European Commission
Information Society and Media

SEVENTH FRAMEWORK PROGRAMME

Specific Targeted Research Projects

SOLDER

Spectrum OverLay through aggregation of heterogeneous DispERsed Bands

FP7 Contract Number: 619687

SOLDER

WP4 – SOLDER system development and integration

D4.1
PoC Scenarios and system interfaces

Contractual Date of Delivery:	30 April 2015
Actual Date of Delivery:	
Responsible Beneficiary:	IS-WIRELESS
Contributing Beneficiaries:	ISI/ATHENA RC, EURECOM, IS-WIRELESS, KCL, SEQ, TCS
Security:	Public
Nature	Report
Version:	1.0

Document Information

Document ID: D4.1.docx
Version Date: 30/04/2015
Total Number of Pages: 45

Abstract

This document will describe the envisioned PoC scenarios and their necessary interfaces. This document serves as an input for the following tasks in work-package 4.

Keywords

Authors

Name	Organisation	Email
Florian Kaltenberger	EURECOM	Florian.kaltenberger@eurecom.fr
Charalampos Antoniadis	Athena_RC/ISI	chantoniad@Isi.gr
George Floros	Athena_RC/ISI	gfloros@isi.gr
Manos Antonopoulos	Athena_RC/ISI	mantonop@isi.gr
Fotis Foukalas	Athena_RC/ISI	foukalas@isi.gr
Jedrzej Stanczak	IS-Wireless	j.stanczak@is-wireless.com
Mateusz Buczkowski	IS-Wireless	m.buczkowski@is-wireless.com
Maciej Wewior	IS-Wireless	m.wewior@is-wireless.com
Oliver Holland	KCL	oliver.holland@kcl.ac.uk
Adnan Aijaz	KCL	adnan.aijaz@kcl.ac.uk
Guillaume Vivier	Sequans	Guillaume.vivier@sequans.com
Sylvain Traverso	Thales	sylvain.traverso@thalesgroup.com

Document History

Revision	Date	Modification	Authors
0.1	18.11.2014	Initial TOC, section editor assignments and editing notes	Florian Kalten-berger
See BSCW file log for detailed revision information			
1.0	30.4.2015	Final version sent to the commission	Florian Kalten-berger

Executive Summary

SOLDER work-package 4 (WP4) is dedicated to the demonstration of some of the key mechanisms developed in work-package 3 (WP3) in real systems based on the scenarios developed in work-package 2. The first task of WP4 was the down-selection of the scenarios from the initial list of 8 scenarios that were identified in D2.3 [7], the identification of the features and algorithms to be demonstrated, and the identification of the platforms that these demonstrations should be carried out on. The result of this process is a list of so called proof-of-concepts (PoCs), where each PoC is based on one particular hardware platform and demonstrates one (or several) functionalities.

In summary, five different PoCs will be developed in WP4. These 5 PoC will cover three different scenarios and will be based on four different platforms. For each of the PoCs we recall the scenario being covered and the platform that is going to be used. We further describe the overall system architecture identifying the building blocks and their interfaces between them. Special emphasis is given to the interfaces between building blocks developed by different partners, as those will be used as a starting point for the subsequent building blocks development and integration. Last but not least we give an outlook of the trials that will be carried out towards the end of the project.

List of Figures

List of Tables

Table of Contents

1. Introduction

SOLDER work-package 4 (WP4) is dedicated to the demonstration of some of the key mechanisms developed in work-package 3 (WP3) in real systems based on the scenarios developed in work-package 2. The first task of WP4 was the down-selection of the scenarios from the initial list of 8 scenarios that were identified in D2.3 [7], the identification of the features and algorithms to be demonstrated, and the identification of the platforms that these demonstrations should be carried out on. The result of this process is a list of so called proof-of-concepts (PoCs), where each PoC is based on one particular hardware platform and demonstrates one (or several) functionalities.

In summary, five different PoCs will be developed in WP4. Table 1 below shows a list of these PoCs along with the considered scenarios they will be used in and the employed platforms for implementation and demonstration. It can be seen that some overlap between the PoCs exist both in terms of scenarios and platforms; however, each of them focus on different aspects of the overall SOLDER architecture as defined in D2.3.

The following sections will describe each of them in more detail, specifying the used hardware platforms, their hardware and software architecture and all the relevant interfaces. Special emphasis is given to the interfaces between building blocks developed by different partners, as those will be used as a starting point for the subsequent building blocks development and integration.

Table 1: List of SOLDER proof-of-concepts

PoC #	PoC name	Scenario	Platform
PoC 1a	Aggregation of TVWS and WiFi for augmented broadcast	Aggregation of heterogeneous spectrum types	KCL-Eurecom TVWS devices
PoC 1b	Aggregation of LTE in licensed and unlicensed bands		ISI Cognitive Radio Interface
PoC 2	Energy efficient transmission technologies	5G Waveforms	Eurecom ExpressMIMO2 platform
PoC 3a	LTE-A inter-band carrier aggregation in homogeneous networks	LTE Carrier Aggregation	Eurecom OpenAirInterface eNB with ISW scheduling algorithm + Sequans UE
PoC 3b	Dynamic Cognitive CA in HetNets: LTE-A and beyond		ISI Cognitive Radio Interface

2. PoC 1: Aggregation of heterogeneous spectrum types

2.1 PoC 1a: Aggregation of TVWS and WiFi for augmented broadcast

2.1.1 Scenario(s)

Here we consider our work on the aggregation of conventional unlicensed spectrum with TV white space spectrum opportunities. Within this context, a scenario is developed for aggregation of LTE MBMS broadcast in TV white space with WiFi transmitting in “conventional” unlicensed spectrum, noting anyway that TV white space is also another form of unlicensed spectrum that is specified on a localised basis through the geolocation database. This scenario matches with the “Augmented Broadcast” scenario presented in Annex 2 of D2.1 [7].

A key purpose of this is to support large-scale broadcast data augmented through aggregation with localized available sources, such as layered video, large-scale software downloads (e.g., OS or application upgrades, that require the same information set to be sent to a very large number of devices at approximately the same time), and perhaps broadcast signaling traffic such as through the Cognitive Pilot Channel (CPC) concept [8]. Figure 1 depicts the scenario of a large-scale broadcast being aggregated with locally-available WiFi to achieve these ends.

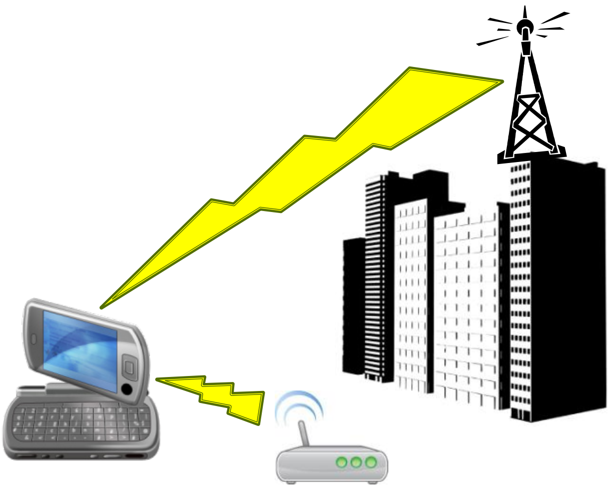


Figure 1: Depiction of aggregation in the "Augmented Broadcast" scenario

2.1.2 Platform(s)

A prototype has been developed for TV white space transmission, based on the UK’s Ofcom TV white spaces framework (taken to the European level in the form of an ETSI harmonized standard [1]). Here, the logical aspects (all aspects of the communication with the Ofcom web listing, the hosen geolocation database, derivation and setting of parameters based on the response etc.), have been implemented in accordance with the framework. RF aspects have been developed in view of achieving spectrum mask characteristics under the framework, and the radio waveform has been developed as a minor adaptation of OpenAirInterface [2]. The hardware that is being used is the PCI-Express Eurecom ExpressMIMO2 software radio, of which KCL has three units, in addition to KCL custom-made PCs built deliberately with the intention of being highly powerful for software radio waveform generation, but still energy efficient for the purpose of easy deployment in the field and in mobile scenarios. For the latter case, the entire set-up is developed such that it is even energy efficient enough to be run from a conventional car cigarette lighter, which is typically (after transformer inefficiency is considered) capable of around 140-150 Watts output at 240V AC.

Figure 2 depicts the basestation-side hardware that has been developed and is being utilized for the Augmented Broadcast scenario.

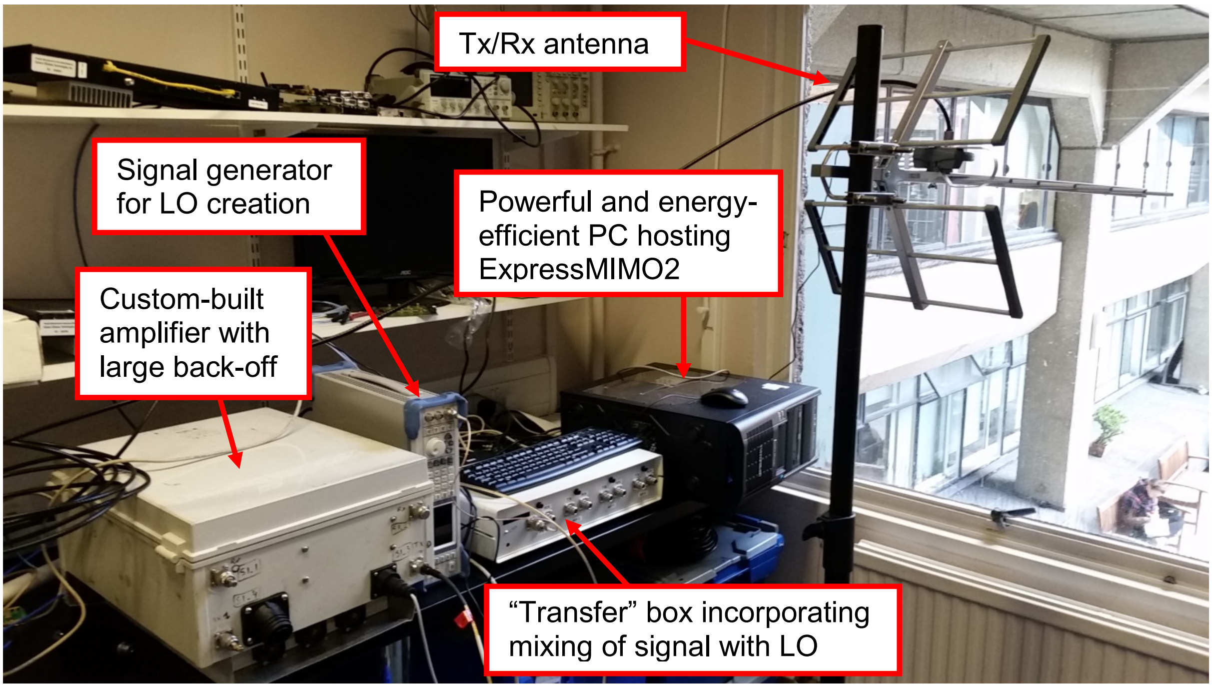


Figure 2: Depiction of the base-station side for the "Augmented Broadcast" scenario

2.1.3 Architecture

Regarding the "Augmented Broadcast" scenario, the core waveform is generated in software on a host PC, fitted with the ExpressMIMO2 software radio PCI Express card. This card is depicted in Figure 3. This card essential provides FPGA functionality to 4 RF chains, all based on the Lime Micro PMS6002D chip. Hence, the card itself is capable of interesting non-contiguous aggregation scenarios. The FPGA (the Spartan 6 LX150T) is essentially concerned with interfacing to the host PC, although there is some limited spare capacity that might be used for other simple RF/PHY and MAC purposes.

The initial intention was merely to transmit directly in TV spectrum using this card, along with custom-built amplification with a very large back-off in order to maintain linearity. This was in an effort to meet the most challenging Class 1 requirements under Ofcom's framework, which has been taken to the EU level in the form of a harmonized standard [1], noting that the ExpressMIMO2 itself has an extremely low power output, of typically around -10 dBm. It was realized that the LMS6002D chips produce spurious emissions towards the lower end of the TV band, hence, in was decided to create the signal at high frequency (namely, GPS—1575MHz), filter there (using a GPS filter), and then down-convert to the desired frequency using an appropriately-configured LO and filtering out the higher-frequency byproduct. The initial intention was to use a LO generated by one of the other ExpressMIMO2 RF chains for down-conversion, however, it was found that this didn't have pure enough properties. A high-quality signal generator (Rohde and Schwarz SMB100A) was therefore drafted in to provide the LO, the frequency and power of this LO being controlled over Ethernet from the host PC.

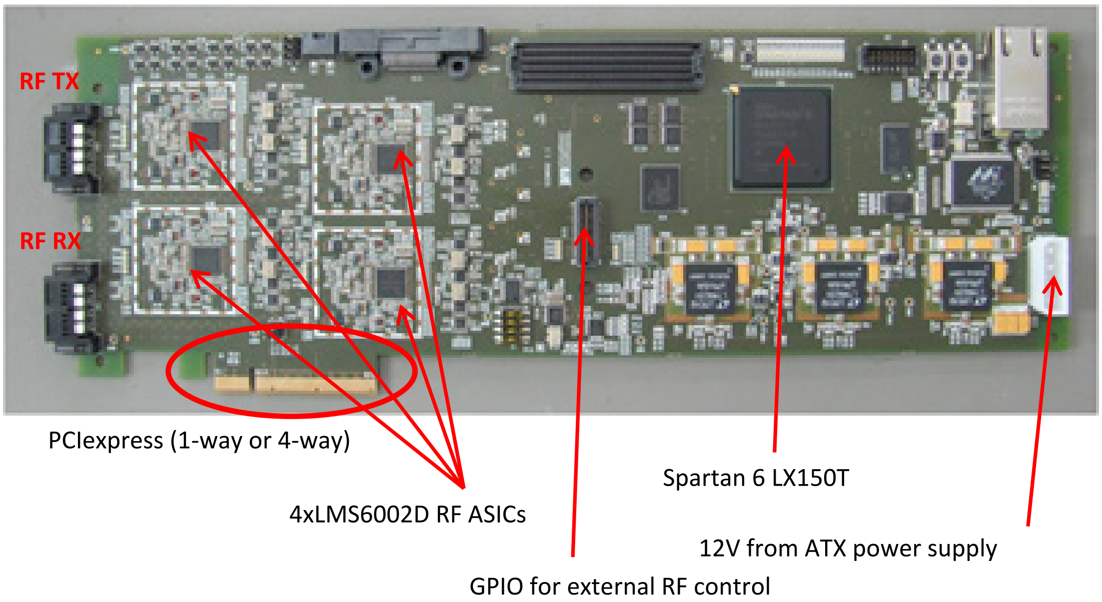


Figure 3: Eurecom ExpressMIMO2

A controller application was programmed in Python on the host PC, which dealt with all logical aspects of the Ofcom framework, i.e., communication with the Ofcom weblisting of geolocation databases, communication with the chosen geolocation database, derivation and setting of parameters, initiation and stopping of the radio waveform generated by OpenAirInterface, and control of the LO (SMB100A) via Ethernet signalling, among other aspects. One example of the functionality of this controller application is present in the GUI-based (Python/Tk) variant of it that polls geolocation databases for the purpose of making aggregation decisions, as has been mentioned in Section 3.4.3 of D2.3.

As part of our trial within the Ofcom TV White Spaces Ofcom engineers have visited King's College London on two occasions in order to perform compliance testing on the equipment that we are utilizing. Compliance testing on the Eurecom/KCL set-up was done specifically with the purpose of verifying the logical aspects. Ofcom have confirmed that the logical aspects were operating as they should do under the framework.

A high-level presentation of our implementation is given in Figure 4. Note, this is not considering the WiFi aspects such as the presence of a WiFi radio on the host PC.

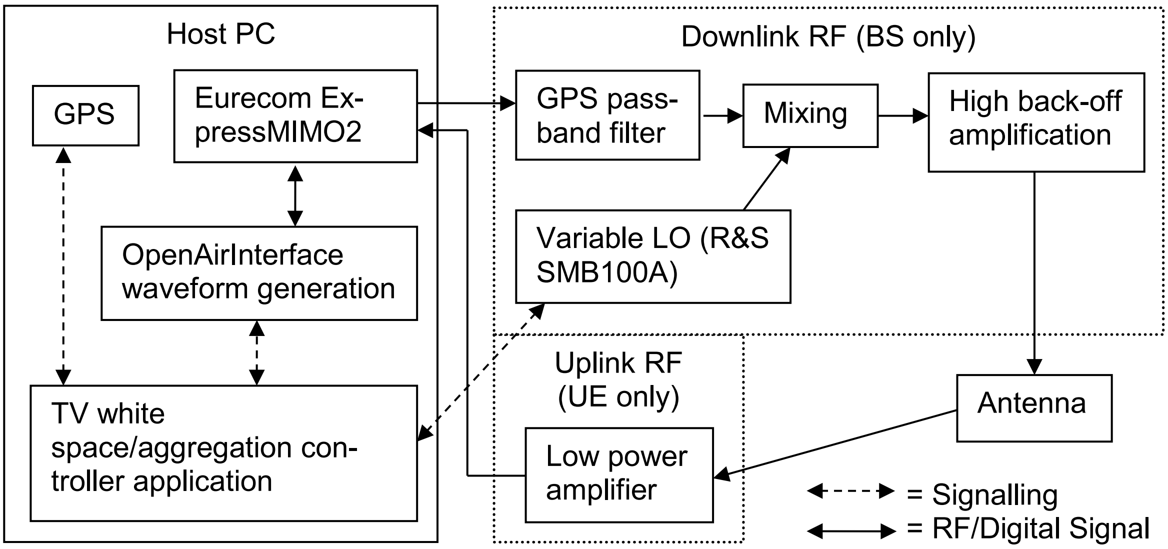


Figure 4: Our aggregation-capable white space device implementation based around the Eurecom ExpressMIMO2 hardware. Note, directions of arrows in the downlink/uplink RF parts (right side) of this depiction represent the base station implementation

2.1.4 Interfaces definition

The key entities involved in this scenario, based on the architecture presented in Section 2 of [10], are the Network Aggregation Manager (NAM), Terminal Aggregation Manager (TAM), Network Aggregation Controller (NAC), Terminal Aggregation Controller (TAC), Network Aggregation Information Collector (NAIC), Terminal Aggregation Information Collector (TAIC), potentially the Centralized Spectrum Aggregation Controller (CSAC), and of course Radio Access Networks (RANs) or radio access more generally.

The associated interfaces are as follows:

- *CSAC-NAM*: This interface is necessary in order ensure oversight is maintained for cases where the decisions on link aggregation and associated management in this scenario are distributed between different legal entities (e.g., operators). Only in such cases will the involvement of the CSAC, hence the implementation of this interface, be necessary. Although potentially useful more generally in the scope of LTE (MBMS) and WiFi aggregation, for the purpose of this PoC everything (including the implementation of the LTE waveform and flow, the implementation of the WiFi flow, the management of the link aggregation, etc.) will be handled on the source PC, which also will act as the LTE (MBMS) base station. Hence, there is unlikely to be a need for the CSAC, or the implementation of this interface, in this PoC.
- *NAM-NAC*: This interface will be necessary to transfer and implement aggregation decisions from the management to the implementation point. In the context of this PoC, both the NAM and the NAC will be handled by the same (host) PC, which will also act as the LTE (MBMS) base station. Hence, the implementation will be between processes on the same PC, noting that many of the management functions (the NAM, and part of the NAC) are implemented as a management/controller application in Python, and that the LTE radio interface will be implemented in c via OpenAirInterface and custom adaptations to OpenAirInterface for this purpose. Hence, part of the NAC functionality will also likely be linked to adaptations to the OpenAirInterface code.
- *TAM-TAC*: Very similar observations apply as to the NAM-NAC interface, with the exception that this interface is of course on the terminal side.

- *NAM-NAIC*: For the purpose of this PoC, again both the NAM and NAIC will be implemented on the same PC, which will act as the LTE (MBMS) base station. This interface and the entities involved will be implemented largely in the Python management/controller application. The interface will extract context information from readily-available functions in WiFi drivers, as well as the OpenAirInterface implementation, in order to assist the link aggregation decisions. An example of such a decision is how the packet-level coded packets are distributed between the WiFi and LTE (MBMS) flows.
- *TAM-TAIC*: Very similar observations apply as to the NAM-NAIC interface, with the exception that this interface is of course on the terminal side.
- *Interfaces to RANs*: RAN or radio access functionality will be provided by OpenAirInterface with the associated ExpressMIMO2 card in the case of LTE (MBMS), and in the WiFi case by the WiFi network at KCL over which the WiFi flow will be forwarded. On the terminal side, the WiFi PCI card in the associated receiver PC of course provides the radio access. In both LTE (MBMS) and WiFi cases, interfacing with the RAN or radio access capabilities is done through invoking sockets and tuning various parameters in the Python management/controller application. Some interfacing will also be done with the c code of OpenAirInterface, as well as the invoking of command-line instructions from within the applications in order to serve certain functions (e.g., emergency kill commands).

2.1.5 Role of partners and timeline

KCL has already implemented all logical aspects, such as the communication with the Geolocation Database, the Ofcom “Database-of-Databases”, and compliance with the other aspects of the Ofcom framework (e.g., implementation of the “Kill Switch” for misbehaving devices, feedback information to the databases on channel usage decisions, periodic re-invoking of Database and “Database-of-Databases” communications, instructions to radio elements based on the interactions with the database and responses, etc.). This has been approved by Ofcom in the visit of Ofcom engineers to KCL. Eurecom has already implemented the RF aspects, and considered some necessary updates to the OpenAirInterface, with some input from KCL to the RF. Remaining work is the implementation the higher-layer aspects (e.g., coding of packets, splitting and distribution among the interfaces, combining the packets at the receive side, interfacing with the application at the transmit and receive side, etc.).

A large proportion of this PoC is already implemented by KCL and Eurecom. The remaining application-level implementation will be done likely at KCL, with some input from Eurecom—especially where the interaction with OpenAirInterface is involved. It is intended that this work will be completed and initial demonstration of this PoC be done in 2015. If this is not completed and initial testing done in 2015, then it is very likely that licensing under the Ofcom TVWS Pilot, which this is part of, will no longer be possible. In such a case, this work will be done through experimentation in conductive mode, or through obtaining test and trial licenses for specific TV channels.

