



The Open-Source SDR LTE Platform for First Responders

**Software Radio Systems**

[www.softwareradiosystems.com](http://www.softwareradiosystems.com)

[www.github.com/srslte](http://www.github.com/srslte)

# Outline

---

- SRS - Software Radio Systems
- NIST PSIAP and OpenFirst
- srsLTE – The Open-Source LTE Software Suite

# Outline

---

- SRS - Software Radio Systems
- NIST PSIAP and OpenFirst
- srsLTE – The Open-Source LTE Software Suite

# SRS

---



Paul Sutton  
Director



Ismael Gomez  
Director



Andre Puschmann  
Senior Engineer



Linda Doyle  
Director



Justin Tallon  
Senior Engineer



Xavier Arteaga  
Senior Engineer



Bastian Bloessl  
Senior Engineer



Pedro Alvarez  
Senior Engineer





## Software Radio

A software radio is a radio system where components such as modulators/demodulators, filters, detectors etc. which have historically been implemented in hardware, are instead implemented in software upon a general-purpose processing platform. Software Radio Systems (SRS) provides high-performance software radio solutions for a range of wireless technologies, leveraging the low-cost and availability of general-purpose hardware to rapidly deliver powerful, flexible systems.

---



## Satellite systems

SRS is developing solutions for a number of satellite communications projects in partnership with the European Space Agency and international collaborators. We provide high-performance software libraries for technologies including DVB-S2 and DVB-RCS2 and integration solutions for hybrid satellite/terrestrial networks.

---



## LTE/ LTE-Advanced

SRS provides L1, L2 and L3 protocol stacks for UE and eNodeB. Our modular, portable srsLTE C library supports custom system design and development on a range of SDR platforms. Our tailored solutions include network test and diagnostic equipment and wide-band LTE scanners. We specialise in custom design and development for demanding applications and environments including machine to machine, airborne and high-speed deployments.

---



## Internet of Things

In the area of machine to machine communications, SRS develops custom low-cost, low-energy waveforms and network protocols. Our SDR basestation designs afford the flexibility to support a diverse range of application domains using a wide range of frequency bands. We specialise in cognitive radio and dynamic spectrum access solutions designed to benefit from evolving regulatory policies in this area.

---

srsLTE / srsLTE

Unwatch 112 Unstar 569 Fork 156

Code Issues 3 Pull requests 1 Projects 0 Wiki Settings Insights

Open source SDR LTE software suite Edit

Add topics

1,242 commits 3 branches 9 releases 16 contributors AGPL-3.0

Branch: master New pull request Create new file Upload files Find file Clone or download

 ismagom Fix memory alignment in PUCCH processing. Fixes #94 Latest commit 2dbc0fe 6 days ago

 cmake/modules Fixed incompatibility with volk1.2 Make pointer type warnings an error 3 months ago

 lib Fix memory alignment in PUCCH processing. Fixes #94 6 days ago

 srsenb Use runtime dir parameter for executables. 2 months ago

 srsue fix segfault in UE PHY tests 2 months ago

 CHANGELOG new release and changelog 3 months ago

 CMakeLists.txt check for compiler flag availability 3 months ago

 COPYRIGHT Updating notices a year ago

 CTestConfig.cmake Updating copyright notices and project name 3 years ago

 CTestCustom.cmake Added scrambling, ratematching and layer mapping tests 4 years ago

 LICENSE Changed license to AGPL 2 years ago

 README.md add apt-get line for Ubuntu to install required packages 3 months ago

 cmake\_uninstall.cmake.in Reorganized the directory structure. Added Graphics support. Added pr... 4 years ago

 README.md

## srsLTE

coverity passed

srsLTE is a free and open-source LTE software suite developed by SRS ([www.softwareradiosystems.com](http://www.softwareradiosystems.com)).

It includes:

- srsUE - a complete SDR LTE UE application featuring all layers from PHY to IP
- srsENB - a complete SDR LTE eNodeB application
- a highly modular set of common libraries for PHY, MAC, RLC, PDCP, RRC, NAS, S1AP and GW layers.

srsLTE is released under the AGPLv3 license and uses software from the OpenLTE project (<http://sourceforge.net/projects/openlte>) for some security functions and for RRC/NAS message parsing.



**MONROE**  
Measuring Mobile Broadband Networks in Europe

**OPEN CALL**

General objectives  
The MONROE project is the first year of the MONROE project. Both a mobile broadband and a mobile broadband MONROE testbed is the focus of the MONROE project. Both a mobile broadband and a mobile broadband MONROE testbed is the focus of the MONROE project. Both a mobile broadband and a mobile broadband MONROE testbed is the focus of the MONROE project. Both a mobile broadband and a mobile broadband MONROE testbed is the focus of the MONROE project.

Number of proposals in the first year up to 10  
Number of proposals per year: 10  
Number of partners per proposal: The target number of participants per proposal is minimum 2  
Number of proposals per year: 10  
Number of partners per proposal: The target number of participants per proposal is minimum 2  
Development of mobile broadband technologies, protocols and anchor experiments. The rules of participation are the same as for any H2020 proposal. The maximum allowed duration of each experiment is 16 Months.  
Language of the proposal: English  
Email address: [monroe@monroe-project.eu](mailto:monroe@monroe-project.eu)  
Call deadline: Tuesday, March 15, 2016 at 17:00 CET (Brussels time)  
Starting date: June, 2016

## “OSMOSIS” OPTIMISATION OF STREAMED MEDIA OVER SATELLITE INFRASTRUCTURES



FACILITY PROJECTS

### FLEX

Flex will establish LTE testbeds for open experimentation aims at improving the performance of mobile broadband in FIBRE, a multi-carrier puzzle cellular access technologies in LTE and Long-Term Evolution (LTE). FLEX's experimentation environment will feature both open and closed loop control and management. The first year of the project will span macro-cell, pico-cell and small-cell setups. FLEX will build upon current LTE control and management and extend it to support the new requirements imposed by the deployment for the new LTE components, and will develop specialised tools for the optimisation of the system. FLEX will also support mobility, with the establishment of both real and emulated mobility functionalities on the testbeds. FLEX will organise and manage the testbeds and will provide a general environment for open source experiments, test innovative services or provide functional extensions of LTE testbeds.

**KEY ACHIEVEMENTS/RESULTS**  
• The three testbed sites are already online and operating.  
• The LTE resources are smoothly integrated in the FIBRE infrastructure.  
• Frameworks and monitoring applications using the LTE equipment are given to the partners.  
• The first Open Call for proposals has been successfully about FLEX by disseminating the project in key events.

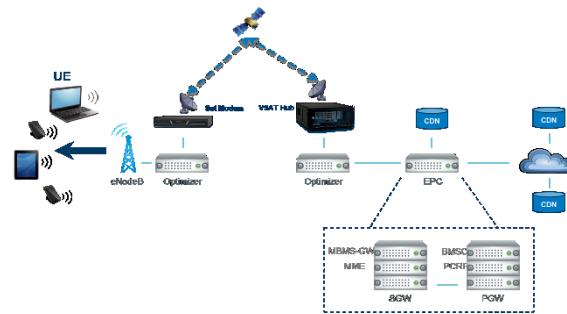
**HOW TO GET INVOLVED?** [www.flex-project.eu](http://www.flex-project.eu)

The FIBRE portal can be reached at [www.flex-project.eu](http://www.flex-project.eu). The FIBRE testbeds are open to experiments and use the infrastructure is included. FLEX has already organized one Open Call for proposals, and a second one is expected to be organized in the second half of the year. The Open Calls are open for innovative usages of the deployed facilities, orchestrated by the partners and the testbeds' operators and their customers. The calls have been planned to take place early in order to provide enough time for the new partners to be integrated in the consortium and provide meaningful contributions.

**PROJECT FACTS**  
COORDINATOR: Prof. Antonios Karalis, University of Thessaly  
PARTNERS: University of Thessaly (General Coordinator), University of Patras, University of Ioannina, University of Ioannina, University of Piraeus, Aristotle Univ. of Thessaloniki, COSMOTE (Greece), Rutgers - The state university of New Jersey (US), NICTA (Australia).



The FIBRE testbeds are open to experiments and use the infrastructure is included.



**WiSHFUL**

**WISHFUL Portable Testbed**

The portable Testbed has been demonstrated at several events (ETSI RRS workshop, EuCNC 2016, Global 5G 2016, and is now used by experimenters in Open Calls. Next demo during WiSHFUL tutorial at Crowncom 2017 (Increasing spectrum efficiency through intelligent and coordinated radio and network control).

Feb  
15  
2017



**SMARTSKY**  
NETWORKS

## SRS partners with SmartSky Networks to deliver true 4G inflight connectivity

Software Radio Systems (SRS) today announced a strategic partnership with SmartSky Networks, a high-performance air-to-ground connectivity network operator, in which SRS will provide test and validation solutions for SmartSky's airborne products. Based on aviation-specific modifications to 4G wireless communications standards, SmartSky 4G delivers affordable and reliable office-like connectivity in the air. As a leading provider...

[Details >>](#)

15th February 2017 / Press / By Paul Sutton

SILICONREPUBLIC

BUSINESS DISCOVERY CAREERS LIFE VIDEO MORE



COMMS

## Cork tech company to deliver 4G in the sky at Mobile World Congress

by John Kennedy

24 FEB 2017 133 SHARES

# Outline

---

- SRS - Software Radio Systems
- NIST PSIAP and OpenFirst
- srsLTE – The Open-Source LTE Software Suite

# Public Safety Innovation Accelerator Program



**National Institute of Standards and Technology**  
Technology Administration, U.S. Department of Commerce

## Welcome to the PSCR Program

PSCR provides research, development, testing, and evaluation to foster nationwide interoperability and advanced communications technology for the nation's public safety community



[Home](#)   [FirstNet Board](#)   [The Network](#)   [Consultation](#)   [Newsroom](#)   [Resources](#)   [About](#)

**PSCR's Innovation Accelerator Program Announces \$30 million Grant Program to 'Pull the Future Forward' for Public Safety**

# Public Safety Innovation Accelerator Program



Search NIST

NIST MENU

NEWS

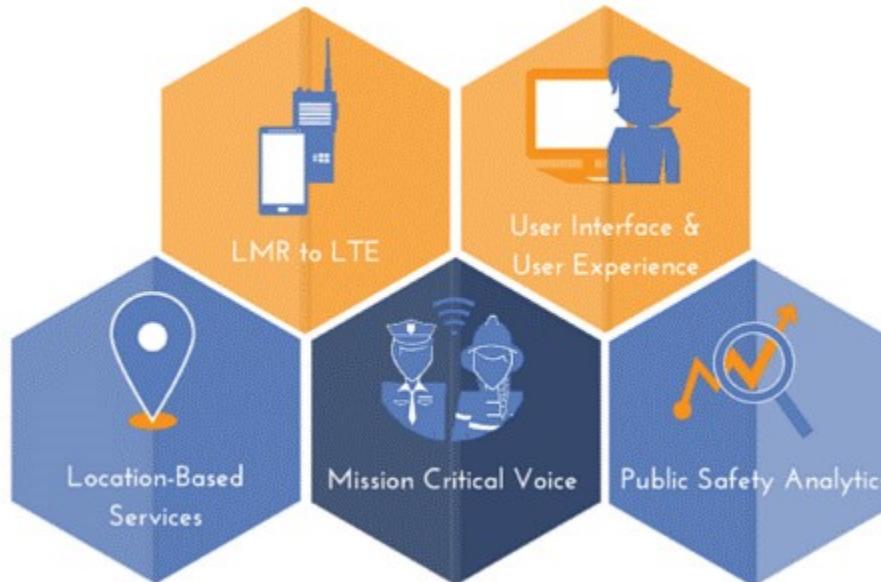
## NIST Awards \$38.5 Million to Accelerate Public Safety Communications Technologies

