Impact of Quantization Noise Errors in FPGA Based WLAN Timing Synchronization with the Schmidl and Cox Algorithm

Jorge Torres Gómez[1], Liset Martínez Marrero[2], Matthias Koepp[3], Rajesh Kumar Sharma[4]
[1]*School of Electrical Engineering and Computer Science, TU Berlin, Berlin, Germany*
[2]*Intelligent Sensors and Networks (ISN), Lucerne University of Applied Sciences and Arts, Lucerne, Switzerland*
[3]*Photonic Networks and Systems Department, Fraunhofer Heinrich Hertz Institute, Berlin, Germany*
[4]*Department of Digital Signal Processing and Circuit Technology, Chemnitz University of Technology, Chemnitz, Germany*
[1]torres-gomez@ccs-labs.org, [2]liset.martinezmarrero@hslu.ch, [3]matthias.koepp@hhi.fraunhofer.de,
[4]rajesh-kumar.sharma@etit.tu-chemnitz.de

***Abstract*—The implementation of accurate solutions to synchronize the transceiver's clocks in real-time operations plays a major role in the current development of communication systems. Considering the orthogonal frequency division multiplexing (OFDM)-based systems, adopted for the deployment of fifth-generation mobile networks (5G), industrial networks, and the wireless local area network (WLAN) standard, several reported solutions focus on field-programmable gate array (FPGA) designs for the clock synchronization in real-time operations. However, reported solutions describe with minor details the impact of the introduced quantization noise error (QNE) based on the implemented bit-width. Their studies provide insights to implement digital solutions of reduced complexity. The current article provides theoretical expressions to characterize accuracy and precision while synchronizing the transmitter's clock. The presented analysis is intended to support designers to evaluate the clock synchronization design considering the bit-width for implementing the given system.**

***Index Terms*—WLAN, Clock Synchronization, FPGA, Quantization Noise Errors.**

I. Introduction

In present-days, wireless communication systems must be designed to cope with challenging application fields, for instance, the massive connection of mobile devices, industrial networks, remote sensing-related applications, as well as intelligent transportation systems [1]–[3]. Specifically, 5G and industrial wireless networks (based on IEEE 802.11 or 802.15.4 standards) adopt the use of OFDM-based modulation schemes in WLAN topologies to implement the physical layer [2], [4].

One of the major challenges in real-time and reliable hardware implementations for wireless communication systems is the synchronization issue. As depicted in Fig. 1, each communicating node operates guided by its own system's clock to transmit packets asynchronously. Based on the received packets, each receiver node must minimize errors between its reference and the transmitter clock to be aligned with the received data. A misalignment of the clocks may produce a self-interference that limits the received signal-to-noise ratio (SNR) to be less than 20 dB [5], which in turn will degrade the rate of transmission. To that end, preamble-based synchronization methods are implemented to identify the dedicated synchronization sequences prepended to the data packet.

Current research project is part of the Elektromobilität mit Redundanter Intelligenter Kommunikations-Architektur (ERIKA) Project, supported by the Bundesministerium für Wirtschaft und Energie, Germany.

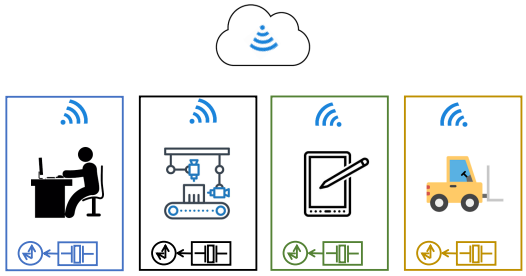

Fig. 1. Wireless interconnection of nodes inside the factory.

