Research Article GU J Sci, Part A, 12(1): 61-71 (2025) 10.54287/gujsa.1645022

Gazi University
Journal of Science
PART A: ENGINEERING AND INNOVATION
http://dergipark.org.tr/gujsa

A Novel Dynamic Clock Generator Circuit for the Threshold Logic Gate

Ahmet UNUTULMAZ[1*]

[1] *Marmara University, İstanbul, Türkiye*

Keywords

Threshold Logic Gate
Dynamic Circuit
Clock Generator

Abstract

Threshold Logic Gate (TLG) has gained attention with the emergence of novel technologies such as memristors. TLG offers improved performance and lower power dissipation while occupying less silicon area. This paper introduces a novel dynamic clock generator circuit that further enhances TLG performance. The proposed circuit replaces the NAND gate-based approach used for clock generation in differential TLG implementations. It reduces the propagation delay of the TLG while reducing its static power dissipation, an important factor in energy-efficient circuit design. Simulations indicate up to a 25% reduction in delay compared to the NAND gate-based approach. Furthermore, the proposed circuit occupies 45% less area than the NAND gate. These findings highlight the potential of the proposed dynamic clock generator for advanced threshold logic implementations, paving the way for further innovations in the field.

Cite

Unutulmaz, A. (2025). A Novel Dynamic Clock Generator Circuit for the Threshold Logic Gate. *GU J Sci, Part A, 12*(1), 61-71. doi:10.54287/gujsa.1645022

Author ID (ORCID Number)

0000-0001-7738-9286 Ahmet UNUTULMAZ

Article Process

Submission Date 23.02.2025
Revision Date 04.03.2025
Accepted Date 12.03.2025
Published Date 26.03.2025

1. INTRODUCTION

The emergence of technologies like memristors has increased the popularity of Threshold Logic Gate (TLG) (Mazumder, 2012). This gate offers several advantages, including enhanced power efficiency, reduced circuit complexity, and increased robustness. Yang et al. (2014) introduced a differential TLG implementation incorporating memristors. This implementation improved circuit reliability and lowered failure rates due to the integration of memristor networks. By embedding memristors into TLGs, circuit resilience is enhanced, and power consumption is reduced (Vrudhula, 2015). This makes them ideal for modern high performance computing applications. A comprehensive overview of mem-resistive threshold logic circuits is provided in (Maan, 2016), covering a range of implementations and their respective benefits. More recently, a physical implementation of a memristor-based TLG was demonstrated in (Papandroulidakis, 2019), reinforcing the practicality and efficiency of these designs. Youn et al. (2024) demonstrated programmable threshold logics using a 32 × 32 memristor crossbar array. TLG implementations in emerging technologies are not limited to memristors. Han et al. (2021) fabricated a bio-inspired reconfigurable threshold logic circuit. There is even a TLG implementation with optical systems. Sarkar et al. (2021) implemented an optical threshold logic gate using a reflector telescopic system and a resonant Fabry-Perot cavity.

*Corresponding Author, e-mail: ahmet.unutulmaz@marmara.edu.tr

Beyond emerging technologies, TLGs have also demonstrated significant advantages in conventional semiconductor applications. Several studies emphasize their benefits, particularly in performance enhancement and power dissipation reduction. For instance, Kulkarni et al. (2016) reported that incorporating TLGs into binary circuits results in performance improvements and energy savings. Their findings suggest that TLGs can effectively optimize computational efficiency by minimizing transistor count and reducing dynamic power consumption, positioning them as a viable solution for high-performance digital circuits. Wagle et al. (2021) proposed the use of TLG in binary FPGAs. The proposed methodology is reported to reduce the area and power of configurable logic blocks while improving their performance.

Unutulmaz et al. (2024) expanded the application of TLGs to multi-valued logic circuits, specifically to ternary logic circuits. Their study demonstrated the potential of differential TLGs in reducing power dissipation, silicon area, and delay. This study shows the potential of TLG for multi-valued logic implementations. Multi-valued logic is advantageous as it allows a single wire to represent more than two distinct logic levels, unlike binary logic, which is limited to only two. The capability of representing more than two levels enables spatial compression of information. This in effect reduces the routing resource requirements (Saxena, 2007) and allows further reduction of silicon area.

A simplified representation of the TLG proposed by Unutulmaz et al. (2024) is shown in Figure 1. This circuit processes two sets of ternary signals and compares them. The operational principles of the circuit, along with a detailed breakdown of its functional components, are elaborated in Section 2.

