International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 11, November 2026
Received: 29-09-2026 Accepted: 03-11-2026 Published: 09-11-2026

Design of a Power-Efficient Reconfigurable FFT Architecture Using Scaling-Free CORDIC and Flip-Flop-Based Clock Gating

Akkidasari Rohith Lokesh Dath[1], M. Lavanya[2]

[1]*PG Scholar, Department of Electronics and Communication Engineering, JNTUA College of Engineering, Anantapur, Andhra Pradesh, India, binu.akkidasari11@gmail.com*

[2]*Assistant Professor, Department of Electronics and Communication Engineering, JNTUA College of Engineering, Anantapur, Andhra Pradesh, India, lavanyakagiya123@gmail.com*

***Abstract*—** The Fast Fourier Transform (FFT) is a basic building block of digital signal processing, yet its hardware versions are costly in power and area because of complex multipliers and always-running clock networks. This paper proposes an Adaptive Low-Power Reconfigurable FFT (ALPRFA) architecture that combines scaling-free CORDIC (Coordinate Rotation Digital Computer) computation with flip-flop-based clock gating. The scaling-free CORDIC generates twiddle factors with shift-and-add operations only, with no explicit multiplier and no scaling stage. Clock gating switches off idle sequential elements, which lowers dynamic power. The design is written in Verilog HDL and implemented on an Artix-7 FPGA for 4-, 8- and 16-point transforms. Compared with conventional designs, it reduces power by 25.4%, 34.1% and 28.8%, respectively, at an unchanged delay and an area overhead of a single LUT.

***Keywords*—** FFT, Scaling-Free CORDIC, Reconfigurable Architecture, Low-Power VLSI, Flip-Flop-Based Clock Gating, Dynamic Power Reduction, FPGA.

I. Introduction

Every wireless receiver, radar and biomedical monitor has to answer the same question again and again: which frequencies make up this signal? The Fast Fourier Transform (FFT) answers it efficiently and has become a core primitive of digital signal processing (DSP). It is used in software-defined radio, OFDM communication, radar, speech and image processing, and biomedical signal analysis. The FFT computes the Discrete Fourier Transform (DFT) in $O(N \log_2 N)$ operations instead of $O(N^2)$, which makes real-time spectral analysis practical.

The DFT converts a discrete-time signal from the time domain to the frequency domain. The FFT splits this computation into small butterfly operations that need far fewer multiplications and additions than a direct DFT.

Hardware implementation remains difficult, however. Each butterfly multiplies by a complex twiddle factor, and these multiplications account for a large share of the hardware complexity, power and processing time. Conventional FFT processors therefore carry dedicated complex multipliers, and the resulting area and power are high.

CORDIC reduces this cost. It is an iterative method that performs vector rotations using only shifts and additions, so no hardware multiplier is needed. Its main drawback is that conventional CORDIC introduces a scale factor at every micro-rotation, which must be removed by an extra compensation stage that adds hardware and delay.

This paper proposes an Adaptive Low-Power Reconfigurable FFT Architecture using scaling-free CORDIC computation with flip-flop-based clock gating. The contributions are fourfold. First, a fourth-order Taylor series approximation removes the scaling operations completely. Second, an improved shift-value prediction method reduces the number of CORDIC iterations. Third, a pre-rotation mechanism widens the angle of convergence. Fourth, flip-flop-based clock gating cuts dynamic power in idle stages. The design is described in Verilog HDL and implemented on an Artix-7 FPGA.

II. Literature Review

Research on FFT and CORDIC architectures has pursued higher throughput, lower power and smaller area. Twelve relevant works are reviewed below to identify the gap that the proposed design addresses.

In [1] proposed a low-latency 64-parallel, 4096-point memory-based FFT for 6G systems. Parallel datapaths and optimised memory scheduling give excellent throughput, but the 64-fold parallelism raises logic use and routing congestion, so the design suits only resource-rich platforms.

Introduced [2] an approximate low-power multiplier based on compression of the partial-product tree. It shortens critical paths and switching activity at a small loss of accuracy. Multiplier-based designs still cost more area and power than shift-and-add methods, and the rounding error may not be acceptable in high-precision FFTs.

Designed a mixed-radix, virtually scaling-free CORDIC rotator for DSP [3]. Mixing radices reduces the number of micro-rotations and the scaling effect, but the extra control logic makes the design harder to verify.