June 13, 2017

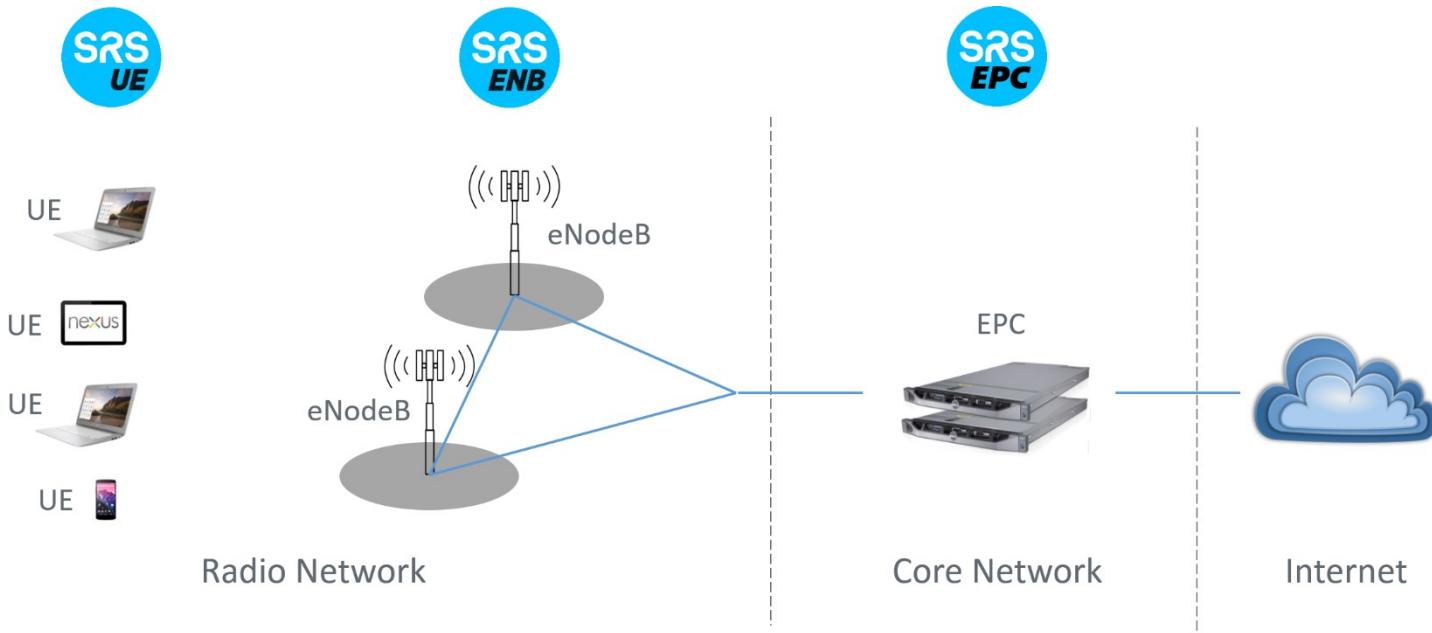
BOULDER, Colo. – The U.S. Commerce Department's National Institute of Standards and Technology (NIST) has awarded \$38.5 million to 33 research and development (R&D) projects aimed at advancing broadband communications technologies for first responders.



MEDIA CONTACT

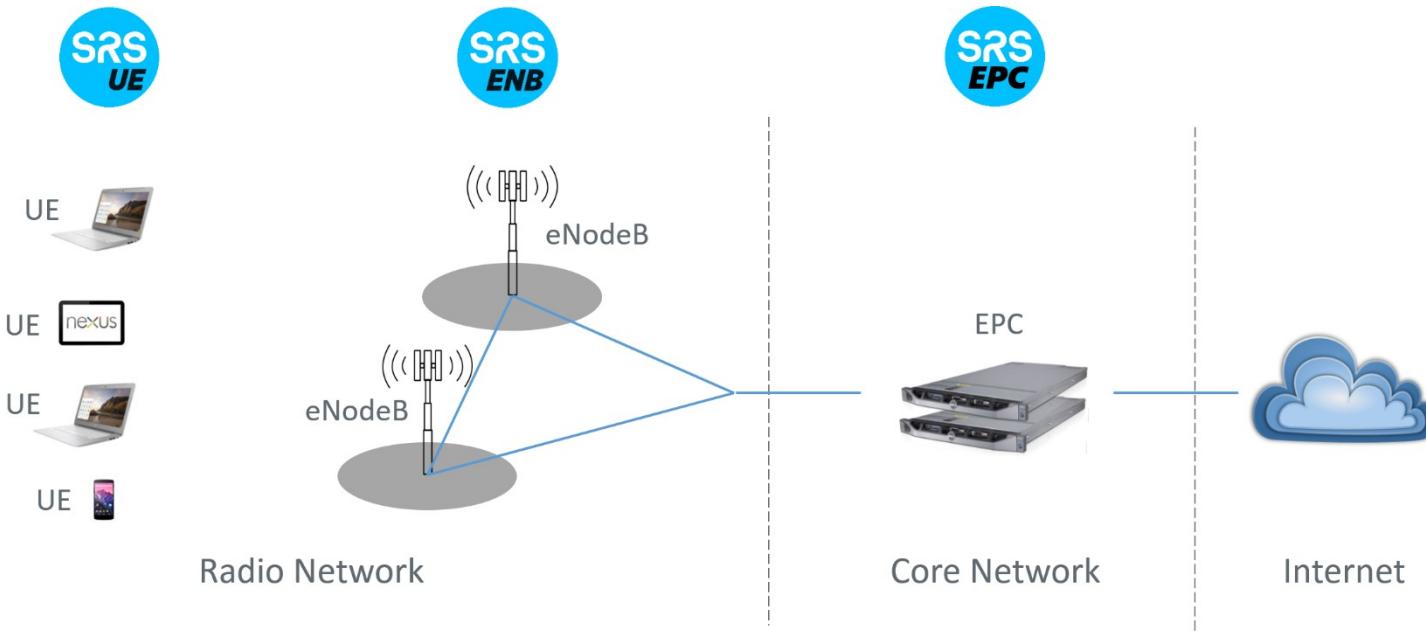


An open-source end-to-end LTE network for public safety research & development.



- A reference implementation of key LTE features for first responders.
- Enabling, supporting and growing the public safety broadband development ecosystem.
- Providing a commercialization path for public safety LTE using proven business models.
- Building upon the proven srsLTE suite of open-source libraries, tools and applications.

An open-source end-to-end LTE network for public safety research & development.



#### Key Requirements:

- Ease of use / Ease of programming new capabilities
- Clarity and completeness of documentation
- Long-term sustainability / Development ecosystem
- Availability for follow-on research / Potential for commercialization

#### R&D Platform Requirements:

- Sufficiently complete implementation / Interoperability
- Sufficient set of components
- Performance comparable to likely operational implementations
- Network scale
- Path towards operational implementation

SrsLTE / srsLTE

Unwatch 98    ★ Unstar 516    Fork 128

Code Issues 6 Pull requests 0 Projects 0 Wiki Settings Insights

Open source 3GPP LTE library

Add topics

1,223 commits 2 branches 9 releases 16 contributors AGPL-3.0

Branch: master New pull request Create new file Upload files Find file Clone or download

**ismagom** forced local variable alignment in dot\_prod\_sss\_avx2 Latest commit d0ee9f9 an hour ago

**cmake/modules** Incompatibility with volk1.2 Make pointer type warnings an error 3 hours ago

**lib** forced local variable alignment in dot\_prod\_sss\_avx2 an hour ago

**srsenb** set default RRC timeout to 30s 21 hours ago

**srsue** fixed some issues with AVX machines 2 hours ago

**CHANGELOG** new release and changelog 22 hours ago

**CMakeLists.txt** check for compiler flag availability 3 hours ago

**COPYRIGHT** Updating notices 9 months ago

**CTestConfig.cmake** Updating copyright notices and project name 2 years ago

**CTestCustom.cmake.in** Added scrambling, ratematching and layer mapping tests 3 years ago

**LICENSE** Changed license to AGPL 2 years ago

**README.md** Edited readme 21 hours ago

**cmake\_uninstall.cmake.in** Reorganized the directory structure. Added Graphics support. Added pr... 3 years ago

README.md

## srsLTE

covertly passed

srsLTE is a free and open-source LTE software suite developed by SRS ([www.softwareradiosystems.com](http://www.softwareradiosystems.com)).

It includes:

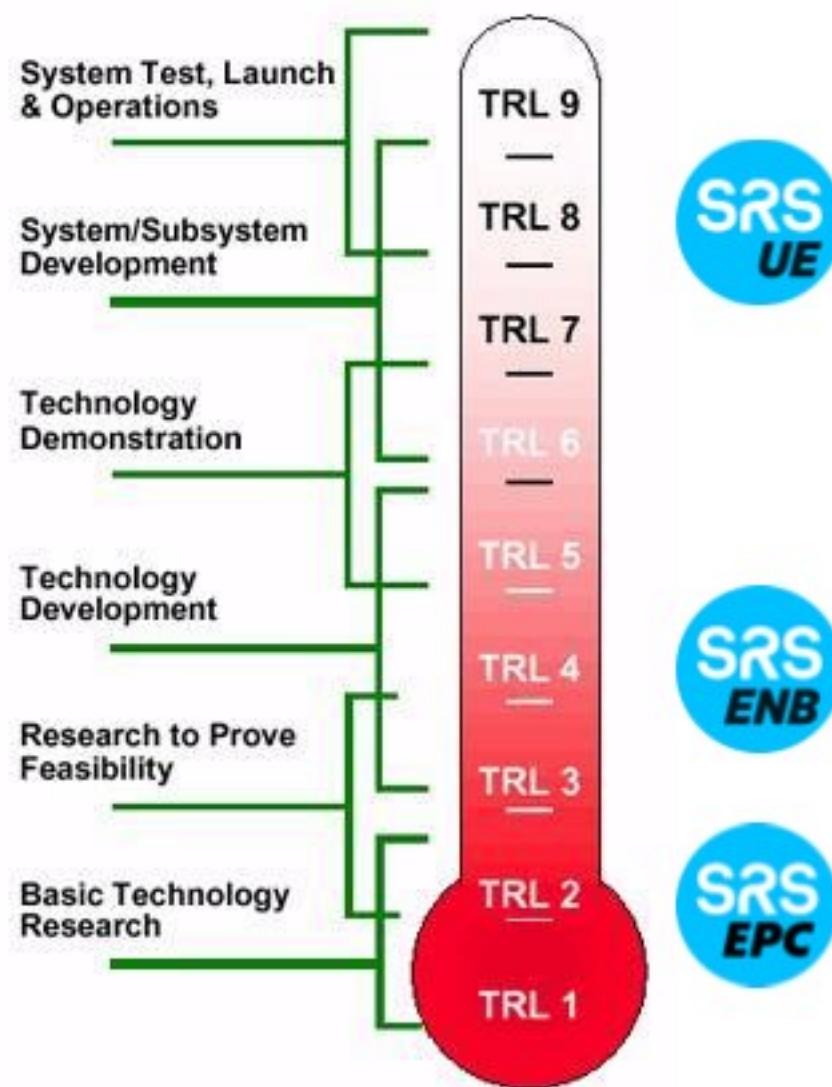
- srsUE - a complete SDR LTE UE application featuring all layers from PHY to IP
- srsENB - a complete SDR LTE eNodeB application
- a highly modular set of common libraries for PHY, MAC, RLC, PDCP, RRC, NAS, S1AP and GW layers.

