UNIVERSIDADE FEDERAL DE PELOTAS
Centro de Desenvolvimento Tecnológico
Programa de Pós-Graduação em Computação

Dissertação

Static CMOS Complex Gates: Electrical Investigation of Design Strategies

Henrique Caldas Kessler

Pelotas, 2022

Henrique Caldas Kessler

Static CMOS Complex Gates: Electrical Investigation of Design Strategies

Dissertação apresentada ao Programa de Pós-Graduação em Computação do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação.

Advisor: Prof. Dr. Marcelo Schiavon Porto

Coadvisors: Prof. Dr. Vinícius Valduga de Almeida Camargo
Prof. Dr. Leomar Soares da Rosa Jr.

Pelotas, 2022

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação na Publicação

K42s Kessler, Henrique

Static CMOS complex gates : electrical investigation of design strategies / Henrique Kessler ; Marcelo Schiavon Porto, orientador ; Vinícius Valduga de Almeida Camargo, Leomar Soares da Rosa Jr., coorientadores. — Pelotas, 2022.

62 f. : il.

Dissertação (Mestrado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2022.

1. Static CMOS complex gates. 2. Cell design automation. 3. Transistor network. 4. Electrical simulation. I. Porto, Marcelo Schiavon, orient. II. Camargo, Vinícius Valduga de Almeida, coorient. III. Jr., Leomar Soares da Rosa, coorient. IV. Título.

CDD : 005

Elaborada por Maria Inez Figueiredo Figas Machado CRB: 10/1612

Henrique Caldas Kessler

Static CMOS Complex Gates: Electrical Investigation of Design Strategies

Dissertação aprovada, como requisito parcial, para obtenção do grau de Mestre em Ciência da Computação, Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas.

Data da Defesa: 25 de fevereiro de 2022

Banca Examinadora:

Prof. Dr. Marcelo Schiavon Porto (orientador)
Doutor em Computação pela Universidade Federal do Rio Grande do Sul.

Prof. Dr. Felipe de Souza Marques
Doutor em Computação pela Universidade Federal do Rio Grande do Sul.

Prof. Dr. Sergio Bampi
Doutor em Engenharia Elétrica pela Stanford University.

AGRADECIMENTOS

Agradeço aos meus orientadores Marcelo Porto, Vinícius Camargo e Leomar da Rosa Jr. pela atenção durante o desenvolvimento de todo o trabalho. A qualidade deste mestrado não seria a mesma sem o apoio de vocês, obrigado.

ABSTRACT

KESSLER, Henrique Caldas. **Static CMOS Complex Gates: Electrical Investigation of Design Strategies**. Advisor: Marcelo Schiavon Porto. 2022. 62 f. Thesis (Masters in Computer Science) – Technology Development Center, Federal University of Pelotas, Pelotas, 2022.

Recent developments in electronic design automation tools vastly reduce the design cost of supergates, enabling an alternative approach to logic synthesis. Despite many design strategies targeting the transistors network in supergates, their comparisons are often limited to metrics such as the number of transistors used or circuit total stack, lacking an in-depth electrical evaluation. This thesis uses an electrical characterization framework to study multiple supergate design strategies. A study on the 3982 logic functions of the 4 input P-class shows that topologies that optimize both pull-up and pull-down networks individually presented better overall electrical characteristics. The results also suggest that reducing the logic gate stack or the number of transistors does not necessarily lead to better performance. Also, a strong dependency between the effectiveness of a supergate design methodology and the logic function is found. The evaluated supergates designs did not possess a defined transistor reordering technique. In this thesis, a well-established reordering algorithm is evaluated and a proposed modification is presented. Observing supergates with different results from the baseline algorithm, the proposed algorithm produced gates with smaller power dissipation and critical delay in over 60% of the studied cases. It is also observed a lack of different transistor sizing techniques in works that use supergates. They are often limited to using minimum transistor dimensions or the Logical Effort technique. In this thesis, a methodology to adapt the Logical Effort technique for low-power applications is proposed. Results show significant improvements on up to 99.9% of the studied cases in power-performance trade-off across multiple simulation environments. Comparing supergates with technology-mapped circuits on small logic functions results shows that supergate-based designs reduce the average power dissipation in 84.4% of the studied cases. Despite the supergate design increasing in average the circuit critical delay by 5.8%, it achieves better power-delay-product in 2823 (70.9%) of the 3982 studied logic functions. The reduction of logic levels is the main factor for gains obtained with supergates due to the glitch power reduction. Applying supergates to a circuit with more than 800 logic gates, small gains in both power dissipation and critical delay can be achieved.

