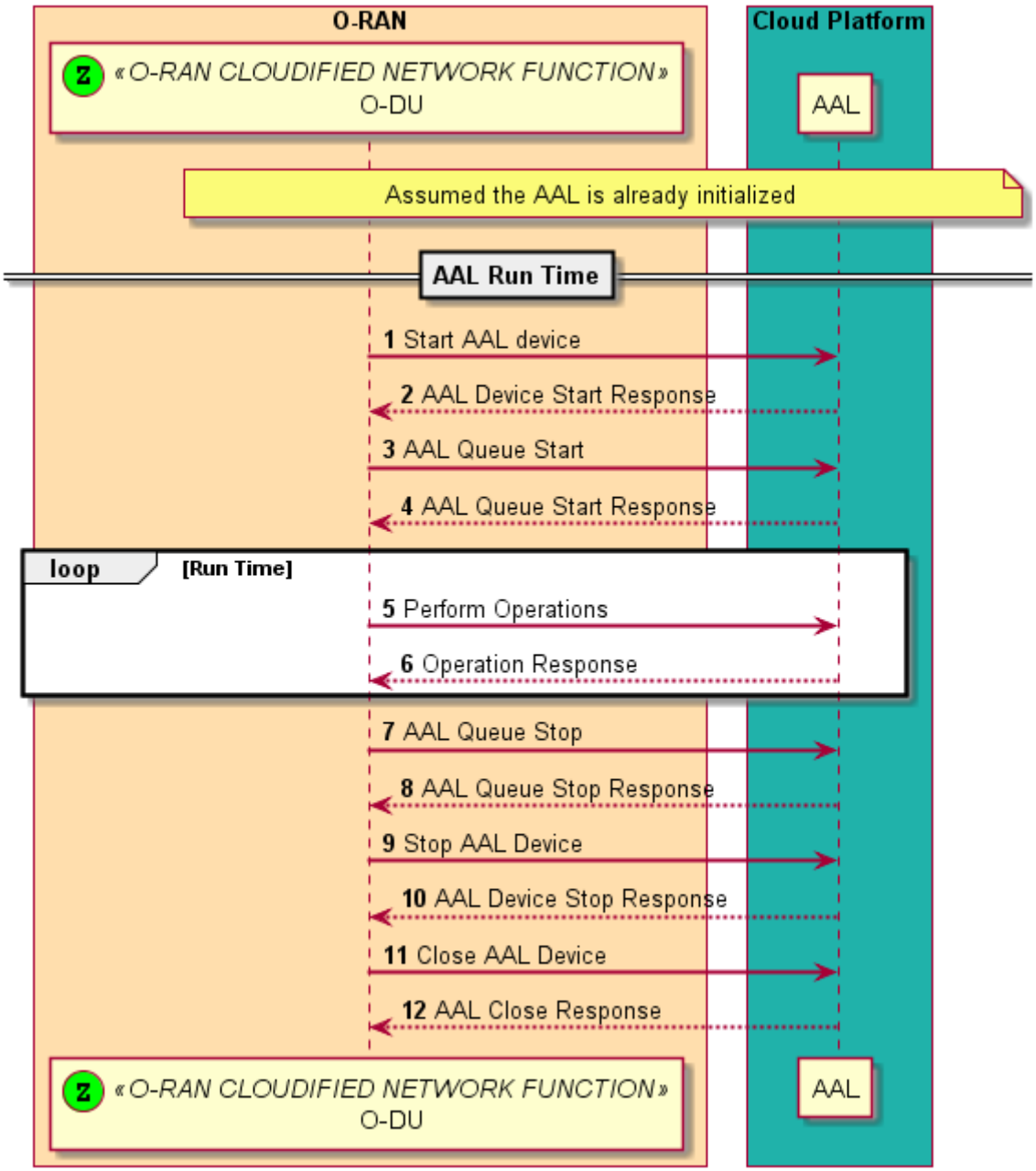
- Allocation of resources, AAL Queues.
- Resetting the AAL Device into a well-known default state.
- Initialization of statistics counters.

The below sequence diagram shows the high-level initialization and configuration procedure for the AALI.



4.1.1.4 Device and Queue management

After initialization, AAL Devices are in a stopped state, so must be started by the application. If an application is finished using an AAL Device, it can close the AAL Device. Once closed, it cannot be restarted. By default, all AAL Queues are started when the AAL Device is started, but they can be stopped individually.



Note

The above sequence diagrams refer to the AAL Device and AAL queue operations. An AAL Device may not directly correspond to a HW Accelerator as an AAL Device may only represent a set of resources of the HW Accelerator. In this case an operation (start, stop etc..) on an AAL Device may not actually translate to a HW Accelerator operation – it is abstracted by the AAL and implementation specific.

4.1.1.5 Statistics

The AALI shall provide an O-RAN Cloudified NF with general statistics upon request. Statistics may include but not limited to operation counts and error counts.

4.1.1.6 Memory Management

O-RAN network functions (O-DU, O-CU, etc.) will be responsible for input, output and operation structure memory allocation and freeing, using AAL defined memory management functions. All other application memory is not required to use the AAL memory management functions.

Device Drivers are free to manage their own internal memory, DMA implementation as needed, the AAL specification does not add any memory requirements to device driver.

Each AAL Profile shall define its own memory requirements and implement its own memory backing if needed.

Each AAL Profile may define its own operation structure memory allocation if needed.

4.1.1.7 Run Time Configurations

Operations are requested to the AAL Device to perform specific HW Accelerated Function(s). Each operation shall be represented by an operation struct that shall define all necessary metadata, configurations and information required for the operation to be processed on an AAL Device. The operation structs shall define the operation type to be performed, including an operation status and reference to the AAL Profile specific operation data which can vary in size and content depending on the AAL profile. Each AAL profile shall define its own operation structure for its specific functions.

Chapter 5. AAL Profiles

An AAL profile specifies a set of Accelerated Functions that a Hardware Accelerator processes on behalf of an application within an O-RAN Cloudified Network Function (e.g. O-DU, O-CU etc.). Accordingly, AAL profiles can be categorized as O-DU AAL profiles, O-CU AAL profiles and so on. The following sections describe these different AAL profile categories in further details.

5.1 O-DU AAL Profiles

An O-DU AAL profile can specify a set of Accelerated Functions within the O-DU protocol stack. These functions may belong to a single layer (e.g., PHY) or span across multiple layers (e.g., PHY and MAC) within O-DU. The current O-DU AAL profiles being studied by O-RAN WG6 are focusing on Accelerated Functions from PHY layer of O-DU.

5.1.1 O-DU Protocol Stack Reference

Figure 5.1 illustrates the building blocks for processing various O-DU PHY layer Downlink (DL) channels and signals (with 7.2-x functional split between O-DU and O-RU) defined by 3GPP in [6] & [7] as part of 5G NR specification.

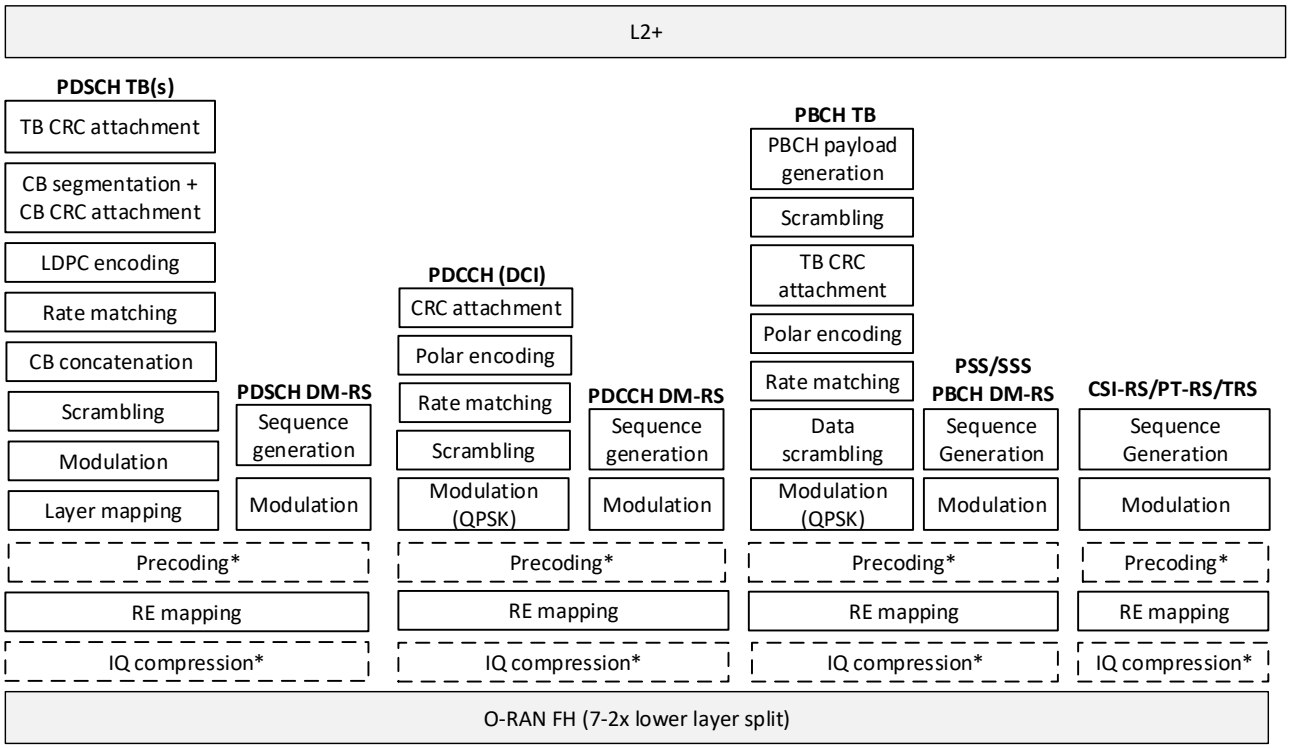


Figure 5.1 O-DU PHY processing blocks for 5G NR Downlink

The O-DU PHY layer in downlink consists of the following physical channels and reference signals:

- Physical Downlink Shared Channel (PDSCH) and associated Demodulation Reference Signal (PDSCH DM-RS).
- Physical Downlink Control Channel (PDCCH) and associated Demodulation Reference Signal (PDCCH DM-RS).
- Synchronization Signal Block (SSB) consisting of
  - Physical Broadcast Channel (PBCH) and associated DMRS (PBCH DM-RS).
  - Primary Synchronization Signal (PSS).
  - Secondary Synchronization Signal (SSS).
- Channel State Information-Reference Signal (CSI-RS) and Tracking Reference Signal (TRS).

- Phase Tracking Reference Signal (PT-RS) for DL.

The downlink physical channels (PDSCH, PDCCH, PBCH) carry information originating from higher layers (i.e. layer 2 and above).

The downlink physical layer processing of data channel (PDSCH) carrying transport blocks consists of the following steps:

TB CRC attachment: Error detection is provided on each transport block (TB) through a Cyclic Redundancy Check (CRC). Refer to Subclause 7.2.1 in [7] for details.

CB segmentation and CRC attachment: The transport block is segmented when it exceeds the code block (CB) size specified by 3GPP [7]. Code block segmentation and code block CRC attachment are performed according to Subclauses 7.2.3 and 5.2.2 of [7].

LDPC encoding: Refer to Subclauses 7.2.4 and 5.3.2 in [7] for details.

Rate matching: Refer to Subclauses 7.2.5 and 5.4.2 in [7] for details.

CB concatenation: Refer to Subclauses 7.2.6 and 5.5 in [7] for details.

Scrambling: Refer to Subclause 7.3.1.1 in [6] for details.

Modulation: Refer to Subclause 7.3.1.2 in [6] for details.

Layer mapping: Refer to Subclause 7.3.1.3 in [6] for details.

RE mapping: Refer to Subclause 7.3.1.5 and 7.3.1.6 in [6] for details on Resource Element (RE) mapping.

The downlink physical layer processing of control channel (PDCCH) carrying Downlink Control Information (DCI) consists of the following steps:

CRC attachment: Error detection is provided on DCI transmissions through a Cyclic Redundancy Check (CRC). Refer to Subclause 7.3.2 in [7] for details.

Polar encoding: Refer to Subclauses 7.3.3 and 5.3.1 in [7] for details.

Rate matching: Refer to Subclauses 7.3.4 and 5.4.1 in [7] for details.

Scrambling: Refer to Subclause 7.3.2.3 in [6] for details.

Modulation: Refer to Subclause 7.3.2.4 in [6] for details.

RE mapping: Refer to Subclause 7.3.2.5 in [6] for details.

The downlink physical layer processing of broadcast channel (PBCH) carrying maximum one transport block consists of the following steps:

PBCH payload generation: Refer to Subclause 7.1.1 in [7] for details.

Scrambling: Refer to Subclause 7.1.2 in [7] for details.

TB CRC attachment: Refer to Subclause 7.1.3 in [7] for details.

Polar encoding: Refer to Subclauses 7.1.4 and 5.3.1 in [7] for details.

Rate matching: Refer to Subclauses 7.1.5 and 5.4.1 in [7] for details.

Data scrambling: Refer to Subclause 7.3.3.1 in [6] for details.

Modulation: Refer to Subclause 7.3.3.2 in [6] for details.

RE mapping: Refer to Subclause 7.3.3.3 in [6] for details.

The downlink physical signals (DM-RS, PSS, SSS, CSI-RS/TRS, PT-RS) correspond to a set of resource elements used by the physical layer but does not carry information originated from higher layers (i.e. layer 2 and above).

Reference Signals (DM-RS, CSI-RS/TRS, PT-RS) and Synchronization Signals (PSS/SSS) are generated using the following steps:

Sequence Generation and Modulation: Refer to Subclauses 7.4.1.1.1 (PDSCH DM-RS), 7.4.1.3.1 (PDCCH DM-RS), 7.4.1.4.1 (PBCH DM-RS), 7.4.1.5.2 (CSI-RS/TRS), 7.4.1.2.1 (PT-RS), 7.4.2.2.1 (PSS) and 7.4.2.3.1 (SSS) in [6] for details.

RE mapping: Refer to Subclauses 7.4.1.1.2 (PDSCH DM-RS), 7.4.1.3.2 (PDCCH DM-RS), 7.4.1.4.2 (PBCH DM-RS), 7.4.1.5.3 (CSI-RS/TRS), 7.4.1.2.2 (PT-RS), 7.4.2.2.2 (PSS) and 7.4.2.3.2 (SSS) in [6] for details.

An O-DU AAL profile for 5G NR downlink shall specify a set of accelerated functions corresponding to one or more than one physical downlink channel(s) and/or physical downlink signal(s).

In addition to the processing blocks mentioned above, each of these downlink physical channels/signals may include some additional functional blocks (e.g. precoding, IQ compression) which are implementation specific and may also depend on system configurations/capabilities (for example, whether a O-DU is connected to a CAT-A/CAT-B O-RU). Each of these physical channels/signals can be implemented with/without these optional functional blocks. The AALI shall expose to the application whether these functional blocks are supported or not within the AAL implementation.

Figure 5.2 illustrates the building blocks for processing various O-DU PHY layer Uplink (UL) channels and signals (with 7.2-x functional split between O-DU and O-RU) defined by 3GPP [6] as part of 5G NR specification.