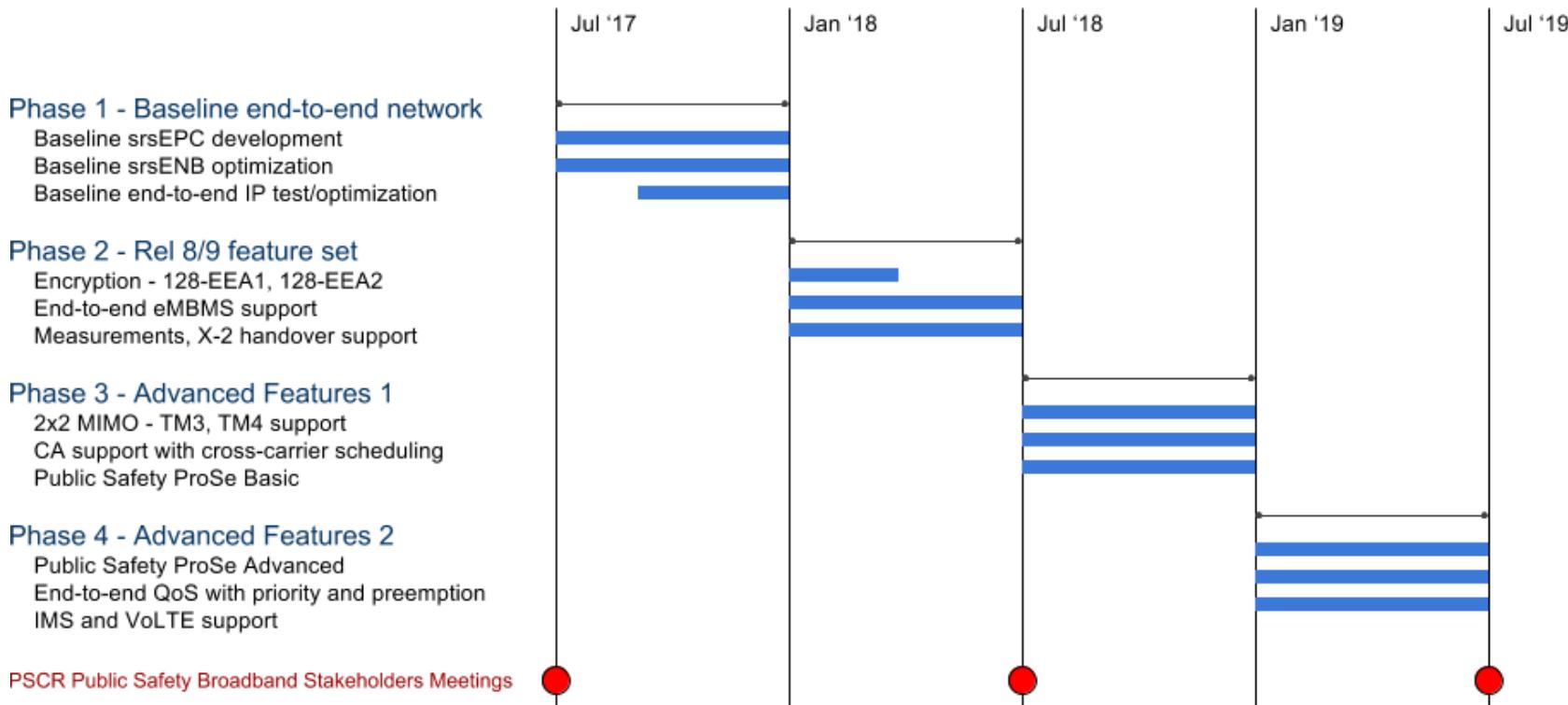
srsLTE is released under the AGPLv3 license and uses software from the OpenLTE project (<http://sourceforge.net/projects/openlte>) for some security functions and for RRC/NAS message parsing.



- GNU Affero General Public License (AGPLv3)
- Ensuring dissemination of the technology
- Maximizing usability
- Safeguarding availability
- Guaranteeing sustainability

[www.github.com/srslte](http://www.github.com/srslte)





### Complete end-to-end LTE network:

- Core network (EPC)
- Basestations (eNodeBs)
- Mobile terminals (UEs)

### Key features for public safety:

- Priority, Pre-emption and Quality of Service (QPP).
- Proximity Services (ProSe)
- Evolved Multimedia Broadcast/Multicast Service (eMBMS)
- Multimedia broadcast single frequency networks (MBSFN)
- Single cell point-to-multipoint (SC-PTM).
- Carrier aggregation.
- IP Multimedia Subsystem (IMS) and Voice-over-LTE (VoLTE)

# Outline

---

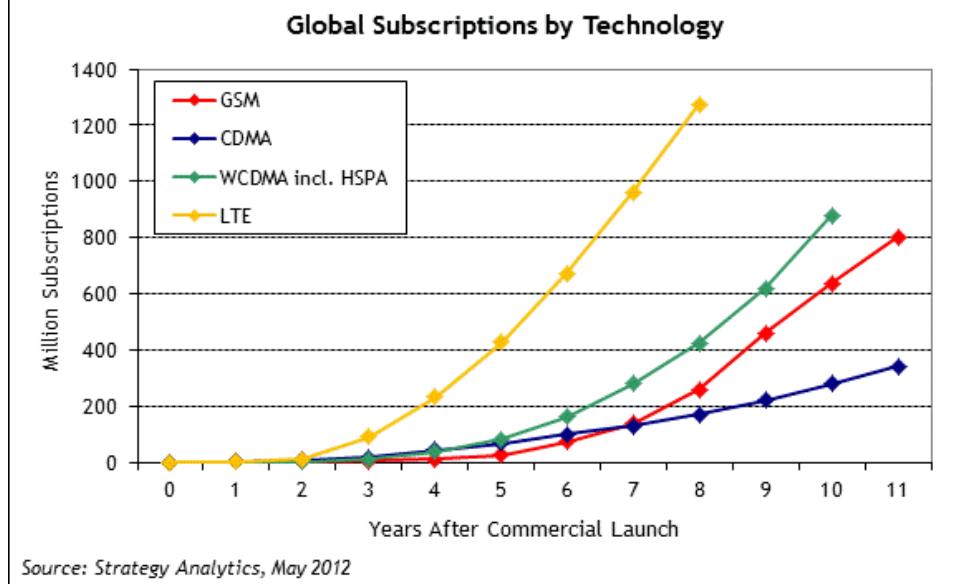
- SRS - Software Radio Systems
- NIST PSIAP and OpenFirst
- **srsLTE – The Open-Source LTE Software Suite**

By the end of March 2017 GSA reports there were:

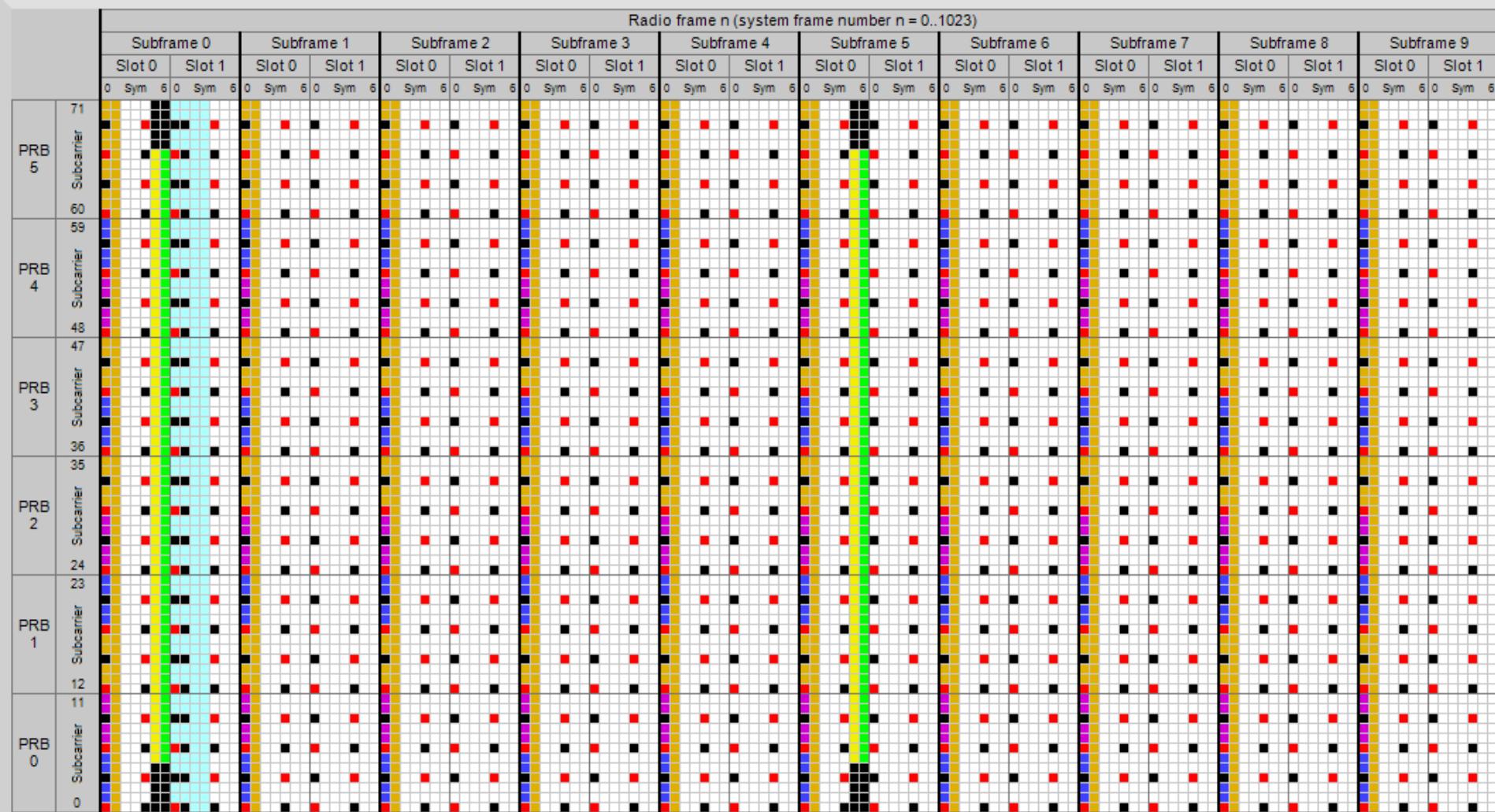
- **774** operators investing in LTE in **202** countries
- **591** commercially launched LTE or LTE-Advanced networks in **189** countries (GSA forecasts c. **646** commercial LTE networks by end-2017)
- **97** LTE-TDD (TD-LTE) networks launched in **56** countries
- **106** commercial VoLTE networks in **55** countries, and **167** operators investing in VoLTE in **74** countries
- **195** launched LTE-Advanced or LTE-Advanced Pro networks, in **95** countries
- **4** NB-IoT and **2** LTE-M networks commercially launched, with **40** NB-IoT and **12** LTE-M networks planned or being trialled
- **18** operators (at least) that have made public commitments to deploy pre-standards '5G' networks, in **13** countries

\* Taken from the GSA report *Evolution from LTE to 5G April 2017*. To download the full report, visit [www.gsacom.com](http://www.gsacom.com)