Due to the need of timely operation of synchronizers, several solutions are reported in FPGA to implement the real-time functioning of the synchronization algorithms. Its major concern is regarded as low complexity designs due to limited processing time and hardware resource utilization. A low complexity, fast processing synchronization algorithm known as Schmidl & Cox algorithm [6] has been often used in real-time FPGA systems; either unchanged [7], or resource optimized, as described in [8], [9] and [10]. The work in [6] has proposed to search the maximum-likelihood between repetitive preamble parts by implementing the auto-correlation procedure. Auto-correlation has a significant advantage over cross-correlation in terms of complexity, but not in its performance [11]. Therefore, systems based on [6], often need to apply a fine correction to the synchronization result afterward, which is also true for the reported solutions in [8] to [10], where [9] and [10] utilize the preamble structure of the standards mentioned above. For high-speed future systems, the work in [12] has proposed to send 1-bit repetitive sequences to provide sufficient performance from [6]. The IEEE802.11 case analyzed below defines a multi-level sequence. To characterize these algorithms, two metrics are widely used, i.e. accuracy

and precision [4]. The former is related to the location of the peaks at the output of the correlator regarding the ideal case, while the precision is given by the variance of their position.

However, the methods reported above are not fully described in terms of the impact of QNE. QNE will introduce additional noise sources in the system which, in turn, will deteriorate the system performance. In this direction, only the report in [13] has analyzed the impact of the fixed point implementation in the synchronization system supported by simulations. However, a detailed analysis to balance synchronization performance (accuracy and precision) in terms of implemented bit-width, is still unreported.

The current article addresses the study of reported solutions to synchronize packets in the IEEE 802.11 standard by analyzing and evaluating the impact of QNE on the system performance. We aim to derive the proper analytic expression to derive the QNE power at the output of the Schmidl and Cox algorithm when implemented in FPGA. The resulting expression will allow evaluating the impact of the total bit-width in the achievable accuracy and precision of the synchronizers. This may also support designers when implementing their own solutions in digital technology.

The content of the paper is organized as follows. Based on frame synchronization algorithms to estimate the symbol time offset (STO), a detailed analytical model to implement hardware time stamping mechanisms and their verification is provided in Section II. As a result, we report the impact on the system performance in terms of the bit-width to represent the digital sequences in Section III. The resulting study and their simulation is provided in Section IV), which will support with further guidelines to designers on the real hardware implementation of OFDM synchronization systems in FPGA.

II. System Model and FPGA description of the clock synchronization algorithm

Based on the IEEE 802.11 standard, the transmitter and receiver clocks are synchronized based on the received preamble sequence [9]. Two synchronization algorithms are typically used to detect the asynchronous arrival of a packet-based on auto-correlation and cross-correlation mechanisms. Auto-correlation is applied to obtain a coarse estimation of the STO (to be implemented with the SS symbols), whereas, cross-correlation operation is implemented for the fine STO estimation with improved accuracy (to be implemented with the LS symbols) as depicted in Fig. 2.

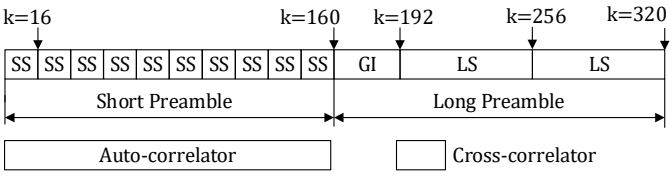

Fig. 2. IEEE Preamble and time line.

Following the reported solution in [9] (based on the Schmidl & Cox algorithm [6]), the coarse STO estimator is implemented by computing the autocorrelation of the received signal $R_{rr}[\tau]$, where $\tau = 16$. The value of $\tau = 16$ is based on the inherent periodicity due to the inserted SS sequence in the short preamble, as depicted in Fig. 2. The auto-correlation is estimated by using 144 samples as $\widehat{R}_{rr}[k] = \mathbf{r}_k^H \cdot \mathbf{r}_{k-16}$, where $\mathbf{r} = [r_k \; r_{k+1} \; ...r_{k+143}]$ represents a vector of 144 samples of the received OFDM signal sampled at $f_m = 20$ MHz. The auto-correlator is computed over 144 samples expanded over all the SS symbols from each received preamble (except for the first 16 samples because of the correlator-lag). Their implementation will produce a peak, ideally located at sample $k = 160$ after the last SS symbol. The sequence for the symbols GI and LS will not exhibit the same periodicity, thus the 16 samples-lag auto-correlator output will be nearly zero over these time intervals.

