JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

FPGA IMPLEMENTATION OF 24-BIT DIGITAL FREQUENCY SYNTHESIZER

[1]K. Satish Kiran, [2]A. Swapna

[1]M. Tech Student, [2]Assistant Professor

[1]Electronics and Communication Engineering,

[1]SVP College of Engineering, Visakhapatnam, India

Abstract: Digital Direct Frequency Synthesizers (DDFS) offer a highly accurate and efficient method for frequency generation, presenting notable advantages over traditional analog synthesizers. The primary components of a DDFS system are the phase accumulator, lookup table (LUT), and digital-to-analog converter (DAC). The phase accumulator is responsible for accumulating phase increments, the LUT holds pre-computed digital waveform samples, and the DAC transforms these digital values into analog signals for waveform generation. Despite their benefits, DDFS systems face challenges such as phase truncation errors, DAC quantization noise, and spurious signals. However, these issues can be addressed through techniques like phase dithering, noise shaping, and system calibration. DDFS technology is well-suited for applications that demand high-frequency resolution and fast frequency switching, such as telecommunications, radar, and software-defined radio systems.

Index Terms - Digital Direct Frequency Synthesizer (DDFS), Phase accumulator, Lookup table (LUT), Frequency Control Word (FCW), and Simulation.

I. INTRODUCTION

Digital frequency synthesis has become a cornerstone technology in modern communication systems and signal processing applications, offering the advantages of high precision, fast frequency switching, and low noise operation. Among the most widely adopted techniques is the Digital Direct Frequency Synthesizer (DDFS), also known as a Numerically Controlled Oscillator (NCO), which enables flexible and accurate generation of sinusoidal signals through purely digital methods. This project focuses on the design and FPGA implementation of a 24-bit DDFS capable of generating output frequencies ranging from 35 kHz to 45 kHz with a fine resolution of 0.3 Hz. The core objective is to realize a reliable and efficient digital frequency synthesis system that meets the stringent demands of modern electronic applications.

The proposed DDFS architecture is based on the fundamental principle of phase accumulation, where a Frequency Control Word (FCW) is incrementally added to a Phase Accumulator on each clock cycle. The output of the accumulator is then mapped to a corresponding amplitude value using a Phase-to-Amplitude Conversion (PAC) block, typically implemented via a lookup table (LUT) to generate a digitally synthesized sine wave. The output frequency (Fout) of the system is defined by the relationship Fout= (fclk×FW)/2^N, where fclk is the reference clock frequency, FW is the frequency control word, and N is the number of bits in the accumulator. By carefully selecting the bit-width of the accumulator and the reference clock, the design achieves fine frequency resolution, as defined by the formula Resolution=fclk/2^N.

A key performance goal of the system is the ability to perform fast frequency switching with minimal latency. This is particularly important in applications that demand real-time frequency agility, such as spread-spectrum communications and adaptive filtering. The DDFS design enables seamless reconfiguration of the output frequency through simple digital updates to the FCW, eliminating the need for complex analog tuning mechanisms. In addition to frequency agility, the synthesizer is optimized to minimize phase noise and spurious signals, which are critical to maintaining high signal integrity in sensitive applications.

The implementation is carried out on an FPGA platform using VHDL as the hardware description language and Xilinx Vivado for synthesis, simulation, and verification. The choice of FPGA offers significant advantages, including parallelism, reconfigurability, and the ability to prototype and validate the system in real-time. Functional verification is conducted using comprehensive testbenches to evaluate output frequencies, switching behavior, and resolution accuracy. Moreover, timing analysis and resource utilization studies are performed to ensure that the design meets performance constraints while maintaining efficiency.

Finally, the designed DDFS is intended to serve as a versatile solution suitable for a wide range of applications, from wireless communication systems to digital instrumentation and embedded signal processing. Future work will explore enhancements such as wider tuning ranges, dynamic frequency hopping capabilities, and optimizations for low power consumption, particularly for mobile and battery-operated devices. Through this project, a scalable and high-performance digital frequency synthesis framework is presented, addressing both current needs and future challenges in the field of digital signal generation.

II. OVERVIEW OF DIGITAL FREQUENCY SYNTHESIZER

Digital Frequency Synthesis (DFS) is a technique used to generate stable and accurate frequency outputs using digital components such as digital logic circuits or Field-Programmable Gate Arrays (FPGAs), in contrast to traditional analog methods. The shift toward digital synthesis has become increasingly prominent in modern electronic systems due to its superior precision, flexibility, and scalability. DFS is particularly essential in applications that require precise frequency control, including communication systems, digital signal processing, instrumentation, and measurement systems. Its digital nature allows for predictable and repeatable performance, making it a reliable choice for high-performance systems.

