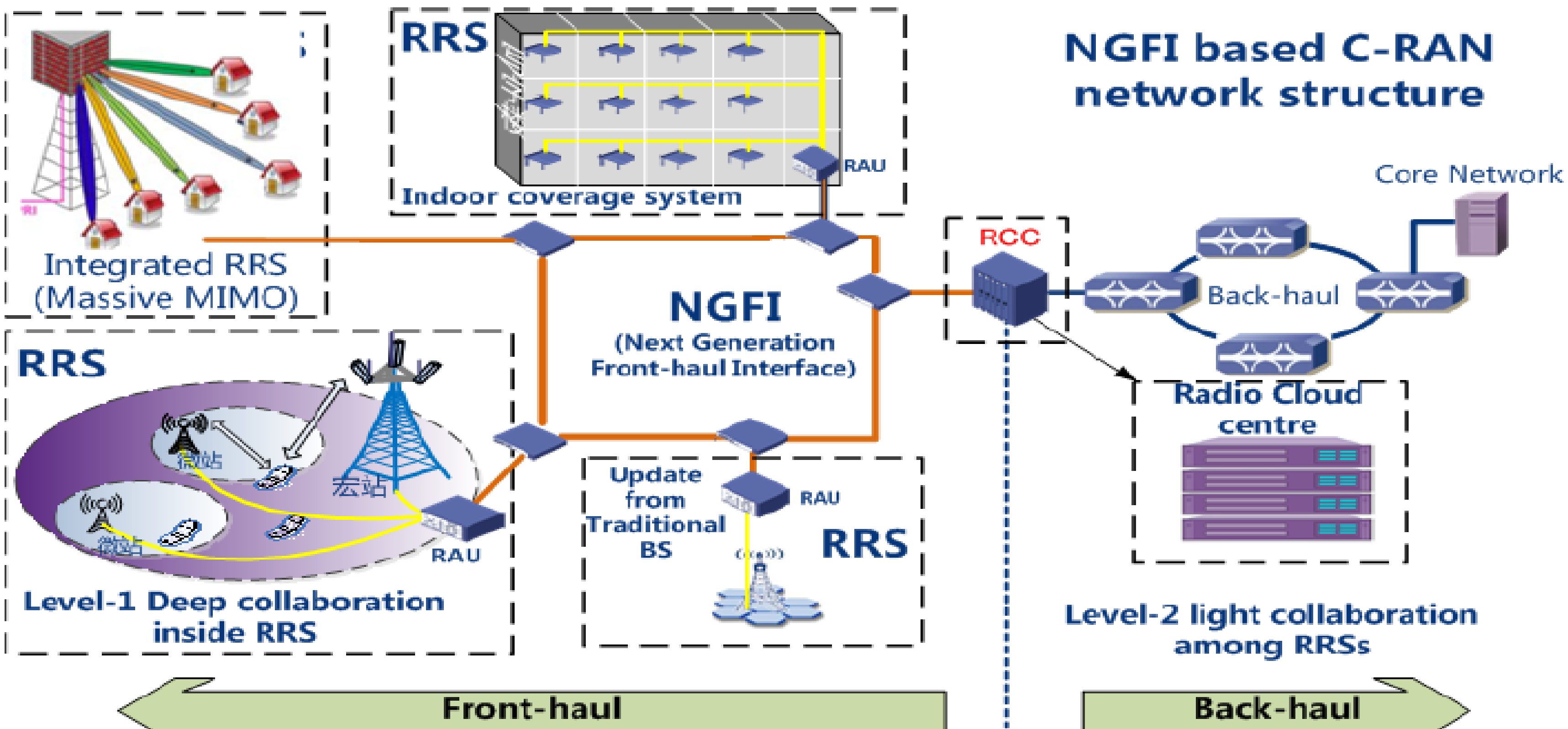
Some Preliminaries

- This description corresponds to the `RU-RAU-split` branch for RAN infrastructure components (located in `targets/DOCS/oai_L1_L2_procedures.pdf`, editable with `ipe`)
- we describe
  1. node functions
  2. current functional splits and packet formats
  3. RAN procedures
  4. process scheduling

NGFI Harmonization in OAI

- New descriptions for OAI RAN infrastructure Node Functions
  - `NGFI_RCC` : Radio Cloud Center
  - `NGFI_RAU` : Radio Aggregation Unit
  - `NGFI_RRU` : Remote Radio Unit
  - `3GPP_BBU` : Baseband Unit
  - `3GPP_eNodeB` : Complete eNodeB



Source: China Mobile - `http://www.windriver.com.cn/windforum/download/wf2015_networking03.pdf`

NGFI split points

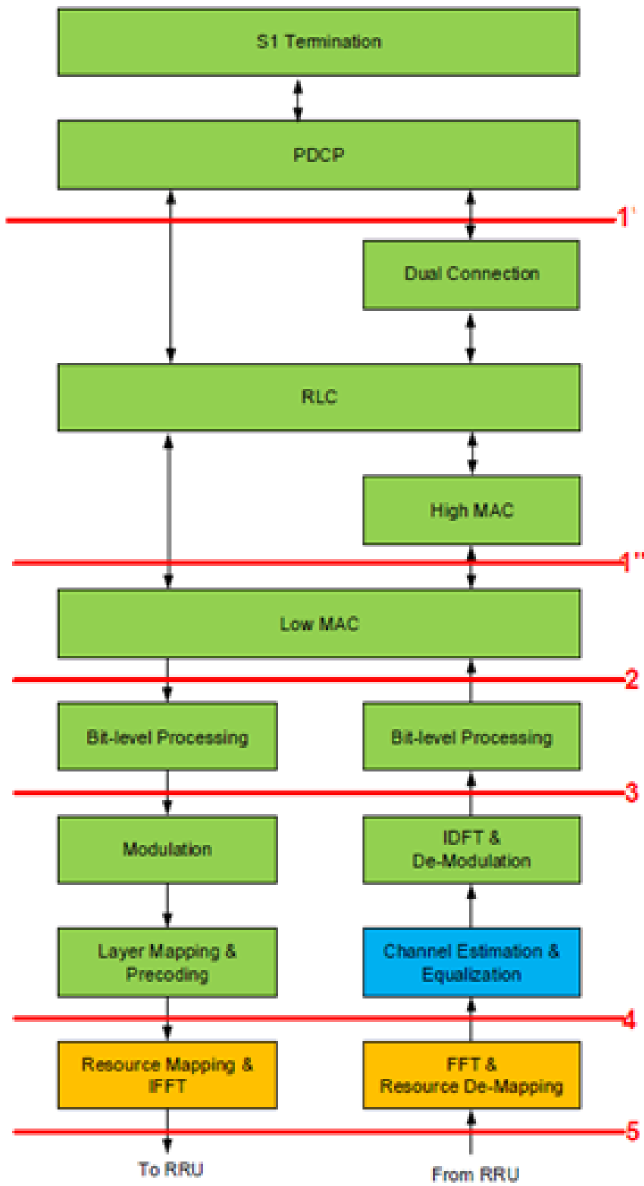
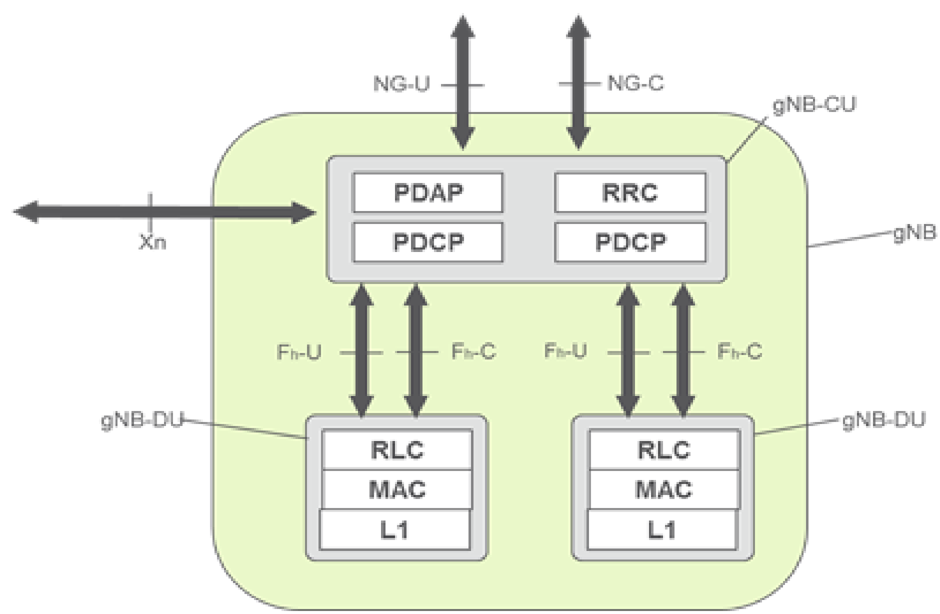
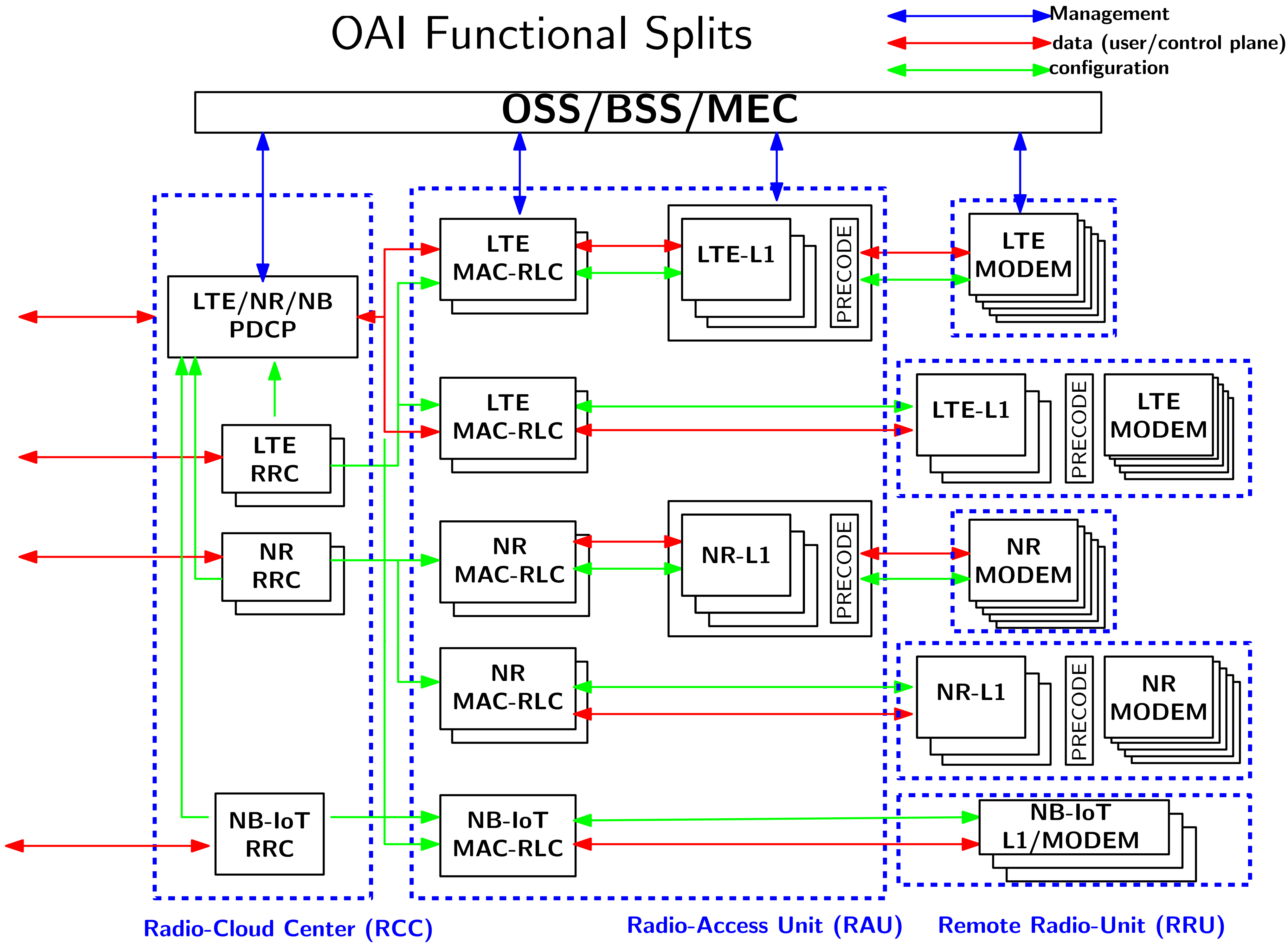


Figure 3-1: Division Plans for the RCC-RRS Interface

- Current OAI implementation (RRU/RCC) supports either
  - IF5 time-domain fronthaul (> 1 GbE required)
  - IF4.5 split (FFTs) (280 Mbit/s/antenna port fronthaul 20 MHz carrier) per carrier/sector
  - Soon IF2 (NFAPI)
  - IF1 for PDCP/RRC soon (3GPP Fh-C/Fh-U)

