Efficient Inner Receiver Design for OFDM-Based WLAN Systems: Algorithm and Architecture

Alfonso Troya, *Member, IEEE,* Koushik Maharatna, *Member, IEEE,* Miloš Krstić, Eckhard Grass, Ulrich Jagdhold, and Rolf Kraemer, *Member, IEEE*

***Abstract*— In this article we propose a complete solution for the so-called Inner Receiver of an OFDM-WLAN system based on the IEEE 802.11a standard. We concentrate our investigations on three key components forming the Inner Receiver namely, the Synchronizer, the Channel Estimator and the Digital Timing Loop. The main goal is the joint optimization of the signal processing algorithms along with the implementation friendly VLSI architecture required for these three key components in order to reduce power, area and latency, without compromising the performance excessively. We provide both the mathematical details and extensive computer simulations to validate our design.**

***Index Terms*— Channel estimation, OFDM, synchronization, wireless LAN.**

I. Introduction

THE use of the OFDM (Orthogonal Frequency Division Multiplex) transmission technique has gained a lot of interest in the recent years due to its spectral efficiency and capability to overcome multi-path fading. In this paper we concentrate on the OFDM-WLAN (Wireless Local Area Network) systems, which are already a reality thanks to the IEEE 802.11a/g standards [1], [2]. The application of OFDM is not restricted to these two standards, but new standardization processes already foresee the application of OFDM in future WLAN [3] and UWB (Ultra Wideband) systems [4].

The key property of OFDM is orthogonality. By this property the system uses the input data to modulate a number of mutually orthogonal sub-carriers. This technique facilitates a high data rate transmission system. However, the whole system performance depends on maintaining the orthogonality of the sub-carriers. If the orthogonality property gets disturbed, unwanted effects such as Inter-Carrier Interference (ICI) and Inter-Symbol Interference (ISI) will occur during signal reception. In general, the orthogonality property of the sub-carriers can be disturbed during the RF Up- and Down-conversion. On top of that the characteristic of the transmission channel may also affect the orthogonality condition. A number of authors have addressed the impact of this type of impairments on OFDM signals in the past years [5], [6]. Thus, in order to make the system work efficiently, we need to re-establish the orthogonality condition at the receiver. The so-called *Inner Receiver* (this term was firstly coined by Heinrich Meyr [7]) is used for this purpose. In essence, there are two main operations carried out inside the Inner Receiver (IRx) namely *Signal Acquisition* and *Channel Correction* as shown in Fig. 1. The acquisition operation is performed by means of a synchronization block, which should be able to perform reliable Frame Detection (FD), and to provide estimations for the Carrier Frequency Offset (CFO) and Symbol Timing Offset (STO). The channel correction operation is needed to estimate and compensate the Channel Transfer Function (CTF), provided that orthogonality has been restored to a great extent by the synchronizer. The final goal is to supply the decoding and demodulator block with In-phase and Quadrature components that are as similar as possible to the original ones.

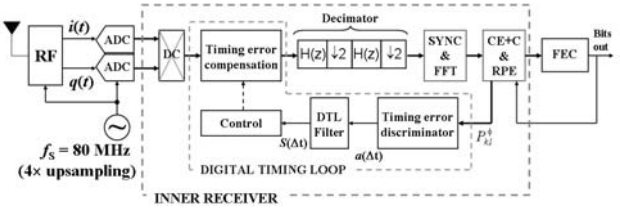


Fig. 1. General block diagram of the proposed Inner Receiver.

Though the IRx is an integrated part of the OFDM-based WLAN system, its design complexity is frequently underestimated. Unfortunately the standards do not provide in general any hints on how to implement the IRx, but it is left as a developer's task. In this article we investigate an efficient realization of the IRx for IEEE 802.11a systems both from the algorithm and VLSI (Very Large Scale Integration) implementation point of view, and provide a complete and practical solution for it. The results developed in this work are applicable to the future standards [3]. In order to develop our solution we start with the algorithm level formulation of the desired functionality of the IRx. The algorithmic development has been considered strictly in conjunction with the possible architectural feasibility of an ASIC (Application Specific Integrated Circuit) implementation. Thus a *joint algorithm and architecture optimization* has been undertaken using power consumption, silicon area, system latency and overall noise performance as the "quality/efficiency" parameters for the system. The power consumption and silicon area have been

Manuscript received July 5, 2005; revised January 11, 2006 and May 29, 2006; accepted July 17, 2006. The associate editor coordinating the review of this paper and approving it for publication was C. Xiao.

A. Troya was with IHP, Frankfurt (Oder), Germany. He is now with Infineon Technologies AG, COM PS CE ALG, 81726 Munich, Germany (e-mail: alfonso.troya@infineon.com).

K. Maharatna is with the University of Southampton, University Road, Southampton, SO17 1BJ, UK (e-mail: km3@ecs.soton.ac.uk).

M. Krstić, E. Grass, U. Jagdhold, and R. Kraemer are with IHP, Frankfurt (Oder), Germany (e-mail: {krstic, grass, jagdhold, kraemer}@ihp-microelectronics.com).

Digital Object Identifier 10.1109/TWC.2007.05481.

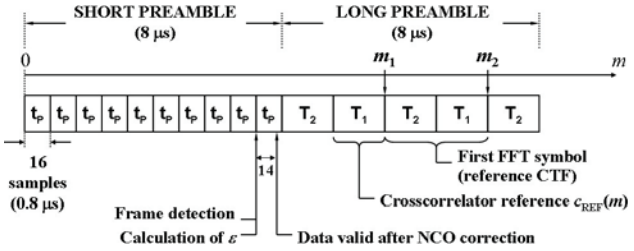


Fig. 2. Preamble symbols as defined by the 802.11a standard together with the timing schedule followed inside the Synchronizer.

considered as two of the main parameters since the system is targeted for mobile and portable applications where saving of battery life as well as the total size of the system are crucial. Latency has been considered from the operation principle of the IEEE 802.11a MAC (Medium Access Control) protocol [1].

Different parts of the present work have been published in different renowned conferences in short form [8], [10], [11], [19]. In this paper we provide a much more detailed and integrated view of the complete IRx solution. The rest of the present paper is organized as follows: after introduction, the main components of the IRx are investigated. Subsequently, an efficient synchronizer architecture is examined in Section II, whose main architecture was foreseen by the authors in [8], [9]. Section III is devoted to the analysis of a decision-directed Channel Estimator (CE). Two blocks are the main focus of our investigations, namely the Noise Reduction Filter (NRF) and the Residual Phase Error (RPE) correction block. The proposed timing loop is analyzed in detail in Section IV and provides a simple method to compensate for the Sampling Clock Frequency Offset (SCFO) based on the RPE estimation supplied by the CE. Section V presents simulation results which show the performance features of the proposed solutions. Finally, in Section VI, some important conclusions are derived.

II. The Synchronizer

The synchronizer is the block responsible for signal acquisition. This term encompasses a number of operations that need to be performed in a very limited period of time in order to minimize latency. For our purpose, synchronization must be finished within the preamble time, i.e. 16 μs, and the following operations must be performed based on the preamble symbols:

1) Frame detection.
2) Determination of the symbol timing.
3) Carrier frequency offset estimation and correction.
4) Extraction of the reference channel estimation.

The order in which these operations are carried out strongly determines the architecture of the synchronizer. The preamble symbols in the 802.11a standard comprise a number of periodic sequences as shown in Fig. 2. This periodic structure suggests a solution based on **autocorrelators** [13], [14]. The proposed implementation shown in Fig. 3 contains two autocorrelators. Each one encompasses a delay line (FIFO-type buffer) of length N_d, a complex conjugate operation, a complex multiplier, and a moving average of length N_{avg}. The moving average is an FIR filter of length N_{avg} with all its coefficients being 1. Let's consider the input signal $r(m)$ to be sampled at frequency f_s and affected by a CFO $f_\epsilon = \epsilon\Delta f$, where ϵ stands for the normalized CFO, and Δf is the subcarrier spacing in the OFDM signal (Δf = 312.5 KHz in the 802.11a). Hence, the input signal $r(m)$ can be expressed as

$$r(m) = s(m) \cdot e^{j2\pi\epsilon\frac{\Delta f}{f_s}m} + v(m), \tag{1}$$

where $j = \sqrt{-1}$, $s(m)$ is the original time sequence and $v(m)$ represents a zero-mean white Gaussian noise process. According to (1), the autocorrelator's output signal $J_x(k)$ is given by

$$\begin{aligned} J_x(k) &= \sum_{l=0}^{N_{avg}-1} r^*(l-k)\cdot r(l-k-N_d) \\ &= e^{-j2\pi\epsilon\frac{\Delta f}{f_s}N_d}\cdot\sum_{l=0}^{N_{avg}-1} s^*(l-k)\cdot s(l-k-N_d) \\ &+ \sum_{l=0}^{N_{avg}-1} s^*(l-k)\cdot v(l-k-N_d)\cdot e^{-j2\pi\epsilon\frac{\Delta f}{f_s}\cdot(l-k)} \\ &+ \sum_{l=0}^{N_{avg}-1} v^*(l-k)\cdot s(l-k-N_d)\cdot e^{j2\pi\epsilon\frac{\Delta f}{f_s}\cdot(l-k-N_d)} \\ &+ \sum_{l=0}^{N_{avg}-1} v^*(l-k)\cdot v(l-k-N_d), \end{aligned} \tag{2}$$

where the suffic x represents either F or C in Fig. 3.

