Hardware Platform and Implementation of a Real-Time Multi-User MIMO-OFDM Testbed

M. Wenk, P. Luethi, T. Koch,
P. Maechler, N. Felber, and W. Fichtner
Integrated Systems Laboratory
ETH Zurich, 8092 Zurich, Switzerland
Email: {mawenk,luethi,koch,maechler,felber,fw}@iis.ee.ethz.ch

M. Lerjen
Communication Technology Laboratory
ETH Zurich, 8092 Zurich, Switzerland
Email: mlerjen@nari.ee.ethz.ch

***Abstract*— This paper describes a modular hardware platform of a multi-user (MU) multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) testbed. The hardware platform is based on multiple field programmable gate arrays (FPGAs), provides four integrated radio-frequency (RF) chains, and has capabilities for extension boards. The performance and modularity of the testbed enables real-time MU-MIMO-OFDM experiments as well as offline processing experiments. To this end, the MIMO physical (PHY) layer of Haene *et al.*, IEEE J-SAC, 2008, has been adapted to the new hardware platform and extended with bi-directional communication facilities and a basic media access control (MAC) layer equipped with Ethernet connectivity.**

I. Introduction

Due to the increased spectral efficiency, the use of multiple antennas at the transmitter and the receiver is becoming state-of-the-art in point-to-point (i.e., single-user) wireless communication systems. Although the performance improvement of MIMO in terms of throughput and/or quality of service entails high signal processing complexity, it has been shown that MIMO is feasible in practice for up to four streams [1].

In MIMO communication, the shift from single-user to multi-user scenarios requires changes in the PHY layer as well as in the MAC layer [2]. To thoroughly assess real-world implementation and performance aspects as well as limitations of MU-MIMO communication, a prototype system is of paramount importance. Various testbeds focusing on implementation aspects of the single-user MIMO PHY layer exist, either running in real-time, e.g. [1], [3], [4], or performing offline processing, e.g. [5], [6]. So far, only little can be found on MU-MIMO testbeds in literature. In [6] for example, transmit beam-forming experiments for MU-MIMO communications on an offline-processing-based testbed have been described. A real-time MU-MIMO implementation and corresponding measurements based on the forthcoming long-term evolution of the 3G air interface can be found in [7].

Contributions: This paper describes an FPGA-based hardware platform and the necessary extensions of [1] in order to obtain MU-MIMO-OFDM communication capabilities. These extensions include a basic MAC layer, an integrated RF transceiver chain, and Ethernet connectivity. Furthermore, MAC throughput measurements are presented.

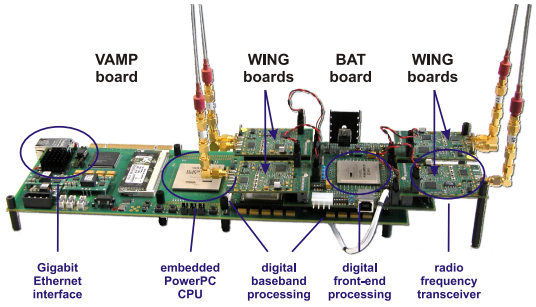

Fig. 1. Overview of the testbed hardware platform (one MIMO terminal).

Outline: The remainder of this paper is organized as follows. Sec. II describes the hardware components of the testbed terminals. Sec. III briefly describes the MIMO PHY and the MAC layer implementation. Sec. IV provides measurement results on inter-frame spacing and throughput on MAC and TCP layer. Sec. V briefly describes how the testbed functionality for offline experiments over real-world channels.

II. Hardware Platform

The former hardware platform described in [1] consists of one XILINX Virtex-II FPGA, two digital-to-analog conversion (DAC) and analog-to-digital conversion (ADC) modules, and four super-heterodyne RF chains. The RF chains are built from Mini-Circuits components. Four of these RF chains occupy about the volume of a desktop PC and require external clock generation. Limited signal processing resources, the bulky RF chains, and the need for a basic MAC layer led to the decision to design a new hardware platform for the MU-MIMO-OFDM testbed. This new platform consists of individual terminals, each providing full 4-stream MIMO transmit and receive capabilities. Fig. 1 shows such a testbed terminal consisting of three different types of printed circuit boards: the VAMP board, the BAT board, and four WING boards.