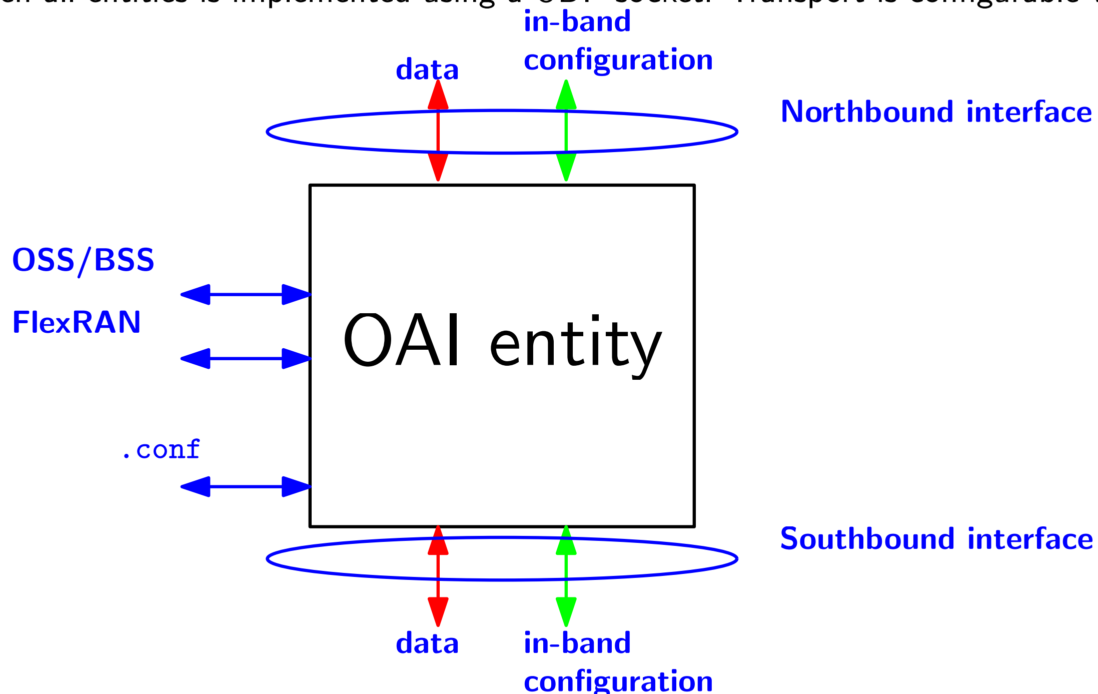


OAI RAN Software Architecture

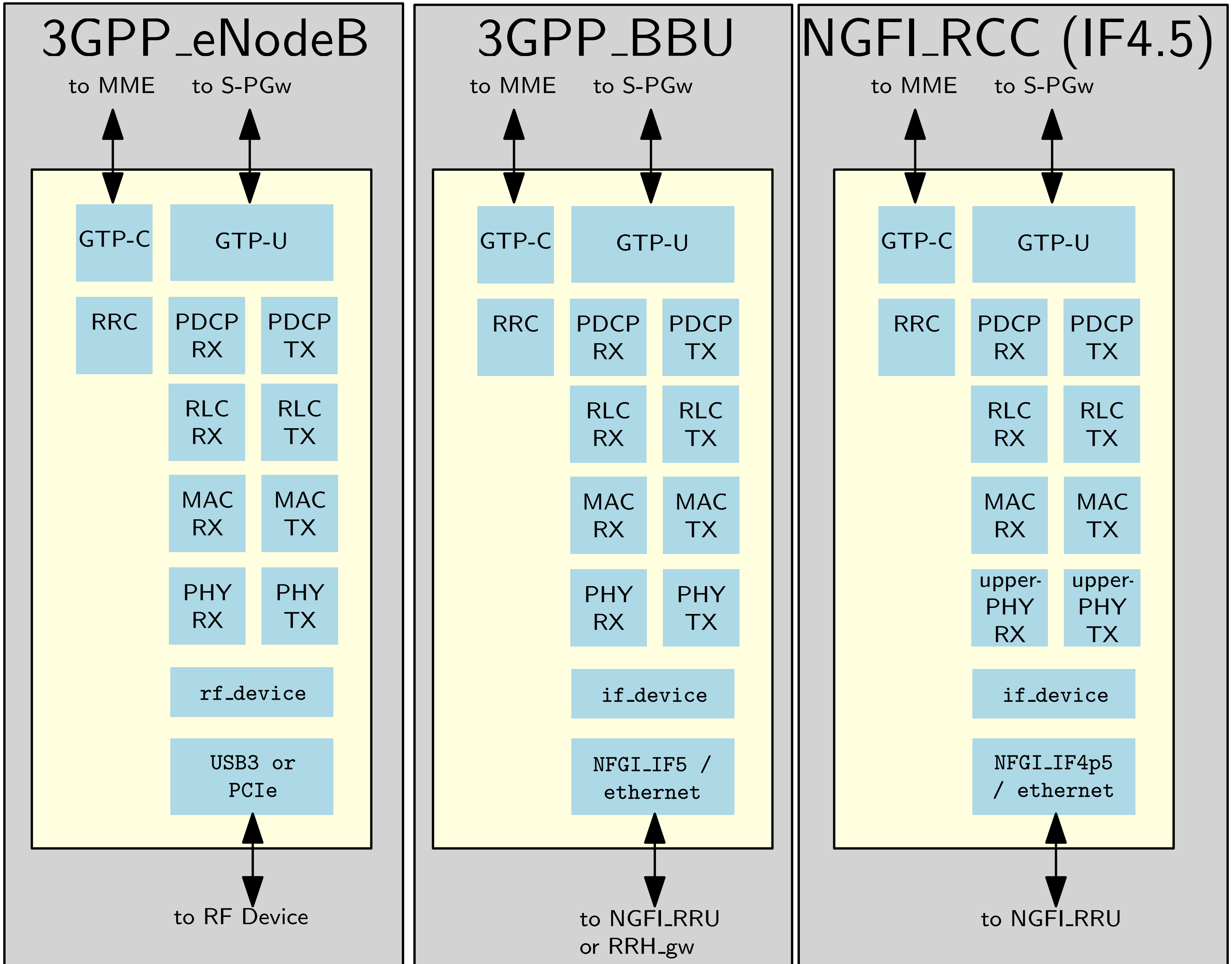

OAI Functional Splits
Management
data (user/control plane)
configuration
OSS/BSS/MEC
LTE/NR/NB PDCP
LTE RRC
NR RRC
NB-IoT RRC
LTE MAC-RLC
LTE MAC-RLC
NR MAC-RLC
NR MAC-RLC
NB-IoT MAC-RLC
LTE-L1
PRECODE
NR-L1
PRECODE
LTE MODEM
LTE-L1
PRECODE
LTE MODEM
NR MODEM
NR-L1
PRECODE
NR MODEM
NB-IoT L1/MODEM
Radio-Cloud Center (RCC)
Radio-Access Unit (RAU)
Remote Radio-Unit (RRU)

OAI Functional Splits

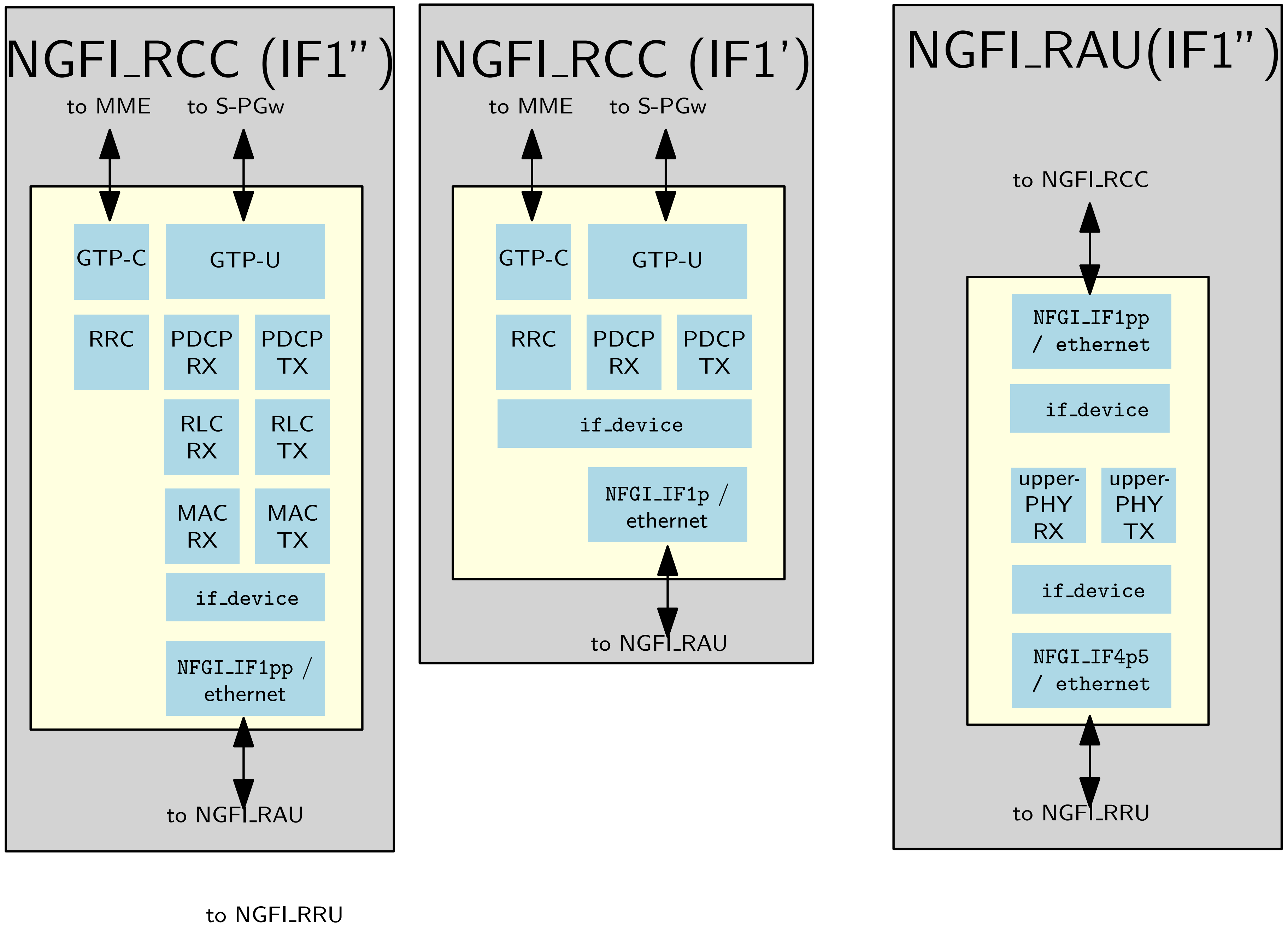
- OAI currently implements the following entities in openairinterface5g
  - LTE-MODEM (eNB 36.211 OFDM modulation/demodulation)
  - LTE-L1 (eNB 36.211/212/213)
  - LTE-MACRLC (eNB 36.321/322)
  - LTE-PDCP (eNB PDCP/GTPU 36.323)
  - LTE-RRC (eNB RRC/SCTP 36.331)
- Each entity comprises
  - a northbound interface (backhaul/midhaul/fronthaul and configuration)
  - a southbound interface (midaul/fronthaul and configuration)
  - one or two management interfaces
  - Three computing nodes
    * **Radio Cloud Center (RCC)** : multiple RRC/PDCP entities
    * **Radio-Access Unit (RAU)**: multiple MACRLC entities with medium-latency midhaul and L1 entities with low-latency fronthaul.
    * **Remote Radio-Unit (RRU)**: Equipment at radio site. Varying degrees of processing elements depending on fronthaul/midhaul interface.
- Each entity has a configuration which is a local file or received via the management interface
- default interface between all entities is implemented using a UDP socket. Transport is configurable via a dynamically-loadable networking device



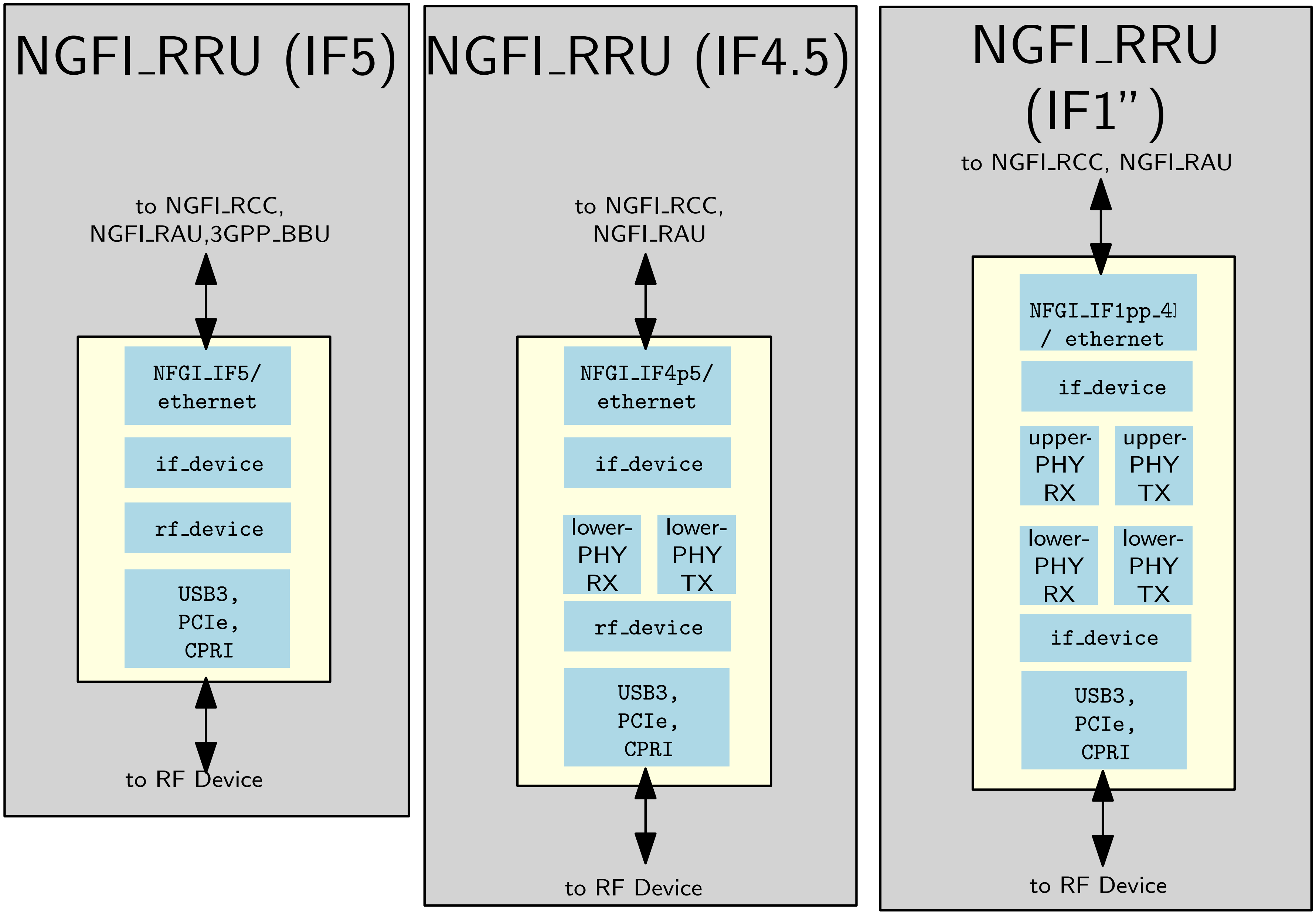
Functional Splits (Current)



Functional Splits (Current)



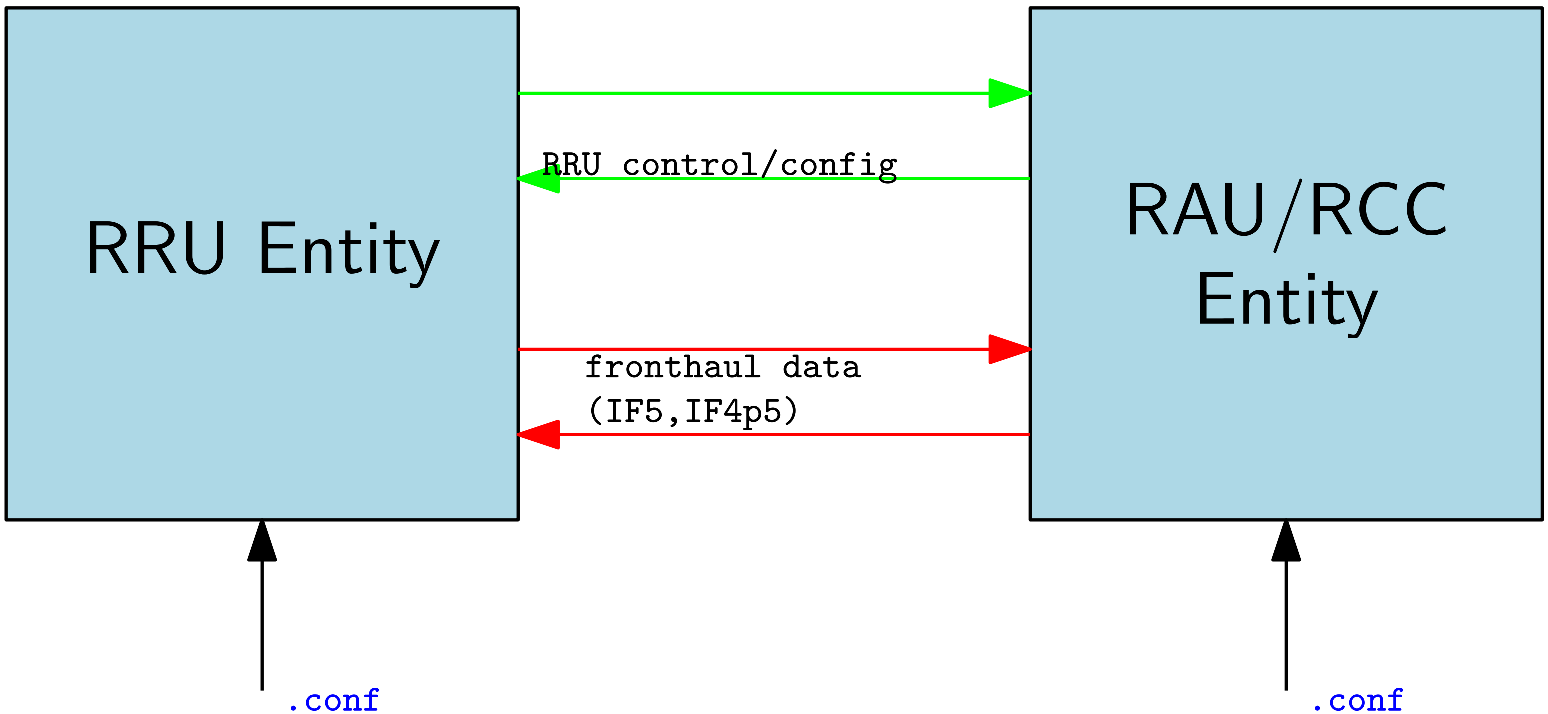
Functional Splits (Current)