At the core of a typical DFS system lies a reference clock signal, which serves as the foundational timing source. This reference is manipulated using a digital subsystem, most commonly a Direct Digital Frequency Synthesizer (DDFS), to generate the desired output frequency. A DDFS employs digital arithmetic and algorithmic processing to produce highly accurate frequency outputs. One of its central components is the phase accumulator, which incrementally adds a value known as the Frequency Control Word (FCW) at each clock cycle. The FCW determines the rate at which the phase accumulator progresses, thereby controlling the output frequency. The resulting phase information is then mapped to amplitude values through a phase-to-amplitude conversion mechanism, typically using a lookup table, and can be converted to an analog signal using a Digital-to-Analog Converter (DAC) if needed.

Compared to conventional analog methods such as Voltage-Controlled Oscillators (VCOs) and Phase-Locked Loops (PLLs), DFS offers significant advantages. These include high frequency accuracy, fast frequency switching capability, and strong immunity to analog noise sources. Digital systems are inherently more robust against environmental variations and component aging, which often affect analog circuitry. Moreover, the reprogrammability of digital logic devices like FPGAs adds a layer of versatility, allowing users to modify or extend functionality with ease. The project titled **FPGA Implementation of a 24-Bit Digital Frequency Synthesizer** leverages these benefits by implementing a high-resolution 24-bit DDFS on an FPGA platform. This design approach enables precise frequency generation with fine resolution and high stability, making it well-suited for a wide range of demanding applications, from telecommunications to medical instrumentation and beyond.

III. PROPOSED DESIGN METHOD

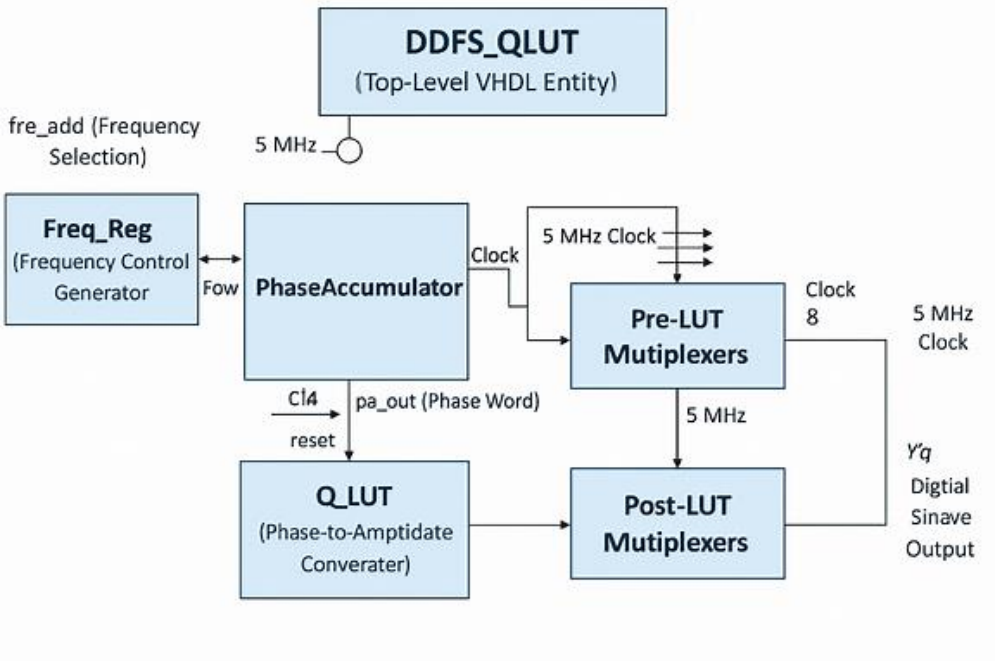

Fig-3.1 Block Diagram of Proposed Methodology

The proposed design for the 24-bit Digital Direct Frequency Synthesizer (DDFS) was implemented and verified using the Xilinx Vivado toolchain. The core frequency synthesis architecture relies on a 24-bit phase accumulator that increment by a value defined by the Frequency Control Word (FCW) on every rising edge of a 5 MHz reference clock. The FCW determines the output frequency, which is computed according to the relation: Fout= (FCW×fclk)/2^N

Where fclk is the system clock frequency (5 MHz), N is the bit-width of the phase accumulator (24 bits), and FCW is the digital control word generated by the frequency register module (Freq_Reg). The Freq_Reg module maps discrete 5-bit frequency selection inputs (fre_add) to corresponding FCW values that cover the desired output frequency range from approximately 35 kHz to 45 kHz.