As depicted in Figure 1, the circuit requires a clock signal for operation. The clock input may be connected to the system clock if there is only one TLG or if the TLG is the first element in a timing path. On the other hand, if a TLG is connected to another one, the subsequent TLG requires a specific clock which will trigger a comparison after the previous one completes its operation. This could be achieved by connecting a NAND gate to the outputs u and $\bar{u}$ (Figure 1) as proposed by Celinski et al. (2002). Connections of the NAND gate are shown in Figure 2.a. The output of the NAND gate, *clk_out* signal, could be used as the clock input to the subsequent TLG circuit.

This paper presents a novel circuitry to replace the NAND gate base clock generation. Performance of the clock generation circuit is improved via a novel dynamic clock generator circuit. The proposed dynamic clock generator is shown in Figure 2.b. The proposed circuit not only improves the performance but also reduces the static power dissipation which is an important factor in low-power and energy-efficient circuit design.

A comparison of the proposed dynamic clock generator against the design introduced by Celinski et al. (2002) is conducted in Section 3. The comparison covers critical performance metrics such as power consumption, delay, and silicon area. The results demonstrate that the proposed dynamic clock generator outperforms the previous design, highlighting its viability for next-generation threshold logic implementations. Section 4 concludes the paper by summarizing the key findings of this study.