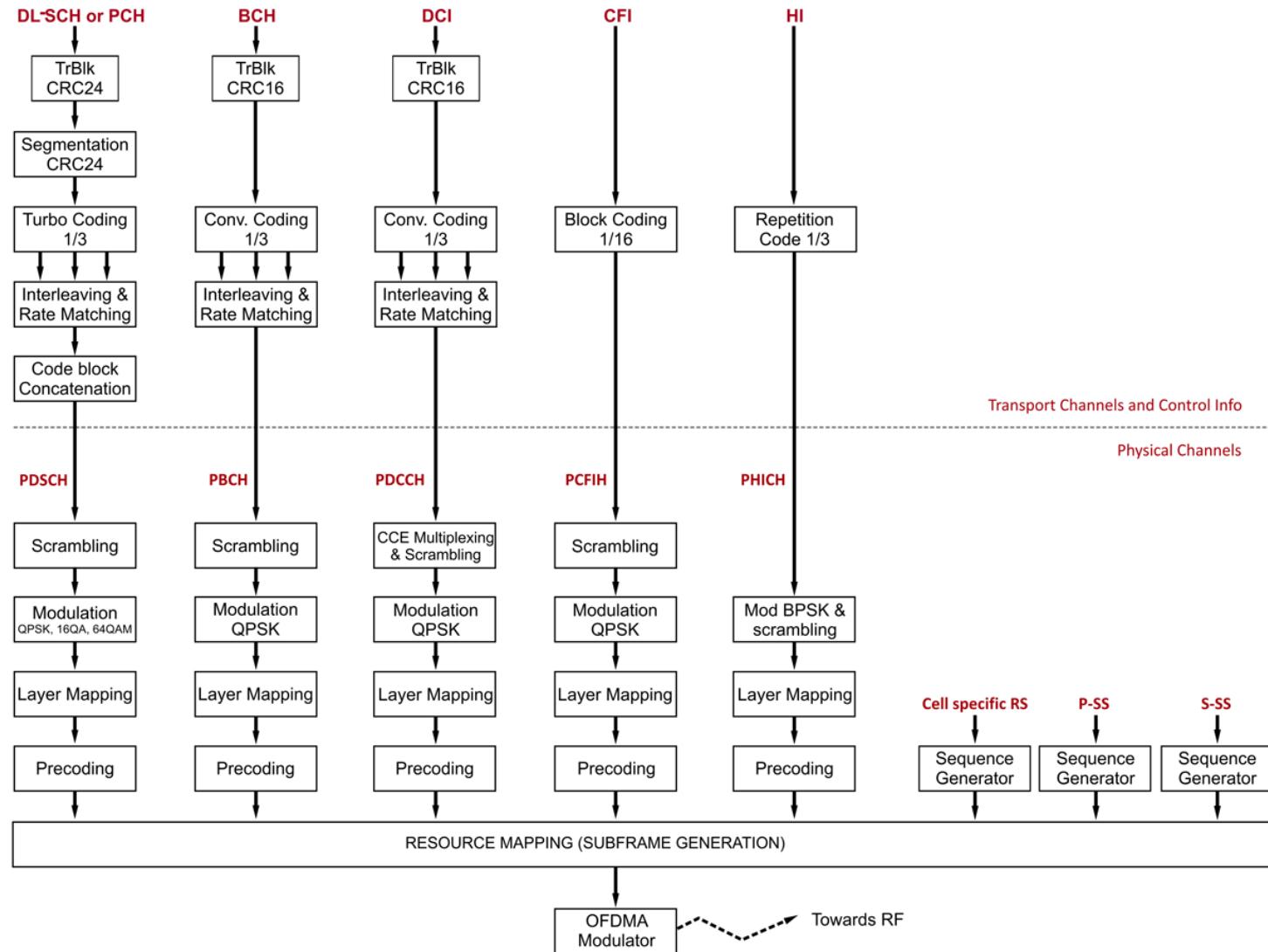
© 2017 Global mobile Suppliers Association



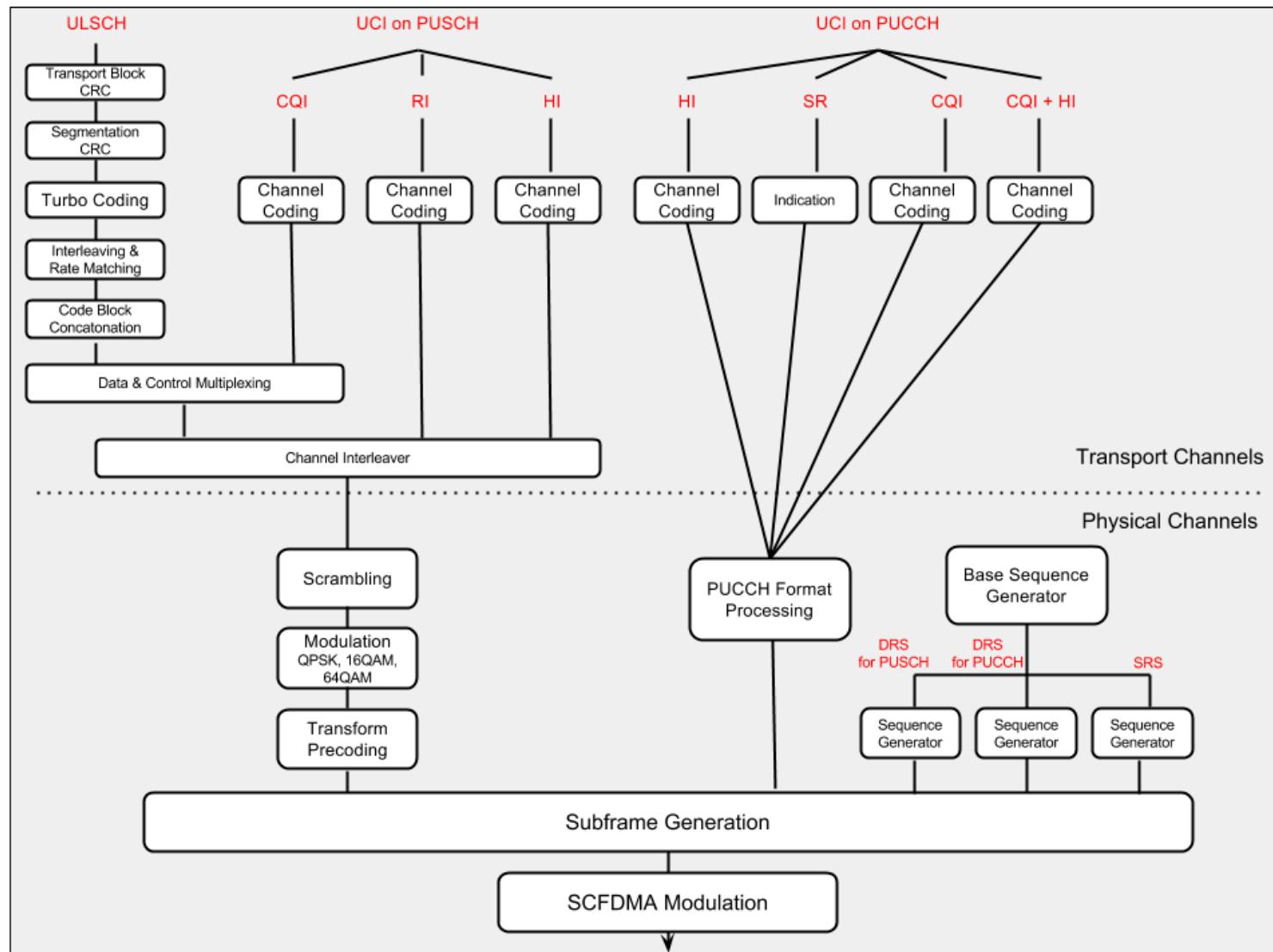
[Color Box]	PSCH (Primary Synchronization Channel)
[Color Box]	SSCH (Secondary Synchronization Channel)
[Color Box]	PBCH (Physical Broadcast Channel)
[Color Box]	RS (cell-specific Reference Signal) for selected Tx antenna port
[Color Box]	Reserved for TDD uplink
[Color Box]	Unused by selected Tx antenna port, or undefined for all ports
[Color Box]	MBSFN (Multicast/Broadcast over Single Frequency Network) region - available for PMCH (Physical Multicast Channel)
[Color Box]	PCFICH (Physical Control Format Indicator Channel)
[Color Box]	PHICH (Physical Hybrid ARQ (Automatic Repeat reQuest) Indicator Channel)
[Color Box]	PDCCH (Physical Downlink Control Channel)
[Color Box]	Available for PDSCH (Physical Downlink Shared Channel)
[Color Box]	TDD guard period in special subframe