2.1.6 Trials

This is one small part of an extensive involvement in the Ofcom TV White Spaces Pilot, as part of a trial led by King’s College London (see [17] – although the pilot is referred to as “ACROPOLIS” and King’s College in this, this is because King’s College initiated the trial in its Coordinatorship of the ICT-ACROPOLIS project [18] which completed at the end of 2013). Referring to Figure 5, this involves the deployment of the LTE evolved Multimedia Broadcast/Multicast Service (eMBMS) base station high up on a building, and taking a mobile device to different locations where it will opportunistically aggregate available WiFi connections with the LTE MBMS broadcast that it is already receiving.

It is noted that some aspects of the aggregation work that SOLDER is doing within this pilot relate to aggregation in TV white space only. Although the prototyping aspects of this so-far relate to only interaction with the Ofcom-approved geolocation databases and making aggregation decisions based on that (see, e.g., Figure 8 of D2.3 [10]), it is possible that further work and trials will be undertaken on prototyping aspects solely within TV white space. To this end, King's College London has recently been extremely fortunate to secure the participating of InterDigital in its trial within the Ofcom Pilot. InterDigital has loaned its aggregation-capable devices to King's College for trialing, which transmit WiFi in TV white space on up to 4 aggregated contiguous or non-contiguous TV channels. King's College persuaded InterDigital to join the Ofcom Pilot based on [19] and the resulting presentation, and has played a significant part in advising on the implementation of the Ofcom framework on the InterDigital devices. It is further noted that King's College has secured the participation of additional partners who have created devices capable of aggregating in TV white space, although the best deployment scenarios for them are being considered and it is too early to name names.

Figure 5 is a depiction of the InterDigital devices and system.

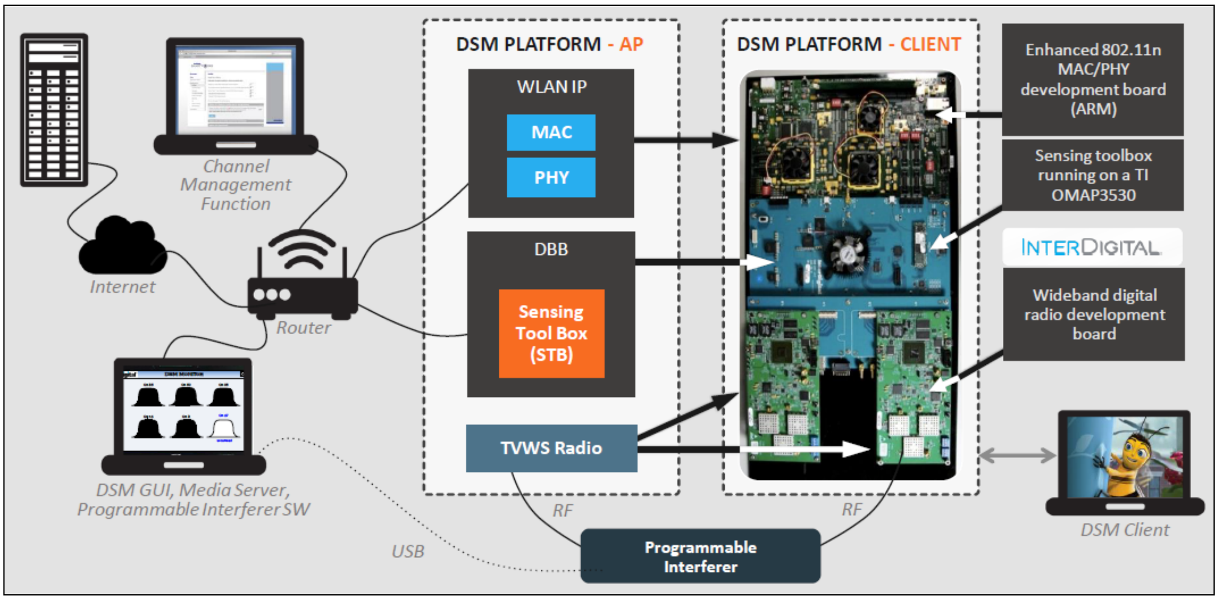


Figure 5: InterDigital WiFi in TV white space devices capable of aggregating up to 4 contiguous or non-contiguous TV channels [20]

2.2 PoC 1b: Aggregation of LTE in licensed and unlicensed bands

2.2.1 Scenario

According to the table of the application scenarios in D2.1, this solution is addressing the LTE in Unlicensed bands scenario and thus, it's included in the PoC 1 that is dealing with the aggregation of heterogeneous spectrum types, i.e. licensed and unlicensed. The demonstration of this scenario relies on the LAA of Rel.13 of LTE-A providing the aggregation of an Unlicensed National Information Infrastructure (U-NII) band to the LTE system efficiently. More details are given below.

2.2.2 Platform

For this scenario the Cognitive Radio Interface (CRI) platform from ISI is used for which details can be found in the Appendix.

2.2.3 Architecture

We omit here the description of the known LTE blocks of Figure 27 and thus, we focus below only on the new ones required built-in within the framework of SOLDER to support the PoC1b, i.e. aggregation of heterogeneous spectrum types. Their functionality is described below in details.

Learning/Cognition block
The Learning/Cognition block is responsible for estimating the unlicensed users activity (typically WiFi users) of the available unlicensed channels of the bands at 5 GHz (UNII). It requires multiple measurements from the power detector for each channel about the existence of UNII signals. It calculates each channel's occupancy time (COT) as the ratio of the number of signal detections to the total amount of detections performed by the power detector. Next, the best Component Carrier i (CC_i)is selected so that COT_i is smaller than the respective COT of every other channel. This learning mechanism is employed periodically so that the best channel in terms of occupancy time is always selected for carrier aggregation.

LBT block
The LBT (Listen Before Talk) block is employed exactly before the beginning of each LTE frame. It requests the current CC_i availability from the power detector and if the result is that the selected CC_i is clear from other signals it enables the utilization of CC_i in the following way. It informs the RF switch to enable the usage of the unlicensed carrier. Next, it utilizes the COT_i value received from the Learning/Cognition block to enable the required Discontinuous Transmission (DTX). This function is used to fill only a portion of the frame with symbols. This portion is decided by the remaining percentage of time that the channel is not utilized by WiFi users i.e. (1-COT_i)*10 subframes.

2.2.4 Interfaces definition

The figure below depicts the procedure of the LTE-U CA (3GPP LAA) implementation according to [6]. The power detector collects multiple measurements about the existence of users on the UNII band for each candidate CC of Band A, B or C at 5 GHz and passes that information to the Learning/Cognition module (UNII CCs Activity). The Learning/Cognition module, upon receiving information about the carriers' activity calculates the best Channel Occupancy Time (COT) among all carriers (COT Estimation). The selected channel is passed to the LBT module (Best CC_i Selection). The LBT module, requests channel availability information from the power detector for the selected channel (CC_i availability). If the channel is available the RF switching module is notified to enable transmission from the unlicensed carrier (CC_i Selection). The COT_i value received from the Learning/Cognition module, is used to enable Discontinuous Transmission (DTX(COT_i)) and thus alerting the Layer Mapper to not fully fill the entire frame with information signals, but only a portion of it for the unlicensed CC (UNII frame structure).

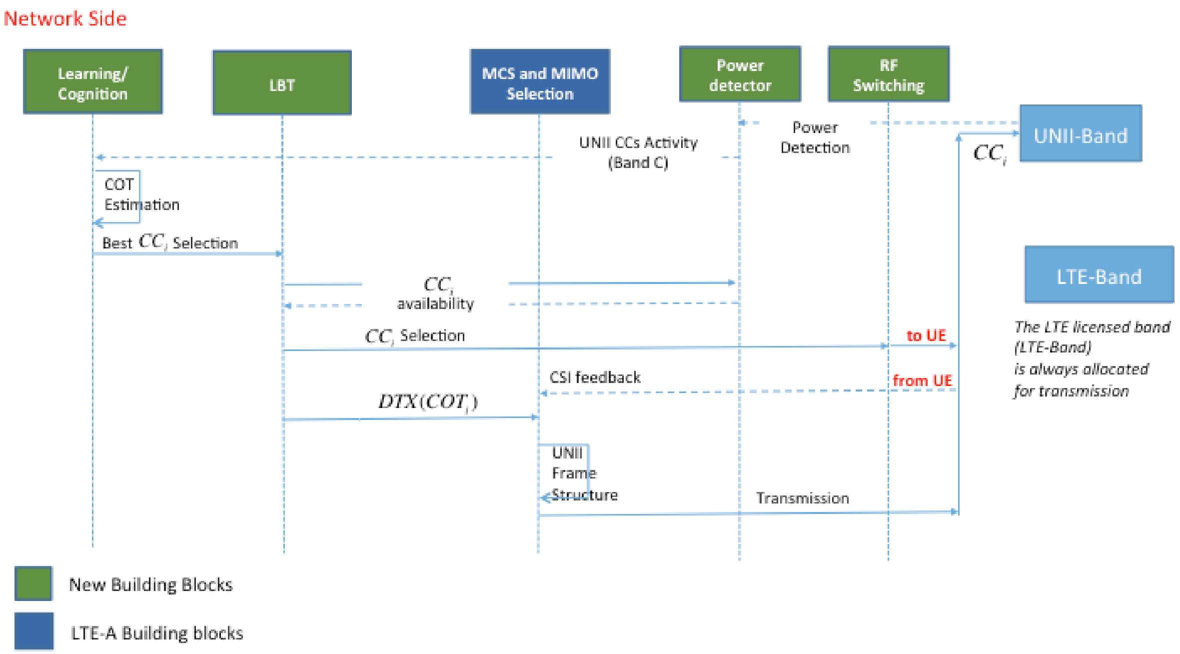


Figure 6: LTE-U CA sequence diagram

The main interfaces involved in the figure above are those provided between the NAC (Network Aggregation Controller) and the NAM (Network Aggregation Manager), which provides SIE (Spectrum Information Extraction) for the carrier selection and spectrum sensing functionalities [10]. In particular, the following messages are exchanged between the involved entities:

- *Power Detection*: The power detector scans unlicensed channels for unlicensed users activity.
- *UNII CCs Activity*: This message contains information collected from the power detector about the unlicensed channels activity.
- *COT Estimation*: COT is calculated for each CC by the Learning/Cognition block.
- *Best CC_i selection*: The message informs the LBT block about the carrier selected and its corresponding COT_i.
- *CC_i Availability*: LBT requests channel availability information from the power detector for the selected channel.
- *CC_i Selection:* This message is sent from LBT to the RF switching after having knowledge about the availability of the particular CC.
- *DTX (COT_i)*: The COT_i value received from the Learning/Cognition module, is used to enable Discontinuous Transmission (DTX) and thus alerting the Layer Mapper to not fully fill the entire frame with information signals.
- *UNII Frame Structure:* Only a portion of the frame contains data for the unlicensed carrier.

Figure 7 depicts the functionality at the UE to complete the transmission of the LTE-U CA. Notably, there is a new interface called SS (Spectrum Selection) in the figure that is responsible to select the CC from the UNII band. We don't provide more details about the feedback calculation, i.e. CQI/PMI/RI reports, since it's known from the LTE system. More research and implementation aspects of this application scenario will be provided in D3.2 and D4.2 respectively.

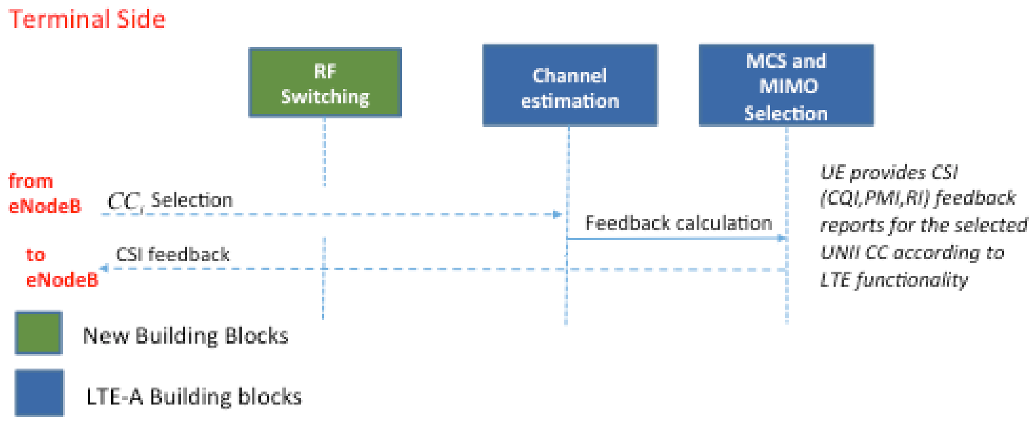


Figure 7: UE side to support the LTE-U CA

2.2.5 Role of partners and timeline

ISI contributes to the PoC1b that will be demonstrated using the LTE tested of the CRI platform. This demonstration will be available in M36.

2.2.6 Trials

The particular scenarios will be demonstrated using the CRI platform including the Tx and the Rx in DL operation. We are going to measure the aggregated throughput under particular unlicensed channel statistics. The efficient design of the learning functionality and the interference knowledge will provide efficient carrier selection from the UNII band at 5 GHz including discontinuous transmission (DTX) for adapting the frame format to WiFi transmission requirements.

3. PoC 2: Energy efficient transmission technologies

3.1 Scenario(s)

The “energy efficient transmission technologies” proof of concept is addressed by the scenario “Toward 5G scenario” described in SOLDER deliverable D2.1 [7].

The aim of this scenario is to show that it is possible to aggregate 2 or more contiguous/non-contiguous intra-band component carriers using a unique PHY layer, and considering single user point-to-point transmission between the UE and the eNodeB in both directions (Uplink and Downlink). The PHY layer is a Filter Bank Multi Carrier (FBMC) scheme which is a candidate for 5G.

In this scenario we emphasize on the transmitter side. We will demonstrate that it is possible to keep the FBMC frequency containment to a reasonable level, while decreasing the transmitter power consumption.
More specifically, we will develop a calibration procedure (Figure 8) followed by a linearization procedure (Figure 9.). The demonstration will be as follows:

1) Launch the self-calibration procedure in order to estimate the radiofrequency impairments
2) The radiofrequency impairments parameters are transmitted to the “RF impairments compensation” blocks
3) A FBMC signal which has its Peak to Average Power Ratio (PAPR) reduced to a predefined level is generated (see “FBMC+PAPR reduction” block in Figure 9).
4) The FBMC signal is transmitted through the Express MIMO2 and the PA, and is digitally converted. The Digital PreDistortion (DPD) is estimated using the transmitted FBMC signal and the digital representation of the PA output signal (see “DPD estimation” block in Figure 9).
5) The same FBMC signal (or another one) is predistorted (see “DPD application” block in Figure 9) and transmitted once again through the Express MIMO2 and the PA.

3.2 Platform(s)

The “energy efficient transmission technologies” PoC will be assessed using the Express-MIMO2 platform provided by Eurecom.

The calibration procedure presented in Figure 8 is required to quickly and efficiently self-calibrate the platform in order to improve the transmitter linearization process. It basically consists in transmitting and acquiring a calibration signal through the transmitter RF part and the feedback loop. The generation and the transmitter/feedback loop radiofrequency impairments estimations are performed off-line thanks to Matlab algorithms developed in WP3.

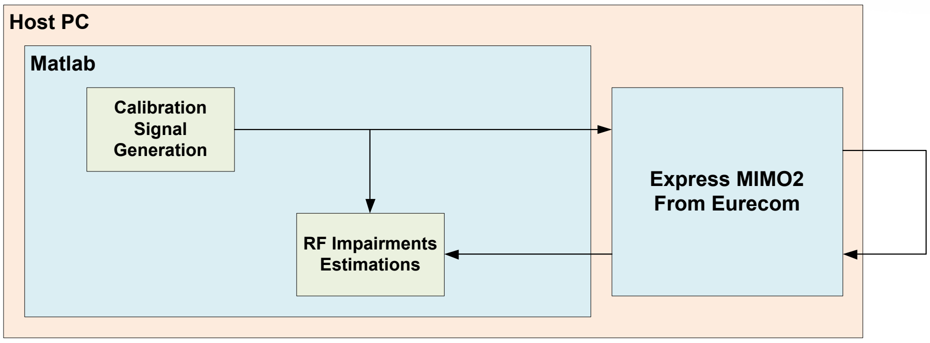


Figure 8: Calibration procedure

The linearization procedure is presented in Figure 9. A power amplifier (PA) composed of a booster and potentially a driver is connected at the ExpressMIMO2 output. The attenuated signal at the PA output is digitally converted thanks to a feedback loop, while a spectrum analyser monitors the transmitted spectrum. The selected algorithms developed in WP3 run on a PC in Matlab.

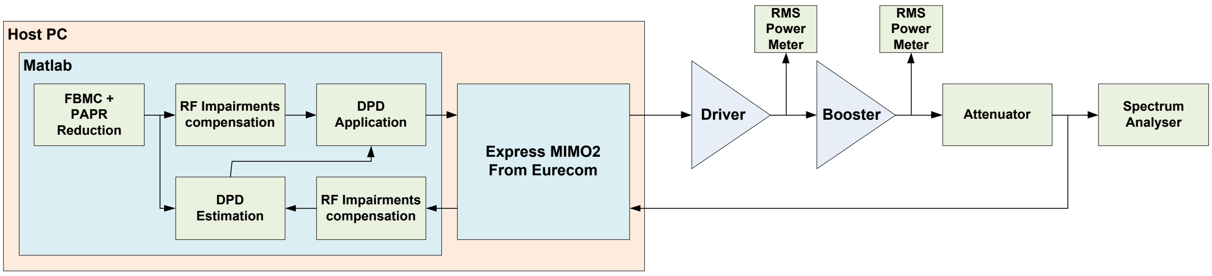


Figure 9: Linearization procedure

3.3 Architecture

The demonstration is composed of 3 building blocks as shown in Figure 10:

- Baseband processing: The selected algorithms coming from WP3 studies will be integrated into this building block. Basically, it consists in a PC equipped with Matlab.
- Tx/Rx Chain: The Express MIMO provided by Eurecom is in charge of up converting the samples coming from the baseband processing to the RF Front-End, and down converting the analog signal coming from the RF Front-End to baseband processing.
- RF Front-End: This building block refers to Device Under Test (DUT) which is a PA to be linearized. In order to cover the Uplink and Downlink transmissions, two different commercial off-the-shelf PAs could be used

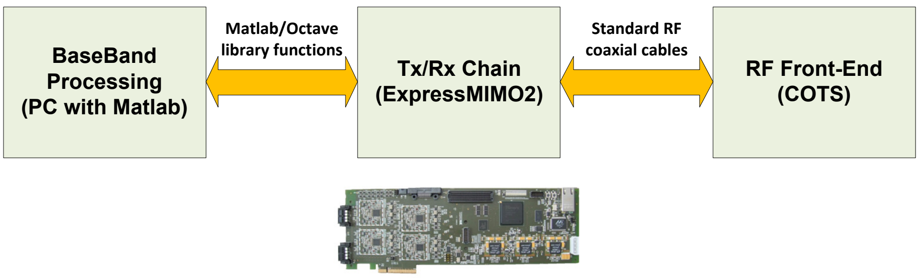


Figure 10: PoC2 Building blocks

3.4 Interfaces definition

The two interfaces related to this PoC are

1. between the baseband processing and ExpressMIMO2: it is basically Matlab/Octave library functions provided by OpenAirinterface.
2. between ExpressMIMO2 and the RF Front-End: it is standard coaxial cables.

3.5 Role of partners and timeline

The Gantt chart for TCS efforts associated with PoC2 is given in Figure 11.

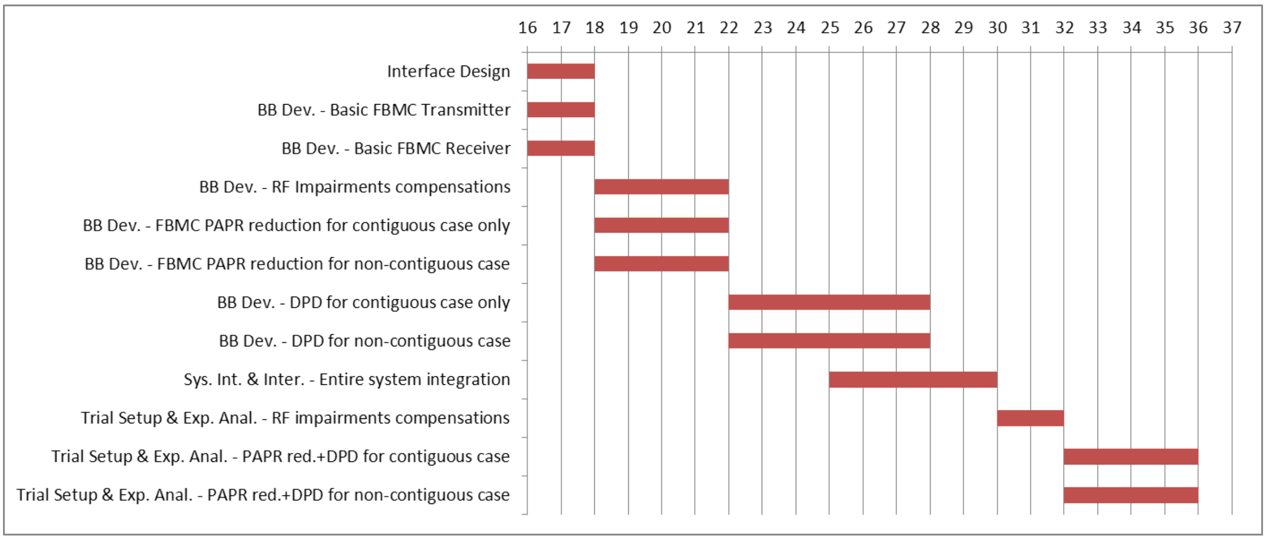


Figure 11: Gantt chart with TCS efforts associated with PoC 2

3.6 Trials

The improvements of the proposed 5G waveform scenario shall be twofold (with regard to 4G):

- Power Amplifier efficiency: we expect a transmitter energy efficiency improvement of 5% thanks to the association of PA linearization and PAPR reduction.
- Out-Of-Band radiation: we expect to improve by a factor 100 (20 dB) the adjacent channel power ratio which is targeted to be lower than 50 dBc. We expect also to provide notch power ratio lower than 50 dBc for the non-contiguous cases.

Note that a positive side effect of the PA linearization is to improve the in-band linearity of the transmitted signal. We will assess this improvement in the demonstration.

