

Lighter: An Open-Source Automatic Clock Gating tool for Dynamic Power Reduction in ASIC

Youssef Kandil

The American University in Cairo
Cairo, Egypt
youssefkandil@aucegypt.edu

Mohamed Shalan

The American University in Cairo
Cairo, Egypt
mshalan@aucegypt.edu

Abstract—This paper introduces **Lighter**, an automatic register clock gating utility designed to primarily reduce dynamic power, and reduce the area of gate-level designs as a subsidiary objective. **Lighter** specifically focuses on enabling automatic clock gating for load-enabled registers by providing a Yosys plugin and complementary technology mapping files. The current version of **Lighter** offers support for both the open-source Sky130 and GF180mcu standard cell libraries, and future versions aim to expand compatibility to include other libraries of other open-source Process Design Kits (PDKs). Through extensive experimentation, **Lighter** has demonstrated remarkable power and area reduction. The maximum power reduction achieved is 43.00%, while area reductions reached up to 58.21%. These results highlight **Lighter**'s significant potential in enhancing efficiency and optimizing resource utilization in digital circuit design.

Index Terms—Dynamic Power Reduction, Clock Gating, Yosys, VLSI Design, Technology Mapping, Sky130, GF180, Open-Source PDK, RTLIL.

I. INTRODUCTION

The semiconductor industry is witnessing a growing need for more powerful digital devices to accommodate the ever-increasing advances in technological applications. As technology advances and feature sizes of modern Process Design Kits PDKs shrink, power optimization becomes a critical aspect of circuit design. Electrical Power reduction in digital systems holds significant importance for several reasons, including enhanced portability, improved reliability, and reduced costs [1]. In portable and battery-operated devices, lower power consumption directly translates to longer battery life, making devices more convenient and energy-efficient. Furthermore, reducing power dissipation improves system reliability by minimizing heat generation, which can degrade component performance and longevity. In large-scale systems, such as data centers, reduced power consumption can significantly lower operating costs by decreasing cooling requirements and energy usage. Consequently, power dissipation has emerged as a critical parameter in the design of low-power VLSI circuits [2].

Digital circuits consume power through two primary components: static and dynamic dissipation. Static power occurs during the idle phase of parts of the circuit due to sub threshold leakage, drain junction leakage and gate leakage due to tunneling, where current flows through the transistor when there is no activity. On the other hand, dynamic power is

associated with circuit-switching activities, which involve the charging and discharging of capacitances while transitioning between logic states [3]. Dynamic power is the dominating component in mature open-source fabrication processes such as Sky130, and it continues to dominate in many cutting-edge fabrication technologies. To better illustrate the comparison between static and dynamic power dissipation, the below table from study [4] presents power dissipation data across various process nodes. For example, in the 45 nm technology node, dynamic power dissipation is approximately 4.720 nW, which is significantly higher than the static power dissipation of 12.991 pW, making dynamic power the dominant component. As the technology scales to smaller nodes, such as 32 nm, 22 nm, and 16 nm, dynamic power dissipation decreases, with values of 2.103 nW, 0.910 nW, and 0.366 nW, respectively. However, static power dissipation increases, indicating a shift in power dynamics. Nonetheless, dynamic power remains the dominant factor in the overall power dissipation.

Technology	Static Power Dissipation (W)	Dynamic Power Dissipation (W)	Total Power Dissipation (W)
45 nm	12.991 p	4.720n	4.733 n
32 nm	15.773 p	2.103 n	2.119 n
22nm	23.782 p	0.910 n	0.934 n
16 nm	46.348 p	0.366 n	0.413 n

Fig. 1. Comparison of Power dissipation on different technology nodes

II. CLOCK GATING

In CMOS circuits, power dissipation occurs during logic switching as fluctuations in input signals lead to the charging and discharging of input, internal, and output capacitors. This phenomenon is especially significant in flip-flops, which are closely tied to the main clock source and experience frequent switching. Clock transitions contribute substantially to overall power consumption, accounting for approximately 15-45% of the total dissipation, which becomes particularly concerning when operating at frequencies in the GHz range [5]. The dynamic power consumption P_{dynamic} can be expressed by the equation (1), highlighting the direct relationship with capacitance C , the square of the voltage V^2 , and the switching frequency f .

$$P_{\text{dynamic}} = C \cdot V^2 \cdot f \quad (1)$$

Clock gating is a process that involves replacing load-enable flip-flops or flip-flops preceded by multiplexers with clock-gated flip-flops. The purpose of clock gates is to restrict the change of the input clock signal of the flip-flops, allowing them to operate only when they are enabled. This prevents the flip-flops from needlessly changing states [5]. By reducing the frequency of state transitions, the dynamic power of the design is automatically reduced. This approach effectively saves a significant amount of power while uncompromising the functionality of the design. The clock gating process is demonstrated in Fig. 2.

