

# 80-Mb/s QPSK and 72-Mb/s 64-QAM Flexible and Scalable Digital OFDM Transceiver ASICs for Wireless Local Area Networks in the 5-GHz Band

Wolfgang Eberle, *Member, IEEE*, Veerle Derudder, Geert Vanwijnsberghe, Mario Vergara, Luc Deneire, *Member, IEEE*, Liesbet Van der Perre, Marc G. E. Engels, *Member, IEEE*, Ivo Bolsens, *Member, IEEE*, and Hugo De Man, *Fellow, IEEE*

**Abstract**—With the advent of mobile communications, voice telecommunications became wireless. Future applications, however, target multimedia, messaging, and high-speed internet access, all expressing the need for a broadband high-speed wireless access technique. Both the domestic multimedia and the wireless local area network (WLANs) business markets are addressed. Established systems deliver 2–11 Mb/s based on spectrally inefficient spread-spectrum techniques, where scalability has reached a limit. The next generation of modems requires spectrally more efficient low-power and highly integrated solutions. We describe here the design of two digital baseband orthogonal frequency division multiplex (OFDM) signal processing ASICs, implementing respectively a quaternary phase-shift keying (QPSK)-based 80-Mb/s and a 64 quadrature amplitude modulation (QAM)-based 72-Mb/s digital inner transceiver. The latter partially matches the Hiperlan/2 and IEEE 802.11a standards. Joint development of signal processing algorithms and architectures along with on-chip data transfer, control, and partitioning leads to a low-power, yet flexible and scalable implementation. Both ASICs were designed in a unique object-oriented C++ design flow starting from algorithm level. The ASICs were successfully tested in a 5-GHz testbed both for file data transfer and web-cam multimedia transmission.

**Index Terms**—Adaptive equalization, burst acquisition, digital signal processing, orthogonal frequency division multiplex, wireless local area networks, wireless transceiver.

## I. INTRODUCTION

WIRELESS digital communication in indoor environments is gaining interest due to its inherent flexibility and mobility advantages. There is both a consumer market for connecting domestic appliances and multimedia without wires and a business market segment for broadband wireless networking. WLANs have a deployment advantage for the fine-grain indoor communication even if combined with a wired

access network such as xDSL to the home or a high-speed company backbone. Spectrum allocation in the 5-GHz range and standardization of up to 54-Mb/s 64 quadrature amplitude modulation (QAM) systems in IEEE 802.11a [1] for the USA and ETSI Hiperlan/2 [2] in Europe have accelerated the migration from research results into implementable solutions. Three bands, at 5.15–5.35, 5.47–5.725, and 5.725–5.825 GHz, are regionally available with subdivision into 20-MHz-wide channels and frequency division multiple access (FDMA).

The performance of wireless indoor networks is limited by the communication channel, which distorts the signal due to reflections and scattering. Two-dimensional (2-D) ray-tracing simulations show, in alignment with measured data [3], that the in-house channel is characterized by a rms delay spread  $\tau_{\text{RMS}}$  of 5 to 40 ns. The corresponding frequency-domain channel response shows frequency dips of up to 30 dB. The coherence bandwidth is between 5 and 25 MHz which, compared to the 20-MHz channel bandwidth, reveals the frequency selective nature of the indoor channel. Fortunately, the indoor channel can be considered quasi-static due to limited object movements. This votes in favor of orthogonal frequency division multiplex (OFDM) due to its capability to resolve spectral frequencies quite accurately compared to a time-domain-based approach.

OFDM [4] is a special case of multicarrier transmission. Modulation and frequency spacing are efficiently performed by an inverse fast Fourier transform (IFFT) on a set of constellation symbols at the transmitter and demodulated with an FFT at the receiver. A cyclic prefix (CP) is inserted between subsequent OFDM symbols. It serves as a guard interval, thus expensive intersymbol interference compensation is not needed. It also transforms the plain OFDM symbol into a pseudocyclic one, which avoids leakage in the FFT in case of group delay or synchronization errors. This reduces equalization cost compared to a multitap high-resolution adaptive time-domain equalizer.

In this paper, we describe the design of two digital ASICs both implementing OFDM inner transceivers to be extended by payload error correction coding only to represent the entire physical layer functionality. Typical data rates are up to 80 Mb/s in quaternary phase-shift keying (QPSK) for the Festival ASIC [5] and up to 72 Mb/s in 64-QAM for the Carnival ASIC [6] after encoding. The 64-QAM design features novel signal processing such as an interpolating frequency domain equalizer together with a CP-based clock offset compensation to achieve the desired receiver performance. Both ASICs contain a robust, yet

Manuscript received March 15, 2001; revised June 4, 2001.

W. Eberle and H. De Man are with the Inter-University Microelectronics Center (IMEC), B-3001 Leuven, Leuven, Belgium, and also with the Katholieke Universiteit Leuven, Belgium.

V. Derudder, G. Vanwijnsberghe, L. Deneire, L. Van der Perre, and M. G. E. Engels are with the Inter-University Microelectronics Center (IMEC), B-3001 Leuven, Belgium (e-mail: wolfgang.eberle@imec.be).

M. Vergara was with the Inter-University Microelectronics Center (IMEC), B-3001 Leuven, Belgium. He is now with Ericsson Mobile Communications AB, Lund, Sweden.

I. Bolsens was with the Inter-University Microelectronics Center (IMEC), B-3001 Leuven, Belgium. He is now with Xilinx, Inc., San Jose, CA 95124 USA.

Publisher Item Identifier S 0018-9200(01)08226-9.

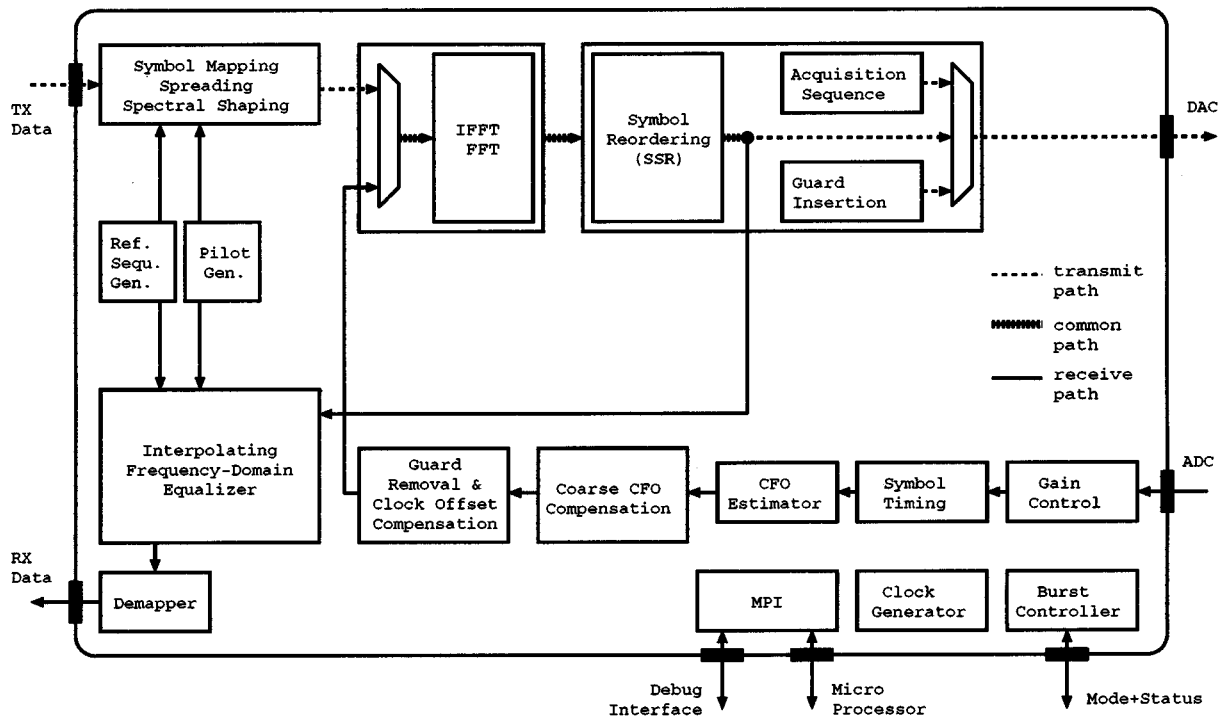


Fig. 1. The top-level block diagram is identical for both festival and carnival. The half-duplex transceivers however differ strongly in their DSP functionality.