In [4] built a reconfigurable CORDIC processor that evaluates trigonometric, hyperbolic and arithmetic functions

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 11, November 2026
Received: 29-09-2026 Accepted: 03-11-2026 Published: 09-11-2026

on one platform. Run-time reconfiguration saves resources, but the added configuration registers, multiplexers and state machines increase the gate count and pipeline depth.

In [5] presented a scaling-free folded hyperbolic CORDIC based on a power-of-2 Taylor series approximation. Folding shares hardware across passes, and the approximation removes the scaling stage, which saves area and power. The convergence range stays small, so extra domain-mapping logic is needed for wider angles.

In proposed [6] RECON, a resource-efficient CORDIC-based neuron architecture. Computing activation functions by shift-and-add lowers area and power compared with look-up tables or multipliers, but the design targets neural workloads and does not address FFT twiddle factors or clock gating.

The developed a CORDIC-based digital hardware model of an adaptive[7] exponential integrate-and-fire neuron. Iterative CORDIC evaluates the required non-linear functions efficiently, but the iteration delay limits the maximum firing rate.

The carried out a fixed-point analysis of Multi-Scale Rotation (MSR) CORDIC and tuned its parameters for FFT use [8]. Careful choice of word length, iteration count and scaling improve FFT accuracy with little hardware overhead. The study is limited to fixed-point designs and ignores power optimisation.

In [9] proposed a hybrid CORDIC with higher-radix stages that lowers latency. Large-angle rotations are used first and small-angle rotations later, which speeds up convergence, but angle selection becomes more complex and extra hardware is required.

The proposed a low-power, memory-based FFT processor for communication systems [10]. Reusing butterfly units and memory banks across stages saves area and power compared with fully parallel pipelines. Memory-access latency grows with FFT size, and advanced clock gating is not used.

In [11] described a feed-forward FFT that stores twiddle factors in look-up tables. Precomputed sine and cosine values raise throughput, but the table memory grows with FFT size, so the approach suits only FPGAs with abundant on-chip memory.

It presented a scaling-free CORDIC (NSF-CORDIC) for memory-based FFT[12]. A fourth-order Taylor series approximation with shift-value prediction removes scaling, reduces iterations and extends the angle coverage, and it lowers slice use, slice-delay product and power. Precomputed shift values in ROM and extra control logic, however, raise complexity and limit flexibility when the FFT size changes.

A. Research Gap

Earlier work improves FFT throughput, area or CORDIC computation, but no single design combines four measures: removing scaling-factor compensation through a higher-order approximation, reducing iterations through shift-value prediction, extending the convergence angle through pre-rotation, and cutting dynamic power through flip-flop-based clock gating. The proposed NSF-CORDIC-based memory FFT with flip-flop clock gating brings all four together in one design.

III. METHODOLOGY

A. Existing Method

Typical FFT hardware uses dedicated complex multipliers for the twiddle-factor rotations. Each complex multiplication needs four real multiplications and two real additions, which consume many FPGA DSP slices and generate significant dynamic power. Memory-based FFT architectures reuse one butterfly processor across all stages, which saves area at the cost of more computation time. A dual-port SRAM bank supplies the correct pair of samples to the processor according to the butterfly schedule.

The existing NSF-CORDIC method replaces the multiplier with shift-and-add micro-rotations. A full rotation is divided into a sequence of elementary rotations, each performed by barrel shifters and adders. The iteration uses fourth-order small-angle approximations of sine and cosine:

$$x_{i+1} = x_i\left[1 - 2^{-(2i+1)} + 2^{-(4i+4)}\right] - y_i\left[2^{-i} - 2^{-(3i+2)}\right] \quad -(1)$$

$$y_{i+1} = x_i\left[2^{-i} - 2^{-(3i+2)}\right] + y_i\left[1 - 2^{-(2i+1)} + 2^{-(4i+4)}\right] \quad -(2)$$

Equations (1) and (2) map onto two hardware units: Unit-I computes the x-coordinate and Unit-II computes the y-coordinate (Fig. 1). Each unit has four-barrel shifters (BS) and four adders working in parallel for high throughput. With a 16-bit word length, the iteration index i runs from 1 to 9. As i grows, the shift amounts 2i+1, 4i+4, i and 3i+2 exceed the 16-bit precision, so the corresponding terms become zero after fixed-point synthesis and their hardware can be removed. The removable elements are BS2 and Add2 (Unit-I) and BS7 and Add7 (Unit-II) for i > 3; BS4 and Add3 (Unit-I) and BS6 and Add5 (Unit-II) for i > 5; and BS1 and Add1 (Unit-I) and BS7 and Add6 (Unit-II) for i > 8.