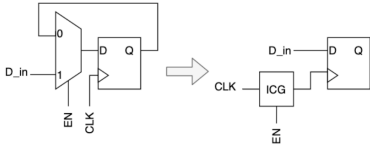


Fig. 2. Flip-Flop Clock Gating

The clock gating process can be carried out at various levels of abstraction in digital circuit design. However, inserting clock gates at the behavioral level, which involves describing the circuit in hardware description languages (HDL) like Verilog or VHDL, can be an exceptionally difficult task. The difficulty is knowing the appropriate scenarios to insert the clock gates while dealing with syntax variations of HDL designs. This process poses a great challenge for both manually and automatically inserting the clock gates.

III. INTRODUCING LIGHTER

Lighter is an open-source tool presented as a Yosys plugin [6], that aims to enhance dynamic power reduction optimization for digital circuits. This tool offers support for open-source Process Design Kits (PDKs) such as SKY130 and GF180 [7] [8]. Lighter addresses the aforementioned challenges by providing a fully automated procedure for flip-flop clock gating while allowing designers to selectively apply the optimization to specific parts of their designs. For designers using different PDKs, Lighter can be easily leveraged by incorporating additional technology mapping files.

A. Overview

Lighter utilizes the APIs offered by Yosys to convert the design into the Register Transfer Level Intermediate Language (RTLIL) abstraction format, which is a logical level of abstraction implemented by Yosys. The RTLIL format represents the design in terms of generic high-level representations of digital blocks known as RTLIL cells [6]. These cells are intentionally designed to be generic, independent of the specific complexities of the HDL design.

Lighter performs its optimization process on the RTLIL level to effectively eliminate the irregularities and specificity inherent in each design. Instead, it treats each design as an abstract graphical representation comprising interconnected nodes. Consequently, the task transitions from being a parsing problem to a graph search problem.

B. Process

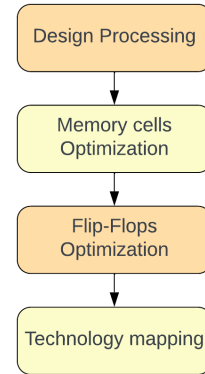


Fig. 3. Clock Gating Process

Lighter utilizes a set of Yosys APIs that collectively execute design optimizations to prepare the design for clock gating. The process, as depicted in Figure 2, primarily focuses on technology mapping to transform load-enabled registers into clock-gated ones. This process can be further presented in the following four steps:

- **Design Processing:** In this step, the HDL design is converted into the RTLIL format where the optimization steps should be performed.
- **Memory Cells Optimization:** The Front-end synthesizer of Yosys identifies register files and memory-like sub-designs and replaces them with RTLIL Memory representations. Through the use of the `memory_collect` API, multiple memory cells are combined into a single generic multiport memory cell. Subsequently, the `memory_map` API converts the memory cells into flip-flops combined with basic logic cell representations.
- **Flip-Flops Optimization:** At this stage, the design comprises regular RTLIL logic cells. To simplify the technology mapping process, Lighter aims to establish a single type of cell to be substituted with clock-gated flip-flops. Thus, it searches for flip-flops preceded by multiplexers and replaces them with load-enabled flip-flops, ensuring functional equivalence.
- **Technology Mapping:** Lighter then maps all load-enabled flip-flops by pre-defined clock-gated flip-flop representations specific to the target PDK given as technology mapping files.

This completes the clock gating process.

IV. PERFORMANCE ANALYSIS

In this section, the analysis process of Lighter is thoroughly demonstrated with a summarised average reduction demonstrated in the following table. Additionally, six designs that underwent clock-gating optimization were functionally tested, and successfully passed the functional tests, validating the effectiveness of the Lighter tool in producing output

designs that maintain full functionality while achieving power reduction.

TABLE I
AVERAGE REDUCTIONS

Average Power Reduction %	Average cell Reduction %
27.14%	19.48%

A. Power Calculation Methodology

To evaluate the performance of Lighter, we measured the dynamic power reduction done on 20 designs which showed a significant improvement. We utilized OpenSta's [9] report_power API for the power calculation process and introduced an accurate approximation methodology for measuring dynamic power.