The FPGA block diagram to implement the coarse STO estimation method is depicted in Fig. 3 [9]. In this diagram, the analog-to-digital converter (ADC) blocks will introduce a two's complement binary representation of the input signal, those to be processed through the blocks connected next to the right. The coarse estimation method is implemented for the absolute value of the auto-correlation procedure regarding $\widehat{R}_{rr}[k]$. To that end, their inner product is decomposed, for the real $\widehat{R}_{rr\_\mathrm{Re}}[k]$ and the imaginary parts $\widehat{R}_{rr\_\mathrm{Im}}[k]$, as $(\sum_{l=1}^{144}(r_{(k-l)\_\mathrm{Re}} \times r_{(k-l-16)\_\mathrm{Re}} + r_{(k-l)\_\mathrm{Im}} \times r_{(k-l-16)\_\mathrm{Im}})$, and $\sum_{l=1}^{144}(r_{(k-l)\_\mathrm{Im}} \times r_{(k-l-16)\_\mathrm{Re}} - r_{(k-l)\_\mathrm{Re}} \times r_{(k-l-16)\_\mathrm{Im}})$, respectively, where the terms $r_{(k-l)\_\mathrm{Re}}$ and $r_{(k-l)\_\mathrm{Im}}$ are the real and imaginary parts of the received preamble, $r_{(k-l-16)\_\mathrm{Re}}$ and $r_{(k-l-16)\_\mathrm{Im}}$ are the delayed real and imaginary parts of the received preamble by 16 samples, respectively, and k is the time index.

The coarse STO estimation method is implemented by properly obtaining these four terms as depicted in Fig. 3. Departing from the received real and imaginary sequences of the preamble, delay and multiplier blocks are connected to produce these four terms. Finally, the resulting four terms are added and interconnected to the average block, which implements a sliding window of 144 samples. The outputs of the average blocks are both squared and added to obtain the coarse estimation output.

III. Modeling the impact of QNE in the clock synchronization performance

The QNE produced by the digital system results from two sources: the finite precision representation of the received signal at the output of the ADC blocks, and the finite representation of the system coefficients to process the signal. In our case, given that the diagram in Fig. 3 is free of coefficients, then the QNE will be only produced by the ADC blocks.

Provided that the system will operate with a finite total number of bits B, a noisy term will be introduced by the quantization procedure, which is typically modeled by a white noise random variable with uniform probability density function in the range $[-\Delta/2 \; \Delta/2]$, where $\Delta = \frac{X_{\mathrm{ADC}}}{2^{2B}}$ is the maximum error introduced by the ADC, and X_{ADC} is

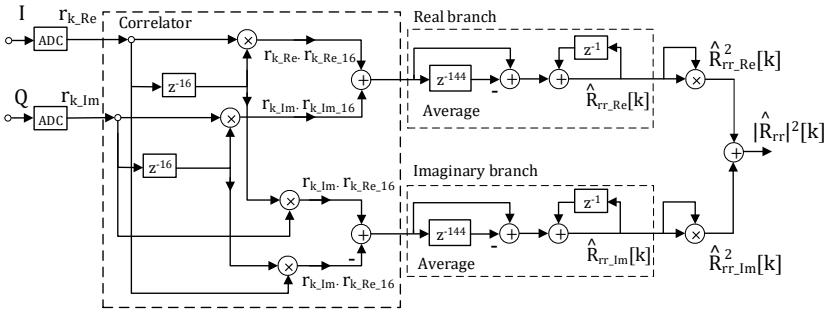

Fig. 3. Block diagram for the coarse STO estimation method (Auto-correlator).

