Efficient Integer Frequency Offset Estimation Architecture for Enhanced OFDM Synchronization

Thinh Hung Pham, *Student Member, IEEE*, Suhaib A. Fahmy, *Senior Member, IEEE*, and Ian Vince McLoughlin, *Senior Member, IEEE*

***Abstract*— In orthogonal frequency-division multiplexing (OFDM) systems, integer frequency offset (IFO) causes a circular shift of the subcarrier indices in the frequency domain. The IFO can be mitigated through strict RF front-end design, which tends to be expensive, or by strictly limiting mobility and channel agility, which constrains operating scenarios. The IFO is, therefore, often estimated and removed at baseband, allowing implementations to benefit from the relaxed RF front-end specifications and to be tolerant to both Doppler shift and multistandard channel selection. This paper proposes a novel architecture for the IFO estimation which achieves reduced power consumption and lower computational cost than contemporary methods, while achieving excellent estimation performance, close to theoretically achievable bounds. A pilot subsampling technique enables fourfold resource sharing to reduce the computational cost, while multiplierless computation yields further power reduction. Performance exceeds that of the conventional techniques, while being much more efficient. When implemented on field-programmable gate array for IEEE 802.16-2009, the dynamic power reductions of 78% are achieved. The architecture and method is applicable to other OFDM standards including IEEE 802.11 and IEEE 802.22.**

***Index Terms*— Digital signal processing (DSP), field programmable gate arrays (FPGAs), orthogonal frequency-division multiplexing (OFDM).**

I. Introduction

ORTHOGONAL frequency-division multiplexing (OFDM) has been widely adopted for both wired and wireless communication systems due to its well-document robustness to multipath and spectral efficiency. However, it is sensitive to the receiver synchronization errors such as carrier frequency offset (CFO) which causes intercarrier interference (ICI), and which is exacerbated by issues such as motion-induced Doppler shifts and local oscillator instability [1]. Large CFO is split into fractional frequency offset (FFO) and integer frequency offset (IFO) for estimation [2]–[6]. The IFO causes a circular shift of the subcarrier in the frequency domain while FFO results in ICI due to lost orthogonality between subcarriers. In many published works on OFDM synchronization, the CFO is split into coarse- and fine-frequency offset estimation processes that determine the IFO and FFO, respectively [7].

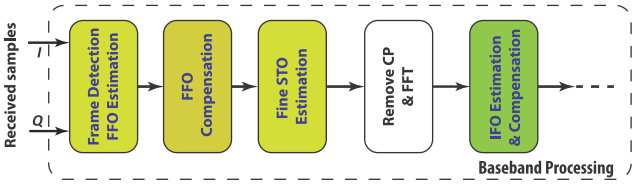

Fig. 1. Baseband processing block diagram.

Fig. 1 shows the typical FFO and IFO estimation process for OFDM. Nogami's method [8] estimates IFO by searching for a correlation maxima between the known preamble symbol and a cyclically shifted version of the received preamble symbol in the frequency domain. The technique is simple, but its performance degrades significantly in frequency selective channels. Maximum likelihood estimation has been used to estimate IFO [5], [6], based upon the observation of pilots from two consecutive OFDM symbols. A reduced complexity alternative is only to compute over one preamble OFDM symbol, using differential encoding among adjacent subcarriers and correlation estimation [2], [3]. Li *et al.* [9] proposed a method using a cross ambiguity function based on an energy-detection metric, which can be computed in the time domain. This provides high accuracy and gives a full range estimation of the IFO in the presence of frequency selective fading, but requires an exhaustive search. Although there has been significant research on IFO estimation methods, this has primarily been restricted to studies in simulation.

Field-programmable gate arrays (FPGAs) have been used to implement software radio systems for over two decades, and provide an ideal platform which is higher performance, and lower power compared with the processor-based software radio [10]. Several authors have presented the hardware implementations of the OFDM-based systems in [11]–[14], but these implementations do not include IFO estimation (only FFO). Without the IFO correction, the worst case CFO must not exceed the largest supported FFO. This can be achieved in practice through very strict and potentially expensive constraints on the RF front-end design, and avoiding application that significantly increase CFO, including mobility (i.e., Doppler-induced CFO [4]) and cognitive radio (switching between multiple frequency bands leads to CFO [15], [16]). Like many recent works [17]–[19],

Manuscript received February 16, 2015; revised May 14, 2015; accepted June 21, 2015. Date of publication August 7, 2015; date of current version March 18, 2016.

T. H. Pham and S. A. Fahmy are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: hung3@e.ntu.edu.sg; sfahmy@ntu.edu.sg).

I. V. McLoughlin is with the School of Computing, University of Kent, Kent CT2 7NZ, U.K. (e-mail: ivm@ustc.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2015.2453207

this paper is concerned with the IFO estimation, either to support large-CFO application scenarios, or to enable cost saving through relaxing strict front-end design criteria. While the state-of-the-art IFO estimation methods achieve good performance, they rely on cross correlation between received signal and known preamble. These theoretical methods lead to very large hardware resource consumption when implemented. At present, there is a lack of published IFO estimation implementation techniques, thus this paper proposes and evaluates an efficient architecture for implementation of IFO estimation. We show that this novel method achieves very accurate IFO estimation on low-power FPGAs with minimal hardware cost, at significantly lower power than the existing solutions. Implementation of this technique in a radio system can allow relaxed front-end design constraints, potentially lowering design cost, as well as enabling greater robustness in high mobility applications.

This paper is organized as follows. Section II describes the signal model for the IFO estimation, while Section III introduced the proposed method in comparison with previous work. Section IV details simulations used to evaluate the method against the conventional approaches. Section V presents the implementation on FPGA and discusses resource requirements and power consumption issues. Finally, the conclusion is drawn in Section VI.

II. Integer Frequency Offset Estimation

We consider a signal *x(n)* of an OFDM system with inverse fast Fourier transform (FFT) length *N*. Assuming that the signal is transmitted over a frequency selective channel with impulse response (CIR) h and length, L_h, corrupted by additive white Gaussian noise (AWGN), the received signal, with frequency offset and timing offset, is expressed in the time domain as

$$y(n) = \sum_{l=0}^{L_h-1} h(l)x(n-\tau-l)e^{i\left(2\pi\xi\frac{n-\tau}{N}+\phi_0\right)} + w(n) \quad (1)$$

where $w(n)$ denotes the AWGN in the time domain, τ and ϕ_0 are residual timing offset (RTO) and error phase, respectively, and ξ is the normalized CFO that can be divided in to an FFO part λ and IFO part ϵ as $\xi = \lambda + \epsilon$. This paper focuses on the IFO estimation, with FFO assumed to be compensated by earlier stages of synchronization, as investigated in detail elsewhere [4], [20].