Keywords: static CMOS complex gates. cell design automation. transistor network. electrical simulation.

RESUMO

KESSLER, Henrique Caldas. **Portas Complexas CMOS Estáticas: Investigação Elétrica em Estratégias de Projeto**. Orientador: Marcelo Schiavon Porto. 2022. 62 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2022.

Desenvolvimentos recentes em ferramentas de automação de projeto eletrônico reduzem enormemente o custo de projeto de supergates, permitindo uma abordagem alternativa à síntese lógica. Apesar de muitas estratégias de projeto visando a rede de transistores em supergates, suas comparações são frequentemente limitadas a métricas como o número de transistores usados ou quantidade de transistores em série, carecendo de uma avaliação elétrica aprofundada. Esta dissertação utiliza uma ambiente de caracterização elétrica para estudar múltiplas estratégias de projeto de supergates. Um estudo sobre as 3982 funções lógicas da P-class de 4 entradas mostra que topologias que otimizam redes pull-up e pull-down individualmente apresentam melhores características elétricas. Os resultados também sugerem que a redução do número de transistores em série de portas lógicas ou o número de transistores não leva necessariamente a um melhor desempenho. Além disso, é encontrada uma forte dependência entre a eficácia de uma metodologia de projeto de supergate e a função lógica. Nesta dissertação, um algoritmo de reordenamento de transistores bem estabelecido é avaliado e uma proposta de modificação é apresentada. Observando o reordenamento, o algoritmo proposto projetou portas com menor dissipação de potência e atraso crítico em mais de 60% dos casos estudados. Observa-se também a falta de diferentes técnicas de dimensionamento de transistores em trabalhos que utilizam supergates. Nesta dissertação, é proposta uma metodologia para adaptar o Logical Effort para redução de potência. Os resultados mostram melhorias significativas em até 99,9% dos casos estudados na relação ao produto potência-atraso em multiplos ambientes de simulação. Comparando supergates com circuitos projetados através de mapeamento tecnológico funções lógicas pequenas, os resultados mostram que projetos baseados em supergates reduzem a dissipação de potência média em 84,4% dos casos estudados. A redução dos níveis lógicos é o principal fator para os ganhos obtidos com supergates devido à redução da potência do glitch. Aplicando supergates a um circuito com mais de 800 portas lógicas, pequenos ganhos tanto na dissipação de potência quanto no atraso crítico podem ser alcançados.

Palavras-chave: portas complexas CMOS estáticas. projeto automático de celulas lógicas. rede de transistores. simulação elétrica.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AES	advanced encryption standard
AIG	and-inverter graph
BDD	binary decision diagram
CMOS	complementary metal oxide semiconductor
DUT	design under test
EDA	electronic design automation
FIN	fan-in
FO	fan-out
IC	integrated circuit
ISOP	irredundant sum-of-products
NMOS	negative-channel metal oxide semiconductor
NSP	non-series-parallel
PDP	power-delay-product
PDK	process design kit
PMOS	positive-channel metal oxide semiconductor
PWL	piece-wise linear
S-BOX	substitution box
SCCG	static CMOS complex gate
SP	series-parallel
SPICE	simulation program with integrated circuit emphasis
STA	static timing analysis
VLSI	very large-scale integration

CONTENTS

1 INTRODUCTION

The use of integrated circuits (ICs) is continuously growing around the world. New applications and stricter specifications raise the demand for more complex and specific ICs, however, these circuits must also comply with restrictions in the design costs and the time-to-market. Electronic design automation (EDA) tools are widely used to tackle this problem. The most employed EDA-friendly methodology is the Standard Cell design, which makes use of a pre-designed library of logic functions, resulting in a highly automated design flow.

However, due to the limited number of logic functions, the IC must be mapped to a matching set of logic gates already defined by the technology library. Authors, who defend the use of small-sized libraries, claim that logic synthesizers are more efficient when using simple cells, resulting in a better design (RICCI; DE MUNARI; CIAMPOLINI, 2007; MATOS; CARRABINA; REIS, 2018). Other authors claim that this semi-custom technique may result in a less efficient implementation of the circuit as the designer is limited to the technology library. Merging logic gates or using a library-free design could further optimize the results obtained during the logic synthesis (MARQUES et al., 2007; MACHADO et al., 2012; OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019; ALBINAGORTA et al., 2019).

