IJETIR

ISSN (Online) 2583-0554

International Journal of Emerging Technologies and Innovative Research (IJETIR)

Volume 6, Issue 8, August 2026

Impact Factor: 5.731

Design and Characterization of Carry Chain-Based Multistage Ring Oscillators on FPGA for Predictable and Tunable Performance

N. Samuel Babu[1] and Chittula Ruchitha[2]

Assistant Professor, ECE Department[1]

PG Scholar, ECE Department[2]

Annamacharya Institute of Technology & Sciences-Hyderabad, Telangana, India

nsamuelbabu460@gmail.com and chittularuchitha78@gmail.com

Abstract: *Ring oscillators are widely employed in hardware security, true random number generation, sensing, and on-chip timing characterization, but conventional FPGA implementations built from lookup-table inverter chains are strongly affected by programmable routing. This paper presents a carry chain-based multistage ring oscillator methodology that repurposes the dedicated fast carry paths of an FPGA as deterministic delay elements. The basic CC_ROv1 architecture uses XOR and multiplexer resources within the carry chain together with a NAND-configured LUT to establish an odd-inversion feedback loop. Enhanced CC_ROv2 and CC_ROv3 configurations introduce pass-through LUT delay elements for finer control of oscillation frequency, energy behavior, and environmental sensitivity. The structured carry path supports scalable stage count, multiphase signal extraction, and analytical delay decomposition while reducing dependence on arbitrary place-and-route decisions. The supplied design study reports qualitative reductions in area and energy relative to conventional LUT-based oscillators and improved predictability, stability, and configurability. Because the source document does not provide a numerical device-level results table, this manuscript reports only source-supported qualitative characterization rather than inventing frequency, power, or utilization measurements. The architecture is particularly applicable to FPGA-based PUFs, TRNGs, voltage/temperature sensing, aging monitoring, and compact embedded systems.*

Keywords: ring oscillator, FPGA, carry chain, CARRY4, PUF, TRNG, delay modeling, hardware security, sensing, tunable oscillator.

I. INTRODUCTION

Ring oscillators (ROs) are compact feedback circuits capable of generating periodic signals without an external clock. Their oscillation period is determined by the cumulative propagation delay of an odd-inversion loop, making them useful not only as digital oscillators but also as sensitive probes of process, voltage, temperature, and aging variation. FPGA implementations are attractive because they allow rapid replication of multiple oscillators for applications such as physically unclonable functions (PUFs), true random number generators (TRNGs), and on-chip monitors [1]– [10].

A conventional FPGA RO typically maps each inversion stage to a lookup table (LUT) and relies on programmable interconnect to close the loop. Although this structure is easy to synthesize, its timing is heavily influenced by place-and-route decisions. Identical RTL descriptions can therefore acquire different routing lengths and propagation delays across implementation runs, which weakens repeatability and complicates frequency matching. The supplied design study identifies routing variability, inefficient LUT consumption, coarse frequency tuning, and uncontrolled environmental sensitivity as the principal limitations of the conventional approach.

The proposed methodology addresses these limitations by exploiting dedicated carry-chain resources. Modern FPGA carry chains consist of tightly coupled multiplexers and XOR elements with fixed local routing intended for high-speed arithmetic. By repurposing these structures as a deterministic signal-propagation path, the oscillator can reduce

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

dependence on general-purpose routing while retaining the programmability of the FPGA fabric. The resulting framework consists of a base multistage design (CC_ROv1) and two tunable variants (CC_ROv2 and CC_ROv3).

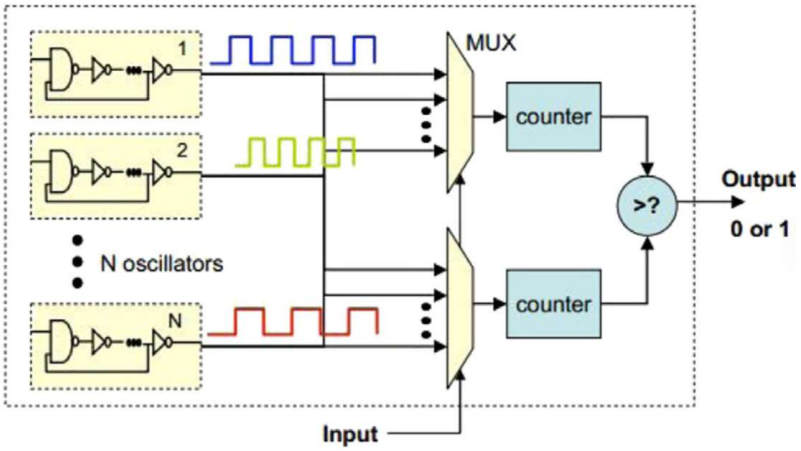

Fig. 1. Multi-oscillator frequency comparison architecture illustrating a representative PUF/TRNG use case.