Some Notes on usage of splits

- IF4p5 corresponds to the split-point at the input (TX) and output (RX) of the OFDM symbol generator (i.e. frequency-domain signals). According to NGFI, IF4 is "Resource mapping and IFFT" and "FFT and Resource demapping". We currently do not try to exploit multiplexing gains for unused spectral components. So, IF4p5 is simply compressed transmitted or received resource elements in the usable channel band.
- The simplest deployment for DAS (indoor) is one NGFI_RCC (IF4p5) and many NGFI_RRU (IF4p5). Spatio-temportal filtering (Precoding, later) is done in RCC and RRU perform IFFT/FFT and signal generation/acquisition. Fronthaul rates in this case are feasible with 1GbE copper links. This allows for PoE in addition to fronthaul data.
- More complex indoor, for instance with RCC in a common data center with outdoor RRS, could be
    1. RCC-RAU with IF1", RAU-RRU with IF4p5. Spatio-temporal filtering is done in frequency-domain in RAU along with full TX and RX processing (L1/L2) for the indoor RRS. Note that IF1' fronthaul on TX to RRU would be difficult because spatio-temporal filtering should be used. RRU does only IFFT/FFT and signal generation/acquisition
    2. RCC-RAU with IF1', RAU-RRU with IF4p5. Here RCC does L2, RAU does L1 and precoding for RRS.
- A massive-MIMO solution would consist either of
    1. an embedded RAU with processing (Spatio-temporal in frequency-domain, lower/upper PHY TX/RX) like the IF1' DAS solution above
    2. or more simply a high-speed fronthaul (IF4p5) with an RAU for multiple sites
    3. directly connected to RCC via high-speed IF4p5 (several virtual cells, precoder and IFFT/FFT in array).
- RCC solution with IF1" would cater to evolved-PDCP for heterogenity (4G,5G,WIFI,IoT)
- Currently supported node functionalities
    1. `3GPP_eNodeB`
    2. `3GPP_eNodeB_BBU` [NGFI_IF5]
    3. `NGFI_RCC` [NGFI_IF4p5]
    4. `NGFI_RRU` [NGFI_IF5]
    5. `NGFI_RRU` [NGFI_IF4p5]

RU - L1 negotiation



Control and Configuration protocol

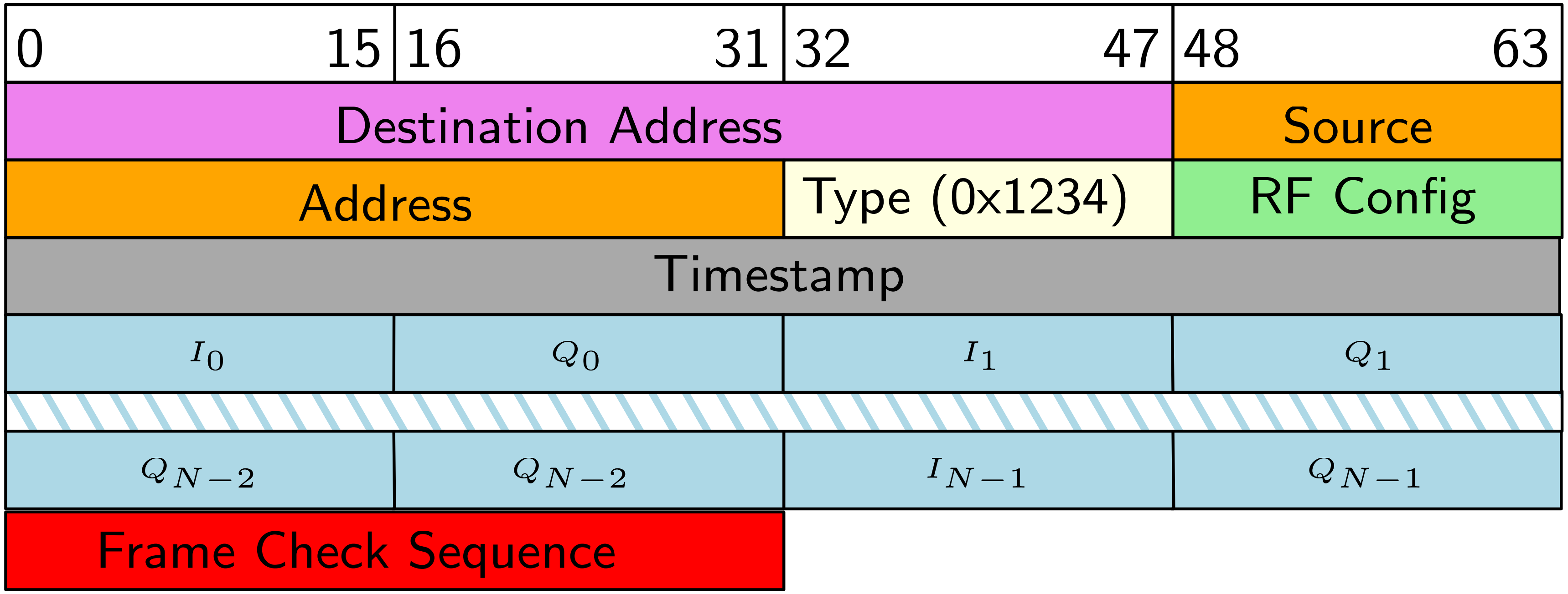
1. RAU→ RRU : heartbeat (capabilities request)
2. RRU→ RAU: capabilities indication
3. RAU→ RRU: configuration (band, dl_Carrier,ul_Carrier, dl_RS_EPRE, rx/tx attenuation)

During steady state, the control port manages the link quality and status (packet losses, synchronization state, start/stop, etc.)

RU - L1 data plane

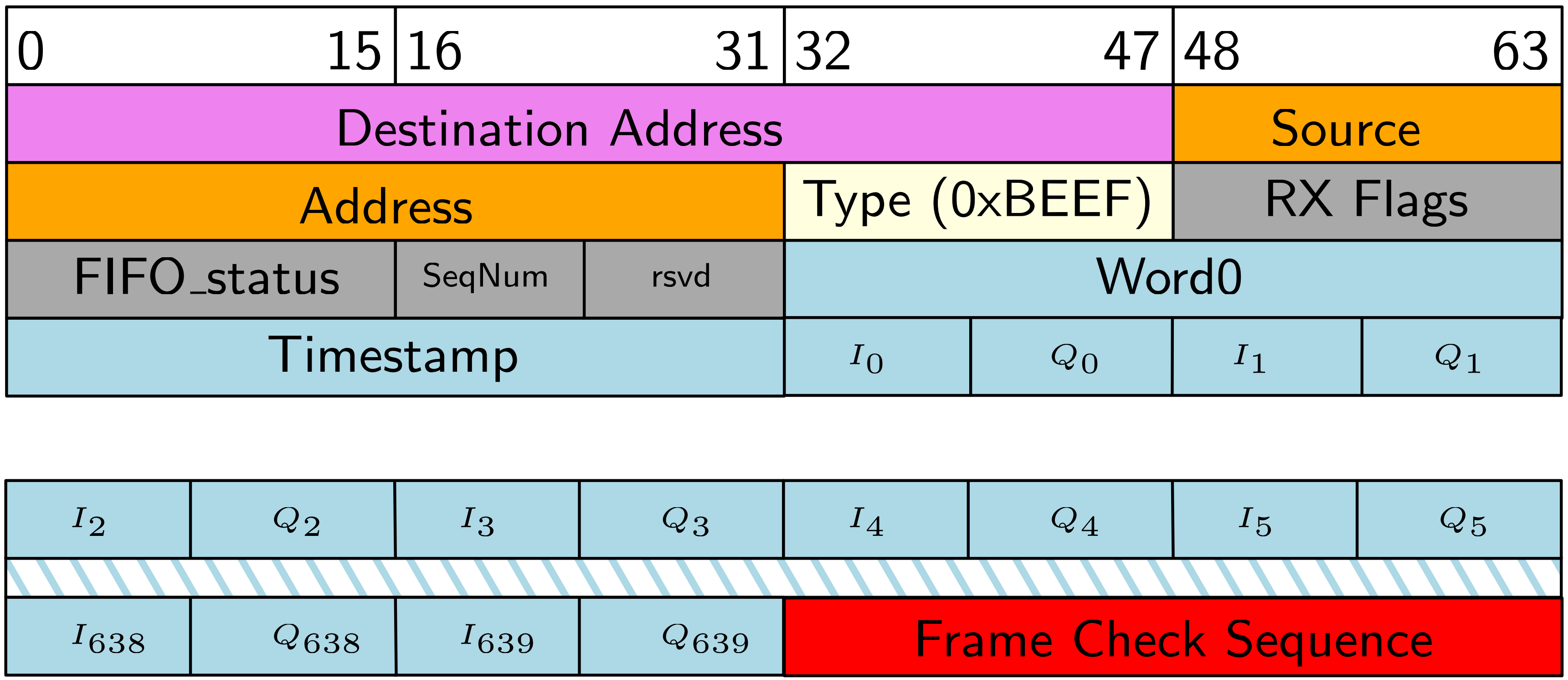
- IF5 interface
  - DL subframes with timestamp (16-bit samples)
  - UL subframes with timestamp (16-bit samples)
  - optional A-law compression (13$\rightarrow$8bit)
- IF4p5 interface
  - DL packets, ofdm symbols with frame/subframe/symbol count
  - UL packets, ofdm symbols with frame/subframe/symbol count (16-bit)
  - UL PRACH packets with frame/subframe count (16-bit)
  - optional A-law compression (13$\rightarrow$8bit)

IF5 Packet Format (16-bit)



- **Type:** 2 byte (16 bit) field that specifies the RoE protocol
- **RX Config:** 16-bit. Currently just antenna index (0-7). Can later be used for gain/timing adjustments.
- **Timstamp:** Timestamp in samples of the first sample of the received packet.
- **data block :** Uncompressed IQ samples, 16-bit resolution for each real and imaginary component. N complex samples per packet. N can be configured at initialization.

IF5 Packet Format (8-bit)



- **Type:** 2 byte (16 bit) field that specifies the RoE protocol
- **RX Flags:** overrun indicator. should be '0'.
- **FIFO status:** 2 bytes. should be '0'.
- **SeqNum:** 1 byte. Sequence number of the ethernet packet.
- **rsvd:** 1 byte. shoult be '0'.
- **Word0:** 4 byte (32-bit). should be '0'.
- **Timstamp:** Timestamp in samples of the first sample of the received packet.

- **data block :** Uncompressed IQ samples, 8-bit resolution for each real and imaginary component. 640 complex samples per packet.

IF4p5 Packet Formats (RAW)

IF4p5 PRACH Packet (RRU→ RAU,RCC)

0 15	16 31	32 47	48 63
Destination Address			Source
Address		Type (0x080A)	Subtype (0x0021)
Reserved		LTE PRACH Configuration	
PRACH data block (one antenna)			
Frame Check Sequence			

- **Type:** 2 byte (16 bit) field that specifies the RoE protocol
- **Subtype:** 2 byte (16 bit) field that specifies the packet subtype
- **Reserved:** 4 byte (32 bit) field reserved
- **LTE PRACH conf:** 4 byte (32-bit) field that details the configuration of the LTE PRACH packet

field (0 is LSB, 31 is MSB)	description
rsvd (0:2)	Reserved.
ant (3:5)	3-bit Antenna index of LTE PRACH packet
RF Num (6:21)	16-bit field indicating the Radio Frame number of this received PRACH packet
SF Num (22:25)	4-bit field indicating the sub-frame number in the radio frame for the LTE PRACH packet Valid range of 0 to 9.
Exponent (26:31)	FFT exponent output (0 if unscaled)

- **PRACH data block:** Uncompressed IQ samples

RE 0 (Real)	RE 0 (Imag)	RE 1 (Real)	RE 1 (Imag)
RE 837 (Real)	RE 837 (Imag)	RE 838 (Real)	RE 838 (Imag)

IF4p5 Packets : ULRE (RRU$\rightarrow$ RAU,RCC)

<table>
<tr><td colspan="2">0 15</td><td colspan="2">16 31</td><td colspan="2">32 47</td><td colspan="2">48 63</td></tr>
<tr><td colspan="6">Destination Address</td><td colspan="2">Source</td></tr>
<tr><td colspan="4">Address</td><td colspan="2">Type (0x080A)</td><td colspan="2">Subtype (0x0019)</td></tr>
<tr><td colspan="4">Reserved</td><td colspan="4">Frame status</td></tr>
<tr><td colspan="2">Gain 0</td><td colspan="2">Gain 1</td><td colspan="2">Gain 2</td><td colspan="2">Gain 3</td></tr>
<tr><td colspan="2">Gain 4</td><td colspan="2">Gain 5</td><td colspan="2">Gain 6</td><td colspan="2">Gain 7</td></tr>
<tr><td>RE 0
Ant 1 (Re)</td><td>RE 0
Ant 1 (Im)</td><td>RE 1
Ant 1 (Re)</td><td>RE 1
Ant 1 (Im)</td><td>RE 2
Ant 1 (Re)</td><td>RE 2
Ant 1 (Im)</td><td>RE 3
Ant 1 (Re)</td><td>RE 3
Ant 1 (Im)</td></tr>
<tr><td colspan="8"></td></tr>
<tr><td>RE $N-4$
Ant R (Re)</td><td>RE $N-4$
Ant R (Im)</td><td>RE $N-3$
Ant R (Re)</td><td>RE $N-3$
Ant R (Im)</td><td>RE $N-2$
Ant R (Re)</td><td>RE $N-2$
Ant R (Im)</td><td>RE $N-1$
Ant R (Re)</td><td>RE $N-1$
Ant R (Im)</td></tr>
<tr><td colspan="4">Frame Check Sequence</td></tr>
</table>

- **Type:** 2 byte (16 bit) field that specifies the RoE protocol
- **Subtype:** 2 byte (16 bit) field that specifies the packet subtype
- **Reserved:** 4 byte (32 bit) field reserved
- **Frame Status:** 4 byte (32 bit) field

field (0 is LSB, 31 is MSB)	**description**
ant (0:2)	The number of Antenna Carriers represented in the packet. Antenna numbers range from 0 to 7 with valid inputs being 0,1, 3 and 7 (1,2,4,8 antennas)
ant start (3:5)	starting antenna number
RF Num (6:21)	16-bit field indicating the Radio Frame number of the UL_RE samples
SF Num (22:25)	4-bit field indicating the sub-frame number in the radio frame for the UL_RE samples Valid range of 0 to 9.
Sym Num: (26:29) **rsvd:** (30:31)	Symbol number. Valid range of 0 to 13. reserved

- **ULRE data block:** compressed IQ samples (8-bit A-law). N is the number of resource elements $N_{\mathrm{RB}}^{\mathrm{UL}}$.

