SYSTEM-LEVEL CHARACTERIZATION OF A REAL-TIME 4×4 MIMO-OFDM TRANSCEIVER ON FPGA

Simon Haene, David Perels, and Wolfgang Fichtner

Integrated Systems Laboratory, ETH Zurich, Switzerland
email: {haene,perels,fw}@iis.ee.ethz.ch
web: http://www.iis.ee.ethz.ch/∼mimo

ABSTRACT

The performance of an FPGA-based MIMO-OFDM testbed is investigated through measurements. The setup includes two real-time terminals, supporting up to 4 spatial streams, and a wideband multipath channel emulator. The performance of the system, transmitting at data rates up to 216 *Mbit/s over a* 20 *MHz channel in the* 2.4 *GHz ISM band, is benchmarked under different channel scenarios. The impact of different algorithm choices at the receiver, including parameter estimation for synchronization and channel estimation, on system-level performance is also evaluated. The FPGA implementation results for the different PHY-layer subsystems provide relevant insight into possible tradeoffs between performance and silicon complexity.*

1. INTRODUCTION

Multiple-input multiple-output (MIMO) combined with orthogonal frequency division multiplexing (OFDM) modulation is the key technology for next generation wireless local area networks (WLANs) and will enable a significant increase in throughput and range. While the throughput is maximized by transmitting multiple spatial data streams concurrently (a technique called spatial multiplexing), the transmission reliability, and hence range, is increased by adopting space-time diversity schemes or beamforming.

In parallel to the standardization of IEEE 802.11n, which is expected to become the first MIMO-OFDM WLAN standard, a number of prototypes have been reported. Given the demanding signal processing required by wideband multi-antenna receiver algorithms, most of the real-time testbeds are based on dedicated VLSI circuits prototyped on modern field programmable gate arrays (FPGAs), rather than on digital signal processors. Incidentally, these FPGA circuits also provide information on the main bottlenecks and silicon area expected on application specific integrated circuits, which is the target technology for high-volume consumer products implementing a MIMO-OFDM physical (PHY) layer.

The release of commercial products based on draft versions of the upcoming IEEE 802.11n standard has sparked a debate on the real-world performance of the new generation WLANs. Most of the initial measurements highlight interoperability issues and bottlenecks in the medium access control layer. The system-level measurements in this paper, instead, restrict the focus on the PHY layer, including the effects of finite wordlength digital signal processing, data conversion, and the radio frequency (RF) transceiver.

Contribution: Reproducible measurements, based on a wideband multipath channel emulator, show the impact of various channel conditions on the achievable data rates. Furthermore, the analysis considers the impact of selected receiver algorithms on both throughput and increase in FPGA resources, hence highlighting possible tradeoffs between receiver performance and silicon area.

Outline: The next section briefly describes the general setup and the hardware components of the measured real-time transceiver. In Sec. 3 the PHY layer digital signal processing operations are detailed and FPGA implementation results are reported. Measurement results are presented in Sec. 4, together with an introduction to the adopted performance metric.

2. TESTBED DESCRIPTION

The testbed under consideration is based on the OFDM modulation parameters (such as FFT size, symbols duration, pilot and data tone allocation) defined in the IEEE 802.11a single-antenna WLAN standard [1], but supports spatial multiplexing with antenna configurations up to 4×4. When transmitting four spatial streams, the data rate is increased by a factor of four compared to the single-antenna standard, achieving a peak data rate of 216 Mbit/s.

As shown in Fig. 1, each terminal consists of a host-PC, a peripheral component interconnect (PCI) based FPGA platform for digital signal processing, and a multi-antenna RF transceiver for the signal conversion into the 2.4 GHz industrial, scientific, and medical (ISM) band.

2.1 PCI-Based FPGA Platform

The employed PCI platform hosts three hardware modules: a large FPGA (Xilinx XC2V6000-6) module for baseband signal processing and two converter modules for digital-to-analog and analog-to-digital signal conversion. On an FPGA (Xilinx XC2V1000-4) provided on each of the converter modules, the baseband signals are digitally converted to an intermediate carrier frequency (IF) of 20 MHz. Hence, the total number of converters is reduced by a factor of two compared to traditional baseband systems with separate converters for the I and Q components. Incidentally, the digital IF also avoids any I/Q imbalance, at the cost of higher sampling

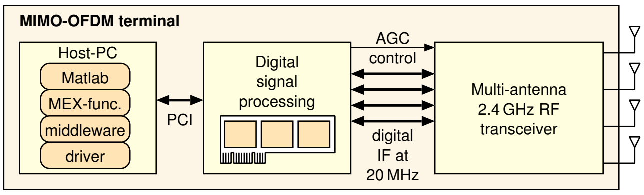

Figure 1: Hardware components of a single MIMO-OFDM terminal.

