Unleashing the potential of open-source in the 5G arena
Overview

- Overview and Ecosystem
- Features of current master & develop branch
- Ongoing feature branches
- Hardware targets
- Installation & Usage
- Debugging tools
What is OpenAirInterface?

- **Open-source software-based implementation of 3GPP Technologies**
  - Starting at LTE (Rel 8), including features from LTE-Advanced (Rel 10/11/12), LTE-Advanced-Pro (Rel 13/14), going on to 5G Rel (15/16/…)
  - Spanning the full protocol stack of 3GPP standard
    - E-UTRAN (eNB, UE)
    - EPC (MME, S+P-GW, HSS)
  - Realtime RF and scalable emulation platforms
  - Works with many SDR platforms (ExpressMIMO2, USRP, LimeSDR, …)

- **Makes it is feasible to put a fully-compliant 4G eNodeB and EPC in a commodity x86-based computer (or data center)**

- **Objectives**
  - Building a community of individual developers, academics and major industrials embracing open-source for 5G
  - Become a strong voice and maybe a game-changer in the 3GPP world
    - Real impact from “the little guys” on 3GPP systems
Collaborative Web Tools

- **Main page:**
  - [https://www.openairinterface.org](https://www.openairinterface.org)

- **Code available from**
  - RAN (eNB + UE)
    - [https://gitlab.eurecom.fr/oai/openairinterface5g](https://gitlab.eurecom.fr/oai/openairinterface5g)
  - EPC
    - [https://github.com/OPENAIRINTERFACE/openair-cn](https://github.com/OPENAIRINTERFACE/openair-cn)

- **Mailing lists**
  - [https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/MailingList](https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/MailingList)

- **Developer meetings (Eurecom & Webex)**
  - General: Tuesday Mornings 11:00 CET
  - NR: Friday, alternating between 11:CET (Europe/Asia) and 16:00 CET (Europe/Americas)

- **Forum in Chinese**
  - [http://bbs.opensource5g.org/forum.php](http://bbs.opensource5g.org/forum.php)

- **Other tools:**
  - [https://openairinterface.slack.com](https://openairinterface.slack.com)
  - [https://trello.com/oaidev](https://trello.com/oaidev)
The OpenAirInterface Software Alliance

- Launched in 2014 as an endowment fund (French “Fonds de Dotation”)
- Current strategic members (Orange, TCL, Nokia Bell Labs, Fujitsu)
- Many associate members (Samsung, Interdigital, ng4t, Cisco, B-COM, INRIA, IMT, TNO, III, Rutgers WINLAB, U. Washington, IITH, BUPT, etc.)
- Goals:
  - Promote OpenAirInterface and its open-source licensing model
  - Support the community of developers and users
**OSA Boards**

- **Strategic Board**
  - OSA strategic members
  - Webex meeting every 2 weeks
  - Overview of recent activity
  - Discussion on
    - strategic objectives
    - Legal matters
    - Links with other organizations (e.g. standardization)

- **Technical Board**
  - Members: OSA, Orange, Nokia, Fujitsu, TCL, EURECOM, BCOM, Fraunhofer IIS, OpenCells
  - Roadmap implementation
  - Architecture evolution
  - Documentation & Coding guidelines
  - Review merge requests
    - Code quality
    - Adherence to architecture
The OAI Licensing model

- FRAND License is based on Apache V2.0 but allows committing software with patent rights into OSA and still keep licensing rights -> Inline with 3GPP fair use licensing policy
- We work closely with ETSI on implications of open-source for licensing/certification
Main achievements (develop branch) since last workshop

- Technical Board developed/improved
  - Workflow / Development / Integration policies,
  - Coding guidelines,
  - Documentation

- Many CI Framework improvements

- Multi-UE/RA scheduling improvements

- TDD configuration 1 (stabilization & integration in CI)

- Parallelization & performance improvements

- Rewrite ITTI

- Bug Fixes

- Code cleanup & debugging tools (see later presentation)

- New asn1c

- Repository reorganization
  - separation of eNB & UE in different executables,
  - Preparation for gNB and nrUE
OPENAIRINTERFACE
RAN FEATURES
OpenAirInterface eNB features (PHY)