II. BACKGROUND AND RELATED WORK

2.1 Conventional LUT-Based FPGA Ring Oscillators

The basic RO condition requires an odd number of effective inversions and sufficient loop delay for a transition to propagate around the feedback path. The source material expresses the oscillation frequency as the inverse of twice the loop propagation delay:

$$f_{RO} = \frac{1}{(2\tau_{loop})} \qquad (1)$$

For a LUT-oriented realization, the loop delay is composed of logic delay and routing delay. A compact representation of the relationship shown in the supplied LUT-based architecture is:

$$\tau_{loop} = \tau_{logic} + \tau_{net} \qquad (2)$$

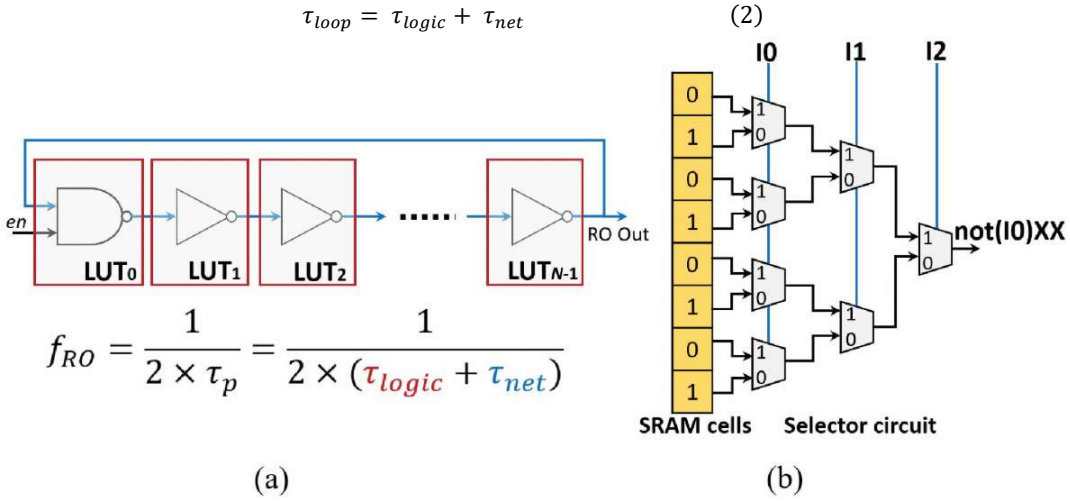

Fig. 2. Conventional LUT-based ring oscillator and SRAM-configured LUT structure from the supplied design study.

Because τ_net is selected by the programmable interconnect and the place-and-route process, it is not inherently deterministic. The source survey also notes that LUT-based oscillators consume one LUT per inversion stage and additional routing resources, and that frequency tuning is normally performed by changing the number of stages, resulting in relatively coarse control.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

2.2 Dedicated FPGA Resources for Oscillator Construction

Alternative FPGA RO architectures have used I/O buffers, DSP blocks, multiplexers, and sequential structures. IOBUF-based oscillators can tune frequency through drive-strength or slew-rate control, whereas DSP- or multiplexer-based loops reduce some routing overhead. However, the supplied literature review describes these approaches as generally limited in scalability, fine-grained delay control, or multistage operation [14]–[17]. These limitations motivate the use of carry chains as a more structured delay fabric.

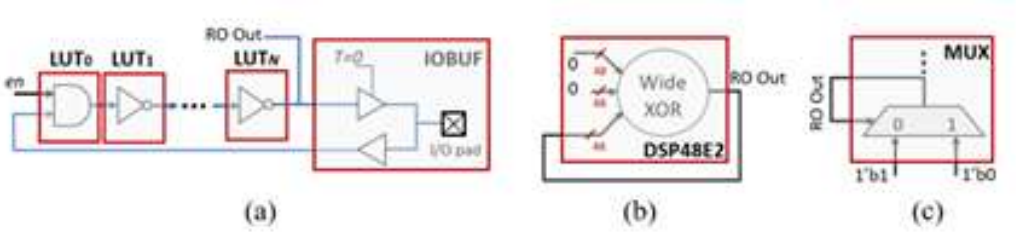

Fig. 3. Alternative FPGA-based oscillator realizations using IOBUF, DSP, and multiplexer resources.