The depth of the adder chain sets the maximum delay of the NSF-CORDIC unit, excluding registers. Barrel shifters are mapped to FPGA LUTs and add little propagation delay, so the adder stages dominate the critical path. Because the number of active adders falls as the iteration count rises, a carefully configured design shortens the critical path.

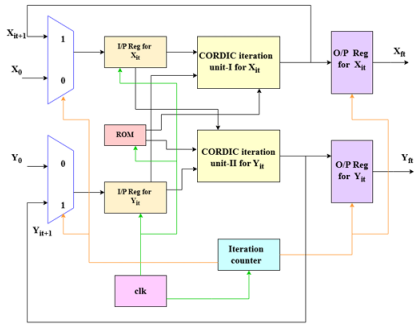


Fig. 1 Architecture of the NSF-CORDIC iteration units I and II

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 11, November 2026
Received: 29-09-2026 Accepted: 03-11-2026 Published: 09-11-2026

B. Proposed Method

The proposed Adaptive Low-Power Reconfigurable FFT consists of an input buffer, adaptive control logic, a scaling-free CORDIC module, FFT butterfly processing blocks, memory storage elements and clock-gated sequential circuits (Fig. 3).

1) Scaling-Free CORDIC Twiddle-Factor Generation: The scaling-free CORDIC module generates the twiddle factors $W_N^k = cos\left(\frac{2\pi k}{N}\right) + j \cdot sin\left(\frac{2\pi k}{N}\right)$ at each butterfly stage, so no ROM coefficient table is needed. The module takes the rotation angle as input and returns the pair (cos θ, sin θ) from Equations (1) and (2). A fourth-order Taylor series approximation gives the required 16-bit accuracy at low hardware cost, so no separate scaling stage is needed (Fig. 2).

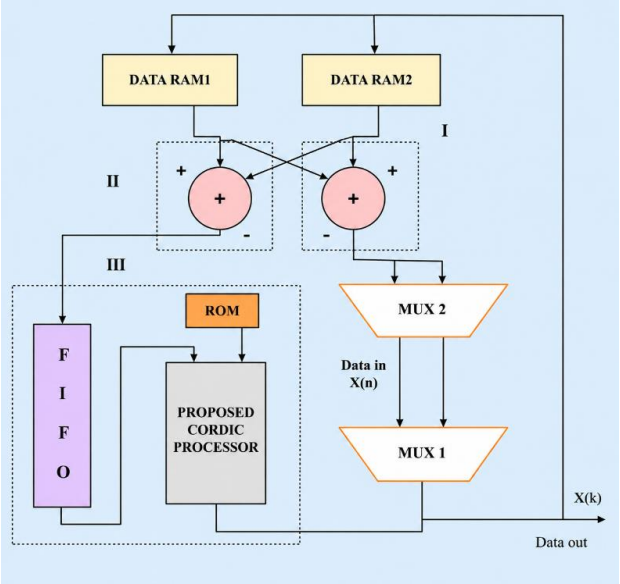


Fig. 2 Architecture of the new scaling-free CORDIC cell

2) Pre-Rotation: The pre-rotation stage uses trigonometric identities to bring any angle into the convergence range. For an angle in the second quadrant, the input is shifted by $-\pi/2$ before the CORDIC iteration and the result is shifted back by $+\pi/2$ afterwards. This allows the NSF-CORDIC to produce all twiddle factors over the full range [0, 2π] without extra correction hardware.

3) Flip-Flop-Based Clock Gating: The dynamic power of a synchronous circuit is $P = \alpha \cdot C \cdot V^2 \cdot f$, where C is the switched capacitance, V the supply voltage, f the clock frequency and α the activity factor. In conventional FFT designs, α stays high because the clock drives every register all the time. In the proposed design, a gating cell between the global clock and each register bank lowers α. The cell is an AND gate preceded by a level-sensitive latch, and it produces the gated clock CK_G from the latched enable and the global clock. The latch prevents glitches when the enable signal changes while the clock is high.