4. PoC 3: LTE inter-band carrier aggregation

4.1 PoC 3a: LTE-A inter-band carrier aggregation in homogeneous networks

4.1.1 Scenario(s)

This proof of concept platform aims at demonstrating the “standard” carrier aggregation as defined by the standardisation and captured in the scenarios D2.1 section 3.1.1, D2.2 section 2 and D2.3 section 3.1 [7][9][10].
To some extent, requirements and some of the technical challenges for implementation were already addressed in section 3 of D3.1 [11]

The outcomes of this platform and demonstration lay mostly into:

- Complement the OpenAirInterface OAI framework by adding CA support on the DL (and deliver it to the open community)
- Develop and validate a flexible UE to support DL inter-band CA on various bands that could serve as a vehicle for interoperability sessions, for demonstration or design-in activities with customers.
- Propose and implement open interface for plug-and-play RRM algorithm, leaving the room for operators to customize the behaviour of their radio network equipment.
- Assess in real life RRM algorithms, after evaluation through simulation.

4.1.2 Platform(s)

The proof of concept platform is made of three parts:

- the OpenAirInterface (provided by Eurecom) for the network equipment side,
- a prototype of terminal equipment (provided by Sequans), and
- RRM algorithms (provided by ISW) as further described in section 4.4.

4.1.2.1 OpenAirInterface eNB

The OpenAirInterface software modem comprises a highly optimised C implementation of all the elements of the 3GPP LTE Rel 8.6 protocol stack for both user equipment (UE) and enhanced node B (eNB). The software modem can be run in simulation/emulation mode or in real-time mode together with a hardware target. EURECOM has developed its own hardware target, called ExpressMIMO2, which supports up to four antennas or carrier and a bandwidth of up to 20MHz. Recently, OAI has also been ported to run on universal software radio peripheral (USRP) B210 platform from Ettus research (National Instruments).

The current software modem can interoperate with commercial LTE Rel8 terminals and can be interconnected with closed-source EPC (enhanced packet core) solutions from third-parties (e.g. OpenEPC from Fraunhofer or the LTEBOX development from Alcatel-Lucent).

The new features to support CA on it are summarized below:

- RRC: adaptation of messaging to (1) move to Release 10; (2) to include specific messages related to CA (configuration / deconfiguration of secondary carriers)
- MAC: activation / deactivation of secondary cell (SCell) ; management of payload to distribute over the multiple CC, support of scheduler API (see section 4.1.4).
- RRM: open the RRM interface to be able to import third party RRM algorithms
  PHY: support of CA mechanisms, especially novel format for control channel

4.1.2.2 Sequans UE

The high level requirements for building the UE prototype were described in D3.1 section 3 [11]. Mainly, it should support CA as per standard with the capability to support many possible frequencies.
As a result, the board design separated the main board with a digital component to support the base band and daughter boards for the RF part. Hence, it should be possible to use the base band with as many as possible RF combination possibility. Moreover, the form factor of the prototype is not as integrated as a final UE which minimize the possible issues of intermodulation between RF bands (or at least such issues could be mitigated by adding external filters if needed).
The mother board has been designed with additional test/probe interfaces in order to provide additional possibility for debugging purpose.
The layout of the mother board is given in Figure 12. The prototype board (with the RF daughter board attached is depicted in Figure 13.

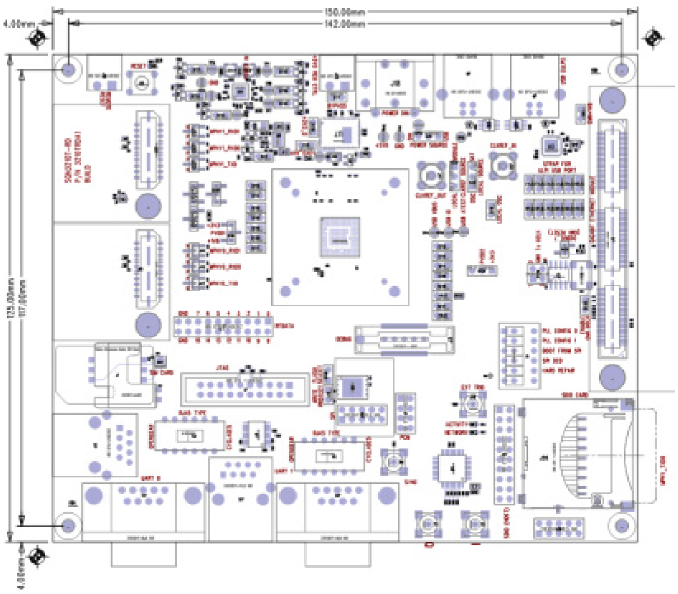


Figure 12: Layout of the mother board

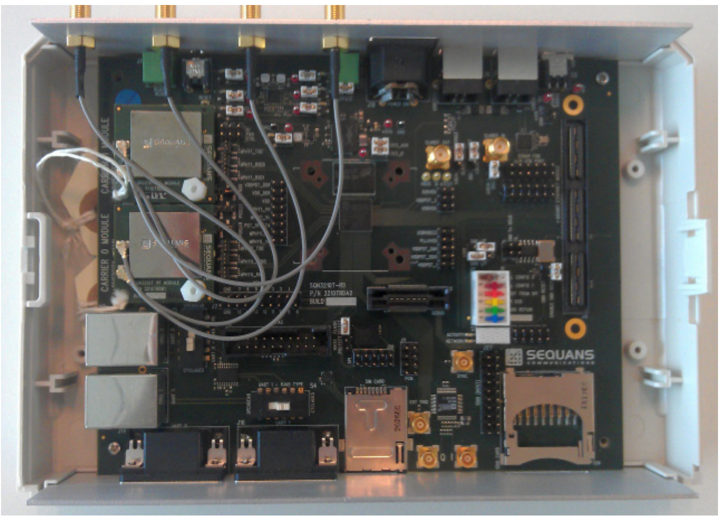


Figure 13: Picture of the mother board (with RF daughter cards attached)

This board has been completely designed by Sequans around a prototype chipset which includes CA supports. Similarly, the RF daughter boards were fully designed by Sequans to fit with the mother board (see Figure 15 and Figure 16). We selected a high speed differential connector as the interface between RF daughter board and the main board (Figure 14). This connector is able to support the transmission of radio signal following DigRF and DigIQ standards.

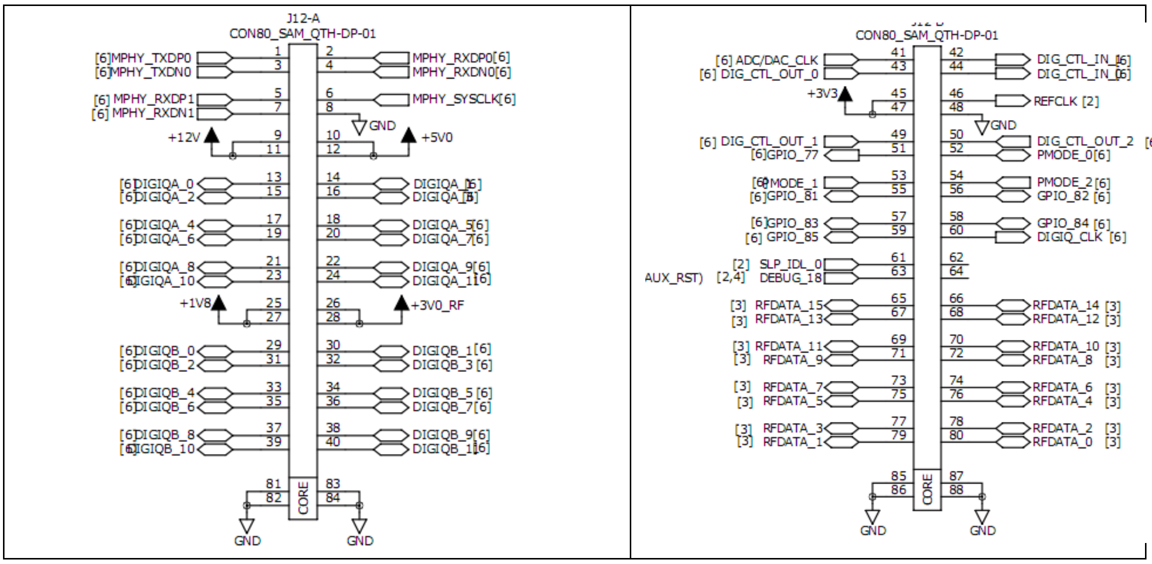


Figure 14: RF interface signal on connector

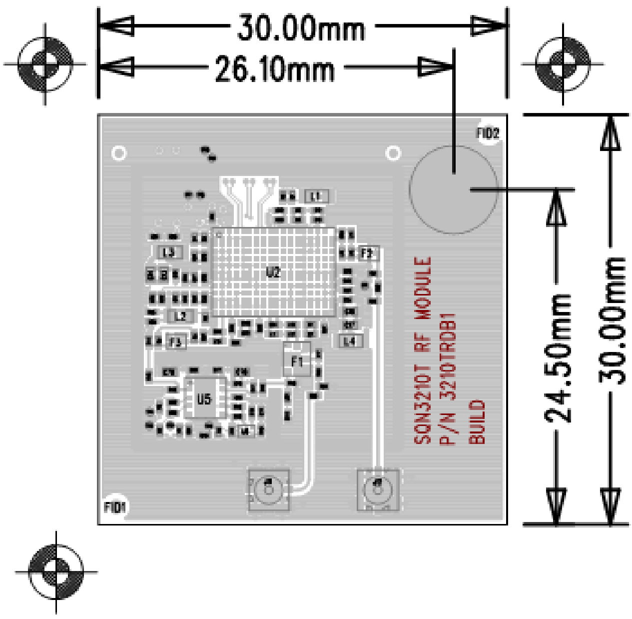
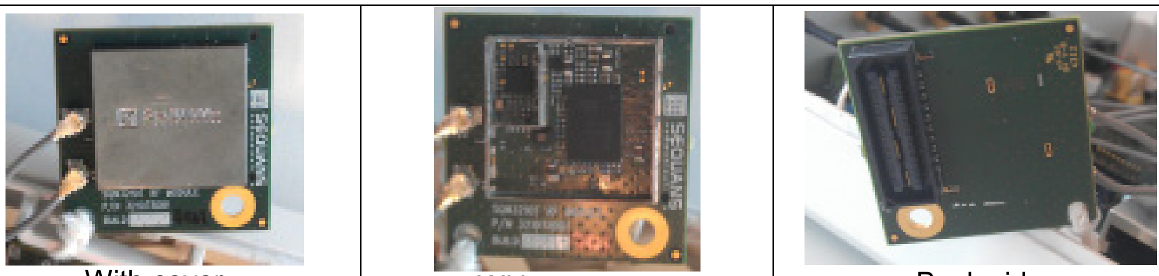


Figure 15: RF daughter board layout



With cover | Without cover | Back side

Figure 16: RF daughter board picture

As per D3.1 requirements [11] two RF board were designed to support respectively band 13 and band 4.

4.1.3 Architecture

The architecture of the PoC follows the standard LTE architecture. A high level figure of this architecture is given in Figure 17. It shows the UE (left) the eNB (middle), with a close-up of the MAC, and the EPC (right) components of an LTE network. Each component is clearly attributed to a partner and the interfaces between the relevant components are highlighted.

Figure 18 shows a more detailed close-up of the MAC of the eNB highlighting the interfaces with the scheduler, which are described in the next section.

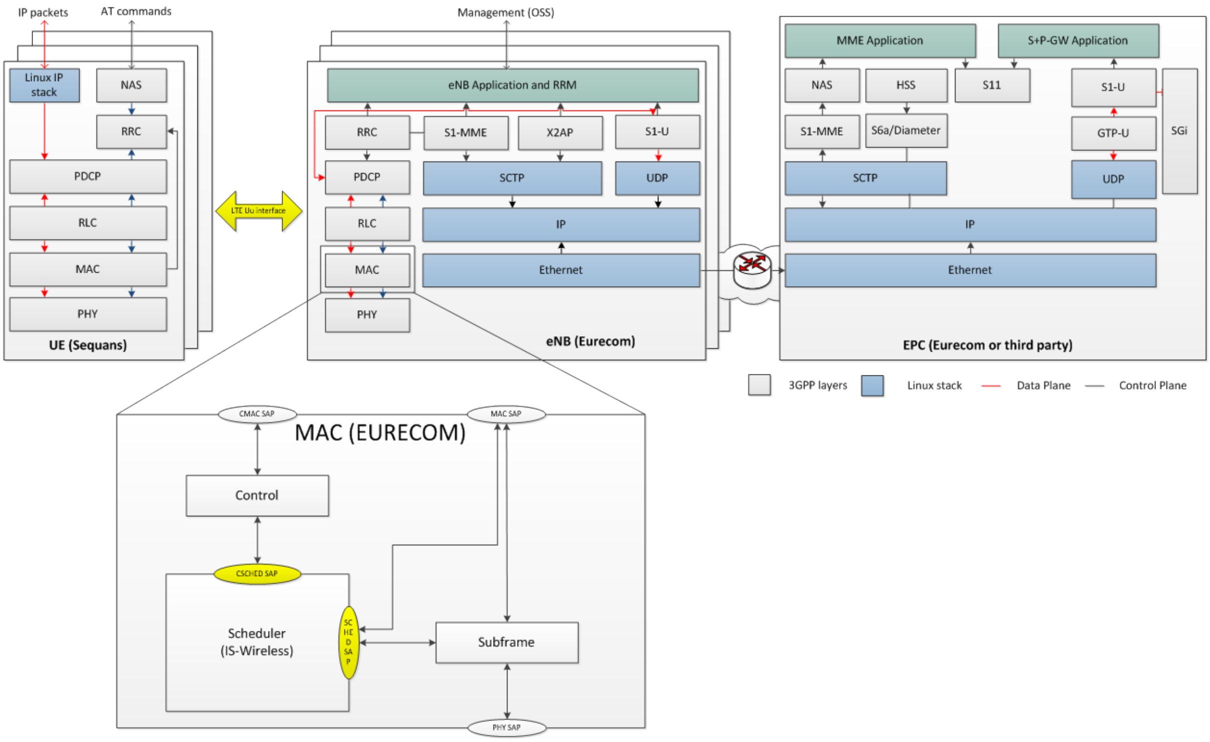


Figure 17: High level functional architecture of an LTE network specifying the responsibility of each partner and the corresponding interfaces.

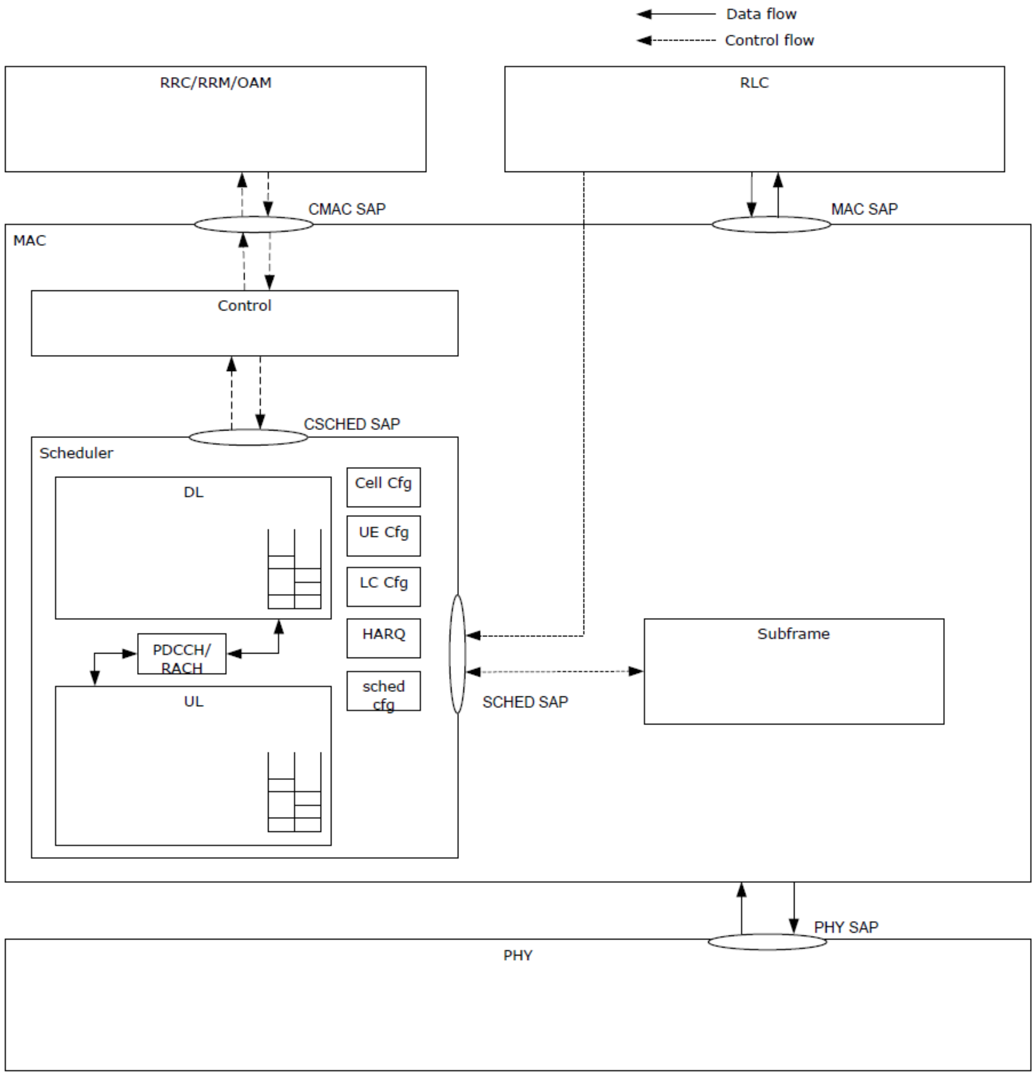


Figure 18: MAC architecture depicting the interfaces with the scheduler [1]

4.1.4 Interfaces definition

The two interfaces related to this PoC are
1. between the eNB and the UE
2. between the scheduler and the MAC in the eNB

The first interface will follow the standard E-UTRAN Uu air interface [5].

The second interface will follow the LTE MAC Scheduler Interface Specification from the Femto Forum [1][3] extended with elements to support carrier aggregation.

As shown in Figure 18, “MAC – scheduler” interface is split into two distinctive parts:
1. CSCHED SAP
2. SCHED SAP

CSCHED SAP interacts with a “Control” entity and is responsible for sending control information to and from MAC sublayer/scheduler. SCHED SAP, on the other hand, is linked with a “Subframe” block. Subframe block triggers scheduler operation (every 1 ms) and receives the outcome of scheduling process. An exhaustive list of primitives exchanged over the two aforementioned interfaces is provided in Section 4 of [3]. The extension of this interface to carrier aggregation is given in the Appendix.

4.1.5 Role of partners and timeline

A Gantt chart for with the different task and their associated efforts is given in Figure 19Figure 19

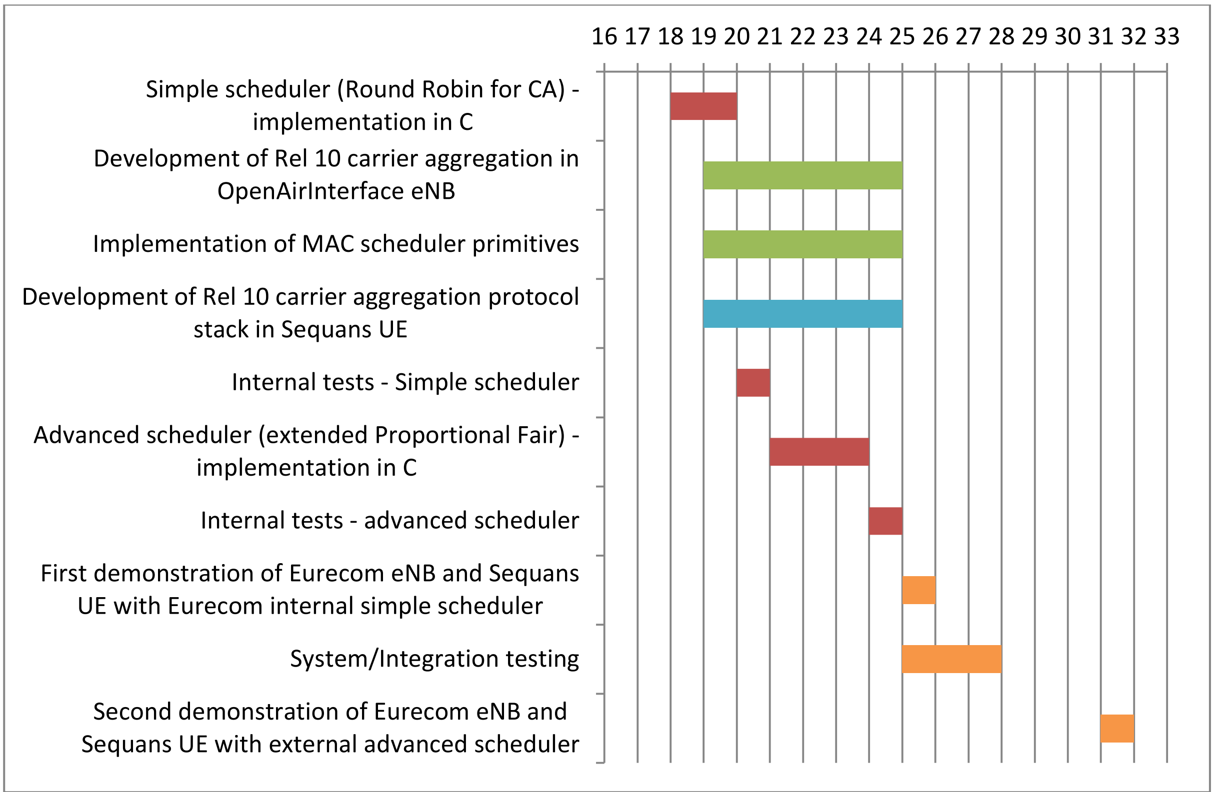


Figure 19: Gantt chart for PoC 3a (Red: ISW, Green: EUR, Blue: SEQ, Orange: all)

4.1.6 Trials

A testbench for the SOLDER project has already been set up at Eurecom and is shown in Figure 20. It consists of a Sequans UE, an application laptop for the UE, the OpenAirInterface eNB, which is a standard PC with the ExpressMIMO2 card inside, and a third party EPC. In this setup, the UE and the eNB are connected by coaxial cables and an attenuator. This setup is useful for development and debugging but it might be also used for the final demonstration, since bands 4 and 13 are in use in France and transmission over the air is not allowed without proper license.