A. VAMP Board

The VAMP board provides the basis of each MIMO testbed terminal. It contains two XILINX Virtex-II Pro FPGAs with embedded PowerPC CPUs, 4 MByte SRAM memory, a socket

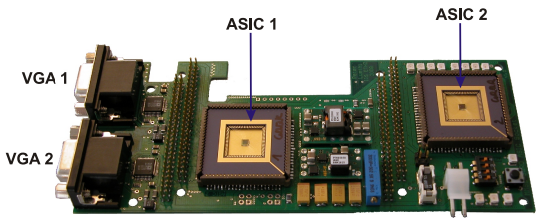

Fig. 2. Adapter board containing two ASIC sockets and two VGA ports.

for optional SDRAM, a Gigabit Ethernet interface, a PCI interface, and digital-IO connectors for additional daughter boards.

B. BAT Board and WING Boards

The BAT board contains the digital front-end including up-/downsampling and filtering. It contains one XILINX Virtex-4 FPGA, 1 MByte SRAM memory, dedicated high-speed interconnections to attach up to four WING boards, configuration logic, and USB connectivity for user control.

Each WING board contains a complete direct-conversion RF chain for wireless transmission and reception. The main components of the RF chain are a DAC/ADC chip (Analog Devices AD9861), a dual-band RF transceiver chip (Maxim Integrated Products MAX2829), and a front-end module for filtering and amplification (Epcos R025B). A separate ADC chip (Analog Devices AD8061) is used to digitize the received signal strength indication (RSSI) signal and provide it to the FPGA for automatic gain control (AGC).

C. ASIC Adapter Board

An optional application specific integrated circuit (ASIC) adapter board can be plugged onto the VAMP board. The ASIC adapter board is shown in Fig. 2. It provides connectivity for two ASICs (CLCC84 package with 84-pins) what allows for outsourcing of timing-critical signal processing tasks. Moreover, the adapter board provides two VGA ports for visualization and debugging purposes.

III. Real-Time MU-MIMO-OFDM Testbed

Fig. 3 shows the overall system architecture of a MU-MIMO-OFDM terminal. It can directly be connected to the Ethernet network or communicate wireless with other terminals over the MIMO PHY layer. The entire testbed can work similar to an Ethernet-based switch with MIMO wireless bridging in-between the end-points. There are three main subsystems involved:

- The Ethernet subsystem enables network connectivity including data transfers from and to the Internet.
- The PowerPC subsystem is responsible for all MAC layer tasks.
- A complete MIMO PHY layer is implemented in the MIMO subsystem. A detailed description of its implementation can be found in [1].

The three subsystems and the external SRAM are interconnected through a common bus. The advanced micro-controller

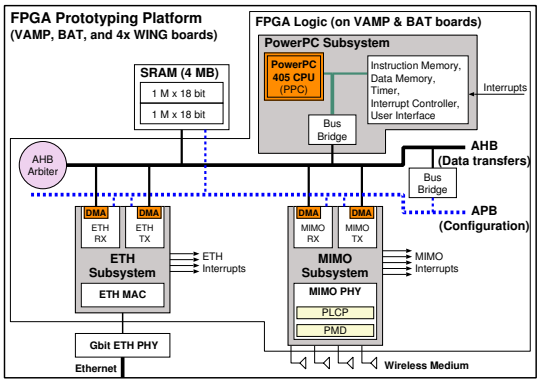

Fig. 3. System architecture of a MU-MIMO-OFDM terminal including PowerPC, MIMO and Ethernet (ETH) subsystem.

bus architecture (AMBA) by ARM has been used, where an advanced high-performance bus (AHB) is used for data transfers and the advanced peripheral bus (APB) for control and configuration of the individual system blocks. High-speed data transfers without CPU interaction are carried out by direct memory access (DMA) engines. The external SRAM is used to buffer data between the MIMO subsystem and the Ethernet subsystem.