Each FFT block has its own enable signal. While the enable is asserted, the clock reaches the registers; when it is deasserted, the gated clock stays low and the registers hold their values. This removes flip-flop toggling in idle stages, which lowers the switching capacitance per cycle. As a result, the dynamic power of each stage falls in proportion to the time the stage is idle.

4) Butterfly Array and Adaptive Control: The butterfly array performs memory reads and writes together with butterfly computations that use the twiddle factors from the NSF-CORDIC module. An N-point FFT needs $log_2 N$ stages, which gives 3, 4, 5 and 6 stages for 8-, 16-, 32- and 64-point transforms. For the selected size, the adaptive FSM controller sets the memory addressing pattern, the twiddle-factor angle sequence and the clock-gate enables, and it keeps the gates of unused stages deasserted. The same hardware can therefore process different FFT sizes without structural change.

5) Implementation: The design is written in Verilog HDL, targeted at the Artix-7 XC7A100T FPGA and synthesised with Xilinx Vivado 2023.1. Timing closure is achieved at 125 MHz with a 7.95 ns critical path. Post-place-and-route power is estimated with the Vivado Power Analyzer using switching activity from post-synthesis simulation.

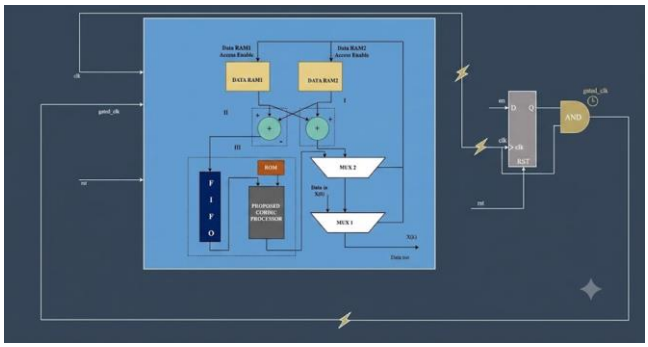


Fig. 3 Memory-based FFT architecture using flip-flop-based clock gating

IV. Results and Discussion

The proposed architecture was simulated and synthesised in Xilinx Vivado. Functional correctness, area, timing and power were evaluated and compared with the conventional design. Fig. 4 shows the schematic of the 8-point design, which combines the reconfigurable FFT units, the scaling-free CORDIC twiddle-factor generator and the clock-gated flip-flops.

2250-3676 ijesatj@gmail.com https://www.ijesat.com/

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 11, November 2026
Received: 29-09-2026 Accepted: 03-11-2026 Published: 09-11-2026

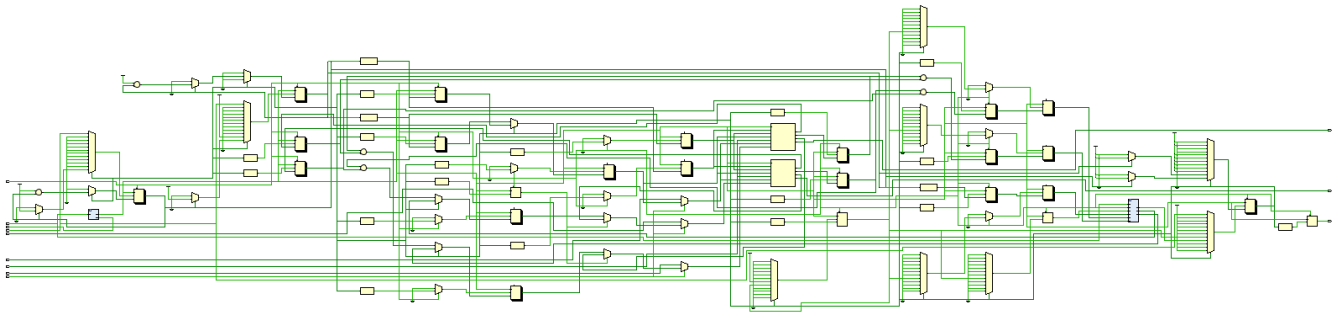


Fig. 4 Schematic diagram of the proposed adaptive low-power reconfigurable FFT architecture

A. Simulation Results

Fig. 5 shows the simulation waveform. After the input data is loaded, the design computes the FFT with the scaling-free CORDIC and produces the real and imaginary outputs, which confirms functional correctness. The waveform also shows that the clock gating works as intended and suppresses unnecessary switching activity.