IF4p5 Packets : DLRE (RAU,RCC $\rightarrow$RRU)

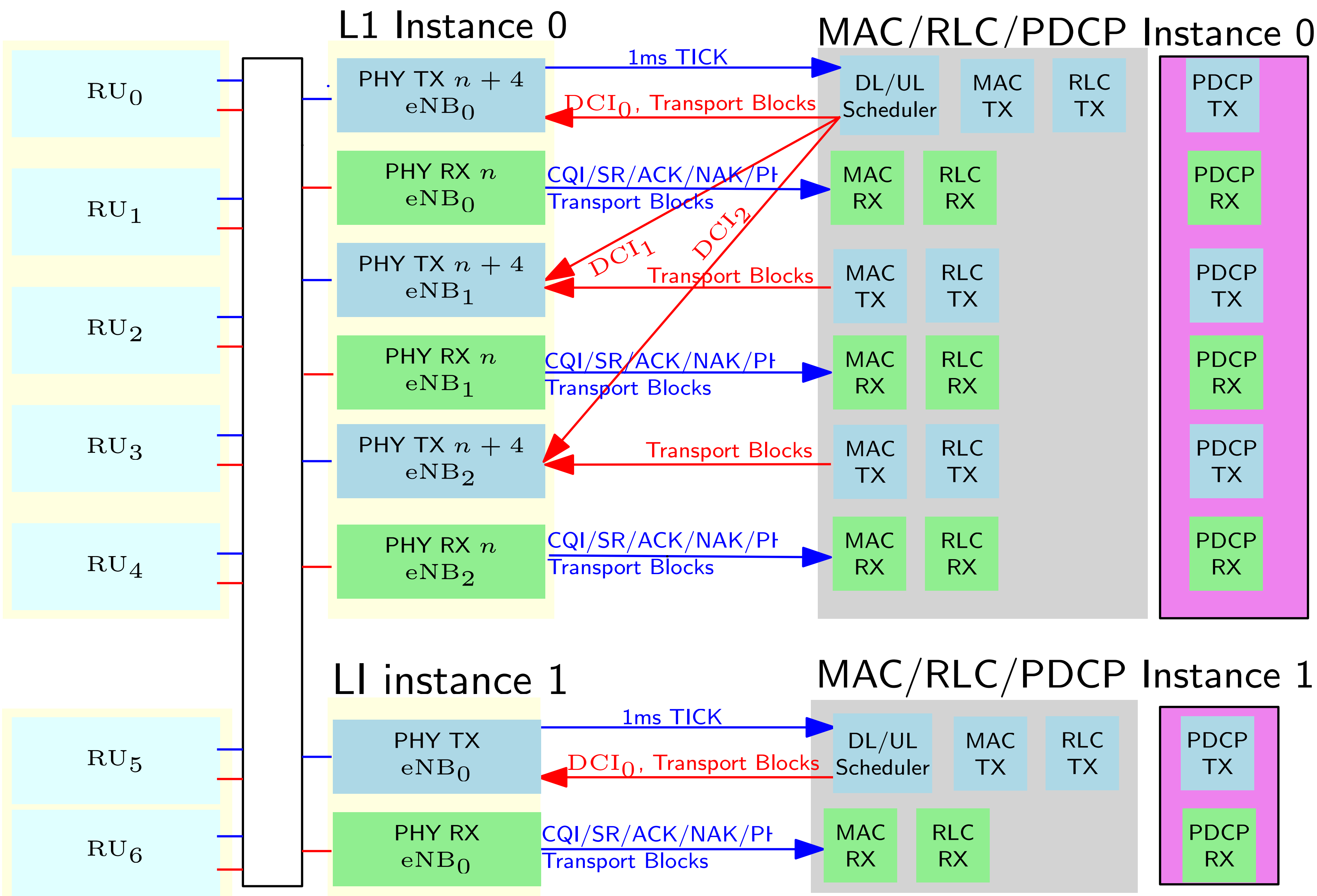
0	15	16	31	32	47	48	63
Destination Address						Source	
Address				Type (0x080A)		Subtype (0x0020)	
Reserved				Frame status			
RE 0 Ant 1 (Re)	RE 0 Ant 1 (Im)	RE 1 Ant 1 (Re)	RE 1 Ant 1 (Im)	RE 2 Ant 1 (Re)	RE 2 Ant 1 (Im)	RE 3 Ant 1 (Re)	RE 3 Ant 1 (Im)
RE $N-4$ Ant R (Re)	RE $N-4$ Ant R (Im)	RE $N-3$ Ant R (Re)	RE $N-3$ Ant R (Im)	RE $N-2$ Ant R (Re)	RE $N-2$ Ant R (Im)	RE $N-1$ Ant R (Re)	RE $N-1$ Ant R (Im)
Frame Check Sequence							

- **Type:** 2 byte (16 bit) field that specifies the RoE protocol
- **Subtype:** 2 byte (16 bit) field that specifies the packet subtype
- **Reserved:** 4 byte (32 bit) field reserved
- **Frame Status:** 4 byte (32 bit) field

field (0 is LSB, 31 is MSB)	description
ant (0:2)	The number of Antenna Carriers represented in the packet. Antenna numbers range from 0 to 7 with valid inputs being 0,1, 3 and 7 (1,2,4,8 antennas)
ant start (3:5)	starting antenna number
RF Num (6:21)	16-bit field indicating the Radio Frame number of this DLRE packet
SF Num (22:25)	4-bit field indicating the sub-frame number in the radio frame for the DLRE packet Valid range of 0 to 9.
Sym Num: (26:29) **rsvd:** (30:31)	Symbol number. Valid range of 0 to 13. reserved

- DLRE data block : compressed IQ samples (8-bit A-law). N is the number of resource elements $N_{\mathrm{RB}}^{\mathrm{DL}}$.

RU/L1 Instances and Component Carriers



RU/L1 Instances and Component Carriers

- *Radio Unit* (RU) is
  - an entity managing a set of **physical** antennas. It can have a *local RF unit* or *remote RF unit*
  - performs precoding of multiple eNB TX streams and OFDM modulation (TX) and demodulation (RX) (part of 36.211)
- *L1 Instance* (indexed by `Mod_id`, or `enb_mod_id`) is a separate set of threads and contexts for the eNB/gNB procedures. There is one MAC/RLC entity associated to all :1 component carriers.
- *L1 Component Carrier* (indexed by `CC_id`) is
  - a software entity managing the L1 procedures (36.213,36.212,36.211) and can act on
    * sectored antenna component
    * Rel10+ component carrier
    * virtual cell for DAS or Massive-MIMO array
  - each L1 instance is managed by one or two threads which operate on a subframe (TX and RX) and can have a *local RU* or *remote RU*
  - if a remote radio unit the eNB performs the 36.213 specifications only (HARQ, etc.) and connects to the remainder via the IF2 midhaul interface.

RU/L1 Instances and Component Carriers

- RU may have both an `if_device` for fronthaul and an `rf_device` for interconnection with a local RF unit
- if the `rf_device` is absent, it must have a southbound fronthaul interface (either IF5 or IF4p5) depending on the local processing of the remote RU
- if the `if_device` is absent, it must have a southbound RF interface and `rf_device`.
- three types of L1 processing are performed by the RU
  - subset of common L1 procedures from 36.211 specifications
  - fronthaul compression/decompression
  - framing
- on TX
  - A-law compression for (NGFI_RAU_IF4p5, NGFI_RAU_IF5)
  - A-law decompression (for NGFI_RRU_IF4p5 and NGFI_RRU_IF5)
  - OFDM modulation and cyclic prefix insertion (for NGFI_RRU_IF4p5,NGFI_RAU_IF5,3GPP_eNodeB_BBU,3GPP_eNodeB)
  - Precoding (for NGFI_RAU_IF5, NGFI_RAU_IF4p5,3GPP_eNodeB_BBU,3GPP_eNodeB)

RU/L1 Instances and Component Carriers

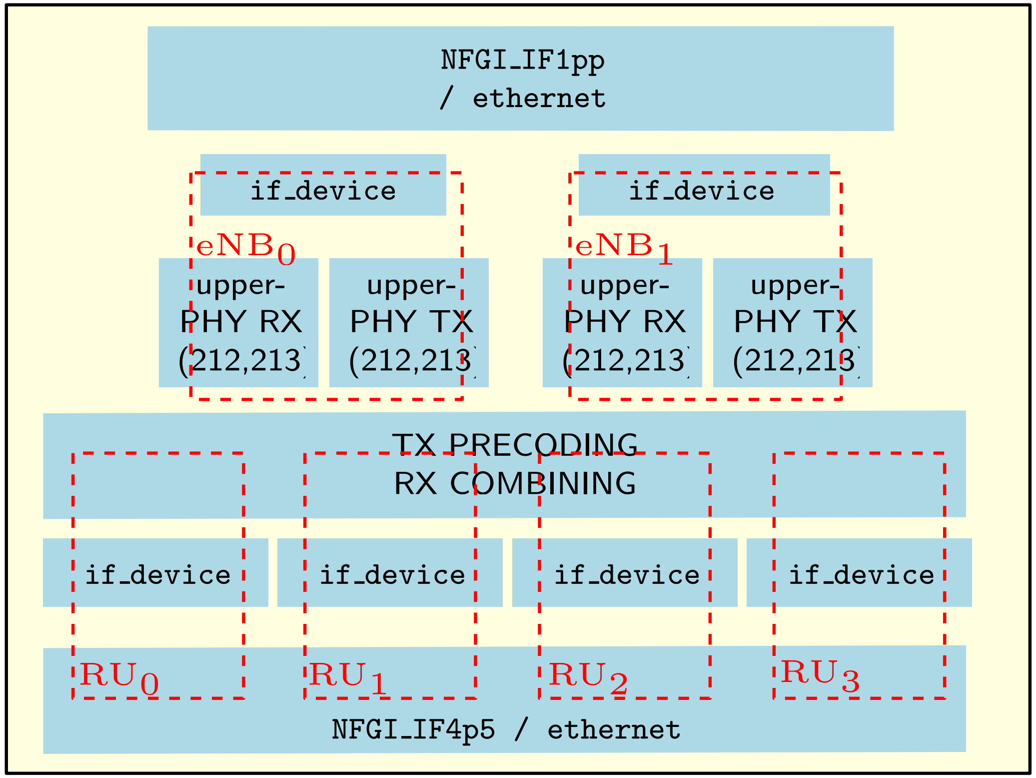
- on RX
  - A-law compression for (NGFI_RRU_IF4p5, NGFI_RRU_IF5)
  - A-law decompression (for NGFI_RAU_IF4p5 and 3GPP_eNodeB_BBU)
  - cyclic prefix removal, frequency-shifting, OFDM demodulation, PRACH DFT (for NGFI_RRU_IF4p5, NGFI_RAU_IF5, 3GPP_eNodeB_BBU, 3GPP_eNodeB)
- On TX path
  - L1 instances/component carriers operate on a set of logical antenna ports (0-3 for TM1-6, 4 for eMBMS, 5 for TM7, 6 for positioning, 7-8 for TM8, etc.)
  - each L1 instance has a list of RUs and the logical antenna ports are mapped to the physical antennas attached to the RUs via the precoding function

RU/L1 Instances and Component Carriers

- Example configurations
  - *itsolated eNB:* one instance and one or several component carriers (multiple-frequencies or antenna sectors). Potentially multiple radio-units (for CoMP). Here there is a common MACRLC instnance driving multiple L1 procedures
  - *indoor DAS system (RCC split with L1/L2 RAU) Multiple layer 2 instances each driving one or more component carriers* here the RAU implements multiple L1/L2 instances and precoding function. Usually with IF2/IF1" xhaul to RCC or potentially also MAC/RLC in RAU with IF1' xhaul to RCC.
  - *massive-MIMO array* same as 2nd indoor DAS system (i.e. integrated L1/L2 RAU with array)

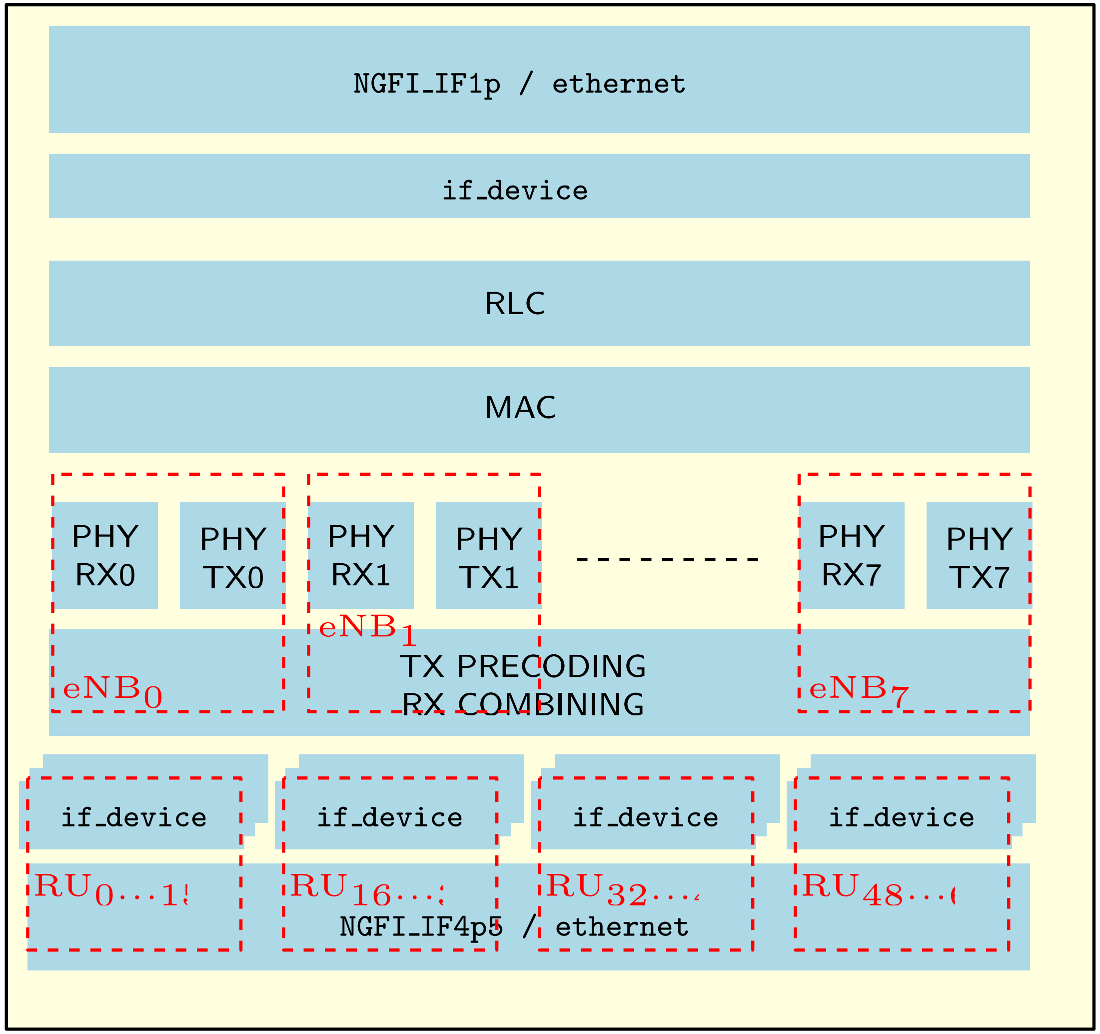
RAU Example (DAS)

- Example: RAU with `NGFI_IF1pp` xhaul (MAC/PHY split) northbound, `NGFI_IF4p5` fronthaul southbound, 2 vCell logical interfaces (2 L1/L2 instances, or 1 L2 instance and 2 CCs), 4 RRUs with `NGFI_IF4p5`



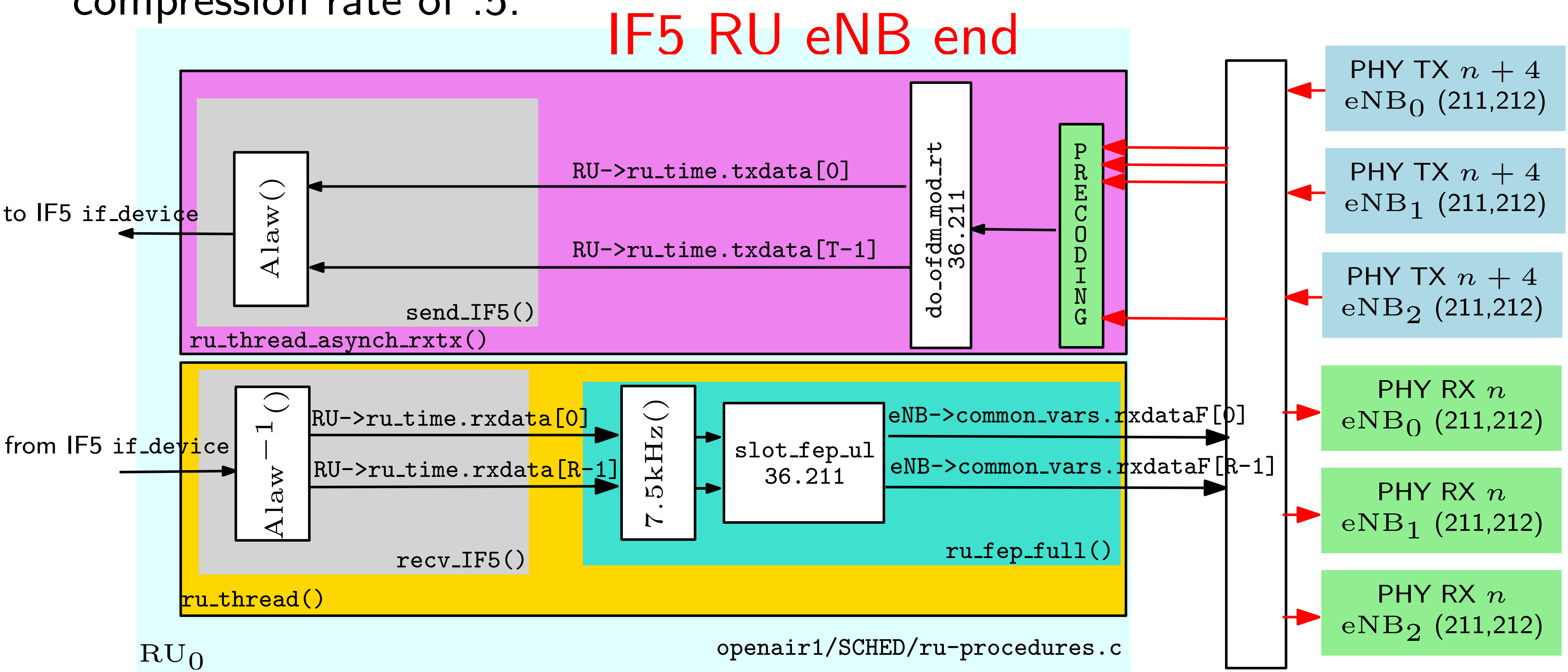
RAU Example (Massive-MIMO)