# LTE Downlink Transmitter (eNodeB-side)

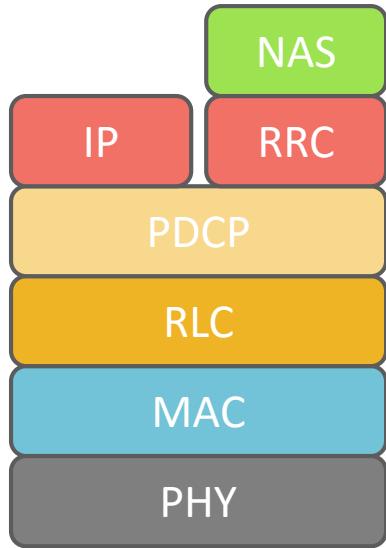


# LTE Uplink Transmitter (UE-side)



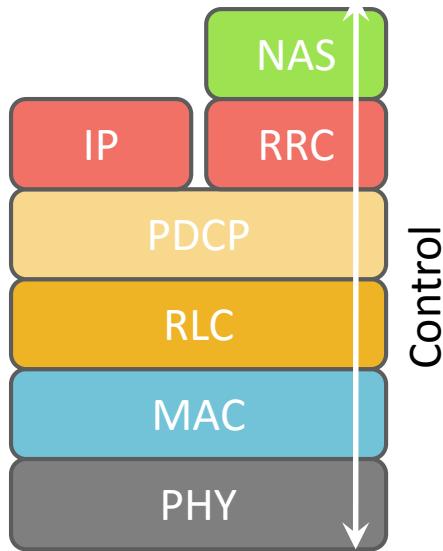
# LTE Overview – protocol stacks

---



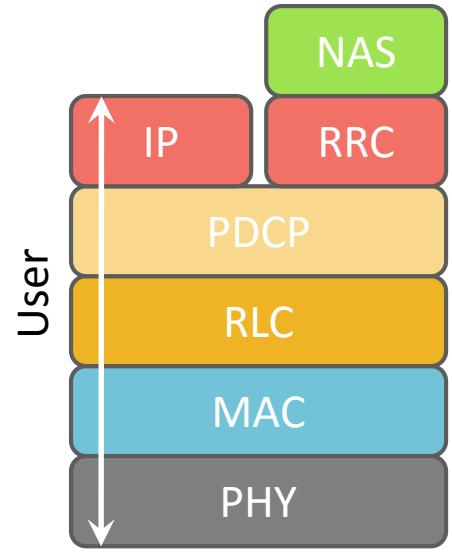
# LTE Overview – protocol stacks

---

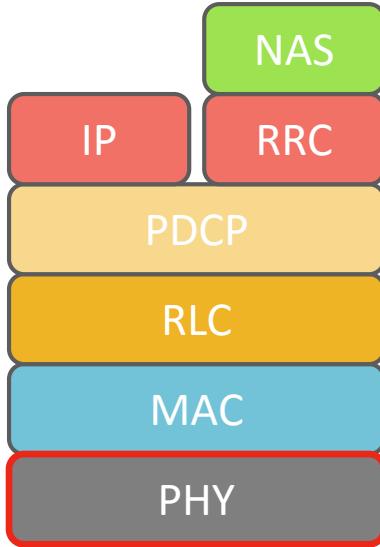


# LTE Overview – protocol stacks

---

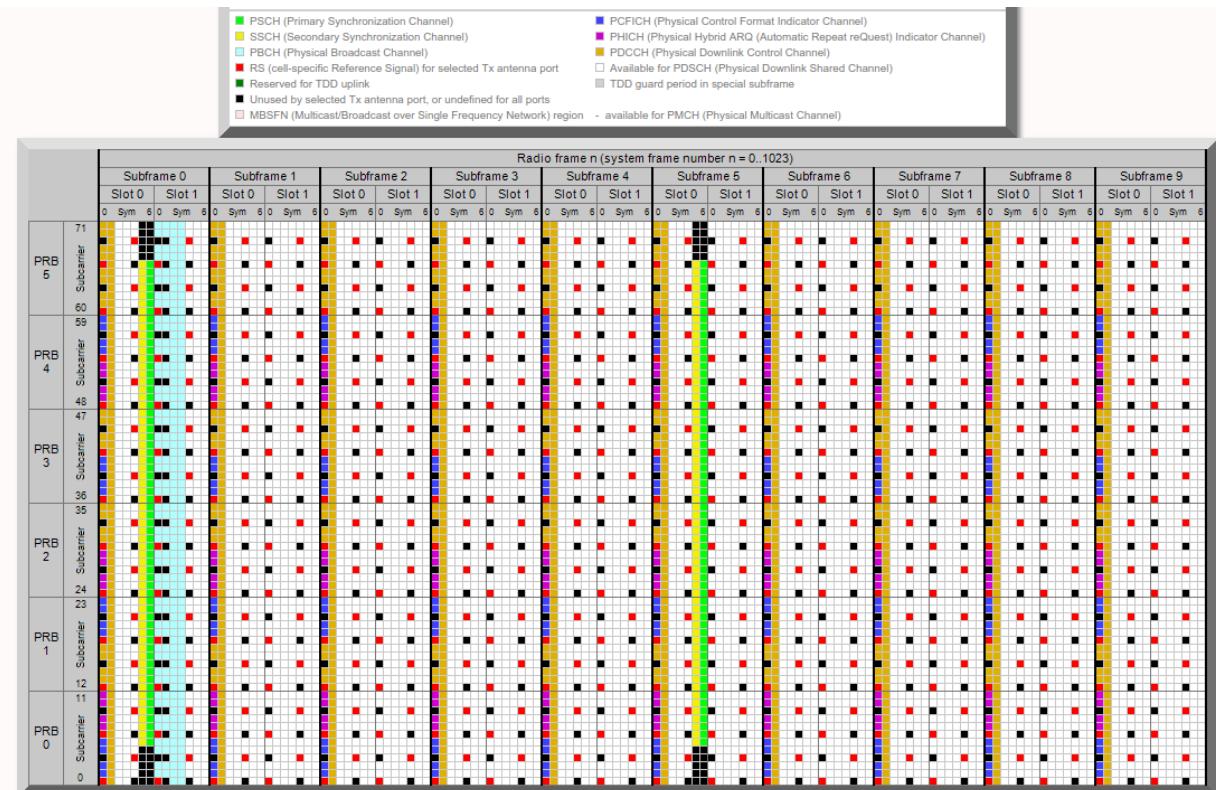


# LTE Overview – protocol stacks

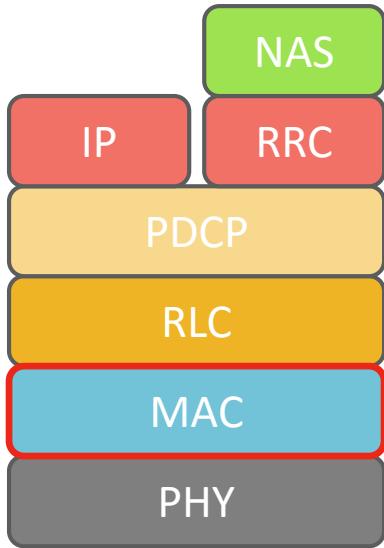


## Physical Layer

- Tx/Rx across air interface
- Very flexible, many multiplexed signals
- OFDM / SC-FDMA



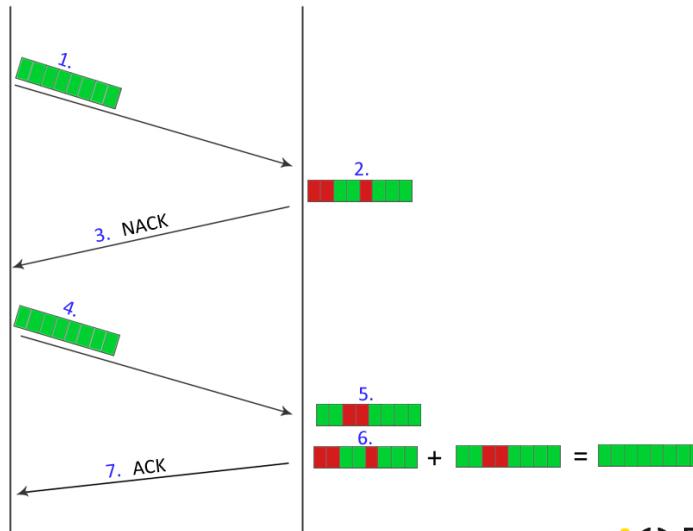
# LTE Overview – protocol stacks



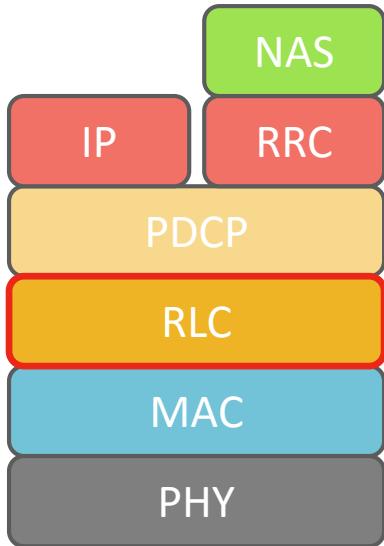
## Medium Access Control Layer

- Prioritize and multiplex logical channel data
- Scheduling
- Link adaptation
  - modulation scheme and coding rate
  - transport block size

Hybrid Automatic Repeat reQuest (HARQ)



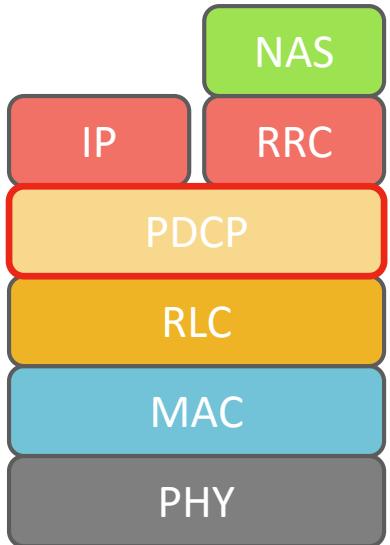
# LTE Overview – protocol stacks



## Radio Link Control Layer

- Segment and concatenate packets
- Error correction through ARQ
- In-sequence packet delivery
- Three modes:
  - Transparent (TM)
  - Unacknowledged (UM)
  - Acknowledged (AM)

# LTE Overview – protocol stacks

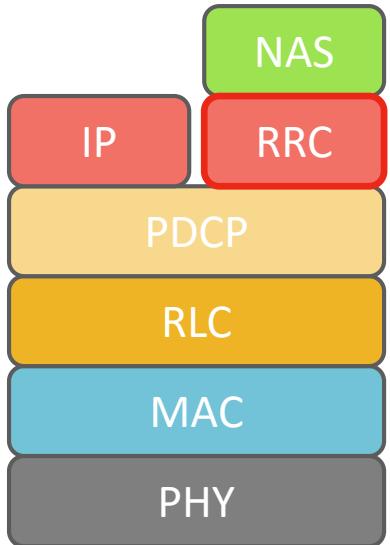


## Packet Data Convergence Protocol Layer

- IP packet header compression (RoHC)
- Ciphering of control and data plane traffic
- Integrity protection of control plane traffic

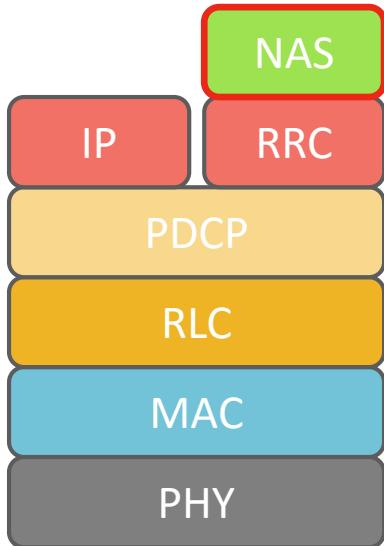
# LTE Overview – protocol stacks

## Radio Resource Control Layer



- Control plane functions:
  - Paging
  - Management of RRC connection with eNodeB
  - Mobility management
  - QoS management

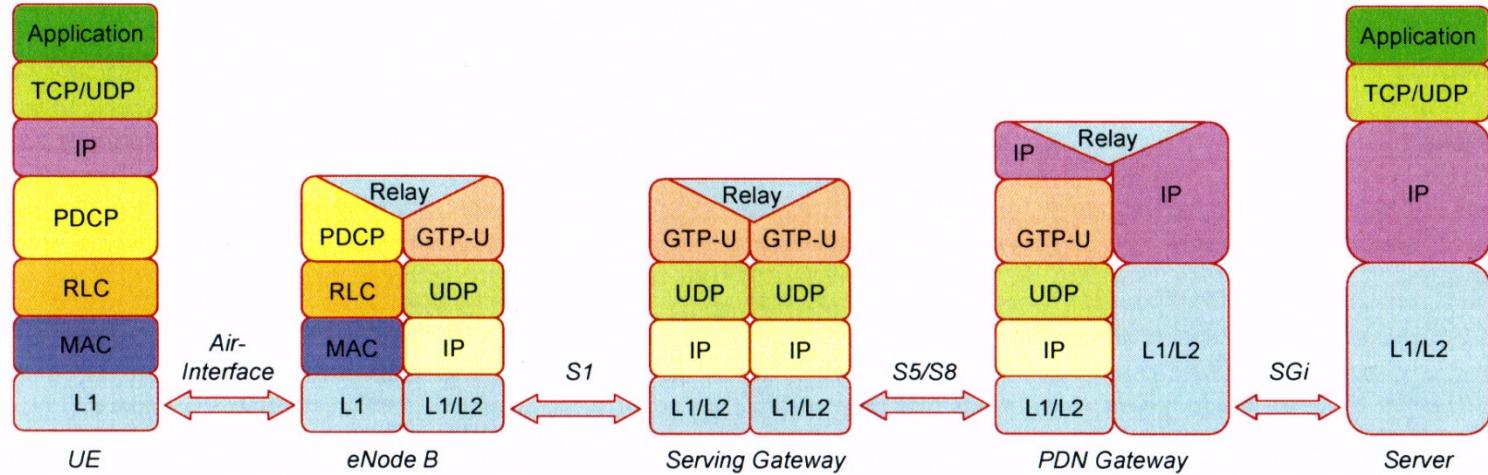
# LTE Overview – protocol stacks



## Non-Access Stratum Layer

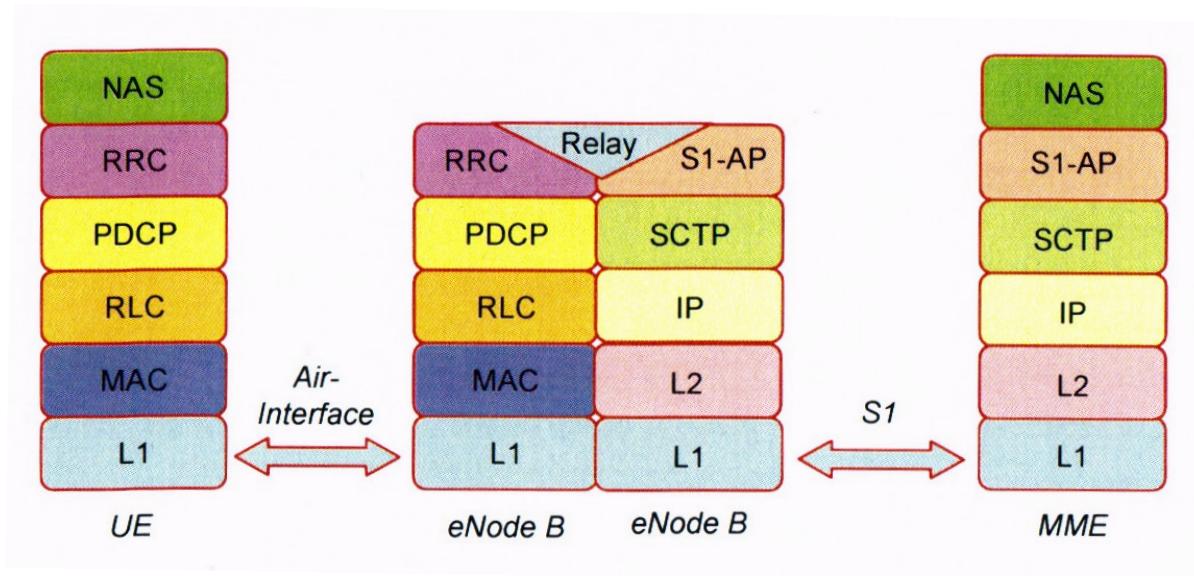
- Control plane connection UE <-> MME:
  - Network attach/detach
  - Authentication
  - Security mode management
  - Identity management
  - Tracking area updates
  - Bearer management
  - PDN connectivity management

