

Implementing Smart Clock Gating in Complex FPGA Systems for Power Reduction

Aruna M, Harinath B, C Mallikarjun, Cheboyina Bhanu Teja, Harshavardhan Reddy

Department of Electronics and Communication Engineering,

Presidency University, Bengaluru

Abstract

This design presents an energy-effective 4-bit computation sense Unit (ALU) design using timepiece gating as a power optimisation technique for VLSI circuits. The ALU is enforced in Verilog HDL and synthesised on the Nexys Artix 7 FPGA board using Xilinx Vivado. Timepiece gating is used to widely enable the timepiece signal only to the needed functional blocks, based on the operation, significantly reducing dynamic power consumption by minimising gratuitous switching exertion. Simulation and conflation results demonstrate notable power savings with minimum area above, proving the effectiveness of this fashion in low-power digital systems such as mobile and bedded operations. The proposed design serves as a practical approach for enforcing energyeffective VLSI circuits in ultramodern powersensitive surroundings.

Keywords: Artix-7 FPGA, ALU, Verilog HDL, Xilinx Vivado.

I. Introduction

The growing demand for protean electronic bias, similar to mobile phones, Iot bias, and batterypowered systems, has made power optimisation a significant design challenge. One of the most constantly used factors in a processor is the computation sense Unit (ALU), which is responsible for executing computation and sense operations.

In this environment, optimising the 4-bit ALU can greatly ameliorate the overall power effectiveness of a digital system. In conventional 4-bit ALU designs, all functional blocks similar to adders, subtractors, and sense units remain active regardless of whether they're being used or not. This results in unnecessary switching activity and

increased dynamic power consumption. Dynamic power is primarily caused by switching capacitive loads and typically dominates total power consumption in most CMOS-based circuits. To mitigate this issue, clock gating is widely used as a power-saving strategy. Clock gating works by disabling the clock signal to idle functional blocks, thereby preventing unnecessary toggling and reducing power consumption. As a result, minimizing clock power is critically important.

Clock gating is a fundamental power reduction technique employed by hardware designers. It is typically implemented at the RTL (Register Transfer Level) using HDL simulations or analysed at the gate level using power analysis tools.

II. Clock Gating

Dynamic Timepiece gating is a broadly embraced and successful control optimisation method in VLSI (Veritably Huge Scale Integration) circuit design. It essentially centres on decreasing energetic control utilization by crippling the timepiece flag in circuit pieces that are practically sit out of gear.

This is by and large satisfied utilising gating sense basics comparable as AND, OR entryways, or specialised locks — to broadly control the proliferation of the timepiece flag based on the utilitarian conditions of the circuit.

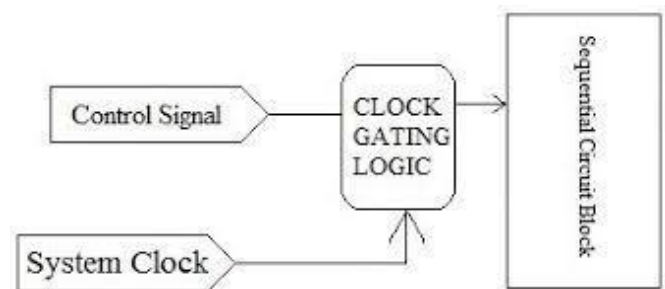


Fig.1: Principle of clock gating.

The foremost advantage of timepiece gating lies in its capability to minimise unwarranted exchanging

effort, in this manner decreasing control scattering in coetaneous advanced frameworks. This approach is particularly healthy in operations where particular utilitarian units or modules stay inert for significant periods, compared to versatile and battery-fuelled inclination. Despite its benefits, upholding timepiece gating requires cautious planning to mitigate issues like timepiece skew, timing infringement, and glitch proliferation, which can contrarily influence the trustworthiness and performance of the framework. In this way, strong planning strategies and confirmation methods are basic to safeguard the secure integration of timepiece gating in complex VLSI frameworks.

III. Artix-7 FPGA

The Nexys A7 is well-suited for coaching computerised arrange, computer armature, had relations with systems, and honest to goodness-time hail planning. Its comprehensive I/ O and peripherals allow it to be utilised in lab environments, summit systems, and without a question, alluring show enhancement.