- The Physical layer implements 3GPP 36.211, 36.212, 36.213 and provides the following features:
  - LTE release 8.6 compliant, and implements a subset of release 10;
  - FDD and TDD configurations 1 (experimental) and 3;
  - Bandwidth: 5, 10, and 20 MHz;
  - Transmission modes: 1, 2 (stable), 3, 4, 5, 6, 7 (experimental);
  - Max number of antennas: 2
  - CQI/PMI reporting: aperiodic, feedback mode 3-0 and 3-1;
  - PRACH preamble format 0
  - All downlink (DL) channels are supported: PSS, SSS, PBCH, PCFICH, PHICH, PDCCH, PDSCH, PMCH;
  - All uplink (UL) channels are supported: PRACH, PUSCH, PUCCH (format 1/1a/1b), SRS, DRS;
  - HARQ support (UL and DL);
  - Highly optimized base band processing (including turbo decoder).
  - Expected throughputs DL
    - 5 MHz, 25 PRBS/ MCS 28 = 16-17 Mbit/s (measured with COTS UE Cat 3/4)
    - 10 MHz, 50 PRBS/MCS 28 = 34-35 Mbit/s (measured with COTS UE Cat 3/4)
    - 20 MHz, 100 PRBS/MCS 28 = ~70 Mbit/s (measured with COTS UE Cat 3/4)
  - Expected throughputs UL
    - 5 MHz, 20 PRBs / MCS 20 = 9 Mbit/s (measured with COTS UE Cat 3/4)
    - 10 MHz, 45 PRBs / MCS 20 = 17 Mbit/s (measured with COTS UE Cat 3/4)
    - 20 MHz, 96 PRBs / MCS 20 = ~35 Mbit/s (measured with COTS UE Cat 3/4)
OpenAirInterface eNB features (MAC)

- The MAC layer implements a subset of the 3GPP 36-321 release v8.6 in support of BCH, DLSCH, RACH, and ULSCH channels.

- The eNB MAC implementation includes:
  - RRC interface for CCCH, DCCH, and DTCH
  - Proportional fair scheduler (round robin scheduler soon)
  - DCI generation
  - HARQ Support
  - RA procedures and RNTI management
  - RLC interface (AM, UM)
  - UL power control
  - Link adaptation
OpenAirInterface eNB features (PDCP)

- The current PDCP is header compliant with 3GPP 36-323 Rel 10.1.0 and implement the following functions:
  - User and control data transfer
  - Sequence number management
  - RB association with PDCP entity
  - PDCP entity association with one or two RLC entities
  - Integrity check and encryption using the AES and Snow3G algorithms
OpenAirInterface eNB features (RLC)

- The RLC layer implements a full specification of the 3GPP 36-322 release v9.3
- RLC TM (mainly used for BCCH and CCCH)
  - Neither segment nor concatenate RLC SDUs
  - Do not include a RLC header in the RLC PDU
  - Delivery of received RLC PDUs to upper layers
- RLC UM (mainly used for DTCH)
  - Segment or concatenate RLC SDUs according to the TB size selected by MAC
  - Include a RLC header in the RLC PDU
  - Duplication detection
  - PDU reordering and reassembly
- RLC AM, compatible with 9.3
  - Segmentation, re-segmentation, concatenation, and reassembly
  - Padding
  - Data transfer to the user
  - RLC PDU retransmission in support of error control and correction
  - Generation of data/control PDUs
OpenAirInterface eNB features (RRC)

- Based on 3GPP 36.331 v14.3.0.
  - System Information broadcast (SIB 1, 2, 3, and 13)
  - RRC connection establishment
  - RRC connection reconfiguration (addition and removal of radio bearers, connection release)
  - RRC connection release
  - RRC connection re-establishment
  - inter-frequency measurement collection and reporting
  - eMBMS for multicast and broadcast (experimental)
  - X2 Handover (really soon now)
  - Paging (to be tested)
  - RRC inactivity timer (recent)
eNB Functional Splits

- IF4.5/IF5: similar to IEEE P1914.1
- FAPI (IF2): specified by small cell forum, implementation (open-nFAPI) by CISCO
- IF1 (F1 in 3GPP Rel 15): first version ready, to be integrated
eNB Functional Split Architecture