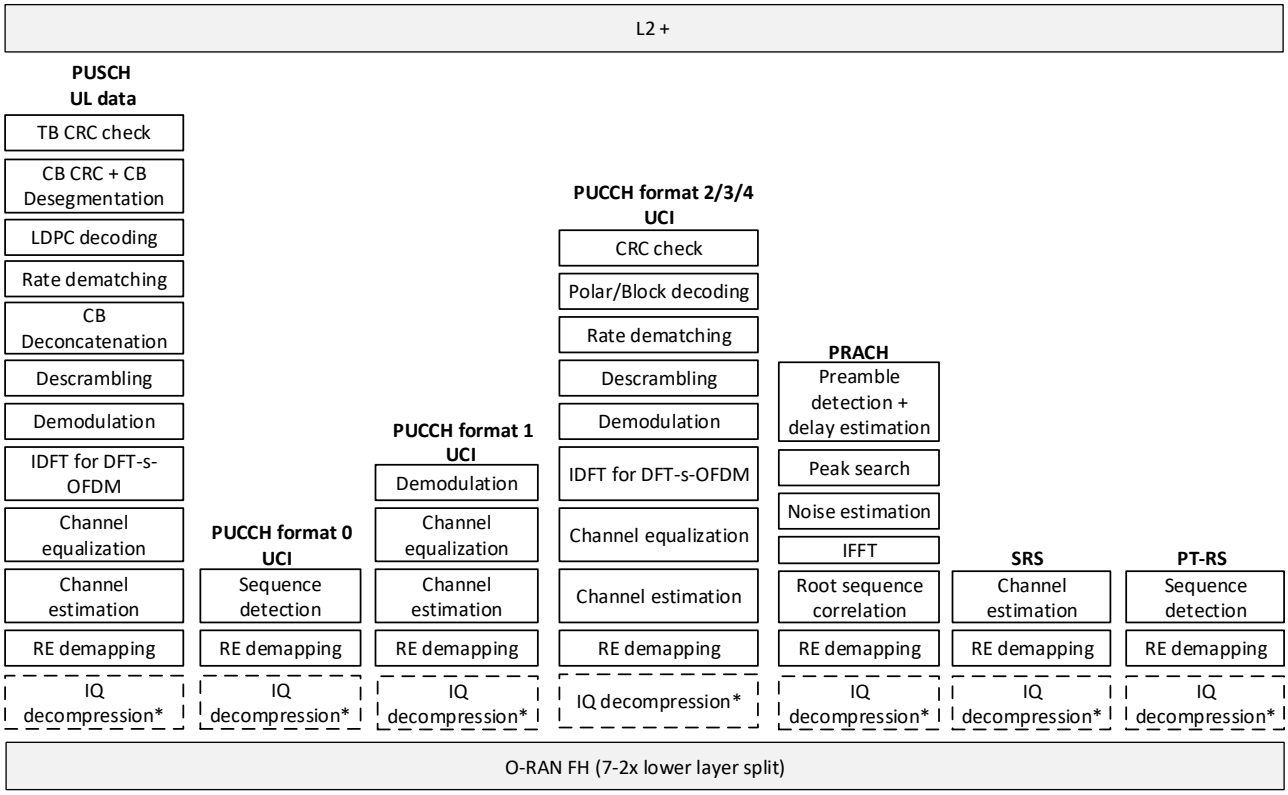


Figure 5.2 O-DU PHY processing blocks for 5G NR Uplink

The O-DU PHY layer in uplink consists of the following physical channels and reference signals:

- Physical Uplink Shared Channel (PUSCH).
- Physical Uplink Control Channels (PUCCH) with formats 0/1/2/3/4.
- Physical Random-Access Channel (PRACH).
- Sounding Reference Signal (SRS).
- Phase Tracking Reference Signal (PT-RS) for UL.

The uplink physical channels (PUSCH, PUCCH, PRACH) carry information originating from higher layers (i.e. layer 2 and above).

The uplink physical layer processing of shared channel (PUSCH) carrying uplink data with or without Uplink Control Information (UCI) consists of the following steps at the receiver (O-DU):

RE de-mapping: Refer to Subclauses 6.3.1.6, 6.3.1.7 and 6.4.1.1.3 of [6] for details on RE mapping at the transmitter.

Channel estimation and equalization: up to O-DU implementation.

Transform precoding (IDFT): optional, only required for DFT-s-OFDM waveform. Refer to Subclause 6.3.1.4 of [6] for details on transform precoding (if applicable) applied at the transmitter.

Demodulation: Refer to Subclause 6.3.1.2 in [6] for details on modulation applied at the transmitter.

Descrambling: Refer to Subclause 6.3.1.1 in [6] for details on scrambling applied at the transmitter.

CB de-concatenation: Refer to Subclause 6.2.6 in [7] for details on CB concatenation applied at the transmitter.

Rate de-matching: Refer to Subclause 6.2.5 in [7] for details on rate matching applied at the transmitter.

LDPC decoding: Refer to Subclause 6.2.4 in [7] for details on LDPC encoding applied at the transmitter.

CB de-segmentation and CB CRC check: Refer to Subclause 6.2.3 in [7] for details on CB segmentation and CB CRC attachment applied at the transmitter.

TB CRC check: Refer to Subclause 6.2.1 in [7] for details on TB level CRC attachments applied at the transmitter.

The uplink physical layer processing for control channel (PUCCH) carrying UCI depends on PUCCH formats.

PUCCH format 0 processing consists of the following steps at the receiver (O-DU):

RE de-mapping: Refer to subclause 6.3.2.3.2 of [6] for details on RE mapping applied at the transmitter.

Sequence detection: The transmitted sequence (refer to Subclause 6.3.2.3 in [6] for details) is detected at O-DU using a non-coherent detector, since PUCCH format 0 does not carry any DM-RS. The detailed design is up to O-DU implementation.

PUCCH format 1 processing consists of the following steps at the receiver (O-DU):

RE de-mapping: Refer to Subclauses 6.3.2.4.2 and 6.4.1.3.1.2 of [6] for details on RE mapping applied at the transmitter.

Channel estimation and equalization: up to O-DU implementation.

Demodulation: Refer to Subclause 6.3.2.4.1 in [6] for details on modulation applied at the transmitter.

PUCCH formats 2/3/4 processing consists of the following steps at the receiver (O-DU):

RE de-mapping: Refer to Subclauses 6.3.2.5.3 and 6.4.1.3.2.2 (format 2); 6.3.2.6.5 and 6.4.1.3.3.2 (formats 3/4) of [6] for details on RE mapping applied at the transmitter.

Channel estimation and equalization: up to O-DU implementation.

Transform precoding (IDFT): optional, only required for DFT-s-OFDM waveform. Refer to Subclause 6.3.2.6.4 of [6] for details on transform precoding (applicable for formats 3/4) applied at the transmitter.

Demodulation: Refer to Subclause 6.3.2.5.2 (format 2) and 6.3.2.6.2 (formats 3/4) in [6] for details on modulation applied at the transmitter.

Descrambling: Refer to Subclause 6.3.2.5.1 (format 2) and 6.3.2.6.1 (formats 3/4) in [6] for details on scrambling applied at the transmitter.

Rate de-matching: Refer to Subclause 6.3.1.4 in [7] for details on rate matching applied at the transmitter.

Polar/Block decoding: Refer to Subclause 6.3.1.3 in [7] for details on Polar/Block encoding applied at the transmitter.

CRC check: Refer to Subclause 6.3.1.2 in [7] for details on CRC attachment applied at the transmitter.

The uplink physical layer processing for random access channel (PRACH) carrying preamble consists of the following steps at the receiver (O-DU):

RE de-mapping: Refer to Subclause 6.3.3.2 in [6] for details on RE mapping applied at the transmitter.

Root sequence correlation: Perform correlation operation between root sequence and received signals. Refer to Subclause 6.3.3.1 in [6] for details on root sequence generation.

IFFT: perform the inverse Fast Fourier Transform (iFFT) operation on the received signal(s).

Noise estimation: perform the noise estimation operation.

Peak search: detect the peak for different root sequences.

Preamble detection and Timing Advance (TA) or delay estimation: determine the preamble sequence(s) received and the corresponding timing advance estimate(s).

The uplink physical signals (SRS, PT-RS) do not carry any information from the higher layers (i.e. layer 2 and above).

The Sounding Reference Signal (SRS) in uplink is received at O-DU using the following steps:

RE de-mapping: Refer to Subclauses 6.4.1.4.3 and 6.4.1.4.4 in [6] for details on RE mapping applied at the transmitter.

Sequence detection and Channel estimation: Up to O-DU implementation. Refer to 6.4.1.4.2 in [6] for details on SRS sequence generation at the transmitter. Channel condition in uplink is estimated at the O-DU based on the processing of received SRS.

The Phase-Tracking Reference Signal (PT-RS) in uplink is received at the O-DU using the following steps:

RE de-mapping: Refer to Subclause 6.4.1.2.2 in [6] for details on RE mapping applied at the transmitter.

Sequence detection: Up to O-DU implementation. Refer to Subclause 6.4.1.2.1 in [6] for details on sequence generation at the transmitter.

An O-DU AAL profile for 5G NR uplink shall specify a set of accelerated functions corresponding to one or more than one physical uplink channel(s) and/or physical uplink signal(s).

In addition to the processing blocks mentioned above, each of these uplink physical channels/signals may include an additional functional block, viz. IQ decompression, which is implementation specific and may depend on system configuration/capability. Each of these physical channels/signals can be implemented with/without this optional functional block. The AALI shall expose to the application whether these functional blocks are supported or not within the AAL implementation.

5.1.2 O-DU Protocol Stack Reference for mMTC

Figure 5.3 illustrates the building blocks for processing various O-DU PHY layer Downlink (DL) channels and signals (with 7.2-x functional split between O-DU and O-RU) defined by 3GPP in [8] & [9] as part of 4G/5G NR specification.

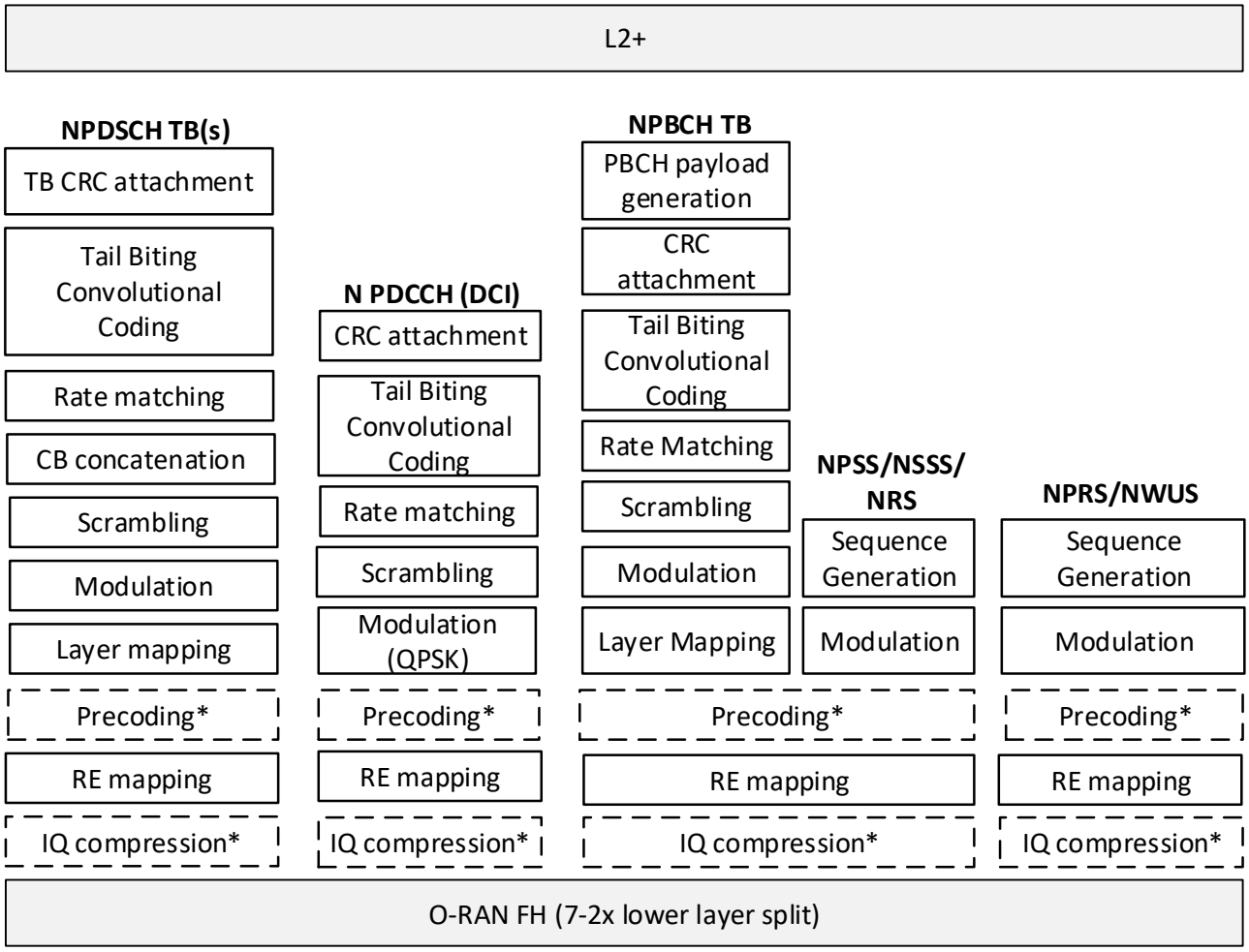


Figure 5.3 O-DU PHY processing blocks for mMTC Downlink

The O-DU PHY layer in downlink consists of the following physical channels and reference signals:

- Narrow-band Physical Downlink Shared Channel (NPDSCH).
- Narrow-band Physical Downlink Control Channel (NPDCCH).
- Narrow-band Physical Broadcast Channel (NPBCH).
- Narrow-band Primary Synchronization Signal (NPSS).
- Narrow-band Secondary Synchronization Signal (NSSS).
- Narrow-band Reference Signal (NRS) and Narrow-band Position Reference Signal (NPRS).
- Narrow-band Wake-Up Signal (NWUS)

The Narrow-band downlink physical channels (NPDSCH, NPDCCH, NPBCH) carry information originating from higher layers (i.e. layer 2 and above).

The Narrow-band downlink physical layer processing of data channel (NPDSCH) carrying transport blocks consists of the following steps:

TB CRC attachment: Error detection is provided on each transport block (TB) through a Cyclic Redundancy Check (CRC). Refer to Subclause 6.4 in [9] for details.

CRC attachment: CRC attachment is performed according to Subclauses 6.4 of [9]

Tail-biting Convolutional coding: Refer to Subclauses 6.2 and 5.1.3.1 in [9] for details.

Rate matching: Refer to Subclauses 6.4 in [9] for details.

Scrambling: Refer to Subclause 10.2.5.2 in [8] for details.

Modulation: Refer to Subclause 10.2.5.3 in [8] for details.

Layer mapping: Refer to Subclause 10.2.5.3in [8] for details.

RE mapping: Refer to Subclause 10.2.5.5in [8] for details on Resource Element (RE) mapping.

The downlink physical layer processing of control channel (PDCCH) carrying Downlink Control Information (DCI) consists of the following steps:

CRC attachment: Error detection is provided on DCI transmissions through a Cyclic Redundancy Check (CRC). Refer to Subclause 6.4 in [9] for details.

