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 sub-carrier indices in the frequency domain. 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. IFO is therefore often estimated and removed at baseband, allowing implementations to benefit from relaxed RF front-end specifications and be tolerant to both Doppler shift and multi-standard channel selection. This paper proposes a novel architecture for 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 four-fold resource sharing to reduce computational cost, while multiplierless computation yields further power reduction. Performance exceeds that of conventional techniques, while being much more efficient. When implemented on FPGA for IEEE 802.16-2009, 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*—OFDM, digital signal processing, field programmable gate arrays.**

I. Introduction and Related Work

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 receiver synchronisation errors such as carrier frequency offset (CFO) which causes inter-carrier 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 (FFO) and integer (IFO) offsets for estimation [2], [3], [4], [5], [6]. 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 synchronisation, the CFO is split into coarse- and fine frequency offset estimation processes, which determine IFO and FFO respectively [7].

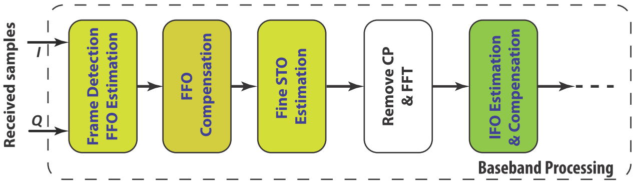


Fig. 1. Baseband processing block diagram.

Fig. 1 illustrates 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 (MLE) has been used to estimate IFO [5], [6], based upon the observation of pilots from two consecutive OFDM symbols. A reduced complexity alternative is to only 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 to processor-based software radio [10]. Several authors have presented hardware implementations of OFDM-based systems in the literature [11], [12], [13], [14], but these implementations do not include IFO estimation (only FFO). Without IFO correction, 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], [18], [19], this paper is concerned with IFO estimation, either to support large-CFO application scenarios, or to enable cost saving through relaxing strict front-end design criteria. While 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

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

I. V. McLoughlin is with the School Computing, University of Kent, UK.

novel method achieves very accurate IFO estimation on low-power FPGAs with minimal hardware cost, at significantly lower power than 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 IFO estimation while Section III introduced the proposed method in comparison to previous work. Section IV details simulations used to evaluate the method against conventional approaches. Section V presents the implemention on FPGA and discusses resource requirements and power consumption issues. Section VI concludes the paper.

II. Integer Frequency Offset Estimation

We consider a signal *x(n)* of an OFDM system with inverse fast Fourier transform (IFFT) length *N*. Assuming 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

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

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

The received preamble symbol at FFT output is

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

where $W(k)$ and $H(k)$ are the frequency domain representations of AWGN and CIR, respectively. As mentioned previously, IFO results in a cyclic shift in the frequency domain. By contrast, 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:

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

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

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 by 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 trade-off 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 synchronisation performance is maintained by adjusting the trade off between accuracy and hardware cost.

In [22], the authors investigated multiplierless correlation based on a conventional transpose form structure and similarly demonstrated a trade-off between cost and accuracy for OFDM timing synchronisation. The authors extended this in [20], to a method for OFDM timing synchronisation plus FFO estimation. The synchronisation 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, 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

Firstly, we assume that the RF front end can provide CFO stability in a range, R_{CFO}, from -14 to $+18$ sub-carrier spacings, which is greatly relaxed compared to the strict RF front end constraints in 802.16 that would typically otherwise lead to increased RF hardware costs. Recall that in many practical implementations, IFO estimation is avoided by restricting the CFO range to within that tolerated by the FFO estimator; typically -2 to $+2$ sub-carrier 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 FFO estimation and correction are performed prior to 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 Eqn. (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]:

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

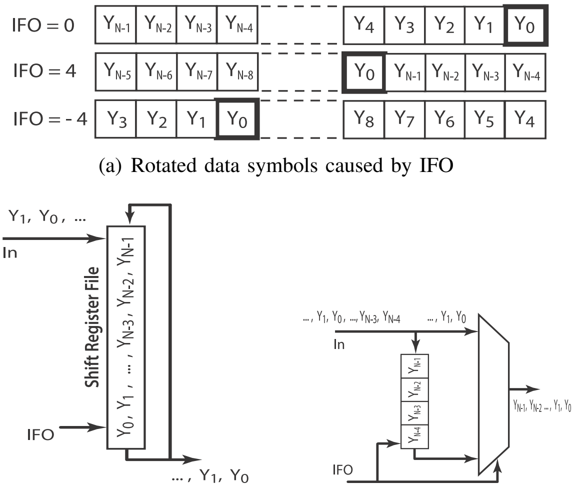
where $s(n)$ denotes the received signal with carrier frequency offset ξ with respect to $x(n)$. The two parts of ξ are estimated based on the angle of M ($\angle M$):