The received preamble symbol at the FFT output is

$$Y(k) = e^{i\left(\phi_0 - 2\pi\frac{\tau k}{N}\right)} H(k-\epsilon)X(k-\epsilon) + W(n) \quad (2)$$

where $W(k)$ and $H(k)$ are the frequency domain representations of AWGN and CIR, respectively. As mentioned above, the IFO results in a cyclic shift in the frequency domain. By contrast, the RTO causes a linear phase rotation on samples in the frequency domain. Based on a differential demodulation of the FFT output, the IFO can be conventionally determined with robustness to frequency selective channel and RTO using the correlation function [2] expressed by

$$\hat{\epsilon} = \underset{\tilde{\epsilon}}{\operatorname{argmax}} \left| \sum_{k=1}^{N} Y^*(k-1)Y(k)X^*(k-\tilde{\epsilon})X(k-1-\tilde{\epsilon}) \right|$$

where $(.)^*$ denotes the complex conjugation, $\hat{\epsilon}$ and $\tilde{\epsilon}$ are the estimated and trial values of ϵ, respectively, $Y(k)$ and $X(k)$ denote the kth frequency symbol index of the received symbol and the known transmitted preamble, respectively, and the symbol size N is equal to the FFT size.

The IFO can be estimated with high precision using cross correlation in the frequency domain, however, implementing this involves a significant hardware overhead, with a multiplier for each element. Sign-bit cross correlation [21] is a widely adopted approach to reducing correlation complexity using only the most significant bit (MSB) of signed numbers for the computation. Complexity is reduced at the cost of performance degradation. Even using such methods, cross correlation remains computationally expensive, especially when dealing with a large FFT size. It should be noted here that several IFO estimation methods have been published which claim robustness to frequency selective channels and RTO. However, published FPGA implementations of these methods are lacking to date, possibly because the hardware costs are considerable—even for sign-bit cross correlation.

III. Proposed IFO Estimation Method

We save hardware resources and reduce power consumption by exploiting redundancy in the IFO computation to enable an efficient resource sharing folded architecture. Furthermore, adjusting the precision of individual correlation computations within this novel architecture allows a fine degree of control of the tradeoff between performance and power consumption. Thanks to the significant hardware cost reduction achieved, this IFO estimator can be feasibly implemented on a low-power, limited-resource FPGA, while simultaneously ensuring synchronization performance is maintained by adjusting the tradeoff between accuracy and hardware cost.

In [22], the authors investigated multiplierless correlation based on a conventional transpose form structure and similarly demonstrated a tradeoff between cost and accuracy for OFDM timing synchronization. The authors extended this in [20], to a method for OFDM timing synchronization plus FFO estimation. The synchronization method performed well with large CFO, however, IFO was neither estimated nor corrected. Note also that these previous papers performed computations in the time domain, before the FFT. However, in order to implement an OFDM system that can tolerate larger CFO, the IFO estimation needs to be implemented after the FFT, in the frequency domain. Thus, this paper develops an algorithm and architecture for efficient IFO estimation (in the frequency domain), achieving excellent performance at lower hardware cost and lower power than existing methods. It is applicable to 802.11, 802.22, and 802.16 (and potentially to other OFDM standards). Due to space limitations, we will present a single case study using the long preamble of IEEE 802.16-2009 [23].

A. Proposed Algorithm

First, we assume that the RF front end can provide the CFO stability in a range, R_{CFO}, from -14 to 18 subcarrier spacings, which is greatly relaxed and compared with the strict RF front-end constraints in 802.16 that would typically

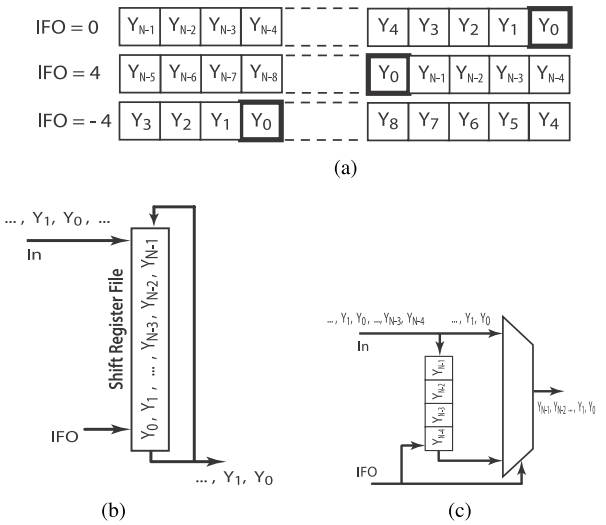

Fig. 2. IFO correction for example scenarios of no IFO ($\hat{\epsilon} = 0$), as well as $\hat{\epsilon} = -4$ and $\hat{\epsilon} = 4$ (top). The conventional and proposed approaches are shown (below) for $\hat{\epsilon} = -4$. (a) Rotated data symbols caused by the IFO. (b) Conventional approach. (c) Proposed method.

Fig. 3. Pilots in the long preamble of IEEE 802.16-2009.

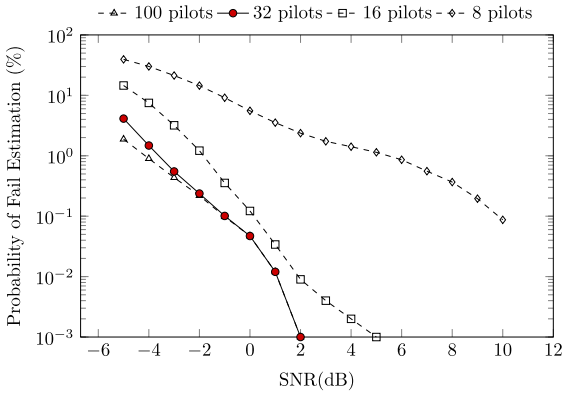

Fig. 4. Failure rate of the IFO estimation for different number of used pillots in the AWGN channel.

otherwise lead to increased RF hardware costs. Recall that in many practical implementations, the IFO estimation is avoided by restricting the CFO range to within that tolerated by the FFO estimator; typically -2 to 2 subcarrier spacings. Section I already explained why it is important to tolerate a larger CFO range, as well as presenting several solutions from other authors. We take advantage, here, of the fact that the FFO estimation and correction are performed prior to the IFO estimation, as shown in Fig. 1. This results in a reduced set of possible IFO values, as will be explained below. Metric M in (3) is widely employed for FFO estimation in recent standard systems. It is computed on the short preamble, consisting of periodic durations with length D [20]

$$M(n) = \sum_{m=0}^{D-1} \{s^*(n+m)s(d+m+D)\} = e^{j2\pi\xi\frac{D}{N}} \sum_{m=0}^{D-1} |x(n+m)|^2 \quad (3)$$

where $s(n)$ denotes the received signal with carrier frequency offset ξ with respect to $x(n)$. The two parts of ξ are estimated based on the angle of M ($\angle M$)

$$\hat{\xi} = \hat{\lambda} + \hat{\epsilon} = \frac{\angle M + 2\pi z}{2\pi D/N} \quad (4)$$

where z is an integer. The FFO is estimated as $\hat{\lambda} = N\angle M/2\pi D$. The remaining part after FFO corrected is the IFO, denoted as $\hat{\epsilon} = zN/D$. $\angle M$ is within the range $-\pi$ to π and for many standards (including IEEE 802.11, 802.16, and 802.22), $N/D = 4$. Hence, the FFO is estimated in the range -2 to 2 subcarrier spacings, and the IFO can be expressed as $\hat{\epsilon} = 4z \in R_{\text{CFO}}$. Hence, there are eight possible values for the IFO after correcting FFO, assuming the given CFO range. Possible IFO values are denoted as $S_{\text{IFO}} = \{-12, -8, -4, 0, 4, 8, 12, 16\}$.