the full ADC dynamic amplitude range [14]. Considering the received signal r_k, then the QNE model will introduce an additional random variable $n_k^{(q)}$ at the output of the ADC as $r_k^{(\mathbf{ADC})} = r_k + n_k^{(q)}$, where $r_k = s_k + n_k^{(g)}$, s_k is the received signal, $n_k^{(g)}$ represents the Gaussian noise signal introduced by the channel, and k is the time index.

To describe the impact of QNE on the implemented coarse clock synchronization method, here we model the output peaks by a triangle function as depicted in Fig. 4 with the solid line. This will be in correspondence with the peaks produced by the coarse method. The amplitude (A) of the triangle function will be equated to the maximum peak of the auto-correlator output as $A = \max R_{rr}^2[k] = \left(\sum_{k=0}^{144} |s_k|^2\right)^2$ in accordance with the diagram in Fig. 3 and the preamble sequence in Fig. 2. The width of the triangle function (τ) is adjusted to provide the same standard deviation of the produced peaks at the correlator output as $\tau = 3\sqrt{\frac{2}{3}} \times \left(\frac{\sum_{1=1}^{144}\left(k-\frac{144}{2}\right)^2 R_{rr}^2[k]}{\sum_{1=1}^{144} R_{rr}^2[k]}\right)^{\frac{1}{2}}$ given that the standard deviation of a given triangle function is $\Delta_\tau = \frac{1}{3}\sqrt{\frac{3}{2}}\tau$ [15].

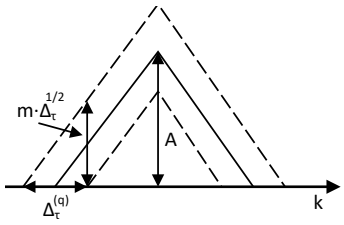

Fig. 4. Modeling the output of the synchronization methods and the impact of the introduced QNE.

Considering the verified QNE standard deviation at the output of both systems, denoted by $\Delta_{oq}^{\frac{1}{2}}$, this corrupting noise will modify the amplitude by a quantity less than $m \times \Delta_{oq}^{\frac{1}{2}}$, with a probability bounded by $P_{\Delta_{oq}} \leq \left(1 - \frac{1}{m^2}\right)$ (Chebyshev's inequality [15]). This is represented by the dashed lines in Fig. 4. By geometric considerations, the introduced spreading in the time domain ($\Delta_\tau^{(q)}$) produced by the QNE spreading at the output of the system ($m \times \Delta_{oq}^{\frac{1}{2}}$) will be given by

$$\Delta_\tau^{(q)} = \frac{\tau}{A} m \Delta_{oq}^{\frac{1}{2}}. \tag{1}$$

Besides, to provide a metric based on the relative introduced spreading by QNE in comparison to the ideal implementation (infinite total number of bits), we evaluate the metric

$$\frac{\Delta_\tau^{(q)}}{\Delta_\tau} = 3\sqrt{\frac{2}{3}}\frac{1}{A} m \Delta_{oq}^{\frac{1}{2}}. \tag{2}$$

By means of the equation in (2) we describe the impact of QNE in the precision of the recovered clock by the term $\Delta_{oq}^{\frac{1}{2}}$. Implicitly, in this model, we consider that QNE will increase the spreading and not the location of peaks, which in turn will imply that the system accuracy is not affected. This assumption will be illustrated later in Section IV.