By considering the sequence $s(m)$ to be uncorrelated with the noise sequence $v(m)$, the last three summands in (2) can be neglected for sufficiently large values of N_{avg}, yielding

$$\begin{aligned} J_x(k) &\approx e^{-j2\pi\epsilon\frac{\Delta f}{f_s}N_d}\cdot\sum_{l=0}^{N_{avg}-1} s^*(l-k)\cdot s(l-k-N_d) \\ &= e^{-j2\pi\epsilon\frac{\Delta f}{f_s}N_d}\cdot\sum_{l=0}^{N_{avg}-1} |s(l-k)|^2, \end{aligned} \tag{3}$$

where it has been considered that the signal $s(m)$ is periodic with a period of N_d samples, i.e. $s(m) = s(m\text{-}N_d)$. From (3) it is straightforward to see that the phase of $J_x(k)$ is only due to ϵ, and hence ϵ could be estimated as follows

$$\hat{\epsilon} = \frac{f_s}{2\pi\cdot N_d\cdot\Delta f}\cdot tan^{-1}(J_x^*(k)). \tag{4}$$

However, there is an important factor that destroys the periodicity, making $s(m) \neq s(m-N_d)$, i.e. the Automatic Gain Control (AGC) settling time, whose influence is analyzed through simulation in Section V. If N_{avg} is a multiple of the minimum periodicity in the preambles (16 samples in case of the 802.11a, Fig. 2) the signal $|J_x(k)|^2$ shows a plateau in the region where the phase of $J_x(k)$ only depends on the CFO, as shown in Fig. 4 for $|J_F(k)|^2$. The arctangent operation in (4) is bounded in the range $[-\pi, +\pi)$. This means that (4) is also bounded as follows:

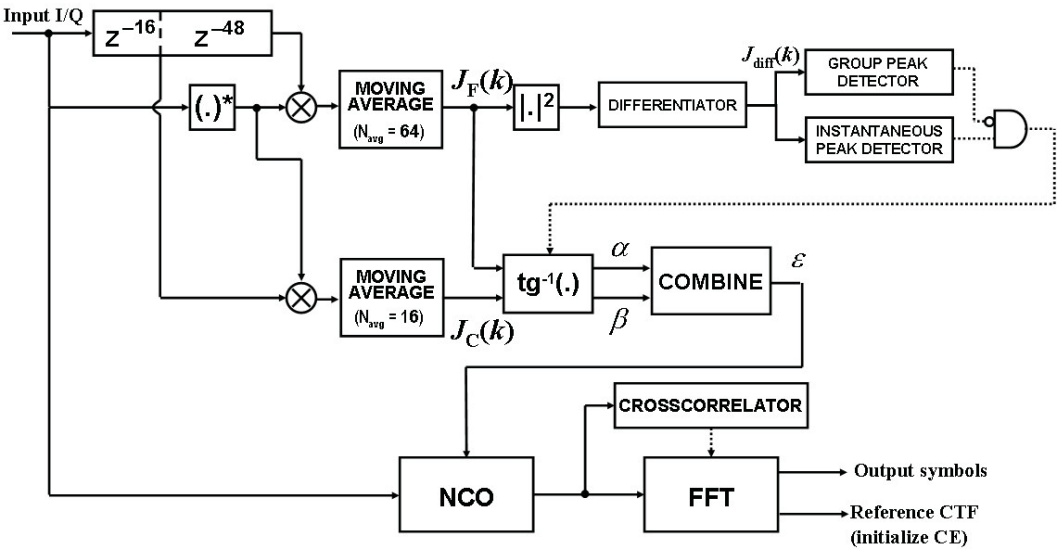


Fig. 3. General scheme of the proposed synchronizer for the IEEE 802.11a standard. The operation $\mathrm{tg}^{-1}(x)$ represents the arctangent of x.

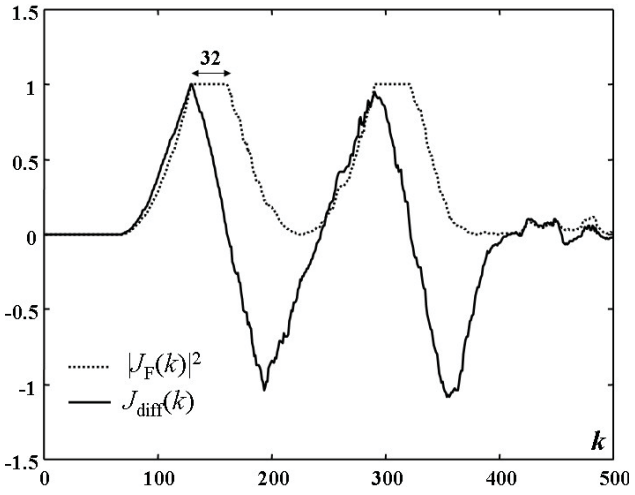


Fig. 4. Signals involved in the Frame Detection algorithm.

$$|\hat{\epsilon}| < \frac{f_s}{2 \cdot N_d \cdot \Delta f}. \tag{5}$$

Since the ratio (f_s/f) is a fixed parameter, the range of estimation of ϵ will only depend on the selected delay N_d in the autocorrelator. In the 802.11a we find that $(f_s/f) = 64$, resulting in $|\hat{\epsilon}| < 0.5$ for $N_d = 64$ and $|\hat{\epsilon}| < 2.0$ for $N_d = 16$.

A. Frame Detection Mechanism

The first operation to be carried out by the synchronizer is FD. We decided to make use of the particular shape of the signal $|J_F(k)|^2$ in order to derive a simple frame detector. Consequently, if we are able to detect the plateau in $|J_F(k)|^2$ (see Fig. 4), this will be the indication that a frame is being received. The proposed plateau detector contains two blocks namely, a *differentiator* and a *peak detector*, as depicted in Fig. 3.

The *differentiator* should indicate the point where the plateau starts. The differentiated signal $J_{diff}(k)$ is obtained as follows:

$$J_{diff}(k) = |J_F(k)|^2 - |J_F(k - N_{diff})|^2, \tag{6}$$

where N_{diff} simply defines the delay applied by the differentiator. The signal $J_{diff}(k)$ is also shown in Fig. 4 with $N_{diff} = 32$.

The autocorrelation block together with the differentiator and the peak detector constantly "peer" the channel. When the peak detector identifies an absolute maximum at the output of the differentiator, the synchronizer will consider that a new frame has arrived and the CFO estimator will be activated. However, due to the noise and more importantly, due to the Automatic Gain Control, the peak detection will not be a trivial task and a smart algorithm will be necessary in order to distinguish absolute from relative maxima [8], [9]. For this purpose the peak detector is also divided into two blocks, namely *group peak detector* and *instantaneous peak detector*, as shown in Fig. 3. The instantaneous peak detector is basically a combination of a comparator and a counter. The present sample $J_{diff}(k)$ coming out from the differentiator is compared with the last recorded maximum J_{max} ($J_{max} = 0$ at $k = 0$). As long as the sample $J_{diff}(k)$ is bigger than J_{max}, the register storing J_{max} will be updated with the new sample $J_{diff}(k)$ as the new encountered maximum and the counter will be reset. If $J_{diff}(k)$ is smaller or equal than J_{max}, the counter will be triggered and it will increase its count by one. If this situation persists until the counter overflows, the instantaneous peak detector will activate a signal stating that a relative peak has been found inside the counting scope of the counter. The group peak detector is used to detect the falling edges in $J_{diff}(k)$, and its main component is also a comparison block. There, the input signal is accumulated in groups of six samples (6-tuples) and the present group is compared with the previous one. If it is smaller, it means that the falling slope has started. If the group peak detector finds a falling edge at the same time as the instantaneous peak detector finds a relative peak, then the detected peak is actually an absolute peak. In the situation where no AGC is present, the signal $|J_F(k)|^2$ shows a plateau of 32 samples. Consequently,

the parameter N_{diff} in (6) was selected to be 16 samples, thus making the FD algorithm to detect the plateau in $|J_F(k)|^2$ at its middle point [8]. This fact justifies the definition of False Alarm Probability done later in Section V.

B. Carrier Frequency Offset Estimation and Correction

According to the specifications in the 802.11a standard [1], all the clocks and carrier signals for the transceiver should be generated from the same crystal oscillator, which should have a maximum relative frequency error of ± 20 ppm. Let's consider an example in which a signal is received at the highest possible carrier frequency of 5,805 MHz (operating channel 161 in the U-NII upper band). The total frequency deviation during down-conversion is then given by $5,805 \cdot \pm 20 = \pm 116.1$ KHz. The whole transmit-receive process introduces an overall carrier frequency error of $|f_\epsilon| = 232.2$ KHz. Normalizing this value with respect to the sub-carrier spacing, Δf = 312.5 KHz, we find the maximum normalized CFO to be $|f_\epsilon/\Delta f|$ = 0.75. The present implementation in Fig. 3 considers the frequency offsets to be in the range ± 1.5, i.e. twice the maximum value required by the standard. This decision is based on a pessimistic approach and was justified by the fact that functional tests had to be carried out using experimental Analog Front-Ends (AFE), which were not entirely fulfilling the specifications.