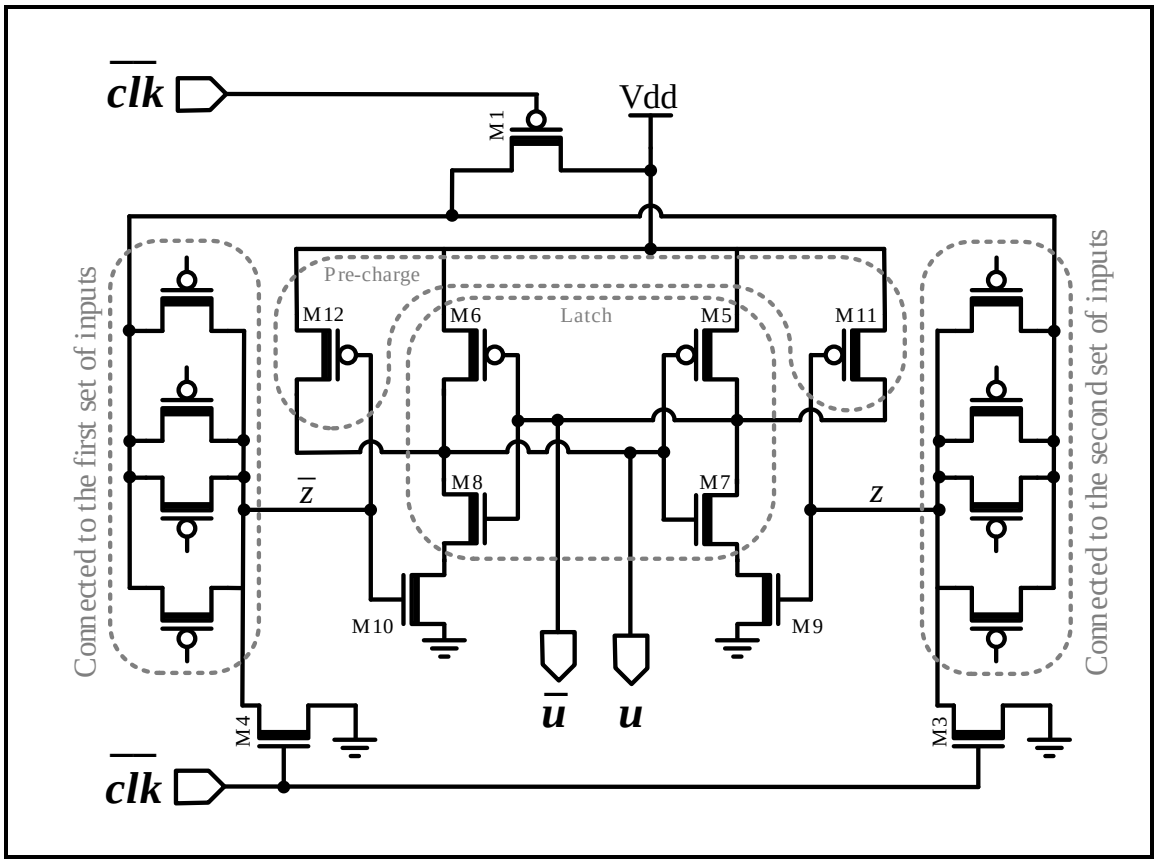

Figure 1. *Schematic of the Ternary TLG Circuit*

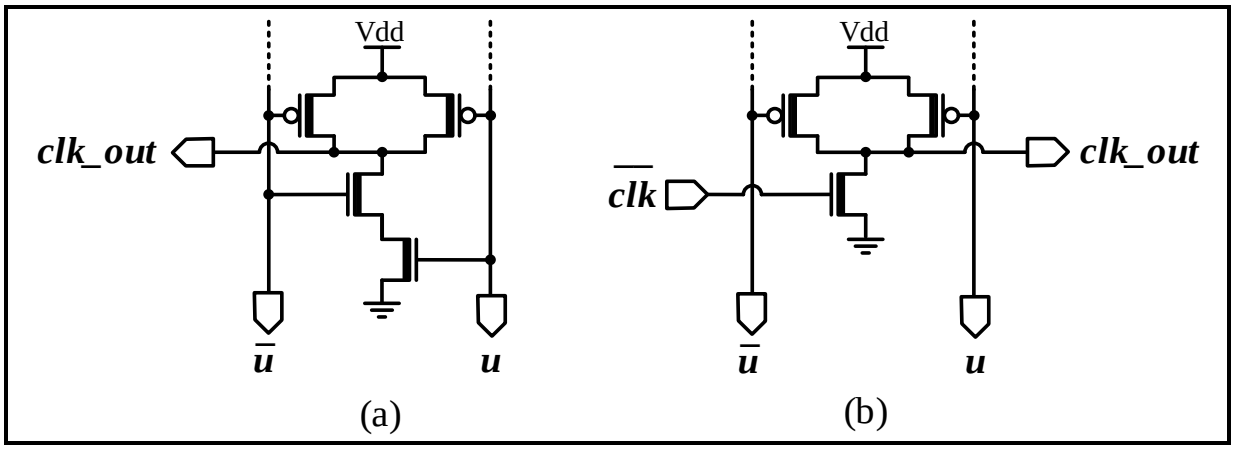

Figure 2. a) *Schematic of the NAND Gate,* ***b)*** *The Proposed Dynamic Circuit*

2. MATERIAL AND METHOD

In this section the operation principles of the TLG circuit shown in Figure 1 and the clock generator circuits shown in Figure 2 are explained.

The circuit in Figure 1 operates in two distinct phases: the reset phase and the comparison phase. Both phases are essential in ensuring the correct functionality of the TLG circuit by preparing and evaluating the internal signals to produce correct outputs.

The reset phase begins when the clock signal transitions from high to low. In this phase, any residual charge from the previous operation is discharged to ground. This ensures correct operation in the following comparison phase. The inverse of the clock signal is connected to the gates of the M3 and M4 transistors, which are NMOS transistors. Thus, when the reset phase starts, M3 and M4 are turned on. These discharge the internal nodes z and $\bar{z}$ to the ground potential. The internal nodes z and $\bar{z}$ are connected to the gates of the M11 and M12 transistors. These transistors are PMOS devices and start conducting as the internal nodes z and $\bar{z}$ are discharged to ground potential. These transistors charge the output nodes u and $\bar{u}$ to the supply voltage Vdd. Setting the output nodes to Vdd establishes a clear initial condition for the comparison phase, enabling the circuit to accurately identify differences between the input signals.

Since z and $\bar{z}$ are discharged to ground, the NMOS transistors M9 and M10 are in cut-off and do not conduct. Although the output nodes u and $\bar{u}$ are charged to Vdd, turning on the NMOS transistors M7 and M8, these transistors still do not conduct because M9 and M10 are off, preventing current flow. This isolation prevents any current paths from the output nodes u and $\bar{u}$ to the ground. Thus, the output remains stable at the Vdd potential during the reset phase.

The second phase, the comparison phase, begins when the clock signal transitions from logic low to high. As the inverse of the clock signal is connected to the NMOS transistors M3 and M4, these transistors are turned off. Simultaneously, the PMOS transistor M1, which is also connected to the inverse of the clock signal, is turned on. During this phase, the transistors connected to the input signals begin charging the internal nodes z and $\bar{z}$ to the supply voltage Vdd. The transistors connected to the inputs are grouped and labeled in Figure 1. Readers are referred to the study of Unutulmaz et al. (2024) for a detailed discussion on the procedure to generate the input signals. This discussion is not in the scope of this paper. Depending on the input configuration, either node z and $\bar{z}$ charges faster than the other. The node that charges more quickly activates its corresponding transistor, M9 or M10, before the other.

At the center of the circuit, there is a latch composed of two cross-coupled inverters, as shown in Figure 1. When either transistor M9 or M10 is turned on, the connected inverter begins discharging its output node, u or $\bar{u}$, to ground. The cross-coupled inverters provide positive feedback, which ensures that the outputs u or $\bar{u}$ rapidly change to either ground or Vdd potentials. This positive feedback mechanism guarantees a clear distinction between the output states.