Control and Management

RCC
- LTE/NR/NB PDCP
- LTE RRC
- LTE RLC/MAC
- NR RRC
- NB-IoT RRC
- NB-IoT RLC/MAC

RAU
- LTE RLC/MAC
- LTE L1-high
- NR RLC/MAC
- NR L1-high
- NB-IoT RLC/MAC
- NB-IoT L1-high + L1-low

RRU
- LTE L1-low
- LTE L1-high
- NR L1-low
- NR L1-high
- NR L1-low

Data
Control
Management
Buggy/Missing/Incomplete Features in develop

- **Essential**
  - TM3/4/8/9/10 : incomplete
  - PUCCH2 : incomplete
  - TDD configurations : incomplete
  - Measurement gap handling : missing
  - Carrier Aggregation : incomplete
  - X2-handover : MR imminent
  - UE needs improvement (more buggy than master)

- **Needs improvement**
  - Multi-UE UL/RA scheduling
  - RLC AM mode
Experimental/upcoming eNB features

- DRX/eDRX handling
- Multi RRU handling and synchronization
- Rel 13 LTE-M
- Rel 14 NB-IoT (see presentation during workshop)
- Rel 14 D2D/Sidelink/ProSe
- Rel 15 5G-NR
Support for eMTC in OAI

- **Release 13 eMTC**
- **Public branch**
  - enhancement-ltem
  - Merge with current develop branch completed
    - Legacy LTE functionality validated on merge, still testing LTE-M
  - CI Framework integration => push to develop imminent
- **Tested with commercial LTE-M Modules**
  - Nimbelink/Pycom Modules (Sequans Cat-M chipset)
  - Nimbelink Modules (QCOM chipset)
  - Nordic Semiconductor (nRF91 prototype)
  - Integration in OAI CI (Nimbelink, Pycom)
- **Currently limited (testing) to CEMode A (CE Levels 0,1)**

(c) Eurecom 2018
Current Status

What was currently implemented

- Configuration: lots of new eMTC-related parameters to be fed to L1/L2 stack (in enhancement-Item)
- RRC (in enhancement-Item branch)
  - Handling of Rel-13 information elements for eMTC
  - eMTC System Information handling
  - Extra bits in MIB
  - SIB1/SI: Quite Different from legacy LTE (repetitions, frequency-hopping, no DCI)
- PRACH handling (in current develop branch)
  - Support for up to 4 CE levels: different number of repetitions per level => signal combining across repetitions for each level
  - New thread for eMTC PRACH (different parametrization in LTE Cell for eMTC)
- MPDCCH (in current develop branch)
  - Support for EPDCCH allocation (only in MPDCCH configuration for now)
  - 3 new DCI formats
  - New procedures (search space, etc.)
  - No Repetitions yet
  - Limited to 4+2 PRB configuration
- PDSCH (in current develop branch)
  - No Repetitions
- PUSCH (in current develop branch)
  - No Repetitions yet
- PUCCH (in current develop branch)
  - No Repetitions yet (removal of slot-frequency hopping)
- MAC (in current develop branch)
  - RA procedures for eMTC
  - Basic scheduler for testing

(c) Eurecom 2018
Overview of current implementation

- Very basic scheduler
  - 1 downlink subframe every 2 frames with fixed mcs
  - 1 uplink subframe every 2 frames with fixed mcs

- Need to test repetition mechanisms for coverage enhancement
  - Will implement basic UE procedures to enhance dlsim/ulsim to test repetitions in TX/RX
eMTC Configuration file

- Default file
  - targets/PROJECTS/GENERIC-LTE-ENB/CONF/enb.band13.tm1.50PRB.emtc.conf
  - New elements
    - Shown on gitlab
Outlook

- Testing of MCL with real modules
- Integration of IoT applications / MEC.
- Design of joint LTE/LTE-M scheduling framework
  - How to share resources efficiently
  - Need UE L2 stub for testing scalability (minimal Cat-M1 functionality in OAI UE)
OAI supporting Rel 14 D2D/Sidelink/ProSe

Application scope
- Public safety: Group Communication (e.g., Availability when cellular networks are not available or fail for e.g. after disasters, earthquakes etc.)
- Proximity services and commercial applications (e.g., local advertising)