Referring to the OFDM symbol sequence in Fig. 2(a), if $\hat{\epsilon} > 0$, data symbols are rotated left by a few places. Compensation is performed using small buffer (of size four symbols for the example of $\hat{\epsilon} = 4$). However, if $\hat{\epsilon} < 0$, the starting data symbol is circular shifted right, wrapping around to be among the final last symbols. Compensation now requires a buffer to store all intervening symbols. For the example where $\hat{\epsilon} = -4$, $N - 4$ symbols will require buffering. This long buffer is shown in Fig. 2(b), where $N = 256$ for IEEE 802.16. To significantly reduce the buffer requirements, we delay computation of the IFO by 12 symbols, so that the compensation shift is always positive: $S'_{\text{IFO}} = \{0{:}4{:}28\}$. This means that the received symbols will only ever need to be shifted right to compensate the IFO, with a maximum shift of 28 rather than 255, hence reducing buffer memory requirements, as shown in Fig. 2.

A second efficiency gain is achieved by creating a resource sharing folded architecture. The conventional IFO estimation is computed across all pilots in the preamble. This results in considerable hardware overhead, especially with a large number of pilots. IEEE 802.16-2009 includes $Np = 100$ pilots in the long preamble, as shown in Fig. 3, distributed with 50 pilots per side at even subcarrier spacings from 2 to 100 and from 156 to 254. The remaining subcarriers are null. A tradeoff is possible by reducing hardware cost using only a subset of pilots for the computation, at the expense of reduced noise robustness. Detailed simulations can help us explore the impact of using a reduced pilot set on performance. Fig. 4 plots the common probability of fail estimation (POFE, see Section IV) measure against signal-to-noise ratio (SNR) when using different numbers of pilots, revealing that there is very little performance difference between using the full 100 pilots and a subset of 32 pilots above about -2 dB SNR.

We, therefore, propose making use of only a subset of pilots for offset estimation—enough to maintain accuracy, but by carefully selecting the chosen pilots in the frequency domain as multiples of 4, we are able to specify a fourfold resource sharing design. Hence, the IFO estimation can be

Algorithm 1 IFO Correlation Computation Algorithm

$Init:\ k = 0;\ \ n = 0$
repeat
  $Every\ 4\ cycles$
  if $4k \in Si$ **then**
    $Calculate\ P_{4k}$
    for $Each\ \tilde{\epsilon} \in S1'_{IFO} = \{0, 4, 8, 12\}$ **do**
      $V_{\tilde{\epsilon}}\ += P_{4k} A_{\tilde{\epsilon}}(n)$
    end for
    for $Each\ \tilde{\epsilon} \in S2'_{IFO} = \{16, 20, 24, 28\}$ **do**
      $V_{\tilde{\epsilon}}\ += P_{4k} A_{\tilde{\epsilon}}(n)$
    end for
    $n\ += 1$
  end if
  $k\ += 1$
until $k > Np/2$

TABLE I
COMPUTATION TIME FOR DIFFERENT APPROACHES

Method	Number of MACs	Number of Cycles
Dedicated Processor	1	$N_I \times N_P$
Accelerated Hardware	N_I	N_P
Proposed Algorithm	N_I/D	$D \times N_P/2$

expressed as

$$\hat{\epsilon} = \underset{\tilde{\epsilon} \in S'_{\text{IFO}}}{\text{argmax}} |V_{\tilde{\epsilon}}|$$
$$V_{\tilde{\epsilon}} = \sum_{k=1}^{Np/4} P(4k) A1_{\tilde{\epsilon}}(k) + P(L+4k) A2_{\tilde{\epsilon}}(k) \qquad (5)$$

where $V_{\tilde{\epsilon}}$ is the cross correlation between received pilots and prerotated known pilots and $P(4k) = Y^*(4k-2)Y(4k)$ denotes the correlation of two consecutive received pilots. Since the pilots of the long preamble are distributed on two sides of the OFDM symbol in the frequency domain, at even subcarrier spacings, L denotes the index of the first pilots in the second half. $A1_{\tilde{\epsilon}}(k) = X^*(4k-2-\tilde{\epsilon})X(4k-\tilde{\epsilon})$ and $A2_{\tilde{\epsilon}}(k) = X^*(L+4k-2-\tilde{\epsilon})X(L+4k-\tilde{\epsilon})$ denote the correlation of two consecutive prerotated known pilots of the first side and second side, respectively, of the preamble symbol corresponding to one IFO value ($\tilde{\epsilon}$). Let $A_{\tilde{\epsilon}}$ be a known coefficient set as $A_{\tilde{\epsilon}} = \{A1_{\tilde{\epsilon}}, A2_{\tilde{\epsilon}}\}$. Let Si denote the set of used pilot indices for the proposed method (i.e., $Si = \{(4{:}4{:}(Np/2)), (L{:}4{:}N)\}$), we present Algorithm 1 to concurrently compute the cross correlations, where the received pilots whose indices are in Si are employed to compute P_{4k}.

Assuming that the data symbols are received from the FFT in sequence, the duration between the employed pilots is four clock cycles. For each value of $\tilde{\epsilon}$ in $S1'_{\text{IFO}}$ and $S2'_{\text{IFO}}$, the corresponding $V_{\tilde{\epsilon}}$ can be computed separately at every clock cycle using two multiply accumulate (MAC) blocks with the corresponding known coefficients. Hence, eight cross correlations can be computed in the duration between two employed pilots. Table I compares this in terms of time-area complexity to the conventional method implemented in a dedicated processor or using accelerated hardware. N_I is the number of possible IFO values and D denotes the duration between two employed pilots ($N_I = 8$ and $D = 4$ for the 802.16 case study). The figures are specified using single-cycle MAC operations and clearly show the two common tradeoff points of maximum sequential operation and maximum parallelism, with the proposed method lying between the two: however, it uses D times less hardware than the latter, but completes in $D/2$ more cycles. Similarly, it uses N_I/D more MACs than the former but completes $D/2N_I$ cycles quicker. Note that the factor of two improvement over a straight-line interpolation between the extreme tradeoffs. Moreover, the IFO estimation is computed in parallel with receiving the OFDM packet. It takes N (i.e., 256 for 802.16) cycles to receive the OFDM packet stream, whereas the IFO estimation takes $D \times (\text{NP}/2)$ (i.e., 200 for 802.16) cycles. The IFO estimation can be done immediately after the preamable packet is received and does not add latency to the OFDM packet stream. Section III-B will determine a method able to achieve this resource sharing for computing $V_{\tilde{\epsilon}}$ as well as in storing the pilots.