Its compatibility with the Vivado Design Suite engages analysts and masterminds to bolt in in veritable-world industry workflows, tallying conflation, put and course, timing analysis, and bitstream period.

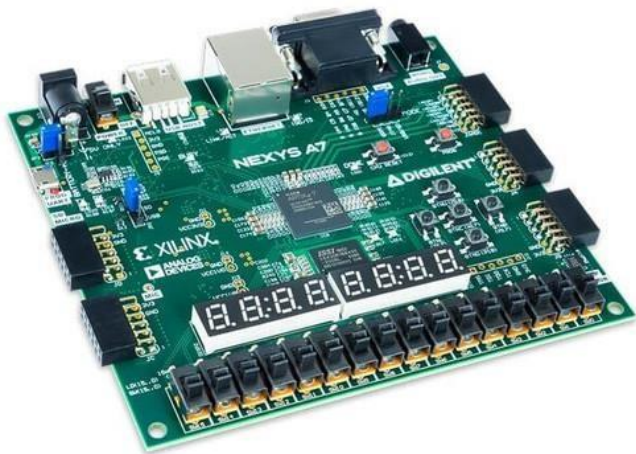


Fig.2: Nexys Artix – 7 FPGA Board

In terms of educational and exploration operations, the Nexys A7 is extensively used in academic

surroundings for tutoring digital system design, VHDL/ Verilog programming, microprocessor architecture, and signal processing. It's especially suitable for enforcing systems like custom ALUS, CPU cores, and power-optimised circuits using methods similar to timepiece gating, which can be explored and validated using on-board coffers.

It's finagled to meet both academic and artificial requirements, furnishing an accessible and important platform for literacy, prototyping, and planting complex digital systems. With 100K sense cells, the Artix-7 FPGA delivers high performance and effective resource application, making it suitable for a wide range of digital design operations.

The board includes 128 MB DDR2 SDRAM for high-speed memory access and 16 MB quadrangle-SPI Flash for non-volatile configuration storage. It also features a microsd card slot for removable data storage, enabling data logging or expansion in embedded operations. For connectivity, the Nexys A7 supports USB-JTAG and USB-UART interfaces for programming and periodical communication, a 10/100 Mbps Ethernet harbourage for network-grounded operations, and several P mod connectors that allow for easy integration of supplemental modules. Stoner commerce is enabled by 16 stoner-configurable switches, 16 LEDS, 5 drive buttons, and a 4 4-number 7- member display. The board also includes a 100 MHz system timepiece, with erected- in PLLS (Phase-Locked Loops) that allow for flexible timepiece generation and timing configurations within the FPGA design.

IV. 4 – BIT ALU

The Arithmetic Logic Unit (ALU) is a critical component of a central processing unit (CPU) or digital system, responsible for carrying out all arithmetic and logical operations. The Arithmetic Logic Unit (ALU) performs fundamental operations such as addition, subtraction, multiplication, and division, along with bitwise operations like AND, OR, XOR, and NOT. It operates based on control signals, known as

opcodes, which specify the operation to be performed on the input data. The ALU typically takes two input operands and produces an output result, along with optional status flags—such as carry, zero, overflow, and sign—that provide additional information about the result. These flags are often used for control flow in larger systems.

As a core component of processors, the ALU is essential for executing instructions and enabling computational logic in digital designs. Its efficiency and performance are critical to optimizing the overall processing capability and power consumption of modern embedded and computing systems.

A Number-crunching Rationale Unit (ALU) is a principal building block of a computerised processor that performs both mathematical and logical operations. A 4-bit ALU is planned to work on 4-bit twofold inputs, making it capable of executing essential computations and logic operations within a restricted information width. It regularly bolsters operations such as expansion, subtraction, consistent AND, consistent OR, XOR, and bit shifting. The 4-bit ALU acknowledges two 4-bit operands along with a control flag (opcode) that decides which operation is to be performed.