OAI integration objectives and testing
- Integration of Rel 14 Sidelink procedures (L1/L2): Status: Completed
- Integration of Rel 14 Uu procedures supporting sidelinlk (in-network): Status: partially integrated
- Interfaces for ProSe applications in UE: Status: Completed
- Integration of Rel 14 CN procedures supporting ProSe and UE-Network Relay: Status: to be integrated (beginning of 2019)
- Testing
  - ProSe application provided from PerspectaLabs (not public)
  - Small field deployment with OAI-based UEs and Infrastructure for off-net, on-net and relay scenarios

(c) Eurecom 2018
OAI supporting Rel 14 D2D/Sidelink/ProSe

- Integration of Rel 14 Sidelink procedures (L1/L2) for 1-to-many and 1-to-1 communication
  - Synchronization channels and procedures: PSS, SSS, SLBCH
  - Sidelink Discovery Channel (SLDCH) and procedures
  - Sidelink Shared Channel (SLSCH) / Control Channel (SLCCH) and procedures

- Integration of Rel 14 Uu procedures supporting sidelink (in-network)
  - RRC signaling supporting sidelink parameterization and resource allocation (e.g., SIB 18, 19, 21, SidelinkUEInformation)
  - Dedicated resource allocation procedures to be integrated

- Interfaces for ProSe applications in UE
  - PC5-D supporting discovery of other UEs in proximity
  - PC5-S supporting signaling for 1-to-1 sidelink connection establishment, maintenance and release
  - PC5-U supporting User-plane direct traffic between two UEs

- Integration of Rel 14 CN procedures supporting ProSe and UE-Network Relay
  - UE-NAS signaling supporting relay operation to be integrated in OAI UE
  - Core network extensions supporting relay operation to be integrated in OAI-CN

(c) Eurecom 2018
OAI supporting Rel 14 D2D/Sidelink/ProSe

- **Testing capabilities:**
  - Emulation testbed
    - MAC-to-MAC emulation mode bypassing PHY procedures
    - Emulation mode which simulates PHY sidelink procedures
  - RF-testbed
    - UE node: NUC PC (8 CPU-core, 8GB RAM) connected with USRP B200-mini
    - USRP devices currently connected with external signal generator to get synchronized

- **Testing status**
  - Off-network unicast/multicast scenarios (LTE-D2D Mode2)
    - Tested in both RF and emulation
  - On-network unicast/multicast scenarios (LTE-D2D Mode1)
    - Tested in emulation, not RF yet
  - Partial coverage (UE-to-Network relay procedures)
    - Initial tests in emulation (non-3gpp aligned for now), not in RF yet

- **Code availability**
  - Currently under private repository
  - To be merged with develop branch of main OAI by beginning of January 2019

(c) Eurecom 2018
5G-NR implementation status (Dec. 2018)

- **Branch**: develop-nr
- **PHY layer additions**
  - Highly efficient 3GPP compliant LDPC encoder and decoder (BG1 and BG2 supported)
  - Highly efficient 3GPP compliant polar encoder and decoder
  - NR-PSS and NR-SSS (validated with R&S FSW)
  - NR-PBCH single beam (validated with R&S FSW)
  - NR-PDCCH (validated with OAI UE)
    - type 0
    - DCI formats: 00, 10
  - NR-PDSCH (validated with OAI UE)
    - Single symbol DMRS, dmrs-TypeA-Position Pos2,
    - DMRS configuration type 1
    - PDSCH mapping type A
- **MAC**
  - Updated FAPI interface to 5G-NR
  - Adapted FAPI interface also to UE
- **Higher layers**
  - Import of 38.331 RRC messages using asn1c (new version)
  - Application to read configuration file and program gNB RRC
  - Generation of MIB (validated with R&S FSW)
  - RRC -> MAC configuration
  - MAC -> PHY configuration using FAPI P5 interface
  - MAC dummy scheduler using FAPI P7 interface
- **Architecture**
  - Non-standalone (Architecture option 3); initially with emulated 4G link
  - standalone (option 2): requires 5G core

- Subcarrier spacing: 30kHz (60kHz, 120kHz)
- Bandwidths: 40MHz, (80MHz, 100MHz)
- Normal cyclic prefix
- Static TDD