1) *Measuring the total power consumption before clock-gating P_{Before}* : The power consumption of the entire design P_{All} was measured at an activity factor of $\alpha_{Normal} = 0.1$, representing the assumption that all cells except flip-flops are active 10% of the time. The power consumption of the flip-flops P_{FF} is calculated at an activity factor of $\alpha_{High} = 1.0$ since they are connected directly to the clock source that has full activity. Using the obtained measurements, the total power consumption before clock gating was calculated using the following equation:

$$P_{Before} = P_{All} * \alpha_{Normal} - P_{FF} * \alpha_{Normal} + P_{FF} * \alpha_{High} \quad (2)$$

2) *Measuring the total power consumption after clock-gating P_{After}* : The power consumption of the entire design P_{All} excluding the added clock gates P_{CG} was measured at an activity factor of $\alpha_{Normal} = 0.1$, representing the assumption that all cells except flip-flops are active 10% of the time. The power consumption of the clock-gated flip-flops P_{CGFF} is now measured at a reduced activity factor of $\alpha_{Low} = 0.05$, reflecting the industry-standard assumption that flip-flops are active only 5% of the time after clock-gating. The power consumption of the not clock-gated flip-flops P_{NFF} and the clock gate cells P_{CG} is measured at a high activity factor of $\alpha_{High} = 1.0$ since they are directly connected to the clock. Using the obtained measurements, the total power consumption after clock gating was calculated using the following set of equations:

$$P_{CG} = P_{CG\alpha_{High}} - P_{CG\alpha_{Normal}} \quad (3)$$

$$P_{CGFF} = P_{CGFF\alpha_{Low}} - P_{CGFF\alpha_{Normal}} \quad (4)$$

$$P_{NFF} = P_{NFF\alpha_{High}} - P_{NFF\alpha_{Normal}} \quad (5)$$

$$P_{After} = P_{All\alpha_{Normal}} + P_{CG} + P_{CGFF} + P_{NCG} \quad (6)$$

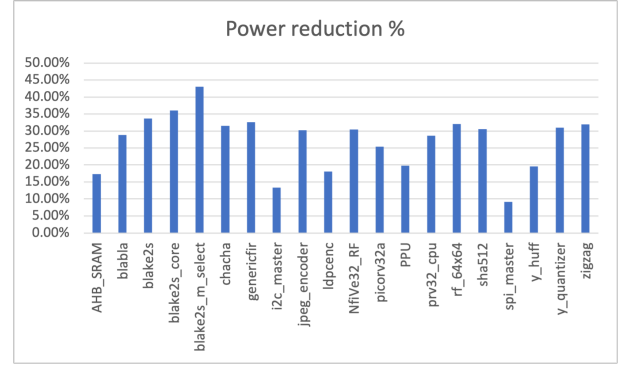


Fig. 4. Power Reduction Summary

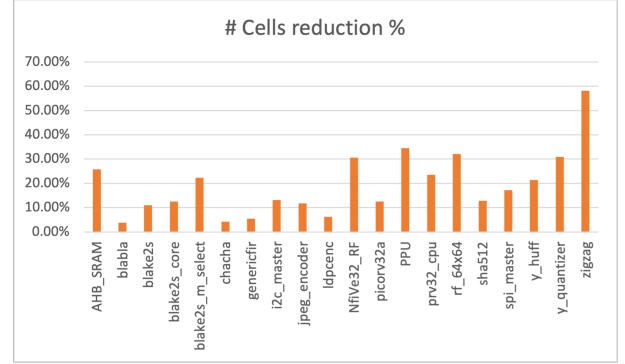


Fig. 5. Area Reduction Summary

TABLE II
POWER REDUCTIONS

Max Power Reduction %	Min Power Reduction %
43.00%	9.16%

TABLE III
AREA REDUCTIONS

Max cell Reduction %	Min cell Reduction %
58.21%	3.72%

B. Power Reduction

By assessing the 20 different designs, each exhibiting distinct sizes and functionalities, Lighter demonstrated a significant reduction in power consumption. On average, Lighter showcased a 27.14% decrease and a maximum of 43%. A visual representation of power reductions is revealed in Fig. 4.

C. Area Reduction

Through the series of optimization steps outlined in the power calculation methodology, Lighter also displayed a remarkable reduction of the overall design size, with an average reduction of 19.48% recorded in the tests. The design modification behind this reduction is the substitution of multiplexers inside a register with a clock gate outside. Instead of implementing a multiplexer for each flip-flop in a register, Lighter utilizes a clock gate connected to each register. This

clock gate performs the same functionality as the traditional multiplexer. Accordingly, Lighter effectively minimizes the number of components required, leading to enhanced area efficiency in digital circuit designs.