In addition, we will use multiplierless correlation with adjustable wordlength to further expand the performance/complexity tradeoff. Although sign-bit cross correlation is often used in conventional implementations to reduce the computational complexity [4], [21], it leads to the reduced precision and hence the reduced estimation performance, especially in frequency selective channels. By contrast, the multiplierless correlation allows a fine degree of control to enhance accuracy over a sign-bit approach, while still allowing a reduction in complexity over a full calculation. In [22], the authors demonstrated a tradeoff between cost and accuracy for multiplierless correlation in the case of OFDM timing synchronization, based on a conventional transpose form structure. We now apply a similar approach to this new IFO frequency estimation architecture, exploring the performance, area, and power consumption effects of changing the wordlength used to represent P_{4k}, in Section IV-B.

B. Proposed Architecture

Fig. 5(a) shows a conventional architecture for computing the IFO estimation, while Fig. 5(b) shows the proposed resource sharing architecture. Cross correlators compute the values of $V_{\tilde{\epsilon}}$ while the ArgMax module finds the maximum of the $V_{\tilde{\epsilon}}$ values in order to identify the corresponding IFO estimate. The conventional approach in Fig. 5(a) employs separate cross correlators to calculate each $V_{\tilde{\epsilon}}$ result. This requires significant resources, including a large number of multipliers which may be not available on a limited-resource FPGA. The novel architecture comprises two parts.

1) Sharing Stored Pilot Memory: There are eight sets of $A_{\tilde{\epsilon}}$ corresponding to eight possible IFOs. These sets of $A_{\tilde{\epsilon}}$ are precomputed and stored separately in a dual-port register file. Thanks to the spreading of the computed pilots, the $A_{\tilde{\epsilon}}$ sets have many identical pilots. This naturally allows sharing between the prerotated pilot sets. Therefore, the *PilReg* block requires only 64 shared memory locations for the eight sets instead of 400; an 84% reduction. Fig. 6 shows the $A_{\tilde{\epsilon}}$ sets and circuitry for combining all $A_{\tilde{\epsilon}}$ sets in the *PilReg* block.

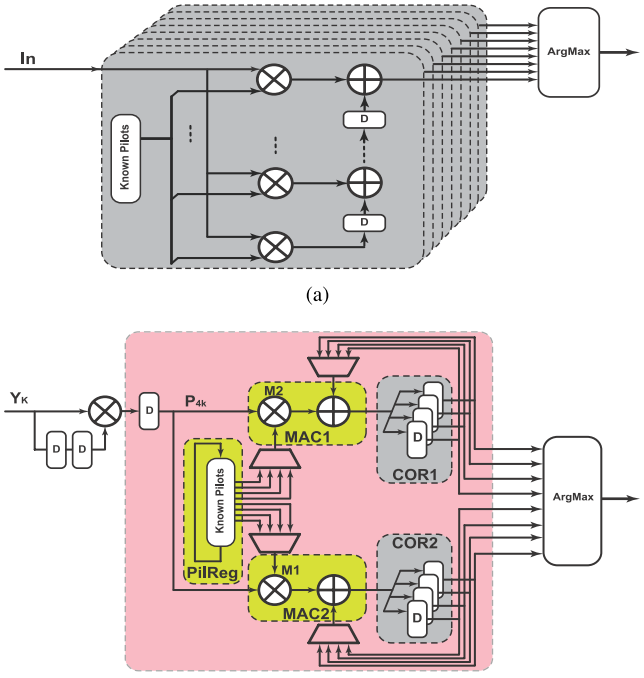

Fig. 5. Architectures of the IFO estimators. (a) Conventional approach. (b) Proposed method.

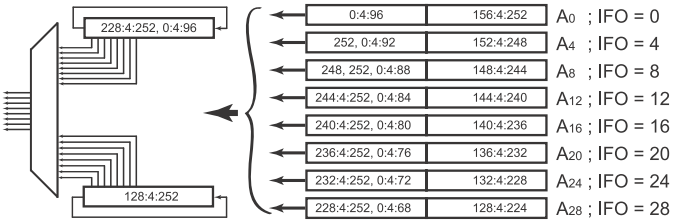

Fig. 6. Circuit of known pilots shift register.

2) Sharing Correlation Resources: The proposed method divides IFO estimation into multiple repeated computations with resource sharing based upon the four-sample timing between selected spread pilots. The pilots that are used to compute the correlation arrive every four cycles so there are three spare cycles between two consecutive computed pilots, allowing one MAC block to be scheduled to sequentially compute four separate correlations. The MAC blocks are thus shared among four sequential $V_{\tilde{\epsilon}}$ computations over four successive clock periods. Fig. 7 demonstrates how this is achieved. P_k is received every clock cycle. P_{4k} is the subsampling of P_k, taking a subset of the MSBs from P_k every four cycles to perform the cross correlation. Two MAC blocks, MAC1 and MAC2, are used to compute the values of eight cross correlations in parallel. Each multiplier performs multiplications sequentially between P_{4k} and the corresponding transmitted pilots in four sets of $A_{\tilde{\epsilon}}$. The products are accumulated to the values of $V_{\tilde{\epsilon}}$. The values of the correlation operations are stored separately in the corresponding buffers in the COR1, COR2 blocks. When the correlation computation is complete, the maximum operation, $\arg\max|V|$, is performed on eight $V_{\tilde{\epsilon}}$ values to estimate the IFO.

To obtain further resource savings, MAC1 and MAC2 are implemented using multiplierless techniques. $V_{\tilde{\epsilon}}$ in (5) is

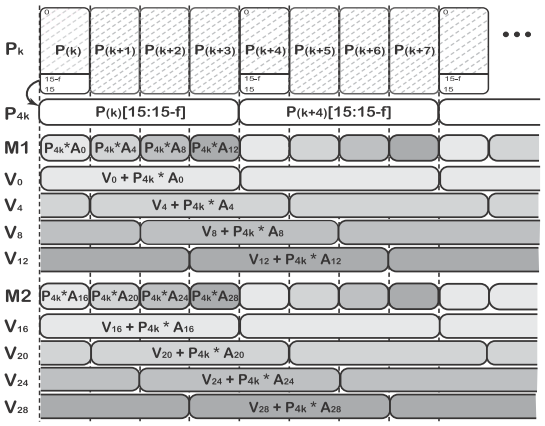

Fig. 7. Resource sharing approach for computing $V_{\tilde{\epsilon}}$.