Come and see our demo!
Hardware Requirements

- **SDR platform**
  - ExpressMIMO2 (discontinued)
  - USRP B200, X300, N300 (recommended)
  - Blade RF
  - LMS-SDR
  - Skylark Iris
  - Syrtem

- **Host PC**
  - A powerful x86 PC (recommended)
    - Intel Core i5, i7, i9
    - Intel Xeon
    - Intel Atom
    - 4 cores, > 3GHz, SSE 4, AVX
  - Low-cost x86 PC
    - Up board (up2), Euclid board
  - ARM (experimental)
    - Odroid

- **Antennas, Duplexers, etc**
## Comparison

<table>
<thead>
<tr>
<th></th>
<th>USRP B210</th>
<th>USRP X310</th>
<th>USRP N310</th>
<th>Blade RF 2</th>
<th>LMS SDR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data acquisition</strong></td>
<td>USB3</td>
<td>Gbit EtherNet, PCIexpress</td>
<td>Gbit Ethernet</td>
<td>USB3</td>
<td>USB3</td>
</tr>
<tr>
<td><strong>MIMO and bandwidth capabilities</strong></td>
<td>2x1 MIMO 20MHz or 2x2 MIMO 10MHz</td>
<td>2x2 MIMO, 120MHz</td>
<td>4x4 MIMO 100MHz</td>
<td>2x2 MIMO 20MHz</td>
<td>2x2 MIMO 20MHz</td>
</tr>
<tr>
<td><strong>RF chip</strong></td>
<td>AD9361</td>
<td>n/a**</td>
<td>AD9371 (x2)</td>
<td>AD9361</td>
<td>LMS7002M</td>
</tr>
<tr>
<td><strong>Frequency range</strong></td>
<td>70MHz – 6GHz (depends on daughterboard)</td>
<td>DC-6GHz (depends on daughterboard)</td>
<td>10 MHz – 6GHz</td>
<td>47MHz to 6GHz</td>
<td>300 MHz – 3.8GHz</td>
</tr>
<tr>
<td><strong>Price</strong></td>
<td>€1,130</td>
<td>~€5,000</td>
<td>~€10,000</td>
<td>$480 - $720</td>
<td>$299</td>
</tr>
<tr>
<td><strong>Duplexing</strong></td>
<td>FDD* or TDD*</td>
<td>FDD* or TDD*</td>
<td>FDD* or TDD</td>
<td>FDD*</td>
<td>FDD* or TDD*</td>
</tr>
<tr>
<td><strong>Output power</strong></td>
<td>10dBm</td>
<td>n/a**</td>
<td>12-18dBm</td>
<td>8dBm</td>
<td>10dBm</td>
</tr>
<tr>
<td><strong>Noise figure</strong></td>
<td>&lt;8dB</td>
<td>n/a**</td>
<td>5.5-7.5dB</td>
<td>?</td>
<td>&lt;7dB</td>
</tr>
<tr>
<td><strong>EVM</strong>*</td>
<td>Very good</td>
<td>Excellent</td>
<td>???</td>
<td>???</td>
<td>Average</td>
</tr>
<tr>
<td><strong>Open source</strong></td>
<td>FPGA/driver</td>
<td>FPGA/driver</td>
<td>FPGA/Driver</td>
<td>All</td>
<td>All</td>
</tr>
<tr>
<td><strong>Compatibility</strong></td>
<td>4G</td>
<td>4G/5G (80MHz with ¾ sampling)</td>
<td>5G up to 100MHz</td>
<td>4G</td>
<td>4G</td>
</tr>
</tbody>
</table>

*needs external RF elements
** depends on daughterboard
*** subjective to the author 😊
Other experimental targets

- **Epiq Sidekiq**
  - Based on AD 9361
  - Mini PCIe or M.2 form factor

- **CPRI - PClexpress**
  - IT Avero
  - Based on Xilinx eval board

- **CPRI gateway**
  - Bell Labs
  - Based on Xilinx or Intel platform

- **Skylark Iris platform**
  - Based on Lime platform
  - Scalable for massive MIMO

- **SYRTEM UED platform**
  - Based on Xilinx ZC706 eval board + AD9371 daughterboard
  - 2 full duplex channels with up to 122.88 MHz sampling
  - Not 100% open source
OAI eNB + OAI UE