# LTE Overview – protocol stacks



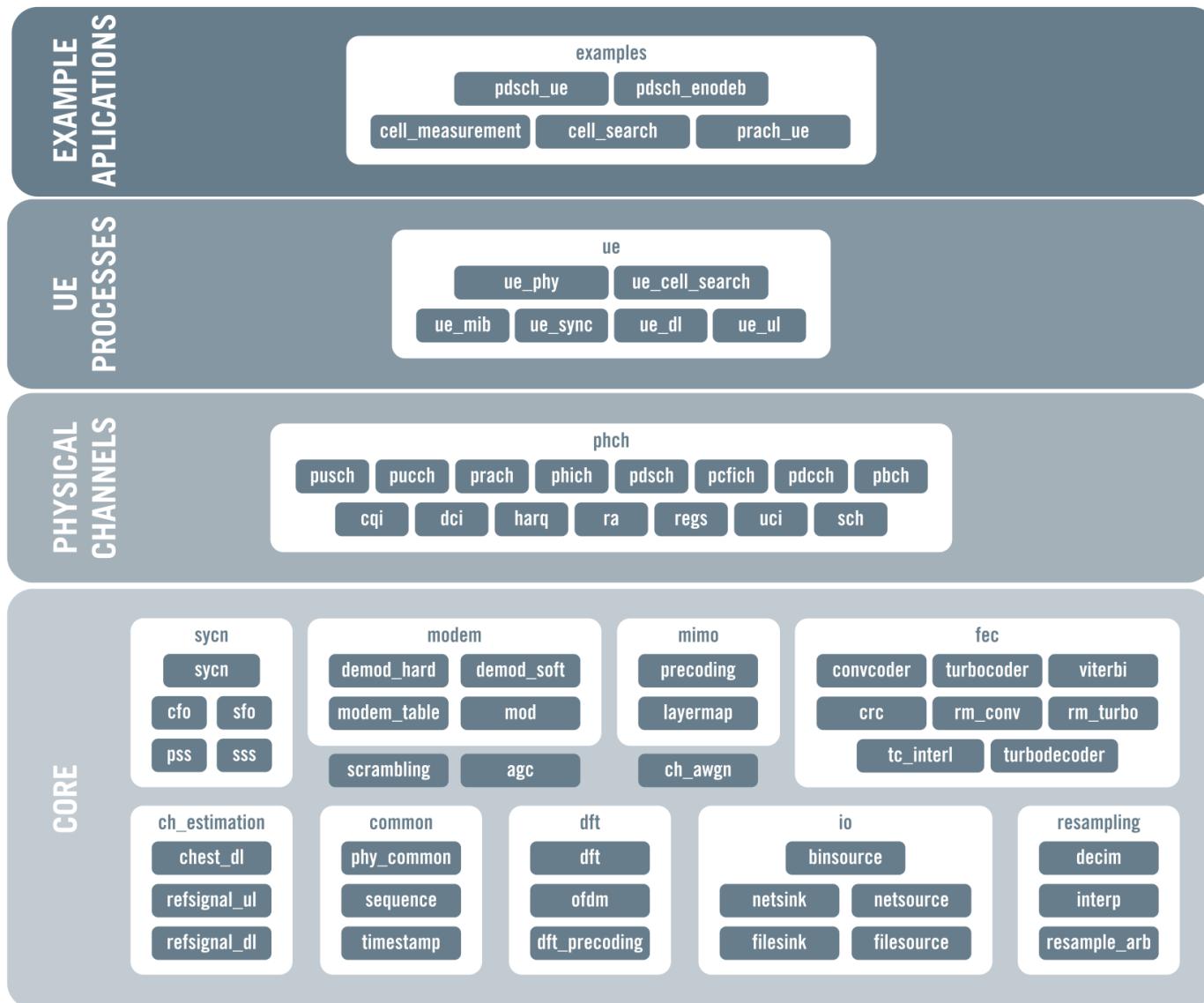
## User plane protocol stacks

# LTE Overview – protocol stacks

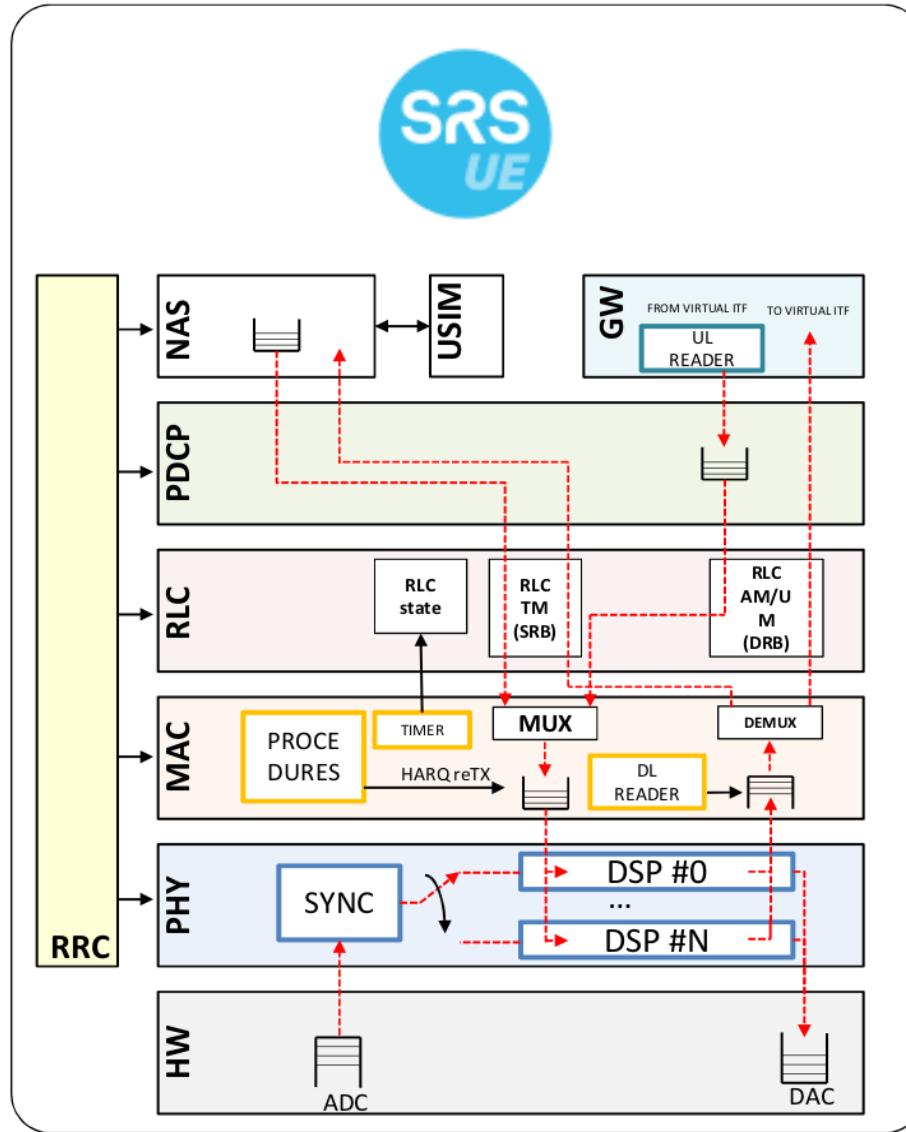


**Control plane protocol stacks**

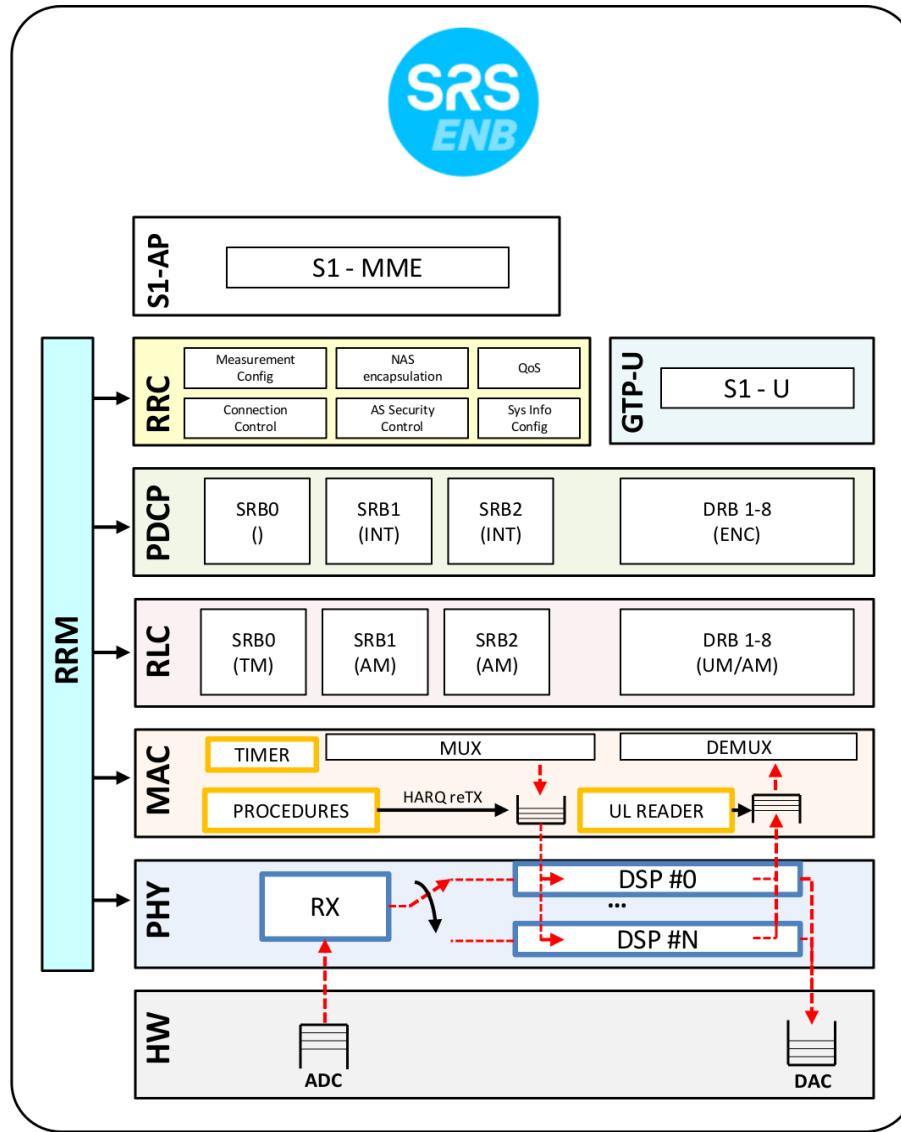
# PHY-layer DSP library - srsLTE



# UE application - srsUE



# eNodeB application - srsENB



# Class/Layer Design

---

- srsUE/srsENB written in C++, srsLTE in C
- Each layer (PHY, MAC, RLC, PDCP, GW) is implemented in a single class
  - Some complex layers use auxiliary sub-classes
- Each layer provides a separate clean C++ pure virtual interface to any other class that make use of it (e.g. passing messages/data between layers)
- Threads only for performance or priority management reasons

# Processing Latency Constraint in LTE

---

In LTE the basic time unit is 1 ms = 1 subframe

The processing latency constraint or *critical time* is **4 ms**, given by:

$N$	$N+1$	$N+2$	$N+3$	$N+4$
-----	-------	-------	-------	-------

- a) N: Reception of DL grant through PDCCH  
N: Decoding of PDSCH  
N+4: Transmission of ACK/NACK HARQ through PUCCH