As an example, the signals on the u and $\bar{u}$ nodes, as well as the *clk* signal, are shown in Figure 3. When the clock is low the circuit is in reset phase. And the circuit is in comparison phase when the clock is high. The phases of the circuit are also indicated in the figure. As discussed in the previous paragraphs, the output nodes u and $\bar{u}$ are at high logic during the reset phase. Celinski et al. (2002) proposed to connect a NAND gate to the output nodes u and $\bar{u}$ as shown in Figure 2.a. Since the inputs of the NAND gate are u and $\bar{u}$ remains at low logic during the reset phase, the output of the NAND gate, *clk_out*, will be high. The *clk_out* signal is also

shown in Figure 3. The comparison phase starts when the clock signal transitions from low to high. In this phase, one of the outputs of the circuit, u or $\bar{u}$, goes low. Then, the NAND gate outputs a high logic as shown in Figure 3. In summary, the output of the NAND gate goes from low to high when the TLG circuit makes its decision. This way, the output of the NAND gate could be used as a clock generator for the subsequent TLG. When the TLG completes its operation, it triggers the following one.

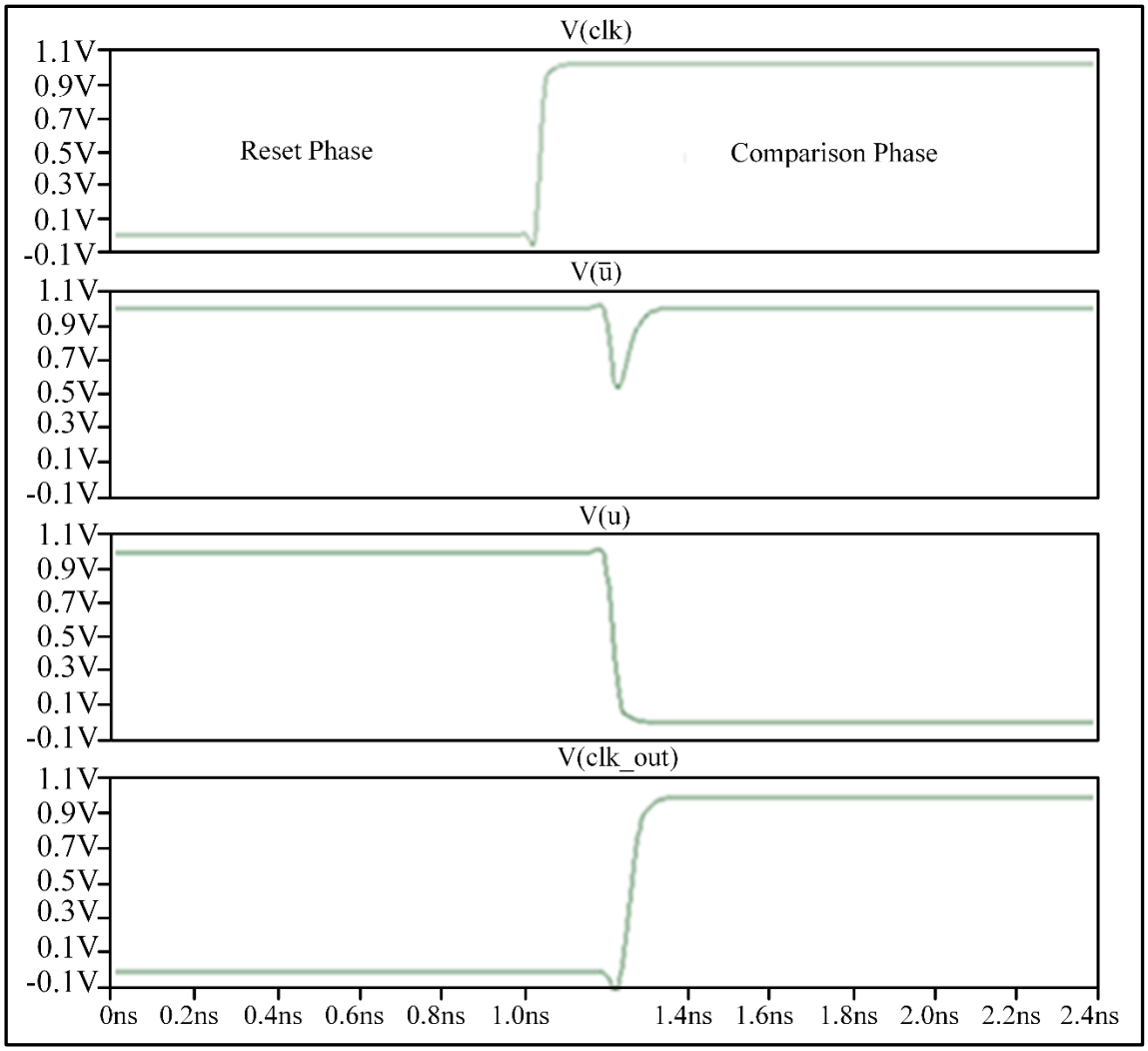

Figure 3. *Simulation Results for the NAND Gate*

This study proposes the logic circuit shown in Figure 2.b as an alternative to the NAND gate. Thus, the proposed circuit is a replacement to the NAND gate introduced by Celinski et al. (2002). The proposed circuit not only uses information from the u or $\bar{u}$ signals, but also information from the clock signal. The simulation results proposed in Section 3 clearly show that the performance of the TLG circuit is improved when the proposed circuit is used as the clock generator instead of the NAND gate.