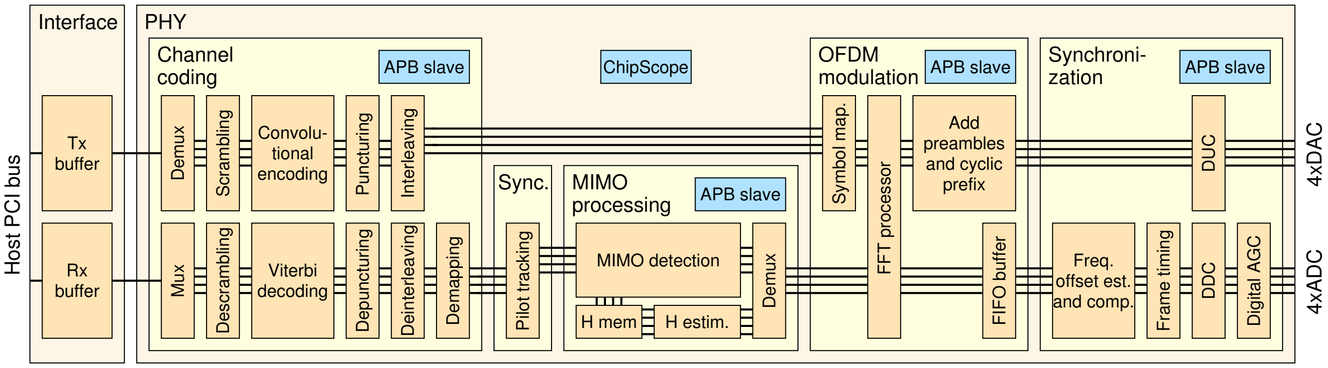

Figure 2: Digital signal processing architecture overview.

rate and resolution. Both the 14 bit digital to analog converters (DACs) and the 12 bit analog to digital converters (ADCs) run at a sampling rate of 80 MS/s.

2.2 Analog RF Frontend

Each terminal is equipped with four single-input single-output (SISO) super-heterodyne RF chains with an analog IF of 475 MHz. The chains are built from discrete RF components and support 20 MHz bandwidth channels with center frequencies in the 2.4 GHz ISM band. The received signal power at the ADCs is regulated by digitally controllable analog attenuators. These attenuators extend the dynamic range of the receive chains by 31 dB. In conjunction with the 12 bit resolution of the ADCs, the useful dynamic range amounts to approximately 70 dB.

2.3 Software Stack

The testbed is controlled from a host-PC through multiple layers of software that allow communication with the hardware. The software stack includes a hardware driver for the PCI board, a transceiver-specific API that provides several functions (for the digital baseband configuration and the transmission or reception of OFDM frames), and a Matlab interface (MEX) function to call these API functions. Demo applications, configuration tools, and measurement sequences are programmed as Matlab scripts.

3. DIGITAL SIGNAL PROCESSING ARCHITECTURE

The digital signal processing includes the MIMO-OFDM PHY layer and an interface to the host-PC. The architecture is shown in Fig. 2, where the PHY layer is partitioned into channel coding, OFDM modulation, synchronization, and MIMO processing. A standard bus interface (AMBA APB) enables run-time configuration and status monitoring of the individual functional units. Additionally, hardware cores generated by a Xilinx software tool (ChipScope) are inserted for in-circuit signal monitoring.

3.1 OFDM Modulation

The OFDM modulation unit includes the mapping of binary data to complex-valued constellation points that determine how amplitude and phase of each OFDM subcarrier are modulated, the computation of the superposition of all modulated subcarriers with an IFFT, and the insertion of cyclic prefixes.

The subcarriers are QAM-modulated according to different modulation orders (ranging from BPSK to 64-QAM), depending on the desired data rate and transmission robustness. A 64-point I/FFT processor with a single radix-4 processing element [2] de/modulates the OFDM symbols. The different spatial streams are processed in a time-interleaved fashion on the same hardware unit. Each frame starts with a preamble that is divided into two parts: the first part consists of special OFDM symbols for frame start detection, automatic gain control (AGC) setting, and frequency offset estimation. The second part includes an OFDM training symbol for each transmit antenna and is dedicated to channel estimation.