mathematically manipulated into what is effectively a MAC form. When one received sample is received, $V_{\tilde{\epsilon}}$ can be expressed as an accumulation

$$\begin{aligned} V_{\tilde{\epsilon}} &= A_{\tilde{\epsilon}} P_{4k} + V_{\tilde{\epsilon}} \\ &= (\Re\{U\} - i\Im\{U\})(\Re\{P_{4k}\} + i\Im\{P_{4k}\}) \\ &\quad + (\Re\{V_{\tilde{\epsilon}}\} + i\Im\{V_{\tilde{\epsilon}}\}) \end{aligned} \tag{6}$$

where $\Re\{.\}$ and $\Im\{.\}$ denote the real and imaginary parts, respectively. $A_{\tilde{\epsilon}}$ is normalized to U whose real and imaginary parts have values in $\{-1, 0, 1\}$, and the wordlength of P_{4k} and $V_{\tilde{\epsilon}}$ in fixed point format can be adjusted to tradeoff between estimation accuracy and hardware resource. The real and imaginary parts of $V_{\tilde{\epsilon}}$ are computed as follows:

$$\begin{aligned} \Re\{V_{\tilde{\epsilon}}\} &= \Re\{U\}\Re\{P_{4k}\} + \Im\{U\}\Im\{P_{4k}\} + \Re\{V_{\tilde{\epsilon}}\} \\ \Im\{V_{\tilde{\epsilon}}\} &= \Re\{U\}\Im\{P_{4k}\} - \Im\{U\}\Re\{P_{4k}\} + \Im\{V_{\tilde{\epsilon}}\}. \end{aligned} \tag{7}$$

In Section IV, we will demonstrate through simulation on different channels that the algorithm and architecture optimizations mentioned above retain competitive estimation accuracy compared with the conventional approaches, while offering significant reductions in hardware resource. This makes it possible to implement a high-performance OFDM receiver on a low-power FPGA that has a limited number of digital signal processing (DSP) blocks.

IV. Simulation

Many variants of the proposed method were simulated in MATLAB using different channel models with the IEEE 802.16-2009 downlink preamble parameter set. Performance was compared with the theoretical performance of some state of the art methods in terms of POFE with respect to channel SNR. The POFE [2], [3], [5] measures the number of failed estimations divided by the total number of IFO estimations. Overall, 100 000 IFO estimations were simulated in AWGN and Stanford University Interim [24] frequency selective channels. The IFO estimation is performed with nonideal FFO compensation, and the FFO is determined and compensated using the method of [4]. The simulation also verifies the performance of the proposed method under the effect of RTO caused by imperfect STO estimation (assuming that the STO estimation is still within the CP and does not

cause ISI). In addition, a randomly generated amount of STO is added in the range from 0 to $N_{\mathrm{CP}} - L_h - 1$ (i.e., the RTO is in range from 0 to $N_{\mathrm{CP}} - L_h - 1$).

We investigated the performance degradation compared with theory as a result of reducing the number of pilots in Section III-A and now investigate the effect of wordlength optimization. In both cases, comparisons are made with established methods in the literature that can be simulated but are otherwise infeasible for hardware implementation, namely, the conventional method in [2] (PCH) that is applied to one training block with 100 pilots, plus two state of the art methods: the first is metric SY from [3] as defined by

$$\mu_{\mathrm{SY}}(\tilde{\epsilon}) = \Re\left\{\sum_{k=1}^{N/2} Y^*_{(2k-2)} Y_{(2k)} X^*_{(2k-\tilde{\epsilon})} X_{(2k-2-\tilde{\epsilon})}\right\}$$

where $\hat{\epsilon} = \mathrm{argmax}_{\tilde{\epsilon}\in S_{\mathrm{IFO}}}\{\mu(\tilde{\epsilon})\}$. Second, the metric MM from [5]

$$\mu_{\mathrm{MM}}(\tilde{\epsilon}) = \Re\left\{e^{i\frac{\pi}{4}} \sum_{k=1}^{N/2} Y^*_{(2k-2)} Y_{(2k)} X^*_{(2k-\tilde{\epsilon})} X_{(2k-2-\tilde{\epsilon})}\right\}$$

where $\Re\{.\}$ denotes the real part. The very large hardware requirement of these metrics does not lend them to feasible implementation on a low cost, low power FPGA (unlike our proposed method). As a result, the authors are unaware of any published circuits for these methods.

A. Performance Comparison

The performance of the proposed method, denoted by *Prop*, is evaluated in comparison with the theoretical performance of state of the art methods by Park *et al.* [2], Shim and You [3] and Morelli and Moretti [5], denoted by PCH, SY, and MM, respectively, in Section IV. The theoretical performance is computed with full precision using a full pilot set (100 pilots). However, it must be noted that implementing this directly in hardware would be prohibitive due to the large number of high precision multiplication operations required. Instead, hardware implementation would conventionally use sign-bit correlation instead of full precision, as mentioned previously. Thus, the full multiplication results shown, here, are undoubtedly better than those achievable in practice, and should be considered as upper theoretical bounds. For more realistic comparisons, we provide results from sign-bit correlation versions of each, denoted by *PCH_sb*, *SY_sb*, and *MM_sb*, respectively. The method in [4] employs sign-bit correlation to implement a joint STO and IFO estimator by performing a long cross correlation in the time domain for STO estimation, plus an exhaustive search across a large potential CFO range. This results in larger hardware usage compared with the frequency domain methods of [2], [3], and [5], and is not, therefore, not included in the comparison.

The proposed method uses 50 spread pilots with indices that are multiples of 4. For the sake of fair comparison, we include an additional implementation of PCH which uses 50 pilots like the proposed method, spaced continuously. This is denoted by *PCH_50*. Fig. 8 plots performance results for all methods in an AWGN channel with RTO and reveals that the proposed

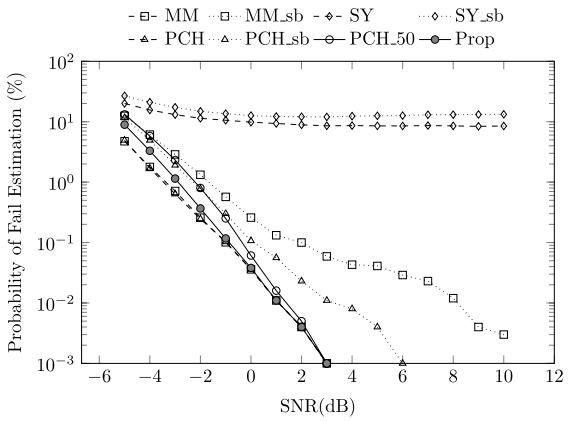

Fig. 8. Failure rate of the IFO estimation methods in AWGN channel with RTO.

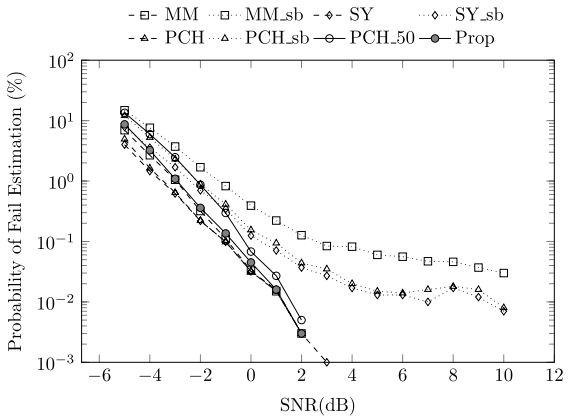

Fig. 9. Failure rate of the IFO estimation methods in SUI1 channel.

method generally performs well, especially at higher SNRs. A more realistic SUI1 channel model is used for Fig. 9, again reveals good performance, particularly at positive SNRs (performance is similarly good in SUI2 which is not plotted for space reasons).