Here, the working principles of the proposed dynamic logic circuit are explained. As already stated, the outputs u and $\bar{u}$ are high during the reset phase. These signals are connected to the PMOS transistors of the proposed circuit in Figure 2.b. Thus, these transistors are turned off during the reset phase. Because the clock signal is

low during this phase, the NMOS transistor in the proposed circuit is forced to turn on. This way, the output of the proposed circuit *clk_out*, is kept low during the reset phase. As the input clock transitions from low to high, the NMOS transistor is turned off. The output *clk_out*, is not driven by any transistors. The output is dynamically kept at ground by the parasitic capacitances at the *clk_out* node. After a short duration, one of the outputs, either u or $\bar{u}$, transitions to low. This duration is measured to be approximately 200ps when the circuit is simulated using ptm-32nm (Zhao, 2006) transistor models. The Vdd potential is set to 1V. The high to low transition turns on one of the PMOS transistors. This subsequently shifts the output of the circuit from low to high. Importantly, there is no short circuit current from PMOS to NMOS during this transition. This characteristic of the circuit enhances the circuit's response time. The current flowing through the PMOS transistor is completely utilized to charge the output node *clk_out*. No charge is lost to ground.

As an example, the output of the proposed circuit as well as the short circuit current from Vdd to ground is shown in Figure 4. When the signal u changes from high to low, the output of the proposed circuit changes from low to high. During this low to high transition, there is no short circuit current as shown in the figure.

When the input clock transitions from high to low logic, the NMOS transistor of the proposed circuit turns on. This transistor starts to discharge the output node *clk_out* to ground. Since one of the outputs u or $\bar{u}$ is high, a temporary short circuit current flows from Vdd to ground. When both outputs u and $\bar{u}$ charge to Vdd, the PMOS transistors are turned off. This prevents further short circuit current. Then, the output clock *clk_out* switches from high to low logic. Consequently, the circuit does not dissipate any static current. As shown in Figure 4, there is some short-circuit current during the high to low transition, but there is no static current. As the simulation results in Section 3 indicate, the delay of proposed circuit, during high to low transition of the output *clk_out*, is less than that of the NAND gate. This is mainly because the NAND gate discharges the output through two series-connected NMOS transistors. The NAND gate needs to discharge the parasitic capacitances of these two NMOS transistors, whereas the proposed circuit discharges the parasitic capacitance of a single NMOS transistor.

To be able to make a fair comparison, we equated the pull-down resistances of the NAND gate and the proposed logic gate. This is achieved by sizing the NMOS transistors of NAND gate twice as wide as the one in the proposed circuit, which speeds up high to low transition of the NAND gate. The sizes of the PMOS transistor are chosen to be the same. The transistor sizes for the NAND gate and the proposed circuit are listed in Table 1.

Table 1. *Transistor Sizes*

W/L	NMOS	PMOS
NAND (Celinski, 2002)	90nm/45nm	60nm/45nm
Dynamic (This Work)	45nm/45nm	60nm/45nm