Tail-biting Convolutional coding: Refer to Subclauses 6.2 and 5.1.3.1 in [9] for details.

Scrambling: Refer to Subclause 10.2.5.2 in [8] for details.

Modulation: Refer to Subclause 10.2.5.3 in [8] for details.

Layer mapping: Refer to Subclause 10.2.5.3 in [8] for details.

RE mapping: Refer to Subclause 10.2.5.5 in [8] for details on Resource Element (RE) mapping.

The downlink physical layer processing of broadcast channel (NPBCH) carrying maximum one transport block consists of the following steps:

NPBCH payload generation: Refer to Subclause 6.4.1 in [9] for details.

TB CRC attachment: Error detection is provided on each transport block (TB) through a Cyclic Redundancy Check (CRC). Refer to Subclause 6.4 in [9] for details.

Scrambling: Refer to Subclause 10.2.5.2 in [8] for details.

Modulation: Refer to Subclause 10.2.5.3 in [8] for details.

Layer mapping: Refer to Subclause 10.2.5.3 in [8] for details.

RE mapping: Refer to Subclause 10.2.5.5 in [8] for details on Resource Element (RE) mapping.

The downlink physical signals (NRS, NPSS, NSSS, NPRS, NWUS) correspond to a set of resource elements used by the physical layer but does not carry information originated from higher layers (i.e. layer 2 and above).

Reference Signals and Synchronization signals (NPSS/NSSS) are generated using the following steps:

Sequence Generation and Modulation and RE mapping : Refer to Subclauses, 10.2.6B (NWUS), 10.2.7.1 (NPSS) and 10.2.7.2 (NSSS) , 10.2.6 (NRS), 10.2.6A (NPRS) in [8] for details.

An O-DU AAL profile for 4G NR downlink shall specify a set of accelerated functions corresponding to one or more than one physical downlink channel(s) and/or physical downlink signal(s).

In addition to the processing blocks mentioned above, each of these downlink physical channels/signals may include some additional functional blocks (e.g. precoding, IQ compression) which are implementation specific and may also depend on system configurations/capabilities (for example, whether a O-DU is connected to a CAT-A/CAT-B O-RU). Each of these physical channels/signals can be implemented with/without these optional functional blocks. The AALI shall expose to the application whether these functional blocks are supported or not within the AAL implementation.

Figure 5.4 illustrates the building blocks for processing various O-DU PHY layer Uplink (UL) channels and signals (with 7.2-x functional split between O-DU and O-RU) defined by 3GPP in [8] & [9] as part of 4G/5G NR specification.

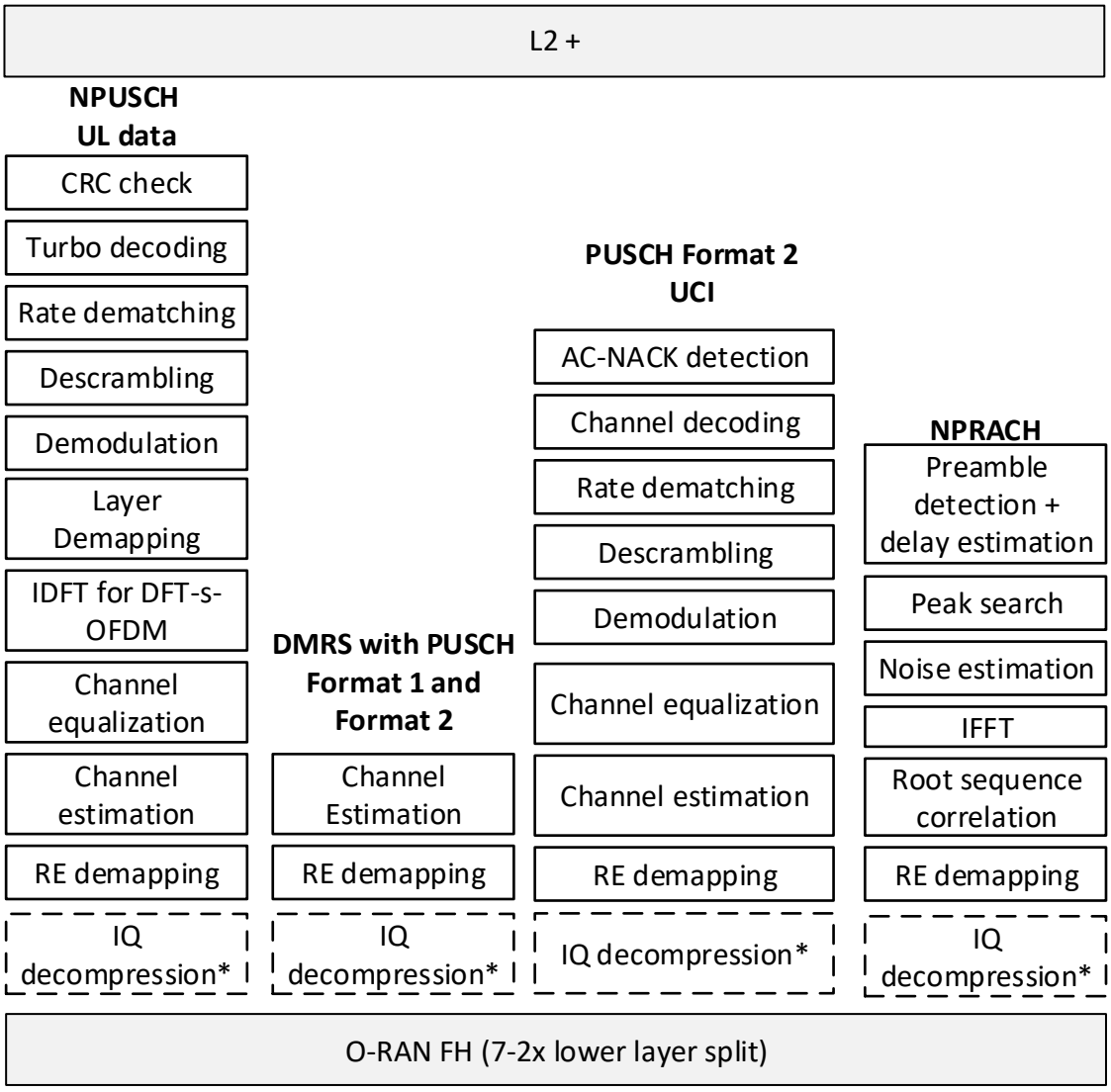


Figure 5.4 O-DU PHY processing blocks for mMTC Uplink

The O-DU PHY layer in uplink consists of the following physical channels and reference signals:

- Narrow-band Physical Uplink Shared Channel (NPUSCH).
- Narrow-band Physical Random-Access Channel (NPRACH).

The uplink physical channels (NPUSCH, NPRACH) carry information originating from higher layers (i.e. layer 2 and above).

The uplink physical layer processing of shared channel (NPUSCH) carrying uplink data with or without Uplink Control Information (UCI) consists of the following steps at the receiver (O-DU):

RE (de)mapping: Refer to Subclauses 5.3.4 of [8] for details on RE mapping at the transmitter/receiver.

Channel estimation and equalization: up to O-DU implementation.

Transform precoding (IDFT): optional, only required for DFT-s-OFDM waveform. Refer to Subclause 5.3.3A of [8] for details on transform precoding (if applicable) applied at the transmitter.

Demodulation: Refer to Subclause 5.3.2 in [8] for details on modulation applied at the transmitter.

Descrambling: Refer to Subclause 5.3.1 in [8] for details on scrambling applied at the transmitter.

Rate de-matching: Refer to Subclause 5.1.4.1 in [9] for details on rate matching applied at the transmitter.

Turbo decoding: Refer to Subclause 5.1.3.2 in [9] for details on Turbo decoding applied at the transmitter.

CRC check: Refer to Subclauses 5.1.1 in [9] for details on TB and CB level CRC attachments applied at the transmitter.

The uplink physical layer processing for shared channel (NPUSCH) carrying UCI depends on NPUSCH formats.

An O-DU AAL profile for 4G/5G NR uplink shall specify a set of accelerated functions corresponding to one or more than one physical uplink channel(s) and/or physical uplink signal(s).

In addition to the processing blocks mentioned above, each of these uplink physical channels/signals may include an additional functional block, viz. IQ decompression, which is implementation specific and may depend on system configuration/capability. Each of these physical channels/signals can be implemented with/without this optional functional block. The AALI shall expose to the application whether these functional blocks are supported or not within the AAL implementation.

5.1.3 O-DU AAL Profile Definitions

O-DU AAL profiles are defined below with future specification(s) to define the AALI for each profile.

5.1.3.1 Profile Definitions General Guidelines

5.1.3.1.1 Naming

As discussed above O-DU AAL profiles are specific to one or more physical channel(s) or signal(s) as such should follow the naming guidelines

- O-DU AAL profiles shall be prefixed with “AAL_”
- O-DU AAL Profiles when specific to a single channel or signal shall include the channel or signal in the name e.g. “AAL_PUSCH”
- O-DU AAL Profiles when common across multiple channels or signals shall not include the channel or signal name, instead just reference the Accelerated Function(s), e.g. AAL_RE-MAPPING
- O-DU AAL Profiles that include a subset of the functional blocks within a channel or signal shall include a functional description after the channel name e.g. AAL_PUSCH_CHANNEL_ESTIMATION

5.1.3.1.2 Data Flow

O-DU AAL Profiles shall specify the data flow supported by the profile as discussed in section 2.5.1.7 and 2.5.1.8 above e.g. Look aside, Inline or other.

Look aside data flow implies the remaining functions not included in the Profile that comprise the channel or signal are implemented in SW on the host CPU and not implemented in the HW Accelerator.

Inline data flow implies that the set of accelerated functions is constituted of the entire U-plane processing of high-PHY channel or signal (with 7-2x PHY functional split) and the IQ data (in DL) or decoded bits (in UL) (post processing) are transferred directly from the accelerator to the Fronthaul interface. (in DL) or Layer 2 (in UL).

The profile shall specify if the data flow includes only user plane, or only control plane, or both.

5.1.3.2 O-DU AAL Profiles for Downlink

5.1.3.2.1 AAL_PDSCH_FEC

Figure 5.5 highlights the set of accelerated functions that define the AAL_PDSCH_FEC Profile. These include

- CRC Generation
- LDPC Encoding
- PDSCH Rate Matching

The AAL_PDSCH_FEC Profile is implemented as a look aside accelerator. The AAL_PDSCH_FEC Profile will support both Transport Block and Code Block operations.

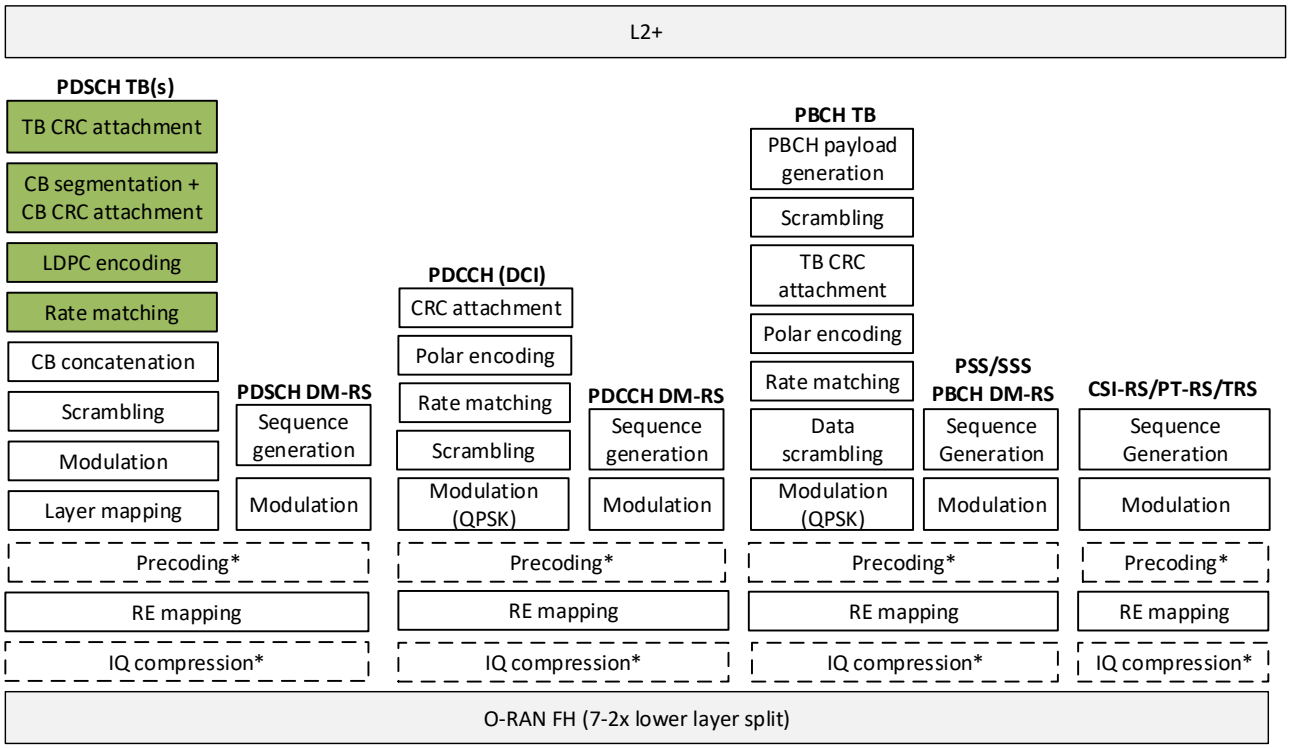


Figure 5.5 AAL_PDSCH_FEC Profile

5.1.3.2.2 AAL_PDSCH_HIGH-PHY

Figure 5.6 highlights the set of accelerated functions that defines the AAL_PDSCH_HIGH-PHY Profile, which includes the processing of PDSCH TB(s) and associated DM-RS.

The set of accelerated functions associated with the processing of PDSCH TB(s) is as follows:

- TB CRC attachment
- CB segmentation and CRC attachment
- LDPC encoding
- Rate matching
- CB concatenation
- Scrambling
- Modulation
- Layer mapping
- Precoding [1]
- RE mapping

[1] Configurable functional block, depends on implementation and/or system configuration

- IQ compression[1]

The set of accelerated functions associated with the processing of PDSCH DM-RS is as follows:

- PDSCH DM-RS sequence generation
- Modulation
- Precoding[1]
- RE mapping
- IQ compression[1]

The AAL_PDSCH_HIGH-PHY Profile is executed in inline acceleration mode.