programmable acquisition scheme that meets the specific needs of low-overhead fast and accurate burst acquisition going far beyond requirements of other OFDM-based systems such as wire-line xDSL [7], digital video (DVB) [8], and digital audio broadcasting (DAB) [9].

We start from the system perspective describing transmission and reception path. Next, we illustrate the flexibility of the designs and the choice for a multiprocessor architecture with distributed token-based control, followed by a detailed analysis of the major signal-processing algorithms and their architectural implementation. Next, system issues such as integration, clocking, and on-chip communication are addressed. The section on design methodology focuses on the CAD challenges when jointly combining algorithmic exploration and architectural refinement in an object-oriented C++ environment. We finally come to a characterization and a comparison of both ASICs derived from measurements and system test results.

## II. SYSTEM VIEW

Both ASICs implement the inner transmitter and receiver datapath (Fig. 1) required for a high-speed wireless OFDM system employing a half-duplex protocol suitable for standard-compliant time-division duplex operation. Hardware resources such as the fast Fourier transform (FFT) are shared between transmitter and receiver, and various datapath reordering tasks are merged into a centralized datapath unit (symbol reordering).

A burst controller (BC) allows self-controlled processing of entire transmission bursts and reception bursts, reducing the load of an external medium access control (MAC) or general-purpose processor. The transceiver only requires initial programming of parameters and triggering of MAC requests

for transmission and reception and delivers status information through a dedicated BC interface.

The ASICs communicate through a first-in-first-out (FIFO)-based transmit and receive interface as a master with the data host in a slave position. Toward the front-end, they provide I/Q interfaces to dual pairs of analog-to-digital converters (ADCs) and digital-to-analog converters (DACs). Additional signals are provided to support analog automatic gain control in the receiver and front-end power-up.

In transmission mode, payload data enters the ASIC through a 6-b parallel interface on request. Data enters the symbol mapper where bits are mapped onto either BPSK, QPSK, 16-QAM, or 64-QAM subcarriers. A programmable number of zero carriers is introduced near dc or Nyquist frequency to accommodate dc notch filtering and lowpass filter rolloff. A BPSK pilot sequence is inserted either on a fixed subset of four carriers or using a rotating pilot pattern with a period of 13 OFDM symbols. Each subcarrier can be individually weighted by a complex value allowing transmitter preemphasis and phase predistortion. The mapper provides a sequential series of 64 carriers, for Festival also 128 or 256 carriers, to the IFFT, denoted as an OFDM symbol. The mapper also adds an entire programmable BPSK OFDM symbol serving as a reference sequence prior to the payload or inserts it periodically into the stream of OFDM symbols. The inverse FFT transforms the frequency domain constellation into a time-domain sequence. Scaling and digital hard clipping is performed at the FFT output to select a suitable peak-to-average power ratio (PAPR) and signal-to-noise ratio. OFDM symbols are then passed to the symbol reordering unit (SSR), which inserts the acquisition preamble and the cyclic prefix. The SSR sends data sampled at the chip clock frequency through a  $2 \times 8$ -b parallel I/Q interface to, e.g., an external DAC pair

or a digital low-IF upconversion stage. Setting the ASIC clock frequency to 20 MHz results in a standard-compliant stream of OFDM symbols.

In reception mode, data is provided from an external ADC pair or a digital low-IF down-conversion stage in  $2 \times 10$ -b format to the gain control and timing synchronization stage. The preamble serves to estimate gain, frame start, and carrier frequency offset (CFO). Before entering the FFT, the CFO on incoming samples is reduced to about  $\pm 4$  kHz, resulting in negligible leakage effects. Also, the guard interval is stripped off, forming again plain OFDM symbols of 64, 128, or 256 subcarriers. The FFT translates them into the frequency domain where the SSR removes zero carriers, and identifies pilot carriers and reference symbols.

Payload-carrying subcarriers are passed to the equalizer along with this extracted information. The equalizer performs an initial channel estimate, based on the BPSK reference symbol, which, in Carnival, is improved by interpolation. At that moment, the acquisition phase has finished and the data reception and tracking phase starts. During the tracking phase, received data is still being compensated by the time-domain CFO. The FFT timing is controlled and updated by a clock offset estimation and compensation evaluating the cyclic prefix. Fine frequency-offset compensation is performed in the equalizer in a decision-directed averaging phase loop updating the channel. Also, time-variations of the channel are traced by means of the pilot scheme, where rotating pilots outperform fixed pilots at the same cost.

The equalizer divides the received constellation by its channel response per subcarrier and provides, through the demapper, either hard decision,  $2 \times 3$ -b soft decision, or  $2 \times 6$ -b high-resolution output to, e.g., an external decoder/interleaver block.

The chips feature an asynchronous microprocessor interface for programming. An additional 5-pin direct control interface allows the MAC to select one out of four operational modes (transmit, receive, programming, and sleep) and watch the status of those modes. Any interunit data bus can be monitored parallel and at full clock speed through an external test interface. For example, this bus can provide an adaptive loading extension or a decoder with the channel estimates. Table I describes the major programming parameters for the two ASICs. An OFDM symbol structure compliant with IEEE and ETSI standards can be achieved by choosing 64 carriers, 16 guard samples, 0 zero carriers near dc, 5 zero carriers near Nyquist, fixed pilot scheme, and a frequency diversity factor of 1.

### III. JOINT ALGORITHM AND ARCHITECTURE DESIGN

In this section, our focus will be on the algorithms and architectures of the major signal processing parts of the OFDM transceiver. We start with the FFT, move on to the novel centralized symbol reordering unit, address time-domain-based burst acquisition, and finally, equalization and tracking in the receiver.

#### A. Fast Fourier Transform (FFT)

The complex FFT is the heart of the OFDM system, converting frequency-domain constellations to time domain and vice versa. The high PAPR of multicarrier signals requires

TABLE I  
FESTIVAL AND CARNIVAL: HIGHLY FLEXIBLE PROGRAMMABLE ASICs

Parameter	Options for Festival QPSK modem	Options for Carnival 64-QAM modem
Number of carriers	64,128,256	64
Cyclix prefix	0:4:28	
Zero carriers near DC	0:1:3	
Zero carriers near Nyquist	0:2:14	0:1:15
Modulation schemes	QPSK	BPSK, QPSK, 16-QAM, 64-QAM
Initial channel estimate	known single OFDM reference symbol	known single OFDM reference symbol with smoothing
Tracking	single-carrier decision-directed  or average carrier phase compensation	forward scheme with fixed or rotating pilots with average carrier phase compensation  optionally periodic reference symbols  smoothing with prog. complex coefficients  cyclic prefix based clock offset tracking
Acquisition	prog. binary code sequence (12:4:20 bits), prog. preamble structure (code repetition), prog. reliability thresholds	
	2 x 64,128,256,512 sequence for CFO estimation	2 x 64,128,256 sequence for CFO estimation
Transmit mask	complex coefficient per carrier	
Frequency domain spreading	prog. codes of length 1,2,4,8	prog. codes of length 1,2,4

careful fixed-point exploration to maximize the performance/cost ratio. Wireless burst operation requires an FFT with low latency and power consumption.