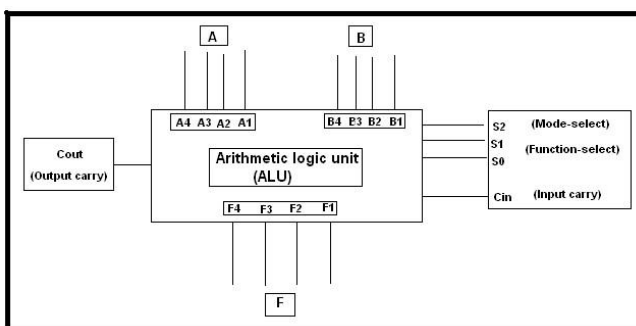


Fig.3: 4 Bit ALU Block Diagram

The yield comprises a 4-bit result along with conceivable status flags such as zero, carry, flood, or negative, which are fundamental for decisionmaking in processors. Due to its straightforwardness and flexibility, a 4-bit ALU is broadly utilised in teaching situations and as a test case for approving power-saving procedures such

as clock gating in FPGA-based advanced systems. When executed on equipment stages like the Nexys A7, a 4-bit ALU can be viably utilised to illustrate low-power design concepts by specifically empowering, as it were, the required useful squares based on the dynamic operation, in this manner minimising switching action and generally control utilisation.

V. Implementation

A. 4 Bit ALU without Clock Gating

A 4-bit Arithmetic Logic Unit (ALU) without clock gating is a basic digital circuit designed to execute standard arithmetic and logic operations on 4-bit binary inputs. It typically includes functions such as addition, subtraction, bitwise AND, OR, XOR, and logical shift operations. In this design, all functional units remain active and continuously receive the system clock signal, regardless of whether they are needed for a given operation.

This constant activity causes unnecessary switching in unused blocks, resulting in higher dynamic power consumption. This is especially problematic in CMOS-based systems, where dynamic (switching) power is a major contributor to total power usage. Although this architecture simplifies control logic and reduces design complexity, it is power-inefficient and therefore less suitable for low-power or battery-operated applications.

Nonetheless, a non-gated 4-bit ALU serves as an excellent baseline for understanding the impact of various power optimisation techniques, such as clock gating, in digital system design.

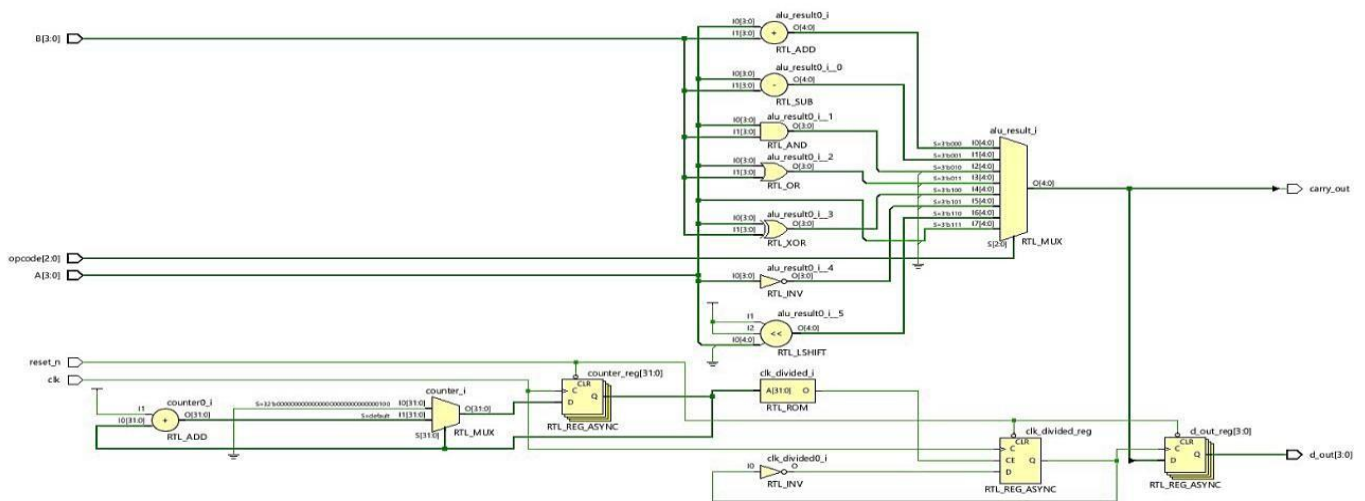


Fig.4: Design of 4-bit ALU without clock gating.