III. PROPOSED CARRY CHAIN-BASED MULTISTAGE RO ARCHITECTURE

3.1 Carry Chain as a Deterministic Delay Element

The internal carry-chain structure contains cascaded multiplexing and XOR logic connected through dedicated local paths. In normal FPGA use, these paths accelerate arithmetic carry propagation. In the proposed oscillator, the same physical structure is used as a predictable sequence of delay and inversion elements. The carry path thereby becomes the dominant propagation route rather than the general-purpose programmable network.

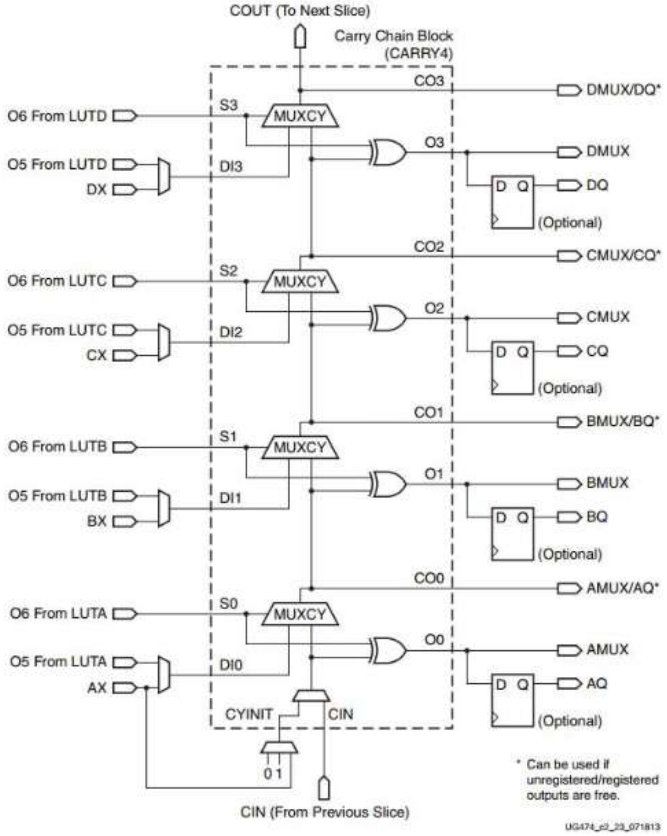

Fig. 4. Internal CARRY4-style carry-chain structure showing the dedicated multiplexer/XOR propagation path.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

3.2 CC_ROv1 Base Architecture

CC_ROv1 is the base scalable configuration. A LUT configured as a NAND function initiates and enables the oscillation loop. The loop then propagates through XOR outputs and carry multiplexers distributed across one or more carry-chain blocks. The source defines the effective number of inversion stages as:

$$N_{inv} = 2N_{CC} + 1 \qquad (3)$$

where N_CC denotes the number of carry-chain units participating in the loop. This structure permits the designer to modify the stage count by cascading additional carry-chain blocks without changing the fundamental routing strategy. Intermediate XOR outputs remain available as phase-shifted taps, enabling multiphase extraction without inserting additional elements into the primary loop.

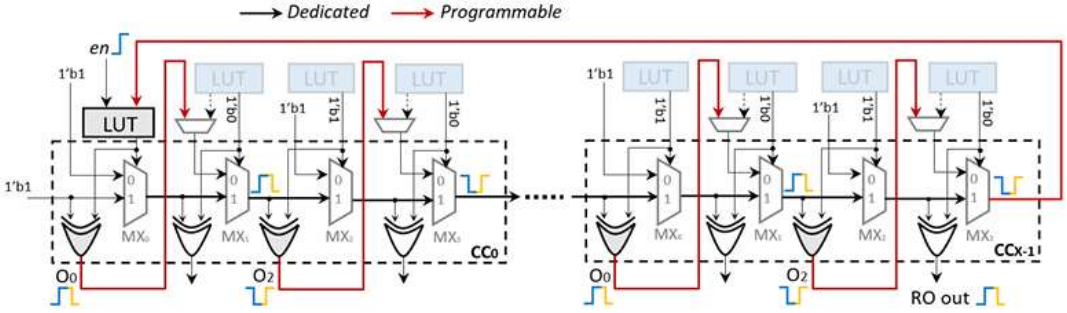

Fig. 5. Proposed CC_ROv1 multistage ring oscillator using dedicated and programmable paths.

3.3 CC_ROv2 and CC_ROv3 Fine-Tuning Variants

The enhanced CC_ROv2 and CC_ROv3 variants insert LUTs configured as pass-through elements into selected propagation segments. These LUTs do not create additional logical inversion; instead, they add controlled delay. The source describes this as a hybrid method that preserves the structured carry-chain path while providing finer adjustment than changing the number of oscillator stages alone.