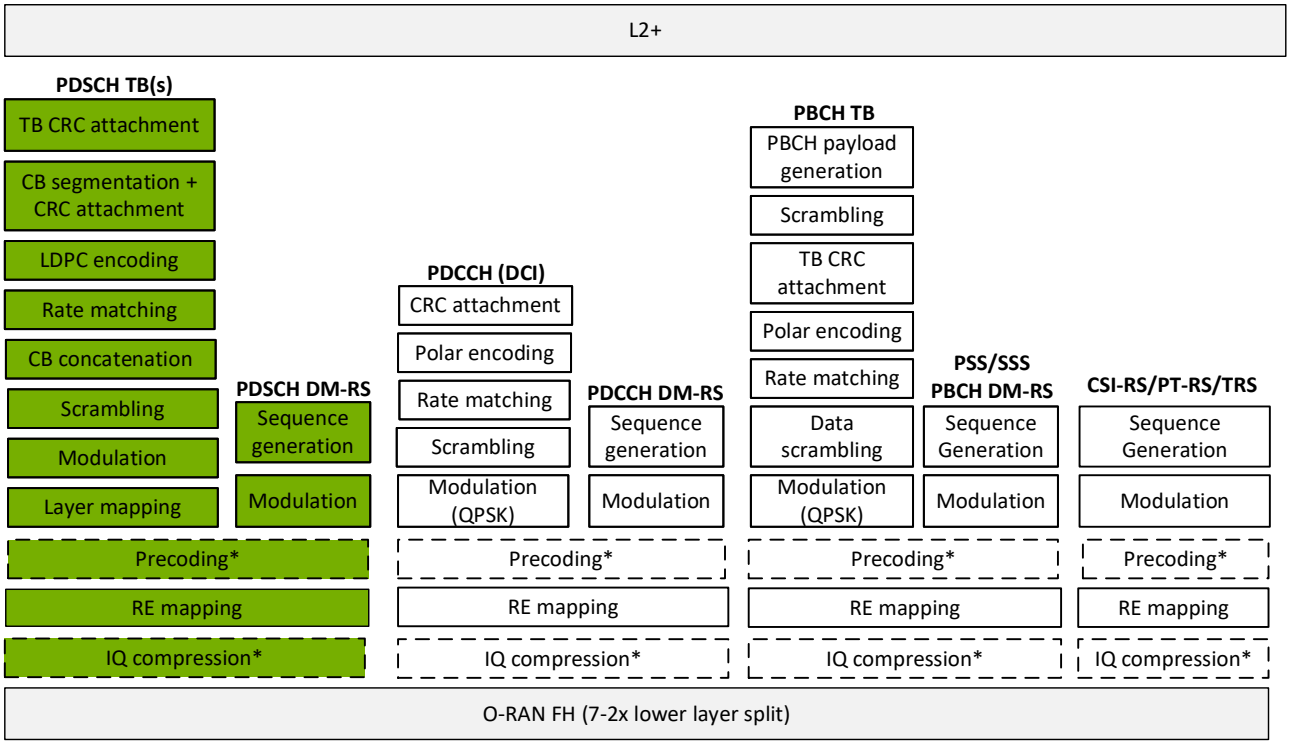


Figure 5.6 AAL_PDSCH_HIGH-PHY Profile

5.1.3.2.3 AAL_PDCCH_HIGH-PHY

Figure 5.7 highlights the set of accelerated functions that defines the AAL_PDCCH_HIGH-PHY Profile, which includes the processing of PDCCH DCI and associated DM-RS.

The set of accelerated functions associated with the processing of PDCCH TB(s) is as follows:

- CRC attachment
- Polar encoding
- Rate matching
- Scrambling

- Modulation (QPSK)
- Precoding[1]
- RE mapping
- IQ compression[1]

The set of accelerated functions associated with the processing of PDCCH DM-RS is as follows:

- PDCCH DM-RS sequence generation
- Modulation
- Precoding[1]
- RE mapping
- IQ compression[1]

The AAL_PDCCH_HIGH-PHY Profile is executed in inline acceleration mode.

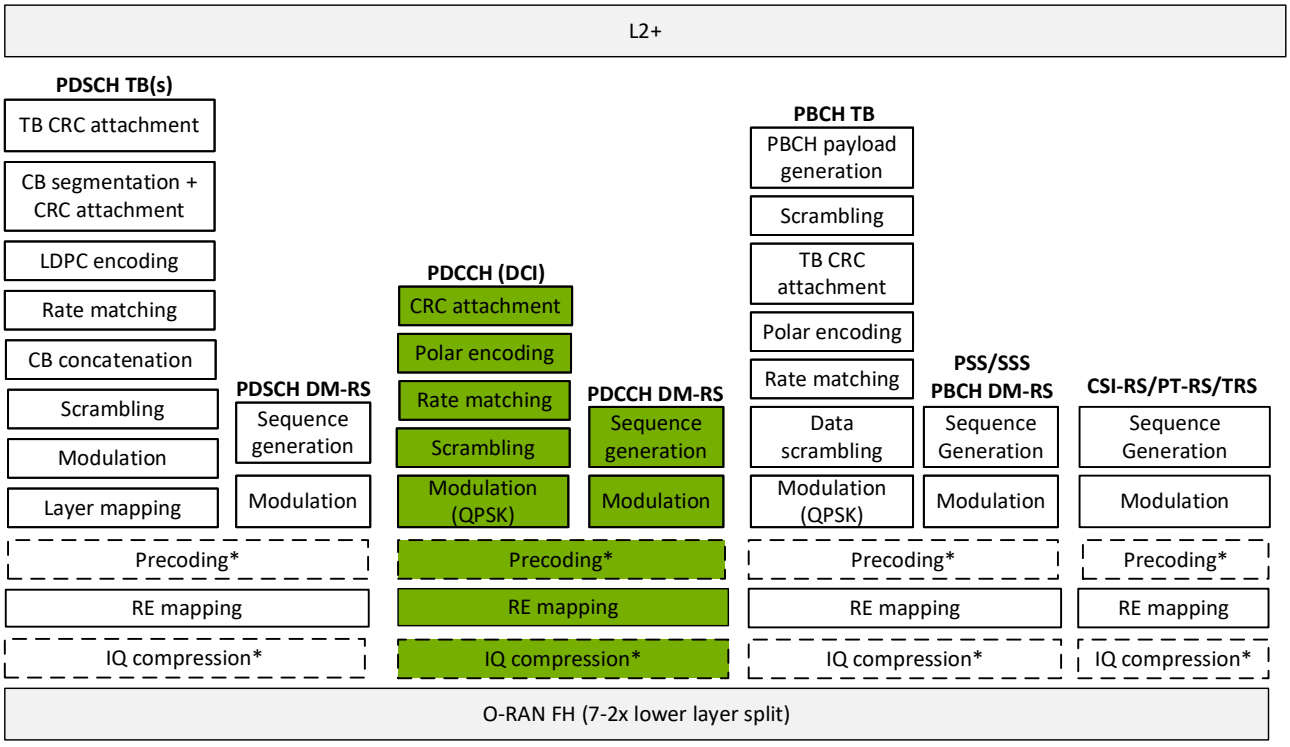


Figure 5.7 AAL_PDCCH_HIGH-PHY Profile

5.1.3.2.4 AAL_PBCH_HIGH-PHY

Figure 5.8 highlights the set of accelerated functions that defines the AAL_PBCH_HIGH-PHY Profile, which includes the processing of PBCH TB and associated DM-RS, PSS and SSS, or in other words, the processing of SSB.

The set of accelerated functions associated with the processing of PBCH TB is as follows:

- PBCH payload generation
- Scrambling
- TB CRC attachment

- Polar encoding
- Rate matching
- Data scrambling
- Modulation (QPSK)
- Precoding[1]
- RE mapping
- IQ compression[1]

The set of accelerated functions associated with the processing of PBCH DM-RS/PSS/SSS is as follows:

- PDCCH DM-RS/PSS/SSS sequence generation
- Modulation
- Precoding[1]
- RE mapping
- IQ compression[1]

The AAL_PBCH_HIGH-PHY Profile is executed in inline acceleration mode.

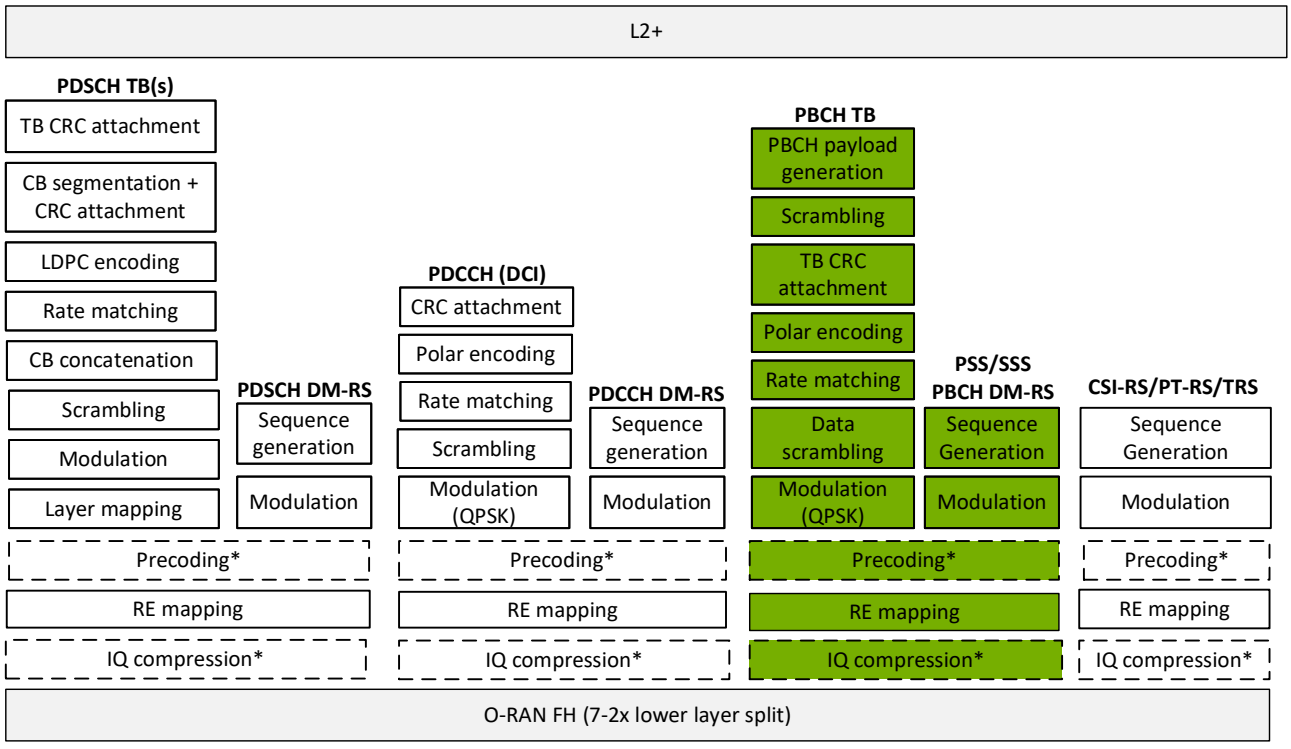


Figure 5.8 AAL_PBCH_HIGH-PHY Profile

5.1.3.2.5 AAL_CSI-RS_HIGH-PHY

Figure 5.9 highlights the set of accelerated functions that defines the AAL_CSI-RS_HIGH-PHY Profile, which includes the following:

- CSI-RS sequence generation
- Modulation
- Precoding[1]
- RE mapping
- IQ compression[1]

The AAL_CSI-RS_HIGH-PHY Profile is executed in inline acceleration mode.

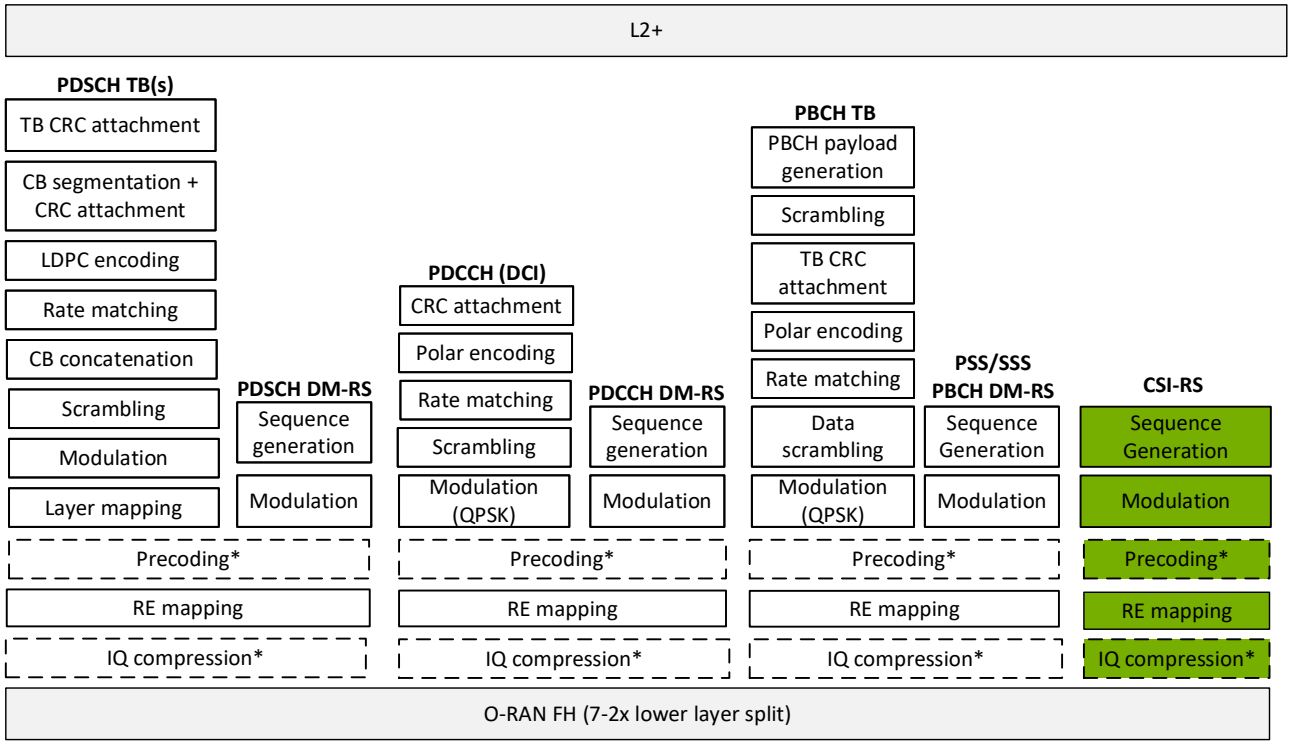


Figure 5.9 AAL_CSI-RS_HIGH-PHY Profile

5.1.3.2.6 AAL_PT-RS-DL_HIGH-PHY

Figure 5.10 highlights the set of accelerated functions that defines the AAL_PT-RS-DL_HIGH-PHY Profile, which includes the following:

- PT-RS sequence generation
- Modulation
- Precoding[1]
- RE mapping
- IQ compression[1]

The AAL_PT-RS-DL_HIGH-PHY Profile is executed in inline acceleration mode.