# Processing Latency Constraint in LTE

---

In LTE the basic time unit is 1 ms = 1 subframe

The processing latency constraint or *critical time* is **4 ms**, given by:

$N$	$N+1$	$N+2$	$N+3$	$N+4$
-----	-------	-------	-------	-------

- b) N: Reception of UL grant or NACK HARQ  
N+4: (re)-Transmission of PUSCH

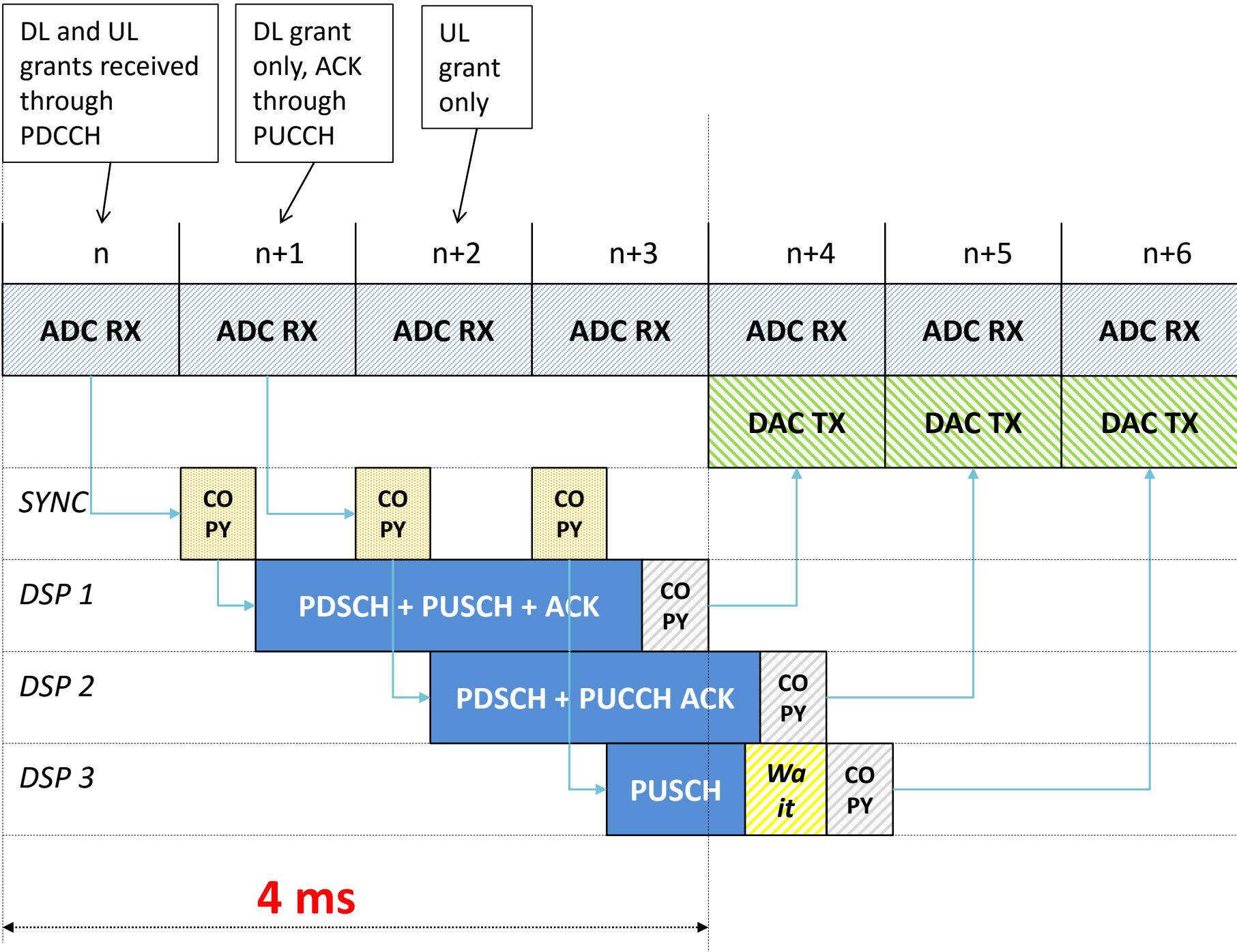
# Processing Latency Constraint in LTE

BUT worse case happens in case of a union of a) and b), then:

$N$	$N+1$	$N+2$	$N+3$	$N+4$
-----	-------	-------	-------	-------

- a+b)
  - N: Reception of DL grant + UL grant or NACK HARQ
  - N: Decoding of PDSCH
  - N+4: (re)-Transmission of PUSCH + ACK/NACK HARQ

In this case, we have 4 ms to receive samples from ADC, decode PDSCH, encode PUSCH and transmit samples to ADC in time.

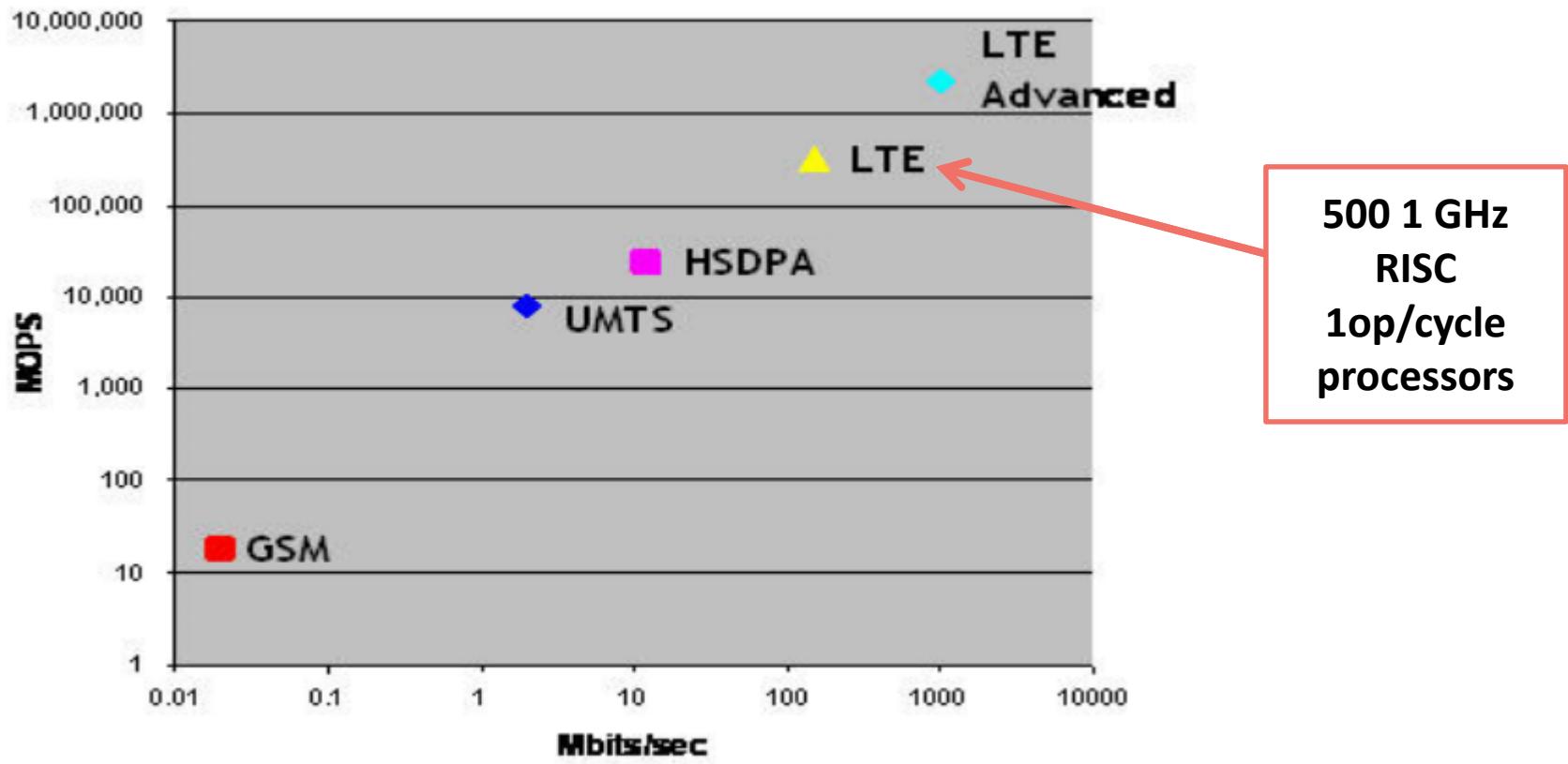


# Some Considerations on Threading

---

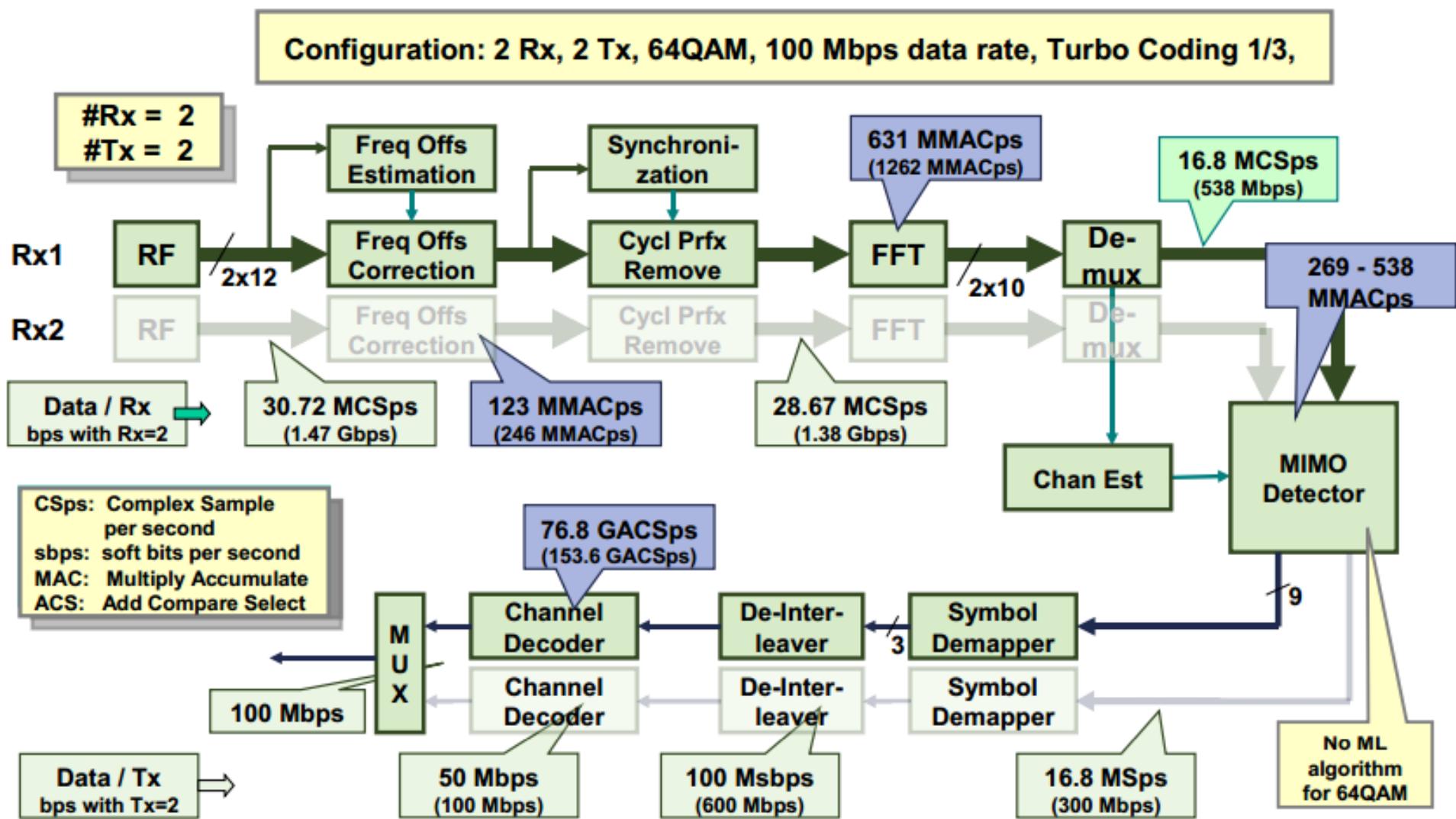
- The maximum useful pipeline depth is 3 stages (3 threads).
- Dividing uplink and downlink in two threads is also inefficient because uplink thread has to wait for downlink thread (i.e. there is no parallelization gain!)
- If more cores are available, we may divide each DSP thread and process multiple streams or codeblocks in parallel.
- Breakdown of the 4 ms deadline:
  - 1.0 ms for RX buffering
  - 0.5 ms for USRP -> Host transport
  - 2.0 ms left for processing
  - 0.5 ms for Host -> USRP transport

# Processing Requirements



Abhijit Shah, "A MultiCore Design Approach for LTE PHY," Tensilica, Inc.