- Example: massive-MIMO RAU with `NGFI_IF1p` fronthaul northbound, 8 L1 component carriers,1 L2 instances, many local RRUs with `NGFI_IF4p5` southbound



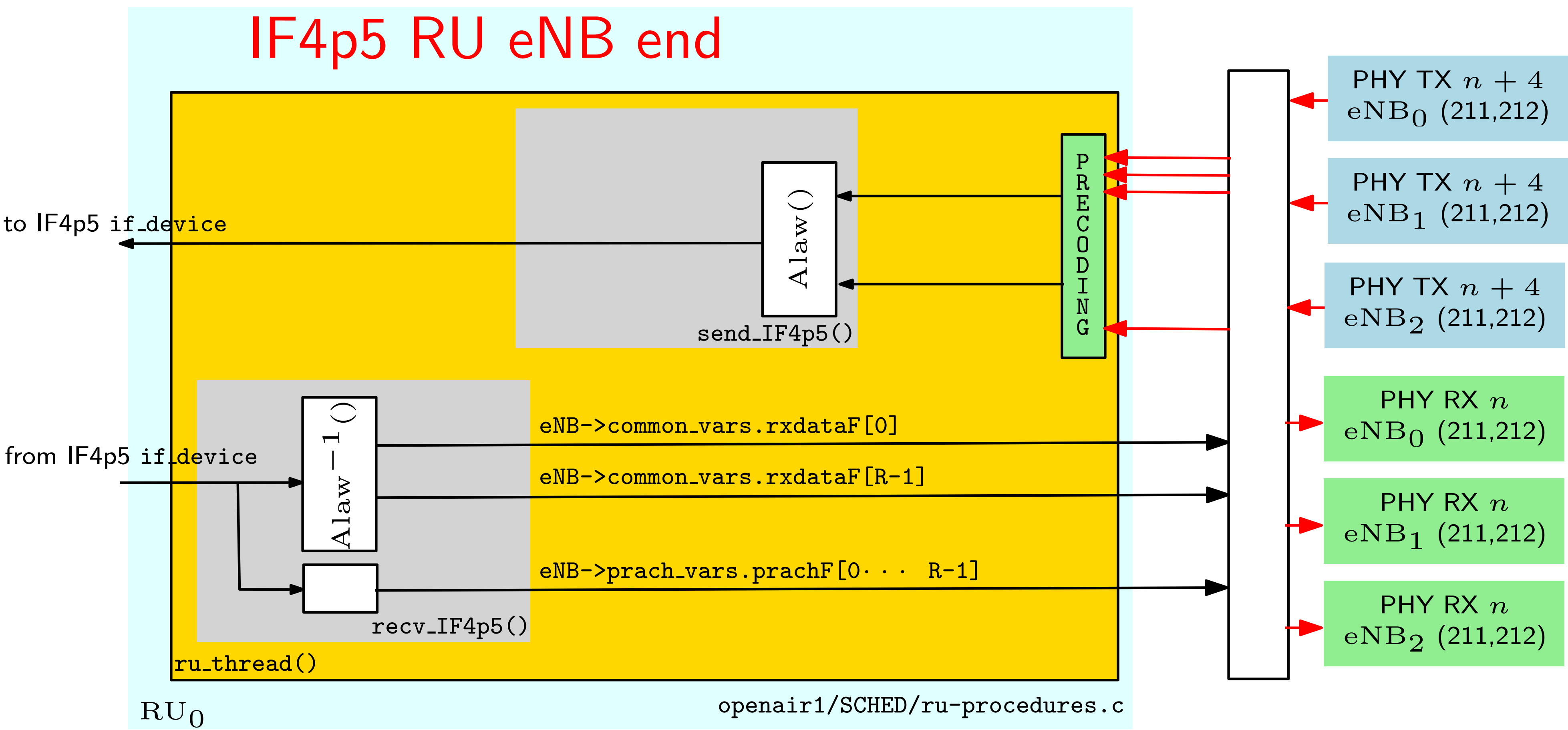
RU Procedures (NGFI_IF5)

- IF5 transports packets of size equal to a subframe and corresponding to a 1ms chunk of signal in the time-domain. This is done via the functions send_if5 and recv_if5, in the layer1 transport procedures (`openair1/PHY/LTE_TRANSPORT/if5_tools.c`). A timestamp is given along with the samples, corresponding to the time (in samples) of the first sample of the packet.
- each block can be compressed with A-law compression, yielding a compression rate of .5.



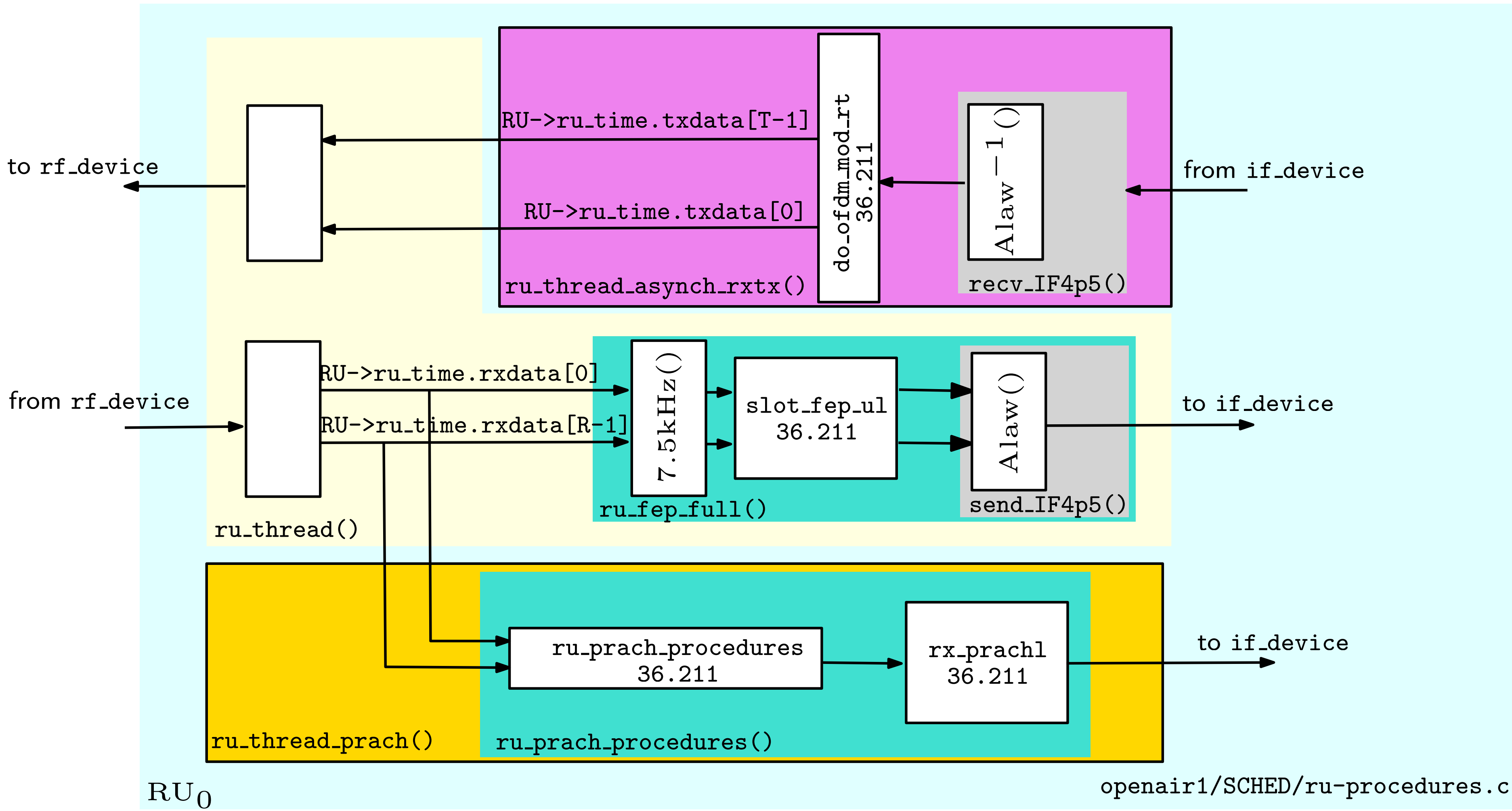
RU Procedures (NGFI_IF4p5)

- IF4p5 transports packets of size equal to an OFDM symbol (for DLRE and ULRE) indexed by the symbol, subframe and frame number. This is done via the functions send_if4p5 and recv_if4p5, in the layer1 transport procedures (`openair1/PHY/LTE_TRANSPORT/if4_tools.c`).
- each block are compressed with A-law compression, yielding a compression rate of .5.



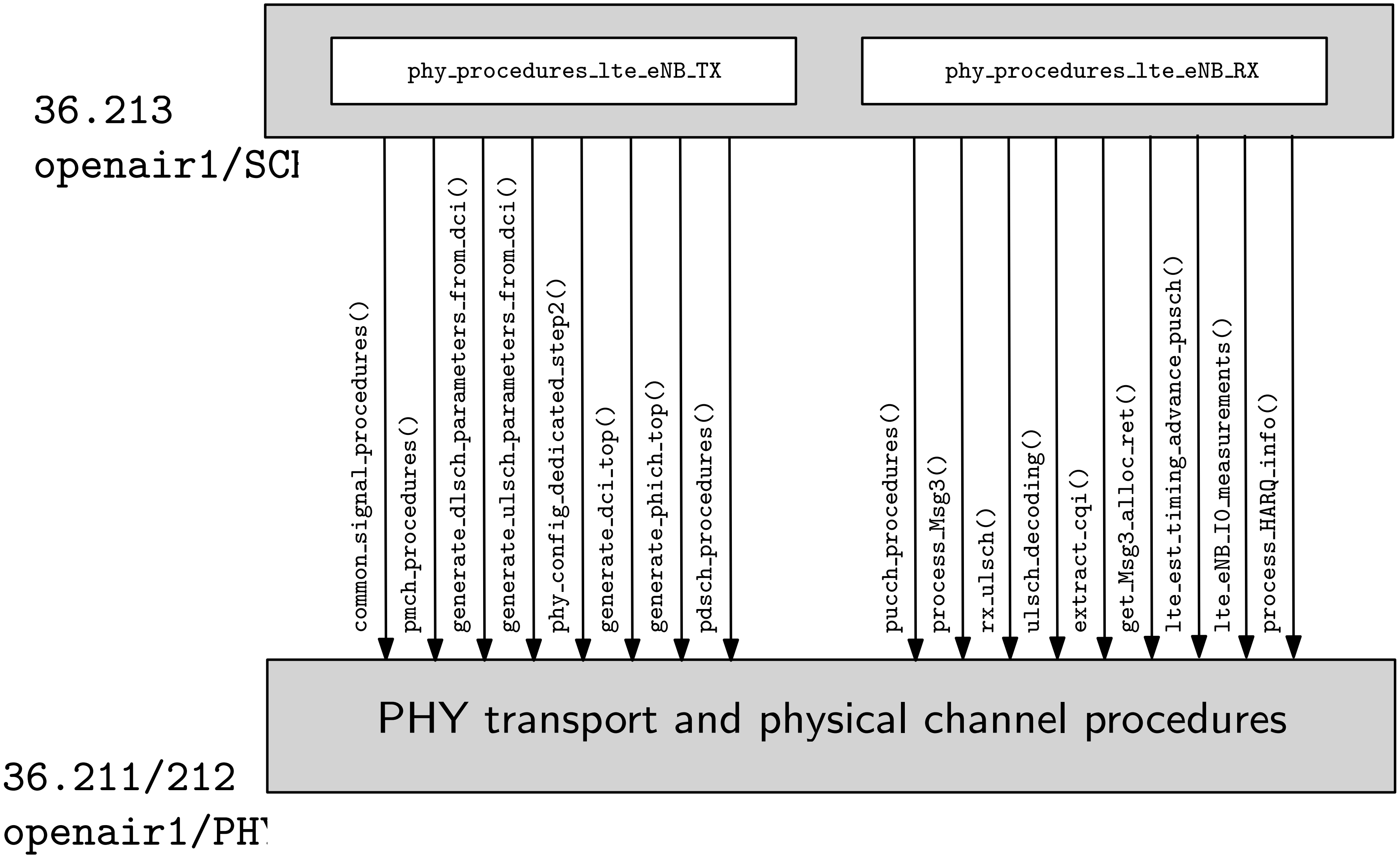
RU Procedures (NGFI_IF4p5)