INSTALLATION
Software Requirements

- **Operating system**
  - Ubuntu >= 16.04.2
    - works for both openairinterface5g and openair-cn
    - For real-time operation, a low-latency kernel is recommended
    - For P/S-GW, gtp kernel module needs to be patched
    - See details on Wiki
  - CentOS Linux release 7.4.1708 (Core)
    - Better real-time performance than Ubuntu low-latency

- **Get code from our gitlab server**
  - RAN (eNB+UE): [https://gitlab.eurecom.fr/oai/openairinterface5g](https://gitlab.eurecom.fr/oai/openairinterface5g)
    - Branch develop latest features (recommended)
    - Several feature branches for cutting-edge developments
OpenAirInterface5G directories

- **cmake_targets**
  - New directory for building all the targets
  - Contains “mother” build_oai script

- **targets**
  - Hardware specific code (drivers, tools, etc)
  - lte-softmodem,

- **openair1**
  - Basic DSP routines for implementing subset of LTE specifications under x86 (36.211, 36.212, 36.213 3GPP specifications)
  - Channel simulation, sounding and PHY abstraction software,

- **openair2**
  - MAC/RLC/PDCP/RRC

- **openair3**
  - Contains interfaces S1-C, S1-U (GTP, SCTP, S1AP) and NAS UE

- **common/utils**
  - Utilities such as the T tracer or the ITTI
Compiling OpenAirInterface5G

- **Top-level build script** ./build_oai located in
  - cd openairinterface5g/cmake_targets

- **Compilation options**
  - -I installs additional required software
  - -w <hw_target> select HW target
  - --eNB compiles the lte-softmodem
  - --UE compiles UE & NAS parts
  - --T-tracer compiles with T support
  - --lte-simulators compiles the unitary simulators
  - --noS1 compiles without NAS (UE and eNB): will go away soon
  - -t ETHERNET compiles Ethernet transport library (for use in IF4.5 or IF5 split)
  - -h help

- **This creates executables in openairinterface5g/targets/bin**
  - Liboai_device.so symbolic link to library of current hardware
  - Liboai_eth_transpro.so Ethernet transport library (IF4.5 or IF5 split)
  - Libparams_libconfig.so Library for parameter handling
  - Libcoding.so Library for channel coding
  - Nasmesh.ko or ue_ip.ko: kernel driver for noS1 mode or UE NAS
How to connect COTS phone to OAI eNB

**Additional requirements**
- Core network (e.g., OAI EPC, see later in training)
- Sim card with corresponding parameters

**Compile eNB**
- `./build_oai --w USRP --eNB`

**Configure eNB**
- `targets/PROJECTS GENERIC-LTE-EPC/CONF/`
- Select the config file that is most appropriate for your configuration (Band and Hardware)
- Check
  - MCC, MNC, TAC (need to match EPC)
  - downlink_frequency, bandwidth, etc
  - IP addresses of S1-MME and S1-U interfaces

**Run eNB**
- `sudo ./lte-softmodem --O <file.conf>`
How to connect OAI UE to OAI eNB

- **Compile UE**
  - `./build_oai --w USRP --eNB [--ue-nas-use-tun | --noS1]

- **Initialize NAS (except when using TUN interface)**
  - “init_nas_s1 UE” or “init_nas_noS1”

- **Run UE**
  - `sudo ./lte-softmodem -U -C <freq> -r [25|50|100] -ue-scan-carrier -ue-txgain xx -ue-rxgain yy (-d)`
How to setup RCC and RRU

- **RCC**
  - `./build_oai -w USRP -t ETHERNET -eNB`
  - Check RU parameters in RCC config file
    - IF name, local and remote IP addresses and ports
    - `local_rf = "no"`
    - `tr_preference = "udp_if4p5"`
  - `./lte-softmodem -O <file.conf>`

- **RRU**
  - `./build_oai -t ETHERNET -eNB`
  - Check RU parameters in RCC config file
    - IF name, local and remote IP addresses and ports
    - `local_rf = "yes"`
    - `tr_preference = "udp_if4p5"`
  - `./lte-softmodem -O <file.conf>`
Troubleshooting

- eNB not connection to MME / RRH
  - Check IP addresses in config files
  - Check MCC, MNC matching

- I get a lot of UUUs and LLLs
  - Check the performance setting of CPU (C-states, CPU frequency)
  - Check USB3 connection (some cables are bad)

- Phone does not connect
  - Analyze S1AP messages in wireshark
  - Check keys in SIM card and HSS
  - ...