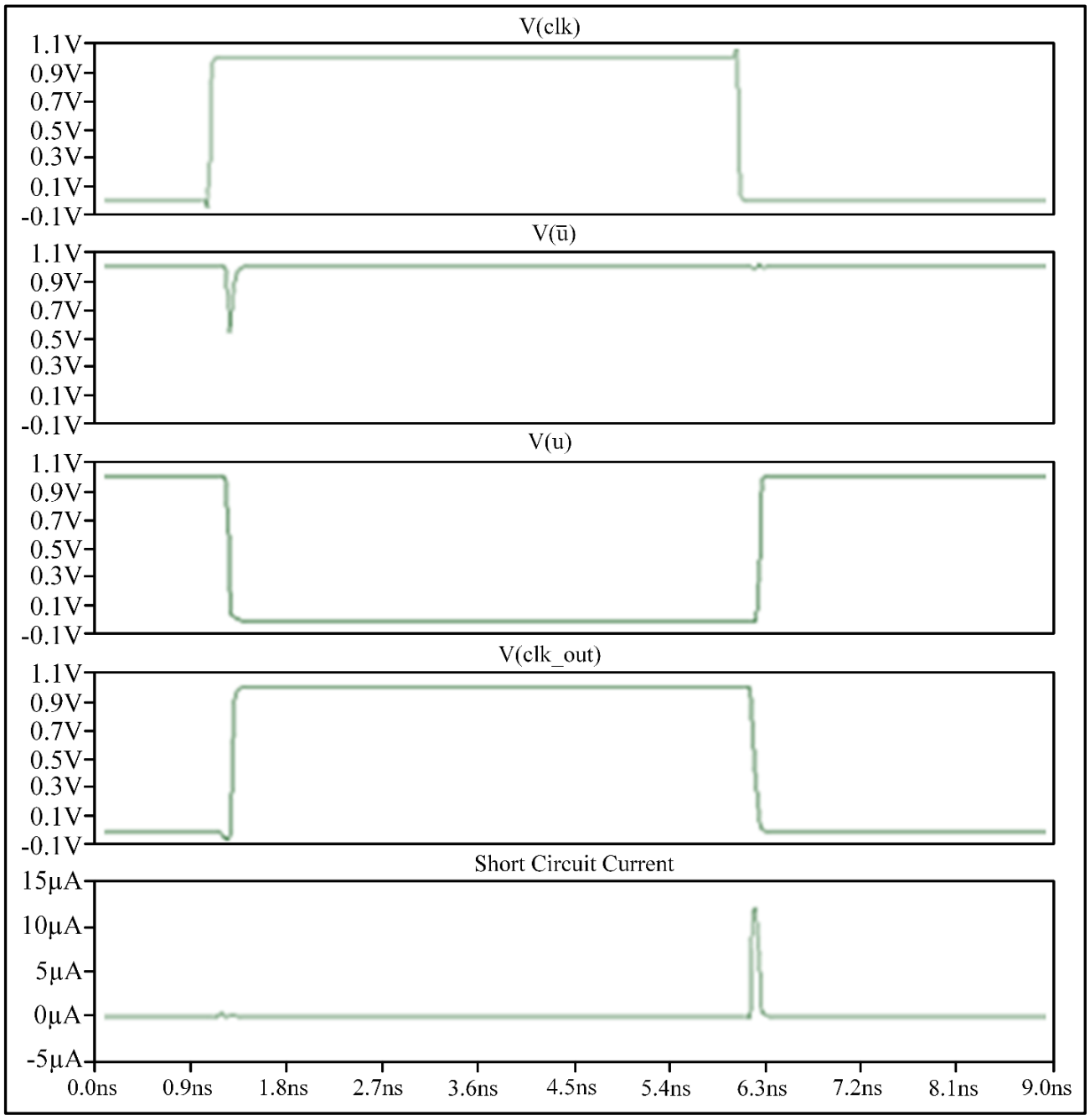

Figure 4. *Simulation Results for the proposed Dynamic Clock Generator*

3. RESULTS AND DISCUSSION

This section presents a comparative analysis of the delay performance of the TLG when implemented using NAND-based and dynamic circuit-based clock generators. Additionally, energy consumption per clock generation and leakage power are evaluated based on simulation results. The ptm-32nm transistor models (Zhao, 2006) are used. The Vdd potential is set to 1V. The simulations are conducted using the LTSPICE simulator. To reduce measurement errors, seven ternary TLG circuits are connected sequentially. The output clock delays are measured for both the circuits using the NAND gate and the proposed dynamic gate. The

choice of seven TLGs is arbitrary and intended solely to ensure that the labels are visually distinguishable in the simulation results. The corresponding simulation results are presented in Figure 5.

The *clk* in Figure 5 is the input clock. The same input clock is applied to both circuits. The clock outputs of the ternary TLG circuits are labeled as *clk_c_i* for the circuits with NAND gates, where $i \in \{1,2,\dots,7\}$ represents the output of the *i*th TLG. Similarly, *clk_d_i* represents the output of the *i*th ternary TLG with the dynamic circuit, where $i \in \{1,2,\dots,7\}$. The results obtained from the simulations are tabulated in Table 2.

Table 2. *Results*

GATEs	**Delay of TLG**		**Area of the Gate (nm2)**	**Dissipated Energy by the Gate (per Operation, fJ)**	**Leakage Power (pW)**
	Comparison (ps)	Reset (ps)			
NAND (Celinski, 2002)	230	230	13500	**0.5**	50
Dynamic (This work)	**215**	**170**	**7425**	0.8	**15**

The ternary TLGs with the NAND gate and the proposed circuit complete their comparison phases in approximately 230ps and 215ps, respectively. The proposed circuit achieves a 6% reduction in delay during the comparison phase. Similarly, the ternary TLGs with the NAND gate and the proposed circuit reset in approximately 230ps and 170ps, respectively, resulting in a 25% reduction in delay during the reset phase. These results indicate that the ternary TLG with the proposed dynamic logic operates faster than the one with the NAND gate. Additionally, the proposed circuitry requires less chip area compared to the NAND gate.