Two autocorrelators with N_d = 64, N_{avg} = 64, and N_d = 16, N_{avg} = 16, respectively, are used. The autocorrelator with N_d = 64 is used to get a *fine* estimation of the CFO ($|\alpha| < 0.5$), whereas the latter is used to obtain a *coarse* estimation of the CFO ($|\beta| < 2.0$). Note that the definition of fine and coarse is not based on the range, but on the accuracy of the estimation, i.e. the length of the moving average. Hence, although α is bounded more restrictively compared to β, it will be less noisy since its moving average is much larger. The final normalized CFO estimation ϵ will be a combination of the values obtained for α and β. Although β has a linear dependency throughout the entire range of possible values of the CFO, i.e. ± 1.5, this is not the case for α. Hence, the final CFO estimation cannot be directly a linear combination of the two estimations α and β. Instead, β will only serve as a range pointer and will provide the integer value of the frequency offset (either -1, $+1$ or 0), whereas α will provide the fractional part of the estimation. The final value of ϵ results from the following function,

$$\begin{aligned}
\epsilon = \alpha;\ & \text{if } -0.25 \le \beta \le 0.25, \\
& \text{or } (\alpha \ge 0 \text{ and } 0.25 < \beta < 0.75), \\
& \text{or } (\alpha < 0 \text{ and } -0.75 < \beta < -0.25), \\
\epsilon = 1 + \alpha;\ & \text{if } \beta \ge 0.75, \\
& \text{or } (\alpha < 0 \text{ and } 0.25 < \beta < 0.75), \\
\epsilon = -1 + \alpha;\ & \text{if } \beta \le -0.75, \\
& \text{or } (\alpha \ge 0 \text{ and } -0.75 < \beta < -0.25). \quad (7)
\end{aligned}$$

The estimation of the CFO will take place in one shot exactly at the time instant when the FD detects the incoming frame, since both autocorrelators exhibit a plateau at that particular point of time. An arctangent calculator is necessary to obtain α and β from $J_F(k)$ and $J_C(k)$, respectively. The correction of the CFO will follow naturally by using a Numerically Controlled Oscillator (NCO) once ϵ has been estimated. In our implementation a novel CORDIC rotator is used in its accumulation mode of operation to compute the arctangent and its rotation mode is used to realize the NCO operation [10], [11], [12].

C. Symbol Timing Estimation

Unlike to what was done during CFO estimation, where the periodicity of the short preamble symbols was the main feature exploited by the estimator, the symbol timing estimation will be obtained by exploiting the direct knowledge of the long preamble symbols.

The main block in the symbol timing estimator is a *cross-correlator*. Its purpose is to compare the input frame with a reference signal, which is directly obtained from the long preamble symbol. The proposed crosscorrelator can only be applied once the samples of the incoming frame have been fully corrected by the NCO and contain no frequency offset.

The fraction of the long preamble symbol selected as the crosscorrelator reference $c_{REF}(m)$ is shown in Fig. 2 and corresponds to the sequence defined as T_1. The reference has a length of 32 complex samples, which is the shortest possible length for this reference in order to obtain appropriate results after correlation. Under an implementation point of view, the complex crosscorrelator is usually a "weak" point in modern communication circuit designs because of its computation complexity, i.e. it requires a large number of complex multipliers and needs large silicon area. Having this in mind, in this implementation we applied a simplified scheme for the crosscorrelator, with simple XNOR 1-bit multipliers that substitute the commonly used complex multipliers. Instead of multiplying *b*-bit complex numbers, the XNOR multiplier performs only the multiplication of the sign bits of the complex input values, considering the Most Significant Bit (MSB) to be '1' when the sample is positive or zero and '0' when it is negative. Based on this, the reference sequence being used in the crosscorrelator is as follows:

$$\begin{aligned}
c_{REF}(31:0)^* = \{&1, 1, 0, 0, 1+j, j, j, j, \\
&j, 0, 1, 1, 0, 0, 1, 1+j, \\
&1+j, 0, 1+j, 1+j, j, 1+j, 1, 1+j, \\
&j, j, 1+j, 1, 1, 1+j, j, 1\}, \quad (8)
\end{aligned}$$

according to the preamble defined in [1], where $(^*)$ stands for complex conjugate.

When the preamble symbols go through the crosscorrelator, the output shows two major peaks at instants m_1 and m_2, Fig. 2. Both peaks will occur when the portions T_1 of the long preamble symbols are inside the crosscorrelator. For our purpose it is enough to detect the first peak by setting a certain threshold at the output of the crosscorrelator. More sophisticated methods based on an active peak search may be used at the expense of increased latency. The 64 samples coming immediately after the first peak, i.e. the sequence $\{T_2, T_1\}$ will be fed into the FFT in order to extract the *reference CTF*. In the 802.11a standard the long preamble symbol is defined as the sequence $\{T_1, T_2\}$, i.e. in our case a cyclic delay of 32 samples is introduced into a sequence of 64 samples.

Therefore, the resulting sequence after FFT calculation has to be multiplied by $(-1)^k$, k = 0, 1, 2, ... 63, in order to eliminate the remaining linear phase.

By observing Fig. 2 we see that the preamble contains the sequence $\{T_2, T_1\}$ twice, i.e. by averaging these two sequences one may reduce the noise power by 3 dB in the reference CTF. Note that in our case, as a measure to reduce the signal processing latency, only one preamble symbol is used to initialize the CE, which implies a penalty of 3 dB in the SNR. This problem will be treated in the next section, when discussing the CE itself.

III. The Channel Estimator

The CE deals with the estimation and correction of the filtering affecting the OFDM signal. This filtering is mainly due to the multipath transmission channel found in wireless communications, but several filters located in the transceiver hardware play an important role as well. As a result, the OFDM symbols are extended in time by an amount equal to the summation of the impulse response lengths of all the filters involved in the transmission and reception chain. Such an extension provokes the leakage of a symbol into the successive one, resulting in ISI. One interesting feature of OFDM signals is their capability to overcome the ISI when appending a Cyclic Prefix (CP) of length N_G to each transmitted OFDM symbol. This has two main advantages: on one hand, the possible leakage from the previous symbol is fully absorbed as long as it is shorter than the cyclic extension. On the other hand, the examination of the OFDM symbols in the frequency domain (after DFT) arises to be much more convenient since now the overall filtering appears inside the OFDM symbols as complex multiplicative factors affecting each of the sub-carriers. In view of this fact, channel correction becomes much easier since it can be realized by means of a complex division in the frequency domain.

The proposed CE algorithm is based on the CD3 (Coded Decision-Directed Demodulation) solution given by Mignone and Morello in [15]. The CD3 is a decision-directed method, whose main advantage is based on the fact that pilot sub-carriers are not necessary for channel estimation, thus increasing the amount of information transmitted on each OFDM symbol. However, there are a number of issues not considered in [15] that make pilot sub-carriers truly necessary, as it will be seen later. In this section we propose the modification of the CD3 channel estimator in order to accommodate two key blocks that will significantly simplify the signal processing required for reliable channel estimation. These two blocks are the Noise Reduction Filter and the Residual Phase Error estimator.

A. Noise Reduction Filter

As it was shown in Section II, the synchronizer provides a reference channel estimation that is used to initialize the CE. Due to the selected architecture, the reference is obtained from a single preamble symbol and hence, a 3 dB penalty in the initial channel estimation occurs. The NRF should help in compensating this penalty by means of the so-called *Low-Rank* approximation. This approach was firstly proposed in [16], [17] for the case in which pilot tones can be used for channel estimation. In our situation the concept is extended to the case where pseudo-pilots are available, i.e. when the CTF (frequency domain) is estimated based on a previous estimation of the received data. The basic idea hinges on the assumption that the Channel Impulse Response (CIR, time domain) is always shorter than the CP of length N_G found at each OFDM symbol. Hence, if an estimation of the CTF is available on vector $\underline{\hat{H}}$, this estimation can be improved by forcing the corresponding CIR, i.e. $\underline{\hat{h}}$ = IDFT$\{\underline{\hat{H}}\}$, to be shorter than the CP. This is done by setting to zero all those samples in vector $\underline{\hat{h}}$ that fall beyond the CP limit since they are considered to be noise. This is equivalent to eliminate the noise components that are orthogonal to the signal of interest. For a particular OFDM symbol l, this operation can be expressed in matrix form as follows

$$\underline{\tilde{H}}_l = \underline{\underline{\Theta}}_{DFT} \cdot \underline{\hat{H}}_l, \tag{9}$$

with

$$\underline{\underline{\Theta}}_{DFT} = \underline{\underline{F}}^H \cdot \underline{\underline{W}} \cdot \underline{\underline{F}}, \tag{10}$$

where $\underline{\hat{H}}_l$ is a $N\times1$ vector with the original CTF estimation for symbol l, $\underline{\tilde{H}}_l$ is the "cleaned" CTF estimation, $\underline{\underline{F}}$ is the N-point IDFT matrix, $\underline{\underline{F}}^H$ is the N-point DFT matrix and $(^H)$ stands for Hermitian transpose. The matrix $\underline{\underline{W}}$ is a $N\times N$ matrix with the form,

$$\underline{\underline{W}} = \begin{pmatrix} \underline{\underline{I}} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \tag{11}$$

with $\underline{\underline{I}}$ standing for the $N_G\times N_G$ identity matrix ($N_G < N$). The matrix $\underline{\underline{W}}$ windows the IDFT of $\underline{\hat{H}}_l$. The matrix $\underline{\underline{\Theta}}_{DFT}$ is referred to as the *Noise Reduction Matrix* (NRM) with dimension $N\times N$.