IF4p5 RU remote-end

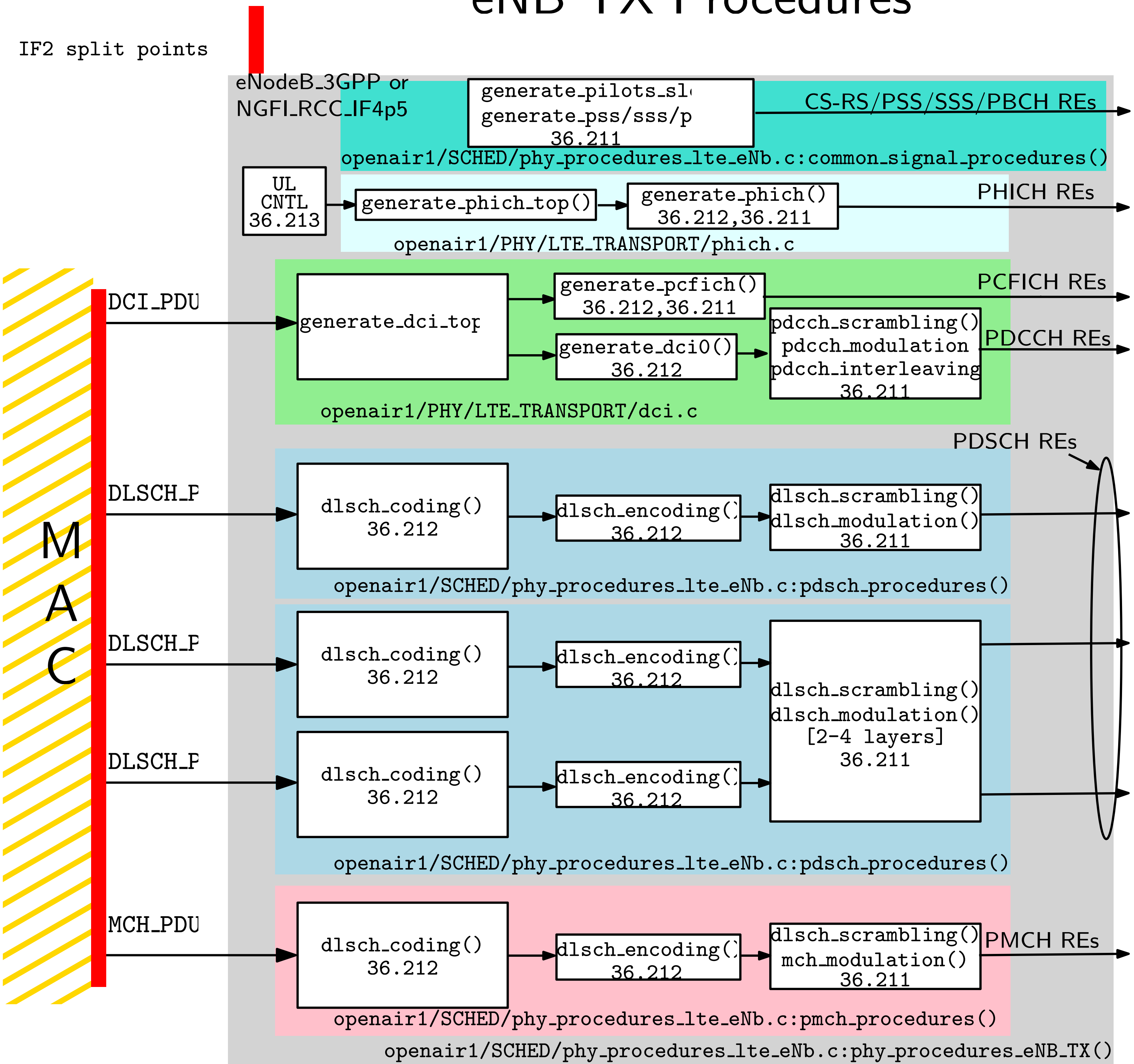


OAI IF1pp Interface

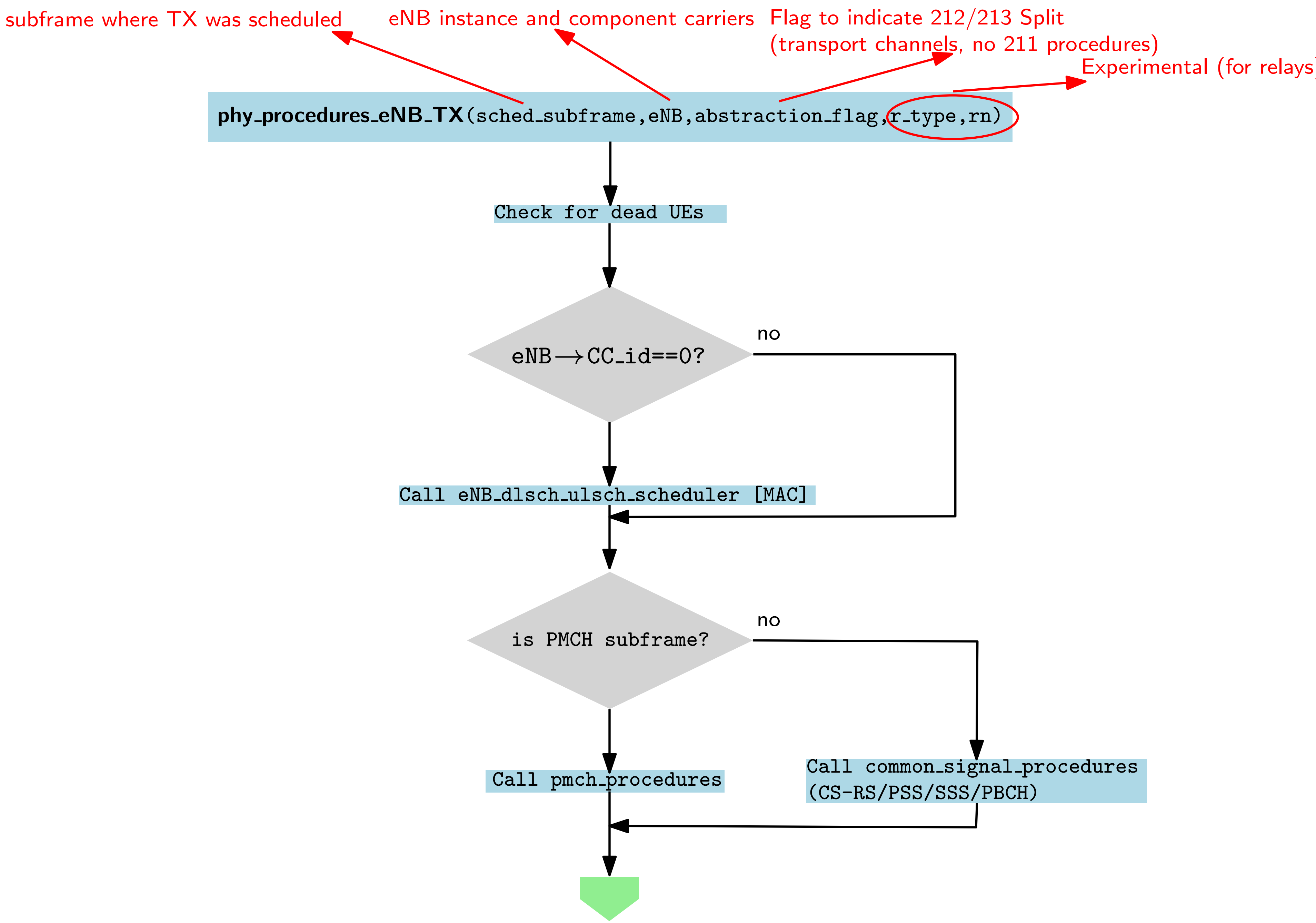
- OAI IF1pp is the interface between the 36.213 Physical Layer Procedures (HARQ, SR, CSI, etc.) and the transport/physical channel processing
- it can be networked, although this is not used as an xhaul interface at the moment.