However, the energy consumption of the proposed circuit is slightly higher than that of the NAND gate. While generating a clock signal, the proposed dynamic circuit consumes 0.8fJ, whereas the NAND gate consumes 0.5fJ. In terms of leakage power, the proposed gate offers an advantage. The leakage power of the proposed circuit is 15pW, compared to 50pW for the NAND gate. These results suggest that the proposed gate is a viable alternative to the NAND gate for applications requiring high performance and for scenarios where minimizing static power consumption is critical.

4. CONCLUSION

The proposed dynamic clock generator enhances the overall performance of TLG by reducing propagation delay. One of the advantages of the proposed circuit is its ability to reduce leakage power dissipation, which is an important factor in low-power and energy-efficient circuit design. By minimizing leakage currents, the circuit helps to improve overall power efficiency. This makes the proposed clock generation circuit a suitable choice for modern electronic applications where power consumption is a critical concern. Although the proposed clock generator effectively reduces leakage power, it does exhibit slightly higher energy consumption compared to the NAND gate.

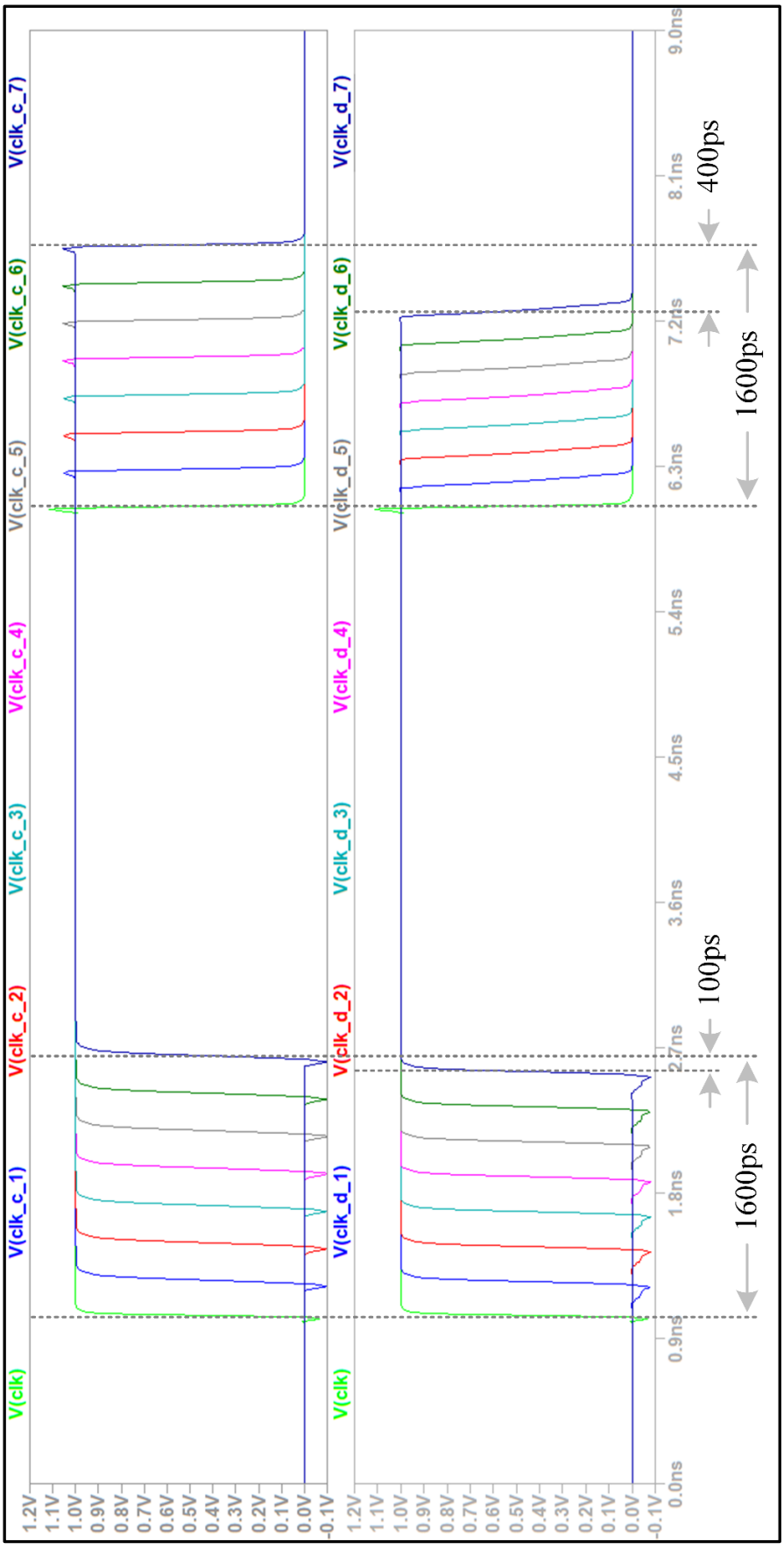

Figure 5. *Simulation Results*

The trade-off between leakage power reduction and transient energy consumption is an important consideration when evaluating the circuit's suitability for various applications. The overall benefits in terms of performance and power efficiency make the proposed circuit an attractive alternative to the NAND gate, particularly in performance-critical applications where speed and static power consumption are of utmost importance. With the growing demand for high-performance and low-power circuits in emerging technologies, this dynamic clock generator could play an important role in optimizing the operation of threshold logic circuits, making them more efficient and effective in real-world applications.

CONFLICT OF INTEREST

The author declares no conflict of interest.

REFERENCES

Celinski, P., López, J. F., Al-Sarawi, S., & Abbott, D. (2002). Compact parallel (m, n) counters based on self-timed threshold logic. *Electronics Letters, 38(13),* 633-635. https://doi.org/10.1049/el:20020438

Han, J. K., Lee, M. W., Yu, J. M., & Choi, Y. K. (2021). A Single Transistor-Based Threshold Switch for a Bio-Inspired Reconfigurable Threshold Logic. *Advanced Electronic Materials*, *7*(5), 2100117. https://doi.org/10.1002/aelm.202100117