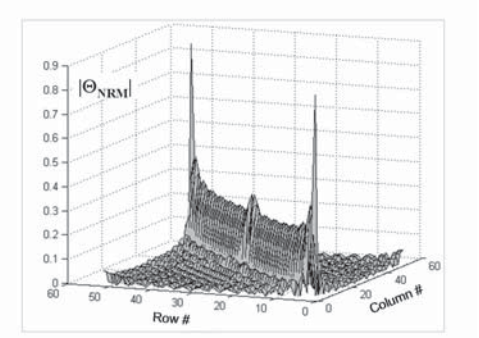
The problem in fact is more complex than this, since in a real scenario not all N sub-carriers are data-bearing sub-carriers. An example is the 802.11a standard, where only N_u out of N sub-carriers contain information, with N_u = 52 and N = 64. In this case the NRM cannot be obtained as in (10), since now the vector $\underline{\hat{H}}_l$ is a column vector with N_u elements, whereas $\underline{\underline{\Theta}}_{DFT}$ is a $N\times N$ matrix. A solution for this particular case is provided in [18], yielding a $N_u\times N_u$ matrix $\underline{\underline{\Theta}}_{NRM}$ as follows,

$$\underline{\underline{\Theta}}_{NRM} = \underline{\underline{F}}_{11}^H \cdot \left(\underline{\underline{F}}_{11} - \underline{\underline{F}}_{12} \cdot \underline{\underline{F}}_{22}^{+} \cdot \underline{\underline{F}}_{21} \right), \tag{12}$$

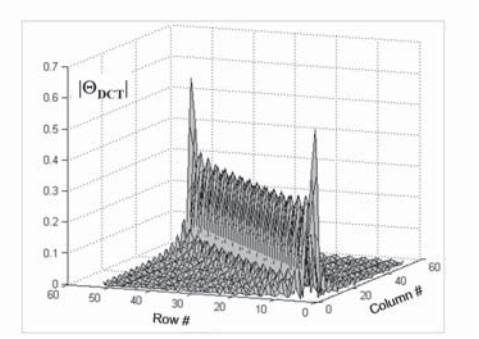
with

$$\underline{\underline{F}}_{22}^{+} = \left(\underline{\underline{F}}_{22}^H \cdot \underline{\underline{F}}_{22} + \gamma^2 \right)^{-1} \cdot \underline{\underline{F}}_{22}^H, \tag{13}$$

where γ is a dummy parameter, $0 < \gamma << N^{-1}$, used to prevent possible numerical instability in the matrix inversion. The matrices $\underline{\underline{F}}_{11}$, $\underline{\underline{F}}_{12}$, $\underline{\underline{F}}_{21}$, and $\underline{\underline{F}}_{22}$ are of dimensions $N_G\times N_u$, $N_G\times(N-N_u)$, $(N-N_G)\times N_u$, and $(N-N_G)\times(N-N_u)$, respectively, and are made of elements $W_N^{nk} = N^{-1/2} \cdot exp\{j(2\pi/N)\cdot n\cdot k\}$.



(a)



(b)

Fig. 5. Noise reduction matrices being considered: (a) Reordered 52×52 DFT-based (Θ_{NRM}); (b) 52×52 DCT-based (Θ_{DCT}).

The matrix $\underline{\underline{F}}_{11}$ corresponds to $n \in [0, N_G - 1]$ (rows 1 to N_G) and $k \in [N - (N_u/2), N - 1] \cup [1, N_u/2]$ (columns 1 to N_u). $\underline{\underline{F}}_{12}$ corresponds to $n \in [0, N_G - 1]$ (rows 1 to N_G) and $k \in [0] \cup [(N_u/2) + 1, N - (N_u/2) - 1]$ (columns 1 to $N - N_u$). $\underline{\underline{F}}_{21}$ corresponds to $n \in [N_G, N - 1]$ (rows 1 to $N - N_G$) and $k \in [N - (N_u/2), N - 1] \cup [1, N_u/2]$ (columns 1 to N_u). Finally, $\underline{\underline{F}}_{22}$ corresponds to $n \in [N_G, N - 1]$ (rows 1 to $N - N_G$) and $k \in [0] \cup [(N_u/2) + 1, N - (N_u/2) - 1]$ (columns 1 to $N - N_u$). The resulting matrix $\underline{\underline{\Theta}}_{NRM}$ is shown in Fig. 5(a) for the case *N* = 64, N_u = 52, N_G = 16. It contains 2,704 complex elements, which must be pre-computed and stored. By means of $\underline{\underline{\Theta}}_{NRM}$, a noise reduction factor given by $v_{dB} = 10 \cdot log_{10}(N^2/(N_G \cdot N_u))$ can be achieved. In the 802.11a case this reduction is as high as 7 dB. It should be noted that the matrix $\underline{\underline{\Theta}}_{NRM}$ is fixed once *N*, N_u and N_G have been selected.

The noise reduction concept explained above might be significantly simplified if the NRM is determined not based upon the DFT but on the DCT (Discrete Cosine Transform). Although the DCT is closely related to the DFT, it has a major ability to *project energy* onto a few transformed coefficients than the DFT has. Nevertheless, according to our design premise, $\underline{\hat{h}}_l$ = IDFT{$\underline{\hat{H}}_l$} should have its energy projected onto a few coefficients. As a means to reduce the CTF estimation noise, the pseudo-CIR $\underline{\hat{h}}_{pseudo,l}$ = IDCT{$\underline{\hat{H}}_l$} may be used instead of $\underline{\hat{h}}_l$. The DCT-based NRM can be written as

$$\underline{\underline{\Theta}}_{DCT} = \underline{\underline{C}}^H \cdot \underline{\underline{W}} \cdot \underline{\underline{C}}, \tag{14}$$

where $\underline{\underline{C}}$ stands for the N_u-point IDCT matrix, $\underline{\underline{C}}^H$ is the N_u-point DCT matrix, and $\underline{\underline{W}}$ is built as in (11) but now with dimensions $N_u \times N_u$. In Fig. 5(a)/(b) it can be seen that both matrices, $\underline{\underline{\Theta}}_{NRM}$ and $\underline{\underline{\Theta}}_{DCT}$, have a very similar magnitude shape, with their major coefficients concentrated around the main diagonal. Nevertheless, the matrix $\underline{\underline{\Theta}}_{DCT}$ only contains real values whereas $\underline{\underline{\Theta}}_{NRM}$ is made of complex coefficients. More interestingly, it is not necessary to pre-calculate the matrix $\underline{\underline{\Theta}}_{DCT}$, as it is the case for $\underline{\underline{\Theta}}_{NRM}$, but we might calculate a forward and reverse N_u-point (52-point in case of the IEEE 802.11a standard) DCT on the vector $\underline{\hat{H}}_l$ in order to reduce the CTF estimation noise.

B. Residual Phase Correction

After FFT calculation and channel correction, a residual phase error remains in the modulated data due to several factors: errors in the estimation of the STO and CFO, Phase Noise, and uncorrected SCFO. When applying the CE algorithm it is considered that the transmitted pilots were assigned the values $\{\pm 1\}$. Furthermore, the channel is supposed not to change significantly during a period of *L* OFDM symbols, *L* being the latency of the CE, so that after channel correction and in the absence of noise the resulting pilots are pure phasors with normalized magnitude given by

$$P_{k,l}^{\phi} \approx e^{j(\delta \cdot k + \theta_l)}, \tag{15}$$

where $\delta \propto L \cdot \xi$, $\theta_l = L \cdot c_0 + (\alpha_l - \alpha_{l-L})$ and k is the frequency index; ξ is the sampling error (in ppm), α_l is the contribution of the Phase Noise (the so-called Common Phase Error) to symbol l, and c_0 is the phase derived from a residual CFO. The method we propose [19], [20] assumes the condition $|\delta \cdot k| << 1$ be satisfied $\forall k \in [-26, +26]$. In this case (15) may be simplified by considering a first order approximation of the complex exponential, yielding

$$\begin{aligned} P_{k,l}^{\phi} &= \Re\{P_{k,l}^{\phi}\} + j \cdot \Im\{P_{k,l}^{\phi}\} \\ &= cos\,(\theta_l) - \delta \cdot k \cdot sin\,(\theta_l) \\ &+ j\,(sin\,(\theta_l) + \delta \cdot k \cdot cos\,(\theta_l))\,. \end{aligned} \tag{16}$$

In (16) four parameters are of interest namely, $cos(\theta_l)$, $sin(\theta_l)$, $\delta \cdot sin(\theta_l)$, and $\delta \cdot cos(\theta_l)$. In order to find these four parameters we must solve the linear system of equations derived from (16) when setting *k* = −21, −7, +7 and +21, corresponding to the pilot tones. Hence, the parameters $cos(\theta_l)$ and $sin(\theta_l)$ can be found straightforwardly as

$$\begin{aligned} cos\,(\theta_l) &= (1/4) \cdot \sum_{i=-21,-7,+7,+21} \Re\{P_{i,l}^{\phi}\}, \\ sin\,(\theta_l) &= (1/4) \cdot \sum_{i=-21,-7,+7,+21} \Im\{P_{i,l}^{\phi}\}. \end{aligned} \tag{17a}$$