- Throughput is very low
  - Check radio conditions: duplexer, antennas, interference
DEBUGGING TOOLS
Debug tools

- **The T tracer**
  - Monitor the eNB in real-time, simulation, or playback mode

- **Telnet server**
  - Monitor and change parameters of the eNB in real-time or simulation

- **Simulators**
  - ulsim/dlsim
  - Basic simulator
  - L1 simulator
  - L2 FAPI simulator*  

*being integrated into development
The T tracer

- The T tracer is a framework to debug and monitor the eNB softmodem.

- Combines logging, timing analysis, signal visualization, MAC PDU analysis (with wireshark)

- It is made of two main parts:
  - an events collector integrated to the real-time processing,
  - a separate set of programs to receive, record, display, replay and analyze the events sent by the collector.

- Can work locally or over network
The T tracer: usage of GUI

- Compile eNB with -T-tracer option:
  - `./build_oai -w USRP -eNB -T-tracer`

- Compile eNB GUI:
  - `cd openairinterface5g/common/utils/T`
  - `make`

- Run lte-softmodem normally
  - `sudo ./lte-softmodem -O <...>`

- Run T tracer GUI
  - `./enb -d ../T_messages`

- Other features
  - Recording & replay
  - VCD file generation (for gtkwave)
- HARQ ACK
- HARQ NAK
- New DCI
- Retr. DCI

eNB GUI
Telnet server

- Telnet server can be used to show and change parameters at runtime
  - Log level and verbosity
  - Threads and their priority
  - Some PHY parameters (e.g. turbo iterations)

- Easily extendable

- Usage
  - ./build_oai -w USRP -eNB -build-telnetsrv
  - sudo ./lte-softmodem -O <…> --telnetsrv
  - Telnet 127.0.0.1 9090
  - Use online help
Simulators

- **dlsim/ulsim**
  - `./build_oai --phy_simulators`

- **Basic simulator**
  - `./build_oai --basic-simulator`
  - See targets/ARCH/tcp_bridge/README.tcp_bridge_oai

- **L1 simulator (ex oaisim)**
  - Based in IF5/IF4.5
  - `./build_oai --eNB -t ETHERNET --noS1`
  - eNB: `sudo ./lte-softmodem -O ../../targets/PROJECTS/Generic-LTE-EPC/CONF/rcc.band7.tm1.if4p5.50PRB.lo.conf`
  - `./build_oai --UE -t ETHERNET --noS1`
  - UE: `sudo ./lte-uesoftmodem -O ../../targets/PROJECTS/Generic-LTE-EPC/CONF/rru.oaisim.conf -A AWGN -r 50 -s 25 --siml1`

- **L2 simulator**
  - Based on nFAPI (IF2)
  - Same build process as L1 simulator but config files for nFAPI
  - See targets/DOCS/nfapi-L2-emulator-setup.txt for details
BACKUP
Use case I: classical 3GPP network

- OAI EPC
- Commercial/3rd party EPC
- OAI eNB
- Commercial/3rd party eNB
- OAI UE
- COTS UE
Use case II: simplified network

- **Non-3GPP setup (no-S1 mode):**
  - OAI eNB <-- OAI UE
Use case III: cloud-RAN

Main target of EURECOM deployment
Epiq Sidekiq

- Based on AD 9361 chipset
  - 70MHz - 6GHz with up to 50MHz bandwidth per channel

- SidekiqTM - MiniPCIe
  - MiniPCIe card form factor (30mm x 51mm x 5mm)
  - 2 independent RF channels (2xRx or Tx+Rx)
  - PCIe Gen1.1 x1 (2.5 Gbps) interface to host + USB 2.0 interface

- SidekiqTM - M.2
  - M.2 T3042-D3-B card form factor (30mm x 42mm x 4mm)
  - Up to 2x2 MIMO
  - PCIe Gen2 x1 (5 Gbps) interface to host + USB 2.0 interface

- Under beta-testing