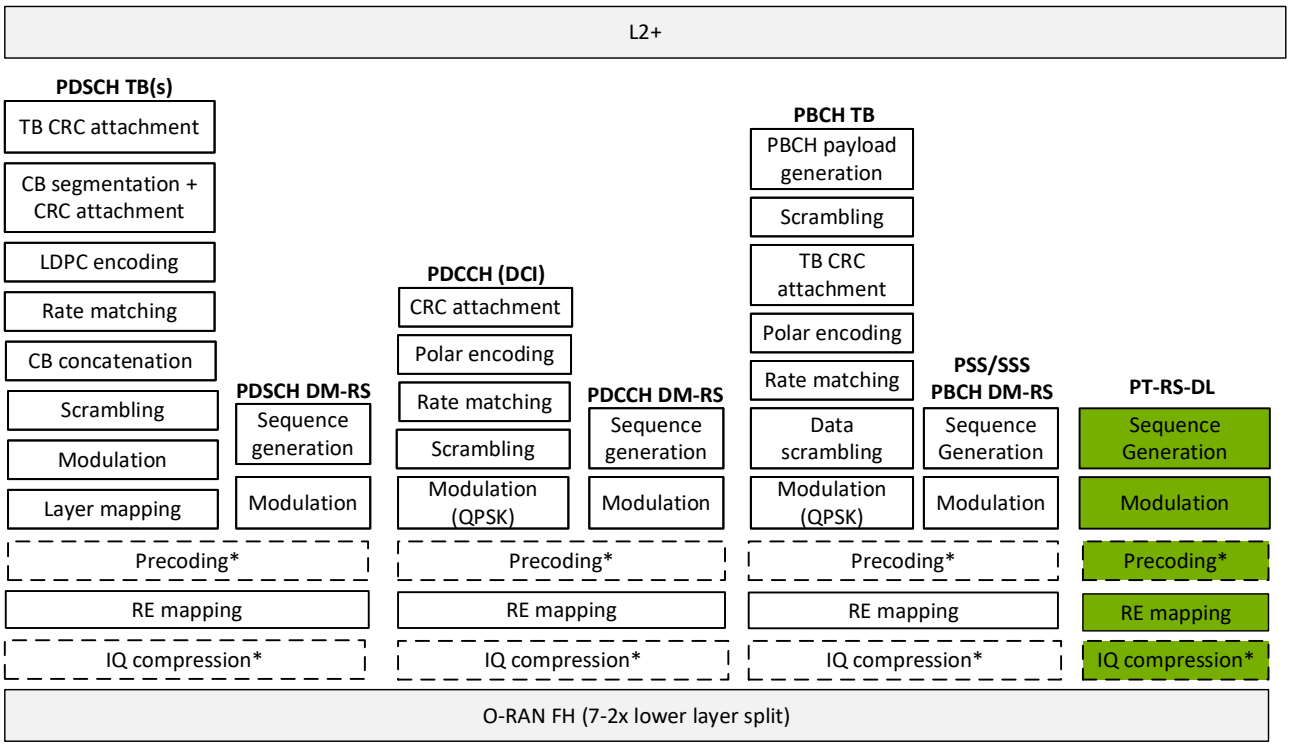


Figure 5.10 AAL_PT-RS-DL_HIGH-PHY Profile

5.1.3.3 O-DU AAL Profiles for Uplink

5.1.3.3.1 AAL_PUSCH_FEC

Figure 5.11 highlights the set of accelerated functions that define the AAL_PUSCH_FEC Profile. These include

- PUSCH Rate De-matching
- LDPC Decoder
- CRC Check

The AAL_PUSCH_FEC Profile is implemented as a look aside accelerator. The AAL_PUSCH_FEC Profile will support both Transport Block and Code Block operations.

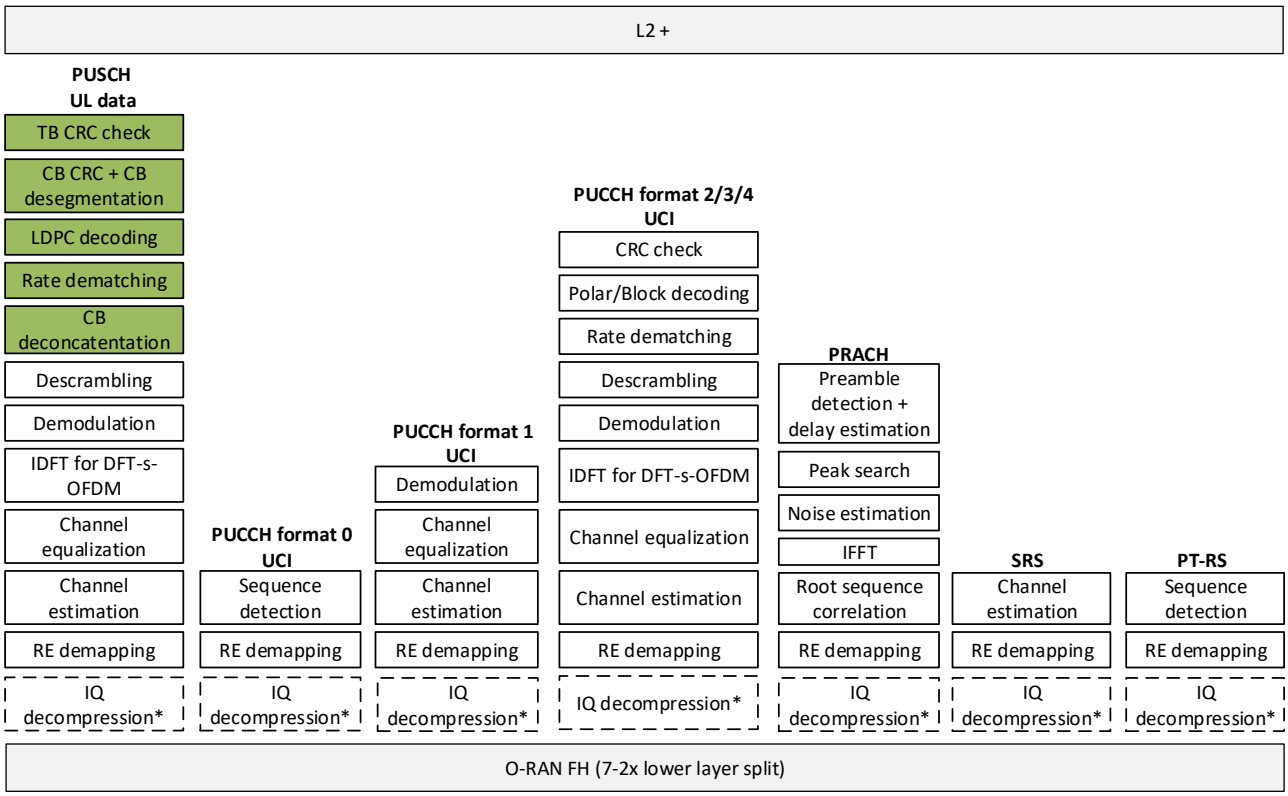


Figure 5.11 AAL_PUSCH_FEC Profile

5.1.3.3.2 AAL_PUSCH_HIGH-PHY

Figure 5.12 highlights the set of accelerated functions that defines the AAL_PUSCH_HIGH-PHY Profile, which includes the processing of PUSCH data (with or without UCI).

The set of accelerated functions associated with the processing of PUSCH data is as follows:

- IQ decompression[1]
- RE de-mapping
- Channel estimation
- Channel equalization
- Transform precoding (optional- only required for DFT-s-OFDM waveform)
- Demodulation
- Descrambling
- Rate de-matching
- LDPC decoding
- CRC check

The AAL_PUSCH_HIGH-PHY Profile is executed in inline acceleration mode.

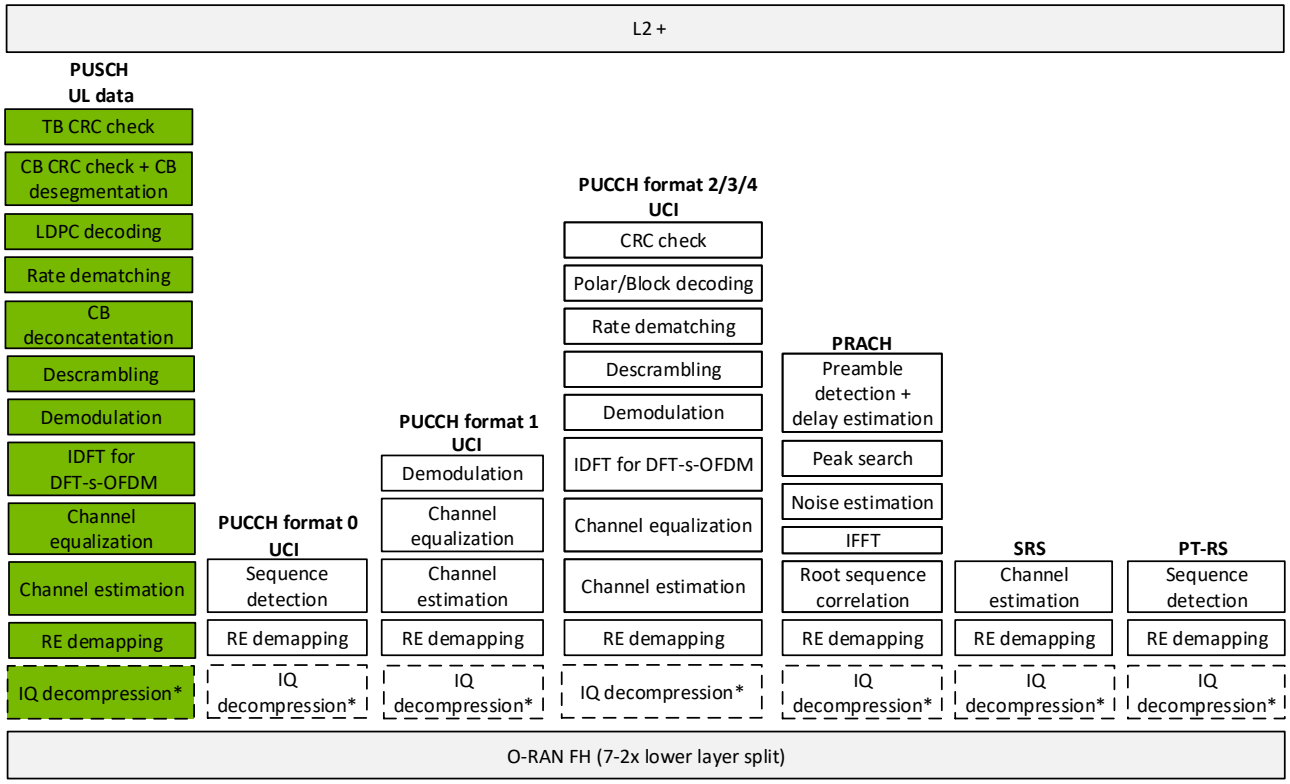


Figure 5.12 AAL_PUSCH_HIGH-PHY Profile

5.1.3.3.3 AAL_PUCCH_HIGH-PHY

Figure 5.13, Figure 5.14 and Figure 5.15 highlight the set of accelerated functions that defines the AAL_PUCCH_HIGH-PHY Profile, which includes the processing of UCI.

The set of accelerated functions associated with the processing of PUCCH UCI depends on the PUCCH format being configured by the application.

5.1.3.3.3.1 PUCCH format 0

The set of accelerated functions associated with the processing of PUCCH UCI using PUCCH format 0 is as follows:

- IQ decompression[1]
- RE de-mapping
- Sequence detection

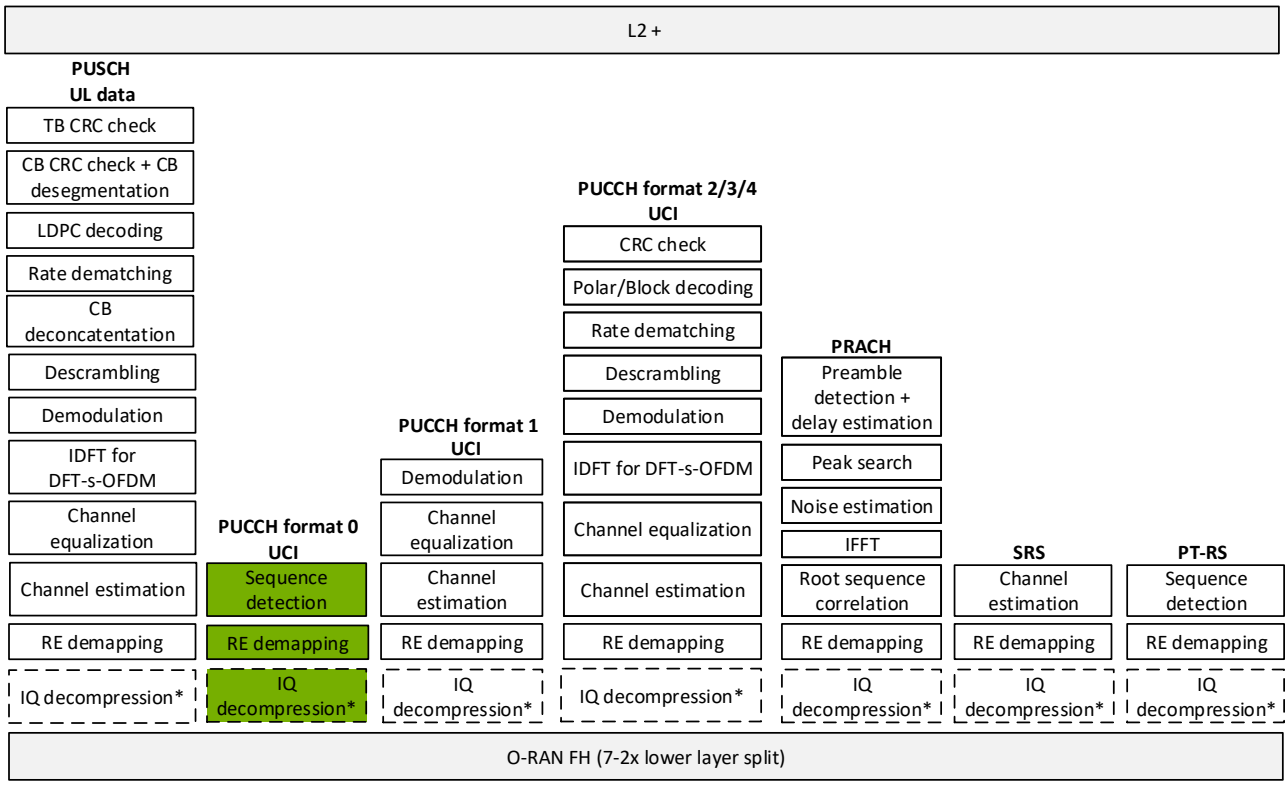


Figure 5.13 AAL_PUCCH_HIGH-PHY Profile (PUCCH format 0)

5.1.3.3.3.2 PUCCH format 1

The set of accelerated functions associated with the processing of PUCCH UCI using PUCCH format 1 is as follows:

- IQ decompression[1]
- RE de-mapping
- Channel estimation
- Channel equalization
- Demodulation