Regarding the parameters $\delta \cdot sin(\theta_l)$ and $\delta \cdot cos(\theta_l)$, the exact expressions are as follows,

$$\delta \cdot sin(\theta_l) = \frac{2}{126}\Re\{P^{\phi}_{-21,l}\} + \frac{3}{126}\Re\{P^{\phi}_{-7,l}\} - \frac{3}{126}\Re\{P^{\phi}_{+7,l}\} - \frac{2}{126}\Re\{P^{\phi}_{+21,l}\},$$
$$\delta \cdot cos(\theta_l) = \frac{2}{126}\Im\{P^{\phi}_{+21,l}\} + \frac{3}{126}\Im\{P^{\phi}_{+7,l}\} - \frac{3}{126}\Im\{P^{\phi}_{-7,l}\} - \frac{2}{126}\Im\{P^{\phi}_{-21,l}\},$$

which have been modified in order to simplify the scaling by the factor 1/126, yielding

$$\delta \cdot sin(\theta_l) \approx \frac{2}{128}\Re\{P^{\phi}_{-21,l}\} + \frac{3}{128}\Re\{P^{\phi}_{-7,l}\} - \frac{3}{128}\Re\{P^{\phi}_{+7,l}\} - \frac{2}{128}\Re\{P^{\phi}_{+21,l}\},$$
$$\delta \cdot cos(\theta_l) \approx \frac{2}{128}\Im\{P^{\phi}_{+21,l}\} + \frac{3}{128}\Im\{P^{\phi}_{+7,l}\} - \frac{3}{128}\Im\{P^{\phi}_{-7,l}\} - \frac{2}{128}\Im\{P^{\phi}_{-21,l}\}. \quad (17b)$$

The foregoing method saves a significant amount of hardware, since neither an arctangent block nor an NCO is needed for RPE estimation and correction, respectively.

IV. The Digital Timing Loop

The general scheme of the IRx shown in Fig. 1 includes a so-called Digital Timing Loop (DTL). The purpose of the DTL is to estimate and correct the SCFO. Each OFDM symbol is composed of 80 samples, before CP extraction and FFT operation, with a sampling rate f_s = 20 MHz. In the case of a sampling oscillator with e.g. 20 ppm frequency error, this turns into f_s = 20,000,400 Hz. Thus 80.0016 samples are obtained for the initial symbol instead of exactly 80, i.e. a timing error of 0.0016 samples. This timing error is not fixed, but it will be 0.0032 samples for the second symbol, 0.0048 for the third one and so on. In essence, the SCFO will be observed as a dynamic timing error that has to be monitored throughout reception. Considering the case of a 6 Mbps transmission, the 802.11a standard allows a frame length of up to 1,367 data symbols, which means that the last OFDM symbol will be affected by a timing error of about 2.2 samples. In our consideration the total SCFO may be as high as 80 ppm (combining the effects from Tx and Rx), yielding a maximum accumulated timing error of 8.8 samples. Since the timing error appears as a linear phase after FFT operation, pilots are very well suited to estimate it. The method shown in the foregoing section for RPE estimation and correction is *a posteriori*, i.e. no attempt is done to correct the main sources causing the phase error, but only the phase error itself. Hence, we need not only a method to estimate the SCFO based on the pilots but also a way to correct for it prior to FFT operation in order to avoid ICI.

A. Timing Error Discriminator

In a first stage, the variable timing error must be estimated. In the estimation we make use of the phase error signal provided by the RPE estimator, i.e. $P^{\phi}_{k,l}$ in (15). The estimator is based on a solution proposed by Yang in [21], in which two reference sequences are defined, namely

$$C^{early}_{p,l} \hat{=} e^{j\frac{\pi}{N}\cdot p},$$
$$C^{late}_{p,l} \hat{=} e^{-j\frac{\pi}{N}\cdot p}, \quad (18)$$

where p corresponds to the pilot sub-carrier position and l is the symbol index.

The RPE signal $P^{\phi}_{k,l}$ is compared with these two references through correlation, thus yielding

$$a(l) = |R_{late}(l)|^2 - |R_{early}(l)|^2, \quad (19)$$

with

$$R_{early}(l) = \sum_{i=0}^{P-1} P^{\phi}_{p_0+i\cdot\Delta,l} \cdot \left(C^{early}_{p_0+i\cdot\Delta,l}\right)^* \approx \sum_{i=0}^{P-1} e^{j\frac{2\pi}{N}\cdot(p_0+i\cdot\Delta)\cdot\left(\Delta t_l-\frac{1}{2}\right)} + V_{early}(l),$$
$$R_{late}(l) = \sum_{i=0}^{P-1} P^{\phi}_{p_0+i\cdot\Delta,l} \cdot \left(C^{late}_{p_0+i\cdot\Delta,l}\right)^* \approx \sum_{i=0}^{P-1} e^{j\frac{2\pi}{N}\cdot(p_0+i\cdot\Delta)\cdot\left(\Delta t_l+\frac{1}{2}\right)} + V_{late}(l), \quad (20)$$

where $V_{early}(l)$ and $V_{late}(l)$ are Gaussian noise components. In (20) it has been considered that pilot sub-carriers are at position $p = p_0 + i\cdot\Delta$, with $0 \le i \le P-1$, P being the total number of pilots per OFDM symbol, and Δ the pilot distance. The approximation done in (20) applies when $P^{\phi}_{k,l}$ adheres to the approximation in (15). The total timing error (in samples) at symbol l in (20) is

$$\Delta t_l = \left(t_\theta - \hat{t}_{\theta,l}\right) + \xi \cdot L \cdot (N + N_G), \quad (21)$$

where t_θ is a residual timing synchronization error ($|t_\theta| < 0.5$ samples), $\hat{t}_{\theta,l}$ is an estimation of t_θ at symbol l, and ξ stands for the SCFO (in ppm). In (20) it is further considered that $|\Delta t_l| \le 0.5$. After low-pass filtering (19), we finally obtain the timing discriminator as follows (sub-index l has been omitted for clarity)

$$S(\Delta t) = \left|\frac{sin\left(\frac{\pi}{N}\cdot P\cdot\Delta\cdot\left(\Delta t+\frac{1}{2}\right)\right)}{sin\left(\frac{\pi}{N}\cdot\Delta\cdot\left(\Delta t+\frac{1}{2}\right)\right)}\right|^2 - \left|\frac{sin\left(\frac{\pi}{N}\cdot P\cdot\Delta\cdot\left(\Delta t-\frac{1}{2}\right)\right)}{sin\left(\frac{\pi}{N}\cdot\Delta\cdot\left(\Delta t-\frac{1}{2}\right)\right)}\right|^2, \quad (22)$$

where P = 4, Δ = 14 and N = 64 in the 802.11a case.

B. Timing Error Correction

The parameter of interest is the relative error existing between the sampling period T_s at the Analog-to-Digital converters (ADC), and the corrected (ideal) sampling time T_I, i.e. $T_I/T_s = 1 + \xi$. These two sampling periods are related as follows,

$$i \cdot T_I + \tau_I \cdot T_I = m_i \cdot T_s + \mu_i \cdot T_s, \quad (23)$$

with

$$m_i = \lfloor i \cdot T_I + \tau_I \cdot T_I \rfloor, \tag{24}$$

where $0 \leq \mu_i < 1$ is the *fractional delay*, m_i is the *basepoint*, i is the discrete timing variable after timing correction (integer value), whereas τ_I represents a fractional part of T_I. The function $\lfloor x \rfloor$ rounds x to the nearest integer towards minus infinity. The timing error compensation is driven by a control block (see Fig. 1), which contains a *control word*, $w(l)$. This parameter is updated on a symbol basis, i.e. every 4 μs, and provides the latest estimate of the ratio T_I/T_s as follows,

$$w(l+1) = w(l) + K_w \cdot e(l), \tag{25}$$

with

$$e(l+1) = e(l) + K_e \cdot a(l), \tag{26}$$

being $a(l)$ as in (19). The parameter K_e defines the bandwidth of the low-pass filter in (26) and it was selected to be 0.01. The parameter K_w is given as $K_w = (2 \cdot S_{max} \cdot (N + N_G))^{-1}$, where S_{max} is the maximum value of $S(\Delta t)$ in (22).

The parameters m_i and μ_i used in the variable interpolator will be recursively computed as explained in [7, page 523]. We already expressed $i \cdot T_I + \tau_I \cdot T_I$ as a function of (m_i, μ_i) in (23). The next sample $(i+1) \cdot T_I + \tau_I \cdot T_I$ is given by

$$(i+1) \cdot T_I + \tau_I \cdot T_I = m_i \cdot T_s + \left(\mu_i + \left(\frac{T_I}{T_s}\right)\right) \cdot T_s. \tag{27}$$

By replacing in (27) the unknown ratio (T_I / T_s) by its estimate $w(m_i)$ we obtain

$$\begin{aligned}(i+1) \cdot T_I + \tau_I \cdot T_I = {} & m_i \cdot T_s \\ & + \lfloor \mu_i + w(m_i) \rfloor \cdot T_s \\ & + [\mu_i + w(m_i)]_{mod1} \cdot T_s. \end{aligned} \tag{28}$$

From the previous, it readily follows the recursion for the estimates,

$$\begin{aligned} m_{i+1} = {} & m_i + \lfloor \mu_i + w(m_i) \rfloor, \\ \mu_{i+1} = {} & [\mu_i + w(m_i)]_{mod1}. \end{aligned} \tag{29}$$

In order to obtain the value for μ_i based on the control word $w(m_i)$ we define the function

$$\eta(m_i, d) \hat{=} \mu_i + w(m_i) - d, \tag{30}$$

with d = 0, 1, 2, ...