The digital phase accumulator generates a phase value that advances proportionally to the FCW, producing a phase ramp that cycles through 2^24 discrete states per full waveform period. This phase information is then fed into a quantized lookup table (Q_LUT) with pre- and post-LUT multiplexers that translate the phase into an 8-bit quantized sine wave output. This digital waveform can be further processed or converted to analog using a DAC if required. The design emphasizes modularity and fast frequency switching by allowing dynamic updates to the FCW without interrupting the phase accumulator's operation.

The implementation process involved writing VHDL code for all modules, synthesizing the design in Vivado, and running detailed simulations. An asynchronous reset signal initializes the phase accumulator and internal registers to ensure known startup conditions. Frequency switching tests were performed by changing the 5-bit frequency selection input (fre_add) in simulation to verify the synthesizer's ability to produce different frequencies accurately and seamlessly.

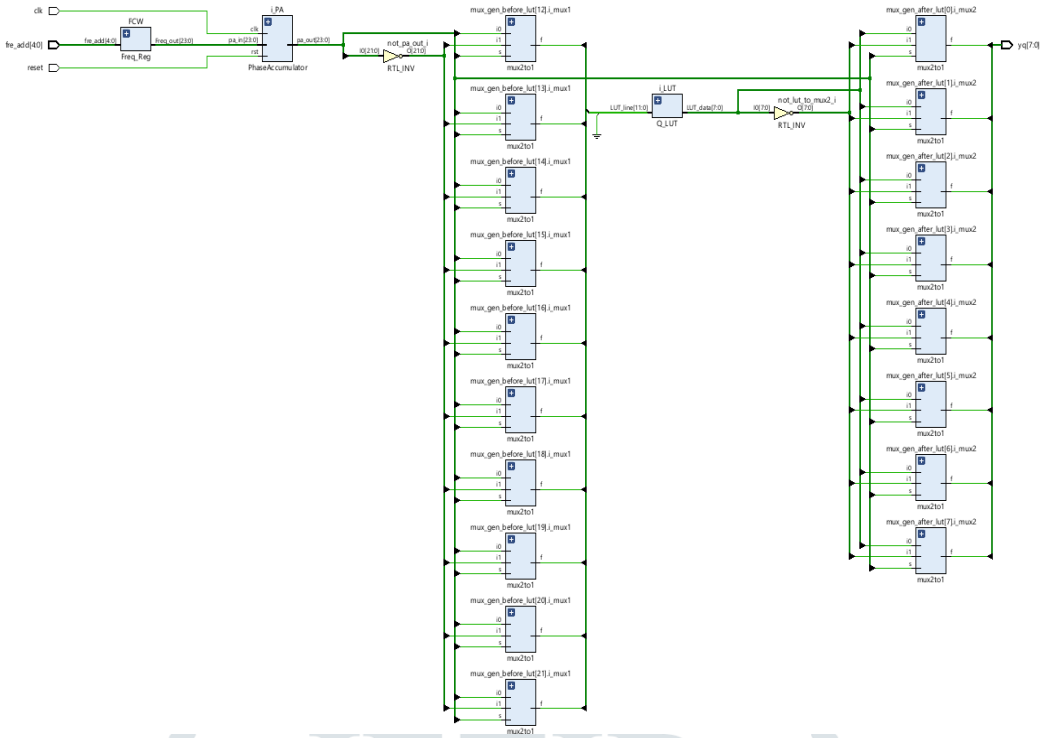

Fig-3.2 Schematic Representation of Digital Frequency Synthesizer

The design also incorporates careful attention to noise reduction and signal integrity. By using a high-resolution 24-bit phase accumulator and an optimized quantized lookup table, the synthesizer minimizes phase truncation errors and spurious frequency components. The modular architecture, including pre- and post-LUT multiplexers, ensures that waveform symmetry and amplitude accuracy are preserved across the entire frequency range. These measures are essential for maintaining low phase noise and high signal quality, which are critical for applications requiring precise and clean frequency generation.