A pipelined complex FFT architecture (Fig. 2) based on radix 2–2 decomposition [10] has been chosen since it achieves both the simplicity of butterflies from a radix-2 scheme and the low number of  $\log_4(N) - 1$  complex multipliers from a radix-4 scheme. Every other multiplier is replaced by rotator logic involving only multiplexing and sign inversion. Using simple butterflies and less multipliers also simplifies control and allows a straight forward design of a variable 64, 128, 256-length FFT. IFFT operation is obtained by conjugation of input and output signals.

The radix 2–2 scheme requires the minimum amount of  $N - 1$  memory locations. Memory is implemented as feedback register banks or dual-port RAMs (128- and 256-word banks only) distributed along the pipeline starting with the maximum wordcount according to a decimation-in-frequency scheme.

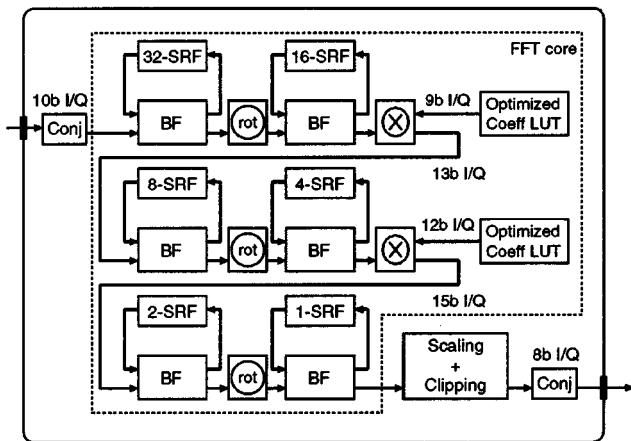


Fig. 2. A radix 2-2 scheme significantly reduces the arithmetic complexity of the fast Fourier transform.

We benefit from the fact that the wordlength through the FFT increases toward the output starting with a small input wordlength, saving 25% memory in 64-carrier mode compared to decimation in time. Compared to a fixed-wordlength implementation, we achieve a reduction of 30% in memory size from the fact that we start with 10 b and end with 15 b. We introduce a fixed scaling by 2 at every butterfly stage, so wordlength increases only at every full multiplier. To derive the two unknowns per multiplier, i.e., the post-multiplier datapath wordlength and the coefficient lookup table (LUT) wordlength, we performed a parametric exhaustive search by simulation [11]. This search becomes feasible since we have reduced the unknown wordlengths to only four in the 64/128-carrier and six in the 256-carrier case.

Scaling and saturation at the output stage facilitate the implementation of digital hard  $I/Q$  amplitude clipping in the transmitter. The choice between 5-b up to 8-b outputs offers dynamic ranges from 30 to 48 dB.

There is a latency of 1 OFDM symbol between the input and the output. In addition, the final FFT implementation has a core delay of ten clock cycles resulting from one pipeline stage per butterfly and two per complex multiplier. The FFT provides its output in bit-reversed order with post-compensation in the SSR.

### B. Centralized Symbol Reordering for Data Transfer Optimization

OFDM symbols are metasymbols compared to conventional single-carrier samples. This inherent scalability makes OFDM powerful. However, to exploit this flexibility, reconfigurable architectures supporting a discrete set of parameter choices are required. In a conventional distributed design process, the design would be first partitioned into modules and then optimized locally per module. Based on a high-level dataflow description, we have analyzed data transfer between signal processing tasks, their intraunit storage and interunit buffering requirements to handle multirate issues. The flexibility in the OFDM symbol structure leads to a large set of I/O rate ratios. More specifically, we encountered buffering issues due to bit-reversed reordering of the FFT output, removal of pilots and zero carriers, despreading, insertion of the programmable

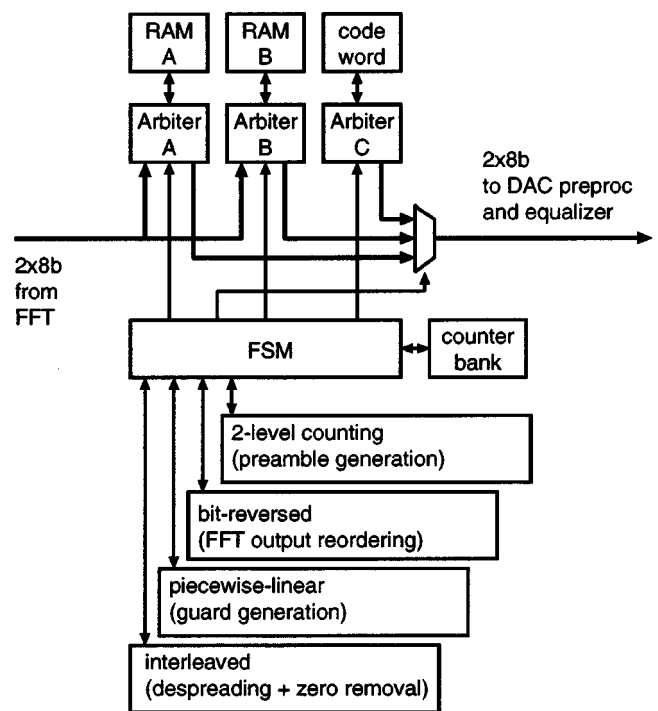


Fig. 3. Symbol-based sample reordering (SSR) unit essentially allows a set of intrasymbol data transfer operations based on a generic architecture.

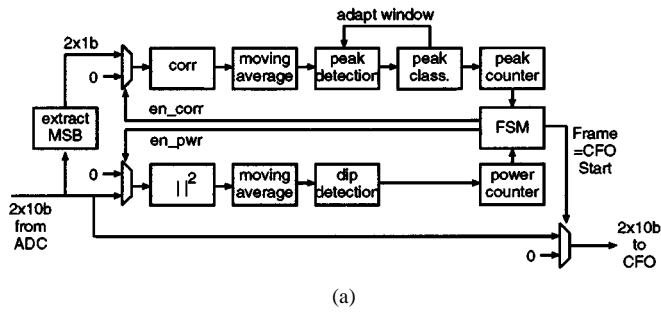
length cyclic prefix and the preamble. Instead of foreseeing distributed buffers which would require worst-case sizing, we centralized the storage in a single unit (Fig. 3) consisting of two single-port RAMs with memory arbiters and a set of address generators. Two address generators run in parallel, producing read and write addresses, respectively. RAM access mode is toggled after every OFDM symbol. This approach results in the minimum amount of memory, i.e., twice the subcarrier number, without additional latency.

### C. Fast Time-Domain Burst Acquisition

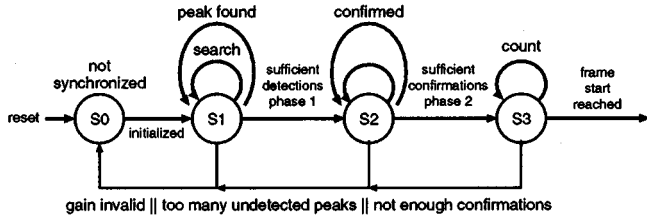
Wireless LAN systems depend on fast burst acquisition to minimize transmission overhead at the physical layer. At the same time, the received signal is distorted by a number of indoor channel and front-end effects. Receiver acquisition has to detect the incoming signal, adapt its signal power, achieve timing synchronization, and compensate for CFO introduced by local oscillator mismatches in transmit and receive front-ends.