Under these experimental conditions, PCH and SY achieve equivalent performance in AWGN without RTO and in the SUI1 channel. In general, SY appears to be very sensitive to large RTO, while MM and PCH exhibit better robustness to RTO. The accuracy of MM is slightly lower than that of the PCH at SNRs <0 dB, while performance is very similar at larger SNRs. Also note the performance of the conventional approach implementations, *PCH_sb*, *SY_sb*, and *MM_sb*, which degrade significantly with SNR, especially in the SUI channels. Recall that these represent the typical implementation of estimators, with expensive multipliers replaced by sign-bit correlation.

Apart from at very low SNRs, the proposed method, *Prop*, achieves almost identical performance to the simulated upper bound PCH, even in the presence of RTO. It should be noted that *Prop* achieves this while allowing the use of resource sharing through sparse pilot computation, achieving a significant hardware saving. The results also show that *Prop*, with its spread pilots, is more accurate than *PCH_50* using the same number of pilots spread continuously.

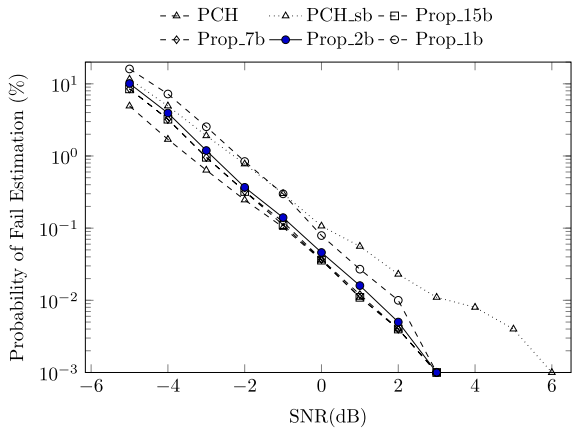

Fig. 10. Failure rate for different wordlengths in the AWGN channel with RTO.

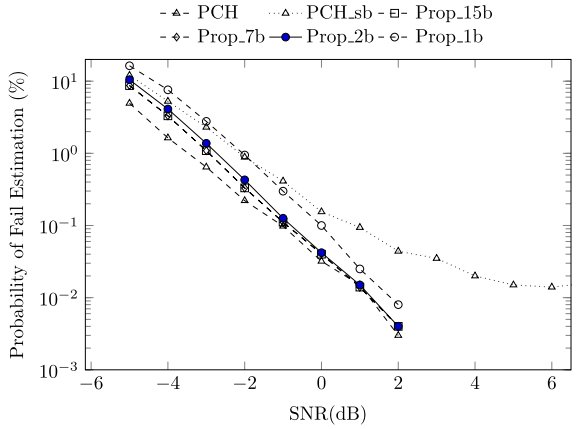

Fig. 11. Failure rate for different wordlengths in an SUI1 channel.

B. Wordlength Optimization

Since sign-bit correlation degrades the IFO estimation performance, especially in frequency selective channels, we instead explore the use of different word length multiplierless correlation to tradeoff between hardware resource and estimation performance. We again compared against the theoretical bound, PCH, and against a conventional sign-bit implementation, *PCH_sb*. Wordlength is specified using the notation Q1.f, meaning a single integer bit and f fractional bits. Evaluations are performed for $f = \{1, 2, 7, \text{and } 15\}$ bits with results plotted using the labels *Prop_1b*, *Prop_2b*, *Prop_7b*, and *Prop_15b*, respectively. Fig. 10 plots the performance in AWGN with RTO for all tested wordlengths. The proposed method performs comparably with PCH (and is better than *PCH_sb* at SNRs exceeding $\sim$2 dB). Fig. 11 show the results when using the more realistic SUI1 channel models. Performance in SUI2 is similar, but not reproduced for space reasons.

It can be seen that each of the tested wordlengths achieves much better performance and exhibits greater robustness to frequency selective channels than the sign-bit conventional realization, *PCH_sb*. In addition, these realizations of the proposed method do not suffer as much degradation in the presence of RTO. Moreover, it is possible to improve low SNR performance by adopting a longer wordlength with the proposed method, at a cost of increased hardware complexity. Increasing wordlength yields decreasing returns: moving from 1 to 2 bits yields a significant gain whereas increasing from 2 to 7 or from 7 to 15 bits has less impact. In general, *Prop_2b* achieves an estimation accuracy close to that of the theoretical performance bound, PCH, at intermediate and higher SNRs, even though it involves computation with fewer bits, and can hence be implemented more efficiently.

V. FPGA Implementation

The analysis in Section IV suggests that the proposed method offers comparable estimation performance to existing methods in the literature. As a result of the simplifications inherent in the proposed approach, this should be achievable at a reduced hardware cost. This section now quantifies this hardware cost for an FPGA-based implementation. It is important to note that these hardware savings are accessible for a number of target implementation devices, although we are interested primarily in the FPGA implementation as part of our work on leveraging FPGA reconfigurability for cognitive radios.

A. Conventional Approach

To obtain the theoretical performance previously discussed in Section IV and denoted as PCH, the computation of the estimation metric in [2], using 100 pilots, would require about 200 complex multipliers, resulting in the use of over 600 DSP blocks. This exceeds the available resources on small devices, and would leave insufficient resources for other tasks on larger devices. As the number of multiplications required for a full implementation is prohibitive, the conventional solution, as we have discussed, is to adopt sign-bit correlation [4]. Thus, the conventional implementation uses all 100 pilots in the long preamble and sign-bit correlation, with multiply_adds eliminated at taps where the pilots of the long preamble are zero or not used. This implementation mirrors the *PCH_sb* in Section IV, and allows us to quantify the benefits of our proposed approach against a known reference benchmark.

B. Proposed Approach

The proposed architecture implemented with several different wordlengths of P_{4k} and $V_{\tilde{\epsilon}}$ in (6) are compared, to allow us to explore the hardware costs associated with the respective implementations. Four-fixed point formats for P_{4k}, are investigated: 1) Q1.1; 2) Q1.2; 3) Q1.7; and 4) Q1.15. $V_{\tilde{\epsilon}}$ is represented correspondingly in formats of Q7.1, Q7.2, Q7.7, and Q7.15 to avoid overflow.