3.2 MIMO Processing

The MIMO processing unit estimates the channel matrices and handles the separation of the spatially multiplexed data streams.

The adopted block-type channel training sequence is designed to be frequency orthogonal, so that, given an OFDM subcarrier, a single transmit antenna is sounding this particular tone exclusively. Which of the antennas is active depends on the OFDM symbol index. With four such training symbols, the frequency-domain channel coefficients of all spatial subchannels can be estimated on a tone-by-tone basis. In order to exploit the correlation between frequency-domain channel coefficients, the initial estimate can optionally be refined with an FFT-processor-based algorithm [2]. The resulting channel matrices are stored in a dedicated memory.

A linear minimum mean squared error (MMSE) detector spatially equalizes the received data streams on a tone-by-tone basis. The same hardware unit [3] computes the MMSE equalizer matrices and applies them to the received vector-symbols. The complex-valued equalizer matrices are stored in place of the channel matrices, thereby minimizing memory requirements. During the latency incurred by the computation of the MMSE estimators, the incoming data is buffered in a FIFO buffer (shown in the OFDM modulation unit in Fig. 2). The MMSE unit implements direct matrix inversion and has therefore stringent numerical requirements. Hence, a tone-by-tone conversion to block-floating point number representation of the channel matrices guarantees the numeric precision of the matrix-inversion unit being optimally exploited also when the transceiver is operating in frequency-selective wireless channels. The application of the computed estimators requires matrix-vector multiplications and the results are converted back to fixed-point representation.

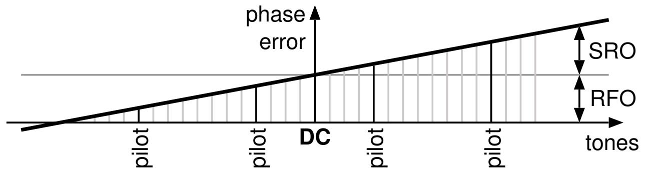

Figure 3: Phase errors on tones of an equalized OFDM symbol due to sample rate offset (SRO) and residual frequency offset (RFO).

Table 1: Required FPGA resources

XC2V6000-6 FPGA				
Block	**Slice**	**%Slice**	**Mult**	**Ram**
Synchronization	5038	17.6	17	5
OFDM modulation	2879	10.1	12	44
MIMO processing	8847	31.0	32	10
Channel decoding	9082	31.8	4	5
Others	2725	9.5	0	70
Total	**28571**	**100**	**65**	**134**

XC2V1000-4 FPGAs				
Block	**Slice**	**%Slice**	**Mult**	**Ram**
DUC	980	19.5	20	0
DDC	997	19.8	6	0
AGC	940	18.7	4	0
Others	2113	42.0	0	39
Total	**5030**	**100**	**30**	**39**

3.3 Frontend and Synchronization

During transmission, the digital baseband signals are I/Q modulated, up-sampled, and filtered by digital up-conversion (DUC) to an IF of 20 MHz before the resulting real-valued signals are converted to the analog domain. The same IF is adopted in the receiver, where digital down-conversion (DDC) performs the inverse operation.

The AGC operates both in the analog and in the digital domain. The received signal power at the input of the ADCs is controlled over the analog attenuators in the RF chains. Additionally, in case of weak input levels, the signal is amplified digitally to optimally exploit the dynamic range of the digital receiver and to minimize the effect of quantization noise. Signal power estimation is based on preambles that are constructed to be frequency orthogonal, i.e., so that only one transmit antenna is sounding a given tone [4].

Frame-timing synchronization is based on the detection of a sudden power increase in the received signals [5]. The detection reliability is improved, in particular at lower signal-to-noise ratios (SNRs), taking into account an additional metric that confirms the distinctive periodicity of the preamble.

Frequency offset estimation is based on the autocorrelation algorithm presented in [6] extended for the MIMO case and operates on the first part of the frame preamble. In particular, a lower estimation variance is achieved, compared to the SISO case, by combining the correlations obtained on different antennas. Frequency offset compensation requires a digital oscillator, whose period depends on the estimated frequency offset. The uncompensated signals are combined with the rotating phasor by a single complex-valued and time-shared multiplier.

The error incurred during the preamble-based estimation leads to residual frequency offset and causes a rotation of the spatially equalized constellation points. This phase deviation, which increases with the OFDM symbol index but is constant over the subcarriers, can be compensated after the MMSE equalizer by derotating all constellation points in an OFDM symbol according to the phase of the BPSK pilot tones [1], which are known to the receiver. Another impairment to be compensated by means of the pilot tones is sampling rate offset, generated by small differences in the DAC and ADC sampling frequencies at the transmitter and receiver. This causes an additional phase error on the received constellation points that increases linearly with the subcarrier index, as shown in Fig. 3. Two pilot-tracking algorithms are evaluated through measurements.