Kulkarni, N., Yang, J., Seo, J. S., & Vrudhula, S. (2016). Reducing power, leakage, and area of standard-cell asics using threshold logic flip-flops. *IEEE Transactions on very large scale integration (VLSI) systems, 24(9),* 2873-2886. https://doi.org/10.1109/tvlsi.2016.2527783

Maan, A. K., Jayadevi, D. A. & James A. P., (2016). A survey of memristive threshold logic circuits. *IEEE Transactions on Neural Networks and Learning Systems*, 28(8), 1734–1746. https://doi.org/10.1109/tnnls.2016.2547842

Mazumder, P., Kang, S.-M. & Waser, R. (2012). *Memristors: Devices, Models and Applications*. In: Proceedings of IEEE, 100 (6), (pp. 1911–1919). https://doi.org/10.1109/JPROC.2012.2190812

Papandroulidakis, G., Serb, A., Khiat, A., Merrett, G. V. & Prodromakis, T. (2019). Practical implementation of memristor-based threshold logic gates. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 66(8), 3041–3051. https://doi.org/10.1109/TCSI.2019.2902475

Sarkar, M., Chakraborty, R., Taki, G. S., & Chakraborty, A. K. (2021). Design of basic logic gates using optical threshold logic. Engineering Research Express, 3(3), 035021. https://doi.org/10.1088/2631-8695/ac1903

Saxena, P., Shelar, R. S., & Sapatnekar, S. (2007). *Routing Congestion in VLSI Circuits: Estimation and Optimization*. Springer Science & Business Media. https://doi.org/10.1007/0-387-48550-3

Unutulmaz, A., & Ünsalan, C. (2024). Implementation and applications of a ternary threshold logic gate. *Circuits, Systems, and Signal Processing*, 43(2), 1192-1207. https://doi.org/10.1007/s00034-023-02512-1

Vrudhula, S., Kulkami, N. & Yang, J. (2015). *Design of threshold logic gates using emerging devices*. In: Proceedings of 2015 IEEE International Symposium on Circuits and Systems, Lisbon, (pp 373–376). https://doi.org/10.1109/iscas.2015.7168648

Wagle, A., & Vrudhula. S. (2021) Heterogeneous FPGA architecture using threshold logic gates for improved area, power, and performance. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 41(6), 1855-1867. https://doi.org/10.1109/TCAD.2021.3099780

Yang, J., Kulkarni, N., Yu, S. & Vrudhula, S. (2014). *Integration of threshold logic gates with RRAM devices for energy efficient and robust operation*, In: Proceedings of 2014 IEEE/ACM International Symposium on Nanoscale Architectures, (pp 39–44).

Youn, S., Lee, J., Kim, S., Park, J., Kim, K., & Kim, H. (2024). Programmable threshold logic implementations in a memristor crossbar array. *Nano Letters*, 24(12), 3581-3589. https://doi.org/10.1021/acs.nanolett.3c04073

Zhao, W., & Cao, Y. (2006). New generation of predictive technology model for sub-45 nm early design exploration. *IEEE Transactions on electron Devices*, 53(11), 2816-2823. https://doi.org/10.1109/ted.2006.884077