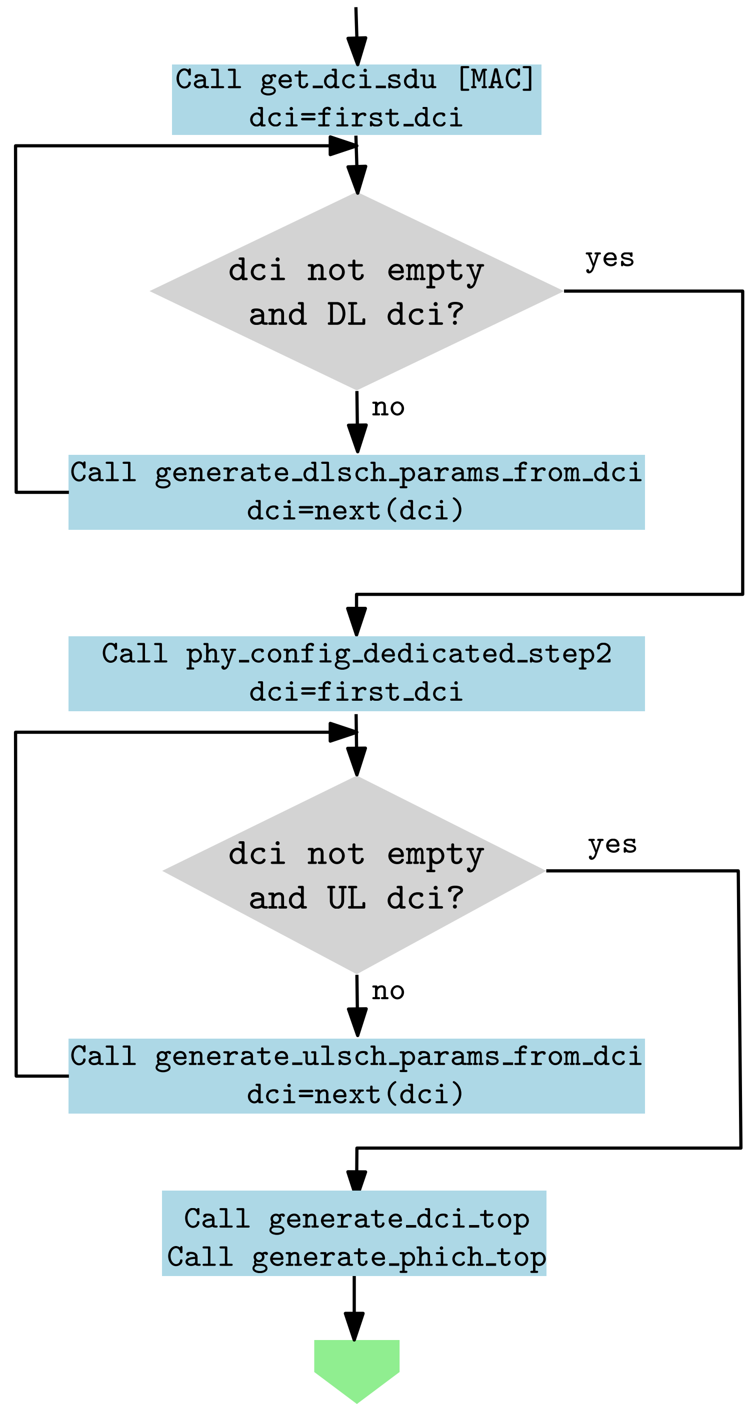
eNB TX Procedures



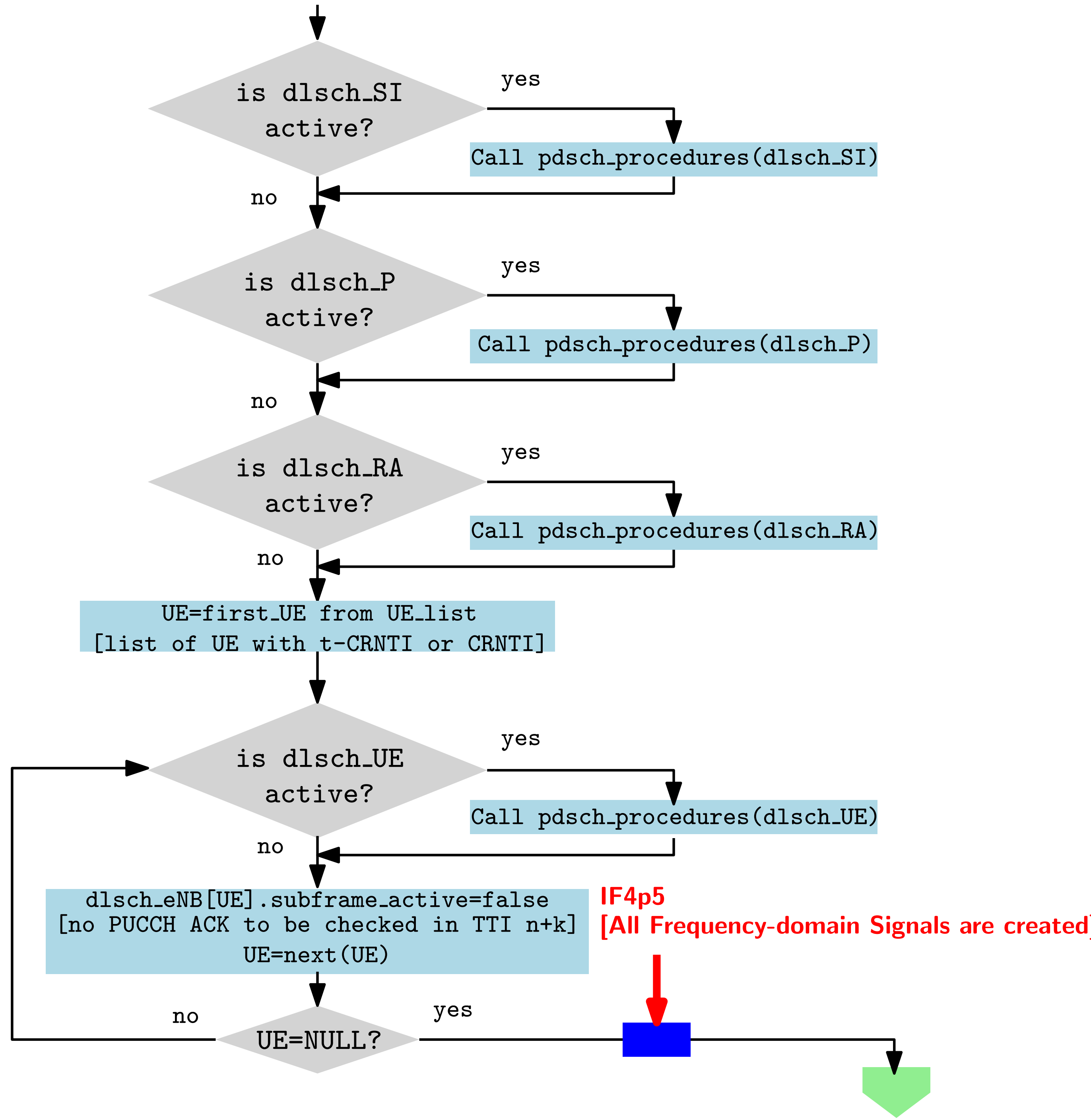
eNB TX Flowchart



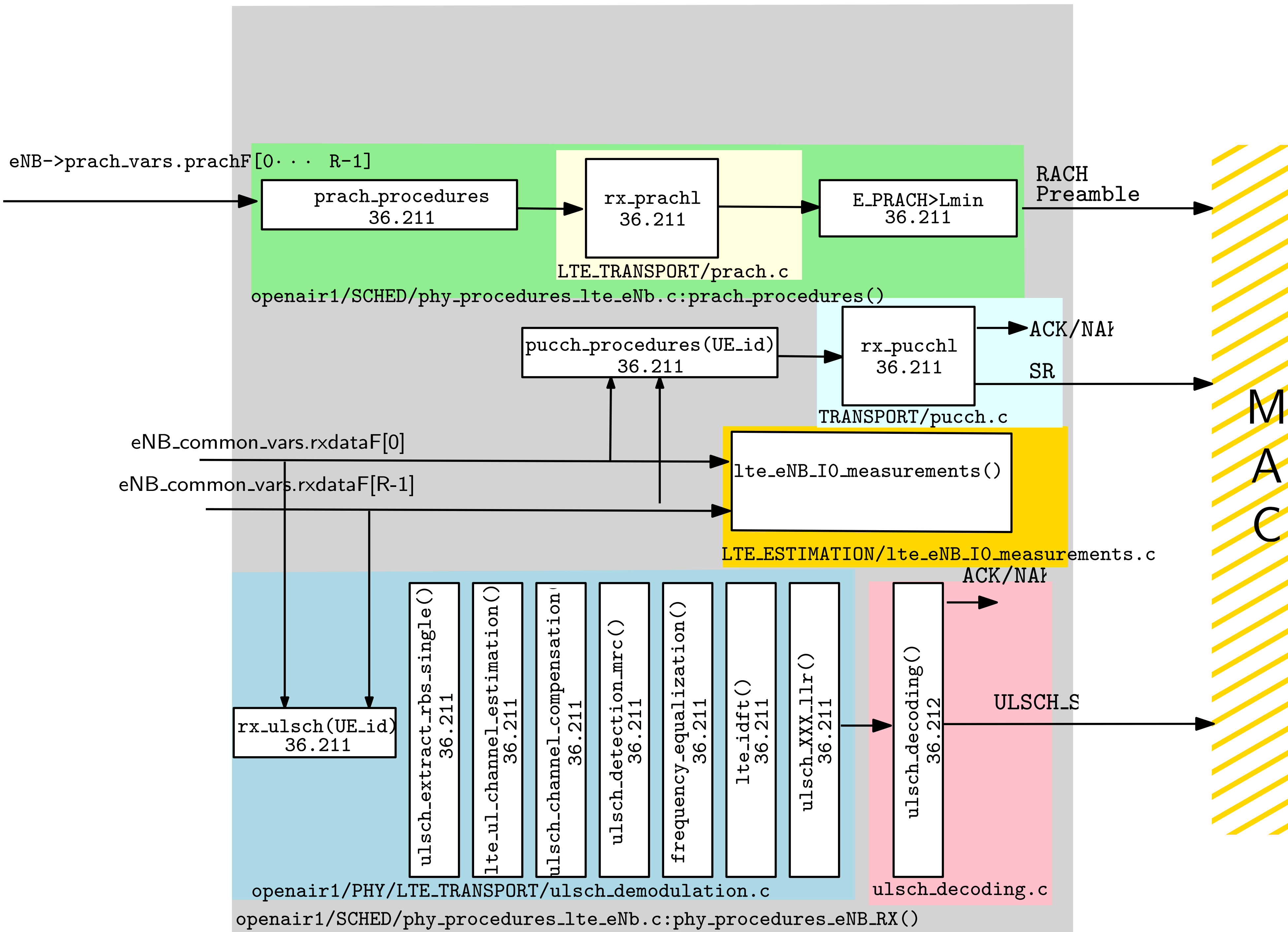
eNB TX Flowchart



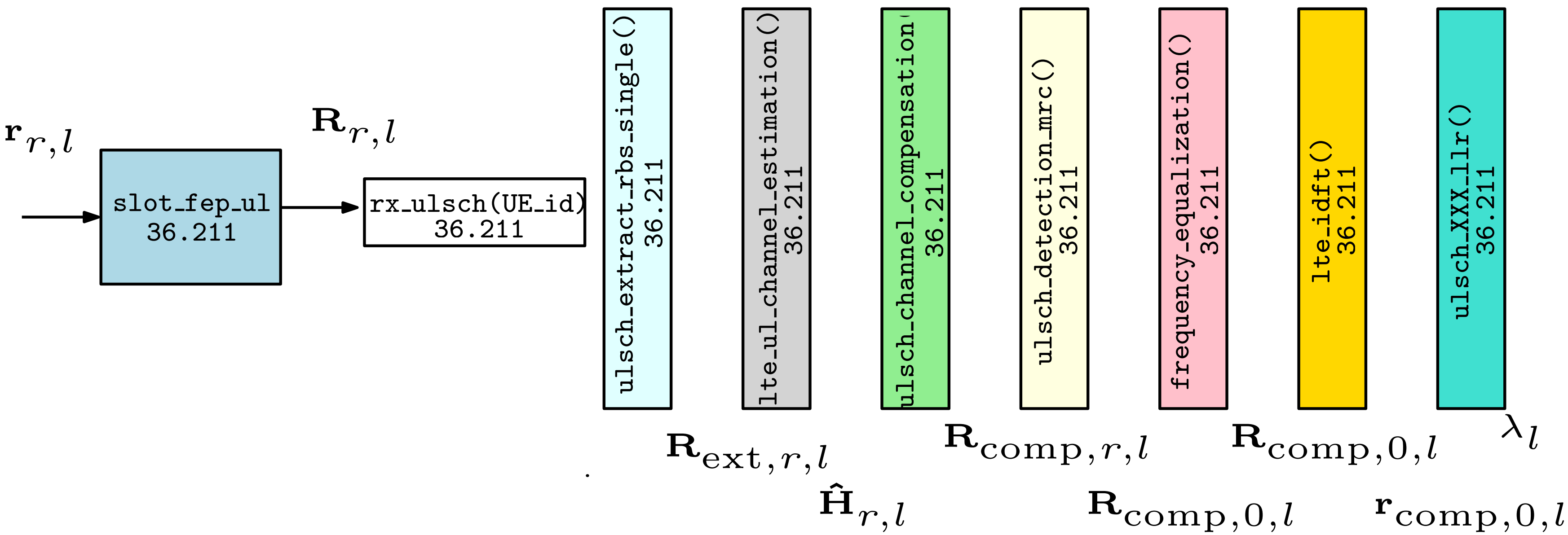
eNB TX Flowchart



eNB PHY RX Procedures



eNB ULSCH Demodulation



$\mathbf{R}_{r,l} = \mathrm{DFT}_{N_{\mathrm{fft}}}(\mathbf{r}_{r,l} \odot \mathbf{F}_{7.5}), r = 0, 1, \cdots, R-1, l = 0, 1, \cdots, N_{\mathrm{symb}} - 1$ **(eNB_common_vars→rxdataF[][])**

$R_{\mathrm{ext},r,l}(n) = R_{r,l}(12\mathrm{firstPRB} + n), n = 0, 1, \cdots, 12N_{\mathrm{PRB}} - 1$ **(eNB_pusch_vars→ulsch_rxdataF_ext[][])**

$\hat{\mathbf{H}}_{r,l} = \mathbf{R}_{\mathrm{ext},r,l} \odot \mathbf{DRS}^*_l(\mathrm{cyclicShift}, n_{\mathrm{DMRS}(2)}, n_{\mathrm{PRS}}),$ **(eNB_pusch_vars→drs_ch_estimates[])**

$\mathbf{R}_{\mathrm{comp},r,l} = \hat{\mathbf{H}}_r \odot \mathbf{R}_{\mathrm{ext},r,l} 2^{-\log_2|H_{\max}|}, \hat{\mathbf{H}}_r = \frac{1}{2}(\hat{\mathbf{H}}_{r,3} + \hat{\mathbf{H}}_{r,10})$
(eNB_pusch_vars→ulsch_rxdataF_comp)

$\mathbf{R}_{\mathrm{comp},0,l} = \frac{1}{R}\sum_{r=0}^{R-1} \mathbf{R}_{\mathrm{comp},r,l}$

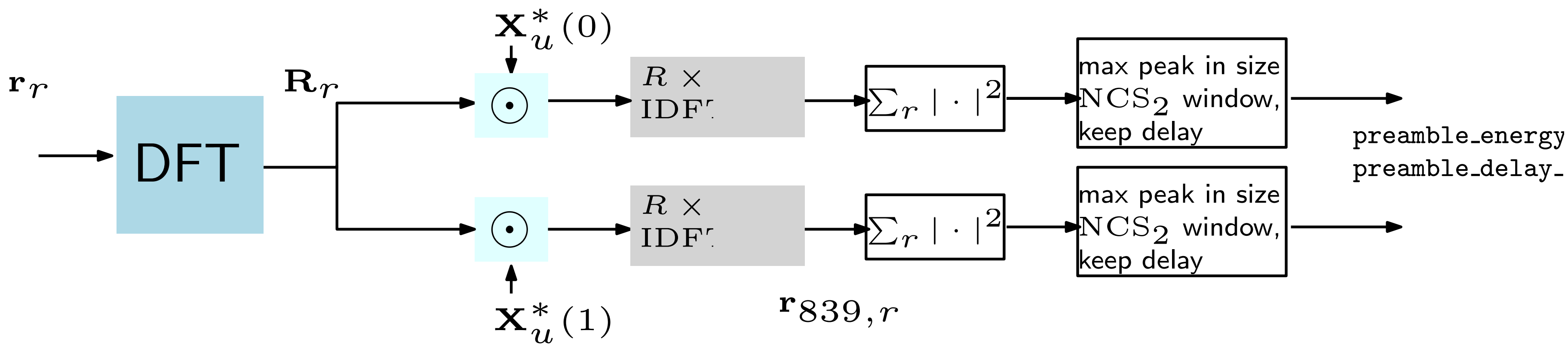
$R_{\mathrm{comp},0,l}(n) = R_{\mathrm{comp},0,l}(n)\dot{Q}_8\left(\frac{1}{|\hat{H}(n)|^2 + I_0}\right), \hat{H}(n) = \sum_{r=0}^{R-1} \hat{H}_r(n)$

$\mathbf{r}_{\mathrm{comp},0,l} = \mathrm{IDFT}_{12N_{\mathrm{PRB}}}(\mathbf{R}_{\mathrm{comp},0,l})$

QPSK : $\lambda_l(2n) = \mathrm{Re}(r_{\mathrm{comp},0,l}(n)), \lambda_l(2n+1) = \mathrm{Im}(r_{\mathrm{comp},0,l}(n))$ **(eNB_pusch_vars→ulsch_llr)**

16QAM : $\lambda_l(4n) = \mathrm{Re}(r_{\mathrm{comp},0,l}(n)), \lambda_l(4n+2) = \mathrm{Im}(r_{\mathrm{comp},0,l}(n))$
$\lambda_l(4n+1) = |\mathrm{Re}(r_{\mathrm{comp},0,l}(n))| - 2\overline{|h(n)|}, \lambda_l(4n+3) = |\mathrm{Im}(r_{\mathrm{comp},0,l}(n))| - 2\overline{|h(n)|}$

eNB PRACH Detection



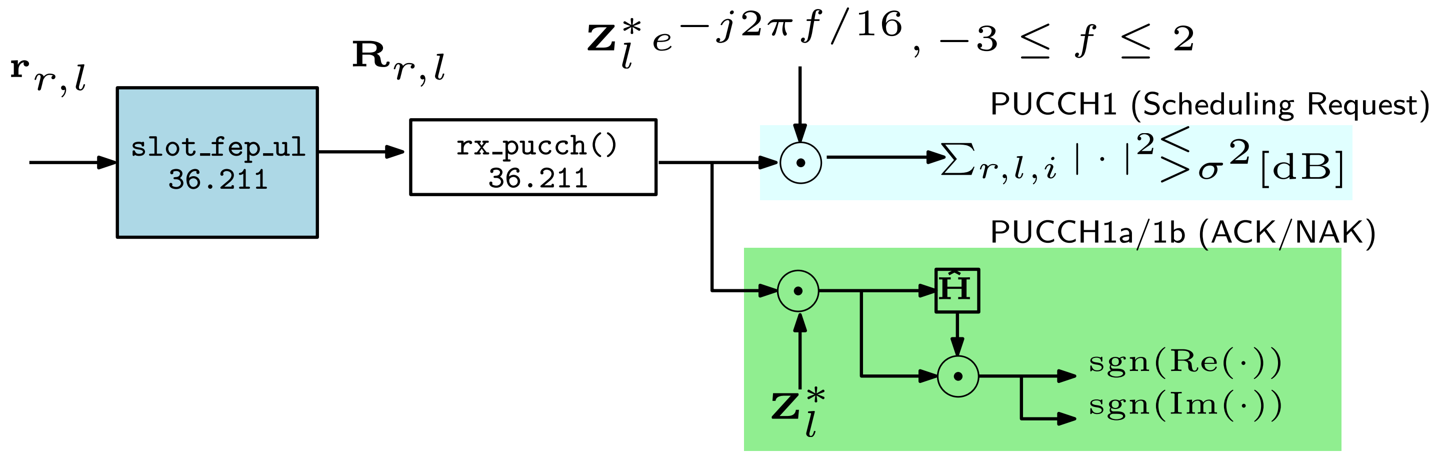
$\mathbf{R}_r = \mathrm{DFT}_{N_{\mathrm{PRACH}}}(\mathbf{r}_r),\ r = 0, 1, \cdots, R-1$ (lte_eNB_prach_vars→rxsigF[])

$\mathbf{R}_{\mathrm{comp},r} = \mathbf{R}_r \odot \mathbf{X}_u^*[i],\ r = 0, 1, \cdots, R-1$ (lte_eNB_prach_vars→prachF[])

$\mathbf{r}_{839,r} = \mathrm{IDFT}_{1024}\left(\mathbf{R}_{\mathrm{comp},r}\right),\ r = 0, 1, \cdots, R-1$ (lte_eNB_prach_vars→prach_ifft[])

- PRACH detection is a quasi-optimal non-coherent receiver for vector observations (multiple antennas)
- correlation is done in the frequency-domain, number of correlations (in the example above 2) depends on *zeroCorrelationConfig* configuration parameter
- peak-detection (for delay estimation) is performed in each NCS time-window

eNB PUCCH1 Detection



- PUCCH1 detection is a quasi-optimal non-coherent receiver (energy detector) for vector observations (multiple antennas) for scheduling request. Care is taken to handle residual frequency-offset.
- PUCCH1A/1B detection is quasi-coherent based on a rough channel estimate obtained on the 3 symbols without data modulation.
- In both cases, correlation is done in the frequency-domain

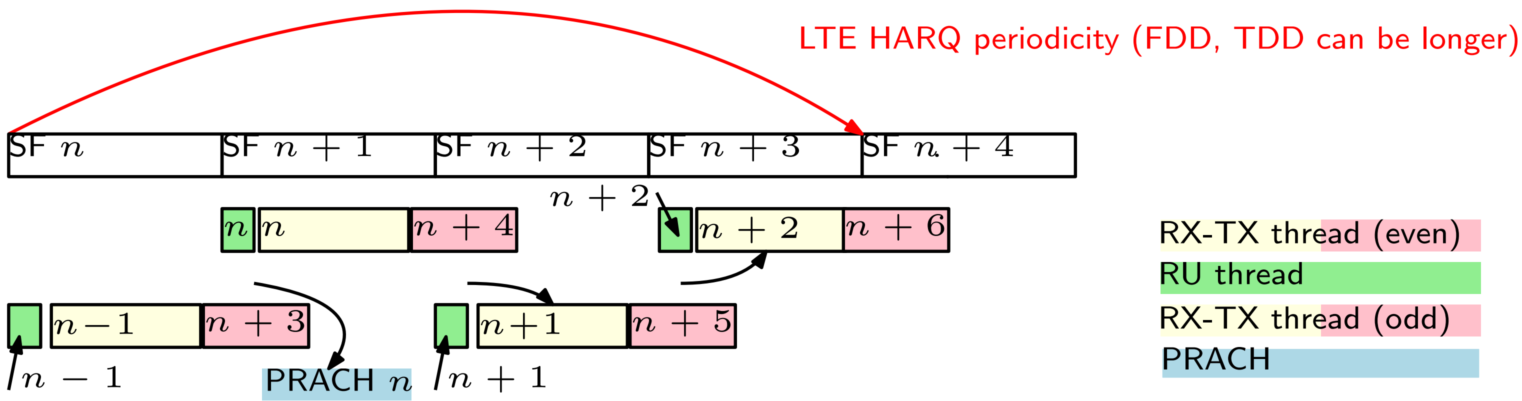
RU Threads

- Threads (all in `targets/RT/USER/lte-ru.c`)
  - `ru_thread`: Thread per RU which sequentially performs
    * read from south interface (RF or IF fronthaul)
    * RX processing for subframe n (if necessary).
    * wakeup eNBs that are waiting for signal (if necessary)
    * wait for eNB task completion (if necessary)
    * do TX processing for subframe $n+4$ (if necessary). Note that this can spawn multiple worker threads for very high order spatial processing (e.g. massive-MIMO or DAS for UDN)
    * do outgoing fronthaul (RF or IF fronthaul)
  - `ru_thread_prach`: Thread for PRACH processing in remote RU (DFT on RX, IF4p5 RRU)
  - `ru_thread_asynch`: Thread for asynchronous reception from fronthaul interface (TX direction in RRU).
- Synchronization on fronthaul interface
  - *synch_to_ext_device* : synchronizes to incoming samples from RF or Fronthaul interface using blocking read
  - *synch_to_other* : synchronizes via POSIX mechanism to other source (other CC, timer) which maintains real-time.

eNB Threads

- Threads (all in `targets/RT/USER/lte-enb.c`)
  - multi RX/TX thread mode (optional)
    - `eNB_thread_rxtx`: 2 threads per CC/Instance which do both RX procedures for subframe n and TX procedures for subframe $n+4$. One operates on even subframes, one on odd. This allows 1ms subframe processing to use multiple-cores.
  - common RU-eNB RX/TX thread (default if single RU/eNB)
    - calls `eNB_top`: procedure per CC/Instance which sequentially
      - blocks on signal from RU
      - RX/TX processing for subframe n and $n+4$
      - signals completion to RU
  - `eNB_prach`: Thread per CC_id/Instance for PRACH processing