The first algorithm estimates the offset and the slope of the curve in Fig. 3 by observing the phase errors on the pilot tones, computed by means of CORDIC units [7]. A different compensating phasor is computed for each subcarrier and a complex-valued multiplier, that is time shared among the spatial streams, applies the computed phase corrections.

A second pilot-tracking algorithm that compensates frequency offset only is examined for comparison: a single phasor is computed for all subcarriers by arithmetically averaging the equalized pilot constellation points, weighted negatively if -1 was transmitted on the corresponding tone. This requires only a small number of additions and hence reduces circuit complexity significantly, compared to the CORDIC-based solution. The derotation is achieved by multiplying all subcarriers with the complex-conjugate version of this phasor. This approach reduces silicon complexity at the cost of tracking precision.

3.4 Channel Coding

As forward error correction scheme, bit-interleaved coded modulation (BICM), based on the convolutional code, interleaving, and puncturing schemes defined in IEEE 802.11a [1], is employed. Per-antenna convolutional coding with subsequent cross-antenna interleaving is adopted in order to enable the real-time decoding on Virtex-II FPGAs, following the approach in [8]. The Viterbi decoder accepts 5 bit wide soft-information and has a traceback length of 54. Bit-scrambling removes long sequences of 0s or 1s in the transmitted payload data. The de/interleavers rely on built-in block RAMs for the bitwise interleaving.

Low-complexity soft-information is extracted, based on the MMSE-equalized and pilot-tracked constellation points, handling each spatially multiplexed data stream independently, according to [9]. For performance comparison, hard-metrics generated by slicing the equalized vector symbols can also be computed instead of soft-metrics.

3.5 FPGA Implementation Results

The entire digital signal processing (as shown in Fig. 2) is running on a single XC2V6000-6 FPGA, with the exception of DUC, DDC, and digital AGC, which are mapped onto two separate XC2V1000-4 FPGAs on the corresponding data

Table 2: Impact on required FPGA resources for different optional receiver algorithms

Algorithm	Slice	%Slice	Mult	Ram
Refined channel est.	710	2.5	-	-
Soft-info extraction	625	2.2	4	2
Block-floating point	454	1.6	-	-
Improved pilot track.	764	2.7	1	1
Total	**2553**	**9.0**	**5**	**3**

Table 3: Considered 4×4 data rates R in [Mbit/s]

Coding rate	Subcarrier modulation			
	BPSK	QPSK	16-QAM	64-QAM
1/2	24	48	96	144
2/3	-	-	-	192
3/4	36	72	144	216

converter modules. The required FPGA resources for both designs are detailed in Tbl. 1, where "Mult" and "Ram" refer to the built-in real-valued 18×18-bit multipliers and 18 kbit SRAM blocks, respectively. The resources reported under "Others" refer to PCI interface, ChipScope cores, intermodule-communication interfaces, and other functional units that are not in strict relation with the PHY layer. The main system clock frequency is 80 MHz, but some parts of the design run at 40 MHz and 160 MHz.

The PHY layer needs a total of 64 RAMs, whereof 32 for the FIFO buffer required to bridge the latency incurred by MIMO preprocessing and channel estimation. This buffer was intentionally oversized (by at least a factor of two) to enable special debug modes that allow access to the received data after the FFT or after the pilot tracking. Hence, a significant amount of memory could be saved, in a real-world system, by eliminating these debug modes.

Channel estimation refinement, extraction of soft-information, block-floating point conversion of channel matrices, and improved pilot-tracking are optional receiver algorithms. Their impact on system performance is reported in the next section while their contribution to the required FPGA resources in Tbl. 1 is given in Tbl. 2. The slice percentage in this second table refers to the total number of occupied slices on the XC2V6000-6 FPGA.

4. MEASUREMENTS

Measurements have been carried out with two real-time terminals communicating over a wideband multipath channel emulator with RF interfaces. The measurement operations are coordinated over a wired TCP/IP network by the receiver terminal, which controls the channel emulator and triggers the transmitter to send data frames. Even though smaller antenna configurations are supported, the focus is on the 4×4 transmission modes listed in Tbl. 3.