The design study characterizes CC_ROv2 as the variant oriented toward improved stability with comparatively low environmental sensitivity, while CC_ROv3 intentionally increases the contribution of programmable delay and is therefore more suitable when sensitivity to temperature or voltage is desired. The variants allow a single architectural family to be adapted either for stable timing/frequency comparison or for sensing-oriented operation.

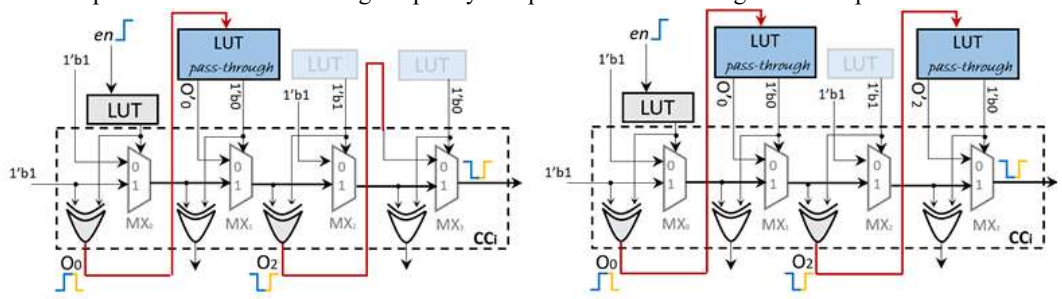

Fig. 6. Enhanced CC_ROv2 and CC_ROv3 configurations with pass-through LUT delay elements for fine tuning.

IV. PROPAGATION DELAY AND FREQUENCY MODELING

The source methodology decomposes the loop into delay contributions associated with LUT access, carry-chain propagation, XOR behavior, multiplexer switching, and residual routing. The key design goal is not to eliminate every programmable component, but to ensure that the dominant part of the oscillator path is carried by dedicated resources whose geometry and routing are substantially more constrained than ordinary LUT-to-LUT interconnect.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

For the tunable variants, pass-through LUTs add a controlled programmable component to the base carry-chain delay. Thus, increasing the number of enabled pass-through elements increases the loop period and lowers oscillation frequency, while bypassing them returns the oscillator toward its faster carry-dominated operating point. The same mechanism also modifies the oscillator response to environmental changes because LUT and carry-chain paths have different delay sensitivities.

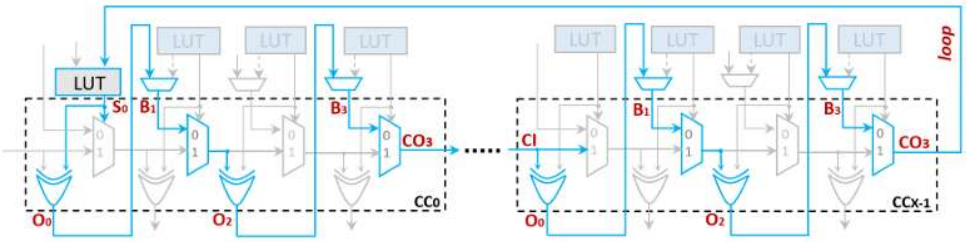

Fig. 7. Propagation path highlighted in the carry chain-based oscillator for delay-component analysis.

This analytical decomposition is valuable because it allows the designer to reason about frequency before final implementation. Unlike a purely LUT-based RO, in which a large portion of total delay may be determined by unconstrained routing, the carry-chain topology makes the dominant delay path structurally visible and repeatable. The source report therefore positions the model as a pre-implementation optimization tool rather than only a post-route measurement method.

V. DESIGN AND CHARACTERIZATION METHODOLOGY

The complete design flow begins with selection of the target oscillator behavior: stable frequency generation, frequency comparison, entropy generation, or environmental sensing. The number of carry-chain blocks is then selected to establish the coarse delay range. A NAND-configured LUT provides enable/control and maintains the odd-inversion condition, after which carry-chain XOR, and multiplexer elements are linked to form the feedback path. When finer tuning is required, CC_ROv2 or CC_ROv3 pass-through LUTs are enabled in selected stages.

Characterization is performed by observing the oscillation frequency at the loop output and, where required, at intermediate phase taps. For security applications, multiple oscillator instances can be counted over a common observation interval and compared. For sensing applications, frequency variation is monitored as temperature, supply voltage, or device aging changes. The study also considers area and energy behavior because reduced LUT use and lower general-purpose routing activity are key motivations for using carry chains.

Stage	Design action	Purpose
1	Select application and target behavior	Determine whether stability, entropy, or environmental sensitivity is prioritized.
2	Choose number of carry-chain units	Set the coarse oscillation-delay/frequency range.
3	Configure NAND enable stage	Provide loop control and odd-inversion condition.
4	Connect XOR/MUX carry path	Create the deterministic multistage propagation loop.
5	Optionally add pass-through LUTs	Fine-tune delay, frequency, energy, and sensitivity.
6	Expose phase taps	Support multiphase measurement, PUF comparison, or entropy extraction.
7	Implement and characterize	Evaluate repeatability, frequency, area, energy, and environmental response.