Fast acquisition prohibits the use of frequency domain signal processing for timing synchronization and CFO estimation, popular in wire-bound systems with long acquisition preambles like ADSL or wireless broadcastings like DAB and DVB, which are not packet-based and where initial data loss can be tolerated.

We have implemented a timing acquisition [Fig. 4(b)] based on a two-phase autocorrelation process [Fig. 4(a)] using a programmable BPSK time-domain code sequence which is repeated according to a second metalevel sequence. Since the sliding window correlator only requires a  $2 \times 1$ -b input, it is very robust against automatic gain control transients and implementable with low area and power cost. A parallel



(a)



(b)

Fig. 4. Robust  $2 \times 1$ -b timing acquisition relies on preamble autocorrelation in combination with signal power monitoring. (a) Datapath. (b) Controller FSM.

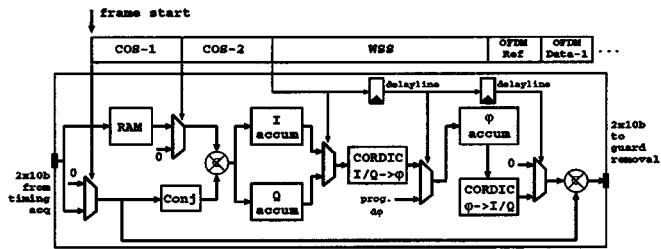


Fig. 5. The carrier frequency offset estimate feeds a phase accumulator and a CORDIC to limit CFO before entering the FFT.

sliding window signal power estimation is used to validate the correlator results. Alternating bipolar correlation peaks during phase 1 determine the relative code sequence start, while the transition to phase 2 defines the absolute frame reference. The receiver only uses information on the codeword length and the metalevel sequence; the codeword itself is not known. Probabilities of false alarm and missing detection depend on the programmed numbers of detected peaks in phase 1 and phase 2, respectively. Phase 3 counts until the frame start when phase 2 has obtained enough confirmations.

Carrier offset is estimated based on a repeated sequence of length 64, 128, 256, or 512, which follows the frame start, based on autocorrelation for multipath immunity reasons. A larger preamble size trades off a higher noise suppression against a lower capture range. Carrier offset must be reduced to a fraction, e.g., 1%–2%, of the 312.5-kHz subcarrier spacing, to achieve negligible intercarrier interference in the FFT. A single-operator sequential CORDIC converts the Cartesian estimate into a phase difference. The evolution of the carrier offset phase is reproduced by a phase accumulator with a pipelined CORDIC stage (Fig. 5). The CORDIC uses a constant input reference  $1 + j0$  to provide a Cartesian output with a conversion accuracy independent of the highly amplitude-varying receive signal.

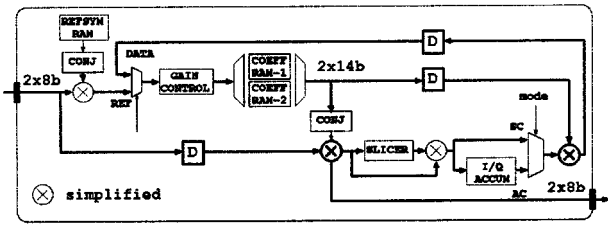


Fig. 6. The Festival equalizer reveals a low-cost equalizer with feedforward channel estimation and feedback decision-directed tracking.

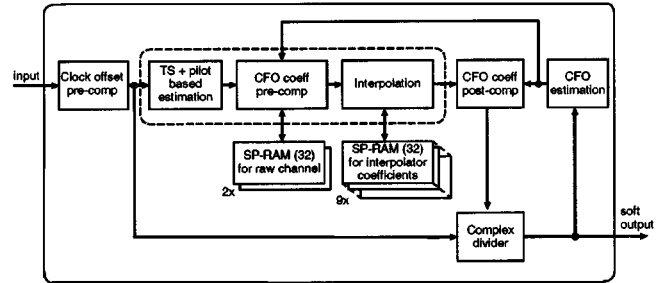


Fig. 7. The Carnival equalizer has an interpolator and divider in addition to the Festival equalizer.

#### D. Adaptive Frequency-Domain Channel Estimation and Tracking

The received signal after the FFT is still affected by multipath fading and contains a remaining low carrier frequency offset. However, by proper choice of the subcarrier spacing relative to the coherence bandwidth, the FFT produces a highly oversampled channel response. This results in a quasi-diagonal channel matrix  $H$  with insignificant contributions on the nondiagonal entries. The equalizer can exploit this in two ways. First, it requires only a single complex channel coefficient per subcarrier to compensate for the channel. Second, the rank of this matrix is reduced, since high oversampling translates into correlated channel coefficients. Thus, we can apply filtering to suppress noise and interpolate a smoothed channel vector from a smaller set of coefficients. This has been implemented in the Carnival ASIC, since the initial reference based estimate was poor for the 16-QAM and 64-QAM case.

The Festival equalizer (Fig. 6) implements the basic one-tap frequency domain equalization, consisting of a single complex multiplier with a coefficient memory to store the channel matrix diagonal [12]. The channel is estimated by multiplying received initial or periodic reference symbols with a known reference. A decision-directed loop estimates either individual subcarrier phase error or average phase error based on QPSK slicing. The channel estimate is thus updated for phase only, tracking such effects as fine carrier frequency offset or, to a limited amount, clock offset. Gain control on  $I$  and  $Q$  parts, using a greatest common divider (GCD) algorithm [12], stabilizes the loop and prevents amplitude drift.

The Carnival equalizer (Fig. 7) also uses the concept of a single complex operator with coefficient memory. 16-QAM

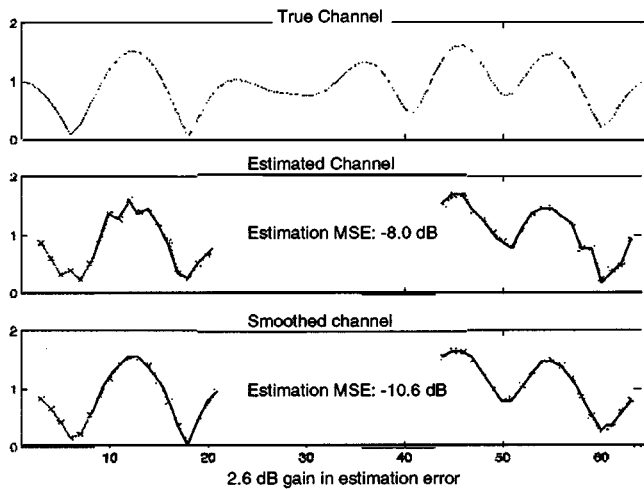


Fig. 8. Simple reference-symbol-based channel estimation reveals poor noise-influenced results. The interpolator improves the channel estimate S/N by 2.5 to 3 dB.

and 64-QAM constellation schemes, however, require accurate amplitude correction, which is performed by a complex divider. In addition to initial and periodic reference symbols, to update part of the channel, a pilot pattern is sent with every symbol. The channel estimate obtained from a single reference symbol still contains a considerable mm error (Fig. 8). A channel interpolator (Fig. 9), consisting of an initial “noisy” stage with the CFO phase error update, is followed by a cascade of four blocks implementing a matrix operation:  $\underline{H}_{\text{smooth}} = S S^H \underline{H}_{\text{noisy}}$ . Matrix  $S$  is a  $64 \times 9$  programmable complex coefficient matrix. The first two stages transform the noisy channel estimate into an impulse response vector of length 9, effectively suppressing any noise present beyond nine taps. The last two stages interpolate the full 64-tap frequency domain channel response from this truncated impulse response vector [13]. The first three stages employ full parallelism such that an interpolated channel tap is again available after one OFDM symbol latency. Coefficient sets are stored in nine RAMs next to a pre-programmed set in a LUT. The interpolator is also used during tracking, improving the channel estimate by 2.5 to 3 dB. Together with the rotating pilot scheme, it is also able to suppress spurs, e.g., from the equalizer feedback loop, reducing error propagation.