In the trials we will use iperf to measure the throughput of the system (one eNB and one UE) with one and two carriers and we will compare the different scheduling algorithms provided by IS-Wireless.

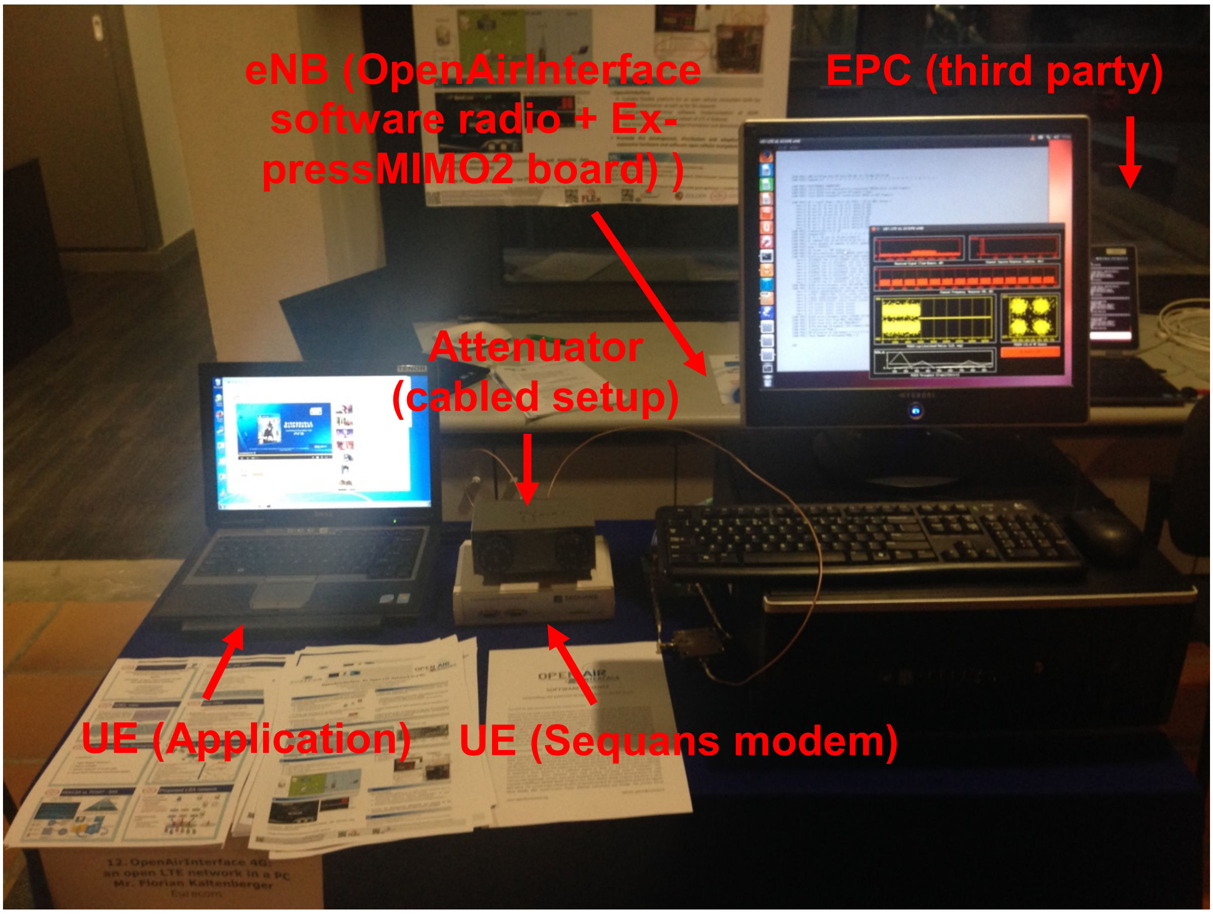


Figure 20: Testbench for inter-band carrier aggregation at Eurecom

4.2 PoC 3b: Dynamic Cognitive CA in HetNets: LTE-A and beyond

4.2.1 Scenario

This application scenario will address the LTE CA according scenario 1 (cf table in Appendix of D2.1). The technology challenge is to aggregate dynamically the best CCs of tri-band CA scenario in order to maximize the aggregated throughput.

4.2.2 Platform

The CRI platform will be used for the demonstration of this scenario. Details can be found in the Appendix A1.

4.2.3 Architecture

The dynamic cognitive CA will be demonstrated using the CRI platform. In this scenario, the following new building blocks are involved in particular.

PMI/RI selection block

In this block a new low-complexity method for selecting the optimal PMI/RI values is proposed. The method exploits the high correlation that adjacent sub-carriers present in typical OFDM systems and calculates the optimal RI value by simply thresholding the eigen-values of the sample correlation matrix. The optimal PMI value is calculated with its turn by minimizing the chordal distance of the sample correlation matrix with the LTE codebook entries, and thus avoiding the computational complexity of existing approaches.

CCs selection block
This new block provides the selection of the best N CCs from the available bands in HetNets deployment. Having the knowledge of the CC statistics from the channel estimation process and having calculated the PMI, RI and CQI of the constituent CCs, the best CCs can be selected in terms of throughout maximization. The criteria for selecting the CCs are the sub-band granularities for both PMI/RI and CQI separately and the bandwidth size. To this end, a new message is required to be sent to the eNodeB and the RF switching block of the UE in order to inform them about the CCs index, i.e. which CCs have been selected. Using such a UE-assisted functionality, it is possible to address the signaling overhead challenge. The low complexity solution of PMI/RI selection provides the efficient implementation of this technique.

4.2.4 Interfaces definition

Figure 21 below depicts a sequence diagram of the PoC3b procedure in order to provide the dynamic cognitive CA in HetNets. The demonstrator will provide 3 bands for which channel estimation can be received at the UE, which uses in order calculate the PMI/RI and CQI values as it is implemented in LTE-A. Notably, the channel estimation will be provided for all CCs of each band in order to provide PMI, RI and CQI calculations for all constituent CCs. The PMI/RI calculation will be accomplished with a new approach in order to reduce the complexity at UE. However, the most important new feature into the LTE-A system is the CCs selection as explained in details in the architecture above.

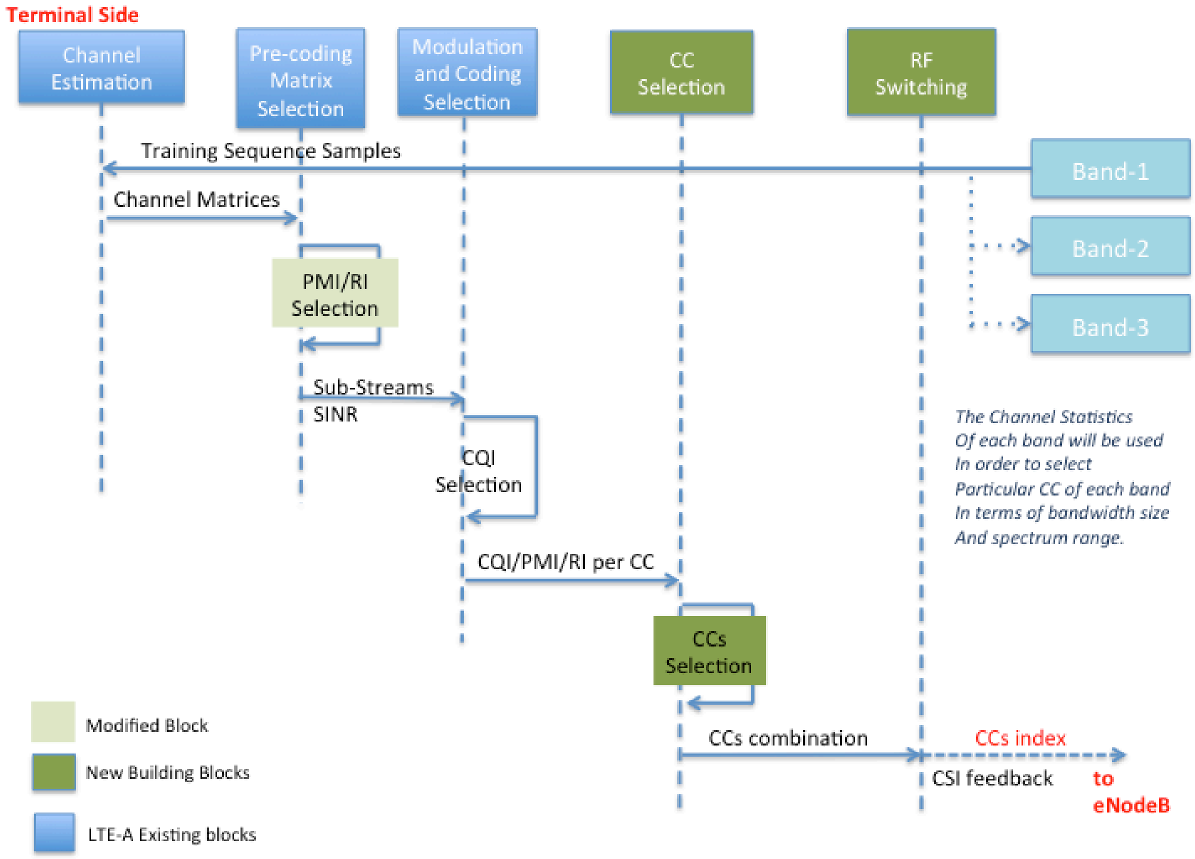


Figure 21: Dynamic Cognitive CA in HetNets sequence diagram

The main interfaces involved in the figure above are those provided between the TAIC (Terminal Aggregation Information Controller) and TAM (Terminal Aggregation Manager), which provides SS (Spectrum Selection) for the aggregation and sensing capabilities respectively [10]. In particular, the following messages are exchanged between the entities:

- *CQI/PMI/RI per CC:* this information is essential providing the information in order to decide which CCs are the best ones to maximize the throughput.
- *CCs combination:* the CC selection block sends the decision about the CC combination to the RF switching of the UE and to the eNodeB a relevant message with the indicators about the selected CCs.

Figure 22 depicts the generic functionality to support the dynamic cognitive CA at the eNodeB side that is identical to the LTE system. However, the rate selection through the MCS and MIMO precoding and layer mapping is carried out for the selected CCs and thus, no more details are depicted in that figure. The RF switching is also notified to transmit over the particular CCs as indicated from the UE. The different research aspects of this scenario will be provided in D3.2 and the implementation aspects in D4.2 according to the time plan of the project.

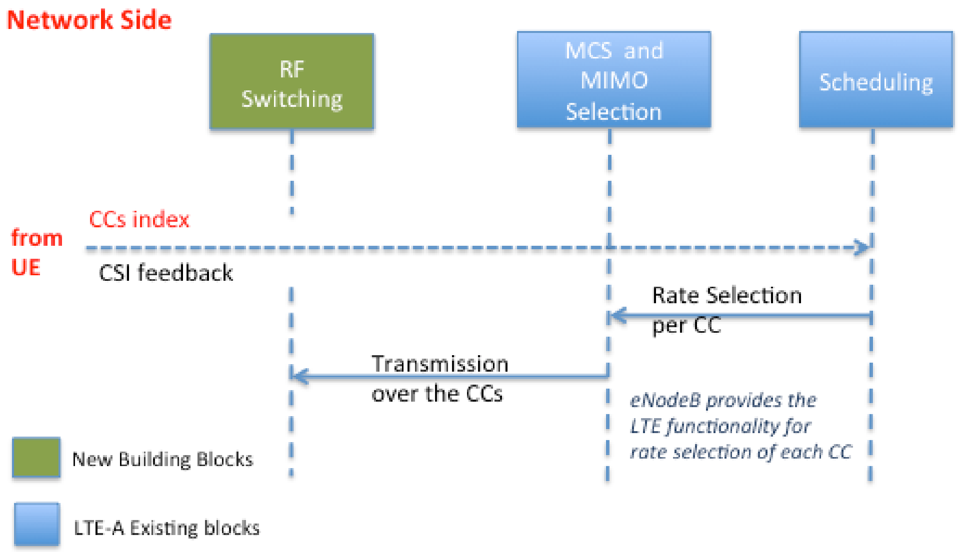


Figure 22: eNodeB to support the Dynamic Cognitive CA

4.2.5 Roles of partners and timeline

ISI will provide the first part of the dynamic cognitive CA demonstration in M24 and second part with full functionality in M36.

4.2.6 Trials

Measurements will be provided using the CRI platform. The trial will be used to provide the throughput results and how it is enhanced using the particular solution. The signaling overhead will be taken also into account.

5. Conclusions

This deliverable described the five different proof-of-concepts to be developed in the SOLDER project, along with their interfaces and timeline for development. This deliverable thereby serves as a starting point for the subsequent work in WP4, namely for the tasks 4.2 Building blocks development and 4.3 System integration. Both tasks will end one year from now, at month 30. The last 6 months of the project are reserved for field trials, fine-tuning of the PoC and data collection for the final demonstration and dissemination of the project.

References

[1] ETSI, "White Space Devices (WSD); Wireless Access Systems operating in the 470 MHz to 790 MHz frequency band; Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive," v1.1.1, April 2014.
[2] OpenAirInterface, http://www.openairinterface.org, accessed March 2015.
[3] Femto Forum, "LTE MAC Scheduler Interface Specification", FF_Tech_001_v1.11, 12-10-2010
[4] 3GPP, "Feasibility study for Further Advancements for E-UTRA (LTE-Advanced)", 3GPP TR 36.912 V9.0.0 (2009-09)
[5] 3GPP, "UTRAN overall description", 3GPP TS 25.401 V10.2.0 (2011-06)
[6] 3GPP TR 36.889 TSG RAN "Study on Licensed-Assisted Access to Unlicensed Spectrum" (Release 13), Nov. 2014.
[7] SOLDER D2.1 "Application scenarios and use cases", http://ict-solder.eu/
[8] O. Holland et al., "Development of a radio enabler for reconfiguration management within the IEEE P1900.4 working group," IEEE DySPAN 2007, Dublin, Ireland, April 2007.
[9] SOLDER D2.2 "Component-level requirements for scenarios and use cases", http://ict-solder.eu/
[10] SOLDER D2.3 "System-level requirements for scenarios and use cases", http://ict-solder.eu
[11] SOLDER D3.1, "Initial report of Heterogeneous Carrier aggregation in LTE-A and beyond", http://ict-solder.eu
[12] Tzi-Dar Chiueh, Pei-Yun Tsai. Baseband Receiver Design for Wireless MIMO-OFDM Communications, 2nd Edition, Wiley-IEEE Press, June 2012.
[13] http://www.samtec.com/standards/vita.aspx.
[14] http://www.xilinx.com/.
[15] http://www.linear.com/product/LT5538.
[16] Zarrinkoub, Houman. Understanding LTE with MATLAB: From Mathematical Modeling to Simulation and Prototyping, Wiley, March 2014.
[17] Ofcom Statement, "Implementing TV White Spaces," http://stakeholders.ofcom.org.uk/consultations/white-space-coexistence/statement, accessed March 2015.
[18] ICT-ACROPOLIS, http://www.ict-acropolis.eu, accessed March 2015.
[19] O. Holland, et. al., "A Series of Trials in the UK as part of the Ofcom TV White Spaces Pilot," IEEE CCS 2014, Rhine River, Germany, September 2014.
[20] A. Cuffaro, "Cognitive Radio Solutions: Evolving Cellular and Unlicensed Spectrum Technologies to Support Spectrum Sharing in Diverse Bands," keynote presentation at IEEE CCS 2014, Rhine River, Germany, September 2014.
[21] Zayen, Bassem; Kaltenberger, Florian; Knopp, Raymond, „OpenAirInterface and ExpressMIMO2 for spectrally agile communication," Book chapter in "Opportunistic Spectrum Sharing and White Space Access: The Practical Reality", Holland, O.; Bogucka, H.; Medeisis, A. & Gediminas, V. (Eds.), Wiley, 2015

List of Acronyms and Abbreviations

Acronym/Abbreviation	Meaning
3GPP	3rd Generation Partnership Project
4G	4th Generation
5G	5th Generation
ABS	Almost Blank Subframes
AP	Access Point
BER	Bit Error Rate
BS	Base Station
BSC	Base Station Controller
CA	Carrier Aggregation
CC	Component Carrier
CIF	Carrier Indicator Field
CoMP	Coordinated MultiPoint
CPC	Cognitive Pilot Channel
CQI	Channel Quality Indicator
CR	Cognitive Radio
CRI	Cognitive Radio Interface
CRS	Cell-specific Reference Signal
CSI	Channel State Information
D2D	Device-to-Device
DAC	Digital to Analog Converter
DFT	Discrete Fourier Transform
DL	Downlink
DPD	Digital PreDistortion
DRS	Demodulation Reference Signal
DSM	Dynamic Spectrum Management
DSS	Dynamic Spectrum Sharing
DTT	Digital Terrestrial Television
DUT	Device Under Test
EAP-SIM	Extensible Authentication Protocol Subscriber Identity Module
eICIC	Evolved Intercell Interference Coordination
EIRP	Equivalent Isotropic Radiated Power
eNB	evolved NodeB
eMBMS	evolved MBMS
E-UTRA	Evolved UMTS Terrestrial Radio Access
FAPI	FemtoForum Application Programming Interface
FBMC	Filter Bank Multi-Carrier
FCC	Federal Communications Commission
FDD	Frequency-Division Duplex
FMC	FPGA Mezzanine Card
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
GFDM	Generalized Frequency-Division Multiplexing
HARQ	Hybrid Automatic Repeat Request
HetNet	Heterogeneous Network
h-RAT	heterogeneous RAT
HSPA	High Speed Packet Access
IFFT	Inverse FFT

ILW	Integrated LTE and WiFi
IP	Internet Protocol
ISM	Industrial Scientific and Medical
LA	Link Adaptation
LAA	Licensed-Assisted Access
LPN	Low Power Node
LQM	Link Quality Metrics
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
LTE-U	LTE in Unlicensed spectrum
MAC	Medium Access Control
MAI	Multiple Access Interference
MBB	Mobile BroadBand
MBMS	Multimedia Broadcast/Multicast Service
MFCN	Mobile/Fixed Communications Networks
MIH	Media-Independent Handover
MIHF	Media-Independent Handover Function
MIMO	Multiple-Input Multiple-Output
MSS	Mobile Satellite Services
MT	Mobile Terminal
MU	Multi-User
NAS	Non-Access Stratum
NC	Non-Contiguous
NC-OFDM	Non-Contiguous Orthogonal Frequency-Division Multiplexing
NE	Network Element
OAI	OpenAirInterface
OFDM	Orthogonal Frequency-Division multiplexing
OFDMA	Orthogonal Frequency-Division Multiple Access
OSI	Open-Systems Interconnection
PA	Power Amplifier
PAPR	Peak-to-Average Power Ratio
PCC	Primary Component Carrier
PCell	Primary Cell
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PHY	Physical Layer
PMI	Precoding Matrix Indicator
PMSE	Programme Making and Special Events
PRB	Physical Resource Block
PU	Primary User
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RAT	Radio Access Technology
RF	Radio-Frequency
RFFE	RadioFrequency Front-End
RI	Rank Indicator
RRC	Radio-Resource Control
RRH	Remote Radio Head
RRM	Radio-Resource Management

RSRP	Reference Signal Received Power
SAP	Service Access Point
SAW	Surface Acoustic Wave
SCC	Secondary Component Carrier
SCell	Secondary Cell
SC-FDMA	Single-Carrier FDMA
SINR	Signal-to-Interference- Plus-Noise Ratio
SME	Spectrum Management Entity
SOTA	State-Of-The-Art
SRS	Sounding Reference Signal
SU	Secondary User
SS	Spectrum Selection
TA	Timing Advance
TAIC	Terminal Aggregation Information Controller
TDD	Time-Division Duplex
TDM	Time-Division Multiplexing
TVWS	TV White Space
UCI	User Control Information
UE	User Equipment
UFMC	Universal Filtered Multi-Carrier
UL	Uplink
U-NII	Unlicensed National Information Infrastructure
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local-Area Network
WWAN	Wireless Wide-Area Network

Appendix

A1. CRI Platform: HW description

The *Cognitive Radio Interface* (**CRI**) platform is developed by the Athena_RC/ISI emulating a LTE PHY layer testbed for the *Base Station* (BS) and the *User Equipment* (UE). The LTE testbed platform is equipped with all the appropriate mechanisms in order to be able to contribute to two PoC scenarios namely dynamic cognitive CA for HetNets deployment and licensed/unlicensed bands aggregation for h-RATs deployment (Poc1b and PoC3b). We give below a figure (Figure 23) showing the “big picture” of the whole system. More specifically the baseband processing of the received/transmitted signal is carried out on a *Field Programmable Gate Array* (**FPGA**) while the *Radio Frequency* (**RF**) processing of the received/transmitted signal is implemented on a *daughter card* using discrete analog components. The digital components are interacting with the analog components via the FMC (FPGA Mezzanine Card) connector existing on the evaluation board that carries the FPGA.

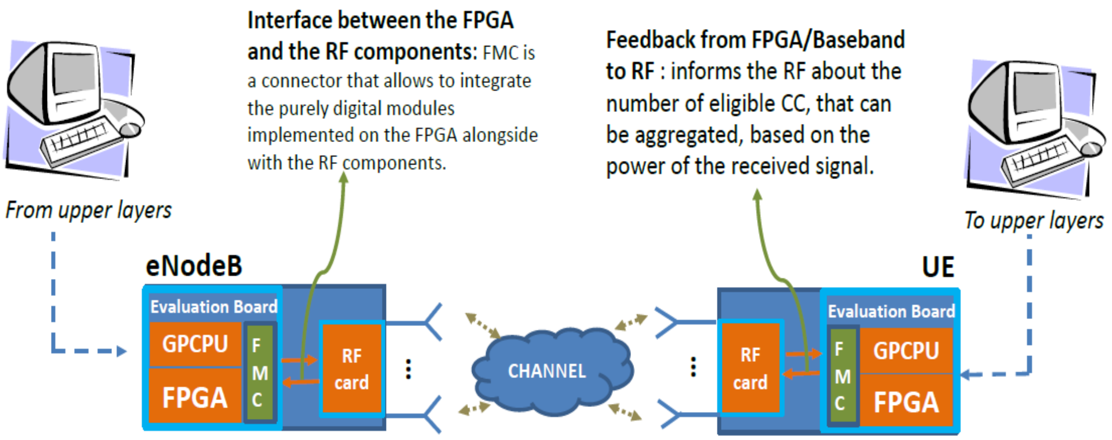


Figure 23 Overview of the CRI platform

RF Prototype

Figure 24 illustrates the architecture of the Daughter-Card, which will be used for both the eNodeB and the UE. The daughter card uses the Samtec High Speed Connectors to connect with the FMC connectors of the FPGA [13]. The Daughter card consists of 4 building blocks. The first building block is the Tx/Rx chains, the second building block is the RFFE (Radio Frequency Front-End), the third one is the power detector and the last one is the SWITCHING. Details about these blocks are given to the architecture in the main sections of this document.