Moreover, the FPGA implementation provides significant advantages in terms of flexibility and real-time performance. Utilizing the parallel processing capabilities of the FPGA, the system can perform rapid frequency switching by simply updating the Frequency Control Word without halting the phase accumulator. This feature is particularly beneficial for dynamic applications such as software-defined radio and communication systems where quick adaptation to changing frequency requirements is necessary. The use of VHDL for hardware description, combined with the Vivado synthesis tools, facilitates efficient resource utilization and timing optimization, ensuring that the design meets both performance and hardware constraints.

IV. SIMULATION RESULTS

The functionality of the 24-bit Digital Direct Frequency Synthesizer (DDFS) was verified through simulations in Xilinx Vivado by comparing the theoretical frequencies derived from the Frequency Control Word (FCW) with the output waveforms observed during simulation. The reference clock frequency used was 5 MHz, and the phase accumulator had a width of 24 bits. The theoretical output frequency was calculated using the formula: Fout= (FCW×fclk)/2^24

In the first test case, the frequency selection input (fre_add) was set to 00001. The Freq_Reg module mapped this input to an FCW value of 120,796. Using the formula, the theoretical output frequency was calculated as 36 kHz. The simulation results confirmed that the phase accumulator incremented correctly, and the Q_LUT module produced a clean and symmetric sine waveform at 36 kHz. The waveform exhibited the correct period, amplitude, and shape, validating both the accuracy of frequency generation and the proper functioning of the phase accumulator, pre/post LUT logic, and the quantized sine output path.

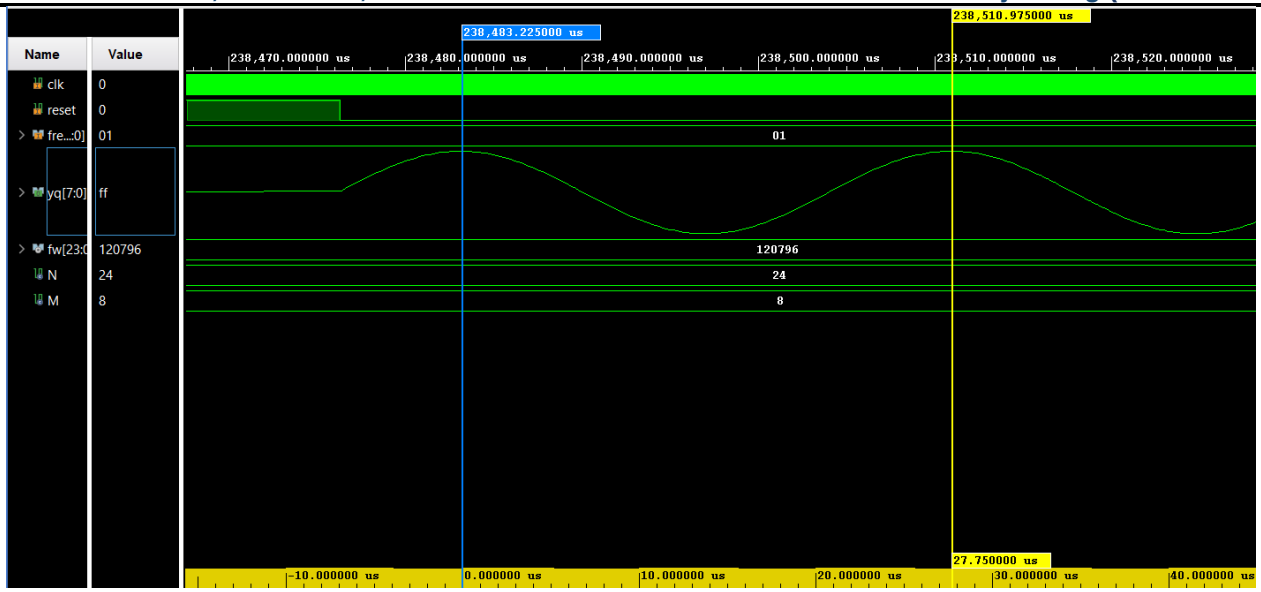

Fig-4.1 Simulation Result for 36 KHz waveform generation

In the second test case, the frequency selection input was set to 01000, which generated an FCW value of 144,284 from the Freq_Reg module. This corresponded to a theoretical output frequency of 43 kHz. The simulated output waveform confirmed this result by displaying a sine wave that cycled faster than in the previous case, as expected for a higher FCW. The waveform maintained proper amplitude, symmetry, and resolution across the higher frequency, with no distortion. This confirmed the synthesizer's ability to scale frequency accurately with different FCW values and demonstrated reliable performance at the upper end of the target frequency range.