A. MIMO PHY Layer

The MIMO PHY layer is based on [1] and the modulation parameters of the PHY layer are summarized in Tbl. I. The PHY header consists of four short preambles, two long preambles, four MIMO training symbols, and one signal field. To adapt the PHY layer to the new hardware platform, it has been partitioned onto the three FPGAs as shown in Fig. 4. The first FPGA on the VAMP board is responsible for the MAC layer, for the MAC-PHY layer interface, and for channel coding and decoding. Channel estimation and linear MMSE detection (summarized as MIMO processing), OFDM de-/modulation (mapping of the binary-valued data stream to constellation points and I/FFT), and synchronization reside on the second FPGA of the VAMP board. Since the RF chain on the WING board no longer uses an intermediate carrier frequency (IF), the digital front-end interface needed to be modified completely. An AGC unit has been implemented on the BAT board that adjusts the amplifiers in the receive path according to the RSSI signal. To reduce the requirements on both, the image rejection filter in the transmit path and the anti-aliasing filter in the receive path, the 20 MHz baseband signal is digitally upsampled by a factor of four in the transmitter and decimated accordingly in the receiver.

B. Media Access Control (MAC) Layer

The MAC layer is implemented on the PowerPC of an FPGA. A polling-based access scheme was chosen in order to keep the MAC protocol simple. One terminal acts as access point (AP) for the entire MIMO network, whereas the other terminals are configured as stations (STAs). A point coordination function in the AP is responsible for polling all STAs regularly according to its internal scheduling algorithm

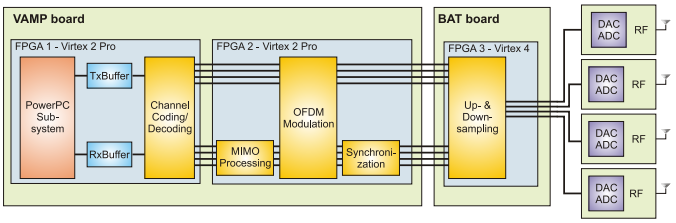

Fig. 4. Partitioning of the MIMO PHY layer onto the three FPGAs.

(e.g., round robin). Before a packet gets transmitted, a 32 byte MAC header is added to the payload data. The header contains the size of the entire frame, the address of the transmitter and receiver, the packet type, the payload type, and the sequence number. The entire MAC layer is implemented in software on the PowerPC. The interaction of the MAC layer with the Ethernet PHY layer and the MIMO PHY layer is handled by interrupts.

IV. Measurement Results

A. Inter-Frame Spacing

The latency introduced by the PHY layer varies with the frame length due to the preprocessing latency [8]. As the detection and decoding of an OFDM symbol takes less time than the OFDM symbol duration t_s, the preprocessing latency can be caught up. The latency of the implemented MAC layer depends on the PowerPC's state (actuality of the cache, interrupts, etc.), but is in the order of 4 µs on average. The total system latency defines the inter-frame spacing and was measured by using a high-performance digital oscilloscope with an RF antenna directly attached to its input. Fig. 5 shows the resulting latency figures for different numbers of transmitted OFDM symbols. For the shortest frames possible, the inter-frame spacing is approximately 75 µs while it reduces to 18 µs for long frames (more than 80 OFDM symbols).

TABLE I
Modulation parameters of the MIMO-OFDM PHY layer based on [1] and MAC layer characteristics.

Channel bandwidth	20 MHz
Number of streams (N_S)	4
Modulation schemes; number of bits per symbol (Q)	BPSK (1), QPSK (2), 16-QAM (4), 64-QAM (6)
Coding rate (R)	1/2, 2/3, 3/4
Number of subcarriers Number of data subcarriers (N_{DSc})	64 48
OFDM symbol duration (t_s)	4 µs
Guard interval	0.8 µs
Subcarrier spacing	312.5 kHz
Inter-frame spacing (t_{ifs})	18-75 µs
PHY header duration (t_{PHY})	44 µs
MAC header length (L_{MAC})	32 bytes
Payload length ($L_{Payload}$)	0-8 kbytes
Frame check sequence length (L_{FCS})	4 bytes

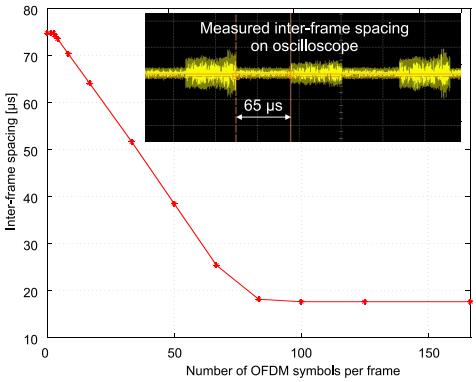

Fig. 5. Inter-frame spacing for different number of OFDM symbols.