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

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

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 4 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 buffer requirements, we delay computation of the IFO by 12 symbols, so that the compensation shift is always positive: $S'_{IFO} = \{0 : 4 : 28\}$. This means that received symbols will only ever need to be shifted right to compensate 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. 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 illustrated in Fig. 3, distributed with 50 pilots per side at even sub-carrier spacings from 2 to 100 and from 156 to 254. The remaining sub-carriers are null. A trade off is possible by reducing hardware cost through 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.


IFO = 0
Y_{N-1} Y_{N-2} Y_{N-3} Y_{N-4} Y_4 Y_3 Y_2 Y_1 Y_0
IFO = 4
Y_{N-5} Y_{N-6} Y_{N-7} Y_{N-8} Y_0 Y_{N-1} Y_{N-2} Y_{N-3} Y_{N-4}
IFO = - 4
Y_3 Y_2 Y_1 Y_0 Y_8 Y_7 Y_6 Y_5 Y_4
..., Y_1, Y_0, ...
In
Shift Register File
Y_0, Y_1, ..., Y_{N-3}, Y_{N-2}, Y_{N-1}
IFO
..., Y_1, Y_0
..., Y_1, Y_0, ..., Y_{N-3}, Y_{N-4} ..., Y_1, Y_0
In
Y_{N-1} Y_{N-2} Y_{N-3} Y_{N-4}
Y_{N-1}, Y_{N-2} ..., Y_1, Y_0
IFO

(a) Rotated data symbols caused by IFO

(b) Conventional approach

(c) Proposed method

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 $\hat{\epsilon} = -4$.

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 four-fold resource sharing design. Hence, the IFO estimation can be expressed as:

$$\begin{aligned} \hat{\epsilon} &= \underset{\tilde{\epsilon}\in S'_{IFO}}{\operatorname{argmax}} |V_{\tilde{\epsilon}}|, \\ V_{\tilde{\epsilon}} &= \sum_{k=1}^{Np/4} P(4k)A1_{\tilde{\epsilon}}(k) + P(L+4k)A2_{\tilde{\epsilon}}(k), \end{aligned} \tag{5}$$

where $V_{\tilde{\epsilon}}$ is the cross-correlation between received pilots and pre-rotated 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

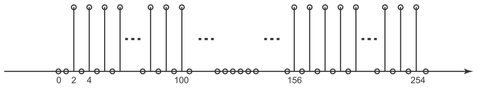


Fig. 3. Pilots in the long preamble of IEEE 802.16-2009.

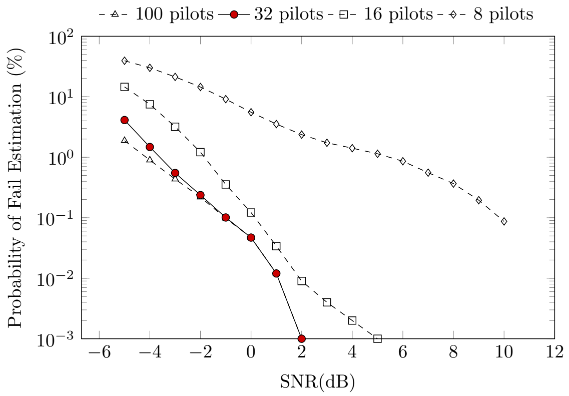


Fig. 4. Fail rate of IFO estimation for different number of used pillots in AWGN channel

sides of the OFDM symbol in the frequency domain, at even sub-carrier spacings, L denotes the index of the first pilots in the second half. $A1_{\tilde{\epsilon}}(k) = X^*(4k-2-\tilde{\epsilon})X(4k-\tilde{\epsilon})$, $A2_{\tilde{\epsilon}}(k) = X^*(L+4k-2-\tilde{\epsilon})X(L+4k-\tilde{\epsilon})$ denote the correlation of two consecutive pre-rotated 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:$\frac{Np}{2}$), (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}.