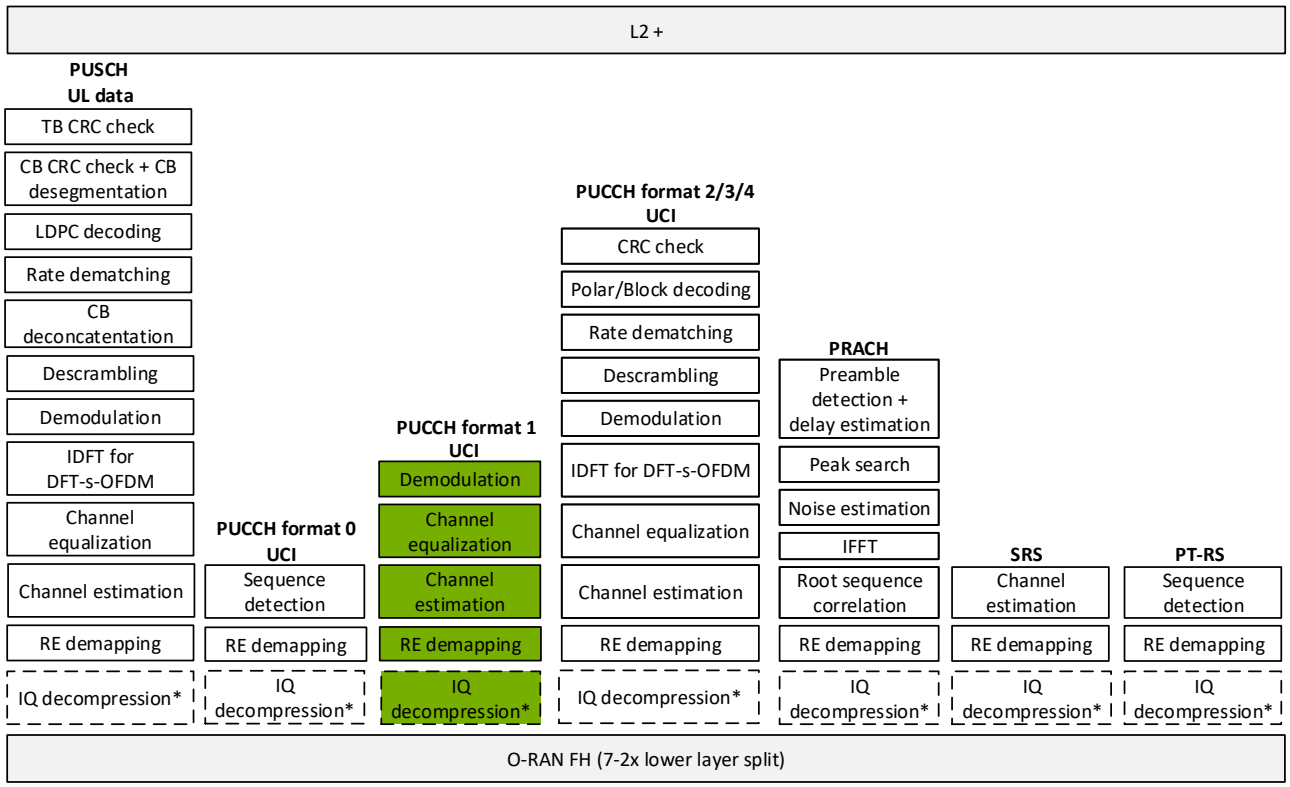


Figure 5.14 AAL_PUCCH_HIGH-PHY Profile (PUCCH format 1)

5.1.3.3.3.3 PUCCH format 2/3/4

The set of accelerated functions associated with the processing of PUCCH UCI using PUCCH format 2/3/4 is as follows:

- IQ decompression[1]
- RE de-mapping
- Channel estimation
- Channel equalization
- Transform precoding (optional- only required for DFT-s-OFDM waveform)
- Demodulation
- Descrambling
- Rate de-matching
- Polar/Block decoding
- CRC check

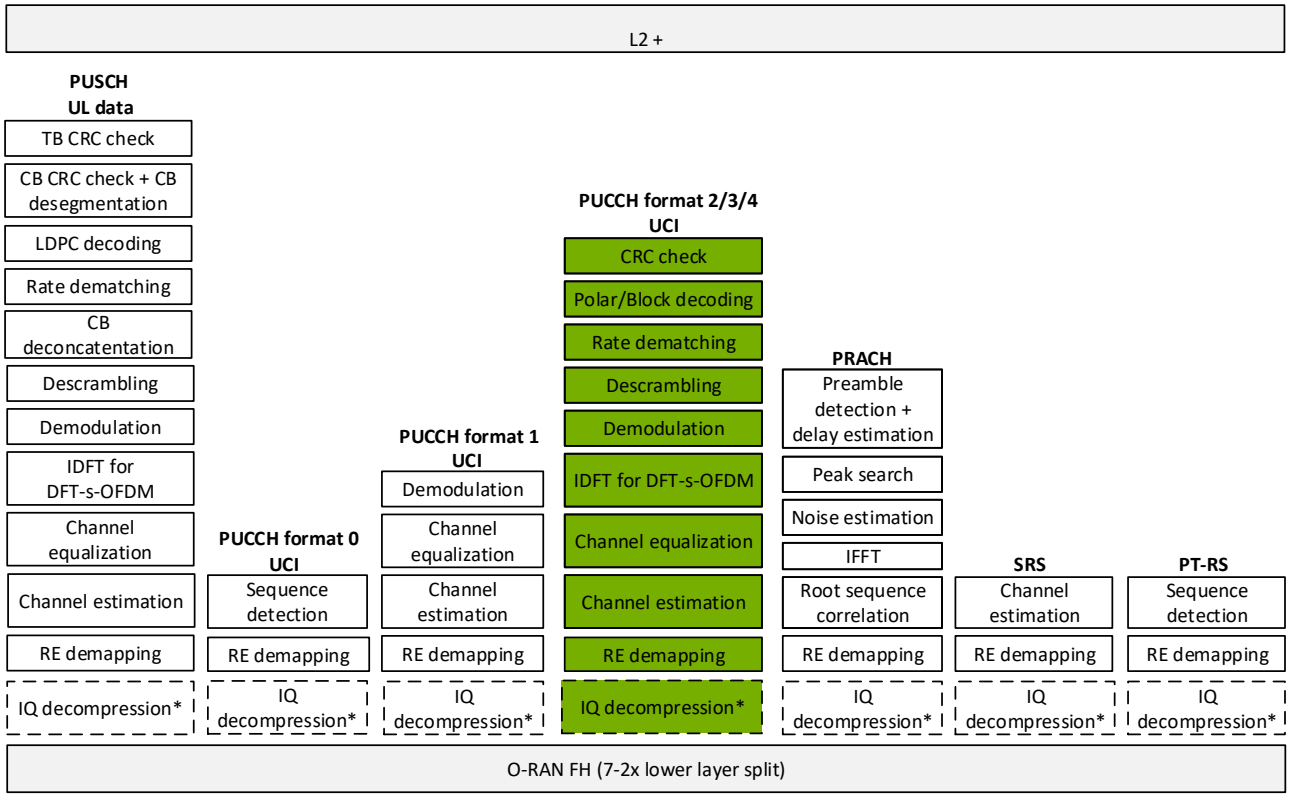


Figure 5.15 AAL_PUCCH_HIGH-PHY Profile (PUCCH format 2/3/4)

The AAL_ PUCCH_ HIGH -PHY profile is executed in inline acceleration mode.

5.1.3.3.4 AAL_PRACH_HIGH-PHY

Figure 5.16 highlights the set of accelerated functions that defines the AAL_PRACH_HIGH-PHY Profile.

The set of accelerated functions associated with the processing of PRACH preamble is as follows:

- IQ decompression[1]
- RE de-mapping
- Root sequence generation and correlation
- IFFT
- Noise estimation
- Peak search for power delay profile
- Preamble detection and delay/timing advance estimation

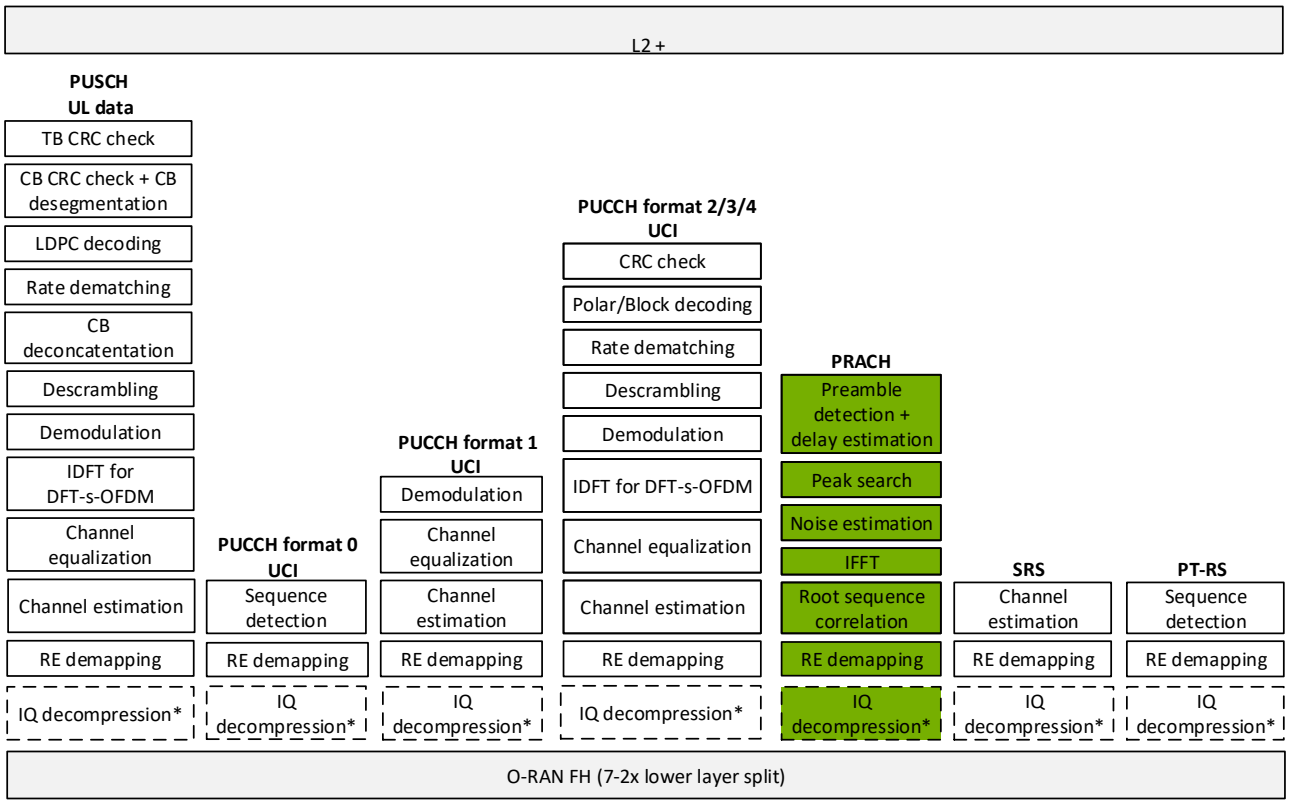


Figure 5.16 AAL_PRACH_HIGH-PHY Profile

The AAL_PRACH_HIGH-PHY Profile is executed in inline acceleration mode.

5.1.3.3.5 AAL_SRS_HIGH-PHY

Figure 5.17 highlights the set of accelerated functions that defines the AAL_SRS_HIGH-PHY Profile.

The set of accelerated functions associated with the processing of SRS is as follows:

- IQ decompression[1]
- RE de-mapping
- Channel estimation

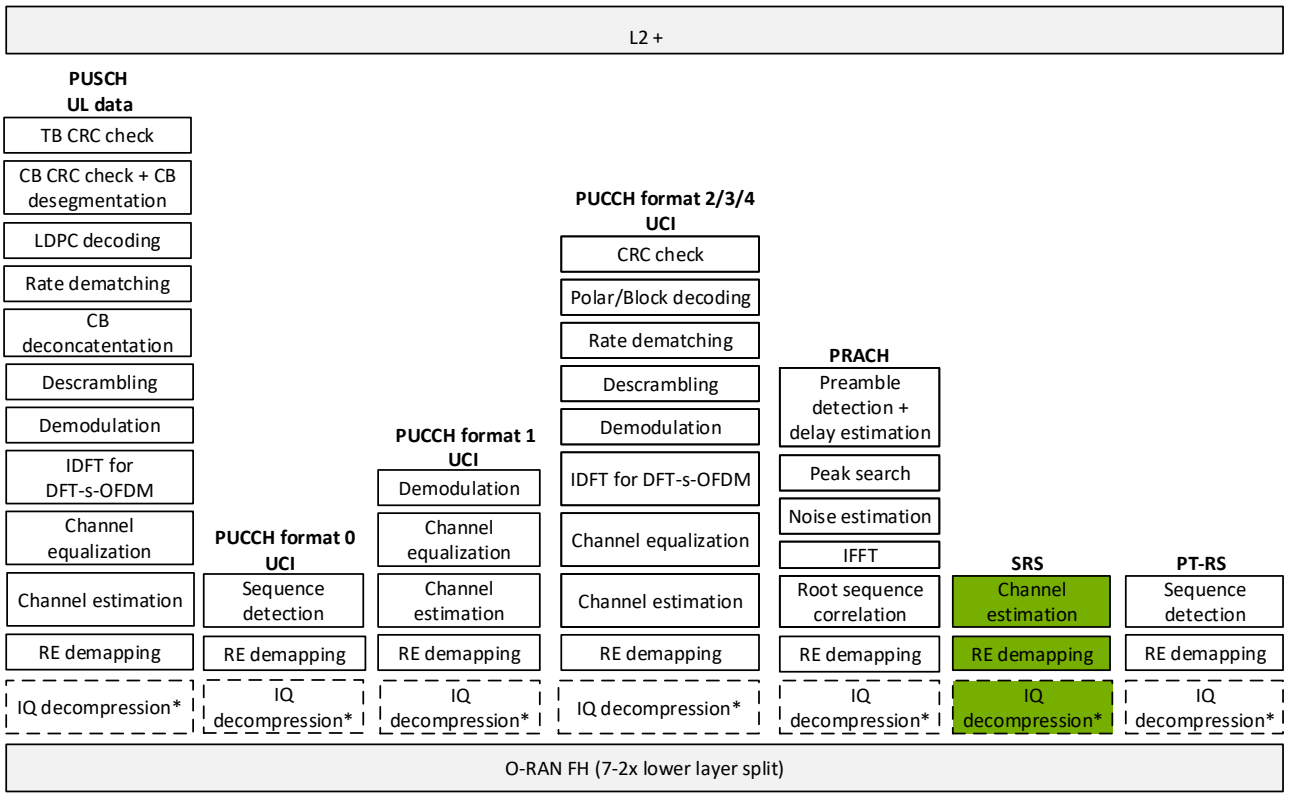


Figure 5.17 AAL_SRS_HIGH-PHY Profile

The AAL_SRS_ HIGH-PHY Profile is executed in inline acceleration mode.

5.1.3.3.6 AAL_PT-RS-UL_HIGH-PHY

Figure 5.18 highlights the set of accelerated functions that defines the AAL_PT-RS-UL_HIGH-PHY Profile.