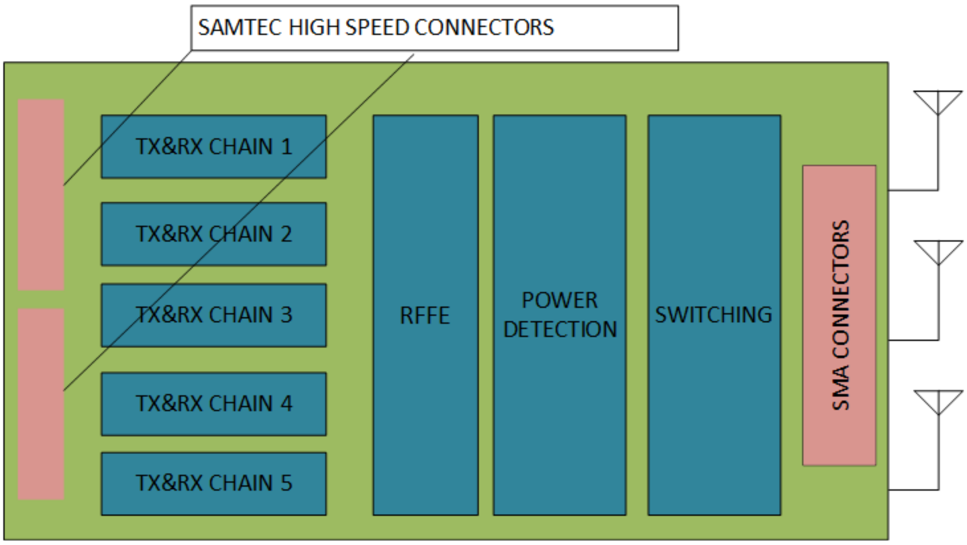


Figure 24 Daughter-Card

Baseband SDR/FPGA platform
The eNodeB is implemented in a Xilinx Zynq-7000 SoC ZC706 board that has a hard-IP dual ARM Cortex-A9 core processor, providing computational power for possible extensions of the CRI platform for upper layers in terms of software, like MAC layer. Alongside, with the eNodeB the integration of the UE is took part in a Xilinx Virtex-6 XC6VLX240T board [14]. Both evaluation board have JTAG and USB connector for connection with a PC, and FMC connectors for the communication with the RF part. The two boards, with all the capabilities that they offer are shown below in Figure 26 [14].

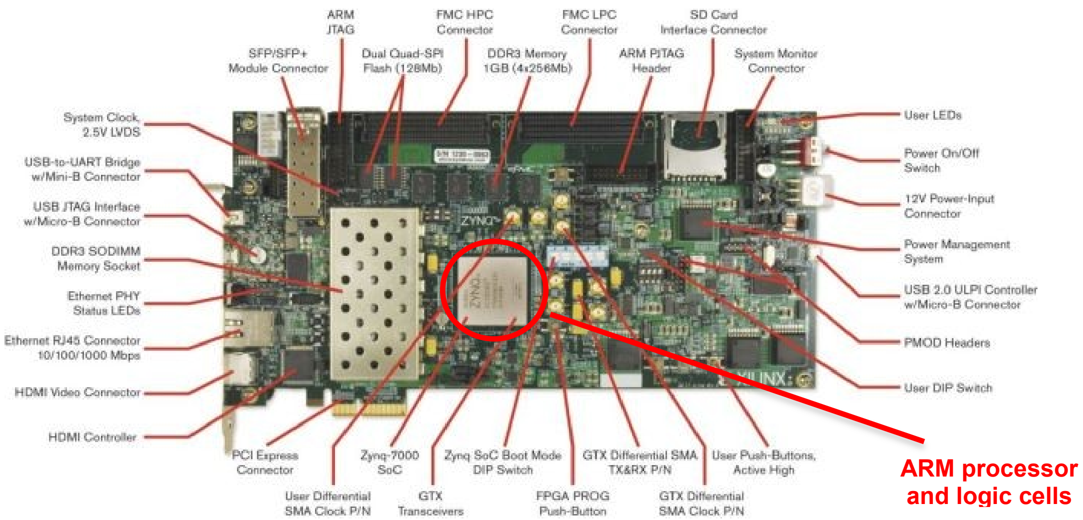


Figure 25 Xilinx platform for eNodeB [14]

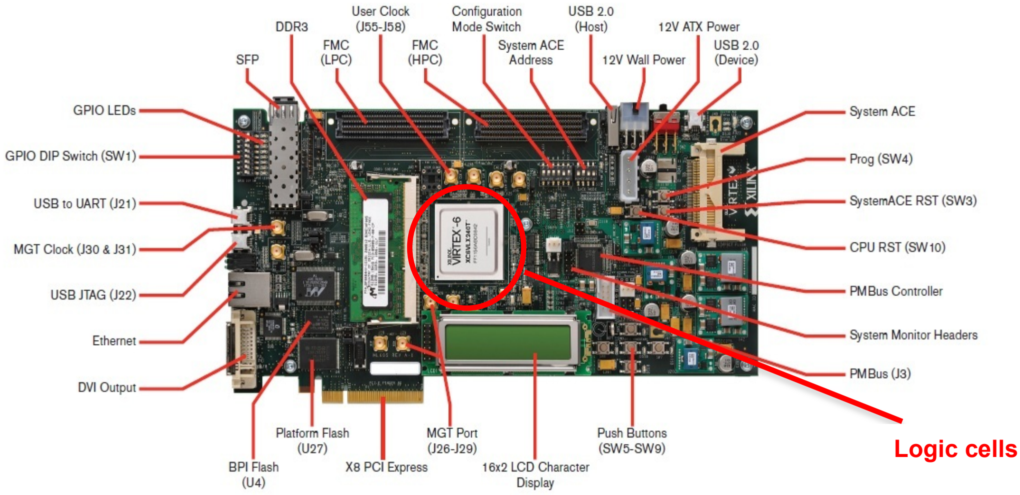


Figure 26 Xilinx platform for UE [14]

A2. CRI Platform: Building Blocks

The RF prototype consists of the 3 building blocks as depicted in Figure 24. More details about the cognitive radio blocks of the prototype are given below:

- **Tx/Rx chains:** Four chains are built up by LMS6002D transceivers to form the Tx/Rx chains for the licensed spectrum bands with a range of frequencies from 300 MHz to 3800 MHz. Moreover the LMS6002D has a tunable bandwidth of up to 28 MHz which is adequate to support the CA scenarios, which will be implemented in this platform. These modules are also capable to support both FDD and TDD operation. The fifth Tx/Rx chain uses the AD9361 chip which is a second generation transceiver with MIMO capabilities and supports a wider range of frequencies from 70 MHz to 6GHz. This chip has a tunable bandwidth of up to 56 MHz, supports both FDD and TDD operation modes and has a superior receiver sensitivity with a noise figure < 2.5 dB.
- **RFFE (RF Front End)**: This block contains Power amplifiers, LNA amplifiers, band-pass filters, diplexers, attenuators, switches and Power detectors. The signals are amplified (attenuated) depending on the level of the power needed for the implementation of each CA scenario.
- **Power detector**; the platform includes RF power detectors which consist of cascaded amplifiers and RF detectors. The output currents from every RF detector are combined and low-pass filtered before applied to the output buffer amplifier. As a result, the final DC output voltage approximates the logarithm of the amplitude of the input signal. Thus, when the baseband receives the feedback regarding the output voltage, by using an adaptive algorithm it can determine whether a signal is present at the channel or not [15].
- **SWITCHING:** This building block consists of switches and power combiners/splitters for both the licensed and unlicensed spectrum aggregation. Through the right switching the signals after the amplification and filtering can be driven to the combiners in order to be aggregated and ready to be sent to the antennas for transmission. Respectively in the reception case the combiners work as power splitters in order to split the signal and drive it to the RFFE through the right switching.

Baseband architecture

The processing in the baseband is LTE-A compatible and includes the main building PHY layer blocks (i.e. functionalities), e.g. MCS, OFDM,, MIMO, channel estimation etc. In the context of SOLDER, we need to build more blocks (with cognitive radio capabilities according to the D2.3 system-level requirements) that support the two main PoCs that the CRI platform can demonstrate. The main architecture of the PHY LTE-A testbed architecture of the CRI platform is depicted in Figure 27 below. With green font are depicted the new introduced blocks that are required for the 2 PoCs as explained in details in section 2.2 and 4.2. We also show for clarity purpose the blocks situated to the RF prototype described above.

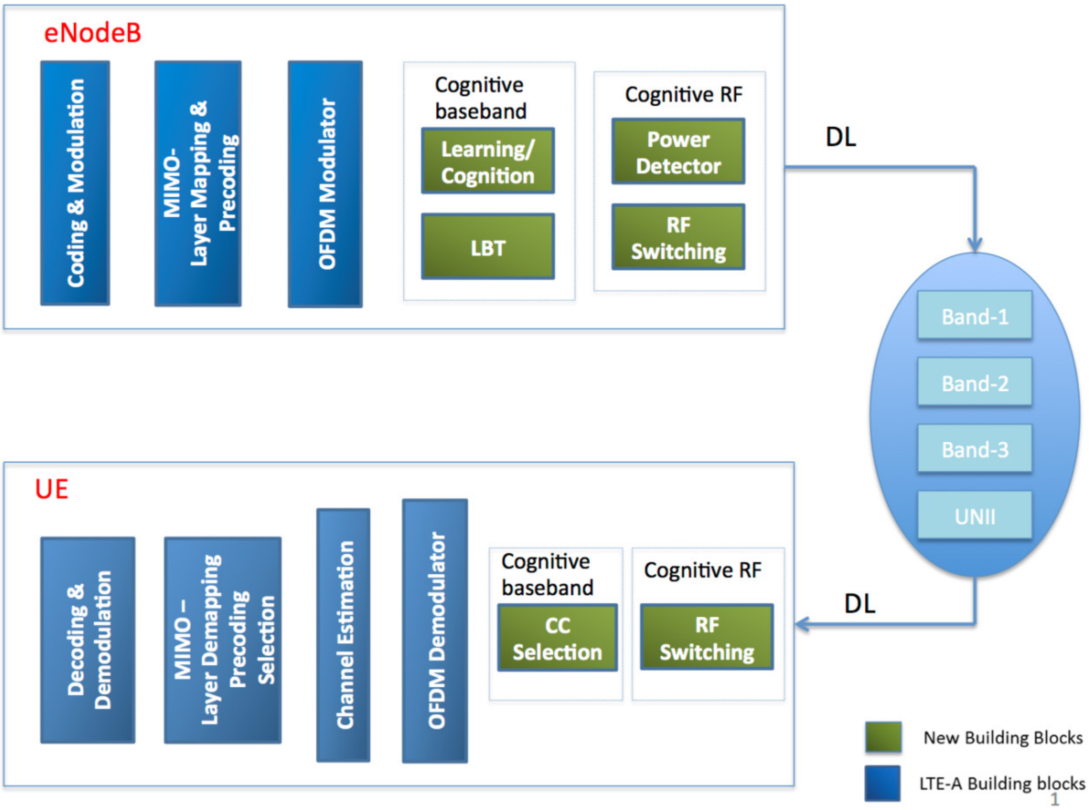


Figure 27: Building blocks of the LTE testbed of the CRI platform

A3. FAPI MAC Scheduler Interface extensions

1. Abstract

This appendix describes all the extensions that are needed in FAPI (FemtoForum Application Programming Interface) MAC scheduler interface defined by Small Cell Forum (formerly known as Femto Forum) [3] to support Carrier Aggregation feature introduced in rel. 10 LTE.

2. General context

This chapter aims to describe how the FAPI based MAC scheduler (with surroundings) looks like from high architectural perspective. In Figure 28 below following assumptions are made:

- implementation of the scheduler(s) is done by IS-Wireless and it's closed source (however in more general approach the scheduler might be implemented by anyone – the only condition is – it must follow the defined interface)
- the other side of the interface is Open Air Interface LTE stack (however in more general approach it might be any stack implementation – the only condition is – it must follow the defined interface)

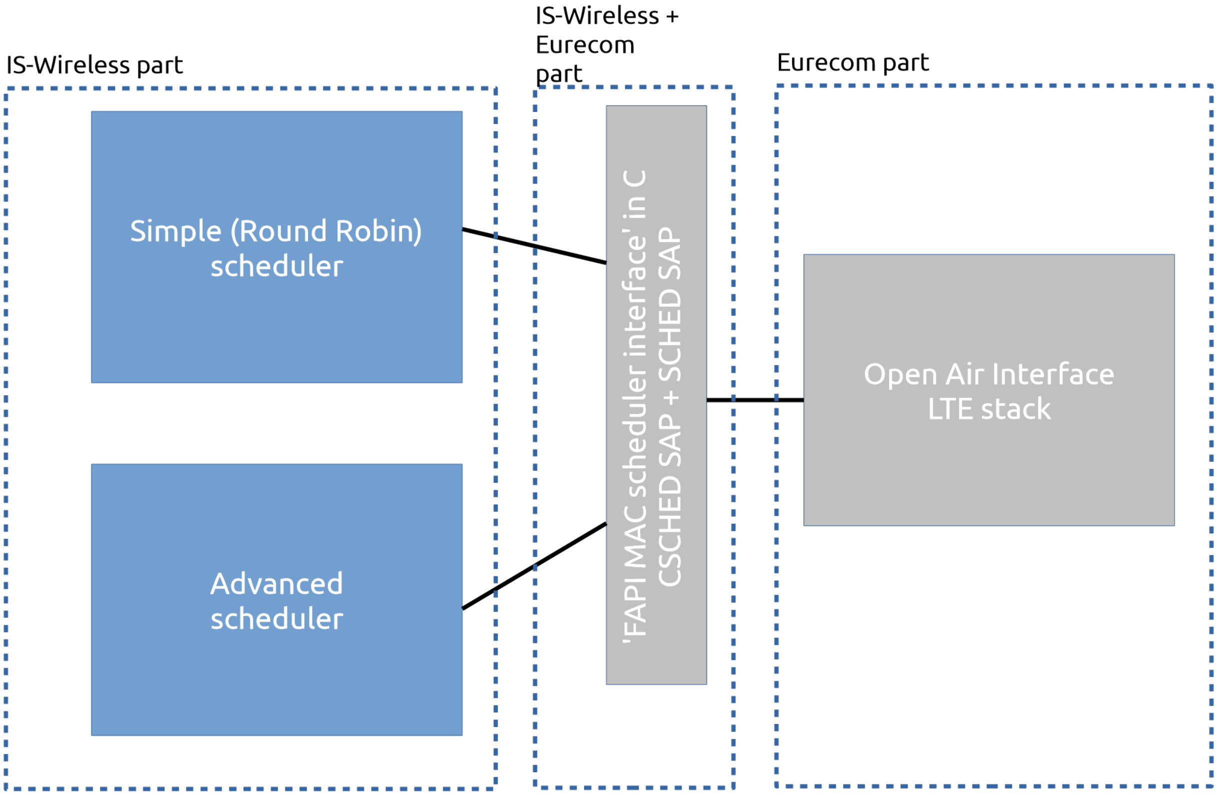


Figure 28: Building blocks and interfaces of the MAC scheduler

3. Interface extensions/modifications

Extensions and modifications are split into 4 sections: CSCHED, SCHED, Parameters and Constants – exactly the same as in the original interface description[3]. Each change is either a modification of the original chapter from [3] or it's something new – this is also indicated.

For each modification there is a chapter mentioned (in parenthesis) that describes the primitive/parameter in original specification [3]. Each change is briefly described why it's needed.

3.1 CSCHED SAP

3.1.1 CSCHED_CELL_CONFIG_REQ Parameters (4.1.1) – MODIFICATION

It shall be possible to send cell configuration for each active component carrier. CSCHED_CELL_CONFIG_REQ parameters shall be modified and look like below:

Elements	Value	Type	Description
nr_carriers	0..MAX_NUM_CCs	uint8_t	The number of elements in the next array
ccConfigList	Variable-length array of ccConfigListElement. See 3.3.1	array of struct	The list of component carrier's configurations.
nr_vendorSpecificList	As stated in [1]		
vendorSpecificList	As stated in [1]		

3.1.2 CSCHED_UE_CONFIG_REQ Parameters (4.1.3) – MODIFICATION

It shall be possible to signal UE CA configuration. Add following elements to the table:

Elements	Value	Type	Description
caSupport	true/false	bool	Indicates if the UE supports CA
crossCarrierSchedSupport	true/false	bool	Indicates if the supports cross carrier scheduling
pcellCarrierIndex	0..256	uint8_t	Carrier index of the UE's PCell. Indicates which of available carriers in eNB is the PCell for the UE, see 4.2
nr_scells	0..MAX_NUM_CCs-1	uint8_t	Number of SCells configured for the UE
scellConfigList	Variable-length array of scellConfig. See 3.3.2	array of struct	The list of SCell configurations
scellDeactivationTimer	0..65535	uint16_t	SCell deactivation timer as described in 36.321.

3.1.3 CSCHED_CELL_CONFIG_UPDATE_IND Parameters (4.1.12) – MODIFICATION

It shall be possible to report PRB utilization measurements for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.2 SCHED SAP

3.2.1 SCHED_DL_CONFIG_IND Parameters (4.2.7) – MODIFICATION

It shall be possible to send scheduling information per each component carrier. This requirement is partly fulfilled by means of SCHED_DL_CONFIG_IND internal parameters modifications (see 3.3.4, 3.3.5 and 3.3.6) but one additional change is needed – the type of *nrOf_PDCCH_OFDM_Symbols* element shall change and 1 additional parameter is needed - like below:

Elements	Value	Type	Description
nr_ofdmSymbolsCount	0..MAC_NUM_CCs	uint8_t	Number of elements in the next array
nrOf_PDCCH_OFDM_Symbols	Variable-length array of pdcchOfdmSymbolCountListElement. See 3.3.3	array of struct	Indicates number of OFDM symbols reserved for PDCCH for each component carrier

3.2.2 SCHED_UL_NOISE_INTERFERENCE_REQ Parameters (4.2.9) – MODIFICATION

It shall be possible to report interference and noise measurements for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3 Parameters

3.3.1 ccConfigListElement – NEW

This structure should contain all original CSCHED_CELL_CONFIG_REQ parameters (defined in [3] 4.1.1) plus one additional element described below.

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.2 scellConfig – NEW

This structure is needed to describe SCell configuration. It is partly based on RRC IE CrossCarrierSchedulingConfig. It shall contain following elements:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2
scellIndex	1..7	uint8_t	SCell index as to be sent in RRCConnectionReconfiguration (IE SCellIndex defined in RRC specification), see 4.2
uesCrossCarrier-Scheduling	true/false	bool	Indicates if cross carrier scheduling shall be used on this SCell. If so, detailed cross configuration is given in next 2 elements.
schedulingCellIndex	0..7	uint8_t	Indicates which cell signals the downlink allocations and uplink grants, if applicable, for the concerned SCell
pdschStart	1..4	uint8_t	Starting OFDM symbol of PDSCH data region for this Scell

3.3.3 *pdcchOfdmSymbolCountListElement – NEW*

This structure is needed to indicate number of OFDM symbols for PDCCH for specific component carrier. It shall contain following elements:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2
pdcchOfdmSymbol-Count	0..4	uint8_t	Current size of PDCCH

3.3.4 *buildDataListElement (4.3.8) – MODIFICATION*

It shall be possible to generate data and DCIs for each component carrier. Also, as De/Activation MAC CE is built by the scheduler (see 4.1), there must be way to send this MAC CE data when signalling the scheduling decision. Add following elements to the table (the last element in the below table shall replace the same element from the original table in[3]):

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2
activationDeactiva-tionCE	0x00..0xFF	uint8_t	Activation/Deactivation MAC Control Element built by the scheduler, to be sent in this TTI
ceBit-map[MAX_TB_LIST]	TA, DRX, CR, AD	uint8_t bitmap	The CEs scheduled for transmission for this TB (TA-Timing Advance Command, DRX-DRX Command, CR-Contention Resolution, AD-Activation/Deactivation)

3.3.5 *buildRAR_ListElement (4.3.10) – MODIFICATION*

It shall be possible to generate RAR data for each component carrier. Add following element to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.6 *buildBroadcastListElement (4.3.11) – MODIFICATION*

It shall be possible to generate broadcast data for each component carrier. Add following element to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.7 *dlDciListElement (4.3.1) – MODIFICATION*

Due to potential use of cross-carrier scheduling the DL DCI shall contain carrier indicator field. (mind that information 'which carrier shall be used to transmit this DCI' is contained in the structure containing dlDciListElement). Add following elements to the table:

Elements	Value	Type	Description
cifPresent	true/false	bool	Indicates if carrier indicator field

			shall be present in DCI
cif	0..7	uint8_t	CIF value

3.3.8 ulDciListElement (4.3.2) – MODIFICATION

Due to potential use of cross-carrier scheduling the UL DCI shall contain carrier indicator field. Also the structure shall contain carrier index to inform on which carrier this DCI shall be transmitted. Add following elements to the table:

Elements	Value	Type	Description
cifPresent	true/false	bool	Indicates if carrier indicator field shall be present in DCI
cif	0..7	uint8_t	CIF value
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.9 cqiListElement (4.3.24) – MODIFICATION

It shall be possible to specify which carrier the CQI concerns. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.10 ulCQI (4.3.29) – MODIFICATION

It shall be possible to specify channel quality for each carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.11 dlInfoListElement (4.3.23) – MODIFICATION

It shall be possible to specify HARQ status for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.12 rachListElement (4.3.6) – MODIFICATION

It shall be possible to specify RACH for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.13 phichListElement (4.3.7) – MODIFICATION

It shall be possible to signal HARQ on PHICH for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.3.14 ulInfoListElement (4.3.12) – MODIFICATION

It shall be possible to deliver amount of received UL data and reception status for each component carrier. Add following elements to the table:

Elements	Value	Type	Description
carrierIndex	0..256	uint8_t	Component carrier identifier, uniquely identifies carrier in the eNB, see 4.2

3.4 Constants

3.4.1 MAX_NUM_CCs – NEW

New constant shall be defined that indicates maximum number of component carriers assigned to UE.

