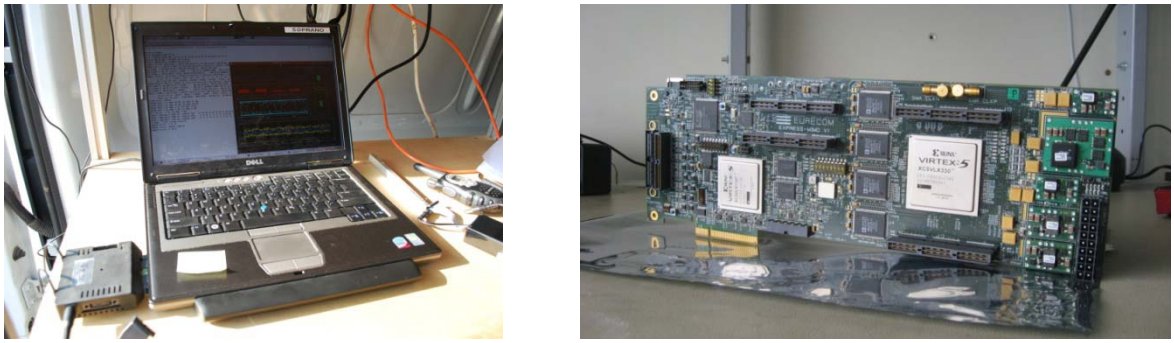
The OpenAirInterface (www.openairinterface.org) platform is an open-source SDR-based hardware/software development platform targeting 4th generation wireless systems such as LTE and medium-range rapidly-deployable mesh networks. The platform uses all-IP wireless networking and implements the PHY and MAC layer in real-time on a combination of generic PC and FPGA-based equipment under open-source tool chains and software development environments (Aeroflex-Geisler LEON/GRLIB, eCos, RTAI, Linux, GNU). The equipment is reasonably-priced and completely open (GNU GPL), both at the hardware and software level. Both PCMCIA-based TDD-only equipment (1.9 GHz 2x2 MIMO 5MHz channelization, 23 dBm UE / 34 dBm eNodeB) and more sophisticated PCI-express-based TDD/FDD equipment (2.6 GHz 4x4 MIMO 20 MHz channelization, 23 dBm UE, 34 dBm eNodeb) can be made available to the consortium. Eurecom has outdoor experimental licenses at 1.9 GHz and 2.6 GHz around its premises in Sophia Antipolis.

The OpenAirInterface equipment consists of the following hardware components:

CardBus MIMO I, a fully software defined radio for 2x2 MIMO communications with a PCMCIA form-factor (see below). The RF section is time-division duplex and operates at 1.900-1.920 GHz with 5 MHz channels with 21dBM transmit power per antenna for an OFDM waveform. The cards house a medium-scale FPGA (Xilinx X2CV3000) allowing for an embedded HW/SW system implementing the physical layer jointly with soft-MODEM routines on the host PC. CardBus MIMO I will be upgraded in 2010 to newer low-power FPGA technologies.

Express MIMO is a baseband processing board in a PCI-express form-factor with on-board data converters providing 8x8 MIMO input/output capacity with up to 40 MHz channels. It is a generic baseband processing engine for high-performance radio signal processing. It uses a standard PCI-express interface which can be controlled via desktop PC or a laptop via an ExpressCard adapter. ExpressMIMO makes use of two high-density Xilinx Virtex 5 FPGAs: LX110T provides a high-speed interface with a PC-based system. The LX110T houses a LEON3-based embedded system which is interconnected with a signal-processing engine in an LX330 FPGA. EURECOM currently uses off-the-shelf RF solutions with ExpressMIMO based either on discrete-components or commercially-available WIMAX/LTE 2.6 GHz chipsets from Maxim semiconductor.



(a) (b)

Figure 1-1: (a) CardBus MIMO I and (b) ExpressMIMO Card

The software components are organized into four areas, which correspond more or less to the different layers of the Open Systems Interconnection (OSI) reference model.

1. Openair0: Wireless Embedded System Design, and RF subsystem
2. Openair1: Baseband signal processing

3. Openair2: Medium-Access Protocols
4. Openair3: Wireless Networking Protocols

OpenAirInterface provides a full real-time open-source implementation of PHY and MAC layers. Currently the implementation supports the CBMIMO1 cards and runs in real-time as a soft-MODEM on the host PC under the real-time operating system (RTOS) RTAI or on multiprocessor system-on-chip (MPSoC) architecture on ExpressMIMO. The PHY layer of the platform targets LTE-like networks and thus uses multiple-input multiple-output orthogonal frequency division multiple access (MIMO-OFDMA) as modulation and multiple access technique. Moreover, PHY signaling strategies are included to provide the means for exploiting channel state feedback at the transmitters in order to allow for advanced PHY allocation of OFDMA resources via the MAC. The medium access control (MAC) layer supports wideband multiuser QoS-aware scheduling and hybrid-ARQ. Furthermore there is a 3GPP radio-link control layer (RLC) and a simple radio resource control (RRC). A non-access-stratum driver provides IPv4/IPv6 interconnection with Linux networking services.

The OpenAirInterface emulation environment can be configured for real-time PC-based targets and user-space non-real-time PC based targets. Both allow for virtualization of network nodes within physical machines and distributed deployment on wired Ethernet networks. Virtualization is done within the same operating system instance and the Linux IP protocol stack is shared among nodes in the same physical machine. Nodes in the network communicate via direct-memory transfer when they are part of the same physical machine and via multicast IP over Ethernet when they are in different machines. The communication between nodes allows for the exchange of transport data at the PHY and MAC interface. The behavior of the wireless medium is simulated using a PHY abstraction unit which simulates the error events in the channel decoder and provides simulated measurements from the PHY in real-time. The remainder of the protocol stack for each node instance uses a complete implementation as would a full-RF system. The emulator is designed to represent the behavior of the wireless access technology in a real network setting while obeying the temporal frame parameters of the air-interface. It makes use of the open-source real-time operating system extension to Linux, RTAI to guarantee hard real-time behavior. With virtualization of the protocol stack, many instances (on the order of 30 on a 2GHz Quad-core Xeon) can reside in the same physical machine. This targets large-scale repeatable emulations on a real protocol stack using real applications.