V. CONCLUSION

This paper introduced Lighter, an automated clock gating tool aimed at reducing dynamic power and minimizing area in VLSI designs. The Yosys framework integration and the support for Sky130 and GF180 standard cell libraries demonstrated significant power and area reductions through intensive testing. The tool's efficiency is essential for low-power designs, contributing to the broader effort of making VLSI designs more energy-efficient while maintaining simplicity and ease of use.

VI. FUTURE WORK

Future work will focus on expanding Lighter's compatibility with other standard cell libraries. Additionally, we aim to explore implementing clock gating techniques not only for flip-flop registers but also for latch-based registers, broadening the scope of optimization across diverse designs.

VII. ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Google's gift. Their support has been instrumental in facilitating our research and development efforts. This project is made available as an open-source initiative, and the repository can be accessed at <https://github.com/AUCOHL/Lighter>.

REFERENCES

- [1] A. Dhanotiya and V. Sharma, "Power reduction in digital VLSI circuits," *International Journal of Research in IT, Management and Engineering*, vol. 4, no. 6, pp. 13-23, 2014.
- [2] Nandita Srinivasan, Navamitha S. Prakash, Shalakra D., Sivaranjani D., Swetha Sri Lakshmi G., B. Bala Tripura Sundari, "Power Reduction by Clock Gating Technique," *Procedia Technology*, Volume 21, 2015, Pages 631-635, ISSN 2212-0173, <https://doi.org/10.1016/j.protcy.2015.10.075>.
- [3] T. Chindhu S. and N. Shanmugasundaram, "Clock Gating Techniques: An Overview," 2018 Conference on Emerging Devices and Smart Systems (ICEDSS), Tiruchengode, India, 2018, pp. 217-221, doi: 10.1109/ICEDSS.2018.8544281. keywords: Clocks;Logic gates;Latches;Flip-flops;Switches;Power demand;Delays;AGFF;Clock gating;data driven;dynamic power;flipflop based clock gating;LACG;latch based clock gating;switching activity;synthesis based CG
- [4] S. Saun and H. Kumar, "Design and performance analysis of 6T SRAM cell on different CMOS technologies with stability characterization," *IOP Conference Series: Materials Science and Engineering*, vol. 561, no. 1, p. 012093, Oct. 2019. doi:10.1088/1757-899x/561/1/012093
- [5] E. Jolly, "A comparative study and review of different clock gating techniques and their application," *Academia.edu*, https://www.academia.edu/83449612/A_Comparative_Study_and_Review_of_Different_Clock_Gating_Techniques_and_their_Application?auto=download
- [6] C. Wolf, J. Glaser, and J. Kepler, "Yosys—a free Verilog synthesis suite," in *Proc. 21st Austrian Workshop on Microelectronics (Austrochip)*, vol. 97, 2013.
- [7] "Welcome to Skywater Sky130 PDK's documentation!," Welcome to SkyWater SKY130 PDK's documentation! - SkyWater SKY130 PDK 0.0.0-369-g7198cf6 documentation, <https://skywater-pdk.readthedocs.io/en/main/>
- [8] "Welcome to globalfoundries 0.18um 3.3V/(5V)6V MCU PDK's documentation!," Welcome to GlobalFoundries 0.18UM 3.3V/(5V)6V MCU PDK's documentation! - GlobalFoundries GF180MCU PDK 0.0.0-111-gde3240d documentation, <https://gf180mcu-pdk.readthedocs.io/en/latest/> (accessed Sep. 21, 2024).
- [9] The-OpenROAD-Project, "The-openroad-project/opensta: Opensta engine," GitHub, <https://github.com/The-OpenROAD-Project/OpenSTA> (accessed Sep. 30, 2024).
- [10] N. Raghavan, V. Akella, and S. Bakshi, "Automatic insertion of gated clocks at register transfer level," *Proceedings Twelfth International Conference on VLSI Design*. (Cat. No.PR00013), pp. 48–54, 1999. doi:10.1109/icvd.1999.745123
- [11] "Automatic sequential clock gating with PowerPro," Siemens Resource Center, <https://resources.sw.siemens.com/en-US/white-paper-overview-of-automatic-sequential-clock-gating> (accessed Sep. 30, 2024).
- [12] N. Agarwal and N. J. Dimopoulos, "Automated power gating of registers using Codel and FSM branch prediction," Springer-Link, https://link.springer.com/chapter/10.1007/978-3-540-73625-7_31 (accessed Sep. 30, 2024).