Elements	Value	Description
MAX_NUM_CCs	2	Maximum number of component carriers assigned to UE

4. Additional information

4.1 SCell activation/deactivation

Decision about SCell activation/deactivation is always taken inside the scheduler and in such cases the scheduler generates proper MAC CE and schedules it for sending to the UE.

4.2 PcellIndex/ScellIndex'ing

There are 2 kinds of component carrier indices in the interface:

- carrierIndex – this is global eNB identifier that clearly and uniquely identifies component carrier in eNB. It does not need to be related to any 3gpp defined indices, it is not related to RRC ServCellIndex value at all. Scheduler (and the other side of the interface) uses this index to signal which component carrier the data concerns.
- scellIndex – this is 1 to 1 copy of SCellIndex from RRCConnectionReconfiguration message that configures SCell(s) for the UE. It is used to identify SCells for given UE. Scheduler uses this index to:
    - generate proper Activation/Deactivation MAC Control Elements
    - generate proper CIF content in DCIs in case of cross carrier scheduling

Femto Forum Ltd . PO Box 23 . Dursley . GL11 . 5WA . UK
tel +44 (0)845 644 5823 fax +44 (0)845 644 5824 email info@femtoforum.org www.femtoforum.org

femtoforum
driving convergence worldwide

LTE MAC Scheduler Interface Specification v1.11

Femto Forum Technical Document

Document number:	FF_Tech_001_v1.11
Date issued:	*12-10-2010*
Document status:	*Document for public distribution*

Legal Notice

Permitted Use of this document
You are permitted to download, use and distribute copies of this document provided that:

(a) you must only use and distribute this document in its entirety without amendment, deletion or addition of any legal notice, text, graphics or other content; and

(b) you must not make this document available for download on any publically accessible bulletin board, website, ftp site or file sharing service.

Disclaimer
This document is provided on an 'as is' basis without guarantees, representations, conditions or warranties as to its accuracy or completeness or that it is free from error. To the extent permitted by law, the Femto Forum Ltd and the contributors to this document exclude all representations, conditions, warranties and other terms which might otherwise be implied by statute, common law or the law of equity.

Patents
It is possible that use of the technical matter published in this document may require the permission of the proprietor of one or more patents. You are entirely response for identifying and where necessary obtaining a licence under such patents should you choose to use any such technical matter. The Femto Forum Ltd has no responsibility in this regard and shall not be liable for any loss or damage suffered in relation to an infringement of any third party patent as a result of such use.

Copyright
This document is subject to copyright owned by the Femto Forum Ltd and/or licensed to the Femto Forum Ltd by its contributing members. You may use and distribute this document free of charge provided that you comply with the provisions set out in this notice. Other than this limited licence, you are not granted any further right, interest or title in this document and the Femto Forum Ltd and/or its contributing members shall at all times remain the sole owner(s) of the copyright in this document.

Trade Marks
The Femto Forum logo and other logo, trade and service marks contained in this document are the property of the Femto Forum Ltd and, where applicable, other third parties. You are not permitted to use or reproduce these marks without the prior written consent of the Femto Forum Ltd or where applicable the third party owner.

1 Table of Contents

2 Scope

This document specifies the MAC Scheduler interface. The goal of this interface specification is to allow the use of a wide range of schedulers which can be plugged into the eNodeB and to allow for standardized interference coordination interface to the scheduler.

3 Interface Overview

The MAC scheduler is part of MAC from a logical view and the MAC scheduler should be independent from the PHY interface.

The description in this interface does not foresee any specific implementation of the interface. What is specified in this document is the structure of the parameters. In order to describe the interface in detail the following model is used:

The interface is defined as a service access point offered by the MAC scheduler to the remaining MAC functionality, as shown in Figure 1. A _REQ primitive is from MAC to the MAC scheduler. A _IND/_CNF primitives are from the MAC scheduler to the MAC. The description using primitives does not foresee any specific implementation and is used for illustration purposes. Therefore an implementation could be message-based or function-based interface. Timing constrains applicable to the MAC scheduler are not yet specified.

For the MAC scheduler interface specification a push-based concept is employed, that is all parameters needed by the scheduler are passed to the scheduler at specific times rather than using a pull-based concept (i.e. fetching the parameters from different places as needed). The parameters specified are as far as possible aligned with the 3GPP specifications

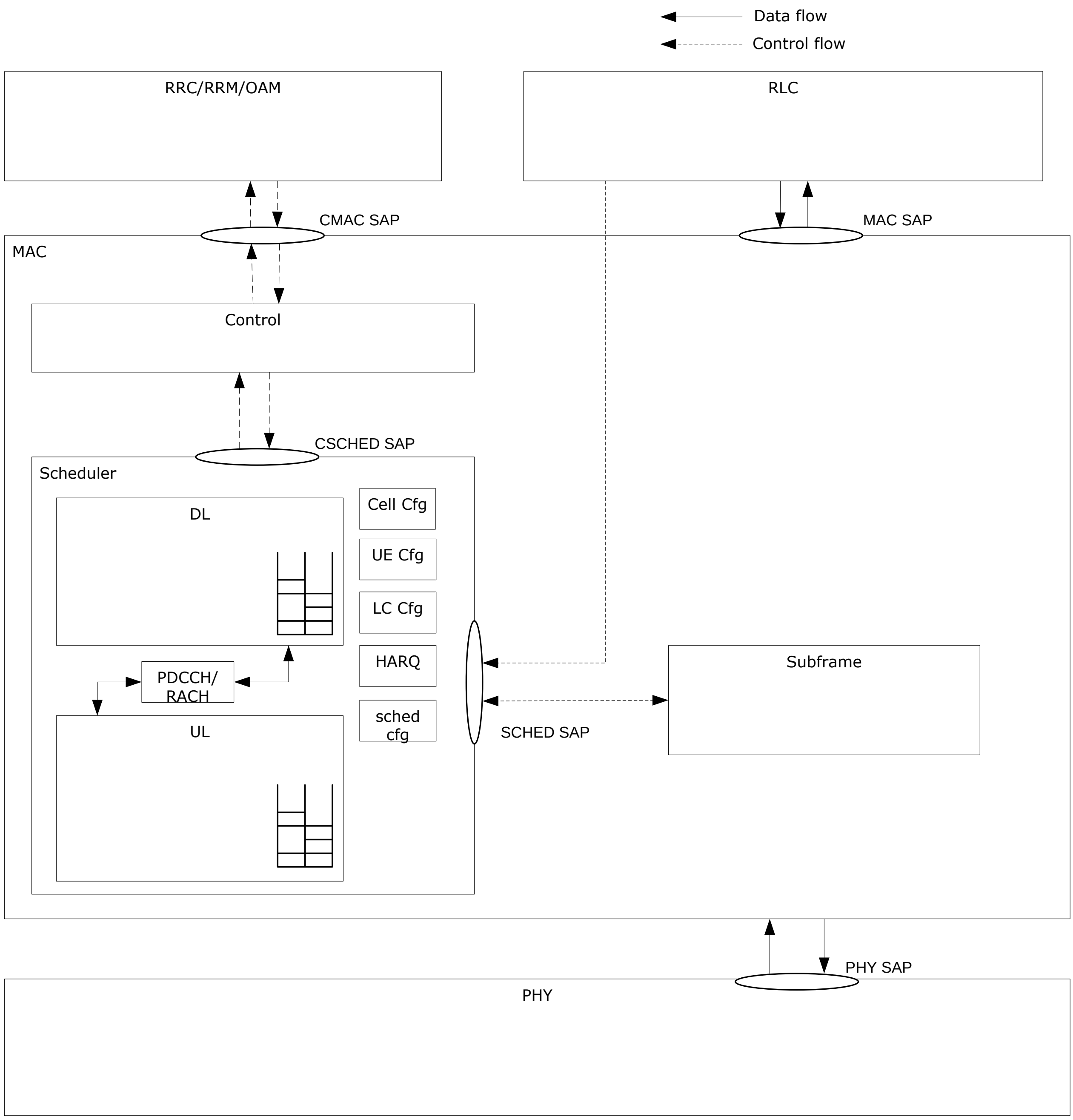


Figure 1: MAC scheduler interface overview

Figure 1 shows the functionality split between the MAC scheduler and the remaining MAC. For the purposes of describing the MAC scheduler interface the MAC consists of a control block and a subframe block, which uses the CSCHED and SCHED SAP respectively. The subframe block triggers the MAC scheduler every TTI and receives the scheduler results. The control block forwards control information to the MAC scheduler as necessary. The scheduler consists of the following blocks:

UL Is responsible for scheduling of the PUSCH resources.

DL Is responsible for scheduling of the PDSCH resources.

PDCCH/RACH Is responsible for shared resources between UL and DL.

HARQ Is responsible for handling HARQ retransmissions, keeping track of the number of retransmissions and redundancy versions.

Cell Cfg Stores the UE configuration needed by the MAC scheduler.

UE Cfg Stores the UE configuration needed by the MAC scheduler.
LC Cfg Stores the logical channel configuration needed by the MAC scheduler.
Sched Cfg Stores the scheduler-specific configuration needed by the MAC scheduler.

4 Detailed interface description

In the following section the messages exchanged at the SAPs are specified.

4.1 CSCHED – MAC Scheduler Control SAP

Table 1 specifies which configuration messages can be used to configure the scheduler.

There is no restriction on the timing of when these messages can be sent, except where otherwise noted in the message description. The RNTI and, if available, the LCID are used to identity the UE/LC between the MAC scheduler and the MAC. In case of a reconfiguration message all parameters previously configured in a message have to be resend, otherwise parameters not present are removed in the scheduler (i.e. no delta configuration is possible)

Primitive name	Reference	Description
CSCHED_CELL_CONFIG_REQ	4.1.1	(re-)configure MAC scheduler with cell configuration and scheduler configuration. The cell configuration will also setup the BCH, BCCH, PCCH and CCCH LC configuration.
CSCHED_CELL_CONFIG_CNF	4.1.2	Cell configuration and scheduler configuration applied
CSCHED_UE_CONFIG_REQ	4.1.3	(re-)configure MAC scheduler with UE specific parameters. A UE can only be configured when a cell configuration has been received.
CSCHED_UE_CONFIG_CNF	4.1.4	UE specific configuration applied
CSCHED_LC_CONFIG_REQ	4.1.5	(re-)configure MAC scheduler with logical channel configuration. A logical channel can only be configured when a UE configuration has been received.
CSCHED_LC_CONFIG_CNF	4.1.6	Logical channel configuration applied
CSCHED_LC_RELEASE_REQ	4.1.7	release a logical channel in the MAC scheduler. A logical channel can only be released if it has been configured previously.
CSCHED_LC_RELEASE_CNF	4.1.8	Logical Channel specific configuration removed
CSCHED_UE_RELEASE_REQ	4.1.9	release a UE in the MAC scheduler. The release of the UE configuration implies the release of LCs, which are still active. A UE can only be released if it has been configured previously.

CSCHED_UE_RELEASE_CNF	4.1.10	UE specific configuration removed
CSCHED_UE_CONFIG_UPDATE_IND	4.1.11	update of UE specific parameters from MAC scheduler to RRC
CSCHED_CELL_CONFIG_UPDATE_IND	4.1.12	update of ctrl channel configuration from MAC scheduler to RRC

Table 1: CSCHED primitives

4.1.1 CSCHED_CELL_CONFIG_REQ Parameters

Elements	Value	Type	Description
pusch_HoppingOffset	0..98	uint8_t	PUSCH resources in RBs. used for hopping. see [2] section 5.3.4
hoppingMode	inter, interintra	enum	see [2] section 5.3.4
n_SB	1,2,3,4	uint8_t	number of subbands. see [2] section 5.3.4
phichResource	PHICH_R_ONE_SIXTH, PHICH_R_HALF, PHICH_R_ONE, PHICH_R_TWO	enum	The number of resources element groups used for PHICH.
phichDuration	normal, extended	enum	see [2] table 6.9.3-1
initialNrOf_PDCCH_OFDM_Symbols	0..4	uint8_t	Nr of PDCCH OFDM symbols. see [2] section 6.9
siConfiguration	see section 4.3.21	struct	The SI configuration
ul_Bandwidth	6,15,25,50,75,100	uint8_t	UL transmission bandwidth in RBs
dl_Bandwidth	6,15,25,50,75,100	uint8_t	DL transmission bandwidth in RBs
ul_CyclicPrefixLength	normal, extended	enum	see [2] section 5.2.1
dl_CyclicPrefixLength	normal, extended	enum	DL cyclic prefix.
antennaPortsCount	1,2,4	uint8_t	Number of cell specific antenna ports. see [2] section 6.2.1
duplexMode	DM_TDD, DM_FDD	enum	Cell is configured in TDD or FDD mode.
subframeAssignment	0..6	uint8_t	DL/UL subframe

			assignment. Only TDD. see [2] table 4.2.2
specialSubframePatterns	0..8	uint8_t	TDD configuration. Only TDD. see [2] table 4.2.1
mbsfn_SubframeConfigPresent	TRUE, FALSE	bool	Indicates if the following mbsfn_ fields are valid or not.
mbsfn_SubframeConfig_RFPeriod [MAX_MBSFN_CONFIG]	1,2,4,8,16.32	uint8_t	The MBSFN radio frame period
mbsfn_SubframeConfig_RFOffset [MAX_MBSFN_CONFIG]	0..7	uint8_t	The radio frame offset
mbsfn_SubframeConfig_SFallocation [MAX_MBSFN_CONFIG]	bitmap 0..9	uint8_t	Indicates the MBSFN subframes
prachConfigurationIndex	0..63	uint8_t	see [2] section 5.7.1
prach_FreqOffset	0..94	uint8_t	see [2] section 5.7.1
ra_ResponseWindowSize	2..8,10	uint8_t	Duration of RA response window in SF. see [1]
mac_ContentionResolutionTimer	8,16,24,32,40,48,56,64	uint8_t	Contention resolution timer used during random access. see [1].
maxHARQ_Msg3Tx	1..8	uint8_t	see [1]
n1PUCCH_AN	0..2047	uint16_t	see [4] section 10.1
deltaPUCCH-Shift	1..3	uint8_t	see [2] section 5.4
nRB_CQI	0..98	uint8_t	see [2] section 5.4
nCS_AN	0..7	uint8_t	see [2] section 5.4
srsSubframeConfiguration	0..15	uint8_t	see [2] table 5.5.3.3-1 and 5.5.3.3-2
srsSubframeOffset	0..9	uint8_t	see [2] section 5.5.3.2
srsBandwidthConfiguration	0..7	uint8_t	SRS bandwidth. see [2] section 5.5.3.2
srsMaxUpPts	TRUE, FALSE	bool	see [2] section 5.5.3.2. Only TDD
enable64QAM	MOD_16QAM, MOD_64QAM	enum	maximum UL modulation supported. see [4] section 8.6.1.

nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.2 CSCHED_CELL_CONFIG_CNF Parameters

Elements	Value	Type	Description
result	SUCCESS, FAILURE	enum	The outcome of the request
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.3 CSCHED_UE_CONFIG_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
reconfigureFlag	TRUE, FALSE	bool	Indicates if this is a reconfiguration for an existing UE or if a new UE is added.
drxConfigPresent	TRUE, FALSE	bool	Indicates if the drx sub-structure is valid or not.
drxConfig	see 4.3.16	struct	The DRX configuration
timeAlignmentTimer	500,750,1280,1920,2560,5120,10240,inf	uint16_t	in subframes. see[1]. Used for controlling synchronization status of the UE, not for the actual timing advance procedure.
measGapConfigPattern	MGP_GP1, MGP_GP2, OFF	enum	Specifies the measurement gap configuration or that it is not applicable. see 3GPP TS 36.133
measGapConfigSubframeOffset	0..79	uint8_t	Specifies the measurement gap offset, if applicable. see 3GPP TS 36.133
spsConfigPresent	TRUE, FALSE	bool	Indicates if the next sub-structure is valid or not
spsConfig	see 4.3.17	struct	The SPS configuration
srConfigPresent	TRUE, FALSE	bool	Inidicates if srConfig struct is present
srConfig	see 4.3.18	struct	The SR configuration request
cqiConfigPresent	TRUE, FALSE	bool	Inidicates if cqiConfig struct is present
cqiConfig	see 4.3.19	struct	The CQI configuration request
transmissionMode	1..7	uint8_t	The configured transmission mode. see

			[4] section 7.1
ue_AggregatedMaximumBitrateUL	0..10000000000	uint64_t	aggregated bit rate of non-gbr bearer per UE. see 3GPP TS 36.413
ue_AggregatedMaximumBitrateDL	0..10000000000	uint64_t	aggregated bit rate of non-gbr bearer per UE. see 3GPP TS 36.413
ue_Capabilities	see 4.3.20	struct	The UE capabilities
ue_TransmitAntennaSelection	none, openLoop, closedLoop	enum	see [4] section 8.7
ttiBundling	TRUE, FALSE	bool	see [1]
maxHARQ_Tx	1..8,10,12,16,20,24, 28	uint8_t	The maximum HARQ retransmission for uplink HARQ. see [1].
betaOffset_ACK_Index	0..15	uint8_t	see [4] table 8.6.3-1
betaOffset_RI_Index	0..15	uint8_t	see [4] table 8.6.3-2
betaOffset_CQI_Index	0..15	uint8_t	see [4] table 8.6.3-3
ackNackSRS_SimultaneousTransmission	TRUE, FALSE	bool	see [4] section 8.2
simultaneousAckNackAndCQI	TRUE, FALSE	bool	see [4] section 10.1
aperiodicCQI_RepMode	rm12, rm20,rm22, rm30, rm31, None	enum	Reporting mode for aperiodic CQI. see [4] section 7.2.1
tdd_AckNackFeedbackMode	bundling,multiplexing	enum	see [3] section 7.3. Only TDD
ackNackRepetitionFactor	0,2,4,6	uint8_t	see [4] section 10.1. 0 means no repetition
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.4 CSCHED_UE_CONFIG_CNF Parameters

Elements	**Value**	**Type**	**Description**
rnti	1..65535	uint16_t	The RNTI identifying the UE.
result	SUCCESS, FAILURE	Enum	The outcome of the request
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.5 CSCHED_LC_CONFIG_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
reconfigureFlag	TRUE, FALSE	bool	Indicates if this is a reconfiguration for an existing UE or if a new UE is added.
nr_logicalChannelConfigList	1..MAX_LC_LIST	uint8_t	The number of array elements.
logicalChannelConfigList [MAX_LC_LIST]	An array of LogicalChannel ConfigListElement see 4.3.4	array of struct	The array of logical channel configurations to be configured.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.6 CSCHED_LC_CONFIG_CNF Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
result	SUCCESS, FAILURE	enum	The outcome of the request
nr_logicalChannelIdentity	1..MAX_LC_LIST	uint8_t	The number of array elements.
logicalChannelIdentity [MAX_LC_LIST]	An array of integers with range 1..10	array of uint8_t	The array of logical channel ID which have been configured/updated.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.7 CSCHED_LC_RELEASE_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
nr_logicalChannelIdentity	1..MAX_LC_LIST	uint8_t	The number of array elements.
logicalChannelIdentity	An array of integers with	array of uint8_t	The array of logical channel ID

[MAX_LC_LIST]	range 1..10		which shall be released.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.8 CSCHED_LC_RELEASE_CNF Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
result	SUCCESS, FAILURE	enum	The outcome of the request
nr_LogicalChannelIdentity	1..MAX_LC_LIST	uint8_t	The number of array elements.
logicalChannelIdentity [MAX_LC_LIST]	An array of integers with range 1..10	array of uint8_t	The array of logical channel ID which have been released.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.9 CSCHED_UE_RELEASE_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.10 CSCHED_UE_RELEASE_CNF Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
result	SUCCESS, FAILURE	enum	The outcome of the request
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.11 CSCHED_UE_CONFIG_UPDATE_IND Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
transmissionMode	1..7	uint8_t	The configured transmission mode. see [4] section 7.1
spsRequestPresent	TRUE, FALSE	bool	Inidicates if spsRequest struct is present
spsRequest	see 4.3.17	struct	The SPS configuration request
srRequestPresent	TRUE, FALSE	bool	Inidicates if srRequest struct is present
srRequest	see 4.3.18	struct	The SR configuration request
cqiRequestPresent	TRUE, FALSE	bool	Inidicates if cqiRequest struct is present
cqiRequest	see 4.3.19	struct	The CQI configuration request
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificList Element see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.1.12 CSCHED_CELL_CONFIG_UPDATE_IND Parameters

Elements	Value	Type	Description
prbUtilizationDL	0..100	uint8_t	Percentage as defined in 36.314
prbUtilizationUL	0..100	uint8_t	Percentage as defined in 36.314
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2 SCHED - MAC Scheduler SAP

Primitive name	Reference	Description
SCHED_DL_RLC_BUFFER_REQ	4.2.1	update buffer status of logical channel data in RLC. The update rate with which the buffer status is updated in the scheduler is outside of the scope of the document.
SCHED_DL_PAGING_BUFFER_REQ	4.2.2	update buffer status of paging messages.
SCHED_DL_MAC_BUFFER_REQ	4.2.3	update buffer status of MAC control elements. The update rate with which the buffer status is updated in the scheduler is outside of the scope of the document.
SCHED_DL_TRIGGER_REQ	4.2.4	Starts the DL MAC scheduler for this subframe
SCHED_DL_RACH_INFO_REQ	4.2.5	Provides RACH reception information to the scheduler
SCHED_DL_CQI_INFO_REQ	4.2.6	Provides CQI measurement report information to the scheduler
SCHED_DL_CONFIG_IND	4.2.5	triggers building of DL MAC PDUs and Subframe Configuration in MAC
SCHED_UL_TRIGGER_REQ	4.2.8	Starts the UL MAC scheduler for this subframe
SCHED_UL_NOISE_INTERFERENCE_REQ	4.2.9	Provides Noise and interference measurement information to the scheduler
SCHED_UL_SR_INFO_REQ	4.2.10	Provides scheduling request reception information to the scheduler
SCHED_UL_MAC_CTRL_INFO_REQ	4.2.11	Provides mac control information (power headroom, ul buffer status) to the scheduler
SCHED_UL_CQI_INFO_REQ	4.2.12	Provides UL CQI measurement information to the scheduler
SCHED_UL_CONFIG_IND	4.2.9	passes the UL scheduling decision (Format 0 DCIs) to MAC