Clock offset between receiver and transmitter sampling clocks, over the burst length of 2 ms (Hiperlan 2) or  $>5$  ms (IEEE), not only has an impact on the subcarrier phase but can shift the actual OFDM symbol out of the FFT frame leading to a low signal-to-interference ratio. Typical values according to IEEE and ETSI standard are as high as  $\pm 40$  ppm of the 20-MHz system reference oscillator. The drift can be estimated by correlating the cyclic prefix with its original counterpart in the same OFDM symbol [Fig. 10(a)]. The correlation peaks are estimated and averaged over more than 32 OFDM symbols to reduce noise on the estimate. Compensation occurs by either dropping an entire sample from or adding one to the cyclic prefix [Fig. 10(b)], resembling a sigma-delta architecture. The shifting events are communicated to the equalizer to adapt the stored subcarrier phases to the instantaneous sample shift.

#### IV. SYSTEM INTEGRATION

The previous section proposed a set of signal processing algorithms and architectures to solve individual problems. When it comes to system design, ease of integration is required. This essentially translates into partitioning the system into building blocks (design units) in such a way that both data transfer and storage costs between design units are low [14], yet the system can still be designed with reasonable effort assuming limited EDA support.

##### A. Partitioning Based on Data Transfer and Storage Cost

Wireless LAN transceivers both require high throughput and low latency, leaving limited space for sequential processing. The FFT processes about 1 Gops/s while the interpolator needs 3 Gops/s. Higher clock speed could reduce parallelism, however introduces more data caches to adapt different rates, which are also induced by the flexible OFDM symbol structure that we proposed. Nevertheless, this multirate problem can be solved by either sharing a common memory or by a distributed memory approach depending on the local processing needs.

On the one side, for the FFT, a distributed memory architecture was found to be superior to a single memory running at higher clock speed with caching from a data transfer power point of view. On the other side, a number of sample-reordering tasks were efficiently implemented with a dual central memory of minimum length in the SSR. Both solutions efficiently use the memory transfer bandwidth while maintaining a regular access pattern. The final on-chip datapath does not contain any caching beyond the minimum required by the signal format defined in the IEEE or ETSI standard. This caching latency is two OFDM symbols for both receive and transmit path evenly divided on FFT processing and bit-reverse reordering.

All design units contain their own local register banks for programming parameters. This supports the IP block concept and eliminates layout dependencies on interconnects. Multiple instantiations in case of common parameters have negligible cost. A single write address for the same parameter in all units and individual read addresses guarantee correct programming and verification.

##### B. Token-Based Distributed Control

To stress the IP concept, a generic communication protocol is required between all design units. We implemented a scheme based on token semantics that follows the natural data flow through transmit and receive path (Fig. 11). A closed token-loop scheme is used between the burst controller and the datapath. Tokens contain three types of information: meta-symbol start, burst state information (BSI), and dynamic datapath information (DDI). Tokens are not sent at the sampling rate, but at the rate of meta-symbols, i.e., at OFDM symbol rate. This token part is returned to the burst controller where it is compared against the burst length. The BSI indicates reference symbols and the last symbol of a burst and is returned by the last unit in the datapath to indicate that an entire burst has been fully processed. DDI can be added to a token by any datapath block to transfer data-dependent information synchronously with the current symbol to

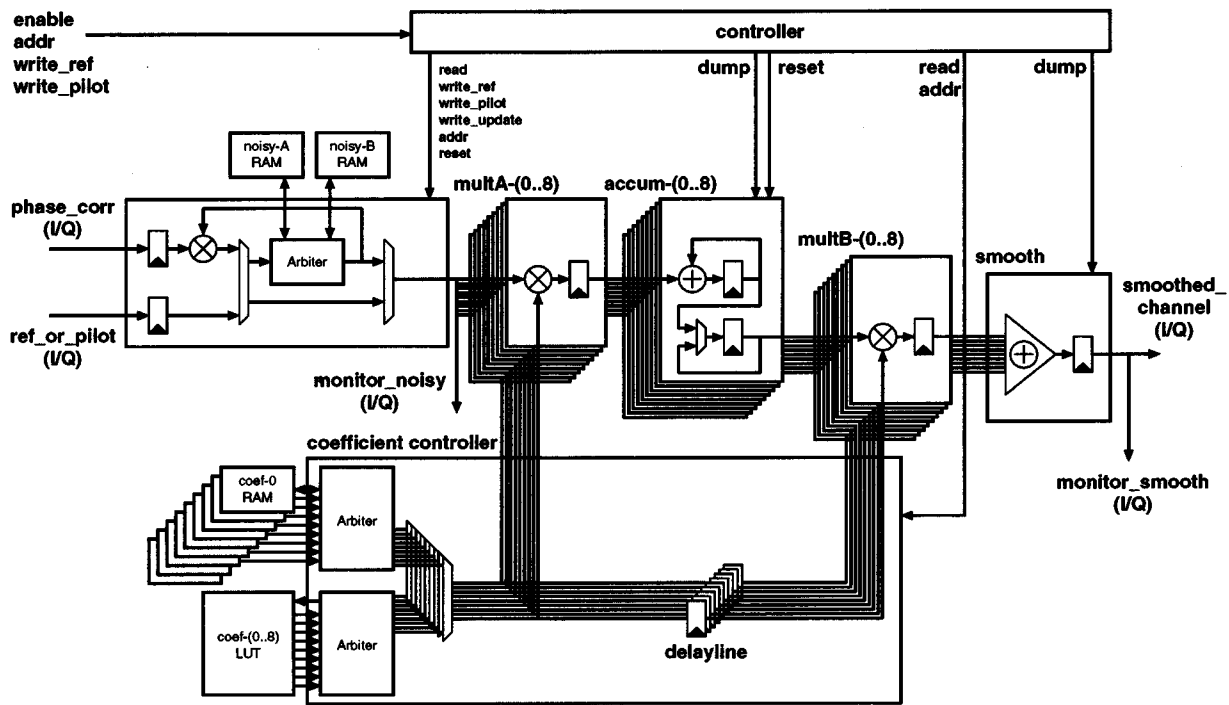


Fig. 9. The impulse response is truncated and interpolated using a fully programmable  $64 \times 9$  transformation matrix.