The set of accelerated functions associated with the processing of PT-RS-UL sequence is as follows:

- IQ decompression[1]
- RE de-mapping
- Sequence detection

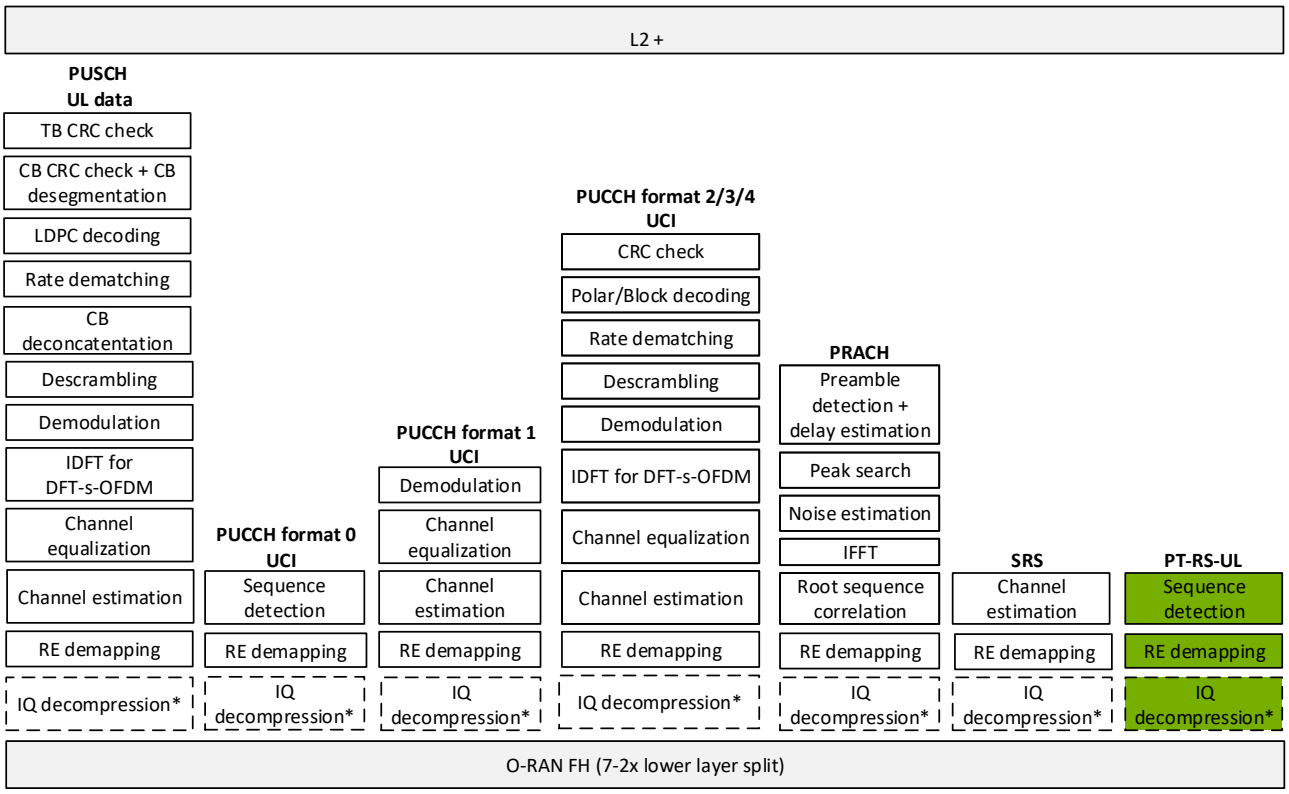


Figure 5.18 AAL_PT-RS-UL_HIGH-PHY profile

The AAL_PT-RS-UL_HIGH-PHY profile is executed in inline acceleration mode.

5.1.3.4 O-DU AAL profiles for mMTC

5.1.3.4.1 AAL_NPDSCH_FEC

Figure 5.19 highlights the set of accelerated functions that define the AAL_NPDSCH_FEC Profile. These include

- CRC Generation
- Tail-Biting Convolutional Coding
- Rate Matching

The AAL_NPDSCH_FEC Profile is implemented as a look aside accelerator.

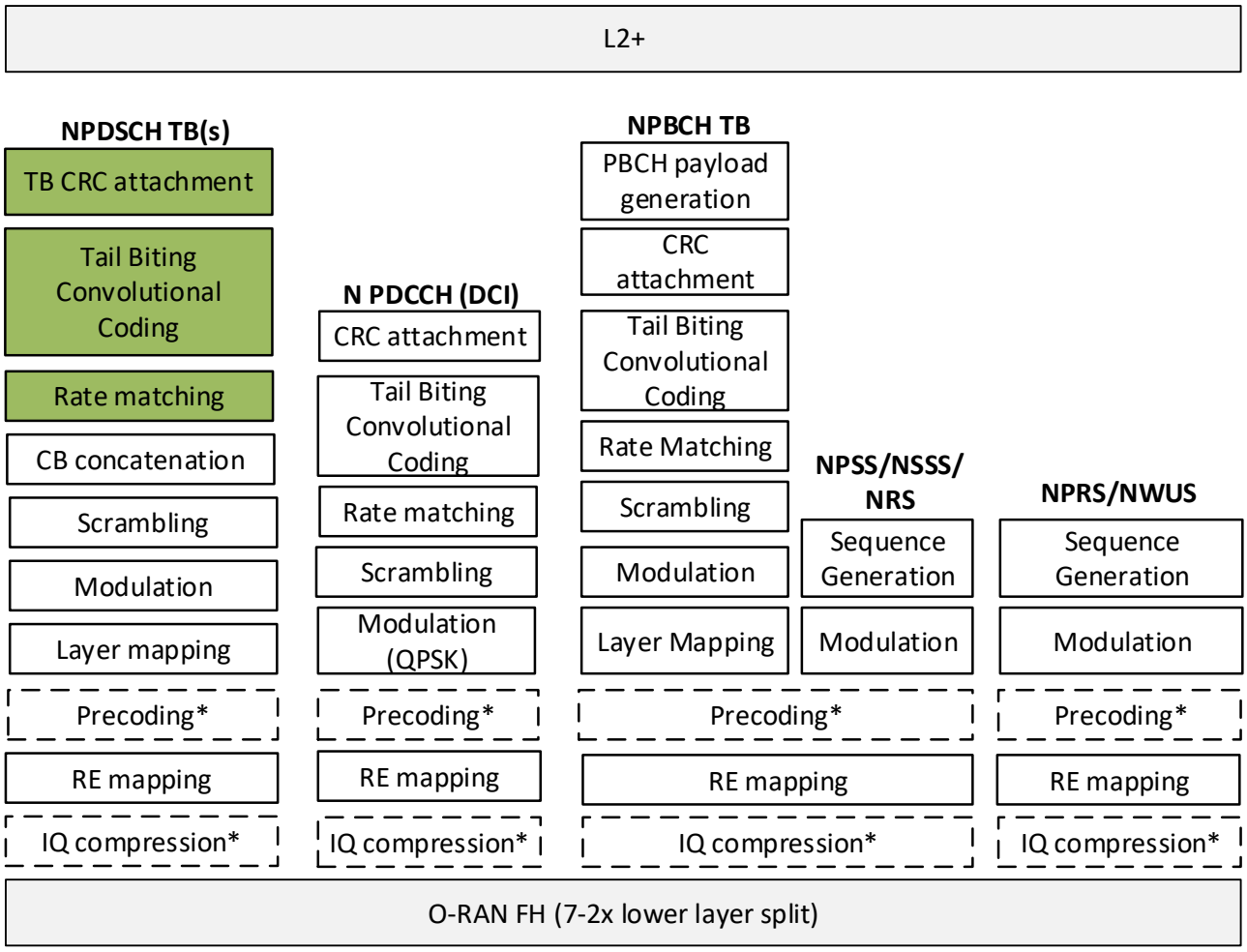


Figure 5.19 AAL_NPDSCH_FEC Profile

5.1.3.4.2 AAL_NPDCCH_FEC

Figure 5.20 highlights the set of accelerated functions that define the AAL_NPDCCH_FEC Profile. These include

- CRC Generation
- Tail-Biting Convolutional Coding
- Rate Matching

The AAL_NPDCCH_FEC Profile is implemented as a look aside accelerator.

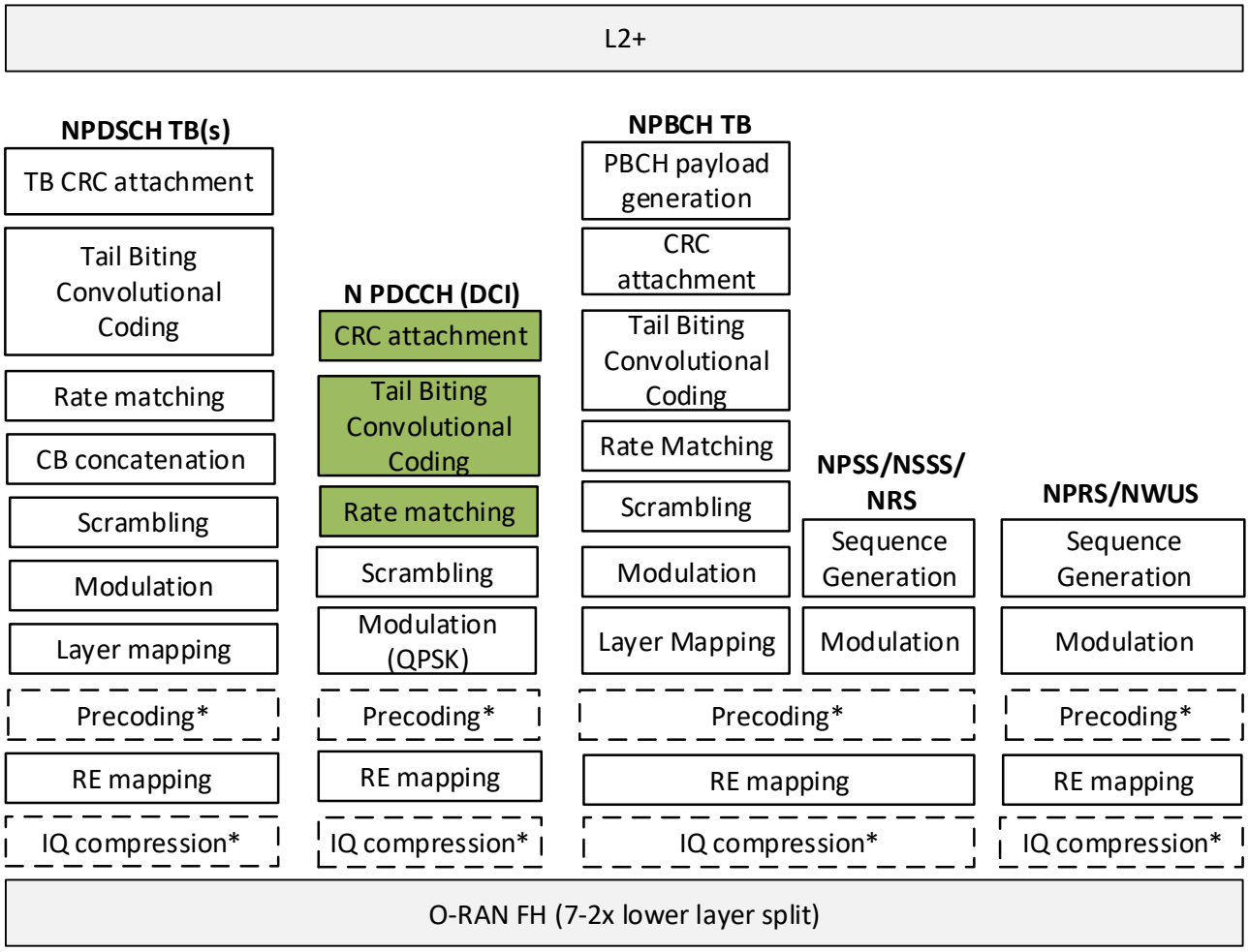


Figure 5.20 AAL_NPDCCH_FEC Profile

5.1.3.4.3 AAL_NPBCH_FEC

Figure 5.21 highlights the set of accelerated functions that define the AAL_NPBCH_FEC Profile. These include

- CRC Generation
- Tail-Biting Convolutional Coding
- Rate Matching

The AAL_NPBCH_FEC Profile is implemented as a look aside accelerator.

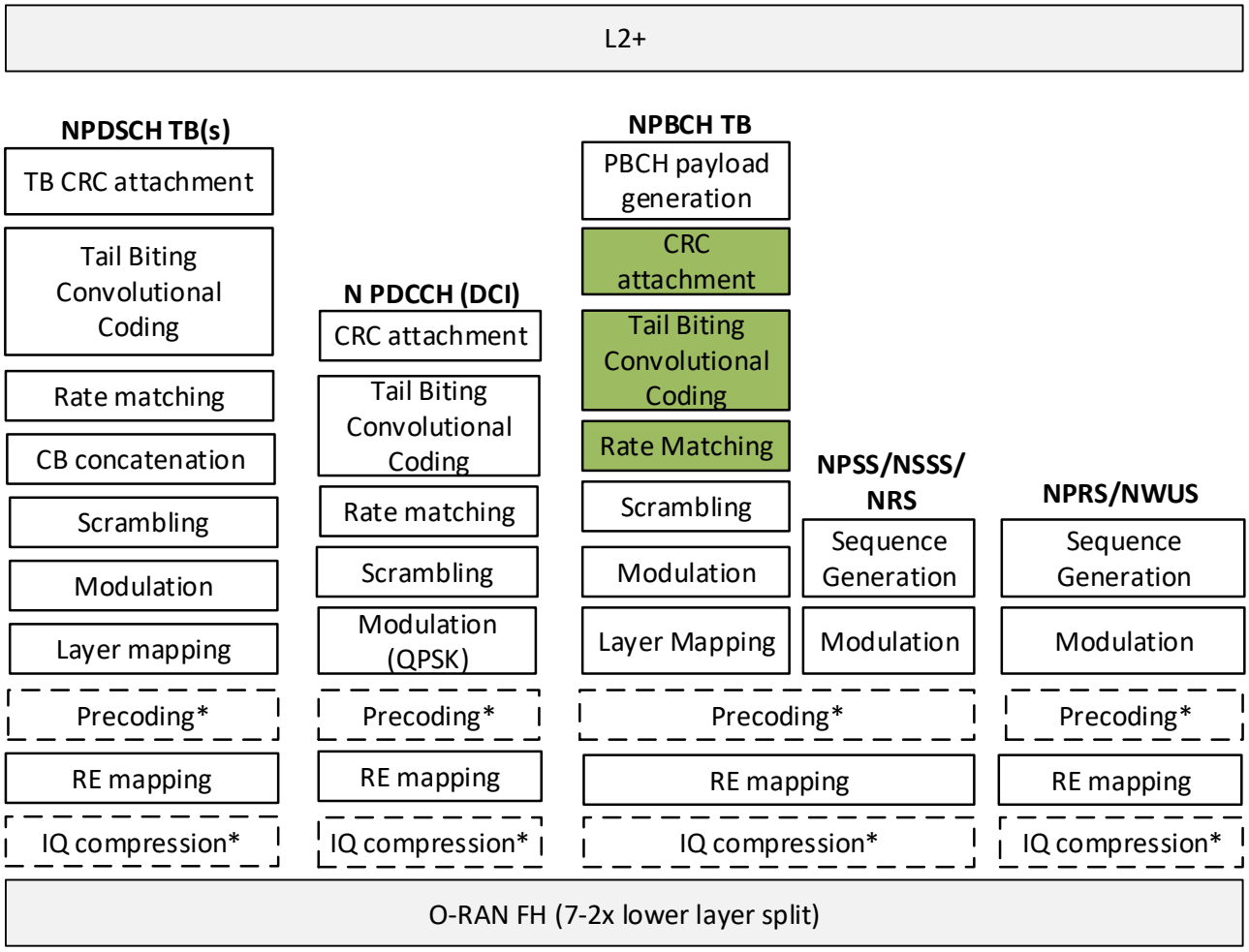


Figure 5.21 AAL_NPBCH_FEC Profile

5.1.3.4.4 AAL_NPUSCH_FEC

Figure 5.22 highlights the set of accelerated functions that define the AAL_NPUSCH_FEC Profile. These include

- CRC Generation
- Turbo Decoding
- Rate Matching

The AAL_NPUSCH_FEC Profile is implemented as a look aside accelerator.