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 $k > Np/2$
```

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 $\tilde{\epsilon}$ in $S1'_{IFO}$, and $S2'_{IFO}$, the corresponding $V_{\tilde{\epsilon}}$ can be computed separately at every clock cycle using two multiply-accumulate blocks with the corresponding known coefficients. Hence, 8 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 multiply accumulate (MAC) operations, and clearly show the two common trade-off 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 the factor of two improvement over a straight-line interpolation between the extreme trade-offs. Moreover, 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 IFO estimation takes $D \times \frac{NP}{2}$ (i.e. 200 for 802.16) cycles. IFO estimation can be done immediately after the preamable packet is received and does not add latency to the OFDM packet stream. The next subsection will determine a method able to achieve this resource sharing for computing $V_{\tilde{\epsilon}}$ as well as in storing the pilots.

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	$D \times N_P/2$

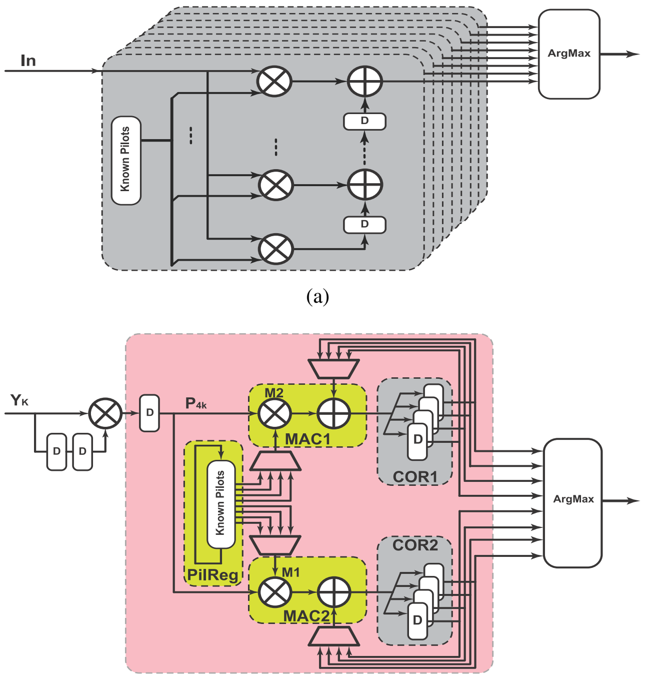
In addition, we will use multiplierless correlation with adjustable wordlength to further expand the performance/complexity trade off. Although sign-bit cross-correlation is often used in conventional implementations to reduce computational complexity [4], [21], it leads to reduced precision and hence reduced estimation performance, especially in frequency selective channels. By contrast, 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 trade-off between cost and accuracy for multiplierless correlation in the case of OFDM *timing* synchronisation, 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:

Sharing stored pilot memory: There are 8 sets of $A_{\tilde{\epsilon}}$ corresponding to 8 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 pre-rotated pilot sets. Therefore, the *PilReg* block requires only 64 shared memory locations for the 8 sets instead of 400; an 84% reduction. Fig. 6 illustrates the $A_{\tilde{\epsilon}}$ sets and circuitry for combining all $A_{\tilde{\epsilon}}$ sets in the *PilReg* block.

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 multiply accumulate block to be scheduled to sequentially compute 4 separate correlations. Multiply accumulate 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



(a)

(b)

Fig. 5. Architectures of IFO estimators: (a) conventional approach, and (b) proposed method.

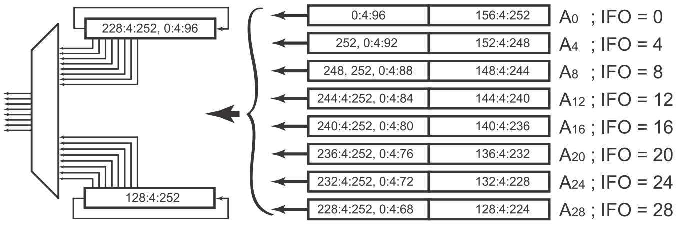


Fig. 6. Circuit of known pilots shift register.

subsampling of P_k, taking a subset of the most significant bits from P_k every four cycles to perform the cross-correlation. Two multiply accumulate blocks, *MAC1* and *MAC2*, are used to compute the values of 8 cross-correlations in parallel. Each multiplier performs multiplications sequentially between P_{4k} and the corresponding transmitted pilots in 4 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, $argmax|V|$, is performed on 8 $V_{\tilde{\epsilon}}$ values to estimate the IFO.

To obtain further resource savings, MAC1 and MAC2 are implemented using multiplierless techniques. $V_{\tilde{\epsilon}}$ in Eqn. (5) is mathematically manipulated into what is effectively a multiply-accumulate 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. $A_{\tilde{\epsilon}}$ are 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 trade off 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 the next section we will demonstrate through simulation on different channels that the algorithm and architecture optimisations mentioned above retain competitive estimation accuracy compared to 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 DSP blocks.