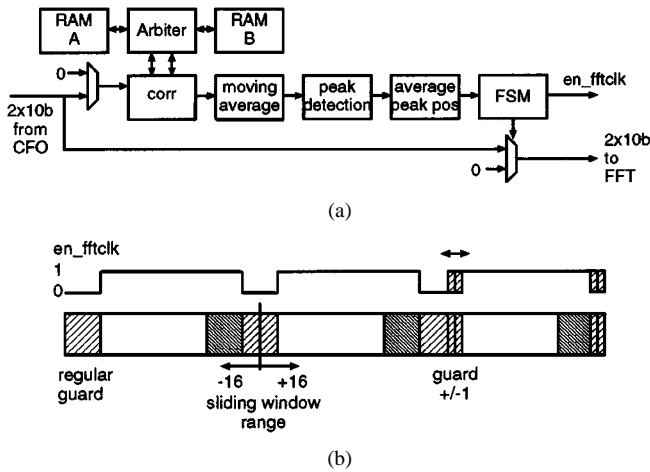


Fig. 10. Clock offset is tracked by guard interval correlation and averaging over multiple OFDM symbols. (a) Architecture. (b) Timing.

another unit down the processing chain. The clock offset estimator uses this to inform the equalizer in case of a FFT frame shift. The token scheme scales with multirate and simplifies also the design task, since a token arrival window is defined instead of a discrete point in time, keeping detailed unit latency information locally.

C. Clocking Strategy

Low-power operation is crucial for portable operation. Power consumption in synchronous systems is dominated by clocking. However, analysis of a typical receive scenario reveals that a receiver remains a considerable amount of time in listening mode searching for a receive signal. Gaining on the average compared to the peak power consumption has been achieved by matching

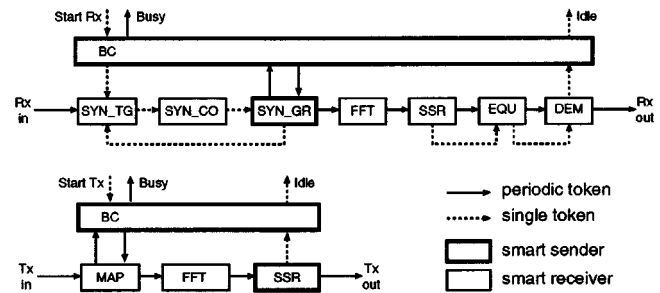


Fig. 11. Receiver and transmitter token flow exploit a closed loop token scheme.

activation of units with the time windows they are effectively required from the networking protocol and burst format point of view, implemented as clock gating with a state-based activation. The burst controller and decentralized smart senders (Fig. 11) control the clock generation. We also use clock gating to implement multirate interfaces between units. Transitions between units operating on different clocks are facilitated through retiming on a common inverted coreclk\_N\_out reducing the potential skew complexity from  $O(n^2)$  to  $O(n)$ , with  $n$  being the number of clocks.

The ASICs are master for all datapath interfaces and provide on-chip generated clock signals. These clocks are generated locally to the other interface I/O signals to allow joint skew optimization.

D. Object-Oriented Design Methodology and Tool Flow

Complex system design requires a smooth tool flow that allows joint optimization and refinement of algorithmic and architecture issues (Fig. 12). We started with a high-level datapath

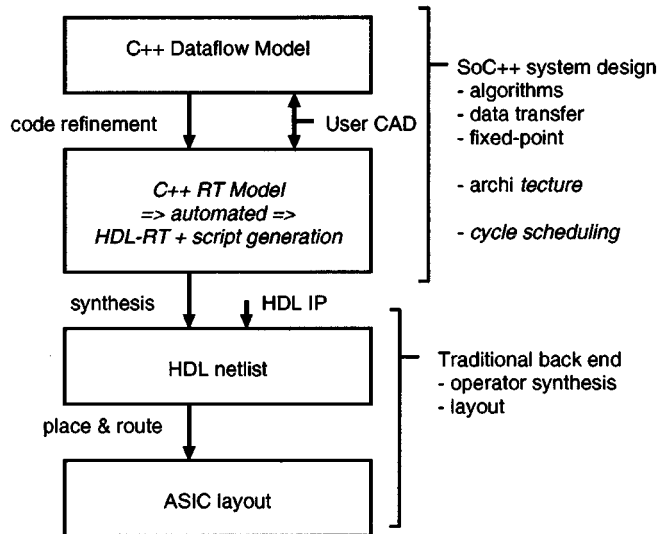


Fig. 12. The object-oriented design flow starts from C++ and results in a conventional HDL-based design flow.

model in C++ using the OCAPI [15] hardware libraries. Performance evaluation, algorithm selection, fixed-point refinement, and functional partitioning were performed on this model. Object-oriented design gives the designer freedom to design generic classes that rather construct hardware from given user constraints. Inheritance and fully parametrizable hierarchical instantiation are strong assets for a clean code database. The transceiver, for example, is instantiated twice and configured either as transmitter or receiver just at the top level. Internally, interconnection and scheduling can be optimized for simulation speed or for hardware match. Also, on-the-fly reconfiguration is possible during simulation.

The C++ dataflow model was refined toward a C++ description based on integrated finite-state machines and data path (FSMD) blocks. It is important to start exploration of data transfer and storage issues already at the dataflow level [16], since this prevents frequent and time-consuming loop back between the FSMD and the dataflow design. Refinement includes mainly operator sharing and scheduling. VHDL RTL code is generated automatically from the C++ FSMD description. Both Festival and Carnival make use of existing native VHDL code. These units were modeled as abstract dataflow blocks to obtain a complete dataflow end to end link. Carnival also used native Verilog code, showing that a C++ entry-level approach can be well integrated into a heterogeneous design flow. From RT on, a conventional standard cell design flow is followed with logic synthesis, floorplanning, and layout steps. Clock tree routing was performed at layout level and included into the back-annotation.

Generated HDL, gate-level, and back-annotated gate-level netlists were all verified against the same test vectors generated from the C++ dataflow model. Extraction of simulation results from RT and gate-level simulations only requires synchronization of control token flow and dataflow at the design top-level to match the different abstraction level. This was the only HDL code modification required to execute all testbenches.

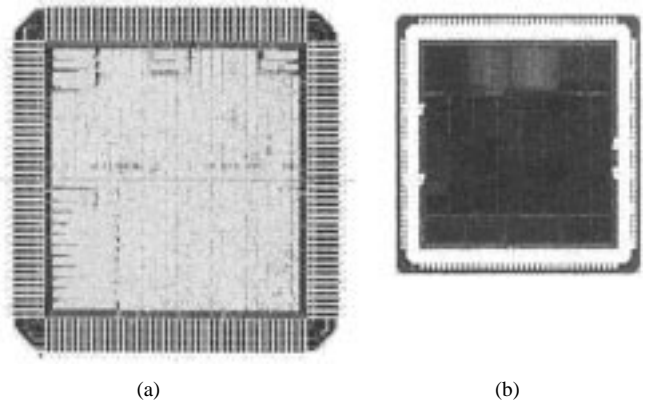


Fig. 13. (a) Carnival (0.18- $\mu\text{m}$  CMOS) outperforms (b) Festival (0.35- $\mu\text{m}$  CMOS) at the cost of a 30% area increase only.