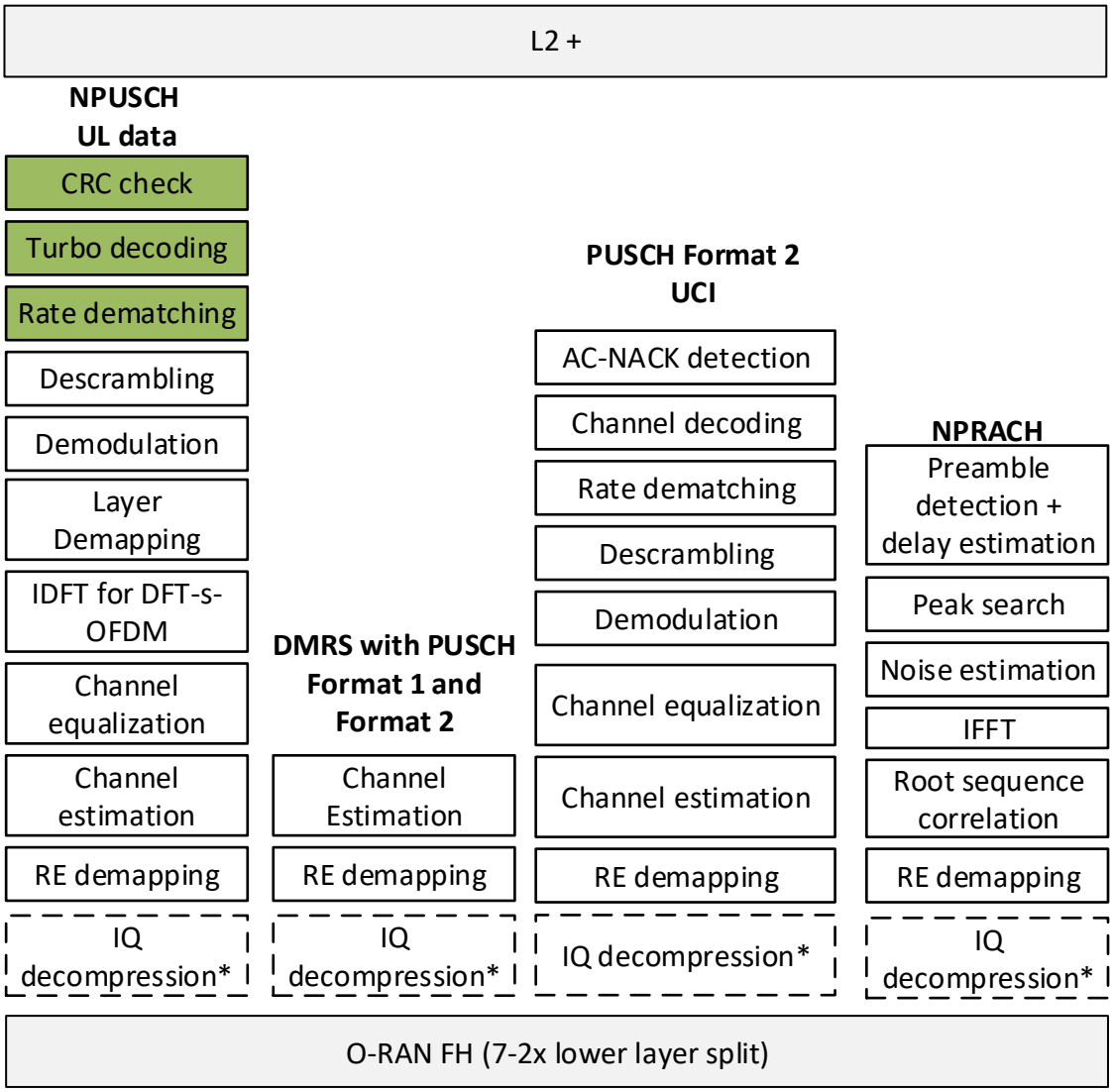


Figure 5.22 AAL_NPUSCH_FEC Profile

5.2 O-CU AAL Profiles

The O-CU AAL profiles shall be part of further study for O-RAN WG6.

Annex ZZZ: O-RAN Adopter License Agreement

BY DOWNLOADING, USING OR OTHERWISE ACCESSING ANY O-RAN SPECIFICATION, ADOPTER AGREES TO THE TERMS OF THIS AGREEMENT.

This O-RAN Adopter License Agreement (the "Agreement") is made by and between the O-RAN ALLIANCE and the entity that downloads, uses or otherwise accesses any O-RAN Specification, including its Affiliates (the "Adopter").

This is a license agreement for entities who wish to adopt any O-RAN Specification.

Section 1: DEFINITIONS

1.1 "Affiliate" means an entity that directly or indirectly controls, is controlled by, or is under common control with another entity, so long as such control exists. For the purpose of this Section, "Control" means beneficial ownership of fifty (50%) percent or more of the voting stock or equity in an entity.

1.2 "Compliant Implementation" means any system, device, method or operation (whether implemented in hardware, software or combinations thereof) that fully conforms to a Final Specification.

1.3 "Adopter(s)" means all entities, who are not Members, Contributors or Academic Contributors, including their Affiliates, who wish to download, use or otherwise access O-RAN Specifications.

1.4 "Minor Update" means an update or revision to an O-RAN Specification published by O-RAN ALLIANCE that does not add any significant new features or functionality and remains interoperable with the prior version of an O-RAN Specification. The term "O-RAN Specifications" includes Minor Updates.

1.5 "Necessary Claims" means those claims of all present and future patents and patent applications, other than design patents and design registrations, throughout the world, which (i) are owned or otherwise licensable by a Member, Contributor or Academic Contributor during the term of its Member, Contributor or Academic Contributorship; (ii) such Member, Contributor or Academic Contributor has the right to grant a license without the payment of consideration to a third party; and (iii) are necessarily infringed by a Compliant Implementation (without considering any Contributions not included in the Final Specification). A claim is necessarily infringed only when it is not possible on technical (but not commercial) grounds, taking into account normal technical practice and the state of the art generally available at the date any Final Specification was published by the O-RAN ALLIANCE or the date the patent claim first came into existence, whichever last occurred, to make, sell, lease, otherwise dispose of, repair, use or operate a Compliant Implementation without infringing that claim. For the avoidance of doubt in exceptional cases where a Final Specification can only be implemented by technical solutions, all of which infringe patent claims, all such patent claims shall be considered Necessary Claims.

1.6 "Defensive Suspension" means for the purposes of any license grant pursuant to Section 3, Member, Contributor, Academic Contributor, Adopter, or any of their Affiliates, may have the discretion to include in their license a term allowing the licensor to suspend the license against a licensee who brings a patent infringement suit against the licensing Member, Contributor, Academic Contributor, Adopter, or any of their Affiliates.

Section 2: COPYRIGHT LICENSE

2.1 Subject to the terms and conditions of this Agreement, O-RAN ALLIANCE hereby grants to Adopter a nonexclusive, nontransferable, irrevocable, non-sublicensable, worldwide copyright license to obtain, use and modify O-RAN Specifications, but not to further distribute such O-RAN Specification in any modified or unmodified way, solely in furtherance of implementations of an ORAN

Specification.

2.2 Adopter shall not use O-RAN Specifications except as expressly set forth in this Agreement or in a separate written agreement with O-RAN ALLIANCE.

Section 3: FRAND LICENSE

3.1 Members, Contributors and Academic Contributors and their Affiliates are prepared to grant based on a separate Patent License Agreement to each Adopter under Fair Reasonable And Non- Discriminatory (FRAND) terms and conditions with or without compensation (royalties) a nonexclusive, non-transferable, irrevocable (but subject to Defensive Suspension), non-sublicensable, worldwide patent license under their Necessary Claims to make, have made, use, import, offer to sell, lease, sell and otherwise distribute Compliant Implementations; provided, however, that such

license shall not extend: (a) to any part or function of a product in which a Compliant Implementation is incorporated that is not itself part of the Compliant Implementation; or (b) to any Adopter if that Adopter is not making a reciprocal grant to Members, Contributors and Academic Contributors, as set forth in Section 3.3. For the avoidance of doubt, the foregoing licensing commitment includes the distribution by the Adopter's distributors and the use by the Adopter's customers of such licensed Compliant Implementations.

3.2 Notwithstanding the above, if any Member, Contributor or Academic Contributor, Adopter or their Affiliates has reserved the right to charge a FRAND royalty or other fee for its license of Necessary Claims to Adopter, then Adopter is entitled to charge a FRAND royalty or other fee to such Member, Contributor or Academic Contributor, Adopter and its Affiliates for its license of Necessary Claims to its licensees.

3.3 Adopter, on behalf of itself and its Affiliates, shall be prepared to grant based on a separate Patent License Agreement to each Members, Contributors, Academic Contributors, Adopters and their Affiliates under Fair Reasonable And Non-Discriminatory (FRAND) terms and conditions with or without compensation (royalties) a nonexclusive, non-transferable, irrevocable (but subject to Defensive Suspension), non-sublicensable, worldwide patent license under their Necessary Claims to make, have made, use, import, offer to sell, lease, sell and otherwise distribute Compliant Implementations; provided, however, that such license will not extend: (a) to any part or function of a product in which a Compliant Implementation is incorporated that is not itself part of the Compliant Implementation; or (b) to any Members, Contributors, Academic Contributors, Adopters and their Affiliates that is not making a reciprocal grant to Adopter, as set forth in Section 3.1. For the avoidance of doubt, the foregoing licensing commitment includes the distribution by the Members', Contributors', Academic Contributors', Adopters' and their Affiliates' distributors and the use by the Members', Contributors', Academic Contributors', Adopters' and their Affiliates' customers of such licensed Compliant Implementations.

Section 4: TERM AND TERMINATION

4.1 This Agreement shall remain in force, unless early terminated according to this Section 4.

4.2 O-RAN ALLIANCE on behalf of its Members, Contributors and Academic Contributors may terminate this Agreement if Adopter materially breaches this Agreement and does not cure or is not capable of curing such breach within thirty (30) days after being given notice specifying the breach.

4.3 Sections 1, 3, 5 - 11 of this Agreement shall survive any termination of this Agreement. Under surviving Section 3, after termination of this Agreement, Adopter will continue to grant licenses (a) to entities who become Adopters after the date of termination; and (b) for future versions of ORAN Specifications that are backwards compatible with the version that was current as of the date of termination.

Section 5: CONFIDENTIALITY

Adopter will use the same care and discretion to avoid disclosure, publication, and dissemination of O-RAN Specifications to third parties, as Adopter employs with its own confidential information, but no less than reasonable care. Any disclosure by Adopter to its Affiliates, contractors and consultants should be subject to an obligation of confidentiality at least as restrictive as those contained in this Section. The foregoing obligation shall not apply to any information which is: (1) rightfully known by Adopter without any limitation on use or disclosure prior to disclosure; (2) publicly available through no fault of Adopter; (3) rightfully received without a duty of confidentiality; (4) disclosed by O-RAN ALLIANCE or a Member, Contributor or Academic Contributor to a third party without a duty of confidentiality on such third party; (5) independently developed by Adopter; (6) disclosed pursuant to the order of a court or other authorized governmental body, or as required by law, provided that Adopter provides reasonable prior written notice to O-RAN ALLIANCE, and cooperates with O-RAN ALLIANCE and/or the applicable Member, Contributor or Academic Contributor to have the opportunity to oppose any such order; or (7) disclosed by Adopter with O-RAN ALLIANCE's prior written approval.

Section 6: INDEMNIFICATION

Adopter shall indemnify, defend, and hold harmless the O-RAN ALLIANCE, its Members, Contributors or Academic Contributors, and their employees, and agents and their respective successors, heirs and assigns (the "Indemnitees"), against any liability, damage, loss, or expense (including reasonable attorneys' fees and expenses) incurred by or imposed upon any of the Indemnitees in connection with any claims, suits, investigations, actions, demands or judgments arising out of Adopter's use of the licensed O-RAN Specifications or Adopter's commercialization of products that comply with O-RAN Specifications.

Section 7: LIMITATIONS ON LIABILITY; NO WARRANTY

EXCEPT FOR BREACH OF CONFIDENTIALITY, ADOPTER'S BREACH OF SECTION 3, AND ADOPTER'S INDEMNIFICATION OBLIGATIONS, IN NO EVENT SHALL ANY PARTY BE LIABLE TO ANY OTHER PARTY OR THIRD PARTY FOR ANY INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES RESULTING FROM ITS PERFORMANCE OR NON-PERFORMANCE UNDER THIS AGREEMENT, IN EACH CASE WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, AND WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. O-RAN SPECIFICATIONS ARE PROVIDED "AS IS" WITH NO WARRANTIES OR CONDITIONS WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE. THE O-RAN ALLIANCE AND THE MEMBERS, CONTRIBUTORS OR ACADEMIC CONTRIBUTORS EXPRESSLY DISCLAIM ANY WARRANTY OR CONDITION OF MERCHANTABILITY, SECURITY, SATISFACTORY QUALITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, ERROR-FREE OPERATION, OR ANY WARRANTY OR CONDITION FOR O-RAN SPECIFICATIONS.

Section 8: ASSIGNMENT

Adopter may not assign the Agreement or any of its rights or obligations under this Agreement or make any grants or other sublicenses to this Agreement, except as expressly authorized hereunder, without having first received the prior, written consent of the O-RAN ALLIANCE, which consent may be withheld in O-RAN ALLIANCE's sole discretion. O-RAN ALLIANCE may freely assign this Agreement.

Section 9: THIRD-PARTY BENEFICIARY RIGHTS

Adopter acknowledges and agrees that Members, Contributors and Academic Contributors (including future Members, Contributors and Academic Contributors) are entitled to rights as a third-party beneficiary under this Agreement, including as licensees under Section 3.

Section 10: BINDING ON AFFILIATES

Execution of this Agreement by Adopter in its capacity as a legal entity or association constitutes that legal entity's or association's agreement that its Affiliates are likewise bound to the obligations that are applicable to Adopter hereunder and are also entitled to the benefits of the rights of Adopter hereunder.

Section 11: GENERAL

This Agreement is governed by the laws of Germany without regard to its conflict or choice of law provisions.

This Agreement constitutes the entire agreement between the parties as to its express subject matter and expressly supersedes and replaces any prior or contemporaneous agreements between the parties, whether written or oral, relating to the subject matter of this Agreement.

Adopter, on behalf of itself and its Affiliates, agrees to comply at all times with all applicable laws, rules and regulations with respect to its and its Affiliates' performance under this Agreement, including without limitation, export control and antitrust laws. Without limiting the generality of the foregoing, Adopter acknowledges that this Agreement prohibits any communication that would violate the antitrust laws.

By execution hereof, no form of any partnership, joint venture or other special relationship is created between Adopter, or O-RAN ALLIANCE or its Members, Contributors or Academic Contributors. Except as expressly set forth in this Agreement, no party is authorized to make any commitment on behalf of Adopter, or O-RAN ALLIANCE or its Members, Contributors or Academic Contributors.

In the event that any provision of this Agreement conflicts with governing law or if any provision is held to be null, void or otherwise ineffective or invalid by a court of competent jurisdiction, (i) such provisions will be deemed stricken from the contract, and (ii) the remaining terms, provisions, covenants and restrictions of this Agreement will remain in full force and effect.

Any failure by a party or third party beneficiary to insist upon or enforce performance by another party of any of the provisions of this Agreement or to exercise any rights or remedies under this Agreement or otherwise by law shall not be construed as a waiver or relinquishment to any extent of the other parties' or third party beneficiary's right to assert or rely upon any such provision, right or remedy in that or any other instance; rather the same shall be and remain in full force and effect.