At the basepoint m_i the value $\eta(m_i, 0)$ is stored in a b-bit register. At every T_s cycle the value of the register is decremented by 1, i.e.

$$\eta(m_i, d+1) = \eta(m_i, d) - 1. \tag{31}$$

As long as $\eta(m_i, d) > 1$, there obviously exists an integer $\lfloor \mu_i + w(m_i) \rfloor > m_i + d$. The criterion to obtain the next basepoint m_{i+1} is $\eta(m_i, d_{min}) < 1$, where d_{min} is the smallest integer for which the condition is fulfilled. Thus, the decrease operation is continued until the condition $\eta(m_i, d_{min}) < 1$ is detected. By definition, the register content $\eta(m_i, d_{min})$ equals μ_{i+1}. Afterwards, the operations are continued for m_{i+1} with the initial value

$$\eta(m_{i+1}, 0) = \eta(m_i, d_{min}) + w(m_{i+1}). \tag{32}$$

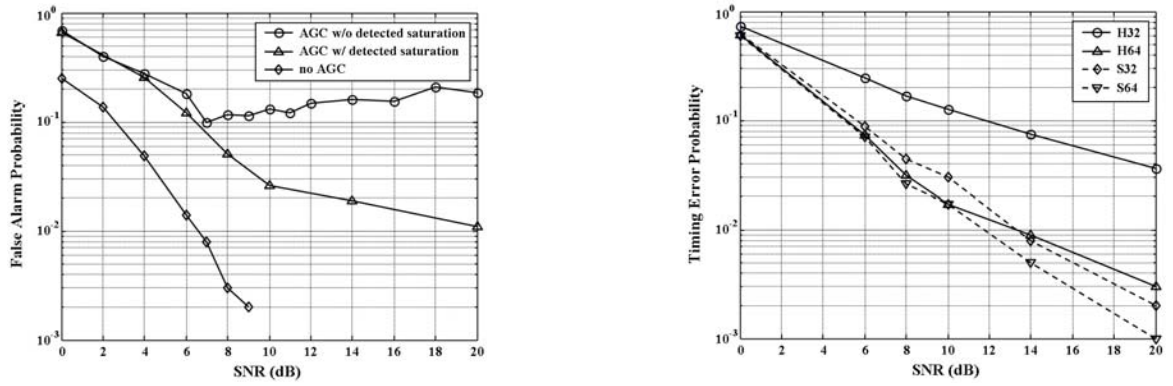
The timing error correction block in Fig. 1 is based on a first order Lagrange polynomial interpolator and makes use of a Farrow structure [7], [22]. Higher order interpolators cannot be used since the DTL becomes *unstable*. The reason for this instability is related to the considerations made in (16) for calculation of $P_{k,l}^{\phi}$, which no longer hold when high order interpolators are used.

V. Simulation Results

This section analyzes the performance of the synchronizer, the CE and the DTL under different transmit conditions through extensive computer simulations. We already mentioned in Section II that the synchronizer was mainly affected by the AGC. Since the attenuation suffered by the transmitted OFDM frame is unknown to the receiver, an AGC able to apply a variable amplification is mandatory prior to ADC. The AGC should be capable of keeping the signal inside a certain voltage range given by the bias voltage of the ADCs. The frame detector found in the synchronizer should be robust against two main effects caused by the AGC:

1) Since the AGC is not able to distinguish the signal of interest from the noise, in the absence of any signal the noise will be amplified in the worst case to the voltage limits of the ADCs. These high noise levels should not provoke false frame detections.
2) In a high SNR situation, the AGC has to change very quickly from a high amplification level to a lower one when the signal is received. Since the AGC cannot react instantaneously to sudden changes in the input power level, the AGC output signal will be heavily saturated for a certain time.

The simulation results related to the synchronizer are depicted in Fig. 6. A channel model A as given in [23] together with a normalized CFO of +1.2 are used in all cases. This corresponds to a Non-Line-Of-Sight (NLOS) channel with a maximum delay spread of 390 ns (50 ns rms). The results for the False Alarm Probability (FAP) are shown in Fig. 6(a). The model used for the AGC considers that only amplitude distortions (saturation) but no phase distortions are introduced into the signal, since these may lead to false frequency estimations. The filter parameters in the feedback loop of the AGC where selected in order to achieve a settling time in which approximately 64 samples (3.2 μs) of the preamble symbols where completely saturated at SNR = 35 dB (worst case settling time). In the definition of FAP used in our simulations, a frame was considered to be correctly detected when the detected starting point was inside a range of ± 16 samples from the "ideal" point, i.e. when no AGC and no channel are used. Fig. 6(a) shows that the FAP decreases with increasing SNR until a certain value of SNR is reached. From this point on, the distortion due to saturation becomes the dominant effect on the preambles and the FAP degrades as



(a) (b)

Fig. 6. Simulation results for the synchronizer (channel A, ϵ = 1.2): (a) Simulated FAP for the frame detector; (b) Simulated TEP at the output of the crosscorrelator.

the SNR increases. Nevertheless, since saturation is easily detectable at the ADC, the previous effect can be highly mitigated by setting to zero all those saturated samples before being delivered to the frame detector. The obtained standard deviation for the normalized frequency offset estimator shows no dependency on the AGC and has a minimum bound of 0.01, i.e. 1% of the sub-carrier spacing. This value helps in determining the number of bits necessary to represent the frequency offset in the arctangent calculator used in the synchronizer. Finally, Fig. 6(b) depicts the Timing Error Probability (TEP) derived from the crosscorrelator. Symbol timing is provided by the position of the first significant peak coming out from the crosscorrelator. The ideal position of the peak, i.e. m_1 in Fig. 2, is known beforehand and a timing error occurs when the estimated position of the peak differs in more than ± 2 samples from the ideal position. Nevertheless this definition of the timing error only makes sense if the CP of the symbol being received immediately after the preamble symbols is considered to be 14 samples long instead of 16 in order to compensate positive timing errors. Four possible versions of the crosscorrelator have been tested depending on the length of the reference signal $c_{REF}(m)$, either 32 or 64 samples, and the type of multiplier, either 1-bit XNOR-based multipliers (Hard crosscorrelator) or floating point multipliers (Soft crosscorrelator) with the number of bits determined by the computer on which the simulation is being run. Results shown in Fig. 6(b) indicate that the selection of a 32-sample H-crosscorrelator may not be appropriate and should be increased to 64 samples. Despite of these results, our first version of the synchronizer considers only a reference of 32 samples in order to reduce the signal processing latency as much as possible.

Fig. 7 depicts the Mean-Square Error (MSE) performance of the proposed channel estimator considering all data rates defined for the 802.11a standard. Channel models A and D [23] are considered in the simulations. Channel D corresponds to a Line-Of-Sight (LOS) channel with a maximum delay spread of 1,050 ns (140 ns rms). In both cases a Doppler frequency of 58 Hz (v = 3 m/s, F_C = 5,805 MHz) was used. In order to smooth the simulation results, we firstly tested 20 different seeds and looked for the one representing an average channel. This seed was used afterwards for the MSE estimation. Each point in Fig. 7 is obtained after averaging the MSE in 10 trials where a frame containing 37 OFDM data symbols is transmitted at each trial. Furthermore, six soft bits were used during demodulation together with a traceback length of 50 bits in the Viterbi decoder. In order to reduce complexity, a hard-output Viterbi decoder was considered. The figures show a substantial improvement in the MSE when a LOS channel is present. The abrupt decrease of the MSE indicates the point from which on the Viterbi decoder is able to provide fully correct output bits. The correctness of these bits is crucial in order to assure the stability of the CE, specially at the higher transmission rates. Furthermore, Fig. 7 also shows that it will be extremely difficult to obtain the maximum data rates (48 and 54 Mbps) in a real wireless channel, even with LOS, since these rates require a SNR well above 30 dB. The standard 802.11a [1] specifies a Packet Error Rate (PER) of 10% measured on 1000-byte frames, which is equivalent to a BER = 1.25e-5. Fig. 8 shows the results of our Monte-Carlo BER simulations based on 1000-byte frames. The same channel seed as in Fig. 7 was used in Fig. 8. It can be seen from Fig. 8 that the higher modulation schemes require very high SNR in order to achieve the minimum BER and we may use them only in very limited scenarios.

Finally, Fig. 9 shows the simulation results for the timing control loop. We simulated only two transmission modes, i.e 12 and 54 Mbps, and represented the Error Vector Magnitude (EVM) as defined in [1]. Frames with 152 OFDM data symbols were generated in all the cases, since this is the maximum number of OFDM data symbols per frame in the 54 Mbps case. The clock error was set to ξ = -80 ppm, which represents a worst case scenario where the actual sampling frequency is below the reference value. Though an ideal channel estimator was taken into consideration, the effects derived from the processing latency involved in the decision-directed channel estimator are included in the simulation results. Hence, the 12 Mbps case involves a processing latency of three OFDM symbols. For the 54 Mbps case, the processing latency is only one OFDM symbol. As it can be seen from Fig. 9, the proposed solution achieves an improvement in terms of EVM in both cases, although this improvement is less significant in case of a NLOS channel.