TABLE II  
CARNIVAL OUTPERFORMS FESTIVAL CONSIDERING SPECTRAL EFFICIENCY AND ENERGY EFFICIENCY AT A MODERATE AREA INCREASE DESPITE SIGNIFICANTLY HIGHER COMPLEXITY

for the same protocol and burst overhead at an uncoded rate of 76 Mb/s	Festival ASIC	Carnival ASIC	Carnival relative to Festival
Efficiency (b/s/Hz)	1.5	3.8	2.5 (better)
Energy per Tx bit (nJ/b)	8.8	2.6	3.4 (less)
Energy per Rx bit (nJ/b)	7.5	2.8	2.7 (less)
Transmit power (mW)	670	199	3.4 (less)
Receive power (mW)	570	212	2.7 (less)
Nominal $f_{clk}$ (MHz)	50	20	2.5 (less)
Technology	CMOS	CMOS	-
Feature size ( $\mu\text{m}$ )	0.35	0.18	-
Supply (V)	3.3	1.8 / 3.3	-
Die size ( $\text{mm}^2$ )	16.4	20.8	1.3 (more)

## V. MEASUREMENTS RESULTS AND PERFORMANCE COMPARISON

Both ASICs have been implemented in digital CMOS technologies: Festival in a 0.35- $\mu\text{m}$  5LM Alcatel Microelectronics and Carnival in a 0.18- $\mu\text{m}$  6LM National Semiconductor process (Fig. 13). Both designs were pad-limited with 144 and 160 pads, respectively. The nominal clock rate is specified up to 50 MHz for Festival and up to 20 MHz for Carnival. Both ICs use embedded SRAM for datapath and parameter storage, with nine instances in Festival compared to 19 in Carnival.

A fair comparison at the same data rate and overhead between Festival and Carnival (Table II) shows the superior spectral efficiency and energy efficiency of the latter at the expense of a moderate area increase of 30%. The highly programmable equalizer occupies 63% of the area in the 64-QAM chip compared to 10% for the FFT. Fixing the coefficient set is reducing this percentage to significantly less than 50%.

Power consumption has been measured separately for 1.8-V core and 3.3-V I/O supply for the Carnival ASIC in typical transmit, receive, and programming scenarios. During transmission, 156-mW I/O and 43-mW core consumption were observed. During reception, the much higher core activity

dominates with 146 mW compared to a lower 66-mW I/O consumption due to less I/O switching. In programming mode, logic switching is zero, but all clocks are enabled, leading to 35-mW I/O and 81-mW core consumption.

Both ASICs were tested in an experimental test setup consisting of a discrete superheterodyne 5-GHz front-end with digital  $4\times$  oversampled IF, a field programmable gate array (FPGA)-based hardware MAC, and a software MAC, and application protocol interface (API) implemented on a PC. Efforts are ongoing toward a full integration with a 5-GHz front-end into a single package [17]. Application tests with web-cam image transmission, video transmission, and file transfers were successfully run between two of these platforms over the air.

## VI. CONCLUSION

The realization of two digital baseband signal-processing ASICs, achieving bit rates beyond 50 Mb/s with moderate technology constraints and area costs, show the viability of cost-efficient deployment of broadband wireless indoor systems for both the consumer market and business applications. The spectrally efficient 64-QAM constellation puts high requirements on transceiver performance. We have shown that novel digital signal-processing techniques, such as an interpolating equalizer, rotating pilots, and guard-interval-based clock offset estimation, can cope with the multipath channel and analog front-end impairments.

The choice of a scalable multiprocessor architecture with distributed control using token semantics allows to maintain a high degree of flexibility and programmability throughout the design. A high reuse percentage in the Carnival design proved the scalability. The object-oriented FSM-D-centric design approach using C++ has shown its strength at higher abstraction levels for system exploration and at FSM-D level for HDL generation even in a heterogeneous mixed-language flow.

Despite the significantly higher signal processing complexity for 16- and 64-QAM, the Carnival ASIC outperforms its predecessor for the 5-GHz band on spectral efficiency and even energy efficiency. The 64-QAM ASIC is also designed beyond the current IEEE 802.11a and ETSI Hiperlan/2 specification with performance-improving add-ons in mind such as adaptive loading [18].

## ACKNOWLEDGMENT

The authors would like to thank National Semiconductor for fabricating the Carnival 0.18- $\mu\text{m}$  CMOS prototype chip.

## REFERENCES

- [1] *WLAN MAC and PHY Specifications: High-speed Physical Layer in the 5 GHz Band*, IEEE Std 802.11a Supplement to IEEE Std Part 11, Sept. 1999.
- [2] *Broadband Radio Access Networks; HIPERLAN Type 2; Physical (PHY) Layer*, ETSI TS 101 475 Technical Specification, Apr. 2000.
- [3] G. J. M. Janssen, P. A. Stigter, and R. Prasad, "Wideband indoor channel measurements and BER analysis of frequency selective multipath channels at 2.4, 4.75, and 11.5 GHz," *IEEE Trans. Commun.*, vol. 44, pp. 1272–1281, Oct. 1996.

- [4] L. J. Cimini, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Trans. Commun.*, vol. 33, pp. 665–675, July 1985.
- [5] W. Eberle, M. Badaroglu, V. Derudder, S. Thoen, P. Vandenameele, L. Van der Perre, M. Vergara, B. Gyselinckx, M. Engels, and I. Bolsens, "A digital 80 Mb/s OFDM transceiver IC for wireless LAN in the 5 GHz band," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, Feb. 2000, pp. 74–75.
- [6] W. Eberle, V. Derudder, L. Van der Perre, G. Vanwijnsberghe, M. Vergara, L. Deneire, B. Gyselinckx, M. Engels, I. Bolsens, and H. D. Man, "A digital 72 Mb/s 64-QAM OFDM transceiver for 5 GHz wireless LAN 0.18- $\mu\text{m}$  CMOS," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, Feb. 2001, pp. 336–337.
- [7] D. V. Veithen *et al.*, "A 70 Mb/s variable-rate DMT-based modem for VDSL," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, Feb. 1999, pp. 248–249.
- [8] C. Mandl, M. Bacher, G. Krampl, and F. Kuttner, "0.35- $\mu\text{m}$  COFDM receiver chip for DVB-T," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, Feb. 2000, pp. 76–77.
- [9] J. A. H. Huisken *et al.*, "A power-efficient single-chip OFDM demodulator and channel decoder for multimedia broadcasting," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1793–8, Nov. 1998.
- [10] A. M. Despain, "Very fast Fourier transform algorithms for hardware implementation," *IEEE Trans. Comput.*, vol. C-28, pp. 333–341, 5 1979.
- [11] M. Vergara, M. Strum, W. Eberle, and B. Gyselinckx, "A 195 kFFT/s 256-points high performance FFT/IFFT processor for OFDM applications," in *Proc. SBT/IEEE Int. Telecommunications Symp.*, vol. 1, 1998, pp. 273–278.
- [12] W. Eberle, M. Badaroglu, V. Derudder, S. Thoen, P. Vandenameele, L. V. d. Perre, M. Vergara, B. Gyselinckx, M. Engels, and I. Bolsens, "Flexible OFDM Transceiver for high-speed WLAN," in *Proc. IEEE VTC*, vol. 5, Oct. 1999, pp. 2677–2681.
- [13] L. Deneire, P. Vandenameele, L. V. d. Perre, M. Engels, and B. Gyselinckx, "A low complexity ML channel estimator for OFDM," in *Proc. IEEE Int. Conf. Communications (ICC)*, June 2001, pp. 1461–1465.
- [14] F. Cathoor, S. Wuytack, E. De Greef, F. Franssen, L. Nachtergaele, and H. De Man, "System-level transformations for low data transfer and storage," in *Low Power CMOS Design*, A. Chakasan and R. Brodersen, Eds. New York: IEEE Press, 1998, pp. 609–618.
- [15] P. Schaumont, S. Vernalde, L. Rijnders, M. Engels, and I. Bolsens, "A design environment for the design of complex high-speed ASICs," in *Proc. 35th Design Automation Conf.*, June 1998, pp. 315–320.
- [16] D. Verkest, W. Eberle, and P. Schaumont, "C++ based system design of a 72 Mb/s OFDM transceiver for wireless LAN," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, May 2001, pp. 433–439.
- [17] P. Wambacq, S. Donnay, P. Pieters, W. Diels, K. Vaesen, W. De Raedt, E. Beyne, M. Engels, and I. Bolsens, "Chip-package co-design of a 5 GHz RF front-end for WLAN," in *Proc. IEEE Int. Solid State Circuits Conf. (ISSCC)*, Feb. 2000, pp. 318–319.
- [18] R. F. H. Fischer and J. B. Huber, "A new loading algorithm for discrete multitone transmission," in *Proc. Globecom*, 1996, pp. 724–728.