Table 1. Source-aligned design and characterization flow for the proposed oscillator family.

VI. RESULTS AND DISCUSSION

6.1 Source-Supported Characterization

The supplied document contains a Results heading but does not provide a numerical results table, device-specific frequency measurements, FPGA part number, power report, or utilization summary. Accordingly, numerical values have

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

not been invented in this paper. The result discussion below is limited to the qualitative findings explicitly supported by the source material.

First, the carry-chain implementation is reported to improve frequency predictability and implementation consistency because the dominant propagation path uses dedicated local routing. Second, replacing a large fraction of LUT inverter stages with carry-chain resources reduces LUT and general routing dependence, which the source associates with lower area utilization and lower energy consumption. Third, CC_ROv2 and CC_ROv3 extend the base structure with fine-grained delay tuning, enabling the designer to trade stability against environmental sensitivity. Finally, the availability of intermediate phase outputs broadens the architecture beyond a single clock source and enables phase-based sensing and entropy-generation functions.

Characteristic	LUT-based RO	CC_ROv1	CC_ROv2	CC_ROv3
Dominant routing	General programmable routing	Dedicated carry path	Carry path + selected LUT delay	Carry path + greater programmable delay
Predictability	Low / P&R dependent	High relative to LUT design	High with fine tuning	High with intentionally increased sensitivity
Frequency tuning	Coarse, usually by stage count	Scalable by carry-chain count	Fine-grained pass-through tuning	Fine-grained pass-through tuning
Environmental behavior	Uncontrolled sensitivity	More structurally stable	Source describes low sensitivity	Source describes enhanced sensitivity
Multiphase outputs	Not emphasized	Supported from intermediate XOR nodes	Supported	Supported
Area/energy trend	Higher LUT/routing dependence	Source reports reduction	Tunable trade-off	Tunable trade-off

Table 2. Qualitative comparison supported by the supplied design document; no unreported numerical values are introduced.

6.2 Implications for PUF and TRNG Systems

For RO-PUF operation, the structured carry path is beneficial because frequency differences are intended to reflect device-specific physical variation rather than accidental implementation differences. Multiple oscillator outputs can be selected through multiplexers, counted over a fixed interval, and compared to form a binary response. The same predictable structure can improve repeatability when the design is replicated across a device, while process variation can still provide the physical diversity required by the PUF mechanism.

For TRNG operation, architecture offers intermediate phase outputs that can be sampled to exploit timing jitter and phase uncertainty. The tunable variants also provide a way to modify the oscillator delay and sensitivity without reconstructing the complete loop. The source material therefore identifies PUFs and TRNGs as major application domains for the architecture.

IJETIR

ISSN (Online) 2583-0554

International Journal of Emerging Technologies and Innovative Research (IJETIR)

Volume 6, Issue 8, August 2026

Impact Factor: 5.731

6.3. Architecture and Synthesis Analysis

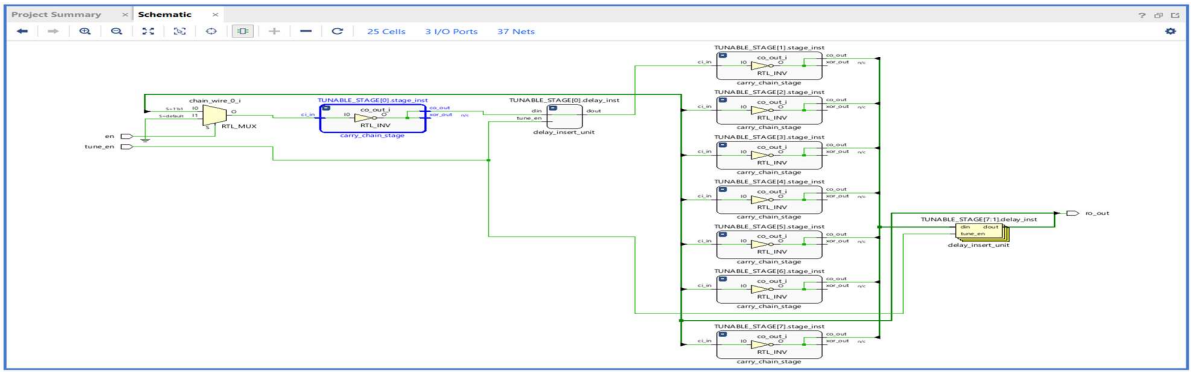

Fig 8. Synthesized schematic of the tunable carry-chain ring oscillator.