# Processing Requirements



Source: "MIMO-LTE A relevant step towards 4G", Thomas Kaiser, MobiMedia, August 27-29, 2007

## 2 essential tools: SIMD and Precompute

- SIMD intrinsics wherever possible
  - e.g. integer arithmetic DSP, turbo decoder, soft demodulator, etc.
  - At compile time, choose generic/SSE4/AVX
- Initially target GPPs, so assume memory is almost free
- Use LUTs or pregenerate signals extensively:
  - Scrambling sequences for each subframe
  - PUCCH signals for each subframe
  - DL/UL reference signals for each subframe
  - CRC
  - Rate matching interleaver
  - ...

## DSP optimizations

---

- We use libfftw, which is fast enough even with non power of 2 sizes. This allows  $\frac{3}{4}$  sampling rates:
  - 10 MHz BW: FFT 768 samples, 11.52 Msamples/s
  - 20 MHz BW: FFT 1536 samples, 23.04 Msamples/s
- This constrains us to use 32-bit complex float for transport and FFT processing.
- Yet to find a good open source integer FFT library...

# Turbo Decoder

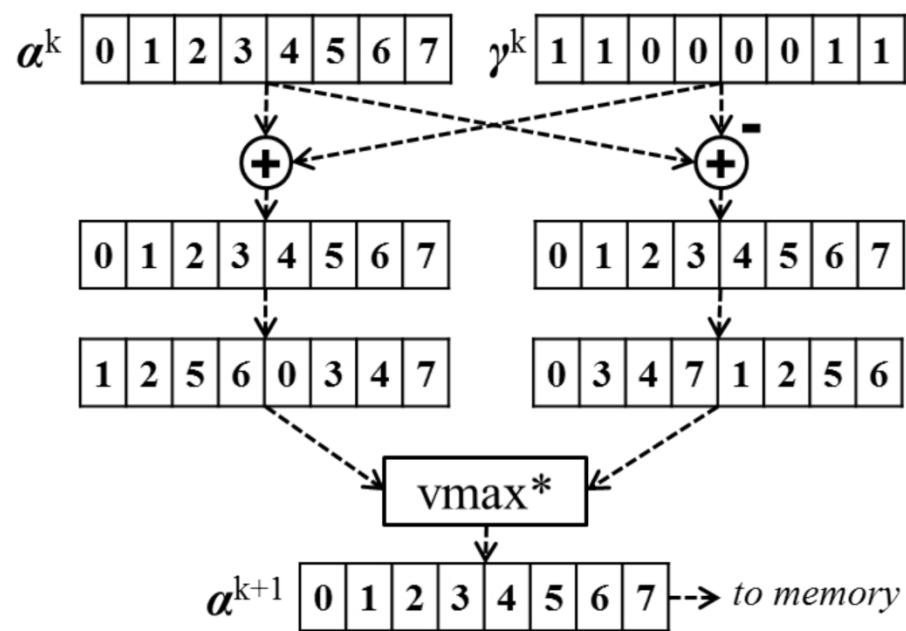
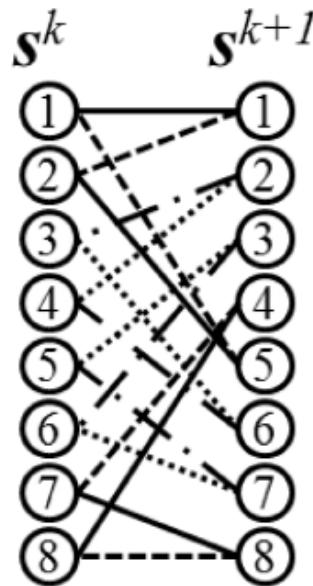
- The Turbo Decoder is the most demanding component:

TABLE I  
TOTAL PDSCH RECEIVER PROCESSING TIME AND BREAK-DOWN OF THE  
CPU UTILIZATION FOR 20 MHZ BANDWIDTH CONFIGURATION.

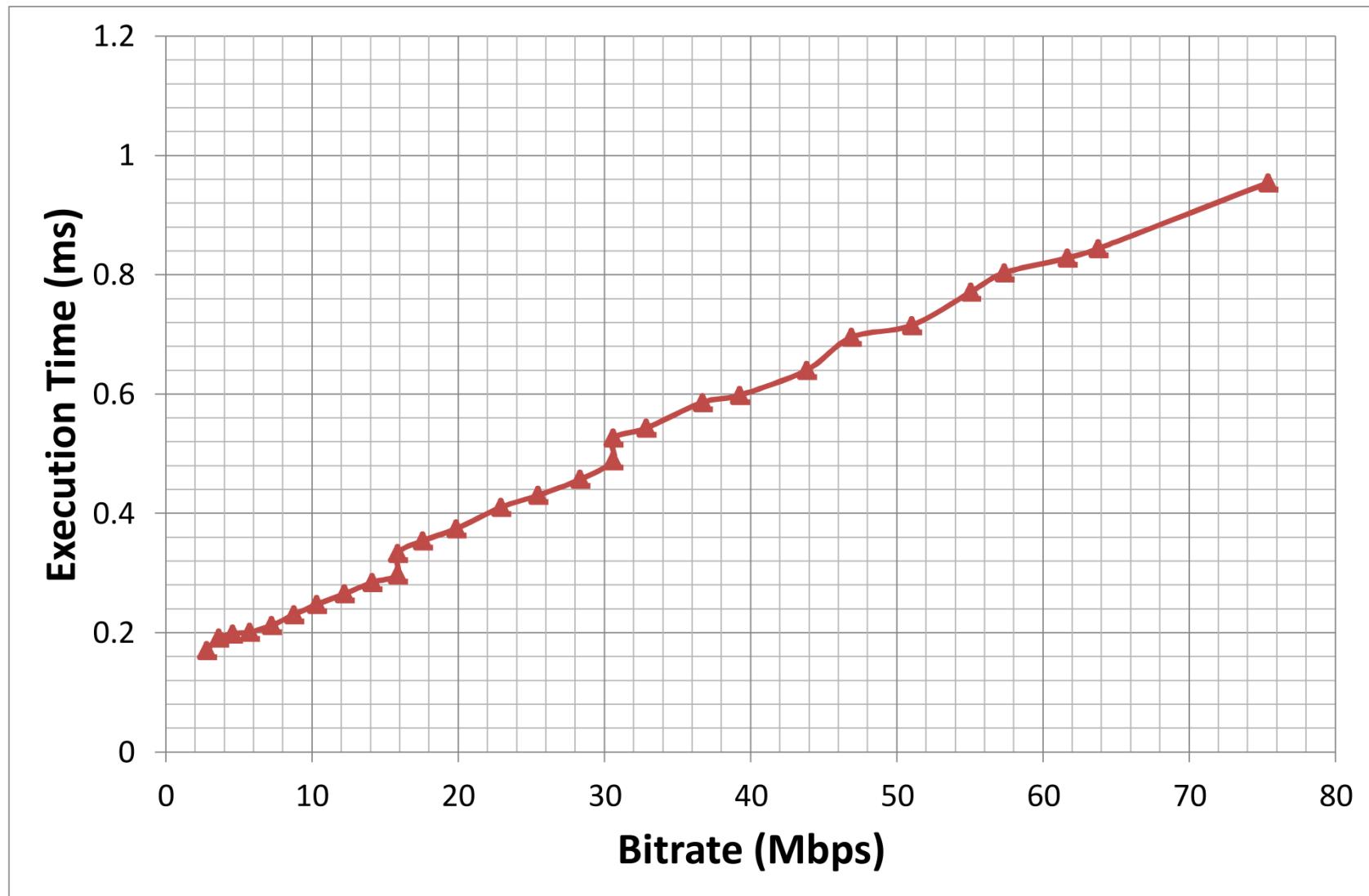
Module Name	Percentage of CPU		
	75 Mbps 64QAM	30 Mbps 16QAM	3.62 Mbps QPSK
Turbo decoder (1 iteration)	78.14 %	64.21 %	20.89 %
OFDM receive processing	6.08 %	11.70 %	33.33 %
Resource Element de-mapping	4.92 %	9.31 %	25.26 %
Rate recovery	4.49 %	5.64 %	8.34 %
CRC checksum	2.92 %	2.23 %	0.72 %
Soft demodulation	1.76 %	2.11 %	3.38 %
Equalization	0.16 %	1.84 %	4.98 %
Others	1.53 %	2.96 %	55.12 %
<b>Total Execution Time</b>	954 $\mu$ s	488 $\mu$ s	170 $\mu$ s

# Turbo Decoder

- We used 16-bit integer arithmetic and 128-bit SSE4 instructions to compute all the trellis (8 states) in parallel
- With 8-bit arithmetic could do 2 codeblocks in parallel
- With AVX2 could do 2 or 4 codeblocks in parallel



# Execution time of PDSCH only @ 20 MHz



[Code](#)[Issues 3](#)[Pull requests 1](#)[Projects 0](#)[Wiki](#)[Settings](#)[Insights](#)

Open source SDR LTE software suite

[Edit](#)[Add topics](#)

1,242 commits

3 branches

9 releases

16 contributors

AGPL-3.0

Branch: master

[New pull request](#)[Create new file](#)[Upload files](#)[Find file](#)[Clone or download](#) **ismagom** Fix memory alignment in PUCCH processing. Fixes #94

Latest commit 2dbc0fe 6 days ago

 cmake/modules

Fixed incompatibility with volk1.2 Make pointer type warnings an error

3 months ago

 lib

Fix memory alignment in PUCCH processing. Fixes #94

6 days ago

 srsenb

Use runtime dir parameter for executables.

2 months ago

 srsue

fix segfault in UE PHY tests

2 months ago

 CHANGELOG

new release and changelog

3 months ago

 CMakeLists.txt

check for compiler flag availability

3 months ago

 COPYRIGHT

Updating notices

a year ago

 CTestConfig.cmake

Updating copyright notices and project name

3 years ago

 CTestCustom.cmake.in

Added scrambling, ratematching and layer mapping tests

4 years ago

 LICENSE

Changed license to AGPL

2 years ago

 README.md

add apt-get line for Ubuntu to install required packages

3 months ago

 cmake\_uninstall.cmake.in

Reorganized the directory structure. Added Graphics support. Added pr...

4 years ago

 README.md

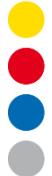
## srsLTE

 srsLTE is a free and open-source LTE software suite developed by SRS ([www.softwareradiosystems.com](http://www.softwareradiosystems.com)).

It includes:

- srsUE - a complete SDR LTE UE application featuring all layers from PHY to IP
- srsENB - a complete SDR LTE eNodeB application
- a highly modular set of common libraries for PHY, MAC, RLC, PDCP, RRC, NAS, S1AP and GW layers.

srsLTE is released under the AGPLv3 license and uses software from the OpenLTE project  
(<http://sourceforge.net/projects/openlte>) for some security functions and for RRC/NAS message parsing.

 OPENFirst