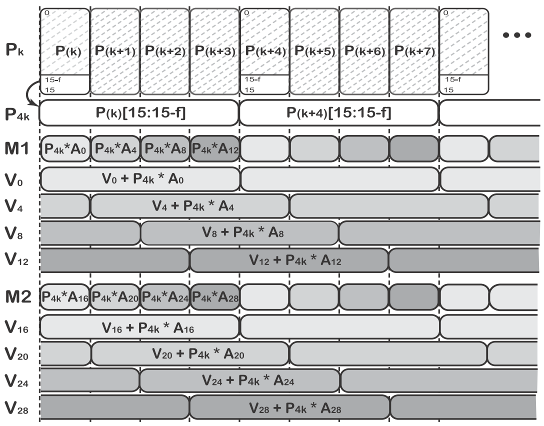


Fig. 7. Resource sharing approach for computing $V_{\tilde{\epsilon}}$.

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 to the theoretical performance of some state of the art methods in terms of POFE with respect to channel SNR. 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 (SUI) [24] frequency selective channels. IFO estimation is performed with non-ideal FFO compensation, and FFO is determined and compensated using the method of Kim and Park [4]. The simulation also verifies the performance of the proposed method under the effect of residual timing offset (RTO) caused by imperfect STO estimation (assuming that 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 $N_{CP} - L_h - 1$ (i.e. the RTO is in range from 0 to $N_{CP} - L_h - 1$).

We investigated the performance degradation compared to theory as a result of reducing the number of pilots in Section III-A and now investigate the effect of wordlength optimisation. 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,

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

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

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

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 *Prop*, is evaluated in comparison to the theoretical performance of state of the art methods by Park et. al. [2], Shim and You [3] and Morelli and Moretti [5], denoted *PCH*, *SY* and *MM*, respectively in the previous subsection. 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 *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 to the frequency domain methods of [5], [2], [3], 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 *PCH_50*. Figs. 8 plots performance results for all methods in an AWGN channel with RTO and reveals that the proposed 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 *PCH* at SNRs below 0 dB, while performance is very similar at larger SNRs. Also note the performance of the conventional approach implementations, *PCH_sb*, *SY_sb*, *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.

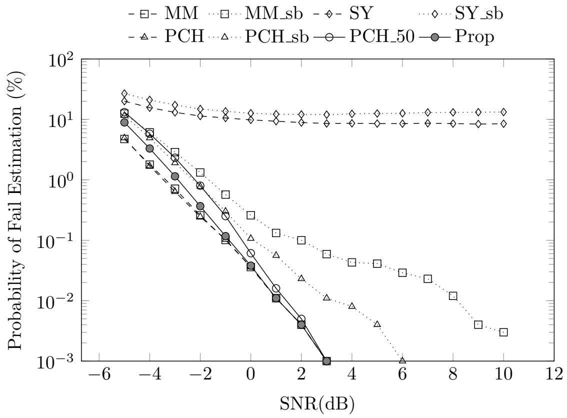


Fig. 8. Fail rate of IFO estimation methods in AWGN channel with RTO.

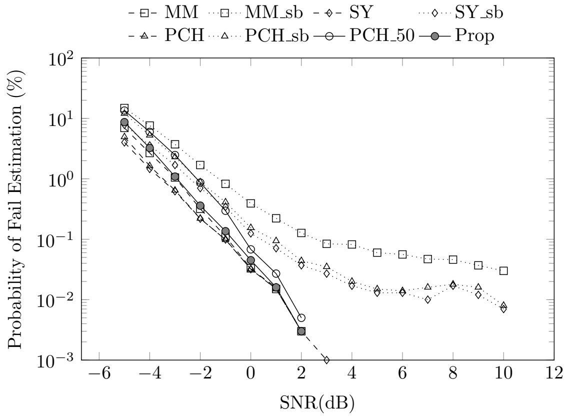


Fig. 9. Fail rate of IFO estimation methods in SUI1 channel.

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.

B. Wordlength Optimisation

Since sign-bit correlation degrades IFO estimation performance, especially in frequency selective channels, we instead explore the use of different word length multiplierless correlation to trade off 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 $Q1.f$,

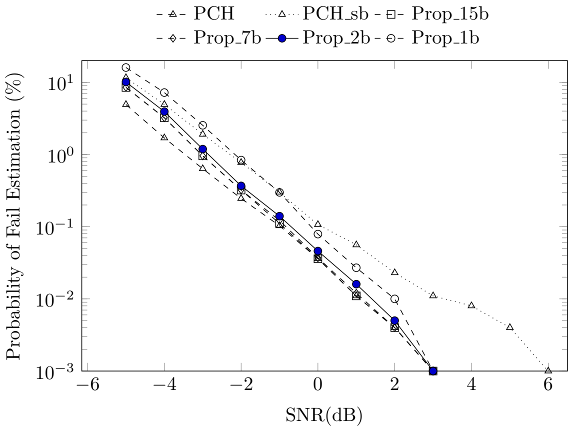


Fig. 10. Fail rate for different wordlengths in AWGN channel with RTO.

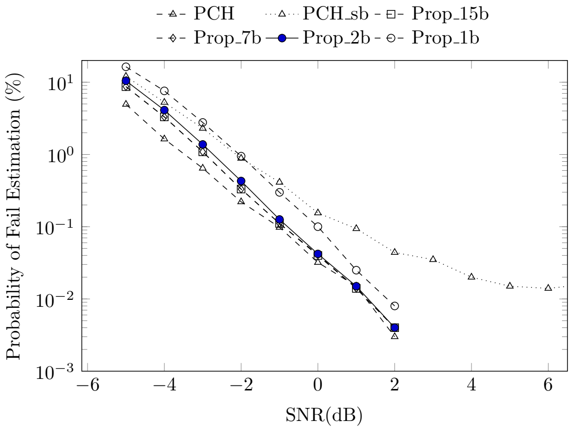


Fig. 11. Fail rate for different wordlengths in an SUI1 channel.

meaning a single integer bit and f fractional bits. Evaluations are performed for $f = \{1, 2, 7 \ \& \ 15\}$ bits with results plotted using the labels *Prop_1b*, *Prop_2b*, *Prop_7b*, and *Prop_15b*, respectively. Figs. 10 plots the performance in AWGN with RTO for all tested wordlengths. The proposed method performs comparably to *PCH* (and is better than *PCH_sb* at SNRs exceeding about 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 realisation, *PCH_sb*. Additionally, these realisations 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 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: Q1.1, Q1.2, Q1.7, and 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 optimised implementation, these circuits are each implemented using two different structures. The first uses