The picture speaks to a 4-bit Number-crunching Rationale Unit (ALU) outlined utilising Enlist Exchange Level (RTL) components, likely executed in an equipment portrayal environment such as Quartus Prime. The ALU gets two 4-bit inputs, labelled A [3:0] and B [3:0], and performs different math and coherent operations based on a 3-bit opcode input. The centre ALU operations incorporate expansion, subtraction, bitwise AND, OR, XOR, reversal (NOT), and a coherent cleared out move. Each of these operations is executed as an isolated RTL module (e.g., RTL_ADD, RTL_SUB), and the yields of these operations are fed into a multiplexer (RTL_MUX). The opcode decides which operation's result is chosen and directed to the yield flag labelled carryout, which, in this setting, likely speaks to the last ALU yield or maybe than a real carry bit.

In addition to the fundamental ALU rationale, the plan includes a clock division and control segment. This portion of the circuit employments counters, adders, and odd registers to produce a partitioned clock flag, which is utilised to time the yield registers.

A counter is augmented with each clock cycle, and its value is stored and nourished into a read-only memory (RTL_ROM), which makes a difference in creating a slower clock flag. This flag is at that point altered and utilised to clock a last 4-bit enlist that stores the ALU output.

This approach makes a difference in synchronising the yield and lessening the operational recurrence, which can be valuable for control effectiveness and investigation.



Fig.5: Waveform of 4-Bit ALU.

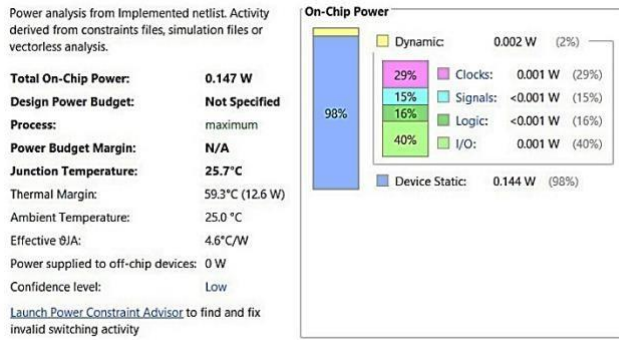


Fig.6: Power on 4 Bit ALU without clock gating.

The power evaluation, conducted using the synthesized netlist and switching activity derived from constraint files or simulation stimuli, reports a total on-chip power consumption of 0.147 watts. No specific power budget was defined for the design, and the analysis was carried out under worst-case process conditions, resulting in an undefined power margin.

Thermal analysis reveals a junction temperature of 25.7°C, with a substantial thermal margin of 59.3°C, equating to a maximum thermal power capacity of 12.6 watts. The ambient temperature during the test was 25.0°C, and the system's effective thermal resistance (θ_{JA}) is calculated at 4.6°C/W. Additionally, there is no power delivery to external components, indicating that all power usage is internal to the chip.

B. 4 Bit ALU with Clock Gating.

A clock-gated 4-bit Arithmetic Logic Unit (ALU) is a power-efficient digital module engineered to execute a range of arithmetic and logic operations on 4-bit binary data. These operations typically include addition, subtraction, and bitwise functions such as AND, OR, and XOR. What sets this ALU apart is the integration of clock gating—a technique that selectively disables the clock signal to inactive functional units.

In addition to performing basic operations and utilising clock gating for power efficiency, a 4-bit ALU with clock gating typically includes a control unit that interprets operation codes (opcodes) to determine which function to execute. The design may use multiplexers to select between different

operation outputs based on the opcode. Internally, the ALU contains separate functional blocks for arithmetic and logic tasks, and clockgating logic ensures these blocks are only active when required. This not only saves power but also minimises switching noise and thermal buildup.

Furthermore, status flags such as zero, carry, overflow, and sign can be generated to provide additional information about the result, which is crucial in decision-making processes in more complex digital systems. This type of ALU is often implemented using HDL (Hardware Description Language) tools like Verilog or VHDL and can be tested using FPGA platforms such as those supported by Quartus Prime Lite Edition.