Table 2 SCHED SAP

4.2.1 SCHED_DL_RLC_BUFFER_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
logicalChannelIdentity	0..10	uint8_t	The logical channel ID. see [1].
rlc_TransmissionQueueSize	0..4294967295	uint32_t	The current size of the transmission queue in byte
rlc_TransmissionQueueHOL_Delay	0..65535	uint16_t	Head of line delay of new transmissions in ms
rlc_RetransmissionQueueSize	0..4294967295	uint32_t	The current size of the retransmission queue in byte
rlc_RetransmissionHOL_Delay	0..65535	uint16_t	Head of line delay of retransmissions in ms
rlc_StatusPDU_Size	0..65535	uint16_t	The current size of the pending STATUS message in byte
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.2 SCHED_DL_PAGING_BUFFER_REQ Parameters

Elements	Value	Type	Description
nr_pagingInfoList	0..MAX_PAGING_LIST		Valid only if rnti==P-RNTI
pagingInfoList[nr_pagingInfoList]	A variable-length array of pagingInfoListElement		Valid only if rnti==P-RNTI
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.3 SCHED_DL_MAC_BUFFER_REQ Parameters

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE. TODO add paging depends on CCPU comment 15.
ceBitmap	TA, DRX, CR	uint8_t bitmap	The CE element which is scheduled to be sent by the MAC. Can be Timing Advance CE, DRX Command CE and Contention Resolution CE.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.4 SCHED_DL_TRIGGER_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF for which the scheduling is to be done.
nr_dlInfoList	0..MAX_DL_INFO_LIST	uint8_t	The number of elements in the UE DL information list
dlInfoList [nr_dlInfoList]	An variable-length array of dlInfoListElement. See 4.3.23	array of struct	The list of UE DL information.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.5 SCHED_DL_RACH_INFO_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
nr_rachList	0..MAX_RACH_LIST	uint8_t	The number of elements in the RACH list
rachList [nr_rachList]	A variable-length array of rachListElement. See 4.3.6	array of struct	The list of detected RACHs
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.6 SCHED_DL_CQI_INFO_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
nr_cqiList	0..MAX_CQI_LIST	uint8_t	The number of CQI reports.
cqiList[nr_cqiList]	A variable-length array of cqiListElement. See 4.3.24	array of struct	The list of DL CQI reports received in one subframe.

nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.7 SCHED_DL_CONFIG_IND Parameters

Elements	Value	Type	Description
nr_buildDataList	0.. MAX_BUILD_DATA_LIST	uint8_t	The number of elements in the buildData list
nr_buildRAR_List	0.. MAX_BUILD_RAR_LIST	uint8_t	The number of elements in the buildRAR list
nr_buildBroadcastList	0.. MAX_BUILD_BC_LIST	uint8_t	The number of elements in the buildBroadcastList. Applicable for BCCH, PCCH
buildDataList [nr_buildDataList]	A variable-length array of buildDataListElement. See 4.3.8	array of struct	The list of resource allocation for UEs and LCs.
buildRAR_List [nr_buildRAR_List]	A variable-length array of buildRAR_ListElement. See 4.3.10	array of struct	The list of resource allocation for RAR
buildBroadcastList [nr_buildBroadcastList]	A variable-length array of buildBroadcastListElement. See 4.3.11	array of struct	The list of resource allocation for BCCH, PCCH
nrOf_PDCCH_OFDM_Symbols	0..4	uint8_t	current size of PDCCH
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.8 SCHED_UL_TRIGGER_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF for which the scheduling is to be done.
nr_ulInfoList	0.. MAX_ULINFO_LIST	uint8_t	number of UL information elements
ulInfoList [nr_ulInfoList]	A variable-length array of ulInfoListElement. See 4.3.12	array of struct	The list of UL information for the scheduler.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.9 SCHED_UL_NOISE_INTERFERENCE_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
rip	-126.0..-75.0	uint16_t. fixed point SXXXXXXX.XXXXXXXX	Received Interference Power. See 36.214. in dBm.
tnp	-146.0..-75	uint16_t fixed point SXXXXXXX.XXXXXXXX	Thermal Noise Power. See 36.214. in dBm
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.10 SCHED_UL_SR_INFO_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
nr_srList	0..MAX_SR_LIST	uint8_t	The number of SR's received.
srList[nr_srList]	A variable-length array of srListElement. see 4.3.13	array of struct	The list of SRs received in one subframe.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next

			array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.11 SCHED_UL_MAC_CTRL_INFO_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
nr_macCE_List	0..MAX_MAC_CE_LIST	uint8_t	Number of MAC control elements
macCE_List[nr_macCE_List]	A variable-length array of maCE_ListElement. see 4.3.14	array of struct	The list of MAC control elements received in one subframe.
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.12 SCHED_UL_CQI_INFO_REQ Parameters

Elements	Value	Type	Description
sfn_sf	0..16377 bit 0-3 SF bit 4-13 SFN	uint16_t	The SFN and SF in which the information was received
ulCqi	see 4.3.29	Struct	
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.2.13 SCHED_UL_CONFIG_IND Parameters

Elements	Value	Type	Description
nr_dciList	0..MAX_DCI_LIST	uint8_t	number of UL DCIs created by the scheduler
nr_phichList	0..MAX_PHICH_LIST	uint8_t	number of PHICH information created by the scheduler
dciList [nr_dciList]	A variable-length array of ulDciListElement. See	array of struct	the list of UL DCI (Format 0) elements

	4.3.2		
phichList [nr_phichList.. MAX_PHICH_LIST]	A variable-length array of phichListElement. See 4.3.7	array of struct	the list of PHICH elements
nr_vendorSpecificList	0..MAX_SCHED_CFG_LIST	uint8_t	The number of elements in the next array
vendorSpecificList		A variable-length array of variable-length vendorSpecificListElement see 4.3.3	Contains scheduler specific configuration received from the OAM subsystem for use by a specific scheduler.

4.3 Parameters

4.3.1 dlDciListElement

Elements	**Value**	**Type**	**Description**
rnti	1..65535	uint16_t	The RNTI identifying the UE.
rb_bitmap	32bit bitmap	uint32_t	The RBs allocated to the UE
rb_shift	0,1	uint8_t	see [4] section 7.1.6.2
res_alloc	0,1,2	uint8_t	type of resource allocation
nr_of_tbs	1..2	uint8_t	The number of transport blocks
tbs_size [MAX_TB_LIST]	0..	uint16_t	The size of the transport blocks in byte
mcs[MAX_TB_LIST]	0..31	uint8_t	The modulation and coding scheme of each TB. see[4] section 7.1.7
ndi[MAX_TB_LIST]	0..1	uint8_t	New data Indicator.
rv[MAX_TB_LIST]	0..3	uint8_t	Redundancy version.
cce_index	0..88	uint8_t	CCE index used to send the DCI.
aggr_level	1,2,4,8	uint8_t	The aggregation level.
precoding_info	2 antenna_ports: 0..6 4 antenna_ports: 0..50	uint8_t	Precoding information.
format	1,1A,1B,1C,1D,2,2A,2B	enum	format of the DCI
tpc	-4,-1,0,1,3,4	int8_t	see [4] section 5.1.1.1
harq_process	0..7	uint8_t	HARQ process number
dai	1,2,3,4	uint8_t	only for TDD
vrb_format	VRB_DISTRIBUTED, VRB_LOCALIZED	enum	see [4] section 7.1.6.3
tb_swap	TRUE, FALSE	bool	TB to CW swap flag. see [3] section 5.3.3.1.5
sps_release	TRUE, FALSE		
pdcch_order	TRUE, FALSE	bool	Indicates if PDCCH is for PDCCH order.
preamble_index	0..63	uint8_t	Preamble index. Only valid if pdcch_order == TRUE.
prach_mask_index	0..15	uint8_t	PRACH Mask index. Only valid valid if pdcch_order == TRUE.
n_gap	GAP1, GAP2	enum	The value for N_GAP
tbs_idx	2,3	uint8_t	The TBS index for Format 1A
dl_power_offset	0,1	uint8_t	For Format 1D. see [4] section

			7.1.5
pdcch_power_offset	-6..4	int8_t	DL PDCCH power boosting in dB

4.3.2 ulDciListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
rb_start	0..99	uint8_t	The start RB allocated to the UE. see [4] section 8.1
rb_len	1..100	uint8_t	The number of RBs allocated to the UE. see [4] section 8.1
tb_size	2..	uint16_t	The size of the transport block in byte
mcs	0..32	uint8_t	The modulation and coding scheme of each TB. see[4] section 8.6
ndi	0..1	uint8_t	New data Indicator.
cce_index	0..88	uint8_t	CCE index used to send the DCI.
aggr_level	1,2,4,8	uint8_t	The aggregation level.
ue_tx_antenna_selection	0,1,3	uint8_t	see [3] section 5.3.3.2. 3 means antenna selection is off.
hopping	TRUE, FALSE	bool	Hopping enabled flag. see [4] section 8.4
n_2_dmrs	0..7	uint8_t	cyclic shift
tpc	-4,-1,0,1,3,4	int8_t	Tx power control command. see [4] section 5.1.1.1
cqi_request	TRUE, FALSE	bool	aperiodic CQI request flag. see [4] section 7.2.1
ul_index	0,1,2,3	uint8_t	UL index. only for TDD
dai	1,2,3,4	uint8_t	DL assignment index. only for TDD
freq_hopping	0..4	uint8_t	The frequency hopping bits. see [4] section 8.4
pdcch_power_offset	-6..4	int8_t	DL PDCCH power boosting in dB

4.3.3 vendorSpecificListElement

Elements	Value	Type	Description
type	PF_WEIGHT1, PF_WEIGHT2, CQI_AVG_FACTOR, etc.	enum	Indicating the type of the value. This types are examples, real types are implementation specific, examples are: PF_WEIGHT1 – The first weight used by a proportional fair scheduler PF_WEIGHT2 – The second weight used by a proportional fair scheduler CQI_AVG_FACTOR – The factor used for averaging CQIs in the scheduler.
length	unsigned integer	uint32_t	The length of the actual value
value	depends on type	struct	The actual value which will be set

4.3.4 logicalChannelConfigListElement

Elements	Value	Type	Description
logicalChannelIdentity	1..10	uint8_t	The logical channel id. See [1]. Note: CCCH is preconfigured.
logicalChannelGroup	0..3, 4	uint8_t	The LC group the LC is mapped to. . 4 means no LCG is associated with the logical channel.
direction	DIR_UL, DIR_DL, DIR_BOTH	enum	The direction of the logical channel.
qosBearerType	QBT_NON_GBR, QBT_GBR	enum	Guaranteed or non-guaranteed bit rate bearer
qci	0..255	uint8_t	The QCI defined in 3GPP TS 23.203. The QCI is coded as defined in 36.413, i.e the value indicates one less than the actual QCI value.
e_RAB_MaximumBitrateUL	0..10000000000	uint64_t	in bit/s. For QBT_GBR only.
e_RAB_MaximumBitrateDL	0..10000000000	uint64_t	in bit/s. For QBT_GBR only.
e_RAB_GuaranteedBitrateUL	0..10000000000	uint64_t	in bit/s. For QBT_GBR only.
e_RAB_GuaranteedBitrateDL	0..10000000000	uint64_t	in bit/s. For QBT_GBR only.

4.3.5 addBufferListElement

Elements	Value	Type	Description
length	1..65535	uint16_t	lengths in bytes
timestamp	32bit	uint32_t	Time packet arrived in PDCP

4.3.6 rachListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The newly allocated t-c-rntis
estimatedSize	56,144,208,256	uint16_t	estimated minimum size of first UL message in bits, based on received RACH preamble

4.3.7 phichListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
phich	ACK, NACK	enum	ACK or NACK to be passed to the UE in the PHICH

4.3.8 buildDataListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
dci	see section 4.3.1	struct	The DL DCI configured for this UE. This may also indicate PDCCH order or SPS release or format 3/3A, in which case there is no associated PDSCH.
ceBitmap[MAX_TB_LIST]	TA, DRX, CE	uint8_t bitmap	The CEs scheduled for transmission for this TB
nr_rlcPDU_List	1..MAX_RLC_PDU_LIST	uint8_t	The number of RLC PDUs to be built
rlcPDU_List [nr_rlcPDU_List][MAX_TB_LIST]	A variable array of rlcPDU_ListElement. See 4.3.9	array of struct	List of parameters for RLC PDU creation.

4.3.9 rlcPDU_ListElement

Elements	Value	Type	Description
logicalChannelIdentity	0..10	uint8_t	The logical channel ID. see [1]
size	1..9420	uint16_t	Maximum length of RLC PDU in bytes

4.3.10 buildRAR_ListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE (in this case it is the Temporary C-RNTI).
grant	integer	uint32_t	20 bit UL grant. see [4] section 6.2
dci	see 4.3.1	struct	The DL DCI configured for this RAR.

4.3.11 buildBroadcastListElement

Elements	Value	Type	Description
type	BCCH, PCCH	enum	The type identifying the broadcast message.
index	0..63	uint8_t	The index of the broadcast message. This identifies which broadcast message (either SIB1, SIx or PCCH) should be transmitted. 0 – SIB1 1..31 – SIx 32..63 - PCCH
dci	see 4.3.1	struct	The DL DCI configured for BCCH and PCCH.

4.3.12 ulInfoListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
ulReception[MAX_LC_LIST+1]	0..65535	uint16_t	The amount of data in bytes in the MAC SDU received in this subframe for the given logical channel.
receptionStatus	Ok, notOk, NotValid	enum	"NotValid" is used when no TB is expected. "Ok/notOk" Indicates successful/unsuccessful reception of UL TB.
tpc	-4,-1,0,1,3,4	int8_t	Tx power control command. see [4] section 5.1.1.1

4.3.13 srListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.

4.3.14 macCE_ListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
maCE_Type	BSR, PHR, CRNTI	enum	Mac Control Element Type
maCE_Value	see 4.3.15	union	Field discrimination is based on maCE_Type

4.3.15 macCE_Value

Elements	Value	Type	Description
phr	0..63,64	uint8_t	The power headroom. see [1] section 6.1.3.6. 64 means no valid PHR is available.
crnti		uint8_t	Indicates that a C-RNTI MAC CE was received. The value is not used
bufferStatus [MAX_NR_LCG]	0..63, 64	uint8_t	The value 64 indicates that the buffer status for this LCG should not to be updated. Always all 4 LCGs are present. see [1] 6.1.3.1

4.3.16 drxConfig

Elements	Value	Type	Description
onDurationTimer	1,2,3,4,5,6,8,10,20,30,40,50,60,80,100,200	uint8_t	Timer in subframes. see [1]
drx_InactivityTimer	1,2,3,4,5,6,8,10,20,30,40,50,60,80,1	uint16_t	Timer in subframes.

	00,200,300,500,750,1280,1920,2560		see [1]
drx_RetransmissionTimer	1,2,4,6,8,16,24,33	uint8_t	Timer in subframes. see [1]
longDRX_Cycle	10,20,32,40,64,80,128,160,256,320,512,640,1024,1280,2048,2560	uint16_t	Long DRX cycle in subframes. see [1]
longDRX_CycleStartOffset	0..2559	uint16_t	Long DRX cycle offset. see [1]
shortDRX_Cycle	2,5,8,10,16,10,21,40,64,80,128,160,256,320,512,640,OFF	uint16_t	Short DRX cycle in subframes. see [1]
drxShortCycleTimer	1..16	uint8_t	Timer in subframes. see [1]

4.3.17 spsConfig

Elements	Value	Type	Description
semiPersistSchedIntervalUL	10,20,32,40,64,80,128,160,320,640	uint16_t	SPS scheduling interval in UL in subframes.
semiPersistSchedIntervalDL	10,20,32,40,64,80,128,160,320,640	uint16_t	SPS scheduling interval in DL in subframes.
numberOfConfSPS_Processes	1..8	uint8_t	number of SPS HARQ processes. see [1]
n1_PUCCH_AN_PersistentListSize	0..4	uint8_t	The size of the list. When spsConfig is included in CSCHED_UE_CONFIG_IND this parameters is ignored
n1_PUCCH_AN_PersistentList[4]	0..2047	uint16_t	see [4] section 10.1. When spsConfig is included in CSCHED_UE_CONFIG_IND this parameters is ignored
implicitReleaseAfter	2,3,4,8	uint8_t	number of empty transmission. see [1] section 5.10.2. When spsConfig is included in CSCHED_UE_CONFIG_IND this parameters is ignored

4.3.18 srRequest

Elements	Value	Type	Description
action	setup,release	enum	Indicates if SR config should be released or changed
schedInterval	5,10,20,40,80	uint8_t	SR scheduling interval in subframes.
dsr_TransMax	4,8,16,32,64	uint8_t	see [1] section 5.4.4

4.3.19 cqiRequest

Elements	Value	Type	Description

action	setup,release	enum	Indicates if SR config should be released or changed
cqiSchedInterval	1,2,5,10,20,32,40,64,80,128,160	uint16_t	CQI scheduling interval in subframes.
riSchedInterval	1,2,4,8,16,20	uint8_t	RI scheduling interval in subframes.

4.3.20 ue_Capabilities

Elements	Value	Type	Description
halfDuplex	FALSE, TRUE	bool	UE only supports half-duplex FDD operation
intraSF_hopping	FALSE, TRUE	bool	UE support of intra-subframe hopping
type2_sb_1	FALSE, TRUE	bool	UE supports type 2 hopping with n_sb > 1
ueCategory	1..5	uint8_t	The UE category
resAllocType1	FALSE, TRUE	bool	UE support for resource allocation type 1.

4.3.21 siConfiguration

Elements	Value	Type	Description
sfn	0..1023	uint16_t	Frame number to apply this configuration.
sib1Length	1..MAX_SI_MSG_SIZE	uint16_t	The length of the SIB 1 message. Unit in bytes
siWindowLength	1,2,5,10,15,20,40	uint8_t	Common SI scheduling window for all SIs. Unit in subframes.
nrSI_Message_List	0..MAX_SI_MSG_LIST	uint8_t	The number of SI messages.
siMessageList [nrSI_Message_List]	variable-length array of siMessageListElement. see 4.3.22	array of struct	List of SI messages to be sent. The index will later be used to identify the message in the buildBroadcastListElement see 4.3.11

4.3.22 siMessageListElement

Elements	Value	Type	Description
periodicity	8, 16, 32, 64, 128, 256, 512	uint16_t	Periodicity of the SI-message Unit in radio frames
length	1..MAX_SI_MSG_SIZE	uint16_t	The length of SI message. Unit in bytes

4.3.23 dlInfoListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
harqProcessId	0..7,8	uint8_t	HARQ process ID. 8 is not present

nr_harqStatus	1.. MAX_TB_LIST	uint8_t	The size if the HARQ status list.
harqStatus[MAX_TB_LIST]	ACK, NACK, DTX	enum	HARQ status for the above process

4.3.24 cqiListElement

Elements	Value	Type	Description
rnti	1..65535	uint16_t	The RNTI identifying the UE.
ri	1..4	uint8_t	The last received rank indication.
cqiType	P10,P11,P20,P21, A12, A22, A20, A30, A31	enum	The CQI format.
wb_cqi[MAX_TB_LIST]	0..15	array of uint8_t	The reported wideband CQI value per codeword.
wb_pmi	0..15	uint8_t	The reported wideband precoding matrix index.
sbMeasResult	see 4.3.25	union	field discrimination is based on cqiType.

4.3.25 sbMeasResult

Elements	Value	Type	Description
ueSelected	see 4.3.26	struct	The subband measurement results for aperiodic UE selected subbands.
higherLayerSelected[MAX_HL_SB]	see 4.3.27	struct	The subband measurement results for aperiodic higher-layer selected subbands.
bwPart	see 4.3.28	struct	The subband measurement results for periodic UE selected subbands. Indicates the current CQI for the best subband in the current bandwidth part.

4.3.26 ueSelected

Elements	Value	Type	Description
sbList[MAX_UE_SELECTED_SB]	0..24	array of uint8_t	The preferred-M subbands reported by the UE.
sb_pmi	0..15	uint8_t	The PMI conditioned on the preferred - M subbands.
sb_cqi[MAX_TB_LIST]	0..15	uint8_t	The CQI for up to two codewords conditioned on the preferred -M subbands.

4.3.27 higherLayerSelected

Elements	Value	Type	Description
sb_pmi	0..63	uint8_t	The PMI conditioned on the current

			subband.
sb_cqi[MAX_TB_LIST]	0..15	uint8_t	The CQI for up to two codewords conditioned on the current subband.