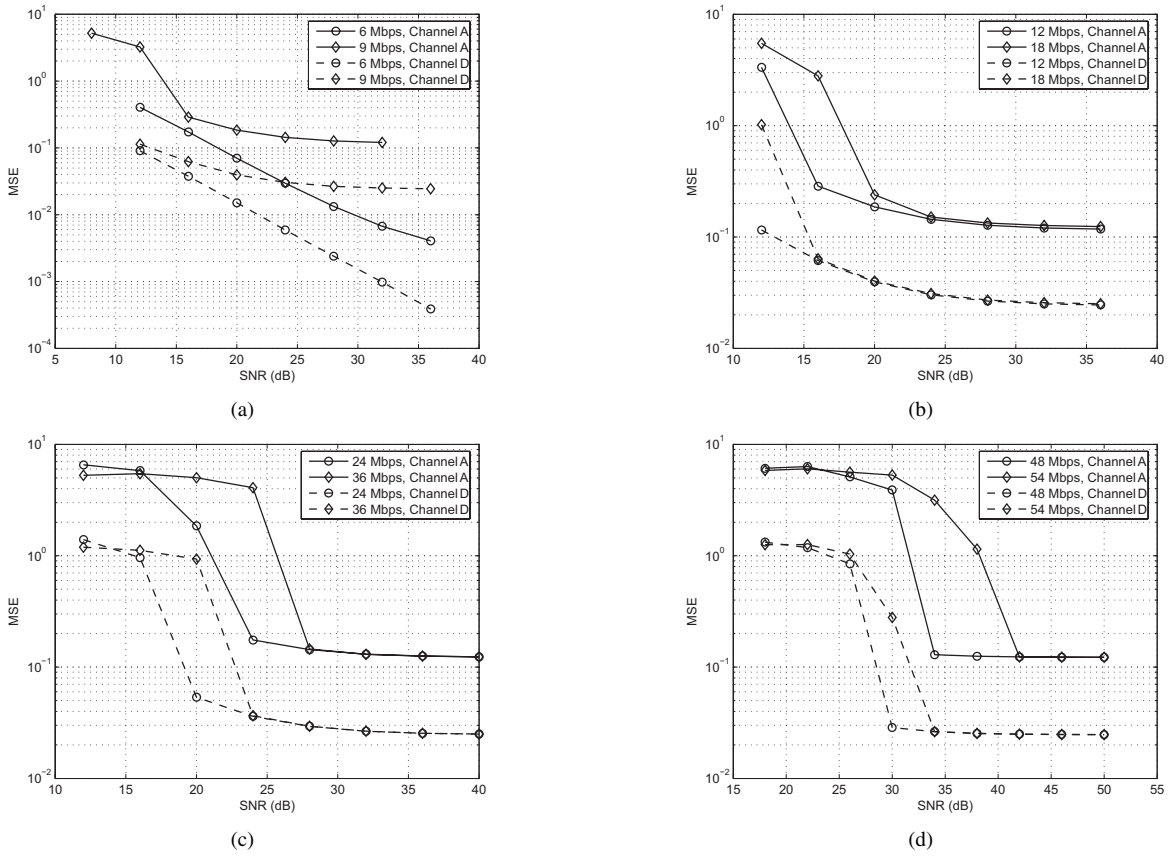


Fig. 7. MSE versus SNR for the proposed modified CD3 channel estimator according to the 802.11a standard: (a) BPSK modulation schemes; (b) QPSK modulation schemes; (c) 16-QAM modulation schemes; (d) 64-QAM modulation schemes.

VI. Conclusion

We have investigated the implementation of the Inner Receiver of an OFDM-WLAN system based on the IEEE 802.11a standard. Solutions for the most critical blocks, i.e. Synchronizer, Channel Estimator and Digital Timing Loop, have been proposed and analyzed under careful consideration of nearly realistic transmit conditions. Hence, although our investigations reveal that the Synchronizer is strongly influenced by the gain control, the proposed architecture is shown to be relatively robust against the AGC effects. Regarding the Channel Estimator, a decision-directed architecture has been examined. Two novel solutions have been incorporated into the design in order to improve the performance. Firstly, a novel DCT-based noise reduction filter exploits the energy compression capabilities of the DCT as a means to reduce the channel estimation noise with a moderate computational load. Secondly, the residual phase error is eliminated by means of an innovative estimator that extremely simplifies the traditional solution based on arctangent plus NCO operation. In order to derive a simple time tracking algorithm we have made use of concepts already established in the literature. However, the way these concepts are applied to an OFDM receiver is novel in our solution. The proposed solution has proven to be applicable in both LOS and NLOS channels. However, the performance of the DTL is limited by the fact that only first order Farrow interpolators assure stability of the algorithm.

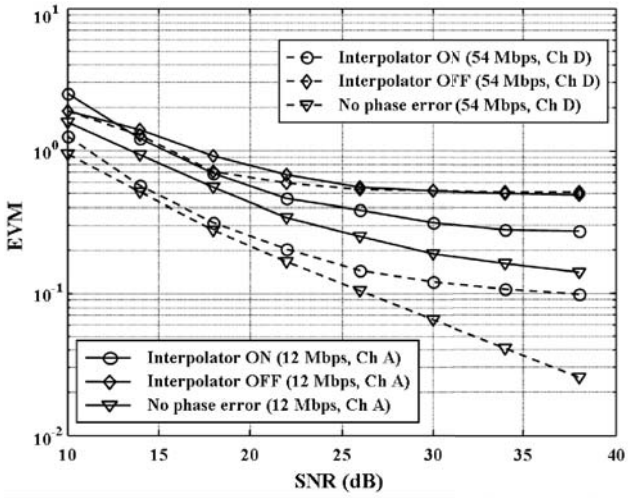


Fig. 9. EVM versus SNR considering the proposed variable timing estimator for 12 Mbps (channel A), and 54 Mbps (channel D) with $\xi = -80$ ppm.

Acknowledgment

The authors would like to thank Dr. Robert J. Peichocki from University of Bristol for his insightful comments to this paper, and Prof. Ulrich Ramacher and Dr. Bertram Gunzelmann for their unconditional support in publishing this paper. We also would like to thank the anonymous reviewers for their constructive criticism.

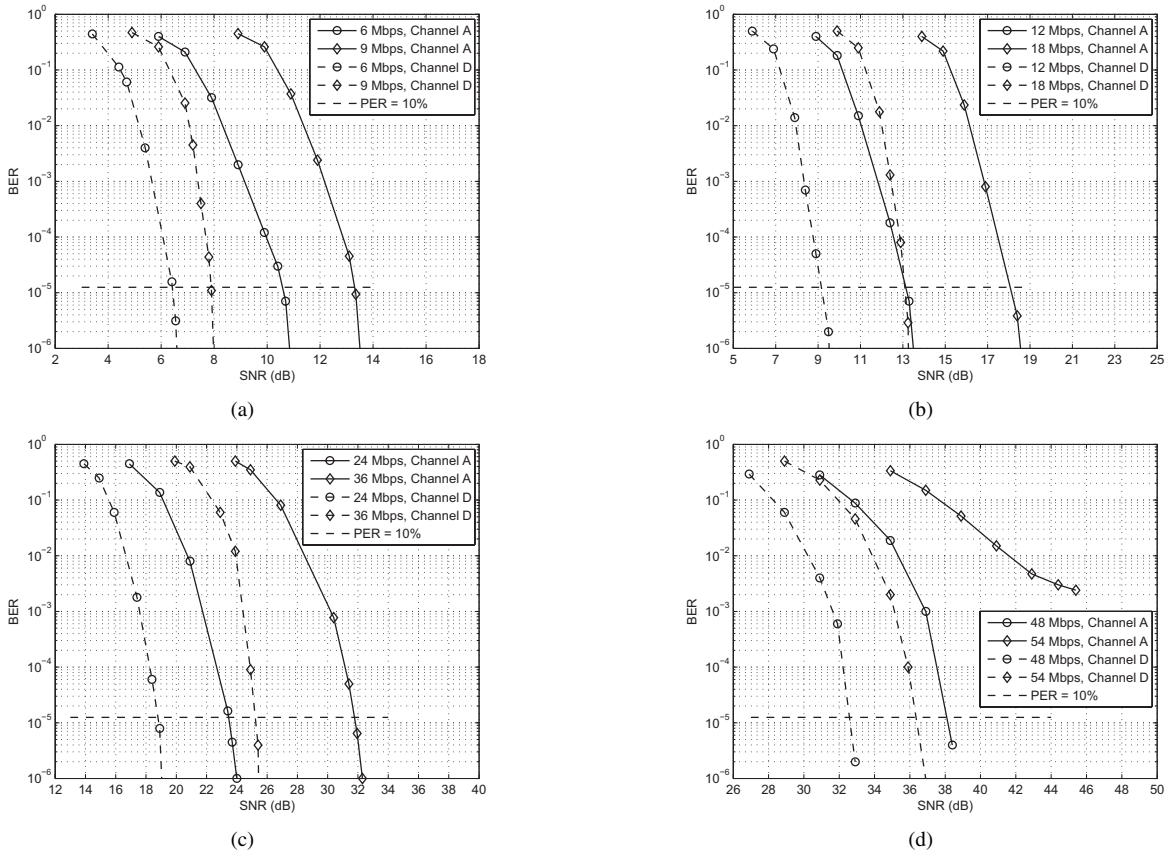


Fig. 8. BER versus SNR for the proposed modified CD3 channel estimator according to the 802.11a standard: (a) BPSK modulation schemes; (b) QPSK modulation schemes; (c) 16-QAM modulation schemes; (d) 64-QAM modulation schemes.