A. QNE output variance for the coarse synchronization method

Considering the coarse method, implemented by the diagram in Fig. 3, the output signal on the upper branch $R_{\text{rr\_Re}}^2[k]$ will be given by

$$R_{rr}^2[k] = R_{16}^2[k] + 2R_{16}\sum_{l=0}^{143} n_{k-l}^{(q)} n_{k-l-16}^{(q)} + \\ + \left(\sum_{l=0}^{143} n_{k-l}^{(q)} n_{k-l-16}^{(q)}\right)^2, \tag{3}$$

where $R_{16}[k]$ represents the auto-correlation of the received signal r_k with lag 16 computed in the ideal case (infinite total number of bits). The second and third terms in (3) represent the effect of random QNE provided the finite implementation of the block diagram in Fig. 3 with a total of B bits. Given the first and second -moments of $n_k^{(q)}$, given by μ_q and σ_q^2, respectively, the variance of the output random signal Δ_{oq} is given by

$$\Delta_{oq} = E\left\{f(\mathbf{x})^2\right\} - E\left\{f(\mathbf{x})\right\}^2, \tag{4}$$

where $\mathbf{x} = [x_k, x_{k-1}, ..., x_{k-143}]$ represents a vector with components $x_k = n_k^{(q)} \times n_{k-16}^{(q)}$ and $f(\mathbf{x})$ is referred to the last two-terms in (3) as

$$f(\mathbf{x}) = 2R_{16}\sum_{l=0}^{143} x_{k-l} + \left(\sum_{l=0}^{143} x_{k-l}\right)^2. \tag{5}$$

Besides, the terms in (4) can be approximated by the first three terms of the following Taylor series as [16]

$$E\{f(\mathbf{x})\} \approx f(\bar{\mathbf{x}}) + f'(\mathbf{x})|_{\mathbf{x}=\bar{\mathbf{x}}}\overline{\mathbf{x}-\bar{\mathbf{x}}} + \\ + \frac{1}{2} \times f''(\mathbf{x})|_{\mathbf{x}=\bar{\mathbf{x}}}\overline{(\mathbf{x}-\bar{\mathbf{x}})^2} \approx \frac{1}{2} \times 2 \times 144\sigma_{\mathbf{x}}^2, \tag{6}$$

and similarly

$$E\{f(\mathbf{x})^2\} \approx \frac{1}{2} \times \left(2R_{16}\sum_{l=0}^{143} x_{k-l} + \left(\sum_{l=0}^{143} x_{k-l}\right)^2\right)\Bigg|_{\mathbf{x}=\bar{\mathbf{x}}} \times \\ \times \left(2 \times 144R_{16} + 2 \times 144\sum_{l=0}^{143} x_{k-l}\right)\Bigg|_{\mathbf{x}=\bar{\mathbf{x}}} \sigma_{\mathbf{x}}^2 \\ \approx 4 \times 144^2 R_{16}\sigma_{\mathbf{x}}^2. \tag{7}$$

Next, by considering each component of the vector $\mathbf{x}$ where $n_{k-l}^{(q)}$ and $n_{k-l-16}^{(q)}$ are independent random variables, then $\overline{x_k} = \mu_{n_{k-l}^{(q)}}\mu_{n_{k-l-16}^{(q)}} = 0$, and $\sigma_{\mathbf{x}}^2 = \frac{\Delta^4}{12^2}$ provided that $\Delta_q = \frac{\Delta^2}{12}$ [14]. Finally, the variance at the output of the coarse synchronization method can be approximated by $\Delta_{oq} \approx 4 \times 144R_{16}^2\Delta^4 - \Delta^8$. Then, having all together, by replacing the obtained result for Δ_{oq} in (1), we obtain the introduced additional variance by the QNE effects.

IV. Results and Discussion

To assess the correctness of the theoretical formulation, we evaluate the relative error regarding the predicted variance (by means of the theoretical expression in (1)) and the produced by the actual implementation of the system in Fig. 3 as $\left(\left|\Delta_{\tau}^{(q)} - \Delta_{\tau,\text{FPGA}}^{(q)}\right| / \Delta_{\tau,\text{FPGA}}^{(q)}\right)$. The obtained ratios are around 2% and are independent of the bit-width, what shows that the theoretical model fits the implementation results with a fixed small error for each value of B.