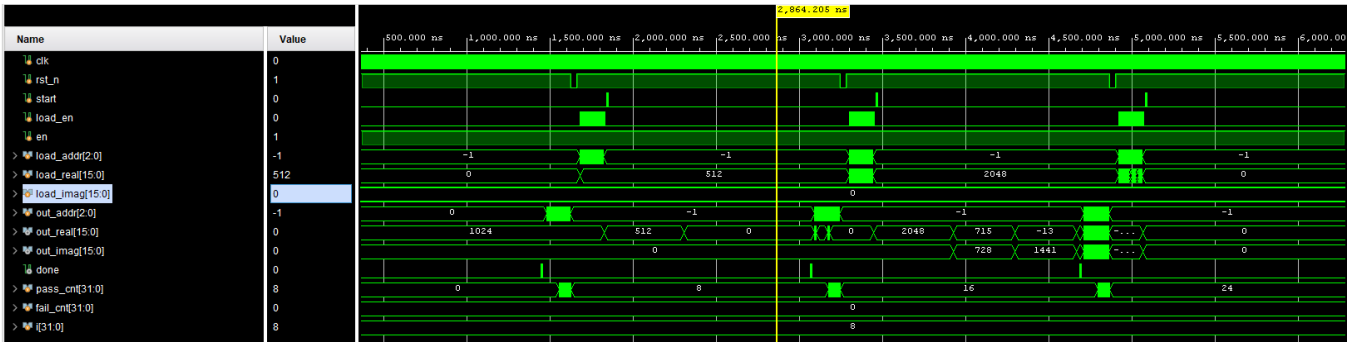


Fig. 5 Simulation waveform of the proposed adaptive low-power reconfigurable FFT

B. Area Utilization

Fig. 6 shows the synthesis result for the 8-point design. It uses 678 of the 134,600 LUTs available on the device. The NSF-CORDIC module accounts for 491 of these LUTs, so the design is compact and suitable for resource-constrained DSP applications.

Name	Slice LUTs (134600)
nsf_fft8_memory_extension	678
s10 (Flipflop_Clk)	0
u_cordic (nsf_cordic_cell)	491

Fig. 6 FPGA resource utilization of the proposed adaptive low-power reconfigurable FFT

C. Timing Analysis

Fig. 7 shows the timing report. The maximum critical path delay is 7.95 ns with 13 logic levels, of which 4.62 ns is routing delay and 3.33 ns is logic delay. The design therefore meets the 125 MHz target on the selected device.

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay
Path 1	∞	13	10	112	u_cordic/iter_cnt_reg[0]/C	u_cordic/y_out_reg[15]/D	7.950	3.333	4.617
Path 2	∞	13	10	112	u_cordic/iter_cnt_reg[0]/C	u_cordic/y_out_reg[12]/D	7.943	3.400	4.543
Path 3	∞	13	10	112	u_cordic/iter_cnt_reg[0]/C	u_cordic/y_out_reg[14]/D	7.905	3.397	4.508

Fig. 7 Timing analysis of the proposed adaptive low-power reconfigurable FFT

2250-3676 ijesatj@gmail.com https://www.ijesat.com/

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 11, November 2026
Received: 29-09-2026 Accepted: 03-11-2026 Published: 09-11-2026

D. Power Analysis

Fig. 8 shows the power report. Flip-flop clock gating reduces switching activity and dynamic power, and the scaling-free CORDIC removes the power drawn by multipliers, so the design is energy efficient.