The picture shows the RTL (Enlist Trade Level) schematic of a 4-bit ALU with timepiece gating, formed in a gear arrangement terrain, likely exercising Quartus or a relative FPGA instrument. At the stylish, inputs A (30) and B (30) talk to the 4-bit operands through the opcode (20) chooses the operation to be performed - similar to Incorporate, SUB, AND, OR, XOR, INV, and right move. These operations are performed in parallel, and their yields are directed into a multiplexer (RTL_MUX), which chooses the final alu_result based on the opcode. The chosen result is at that point transferred to the abnegate along with a carry out. Below the ALU operations, the timepiece gating element is executed.

A disconnected timepiece (clk_divided) is made by exercising a counter, ROM, and comparator system of logic. This file is at that point checked for changes, exercising XOR and comparison places to detect if there is any modification in operation or data.

However, an engage signal (ce_sync) is expressed, which grants the reopened timepiece (gated_clk) to drive the ALU's registers (d_out_reg), if a modification is honoured. This ensures that trading development, as it were, happens when there is a noteworthy data or control change, along these lines saving live control.

The use of colourful RTL_REG_ASYNC places amp highlights noncongruent enrol control for managing with reset and engaging operations, all contributing to a power-effective and measured ALU plan.

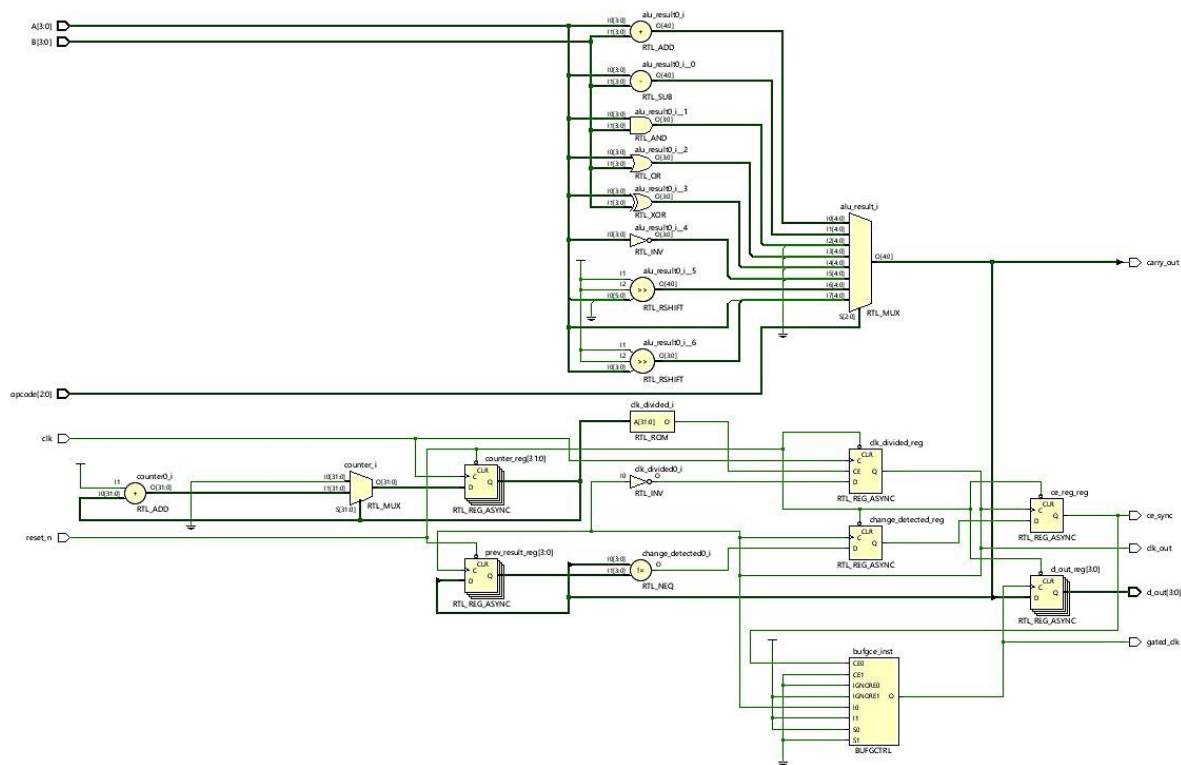


Fig.7: Design of 4-Bit ALU with Clock Gating.