As an evaluation of the impact of QNE for a given SNRs, Fig. 5 depicts the accuracy and precision (in samples) for the coarse estimator. Given the clock difference between the transmitter and the reference in the receiver, denoted by ϵ, accuracy is measured by their mean value $\bar{\epsilon}$, whereas precision is represented by their standard deviation $(\overline{\epsilon^2})^{\frac{1}{2}}$. These are the two common metrics to evaluate how well the system is aligned with the incoming frame [4]. Regarding the implemented design as represented by the block diagram in Fig. 3, nearly similar results are obtained independent of the total number of bits (8, 12, 16, and 32). The constant behavior regarding the bit-width can be explained considering the white statistical process regarding the QNE. By implementing the auto-correlator procedure, the white noise QNE process is effectively canceled out as long as white noise processes have zero correlation except for $\tau = 0$. Consequently, this result implies that an increased total number of bits will not improve performance, but introduce larger resource utilization and increase power consumption. Usage of high-resolution DSP blocks but general FPGA logic might also be sub-optimal, shown for cross-correlation in [17].

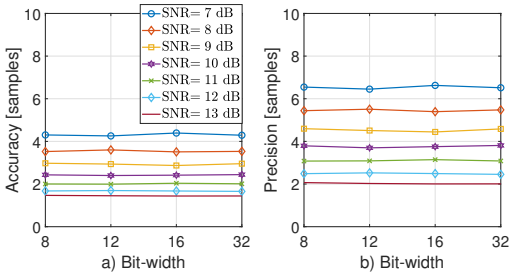

Fig. 5. Impact of the QNE in the SNR range [7 20] dB regarding accuracy and precision for the coarse synchronization method.

Fig. 6 evaluates the symbol error rate (SER) to account for the impact of the QNE when transmitting quadrature shift keying (QPSK)-OFDM symbols. To that end, the MatLab® [1] code provided in [18] was used to obtain these curves, where misalignments were introduced by implementing a fractional delay filter as in [19]. Following the curves in Fig. 6, the 8-bits case introduces the largest performance difference, while the 12, 16, and 32 bits behave nearly similar to the ideal correlator. Future studies will be conducted to analyze in further detail this marked difference.

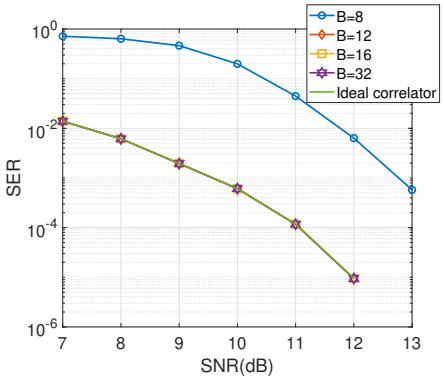

Fig. 6. Impact of the QNE in the SER for QPSK symbols transmitted using OFDM.

V. Conclusions and Outlooks

We investigated the impact of QNE on auto-correlation based symbol time offset estimation for an OFDM based system. To that end, we derived theoretical expressions to account for the impact of the QNE based on the Schmidl & Cox algorithm. Results exhibited the validation of the derived theoretical expressions with a close correspondence to the current implementation. The precision and accuracy also exhibited not being substantially influenced by the QNE. Larger bit-width (at least beyond 8-bits) will not produce an improved performance regarding accuracy and precision, but increases the complexity. The analysis regarding sampling and carrier frequency offset is still to be considered. The impact on performance by different algorithm resolutions also remains open for future work.

[1] MATLAB is a registered trademark of The MathWorks, Inc.

REFERENCES

[1] Y. Yang and K. Hua, "Emerging Technologies for 5G-Enabled Vehicular Networks," *IEEE Access*, vol. 7, pp. 181 117–181 141, 2019.