B. Overall System Performance

In order to evaluate the performance of the testbed including the new RF chains and the MAC layer, measurements with a channel emulator have been conducted. Unfortunately, bi-directional 4-stream communication is not directly supported by the available channel emulator. Therefore, uni-directional measurements of the frame error rate for different frame lengths and the IEEE TGn channel model A (flat-fading) and C (residential, 200 ns maximum delay spread) [9] were carried out. The measurements were taken for 100 channel realizations and 100 frames per channel with random payload data. Block fading has been assumed, i.e., the emulated channel does not change during the transmission of one MIMO-OFDM frame. All measurements include RF impairments such as carrier-frequency offset and sampling-rate offset.

Based on the figures and definitions presented in Tbl. I, bits per channel use (BPCU) for the bi-directional MAC layer throughput (under the assumption of equal frame length in both directions) can be computed as follows:

$$\text{BPCU} = N_S R Q N_{DSc}. \tag{1}$$

The packet duration t_p can then be computed as

$$t_p = \frac{L_{MAC} + L_{Payload} + L_{FCS}}{\text{BPCU}} t_s. \tag{2}$$

The total frame duration t_f is given by

$$t_f = t_{PHY} + t_p + t_{ifs} \tag{3}$$

where t_{PHY} includes all preambles, training symbols and the signal field. If the frame error rate (FER) is considered, a time-out needs to be defined. The minimum timeout corresponds to $2t_f$ and the average frame transmission duration t_{favg} becomes

$$t_{favg} = (1 - \text{FER}) t_f + \text{FER} 2 t_f. \tag{4}$$

The raw bi-directional MAC layer throughput then corresponds to

$$\Theta = \frac{L_{Payload}}{t_{favg}} \text{ [Mbps]}. \tag{5}$$

The achievable throughput is shown in Fig. 6 for BPSK and QPSK modulation and the two channel scenarios (TGn-A and TGn-C) and is based on the measured FER and

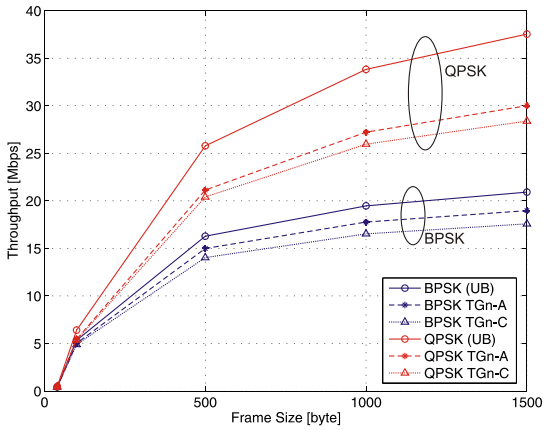

Fig. 6. Measured throughput based on the FER for TGn-A and TGn-C channel scenarios.

computed according to (5). Furthermore, under the assumption of no frame errors, the upper bound (UB) on the achievable throughput is plotted.

Over-the-Air Measurements: Over-the-air measurements were carried out using the Iperf-tool [10]. This software measures the bandwidth and the quality of a network link. For these measurements, a PC was connected to each terminal's Ethernet port on the VAMP board. The Iperf-tool on one PC was used as server, whereas on the other PC, it was configured as client. The two terminals were placed about three to five meters apart. Tbl. II summarizes the measured 4-stream, over-the-air transmission control protocol (TCP) throughput for BPSK, QPSK, and 16-QAM modulation. The listed numbers correspond to the average measured throughput over 100 seconds for each modulation scheme.

TABLE II
MEASURED OVER-THE-AIR TCP THROUGHPUT.

	BPSK	QPSK	16-QAM
Throughput [Mbps]	15	22	25

V. OFFLINE TESTBED

The BAT and WING boards can also be used as offline testbed. In this configuration, most of the signal processing takes place in Matlab. Only the transmission from one terminal to another is carried out in real-time. To this end, the time-domain transmit signal is written into a buffer on the FPGA of the BAT board. When transmitting, the data in the buffer is upsampled and transmitted over the RF chain to another terminal, where the incoming data is downsampled and recorded, before it gets processed in Matlab. This concept is illustrated in Fig. 7 and enables an convenient and flexible assessment of new algorithms or protocols over real-world channels.