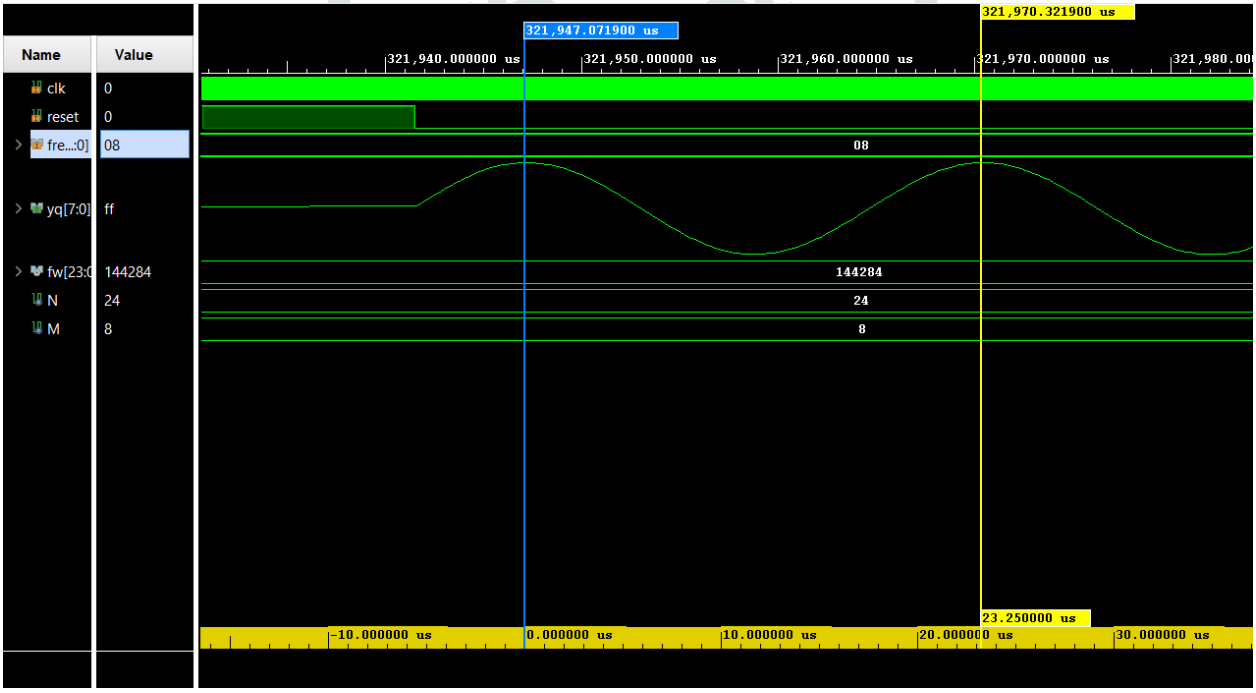

Fig-4.2 Simulation Result for 43 KHz waveform generation

The simulation also included dynamic testing, where the fre_add input was changed during runtime. The synthesizer transitioned between frequency values smoothly and without glitches, confirming the robustness of the design under varying conditions. These results validate the correctness of the DDFS design and its suitability for applications that require fine frequency control and fast switching.

V. CONCLUSION AND FUTURE SCOPE

The 24-bit Direct Digital Frequency Synthesizer (DDFS) was successfully designed, implemented using VHDL, and functionally verified through simulation in Xilinx Vivado. The top-level module, DDFS_QLUT, effectively integrated the Frequency Control Word Generator (Freq_Reg), Phase Accumulator, quantized sine lookup table (Q_LUT), and associated multiplexers in a modular architecture that ensured efficient use of FPGA resources. Operating with a 5 MHz clock, the synthesizer accurately generated frequencies within the 35 kHz to 45 kHz range, with simulation results confirming that the output waveforms matched theoretical calculations in both amplitude and timing. The design demonstrated correct behavior across key components, including asynchronous reset, phase accumulation, and frequency switching, validating the effectiveness of the LUT-based architecture. Despite the absence of an automated testbench, the functional accuracy and stability were verified through

manual simulation, achieving all design goals and establishing the synthesizer as a viable solution for precision frequency generation in digital systems.