[2] A. A. Zaidi, R. Baldemair, V. Moles-Cases, N. He, K. Werner, and A. Cedergren, "OFDM Numerology Design for 5G New Radio to Support IoT, eMBB, and MBSFN," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 78–83, Jun. 2018.

[3] S. Vitturi, C. Zunino, and T. Sauter, "Industrial Communication Systems and Their Future Challenges: Next-Generation Ethernet, IIoT, and 5G," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 944–961, Jun. 2019.

[4] A. Mahmood, R. Exel, H. Trsek, and T. Sauter, "Clock Synchronization Over IEEE 802.11—A Survey of Methodologies and Protocols," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 907–922, Apr. 2017.

[5] X. Lin, L. Jiang, and J. G. Andrews, "Performance Analysis of Asynchronous Multicarrier Wireless Networks," *IEEE Transactions on Communications*, vol. 63, no. 9, pp. 3377–3390, Sep. 2015.

[6] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1613–1621, Dec. 1997.

[7] A. Ishtiaq, A. Javed, A. Akhtar, U. B. Zulfiqar, and M. D. Nisar, "Efficient implementation of OFDM waveform on Xilinx FPGA," in *2017 International Symposium on Wireless Systems and Networks (ISWSN)*, 2017, pp. 1–5.

[8] F. J. Lopez-Martinez, E. Martos-Naya, J. T. Entrambasaguas, and M. Garcia-Abril, "Low complexity synchronization and frequency equalization for OFDM systems," in *2007 14th IEEE International Conference on Electronics, Circuits and Systems*, 2007, pp. 987–990.

[9] M. J. Canet, J. Valls, V. Almenar, and J. Marín-Roig, "FPGA implementation of an OFDM-based WLAN receiver," *Microprocessors and Microsystems*, vol. 36, no. 3, May 2012.

[10] L. Orozco-Galvan, R. Parra-Michel, F. Peña-Campos, R. Jaramillo-Ramirez, and E. Romero-Aguirre, "A resource efficient symbol synchronizer implementation for the IEEE 802.11 protocol," in *2018 IEEE 9th Latin American Symposium on Circuits Systems (LASCAS)*, 2018, pp. 1–4.

[11] A. Fort, J.-W. Weijers, V. Derudder, W. Eberle, and A. Bourdoux, "A performance and complexity comparison of auto-correlation and cross-correlation for OFDM burst synchronization," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, vol. 2, 2003, pp. II–341.

[12] S. Jacobsson, C. Lindquist, G. Durisi, T. Eriksson, and C. Studer, "Timing and frequency synchronization for 1-bit massive MU-MIMO-OFDM downlink," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2019, pp. 1–5.

[13] J. Vazquez, B. Castan, and L. Campoy, "Quantification and round influence on FPGA implemented algorithms for an OFDM radio interface," in *2004 IEEE 15th International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE Cat. No.04TH8754)*, vol. 4, Sep. 2004, pp. 2534–2538 Vol.4.

[14] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing: Pearson New International Edition*, 3rd ed. Harlow: Pearson Education Limited, 2014.

[15] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, 4th ed. Boston, Mass.: McGraw-Hill Europe, Jan. 2002.

[16] P. Whittle, *Probability via Expectation*, 4th ed., ser. Springer Texts in Statistics. New York: Springer-Verlag, 2000. [Online]. Available: https://www.springer.com/gp/book/9780387989556

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

[18] "SER Simulation for OFDM Link - MATLAB & Simulink - MathWorks Deutschland." [Online]. Available: https://de.mathworks.com/help/comm/ug/802-11-ag-ber-simulation.html

[19] "Fractional Delay Filters Using Farrow Structures - MATLAB & Simulink - MathWorks Deutschland." [Online]. Available: https://de.mathworks.com/help/dsp/ug/fractional-delay-filters-using-farrow-structures.html