VI. CONCLUSION

In this paper, a testbed is presented that serves as platform for real-time MU-MIMO-OFDM experiments. The addition of an embedded MAC layer to the PHY layer of [1] enables experiments involving cross-layer, real-time, and space-division multiple access (SDMA) aspects, such as, e.g., rate

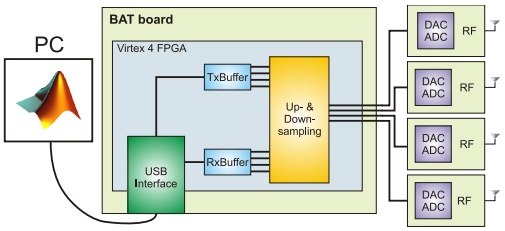

Fig. 7. Block diagram of the offline testbed.

adaptation, feedback of channel state information, or MU scheduling algorithms. Initial measurement results show that reliable bi-directional 4-stream communication is possible. The total available digital signal processing power on the three FPGAs and on the ASIC adapter supports experiments with highest-performance MU-MIMO-OFDM algorithms. Additionally, offline experiments over real-world channels can be carried out on the same hardware platform to assess algorithms in a real-world environment.

ACKNOWLEDGMENT

The authors wish to thank S. Häne for the channel emulator instructions and C. Studer for the invaluable discussions. This work was supported by the STREP project MASCOT (IST-026905) within the Sixth Framework of the European Commission.

REFERENCES

[1] S. Haene, D. Perels, and A. Burg, "A real-time 4-stream MIMO-OFDM transceiver: System design, FPGA implementation, and characterization," *IEEE J. Select. Areas Commun.*, vol. 26, no. 6, pp. 877–889, Aug. 2008.

[2] D. Gesbert, M. Kountouris, R. Heath, C.-B. Chae, and T. Salzer, "Shifting the MIMO paradigm," *IEEE Signal Processing Mag.*, vol. 24, no. 5, pp. 36–46, Sep. 2007.

[3] C. Mehlführer, M. Rupp, F. Kaltenberger, and G. Humer, "A scalable rapid prototyping system for real-time MIMO OFDM transmissions," in *Proc. 2nd IEE/EURASIP Conference on DSP enabled Radio*, Sep. 2005, pp. 7–14.

[4] T. Haustein, A. Forck, H. Gäbler, V. Jungnickel, and S. Schiffermller, "Real-time signal processing for multiantenna systems: Algorithms, optimization, and implementation on an experimental testbed," *EURASIP Journal on Applied Signal Processing*, vol. 2006, 2006.

[5] R. Rao, S. Lang, and B. Daneshrad, "Field measurements with a 5.25 GHz broadband MIMO-OFDM communication system," *IEEE Trans. Wireless Commun.*, vol. 6, no. 8, pp. 2848–2859, Aug. 2007.

[6] K. Nishimori, R. Kudo, Y. Takatoti, A. Ohta, and K. Tsunekawa, "Performance evaluation of 8×8 multi-user MIMO-OFDM testbed in an actual indoor environment," in *IEEE 17th International Symp. on Personal, Indoor and Mobile Radio Comm.*, Sep. 2006, pp. 1–5.

[7] T. Wirth, V. Jungnickel, A. Forck, S. Wahls, H. Gaebler, T. Haustein, J. Eichinger, D. Monge, E. Schulz, C. Juchems, F. Luhn, and R. Zavrtak, "Phy 51-3 - realtime multi-user multi-antenna downlink measurements," in *IEEE Wireless Comm. and Netw. Conf. (WCNC'08)*, Apr. 2008, pp. 1328–1333.

[8] A. Burg, S. Haene, D. Perels, P. Luethi, N. Felber, and W. Fichtner, "Algorithm and VLSI architecture for linear MMSE detection in MIMO-OFDM systems," in *Proc. IEEE Int. Symp. on Circuits and Systems (ISCAS'06)*, May 2006.

[9] V. Erceg and et al., *TGn channel models*. IEEE 802.11-03/940r4, May 2004.

[10] N. L. for Applied Network Research (NLANR)/Distributed Applications Support Team (DAST), "Iperf, measuring maximum TCP and UDP bandwidth performance," 2005, http://dast.nlanr.net/Projects/Iperf/, http://openmaniak.com/iperf.php.