4.1 Performance Metric

The average throughput is adopted as performance metric and computed according to the following procedure. A channel realization labeled k is drawn according to the desired statistics and uploaded to the emulator. For all the data rates $R[m]$ in Tbl. 3, multiple OFDM frames carrying 500 bytes payload data each are transmitted over the selected channel realization. By checking the received data, an empirical frame error rate $\mathrm{FER}[m,k]$ is computed for each of the transmission modes. Assuming genie-aided adaptive modulation, the best throughput $D[k]$ over the specific channel realization under consideration is given by

$$D[k] = \max_m R[m]\big(1 - \mathrm{FER}[m,k]\big). \tag{1}$$

In order to compute $\mathrm{FER}[m,k]$, 20 frames are transmitted per transmission mode and channel realization. Finally, the performance metric is obtained by averaging $D[k]$ over 500 channel realizations.

Performance charts are obtained by computing the average throughput for different receive powers. The MIMO-OFDM frames at the channel emulator input, generated by the transmitting terminal, have a constant power level. After applying the channel model, the average power at the RF inputs of the receiver is controlled by adjusting the output gain of the emulator. All measurements were taken with 100 kHz frequency offset and 1.6 kHz sample-rate offset, which corresponds to a 20 ppm frequency precision.

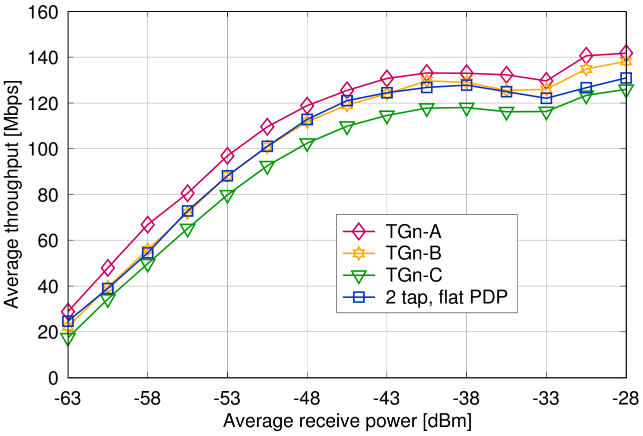

Figure 4: Average throughput for different channel scenarios.

4.2 Performance for Different Channels

The measured average throughput under different block-fading channel scenarios is shown in Fig. 4. The TGn channel models A, B, and C [10] were generated assuming an antenna spacing of 2 wavelengths at the transmitter and at the receiver. For comparison a spatially uncorrelated Rayleigh-fading channel with flat power delay profile and 2 taps spaced 50 ns apart is also considered. From the chart in Fig. 4 it can be observed that the system suffers from increasing channel lengths, even though the impact of different channel types is not very significant.

Even though the highest PHY layer data rate is 216 Mbit/s, the measured average throughputs saturate between 120 Mbit/s and 140 Mbit/s. This can be attributed to channel emulator and multi-antenna RF transceiver, which limit the achievable receiver SNR. Additionally, the word-width of the multipliers in the MMSE equalizer must also be increased for SNRs beyond 30 dB [3].

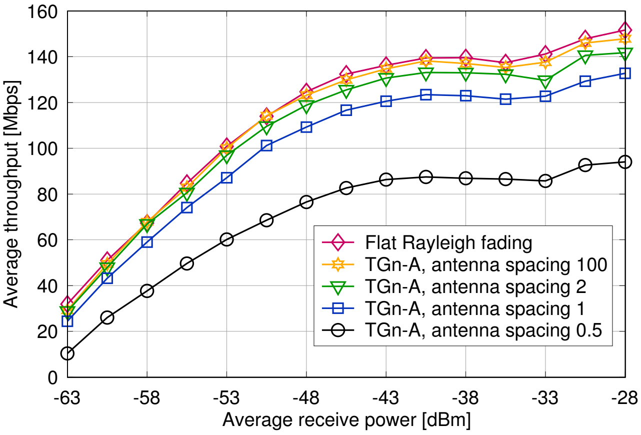

Figure 5: Impact of different antenna spacings (in wavelengths) on the average throughput achieved over TGn-A.