4.3.28 bwPart

Elements	Value	Type	Description
bwPartIndex	1..4	uint8_t	The bandwidth part for which CQI is being reported.
sb	0..11	uint8_t	The preferred subband in the current bandwidth part.
cqi	0..15	uint8_t	The CQI conditioned on the selected subband

4.3.29 ulCQI

Elements	Value	Type	Description
sinr[MAX_SINR_RB_LIST	-20.0..30.	fixed point u_int16_t Sxxxxxxxxxxx.xxx	The SINR measurement based on the resource given in type. In case of PUCCH only the first index is used. For PRACH the first 6 indices are used. For PUSCH and SRS each index represents one RB. The SINR is given in dB
type	SRS, PUSCH, PUCCH_1, PUCCH_2, PRACH	enum	SRS – measurement on SRS PUSCH – measurement on PUSCH PUCCH_1 – measurement done on Format 1 resource PUCCH_2 – measurement done on Format 2 resource PRACH – measurement done on PRACH

4.3.30 pagingInfoListElement

pagingIndex	32..63	uint8_t	The index used to identify the scheduled message, will be retuned in SCHED_DL_CONFIG_IND
pagingMessageSize		uint16_t	The size of the paging message
pagingSubframe	0..9	uint8_t	The subframe during which the message shall be sent

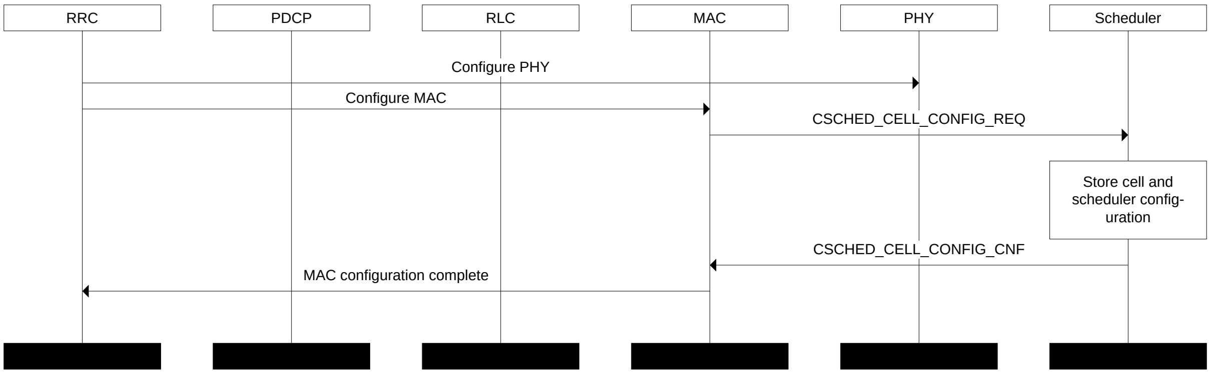
4.4 Constants

Elements	Value
MAX_SCHED_CFG_LIST	10
MAX_LC_LIST	10

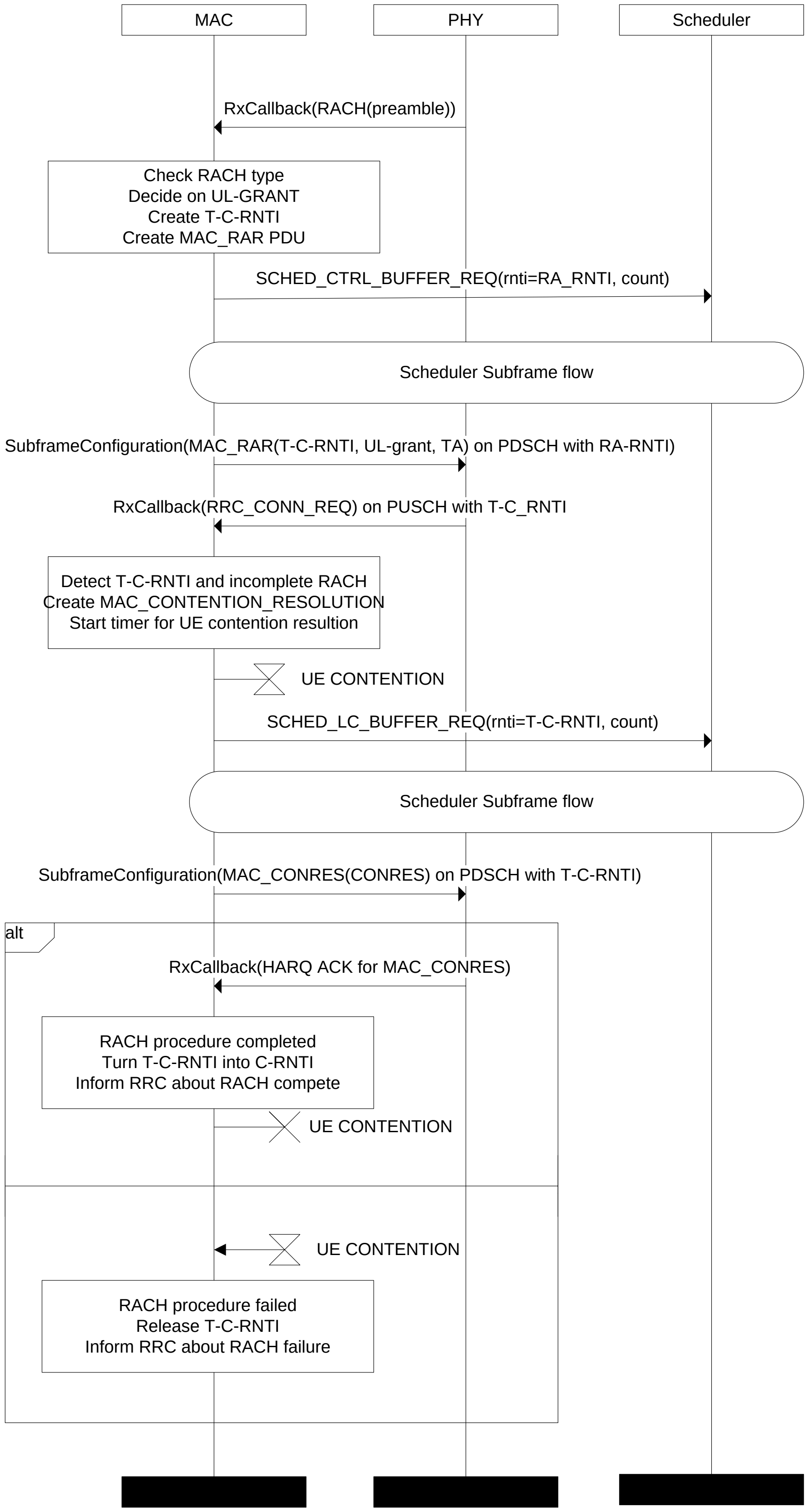
MAX_CTRL_BUF_LIST	4
MAX_BUFF_ADD_LIST	30
MAX_RACH_LIST	30
MAX_DL_INFO_LIST	30
MAX_BUILD_DATA_LIST	30
MAX_BUILD_RAR_LIST	10
MAX_BUILD_BC_LIST	3
MAX_ULINFO_LIST	30
MAX_DCI_LIST	30
MAX_PHICH_LIST	30
MAX_TB_LIST	2
MAX_RLC_PDU_LIST	30
MAX_NR_LCG	4
MAX_MBSFN_CONFIG	5
MAX_SI_MSG_LIST	32
MAX_SI_MSG_SIZE	65535
MAX_CE_LIST	30
MAX_CQI_LIST	30
MAX_UE_SELECTED_SB	6
MAX_HL_SB	25
MAX_SINR_RB_LIST	100
MAX_SR_LIST	30
MAX_MAC_CE_LIST	30

5 Scenarios

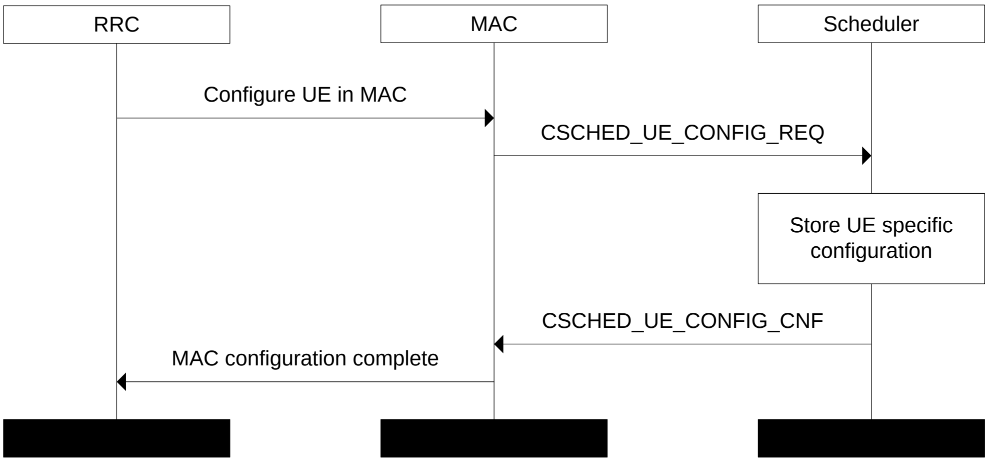
5.1 Cell Setup


RRC
PDCP
RLC
MAC
PHY
Scheduler
Configure PHY
Configure MAC
CSCHED_CELL_CONFIG_REQ
Store cell and scheduler config-uration
CSCHED_CELL_CONFIG_CNF
MAC configuration complete

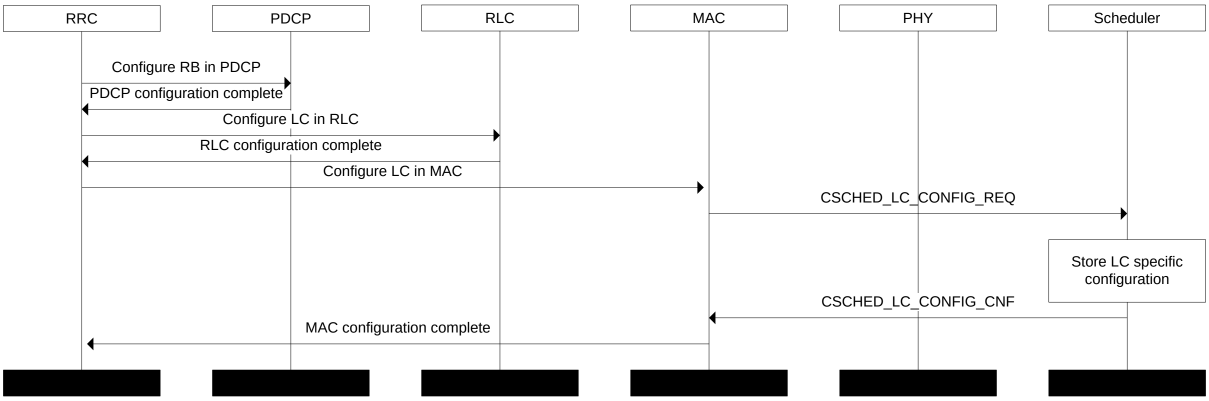
5.2 RACH procedure



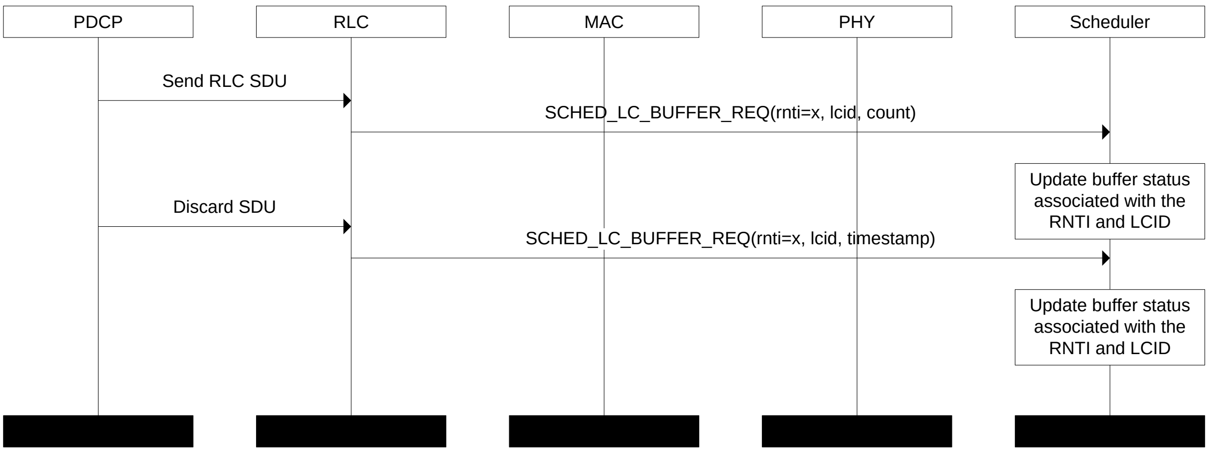
5.3 UE configuration



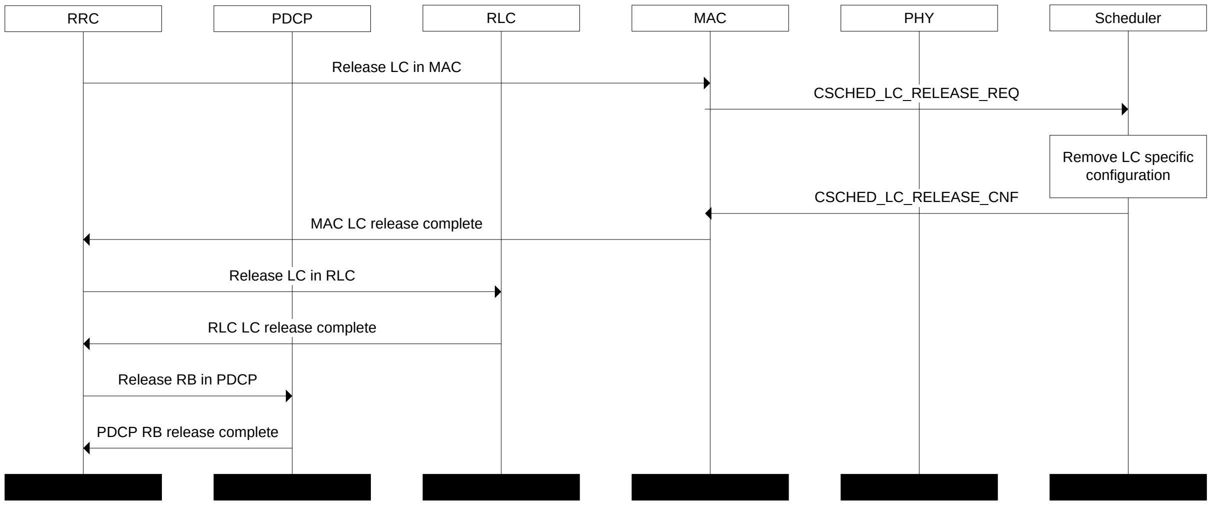
5.4 Radio Bearer Setup



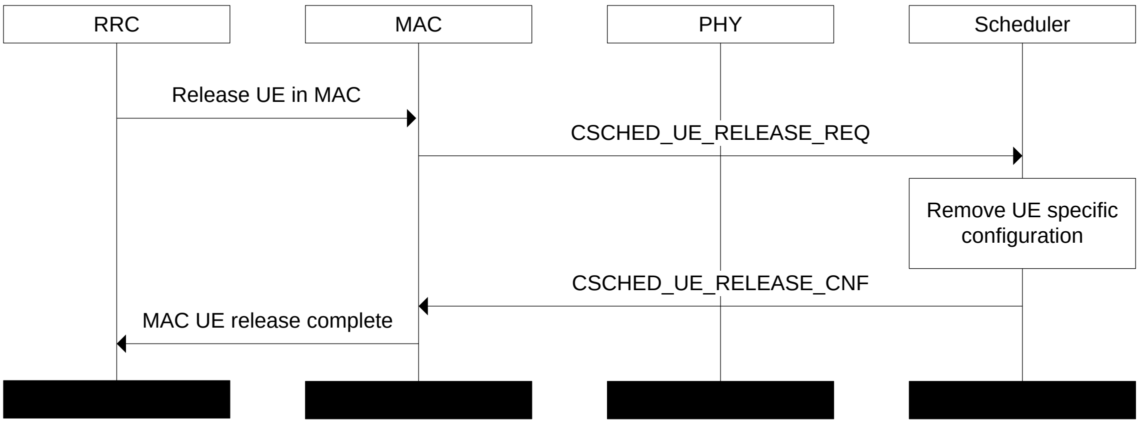
5.5 Handling of logical channel buffer status



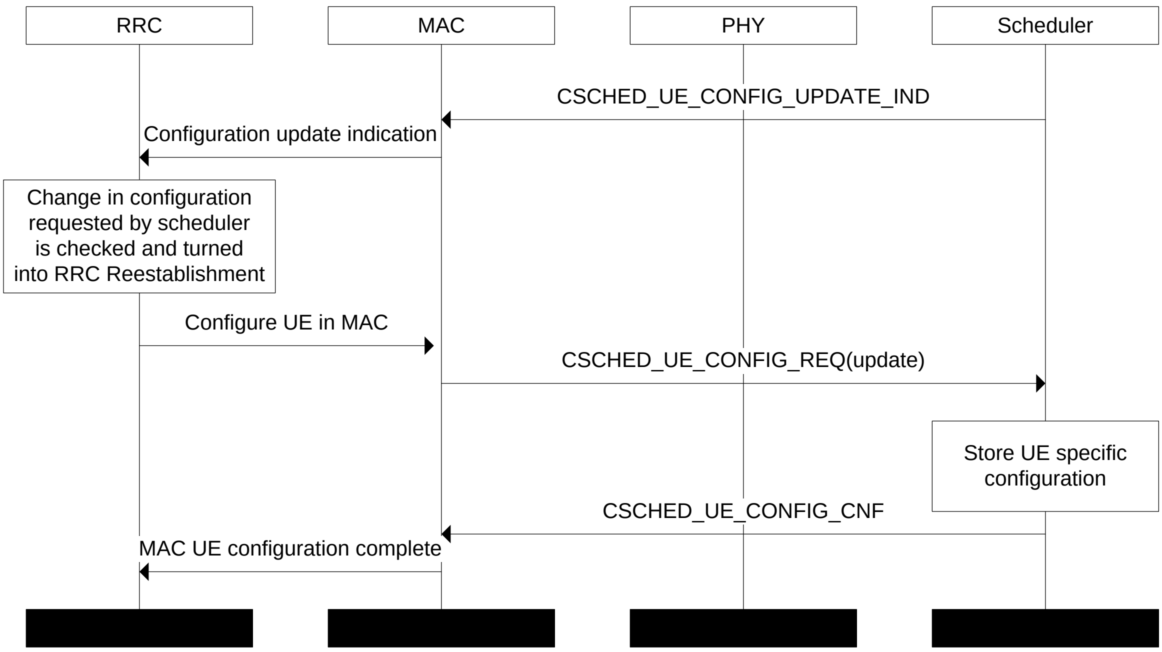
5.6 DRB release



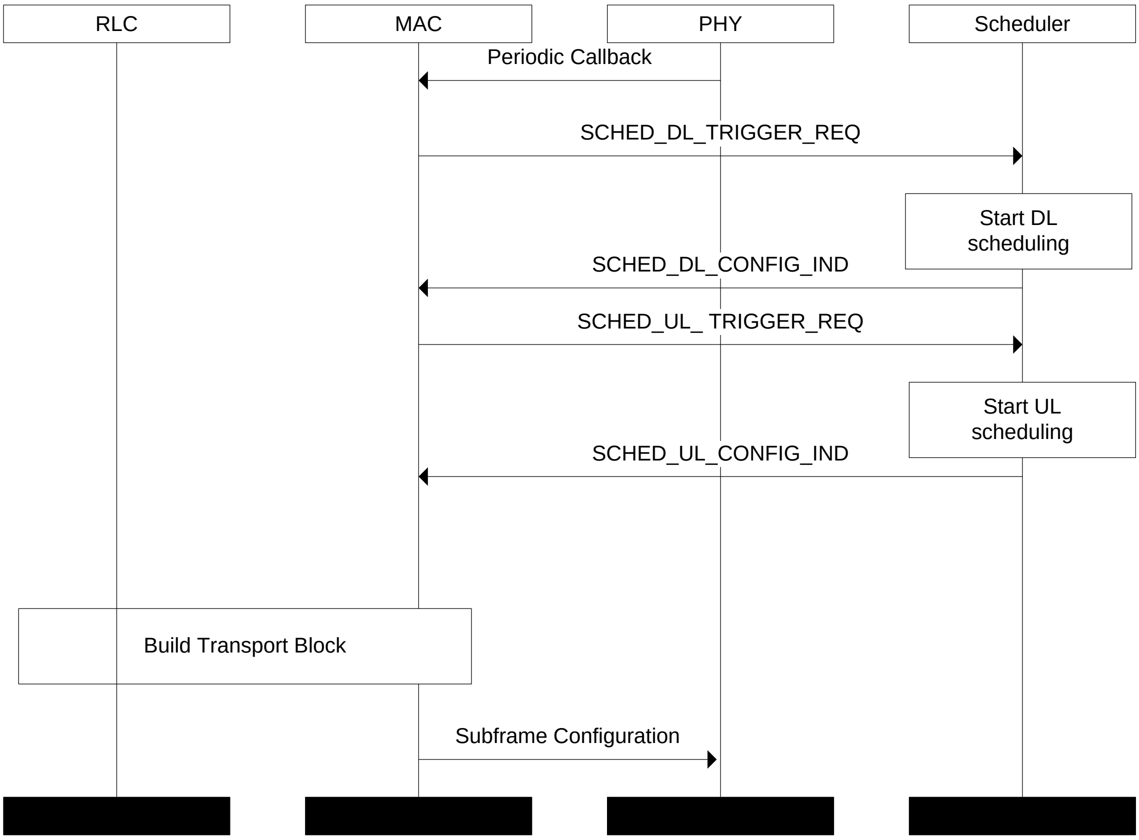
5.7 UE release



5.8 UE configuration update by MAC scheduler



5.9 Scheduler Subframe flow



6 References

[1] 3GPP TS 36.321: "Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification (Release 8)", v8.5.0, March 200.

[2] 3GPP TS 36.211: "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation (Release 8)", v8.6.0, March 2009.

[3] 3GPP TS 36.212: "Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding (Release 8)", v8.6.0, March 2009.

[4] 3GPP TS 36.213: "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures (Release 8)", v8.6.0, March 2009.

[5] 3GPP TS 36.331: "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol Specification (Release 8)", v8.5.0, March 2009

7 Revision History

Version	**Description**
1.11	Prepared for public release

8 Appendix A: Performance and Functional Requirements for the LTE Femtocell Scheduler API

This appendix provides a high level overview of performance and functionality requirements for LTE schedulers that utilize the LTE femtocell scheduler API framework being defined by Femto Forum WG2. These requirements are not totally comprehensive but represent a set of basic requirements that would be reasonably expected by an operator from an LTE scheduler residing in an LTE home eNodeB.

- Satisfy latency and packet error loss characteristics of each QCI class standardized in 3GPP 23.203 Table 6.1.7 under the following conditions:
    - Single user case: one user accesses any one of the example services in below table via a home eNodeB.
    - Multiple user/services case: one or several users simultaneously access more than one of the example services in below table via a home eNodeB.
- Satisfy Guaranteed Bit Rate (GBR), Minimum Bit Rate (MBR, as applicable, for each service data flow managed by the scheduler under the following conditions:
    - Single user case: one user accesses any one of the example services in below table via a home eNodeB.
    - Multiple user/services case: one or several users simultaneously access more than one of the example services in below table via a home eNodeB.
- Enforce downlink maximum bit rate for sum of downlink bearers based on UE-AMBR and APN-AMBR (for non-GBR flows). Enforce corresponding uplink maximum bit rates.
- Interact with admission and load control mechanisms to ensure that new users are admitted only when QoS requirements of existing and newly added users/bearers can be met.
- When system load exceeds certain pre-defined thresholds, judiciously select lowest priority bearers for service downgrade.
- Dynamically perform frequency selective and frequency diverse scheduling (localized and distributed virtual resource blocks) depending upon channel conditions, QoS requirements, etc.
- Dynamically adapt transport block size selection, MIMO mode selection, and rank depending upon Channel Quality Indicator (CQI), Pre-coding Matrix Indicator (PMI), and Rank Indication (RI) feedback from UEs while taking into account the status of data buffers.
- Provide higher priority to HARQ re-transmissions versus new transmissions for a bearer.
- Monitor current packet allocations and overall system load.