only logic elements (LE) for computation, while the second uses Xilinx DSP48A1 [25] primitives. Considering (7), $\Re\{V_{\tilde{\epsilon}}\}, \Im\{V_{\tilde{\epsilon}}\}$ can be computed effectively using two DSP blocks as 3-input adders, instead of 4 blocks as would be usual. Fig. 12 illustrates 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 optimised for QPSK modulated pilots (since their amplitudes are identical) as specified in IEEE 802.16, as well as in most OFDM-based standards. The normalisation performed in (7) allows the correlation to be reduced to two DSP blocks operating as 3-input adders (instead of 4 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.

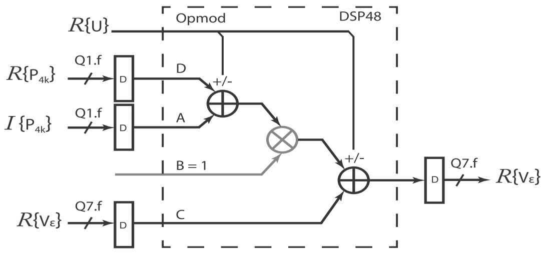


Fig. 12. DSP block based 3-input adder for correlation.

C. Implementation Results

The circuits were synthesised 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 flip-flops (FFs), look-up tables (LUTs), and DSP blocks, along with dynamic power consumption, as summarised in Table II.

TABLE II
RESOURCE UTILISATION 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

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 logic elements (LE) 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 3-input additions. The *Prop_fb_LE* implementations use 3 DSP blocks to compute P_{4k}, while *Prop_fb_DSP* require an additional 4 DSP blocks to perform the correlation. *Prop_fb_LE*, *Prop_fb_DSP* both 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 to the conventional *conv_100_sb* approach.

The maximum circuit frequencies, reported after place and route, are 142 MHz, 136 MHz 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 below 25 MHz.

A post-place-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 less than 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 utilisation. Since IFO estimation can contribute significantly to the complexity of a robust OFDM synchroniser design, this work is important for multi-standard 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 analogue 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 sub-sampling to enable a four-fold resource sharing architecture to reduce both hardware complexity as well as power consumption. Meanwhile, multiplierless correlation with optimised 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 post place-and-route implementation (to determine detailed resource utilisation 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 sub-sampled 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. D'Andrea, and U. Mengali, "Frequency ambiguity resolution in OFDM systems," *IEEE Communications Letters*, 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. Vehic. Techn. Conf. (VTC)*, 2002, pp. 2116–2120.

[3] E.-S. Shim and Y.-H. You, "OFDM integer frequency offset estimator in rapidly time-varying channels," in *Asia-Pacific Conf. on 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. on Very Large Scale Integration (VLSI) Systems*, 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 Transactions on Wireless Communications*, 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. on Broadcasting*, vol. 55, no. 1, pp. 95–108, Mar. 2009.

[7] C. Shahriar, M. La Pan, M. Lichtman, T. Clancy, R. McGwier, R. Tandon, S. Sodagari, and J. Reed, "PHY-Layer Resiliency in OFDM Communications: A Tutorial," *IEEE Communications Surveys Tutorials*, 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 *Sixth IEEE Inter. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 3, Sep. 1995.

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

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