In order to obtain a comprehensive optimized implementation, each of these circuits are implemented using two different structures. The first uses only logic elements (LEs) for computation, while the second uses Xilinx DSP48A1 [25] primitives. Considering (7), $\Re\{V_{\tilde{\epsilon}}\}$ and $\Im\{V_{\tilde{\epsilon}}\}$ can be computed effectively using two DSP blocks as 3-input adders, instead of four blocks as would be usual. Fig. 12 shows how this is done for $\Re\{V_{\tilde{\epsilon}}\}$ and $\Im\{V_{\tilde{\epsilon}}\}$ is similar. Note that the solution presented in Fig. 12 is optimized for QPSK modulated pilots (since their amplitudes are identical) as specified in IEEE 802.16,

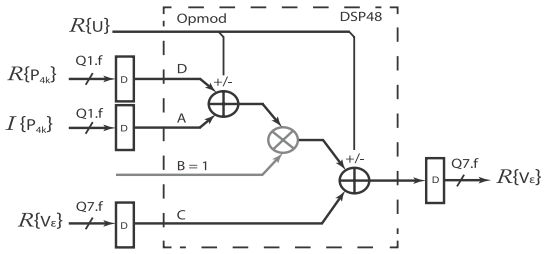

Fig. 12. DSP block-based three-input adder for correlation.

TABLE II
RESOURCE UTILIZATION AND DYNAMIC POWER CONSUMPTION

IFO est. Cir.	FFs	LUTs	DSPs	Frq (MHz)	D. Power
conv_100p_sb	3270 (3%)	1837 (3%)	3	142	42 mW
Prop_1b_LE	328 (1%)	370 (1%)	3	136	9 mW
Prop_2b_LE	350 (1%)	390 (1%)	3	136	10 mW
Prop_7b_LE	460 (1%)	471 (1%)	3	136	12 mW
Prop_15b_LE	735 (1%)	696 (1%)	3	134	17 mW
Prop_1b_DSP	328 (1%)	306 (1%)	7	78	11 mW
Prop_2b_DSP	350 (1%)	319 (1%)	7	78	12 mW
Prop_7b_DSP	460 (1%)	379 (1%)	7	77	14 mW
Prop_15b_DSP	735 (1%)	591 (1%)	7	77	18 mW

as well as in most OFDM-based standards. The normalization performed in (7) allows the correlation to be reduced to two DSP blocks operating as three-input adders (instead of four DSP blocks with multipliers as would be usual). These methods correspond to *Prop_1b*, *Prop_2b*, *Prop_7b*, and *Prop_15b* that were investigated for estimation accuracy in Section IV.

C. Implementation Results

The circuits were synthesized and fully implemented using Xilinx ISE 13.2, targeting the low-power Xilinx Spartan-6 XC6SLX75T FPGA. The results are reported in terms of the number of flipflops (FFs), lookup tables (LUTs), and DSP blocks, along with dynamic power consumption, as summarized in Table II.

conv_100p_sb refers to the conventional approach, implemented using sign-bit correlation over 100 pilots. *Prop_fb_LE*, *Prop_fb_DSP*, in which $f = 1$, 2, 7, and 15 (corresponding to received sample format Q1.f), denote the circuits of corresponding wordlengths implemented using LEs and DSP blocks, respectively. Table II reveals that the proposed implementation achieves a significant reduction in resource usage and dynamic power consumption.

The hardware resources used by *Prop_fb_LE* and *Prop_fb_DSP* increase gradually, in terms of FFs and LUTs as the wordlength increases. The number of FFs used in *Prop_fb_DSP* and *Prop_fb_LE* is equal, while *Prop_fb_DSP* uses fewer LUTs, since the DSP blocks are used for the three-input additions. The *Prop_fb_LE* implementations use three DSP blocks to compute P_{4k}, while *Prop_fb_DSP* require an additional four DSP blocks to perform the correlation. Both *Prop_fb_LE* and *Prop_fb_DSP* consume far fewer LUTs and FFs than the conventional *conv_100p_sb* sign-bit implementation. For *Prop_2b_LE*, the number of FFs and LUTs is reduced by 90% and 79%, respectively, compared with the conventional *conv_100_sb* approach.

The maximum circuit frequencies, reported after place and route, are 142, 136, and 78 MHz for the conventional sign bit-based and proposed LE- and the proposed DSP-based circuits, respectively, comfortably exceed the timing requirements for most OFDM-based systems, particularly for 802.16 whose sampling frequency is <25 MHz.

A postplace-and-route simulation was used to estimate the power consumption of the system at a clock rate of 50 MHz using the Xilinx XPower tool—also shown in Table II—revealing that *Prop_fb_LE* implementations consume less power than the equivalent *Prop_fb_DSP* implementations. All implementations of the proposed method consume significantly less power than the conventional implementation. For example, *Prop_2b_LE* consumes just 22% of the power required for *conv_100p_sb*.

Section IV established that *Prop_2b_LE* easily outperforms the conventional approach in terms of estimation accuracy. Now, we can see that it does so with a significant hardware resource saving, as well as significantly reduced power consumption. In fact, the estimation performance of *Prop_2b_LE*, in AWGN and SUI channels (except at very low SNR), is close to the theoretical bound of PCH, which would demand a significant amount of the FPGAs resources if it were implemented conventionally. Meanwhile, *Prop_2b_LE* is extremely efficient, consuming <1% of the resources available on a low-power Spartan-6 XC6SLX75 FPGA.

VI. CONCLUSION

This paper has investigated IFO estimation in OFDM-based systems such as IEEE 802.16. A technique is proposed for efficient implementation of IFO estimation, which aims in particular for a low-power and low-resource utilization. Since IFO estimation can contribute significantly to the complexity of a robust OFDM synchronizer design, this paper is important for multistandard radios, as well as applications such as mobile radios where significant frequency variation is expected. Robust IFO estimation can also allow for a relaxation of analog RF constraints at the radio front end, potentially leading to a reduced cost implementation.

A novel timing algorithm was proposed that is allied with pilot subsampling to enable a fourfold resource sharing architecture to reduce both hardware complexity as well as power consumption. Meanwhile, the multiplierless correlation with optimized wordlengths is explored to improve estimation accuracy over a conventional implementation using sign-bit correlation. The proposed algorithm and architecture have been evaluated theoretically, in simulation (to determine system-level IFO estimation performance), through synthesis and postplace-and-route implementation (to determine detailed resource utilization and power consumption figures). The proposed method estimation performance matches current state-of-the-art methods that employ multiplier-based correlation, yet achieves a significant reduction in hardware requirements. Dynamic power consumption of the proposed method is reduced by 78% over even an efficient sign-bit

version of the conventional approach, yet offers much better estimation performance in both AWGN and frequency selective channels.

Beyond IEEE 802.16-2009, the folded resource sharing method presented in this paper, which leverages subsampled OFDM pilots and adjustable word length multiplierles correlation, is compatible with other OFDM standards including IEEE 802.11 and IEEE 802.22.

REFERENCES

[1] M. Morelli, A. N. D'Andrea, and U. Mengali, "Frequency ambiguity resolution in OFDM systems," *IEEE Commun. Lett.*, vol. 4, no. 4, pp. 134–136, Apr. 2000.