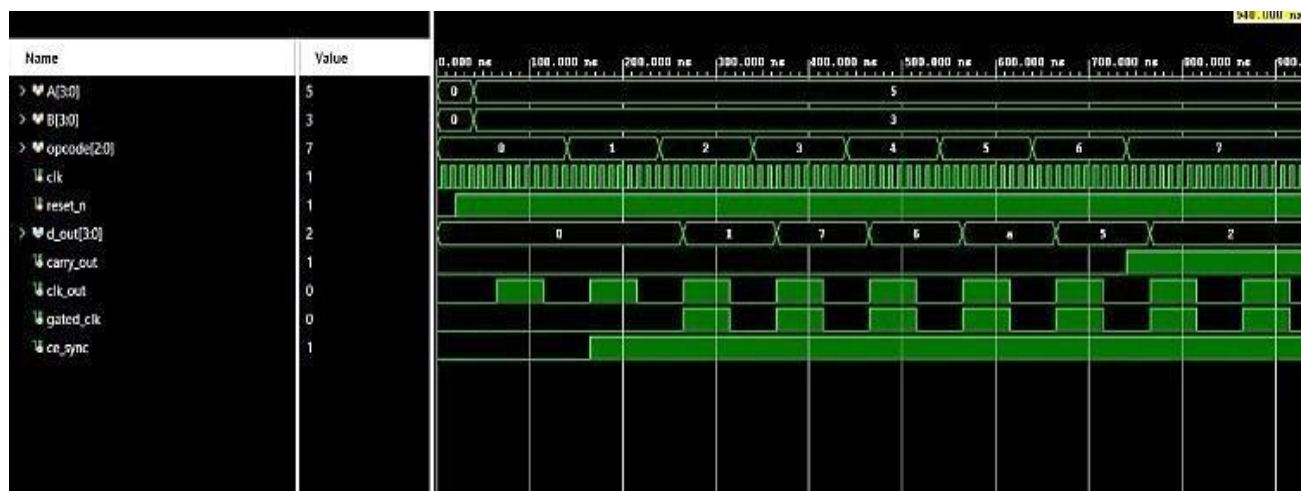


Fig.8: Waveform of 4-Bit ALU with Clock Gating.

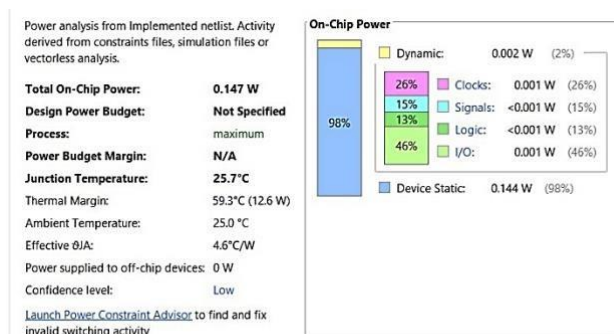


Fig.9: Power on 4-bit ALU with clock gating.

The power analysis, conducted from the implemented netlist and switching activity derived.

From constraint files or simulation vectors, reports that the total on-chip power consumption is 0.147 W. The design power budget has not been specified, and the process condition for the analysis is set to maximum. The power budget margin is not available. In terms of the on-chip power breakdown, the report shows that static power dominates, consuming 0.144 W, which represents 98% of the total on-chip power. The dynamic power is relatively low, at 0.002 W, constituting just 2% of the total. Within the dynamic power components, clocks consume 26% (0.001 W), signals account for 15% (less than 0.001 W), logic uses 13% (less than 0.001 W), and I/O operations contribute 46% (0.001 W) of the dynamic power.