The synthesized schematic contains eight indexed TUNABLE_STAGE instances (0 through 7). Each stage contains a carry_chain_stage represented by inverter-like RTL logic, and the stages are interconnected through the chain signal. An RTL multiplexer at the input provides the enable-controlled feedback path. A delay_insert_unit appears within the tunable path and a second delay unit is visible near the final stage/output path; tune_en controls selectable propagation delay. This structure agrees with the project methodology: dedicated carry-chain stages establish the main deterministic path, while selectable delay elements modify the loop delay for tuning.

The stage count is confirmed by the eight indexed stage instances and by NUM_STAGES = 00000008 in the waveform window.

The schematic summary reports 25 cells, 3 I/O ports, and 37 nets, indicating a compact elaborated design.

The feedback topology and delay control are structurally present, but a synthesized schematic alone cannot prove sustained oscillation.

Fig 9. Vivado synthesis completion report for the carry-chain RO design.

Vivado reports that synthesis finished with zero errors and zero critical warnings. Three warnings were issued, but their full text is not visible in the supplied output and must be reviewed before sign-off. The run produced a synthesized netlist and design checkpoint. The report also shows a top-level instance-area entry of three cells; this hierarchy-oriented entry should not be treated as the complete physical resource-utilization result, because the schematic separately reports 25 elaborated cells and no LUT/FF/carry-site utilization table is supplied.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

6.4. Device-Level View

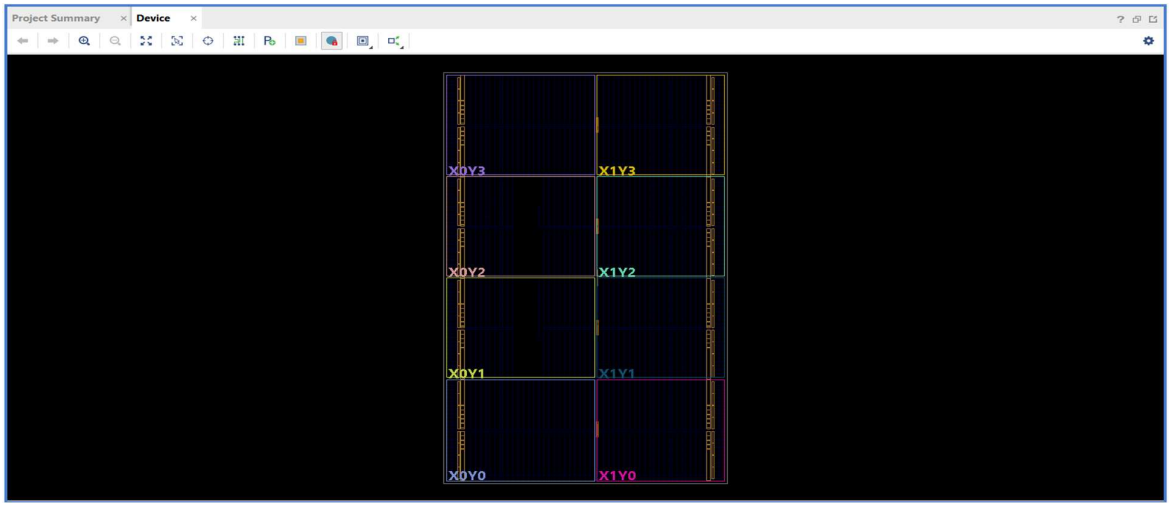

Fig 10. Vivado device view showing the target FPGA clock-region grid.

The device view spans the visible clock regions X0Y0 through X1Y3. It confirms that an implementation target has been opened in the device environment. However, the screenshot does not provide a readable highlighted set of placed oscillator cells, route segments, or a utilization legend. It therefore cannot support quantitative claims about placement compactness, carry-column occupancy, routing length, congestion, or location-to-location repeatability. Those claims require a zoomed placed-and-routed view together with utilization, placement, and timing reports.