**Wolfgang Eberle** (M'00) received the M.S. degree in electrical engineering from the Saarland University, Saarbrücken, Germany, in 1996, with specialization in microwave engineering and telecommunication networks. He is currently working toward the Ph.D. degree in electrical engineering at the Katholieke Universiteit Leuven, Belgium.

He joined the Wireless Systems Group of IMEC, Leuven, Belgium, in 1997, where he has been working on system design, algorithm development, digital signal processing, and VLSI implementation of digital OFDM-based wireless LAN modems. In late 2000, he joined the Mixed-Signal and RF Applications Group of IMEC where he is currently focusing on mixed-signal system design tradeoffs, transmitter linearization, and CAD for system-level behavioral and architectural simulation applied to wireless LANs.



**Veerle Derudder** received the B.S. degree in electrical engineering from KHBO, Oostende, Belgium, in 1990.

She joined IMEC, Leuven, Belgium, in 1990, working on the design of parameterized module generators for DSP applications. In 1995, she became responsible for the ASIC test strategy. She has also been involved in the design of ASICs for spectrum spectrum satellite modems, satellite navigation receivers, and OFDM transceivers. She is currently a Senior Design Architect working on

Turbo decoding ASICs.



**Geert Vanwijnsberghe** received the M.S. degree in electrical engineering from the Katholieke Universiteit Leuven, Belgium, in 1984.

He joined the Inter-University Microelectronics Center (IMEC), Leuven, Belgium, in 1986 as a CAD Support Engineer. Since 1990, he has been a Project Engineer for ASIC design in the Invomec division, being involved in mostly digital ASIC design in close cooperation with industry and the European Space Agency (ESA). Since 1999, he has worked on a team on designing OFDM ASICs for

wireless LAN applications. He has also been responsible for several CAD training courses for major international companies.



**Mario Vergara** received the B.S. degree in electronics engineering from the University of Cauca, Colombia, in 1993, and the M.S. degree in electrical engineering from Sao Paulo University, Brazil, in 1998.

From 1998 to 2000, he was with WISE group at IMEC, Leuven, Belgium. His work focused on the ASIC design of a high-speed FFT/IFFT core suited for OFDM applications. In 2000, he joined the WCDMA design group at Ericsson Mobile Communications AB, Lund, Sweden, where he is

involved in the ASIC design of WCDMA platforms for UMTS applications.



**Luc Deneire** (M'99) received the Eng. degree in electronics from the University of Liège, Belgium, in 1988, the Eng. degree in telecommunications from University of Louvain-La-Neuve in 1994, and the Ph.D. degree in signal processing from Eurecom, Sophia Antipolis, France, in 1998.

In 1999, he was a Consultant for Texas Instruments, Villeneuve-Loubet, France, and since late 1999, he has been a Senior Researcher at IMEC, Leuven, Belgium. He is working on signal processing algorithms involved in wireless communications, specifically for wireless LANs and wireless personal area networks.

His main interests are equalization and channel estimation, modulation theory, smart antennas, and link adaptation.



**Liesbet Van der Perre** received the M.S. degree in electrical engineering from the Katholieke Universiteit Leuven, Belgium, in 1992, the M.S. degree from the ENST, Paris, France, and the Ph.D. degree in Electrical Engineering from the Katholieke Universiteit Leuven in 1997.

She is currently a Senior Researcher in the wireless systems group of IMEC, Leuven, Belgium. Her work focuses on system design and digital modems for high-speed wireless communications. She is also a part-time Professor at the University of Antwerp,

Antwerp, Belgium.



**Marc G. E. Engels** (M'96) received the engineering degree in 1988 and the Ph.D. degree in 1993, both from the Katholieke Universiteit Leuven, Leuven, Belgium.

He is currently the Director of the telecom department (DISTA) at IMEC, Leuven, Belgium. His main research activity is in the implementation of telecommunication systems on a chip. His current work is focused on broadband wireless systems, such as wireless local area networks (WLANs) and wireless personal area networks (WPANs). For these systems, the department investigates the DSP processing, the mixed-signal RF front-end and run-time configurable functionality. A major emphasis of the department is also on a C++-based design methodology to realize these applications onto VLSI in an efficient way. Previously, he performed research at the Katholieke Universiteit Leuven, Belgium, Stanford University, Stanford, CA, and the Royal Military School, Brussels, Belgium.

Dr. Engels is an active member of the SITEL, the Royal Flemish Engineering Society (KVIV) Telecommunications Society, and the IEEE Benelux chapter on vehicular technology and telecommunications. He is also an Associate Editor of IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION.



**Ivo Bolsens** (M'97) was born in Wilrijk, Belgium, in 1958. He received the electrical engineering and Ph.D. degrees from the Katholieke Universiteit Leuven, Belgium, in 1981.

He joined the CAD group, ESAT Laboratory, Katholieke Universiteit Leuven, in 1981, where he worked on the development of an electrical verification program for VLSI circuits and on mixed-mode simulation. In 1984, he joined the Interuniversity Microelectronics Center (IMEC), Leuven, where he started doing research on the development of knowledge-based verification for VLSI circuits, exploiting methods in the domain of artificial intelligence. In this context he introduced functional programming, using Lisp, and object-oriented programming, using Smalltalk. In 1989, he became responsible for the application and development of the Cathedral-2, and later the Cathedral-3, architectural synthesis environment. He was also heading the application projects that produced the first silicon, generated by these software environments. In 1993, he became head of the Applications and Design Technology Group, focusing on the development and application of new design technology for mobile communication terminals. In this context, he was responsible for the implementation of a programmable spread-spectrum transceiver for satellite communications.

Dr. Bolsens was the recipient in 1986 of the Darlington Award of the IEEE Circuits and Systems Society for best paper published by the IEEE CAS Society that bridges the gap between theory and practice. He received a distinguished paper citation at the 1991 International Conference on CAD. In 1993, he received a Best Circuit Award from the EUROASIC-EDAC conference.



**Hugo De Man** (M'81-SM'81-F'86) was born in Boom, Belgium, on September 19, 1940. He received the electrical engineering degree and the Ph.D. degree in applied sciences from the Katholieke Universiteit Leuven, Leuven, Belgium, in 1964 and 1968, respectively.

From 1969 to 1971, he was with the Electronic Research Laboratory, University of California, Berkeley, as an ESRO-NASA Postdoctoral Research Fellow, working on computer-aided device and circuit design. In 1971, he returned to the University of Leuven as a Research Associate of the NFWO (Belgian National Science Foundation). In 1974, he became a Professor at the University of Leuven. During the winter quarter of 1974-1975 he was a Visiting Associate Professor at the University of California, Berkeley. From 1984 to 1995, he was Vice-President of the VLSI systems design group of IMEC, Leuven. Since 1995, he has been a Senior Research Fellow of IMEC responsible for research in system design technologies.

Dr. De Man is a corresponding member of the Royal Academy of Sciences, Belgium, and a member of the Royal Flemish Engineering Society (KVIV).