VI. CONCLUSION

In conclusion, the perpetration of a 4-bit ALU with and without timepiece gating on the Nexys Artix 7 FPGA board using Xilinx Vivado demonstrates a significant comparison in power effectiveness and resource optimisation for VLSI circuit design. Without timepiece gating, the ALU operates continuously, leading to advanced dynamic power consumption due to gratuitous timepiece signal switching across all functional blocks, indeed, when not in use.

In contrast, integrating clock gating allows precise activation of only the required portion of the circuit based on control signals, minimising switching activity and reducing overall power consumption. This fashion proves particularly salutary for energy-effective design in battery-operated or lowpower bedded systems. Also, the use of the Artix-7 7 FPGA and Vivado's power analysis tools enables precise monitoring of power operation, helping validate the advantages of timepiece gating in real tackle. Overall, this trial highlights the practical significance of low-power design strategies like timepiece gating in ultramodern VLSI systems.

VII. REFERENCES

- [1] C. Ashok Kumar, B. Madhavi, and K. Lal Kishore, "Enhanced Clock Gating Technique for Power Optimization in SRAM and Sequential Circuits," *Journal of Automation, Mobile Robotics & Intelligent Systems*, vol. 15, no. 3, pp. 23–29, 2021.
- [2] Arpitha V and S. Rangaswamy, "A Survey on Low Power Clock Tree Design," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 7, pp. 45–49, 2021.
- [3] M. Pedram, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," *Journal of VLSI Signal Processing Systems*, vol. 29, no. 1, pp. 123–133, 2001.
- [4] Veena S. Chakravarthi and K. S. Gurumurthy, "Performance Comparison of Various Clock Gating Techniques," *IOSR Journal of VLSI and Signal Processing*, vol. 5, no. 1, pp. 15–20, Jan.–Feb. 2015.
- [5] J. Prasad and K. R. Ramesh, "Comparative Survey of Various Low Power Clock Gating Techniques for ALU Design," *Australian Journal of Basic and Applied Sciences*, vol. 8, no. 10, pp. 231–238, 2014.
- [6] S. Huda, M. Mallick, and J. H. Anderson, "Clock Gating Architectures for FPGA Power Reduction," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 1840–1849, Oct. 2015.
- [7] P. K. Gupta, "Dynamic Clock Gating Techniques for Low Power VLSI Circuits: A Review," *International Journal of Advanced Research in Electronics and Communication Engineering*, vol. 6, no. 3, pp. 21–26, Mar. 2017.
- [8] F. Rivoallon, "Reducing Switching Power with Intelligent Clock Gating," *Xilinx White Paper WP370*, Mar. 1, 2011.
- [9] M. S. Alam and A. Shukla, "Power-Aware Clock Gating for High-Speed CMOS Circuits," *Journal of Low Power Electronics*, vol. 10, no. 4, pp. 340–348, 2014.
- [10] P. J. Schenmakers and J. F. M. Theeuwen, "Clock Gating on RTVHDL," *IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 2, pp. 893–896, 2016.
- [11] R. G. Wagh and R. V. R. L. N. M. Raju, "A Comparative Study of Clock Gating Techniques for Low Power VLSI Design," *International Journal of VLSI Design & Communication Systems*, vol. 7, no. 6, pp. 43–50, 2016.
- [12] A. Y. Son, P. L. Ramkumar, and M. A. D. R. Ghany, "Clock Gating and Power Management Techniques for Low Power Design of VLSI Circuits," *International Journal of Advanced Computer Science and Applications*, vol. 5, no. 3, pp. 17–24, Mar. 2014.
- [13] H. B. Gohar, L. P. Singh, and V. S. Chakravarthi, "Clock Gating Technique for Power Reduction in Digital Circuits: A Review," *International Journal of Electronics and Electrical Engineering*, vol. 9, no. 2, pp. 123–135, 2017.
- [14] A. Chien and L. A. Piazza, "Clock Gating with Multiple Clock Domains for Low-Power and High-Speed Systems," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 6, pp. 1413–1422, June 2012.
- [15] M. V. G. Sharma, S. P. Agarwal, and A. K. Gupta, "Clock Gating Techniques for Low Power Design in Deep Submicron Technologies," *Proceedings of the International Conference on VLSI Design*, pp. 314–319, 2009.