6.5. Power and Thermal Results

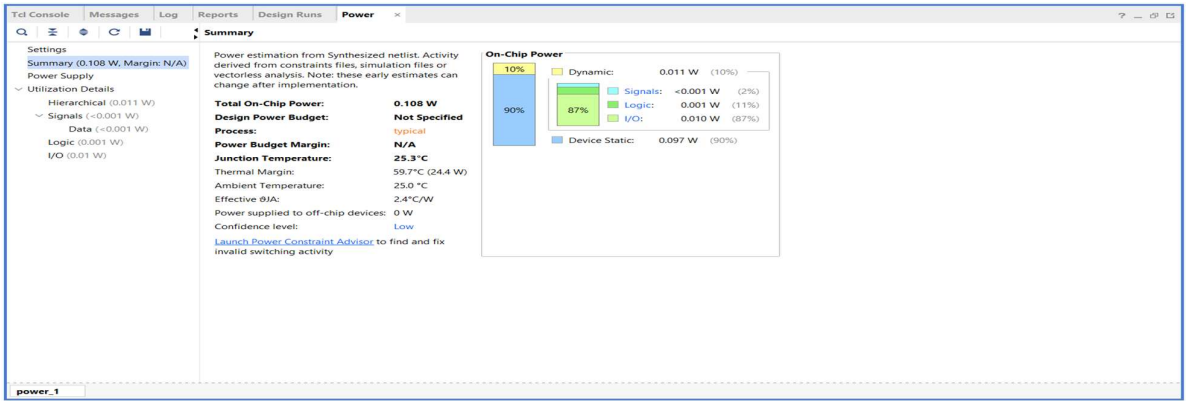

Fig 11. Vivado post-synthesis power-estimation summary.

The total estimated on-chip power is 0.108 W. Device static power dominates at 0.097 W (90%), while dynamic power is 0.011 W (10%). Within the dynamic component, I/O contributes 0.010 W (87% of dynamic power), logic contributes 0.001 W (11%), and signals contribute less than 0.001 W (2%). The result indicates that the core logic has a small, estimated switching-power contribution relative to device leakage and I/O activity.

The junction-temperature estimate is 25.3 °C for a 25.0 °C ambient, with effective θJA = 2.4 °C/W. Vivado reports a thermal margin of 59.7 °C, equivalent to 24.4 W under the selected assumptions. This suggests ample thermal headroom for the synthesized design.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 8, August 2026
Impact Factor: 5.731

6.6. Functional Simulation Assessment

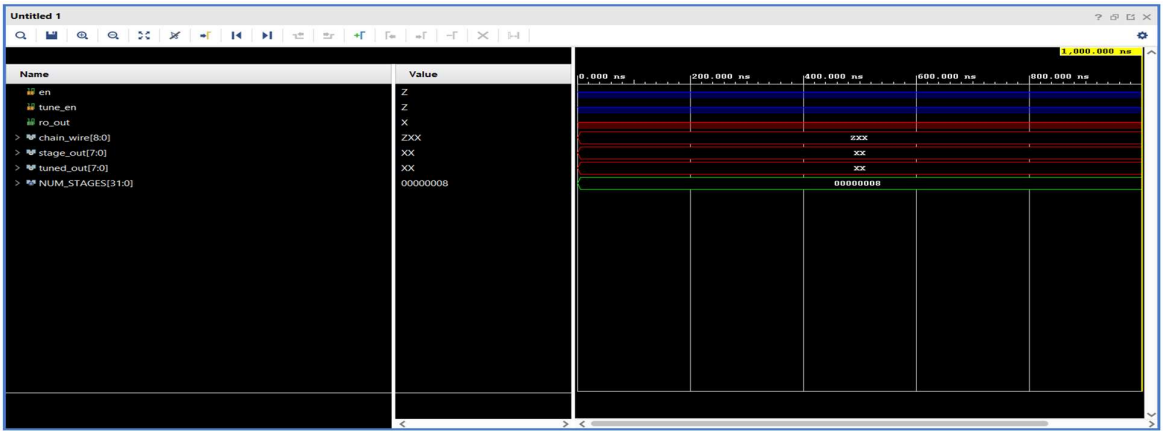

Fig 12. Supplied behavioral waveform over the 0–1000 ns interval.

Across the displayed 1 µs interval, the two control inputs en and tune_en are high impedance (Z). The oscillator output ro_out is unknown (X), chain_wire[8:0] is unresolved, and stage_out[7:0] and tuned_out[7:0] remain unknown. The constant NUM_STAGES value is correctly shown as eight, but no valid output transition is observed. This prevents calculation of oscillation period, frequency, duty cycle, phase offsets, start-up time, or tuning step.

VII. APPLICATIONS AND PRACTICAL DESIGN CONSIDERATIONS

Beyond PUFs and TRNGs, the proposed oscillator can serve as an on-chip temperature or voltage sensor because changes in propagation delay appear as shifts in oscillation frequency. CC_ROv2 is suited to applications where stability is preferred, while CC_ROv3 can be configured to increase sensitivity for sensing. The architecture can also support aging monitoring, timing characterization, multi-oscillator frequency comparison, and compact FPGA-based embedded systems.