REFERENCES

[1] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: High speed physical layer in the 5 GHz band," IEEE P802.11a/D7.0, Part II, 1999.

[2] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. Amendment 4: Further higher data rate extension in the 2.4 GHz band," IEEE 802.11G. Standard for IT - Telecommunications and information exchange between systems LAN/MAN - Part II, 2003.

[3] [Online.] Available: www.wigwam-project.com

[4] [Online.] Available: www.multibandofdm.org

[5] M. Speth, S. A. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for wireless broad-band systems using OFDM - Part I," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1668–1677, Nov. 1999.

[6] P. Robertson and S. Kaiser, "Analysis of the effects of phase-noise in orthogonal frequency division multiplex (OFDM) systems," in *Proc. IEEE ICC*, June 1995, pp. 1652–1657.

[7] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital Communication Receivers: Synchronization, Channel Estimation, and Signal Processing*. New York: Wiley, 1998.

[8] M. Krstić, A. Troya, K. Maharatna, and E. Grass, "Optimized low-power synchronizer design for the IEEE 802.11a standard," in *Proc. IEEE ICASSP*, Apr. 2003, vol. 2, pp. 333–336.

[9] A. Troya, K. Maharatna, M. Krstić, and E. Grass, "Method and device for frame detection and synchronizer." PCT Patent WO 2004/008706 A2, pending, Jan. 22, 2004.

[10] K. Maharatna, A. Troya, S. Banerjee, and E. Grass, "A CORDIC like processor for computation of arctangent and absolute magnitude of a vector," in *Proc. IEEE ISCAS*, May 2004, vol. 2, pp. 713–716.

[11] K. Maharatna, A. Troya, S. Banerjee, E. Grass, and M. Krstić, "A 16-bit CORDIC rotator for high-speed wireless LAN," in *Proc. IEEE PIMRC*, Sep. 2004, vol. 3, pp. 1747–1751.

[12] K. Maharatna, S. Banerjee, E. Grass, M. Krstić, and A. Troya, "Modified virtually scaling-free CORDIC rotator algorithm and architecture," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 11, pp. 1463–1474, Nov. 2005.

[13] 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.

[14] B. Stantchev and G. Fettweis, "Burst synchronization for OFDM-based cellular systems with separate signaling channel," in *Proc. IEEE VTC*, May 1998, pp. 758–762.

[15] V. Mignone and A. Morello, "CD3-OFDM: A novel demodulation scheme for fixed and mobile receivers," *IEEE Trans. Commun.*, vol. 44, no. 9, pp. 1144–1151, Sep. 1996.

[16] J.-J. van de Beek, O. Edfors, M. Sandell, S. K. Wilson, and P. O. Börjesson, "On channel estimation in OFDM systems," in *Proc. IEEE VTC*, July 1995, vol. 2, pp. 815–819.

[17] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Börjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Commun.*, vol. 46, no. 7, pp. 931–939, July 1998.

[18] H. Schmidt, V. Kühn, K.-D. Kammeyer, R. Rueckriem, and S. Fechtel, "Channel tracking in wireless OFDM systems," in *Proc. 5th World Multi-Conference on Systemics, Cybernetics and Informatics*, July 2001, vol. 4, pp. 402–406.

[19] A. Troya, M. Krstić, and K. Maharatna, "Simplified residual phase correction mechanism for the IEEE 802.11a standard," in *Proc. IEEE VTC-Fall*, Oct. 2003, vol. 2, pp. 1137–1141.

[20] A. Troya, K. Maharatna, and M. Krstić, "Verfahren und Vorrichtung zur Fehlerkorrektur von Multiplex-Signalen." PCT Patent WO 2004/036863 A1, pending, Apr. 29, 2004.

[21] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "An improved combined symbol and sampling clock synchronization method for OFDM systems," in *Proc. IEEE Wireless Commun. Networking Conf.*, Sep. 1999, vol. 3, pp. 1153–1157.

[22] L. Erup, F. M. Gardner, and R. A. Harris, "Interpolation in digital modems - Part II: Implementation and performance," *IEEE Trans. Commun.*, vol. 41, no. 6, pp. 998–1008, June 1993.

[23] "Criteria for comparison," ETSI Technical Report 30701F, BRAN WG3 PHY Subgroup, May 1998.

Alfonso Troya (M'95) was born in Barcelona, Spain, in 1975. He received the M.Sc. degree in Telecommunications Engineering from the Technical University of Catalonia, Barcelona, Spain, in 1999, and the Dr.-Ing. degree from Brandenburg University of Technology, Cottbus, Germany, in 2004.

He joined the IHP (Institute for High Performance microelectronics), Frankfurt (Oder), Germany, in 1999, as a Research Associate in the Wireless Communication Systems Department, where he worked on the development and implementation of digital signal processing algorithms for broadband wireless communication systems. In October 2004 he joined Infineon Technologies AG, Munich, Germany, as an Algorithm Concept Engineer. He is currently involved in the development of OFDM-based communication systems and their implementation on Software-Defined Radio architectures.

Dr. Troya is a member of the IEEE Signal Processing Society.

Koushik Maharatna (M'02) received the M.Sc. degree in Electronic Science from Calcutta University, Calcutta, India, in 1995 and the Ph.D. degree from Jadavpur University, Calcutta, India, in 2002. From 1996 to 2000, he was involved in projects sponsored by the Government of India undertaken at the Indian Institute of Technology (IIT), Kharagpur, India. From 2000 to 2003, he was a Research Scientist with IHP, Frankfurt (Oder), Germany. During this phase, his main involvement was related to the design of a single-chip modem for the IEEE 802.11a standard. In August 2003, he joined the Department of Electrical & Electronics Engineering, University of Bristol, Bristol, UK as a Lecturer. From October 2006 he is with the School of Electronics and Computer Science, University of Southampton, where he is currently a Senior Lecturer. His research interests include development of VLSI architectures for the application in DSP and communication, computer arithmetic, low-power design, and analog signal processing.

Dr. Maharatna has served as session chair for IEEE ISCAS 2005 and VLSI design Conference 2006, and also acted as a reviewer for several IEEE Journals and Conferences. He is currently a member of the Engineering and Physical Research Council (EPSRC) college in the UK. He is a member of the IEEE Circuits and Systems Society.

Miloš Krstić was born in Niš, Serbia, in 1973. He received the Dipl.-Ing. and the M.Sc. degrees in Electronics from the University of Niš, Serbia, in 1997 and 2001, respectively, and the Dr.-Ing. degree from Brandenburg University of Technology, Cottbus, Germany, in 2006.

Since 2001 he has been with IHP, Frankfurt (Oder), Germany, as a Research Associate within the Wireless Communication Systems Department, where he is currently working on low-power digital design techniques for wireless applications and globally-asynchronous locally-synchronous (GALS) methodologies for digital systems integration.

Eckhard Grass received his Dr.-Ing. degree in Electronics from the Humboldt University, Berlin, Germany, in 1992.

He worked as a Visiting Research Fellow at Loughborough University (U.K.) from 1993 to 1995 and as a Senior Lecturer in Microelectronics at the University of Westminster, London, U.K., from 1995 to 1999. He has been with IHP, Frankfurt (Oder), Germany, since 1999, where he currently leads a project on the development and implementation of a wireless broadband communication system in the 60 GHz band. His research interests include data-driven (asynchronous) signal processing structures and low-power VLSI implementation of communication systems.

Ulrich Jagdhold received the Diploma in Physics (M.Sc. degree) from the Technical University of Dresden, Dresden, Germany, in 1987. From 1987 to 1996, he was with the Technology Integration Group of the IHP, Frankfurt (Oder), Germany, working on CMOS, BiCMOS, and SiGe technologies and device physics. In 1997 he joined the Wireless Communication Systems Department of IHP, where he has been working on WLAN system development projects, focusing on baseband integration issues and ASIC design, including development of digital CMOS libraries.

Rolf Kraemer (M'79) received his Diploma and Ph.D. degrees in Electrical Engineering and Computer Science from the RWTH Aachen, Germany, in 1979 and 1985, respectively. He has worked for 15 years in R&D of communication and multimedia systems at Philips Research Laboratories in Hamburg and Aachen. Since 1998 he is a professor of systems at the Brandenburg University of Technology, Cottbus, Germany. He also leads the Wireless Communications Systems Department of the IHP, where his research focus is on wireless Internet systems spanning from application down to Systems-on-Chip. He is co-founder of the startup company *lesswire AG*, where he holds the position of the CTO.

Prof. Kraemer has published over 150 conference and journal papers, and holds 16 international patents. He is a member of the IEEE Computer Society, the VDE-NTG, and the German Informatics Society.