[11] J. Guffey, A. Wyglinski, and G. Minden, "Agile radio implementation of OFDM physical layer for dynamic spectrum access research," in *IEEE Global Telecommunications 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. on Circuits and Systems I: Regular 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. on Very Large Scale Integration (VLSI) Systems*, vol. 20, 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. on Circuits and Systems I: Regular Papers*, vol. 59, no. 2, pp. 399–408, Feb. 2012.

[15] T. Pham, S. Fahmy, and I. McLoughlin, "Efficient multi-standard cognitive radios on FPGAs," in *International Conference on Field Programmable Logic and Applications (FPL)*, Sept. 2014.

[16] Y. Zhang, J. 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 Transactions on Circuits and Systems I: Regular 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 Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1582–1592, March 2014.

[18] ——, "Integer frequency offset estimation and preamble identification in WiMAX systems," in *IEEE International Conference on Communications (ICC)*, June 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 *IEEE International Congress on Image and Signal Processing (CISP)*, Oct. 2014, pp. 1012–1016.

[20] T. H. Pham, I. V. McLoughlin, and S. A. Fahmy, "Robust and Efficient OFDM Synchronisation for FPGA-Based Radios," *Circuits, Systems, and Signal Processing*, vol. 33, no. 8, pp. 2475–2493, Aug. 2014, Springer.

[21] L. Schwoerer, "VLSI suitable synchronization algorithms and architecture for IEEE 802.11a physical layer," in *IEEE Inter. Symp. on Circuits and Systems (ISCAS)*, vol. 5, 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. on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 8, pp. 1549–1553, Aug. 2013.

[23] *IEEE Standard for Local and Metropolitan Area Networks Part16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE std.802.16-2009.

[24] V. Erceg, K. V. S. Hari, and M. S. Smith, "Channel models for fixed wireless applications," *Tech. Rep. IEEE 802.16a-03/01*, Jul. 2003.

[25] *UG389: Spartan-6 FPGA DSP48A1 Slice*, Xilinx Inc., August 2009.

Pham Hung Thinh (S'13) received his B.S. degree in Electrical and Electronic Engineering from Ho Chi Minh City University of Technology, Vietnam, and the M.Sc. degree in Embedded Systems Engineering from the University of Leeds, U.K., in 2007 and 2010, respectively. He is completing his Ph.D. in the joint Nanyang Technological University-Technische Universität München program in Singapore in 2015 and is now a Research Associate at 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, UK, in 2003 and 2007, respectively.

From 2007 to 2009, he was a Research Fellow at Trinity College Dublin, and a Visiting Research Engineer with Xilinx Research Labs, Dublin. Since 2009, he has been an Assistant Professor with the School of Computer Engineering at Nanyang Technological University, Singapore. His 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, the IBM Faculty Award in 2013, and is also a senior member of the ACM.

Ian Vince McLoughlin split his career between the electronics R&D industry and academia, based in five countries on three continents. He became a Chartered Engineer (UK) in 1998, A Senior Member of IEEE in 2004, a D'Ingenieur Europeen (EU) in 2008, and a Fellow of the IET in 2013. He is a professor in the School of Computing at the University of Kent, UK. He was previously a professor at the University of Science and Technology of China (USTC) and before that a faulty member for 10 years at Nanyang Technological University, Singapore and working in UK and New Zealand industry for 10 years. His Ph.D. was completed at the University of Birmingham, UK, in 1997. He has published over 200 papers, has 13 patents, and books with Cambridge University Press and McGraw-Hill.