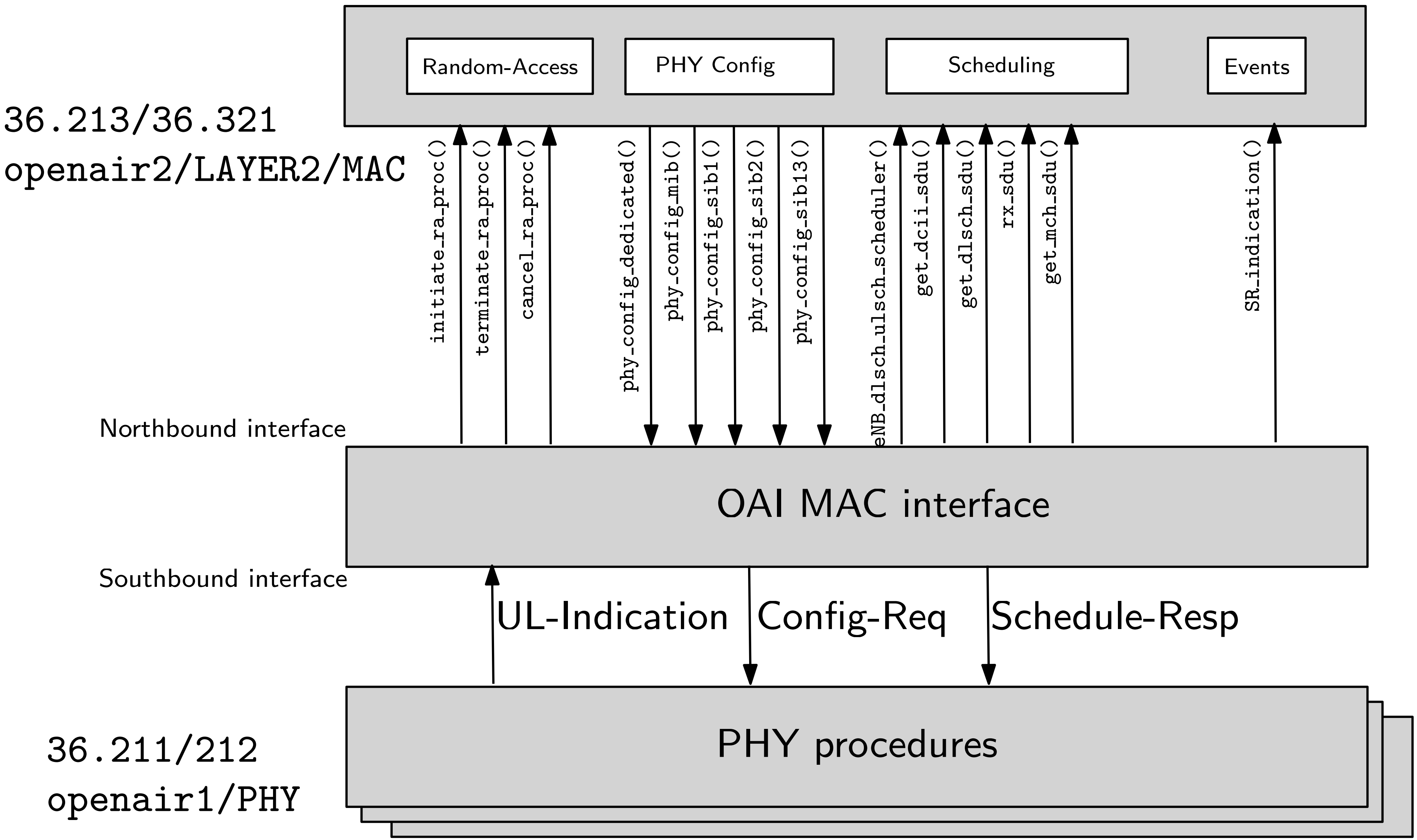
eNB Timing (multi-thread mode)



- The current processing requires approximately 1ms peak in each direction (basically 1 core RX, 1core TX). The current architecture will work on a single core if the sum of RX and TX procedures is limited to 1ms. It can fit on 2 cores if the sum of RX,TX and PRACH is less than 2ms.
- three threads, RX-TX even, RX-TX odd and PRACH. RX-TX blocks until woken by the RU thread with a new RX subframe n that is linked to this eNB process. The RX-TX thread performs ue-specific processing for subframe n and then TX common and ue-specific processing for subframe $n+4$ (frequency-domain generation only). This insures the data dependency between TX $n+4$ and RX n is respected. The duration of this thread should be less than 2ms which can compensate some jitter on the RX processing.

OAI IF1” Interface (can be NFAPI)

- OAI IF1” is the interface between the 36.321 Medium-Access (MAC) Layer Procedures and the 36.213 Physical Layer Procedures. It links several PHY instances to one MAC instance.
- It is a configurable (dynamically loadable) module which can implement an (N)FAPI P5/P7 or a simpler interface.



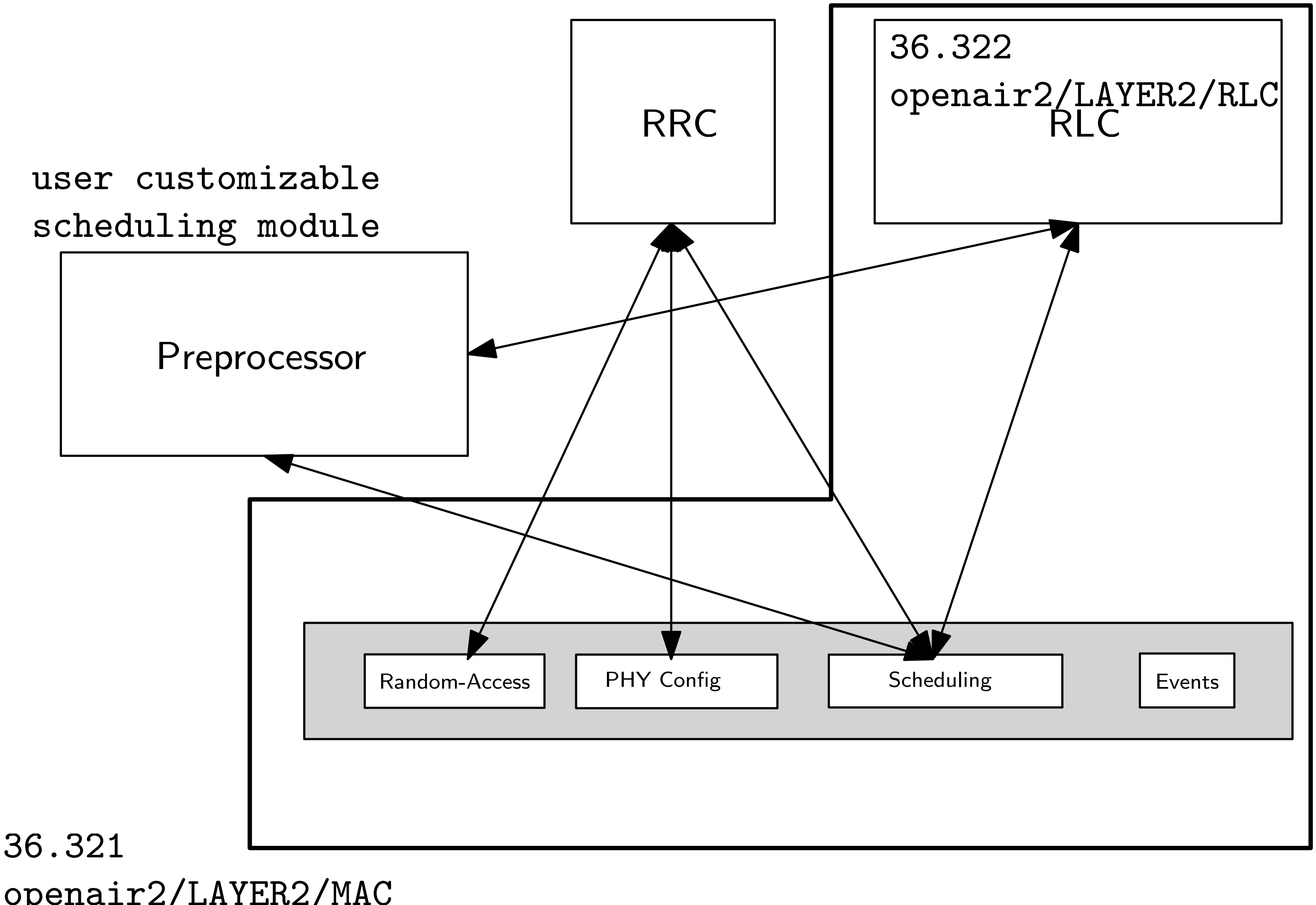
OAI IF1" Interface

- The PHY end uses three basic messages
  - CONFIG_REQ: this provides the cell configuration and UE-specific configuration to the PHY instances. This comprises the following FAPI P5/P7 messages
    1. CONFIG.request
    2. UE_CONFIG.request (**not used in OAI PHY)
  - UL_INDICATION This is an uplink indication that sends all UL information received in one TTI, including PRACH, if available. It also provides the subframe indication for the DL scheduler. It maps to the following FAPI P7 messages
    1. SUBFRAME.indication
    2. HARQ.indication
    3. CRC.indication
    4. RX_ULSCH.indication
    5. RX_SR.indication
    6. RX_CQI.indication
    7. RACH.indication
    8. SRS.indication
  - SCHEDULE_REQUEST This message contains the scheduling response information and comprises the following FAPI P7 messages
    1. DL_CONFIG.request
    2. UL_CONFIG.request
    3. TX.request
    4. HI_DCI0.request
- The module is registered both by PHY and MAC and can implement different types of transport (NFAPI, function call, FAPI over UDP, etc.). During registration, fuction pointers for the different messages are provided for the module to interact with either PHY or MAC or both if they are executing in the same machine. Note that for a networked implementation (e.g. NFAPI), there are north and south components running in different machines.

OAI IF1'' Interface

- The PHY-layer timing is assumed to be
  1. wait for subframe indication n from HW
  2. trigger PRACH if n has PRACH (parallel thread)
  3. trigger UE specific RX procedures for n if n is UL
  4. assemble `UL_INDICATION` and send to MAC
  5. wait for `SCHEDULE_REQUEST`
  6. do TX procedures if $n+4$ is TX and RX programming if $n+4+k$ is UL
- The MAC-layer timing is assumed to be
  1. do all UL processing for subframe n if n is UL after unraveling of UL_INDICATION in MAC module
  2. wait for call to `eNB_dlsch_ulsch_scheduler`
  3. do DL scheduling for $n+4$ if it is DL
  4. do UL scheduling for $n+8$ if it is UL
  5. return from eNB_dlsch_ulsch_scheduler
  6. let MAC module form SCHEDULE_REQUEST

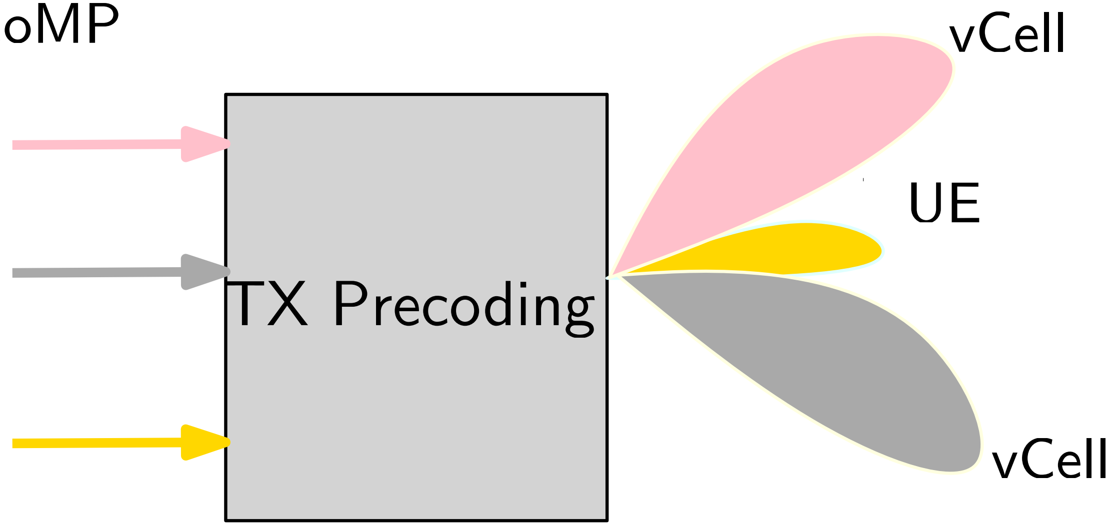
OAI MAC



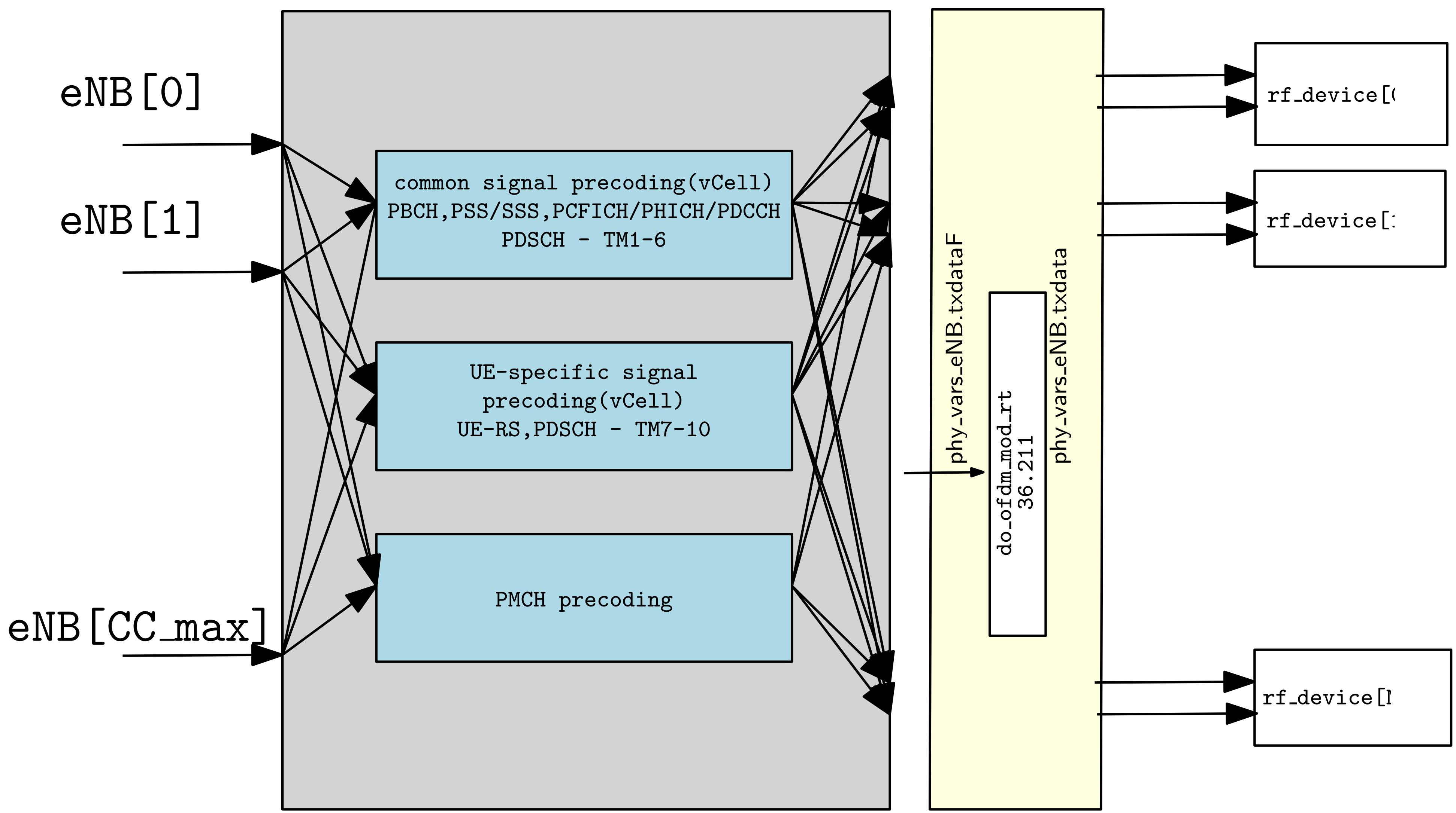
RRC
36.322
openair2/LAYER2/RLC
RLC
user customizable
scheduling module
Preprocessor
Random-Access
PHY Config
Scheduling
Events
36.321
openair2/LAYER2/MAC

TX Precoding

- Spatio-temporal filtering for muli-cell (vCell) and multi-user transmission. Input and output are frequency-domain signals.
- can be applied to Rel-10/11/12/13 physical channels and Rel-8 common channels
  - UE-specific precoding (TM7-10)
  - vCell-specific precoding (PDCCH + TM1-6) for groups of UEs
  - PMCH vCells
- Precoding applicable to
  1. indoor DAS
  2. outdoor co-localized arrays (e,g, Massive-MIMO)
  3. outdoor CoMP



TX Precoding (to RF device)



TX Precoding (to IF device, NGFI_IFv4p5)