The principal limitation of the supplied study is the absence of a complete numerical experimental dataset in the provided document. Device family, implementation constraints, measured frequencies, power values, and statistical repeatability results are not specified in the Results section. These measurements are necessary before making device-independent quantitative claims. The presented work should therefore be interpreted as an architecture and characterization methodology supported by qualitative implementation findings rather than a complete silicon-calibrated benchmark.

VIII. FUTURE WORK

The source proposes several extensions: adaptive control that changes oscillator parameters in response to operating conditions; large arrays of carry-chain oscillators for PUF, TRNG, and high-resolution sensing; evaluation on newer FPGA process nodes; machine-learning-assisted prediction and optimization; security hardening against side-channel attacks; and application-specific adaptation for biomedical sensing, wireless systems, and real-time monitoring. A particularly important next step is a multi-device experimental campaign that reports frequency distributions, run-to-run placement variation, power, resource use, and temperature/voltage coefficients for CC_ROv1, CC_ROv2, CC_ROv3, and a matched LUT-based baseline.

IX. CONCLUSION

A predictable and tunable multistage FPGA ring oscillator architecture based on dedicated carry-chain resources has been presented. The design replaces the routing-dominated LUT inverter chain with a structured path formed by carry-chain XOR and multiplexer elements. CC_ROv1 provides the scalable base architecture, while CC_ROv2 and CC_ROv3 introduce pass-through LUT delays for finer control of oscillator frequency and environmental response. The source study reports improved implementation consistency together with reduced area and energy relative to conventional LUT-based oscillators and highlights multiphase extraction as an important architectural advantage. The methodology

therefore provides a strong basis for hardware-security primitives, random-number generation, sensing, and FPGA timing characterization. Quantitative device-level benchmarking remains necessary to establish the exact performance gains on specific FPGA families.

REFERENCES

[1] F. Kodýtek, R. Lórencz, and J. Bucek, "Three counter value based ROPUFs on FPGA and their properties," *Microprocessors and Microsystems*, vol. 88, Feb. 2022.
[2] B. Halak, M. Zwolinski, and M. S. Mispan, "Overview of PUF-based hardware security solutions for the Internet of Things," in *Proc. IEEE 59th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, 2016.
[3] M. Barbareschi *et al*., "A ring oscillator-based identification mechanism immune to aging and external working conditions," *IEEE Trans. Circuits Syst. I: Regul. Papers*, 2018.
[4] B. Yang *et al*., "ES-TRNG: A high-throughput true random number generator," *IACR Trans. Cryptographic Hardware Embedded Syst. (TCHES)*, 2018.
[5] B. Richitha and J. MD, "Fault space transformation is a versatile method for implementing fault-tolerant parallel filters," *Accent Journal of Economics Ecology & Engineering*, vol. 9, no. 7, pp. 131–137, Jul. 2024.
[6] V. B. Raju, J. MD, K. Keshamoni, and D. S. Reddy, "AI-driven water quality prediction and aquatic pollution monitoring using machine learning algorithms," *International Journal of Aquatic Research and Environmental Studies*, vol. 6, no. S2, pp. 878–885, 2026, doi: 10.70102/gwy42c46.
[7] M. A. Prada-Delgado *et al*., "Auto-calibrated RO-TRNG," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2020.
[8] D. S. Reddy, V. B. Raju, J. MD, and K. Keshamoni, "AI-based aquatic plastic waste detection using underwater image processing and deep learning algorithms," *International Journal of Aquatic Research and Environmental Studies*, vol. 6, no. S2, pp. 902–909, 2026, doi: 10.70102/ate39q94.
[9] S. Moini *et al*., "Voltage sensor implementations for FPGA attacks," *ACM Trans. Reconfigurable Technol. Syst. (TRETS)*, 2022.
[10] I. Giechaskiel *et al*., "Measuring long wire leakage with ROs," in *Proc. Int. Conf. Field Programmable Logic Appl. (FPL)*, 2019.
[11] M. Ebrahimi and Z. Navabi, "FPGA aging detection using delay paths," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, 2020.
[12] M. M. Alam *et al*., "Recycled FPGA detection using delay characterization," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 2019.
[13] J. MD, K. Keshamoni, D. S. Reddy, and V. B. Raju, "Underwater audio species classification using dual-path deep learning," *International Journal of Aquatic Research and Environmental Studies*, vol. 6, no. S2, pp. 886–901, 2026, doi: 10.70102/q8gzzg88.
[14] M. Elnawawy *et al*., "Role of FPGA in IoT applications," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT)*, 2019.
[15] C. Gu *et al*., "Evaluation of RO-PUFs on Xilinx FPGAs," 2019.
[16] A. Wild *et al*., "Reliable RO-PUF implementation challenges," 2016.
[17] J. Burgiel *et al*., "IOBUF-based ring oscillators," 2021.