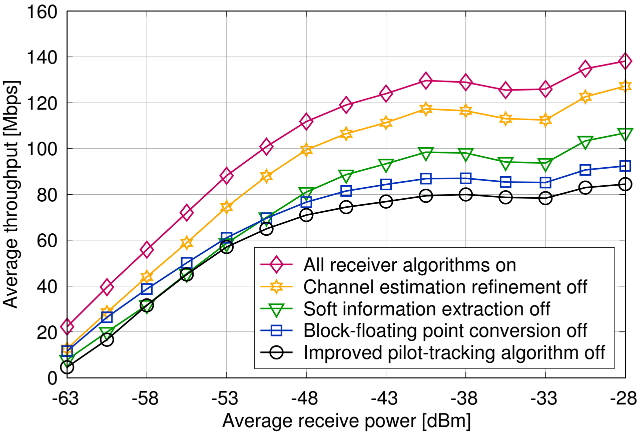

Figure 6: Impact of different receiver algorithms on the average throughput achieved over a TGn-B channel. The topmost curve refers to the performance when all receiver algorithms are enabled. With each additional algorithm that is disabled, the performance degrades further.

4.3 Impact of Antenna Correlation

When reducing the antenna spacing, the correlation between the received signals increases and MIMO equalization with a linear MMSE equalizer becomes less reliable. The impact of this correlation is shown in Fig. 5 for the particular case of a TGn-A channel model, where the antenna spacing is set to 100, 2, 1, and 0.5 wavelengths. For comparison a flat Rayleigh-fading channel, corresponding to the special case of TGn-A without any spatial correlation, is considered. As expected, the system suffers significantly when the antenna spacing is reduced below one wavelength.

4.4 Impact of Receiver Algorithms

The impact of different receiver algorithms on the average throughput achieved over a TGn-B channel is shown in Fig. 6. The topmost curve corresponds to the performance when all receiver algorithms, including the ones listed in Tbl. 2, are enabled. For comparison, the performance was measured after disabling the four optional signal processing algorithms, one after the other. From Tbl. 2 and the curves in Fig. 6, it can be seen that a significant increase in performance is achieved with less than 10% hardware overhead.

5. CONCLUSION

A real-time MIMO-OFDM physical layer transmitting at a peak data rate of 216 Mbit/s over 20 MHz bandwidth was prototyped and characterized through measurements. Compared to a SISO system many functional units for synchronization, OFDM modulation, and channel coding units are replicated for each spatial stream. The equalization of spatial streams, instead, is specific to MIMO receivers and requires about 30% of the FPGA slices and 50% of the multipliers with linear MMSE detection.

The measured average data rates are clearly beyond those achievable by single-antenna IEEE 802.11a WLAN systems, which transmit over the same bandwidth but are limited to a peak data rate of 54 Mbit/s. The system performance is affected by increasing channel lengths and by increasing antenna correlation. As expected, the MIMO gain is reduced when the antenna spacing is not sufficient.

By investigating selected receiver algorithms, including parameter estimation for synchronization and channel estimation, the impact of more sophisticated signal processing has been shown to be significant. With thorough hardware-oriented algorithm design, the implementation of the considered algorithms increases the overall silicon complexity by less than 10%.

REFERENCES

[1] *IEEE 802.11a Standard*, iSO/IEC 8802-11:1999/Amd 1:2000(E).

[2] S. Haene, A. Burg, P. Luethi, N. Felber, and W. Fichtner, “FFT processor for OFDM channel estimation,” in *Proc. IEEE ISCAS*, 2007.

[3] 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 ISCAS*, 2006.

[4] D. Perels, A. Burg, S. Haene, N. Felber, and W. Fichtner, “An automatic gain controller for MIMO-OFDM WLAN systems,” in *Proc. IEEE ICCSC ’06*, vol. 1, 2006, pp. 55–60.

[5] D. Perels, C. Studer, and W. Fichtner, “Implementation of a low-complexity frame-start detection algorithm for MIMO systems,” in *Proc. IEEE ISCAS*, 2007.

[6] T. M. Schmidl and D. C. Cox, “Robust frequency and timing synchronization for OFDM,” *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1613–1621, Dec. 1997.

[7] B. Parhami, *Computer Arithmetic, Algorithms and Hardware Design*. Oxford University Press, 2000.

[8] S. Haene, A. Burg, D. Perels, P. Luethi, N. Felber, and W. Fichtner, “FPGA implementation of Viterbi decoders for MIMO-BICM,” in *Proc. 39th IEEE Asilomar Conf. Signals Syst. Comput.*, 2005.

[9] I. B. Collings, M. R. G. Butler, and M. McKay, “Low complexity receiver design for MIMO bit-interleaved coded modulation,” in *IEEE Int. Symp. on Spread Spectrum Techniques and Applications*, 2004, pp. 12–16.

[10] V. Erceg *et al.*, *TGn Channel Models*, IEEE 802.11 document 03/940r4.