While the present implementation successfully demonstrates accurate sine wave synthesis, several enhancements can be proposed to extend its functionality and performance. Expanding the frequency range—either by increasing the system clock or refining the FCW resolution—would allow the design to support both low-frequency and high-frequency applications. Improving amplitude resolution by increasing the LUT output from 8-bit to 10 or 12-bit would reduce quantization noise and enhance spectral quality. Memory optimization techniques, such as compressed LUT storage or CORDIC-based waveform generation, could significantly reduce resource consumption on FPGA platforms. On the verification side, integrating a fully automated VHDL testbench with self-checking mechanisms would enable systematic and repeatable testing. Future work could also focus on real-time FPGA implementation for hardware-based validation, analyzing power usage and performance under real-world conditions. Additionally, extending the architecture to support multi-channel output with programmable phase offsets could make the system more suitable for complex applications like beamforming and multi-tone signal generation. These improvements would enhance the design's scalability, efficiency, and readiness for deployment in advanced digital communication and signal processing systems.

VI. REFERENCES

[1] C. Shan, Z. Chen, H. Yuan, and W. Hu, *"Design of Direct Digital Frequency Synthesizer Based on FPGA,"* in Proc. 2011 Int. Conf. Electrical and Control Engineering (ICECE), Yichang, China, Sep. 16–18, 2011, pp. 4974–4977, doi: 10.1109/ICECENG.2011.6057152.

[2] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2006.

[3] M. J. Smith, *Digital Signal Processing: A Practical Guide for Engineers and Scientists*, 2nd ed. Burlington, MA, USA: Newnes, 2003.

[4] M. D. Ciletti, *Advanced Digital Design with the Verilog HDL*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2010.

[5] S. Palnitkar, *Verilog HDL: A Guide to Digital Design and Synthesis*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003.

[6] Xilinx Inc., *Vivado Design Suite User Guide: Designing with IP*, UG896 (v2023.1), 2023.

[7] Xilinx Inc., *7 Series FPGAs Data Sheet: Overview*, DS180, 2022.

[8] R. Lyons, *Understanding Digital Signal Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2010.

[9] T. K. Saha, *Direct Digital Frequency Synthesis: Theory, Design and Applications*, New York, NY, USA: Springer, 2014.

[10] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*, New York, NY, USA: Wiley, 1999.

[11] IEEE Standard 1076-2008, *IEEE Standard VHDL Language Reference Manual*, IEEE, 2008.

[12] Xilinx Inc., *Digital Frequency Synthesis Using FPGA Application Note*, XAPP125, 2018.

[13] R. E. Best, *Phase-Locked Loops: Design, Simulation, and Applications*, 6th ed. New York, NY, USA: McGraw-Hill, 2007.

[14] B. P. Lathi and Z. Ding, *Modern Digital and Analog Communication Systems*, 4th ed. Oxford, UK: Oxford University Press, 2009.

[15] C. K. Alexander and M. N. O. Sadiku, *Fundamentals of Electric Circuits*, 6th ed. New York, NY, USA: McGraw-Hill, 2016.

[16] R. J. Baker, *CMOS: Circuit Design, Layout, and Simulation*, 3rd ed. Hoboken, NJ, USA: Wiley-IEEE Press, 2010.

[17] S. Gupta and R. K. Sharma, *FPGA-Based Digital Signal Processing and Communication Systems*, New York, NY, USA: Springer, 2018.

[18] Xilinx Inc., *FPGA and CPLD Solutions Data Sheet*, DS200, 2020.

[19] J. W. Cooley and J. W. Tukey, *"An Algorithm for the Machine Calculation of Complex Fourier Series,"* Math. Comput., vol. 19, no. 90, pp. 297–301, Apr. 1965.

[20] W. Kester, *Analog-Digital Conversion Handbook*, 4th ed. Boston, MA, USA: Newnes, 2010.

[21] M. S. Mahmoud and F. M. Ghannouchi, *High Performance Frequency Synthesis Techniques for RF and Microwave Applications*, New York, NY, USA: Springer, 2015.

[22] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*, 2nd ed. Hoboken, NJ, USA: Wiley, 2008.

[23] M. M. Rashid, *FPGA-Based Implementation of Digital Systems: Theory and Practice*, New York, NY, USA: Wiley, 2016.

[24] J. V. Hatfield and P. J. Hurst, *Direct Digital Synthesizers: Fundamentals, Architectures, and Applications*, Boston, MA, USA: Artech House, 2005.