[2] M. Park, N. Cho, J. Cho, and D. Hong, "Robust integer frequency offset estimator with ambiguity of symbol timing offset for OFDM systems," in *Proc. IEEE 56th Veh. Technol. Conf. (VTC)*, 2002, pp. 2116–2120.

[3] E.-S. Shim, Y.-H. You, S.-J. Kang, Y.-S. Kim, D.-H. Kim, and J.-W. Cho, "OFDM integer frequency offset estimator in rapidly time-varying channels," in *Proc. Asia-Pacific Conf. Commun.*, 2006, pp. 1–4.

[4] T.-H. Kim and I.-C. Park, "Low-power and high-accurate synchronization for IEEE 802.16d systems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 12, pp. 1620–1630, Dec. 2008.

[5] M. Morelli and M. Moretti, "Integer frequency offset recovery in OFDM transmissions over selective channels," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5220–5226, Dec. 2008.

[6] D. Toumpakaris, J. Lee, and H.-L. Lou, "Estimation of integer carrier frequency offset in OFDM systems based on the maximum likelihood principle," *IEEE Trans. Broadcast.*, vol. 55, no. 1, pp. 95–108, Mar. 2009.

[7] C. Shahriar *et al.*, "PHY-layer resiliency in OFDM communications: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 292–314, 2015.

[8] H. Nogami and T. Nagashima, "A frequency and timing period acquisition technique for OFDM systems," in *Proc. 6th IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, vol. 3. Sep. 1995, pp. 1010–1015.

[9] D. Li, Y. Li, H. Zhang, L. J. Cimini, and Y. Fang, "Integer frequency offset estimation for OFDM systems with residual timing offset over frequency selective fading channels," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2848–2853, Jul. 2012.

[10] M. Cummings and S. Haruyama, "FPGA in the software radio," *IEEE Commun. Mag.*, vol. 37, no. 2, pp. 108–112, Feb. 1999.

[11] J. D. Guffey, A. M. Wyglinski, and G. J. Minden, "Agile radio implementation of OFDM physical layer for dynamic spectrum access research," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2007, pp. 4051–4055.

[12] A. Troya, K. Maharatna, M. Krstic, E. Grass, U. Jagdhold, and R. Kraemer, "Low-power VLSI implementation of the inner receiver for OFDM-based WLAN systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 2, pp. 672–686, Mar. 2008.

[13] S. J. Hwang, Y. Han, S. W. Kim, J. Park, and B. G. Min, "Resource efficient implementation of low power MB-OFDM PHY baseband modem with highly parallel architecture," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 7, pp. 1248–1261, Jul. 2012.

[14] W. Fan and C.-S. Choy, "Robust, low-complexity, and energy efficient downlink baseband receiver design for MB-OFDM UWB system," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 2, pp. 399–408, Feb. 2012.

[15] T. H. Pham, S. A. Fahmy, and I. V. McLoughlin, "Efficient multi-standard cognitive radios on FPGAs," in *Proc. 24th Int. Conf. Field Program. Logic Appl. (FPL)*, Sep. 2014, pp. 1–2.

[16] Y. Zhang, J. H. Mueller, B. Mohr, and S. Heinen, "A low-power low-complexity multi-standard digital receiver for joint clock recovery and carrier frequency offset calibration," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 12, pp. 3478–3486, Dec. 2014.

[17] M. Morelli, L. Marchetti, and M. Moretti, "Maximum likelihood frequency estimation and preamble identification in OFDMA-based WiMAX systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1582–1592, Mar. 2014.

[18] M. Morelli, L. Marchetti, and M. Moretti, "Integer frequency offset estimation and preamble identification in WiMAX systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 5610–5615.

[19] Z. Zhang, L. Ge, F. Tian, F. Zeng, and G. Xuan, "Carrier frequency offset estimation of OFDM systems based on complementary sequence," in *Proc. 7th IEEE Int. Congr. Image Signal Process. (CISP)*, Oct. 2014, pp. 1012–1016.

[20] T. H. Pham, I. V. McLoughlin, and S. A. Fahmy, "Robust and efficient OFDM synchronization for FPGA-based radios," *Circuits, Syst., Signal Process.*, vol. 33, no. 8, pp. 2475–2493, Aug. 2014.

[21] L. Schwoerer, "VLSI suitable synchronization algorithms and architecture for IEEE 802.11a physical layer," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, vol. 5. May 2002, pp. 721–724.

[22] T. H. Pham, S. A. Fahmy, and I. V. McLoughlin, "Low-power correlation for IEEE 802.16 OFDM synchronization on FPGA," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 8, pp. 1549–1553, Aug. 2013.

[23] *IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE Standard 802.16-2009.

[24] V. Erceg *et al.*, *Channel Models for Fixed Wireless Applications*, document IEEE 802.16a-03/01, Jul. 2003.

[25] *UG389: Spartan-6 FPGA DSP48A1 Slice*, Xilinx Inc., San Jose, CA, USA, Aug. 2009.

Thinh Hung Pham (S'13) received the B.S. degree in electrical and electronic engineering from the Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam, in 2007, and the M.Sc. degree in embedded systems engineering from the University of Leeds, Leeds, U.K., in 2010. He is currently pursuing the Ph.D. degree in the joint Nanyang Technological University—Technische Universität München program in Singapore.

He is a Research Associate with the TUM CREATE Centre for Electromobility, Singapore.

Suhaib A. Fahmy (M'01–SM'13) received the M.Eng. degree in information systems engineering and the Ph.D. degree in electrical and electronic engineering from Imperial College London, London, U.K., in 2003 and 2007, respectively.

He was a Research Fellow with Trinity College Dublin, Dublin, Ireland, and a Visiting Research Engineer with Xilinx Research Labs, Dublin, from 2007 to 2009. Since 2009, he has been an Assistant Professor with the School of Computer Engineering, Nanyang Technological University, Singapore. His current research interests include reconfigurable computing, high-level system design, and computational acceleration of complex algorithms.

Dr. Fahmy was a recipient of the best paper award at the IEEE Conference on Field Programmable Technology in 2012, and the IBM Faculty Award in 2013. He is currently a Senior Member of the Association for Computing Machinery.

Ian Vince McLoughlin (SM'04) received the Ph.D. degree from the University of Birmingham, Birmingham, U.K., in 1997.

He split his career between the electronics research and development industry and academia, based in five countries on three continents. He was a Faulty Member with Nanyang Technological University, Singapore, for 10 years, and the U.K. and New Zealand industry for 10 years. He was a Professor with the University of Science and Technology of China, Hefei, China. He became a Chartered Engineer in U.K. in 1998. He was a D'Ingenieur European in 2008. He is currently a Professor with the School of Computing, University of Kent, Kent, U.K. He has authored over 200 papers, holds 13 patents, and books with Cambridge University Press and McGraw-Hill.

Prof. McLoughlin was a fellow of the Institution of Engineering and Technology in 2013.