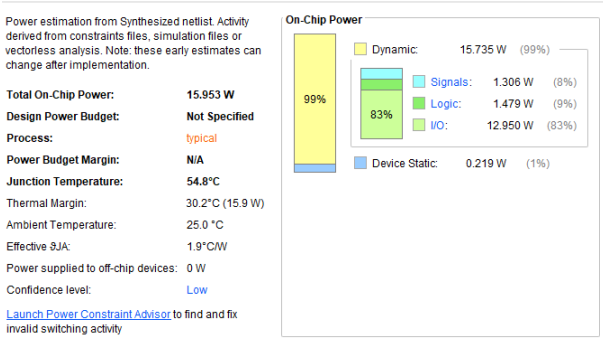


Fig. 8 Power analysis of the proposed adaptive low-power reconfigurable FFT

E. Comparison with the Conventional Design

Table I compares the conventional FFT with the proposed flip-flop clock-gated FFT for 4, 8 and 16 points. Power falls from 19.52 mW to 14.561 mW (25.4%) for 4 points, from 24.206 mW to 15.953 mW (34.1%) for 8 points, and from 23.786 mW to 16.938 mW (28.8%) for 16 points. The delay is identical in every case, and the area overhead is a single LUT. The average power saving is about 29.4%.

TABLE I
POWER, AREA AND DELAY COMPARISON OF CONVENTIONAL AND CLOCK-GATED FFT

FFT Size	Design	Area (LUTs)	Power (mW)	Delay (ns)	Power Saving
4-point	Conventional	261	19.52	4.652	—
	Proposed	262	14.561	4.652	25.4%
8-point	Conventional	677	24.206	7.950	—
	Proposed	678	15.953	7.950	34.1%
16-point	Conventional	657	23.786	8.108	—
	Proposed	658	16.938	8.108	28.8%

V. CONCLUSIONS

This paper presented an Adaptive Low-Power Reconfigurable FFT architecture that combines scaling-free CORDIC twiddle-factor computation with flip-flop-based clock gating. Replacing complex multipliers and scaling-compensation circuits with shift-and-add logic reduces hardware complexity, area and delay. Gating the clock of idle stages reduces switching activity and cuts dynamic power by 25.4% to 34.1% for 4-, 8- and 16-point FFTs, with unchanged delay and an area overhead of one LUT. The architecture suits FPGA and ASIC implementations in wireless communication, OFDM systems, image processing, biomedical signal analysis, radar and IoT devices. Future work can add adaptive voltage scaling, power gating and machine-learning-based power management.

ACKNOWLEDGMENT

The authors thank the Department of Electronics and Communication Engineering, JNTUA College of Engineering, Anantapur, for providing the resources and support for this research, and the researchers whose earlier work laid the foundation for this study.

REFERENCES

[1] Z. Kaya and M. Garrido, "Low-latency 64-parallel 4096-point memory based FFT for 6G," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 70, no. 10, pp. 4004–4014, Oct. 2023.
[2] S. Shayanki and C. Paramasivam, "Low-power high-speed approximate multiplier design using compression techniques," in *Micro-Electronics and Telecommunication Engineering*, vol. 617. Singapore: Springer, 2023, pp. 273–285.
[3] A. Changela, M. Zaveri, and D. Verma, "Mixed-radix, virtually scaling free CORDIC algorithm-based rotator for DSP applications," *Integration*, vol. 78, pp. 70–83, May 2021.
[4] S. Aggarwal, P. K. Meher, and K. Khare, "Concept, design, and implementation of reconfigurable CORDIC," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 4, pp. 1588–1592, Apr. 2016.
[5] A. Verma, K. Kiyawat, B. P. Das, and P. K. Meher, "An efficient scaling-free folded hyperbolic CORDIC design using a novel low complexity power-of-2 Taylor series approximation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 31, no. 8, pp. 1167–1177, Aug. 2023.
[6] G. Raut, S. Rai, S. K. Vishvakarma, and A. Kumar, "RECON: Resource efficient CORDIC-based neuron architecture," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 170–181, 2021.
[7] M. Heidarpour, A. Ahmadi, and R. Rashidzadeh, "A CORDIC based digital hardware for adaptive exponential integrate and fire neuron," 2016.
[8] S. Y. Park and Y. J. Yu, "Fixed-point analysis and parameter selections of MSR-CORDIC with applications to FFT designs," 2012.
[9] R. Shukla and K. C. Ray, "Low latency hybrid CORDIC algorithm," 2014.
[10] V. Vinod et al., "Development of a communication system by a high productivity, low power consumption, and memory-based architecture incorporating an FFT processor," 2024.
[11] Y. Harshitha and C. Paramasivam, "Design of low complexity high performance LUT based feed-forward FFT architecture," 2020.
[12] M. Ayinala, M. Brown, and K. K. Parhi, "Pipelined parallel FFT architectures via folding transformation," 2012.