Recent developments in EDA tools, presented in (OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019; ALBINAGORTA et al., 2019), show a standard cell-like approach, in which the logic synthesis is optimized for a much larger set of logic functions, and the gates for each one are designed automatically according to the designer's need. This library-free design technique, also referred to as design on-the-fly in this work, enables the use of complex gates, which reduces the number of transistors and the average length of wires. Regarding circuit performance, the authors also claim that this approach results in reduced static and dynamic power. Other applications of complex gates may include the reduction of the soft error rate on the circuit (REIS et al., 2020) and the improvement of the circuit resilience to process variation and aging effects, like BTI (DUAN; ZWOLINSKI; HALAK, 2018; KESSLER et al., 2020).

The design on-the-fly approach allows multiple structures for a logic function implementation using Static CMOS Complex Gates (SCCG). For instance, a complex gate transistor network may consist of only series-parallel (SP) associations (MARTINS et al., 2010; OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019) or they could also benefit from non-series-parallel (NSP) associations (ROSA JUNIOR et al., 2006; POSSANI et al., 2015).

In the literature, the number of transistors or stack (quantity of transistors associated in series) is commonly used as a metric to compare different designs (ROSA JUNIOR et al., 2006; MARQUES et al., 2007; MACHADO et al., 2013; POSSANI et al., 2015; OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019). The optimization of a topological metric may not imply better electrical behavior. Considering this situation, this work aims to evaluate how the transistor network generation methods and their topological metrics correlate with performance metrics on a lower abstraction level, and this is achieved through SPICE simulations. Four complex gates generation techniques were evaluated in this work, three that are contained in the SwitchCraft framework (CALLEGARO et al., 2010) and Kernel Finder (KF) proposed by Possani (POSSANI et al., 2015).

To evaluate the supergate design, two main experiments were conducted. An electrical evaluation on the 4 input P-Class (SASAO, 2012) (consists of 3982 logic functions) and a gate-merging (post-processing) on a commercially mapped circuit. Due to a large number of experiments, an automated simulation framework for electrical comparison between logic equivalent circuits was developed.

1.1 Objectives and Contributions

The goal of this thesis is to evaluate the electrical characteristics of automatically designed supergates to provide better guidance for works on a higher abstraction level. The main study uses planar bulk CMOS transistors and focuses on design strategies proposed in the literature. The contributions of this thesis can be summarized as follows.

1. An automatic SPICE simulation methodology is proposed, allowing the characterization and the comparison of equivalent circuits;
2. An electrical comparison of four supergate design strategies at the SPICE level;
3. A discussion regarding transistor reordering algorithms and their impact on the electrical characteristics of supergates;
4. A study on discrete transistor sizing based on Logical Effort;
5. A discussion and evaluation of the possibility of using supergates on the design of standard-cell-based VLSI circuits.

1.2 Thesis Organization

This thesis is organized as follows. Chapter 2 details the electrical simulation methodology and environment. Chapter 3 describes and compares the supergates design methodologies. Chapter 4 evaluates the impact of transistor reordering techniques on the electrical characteristics of supergates. Chapter 5 explores transistor sizing values on supergates. Chapter 6 compares a standard-cell library mapped circuit and a proposed post-processing methodology to insert supergates. Chapter 7 presents the conclusions, final remarks, and direction for future works.

2 ELECTRICAL SIMULATION METHODOLOGY

In this thesis, to properly evaluate multiple design strategies, an automated methodology is proposed. Fig. 1 illustrates a flowchart with the steps carried out. Given a logic function, two modules will use it as input, the simulation framework, and the supergate design technique. The simulation framework requires the logic function to generate the electrical stimulus and the SPICE file environment ready for simulation. More details on the simulation framework are presented in the next section.

As for the supergate design strategy, this thesis does not propose a design strategy but focuses on evaluating already proposed techniques. Chapter 3 will discuss which supergates were evaluated. As an optional supergate optimization step, Chapter 4 discusses the application of two transistor reordering algorithms. Lastly, for transistor sizing this work uses mostly Logical Effort (SUTHERLAND et al., 1999), however a study on alternative sizing values is done in Chapter 5.

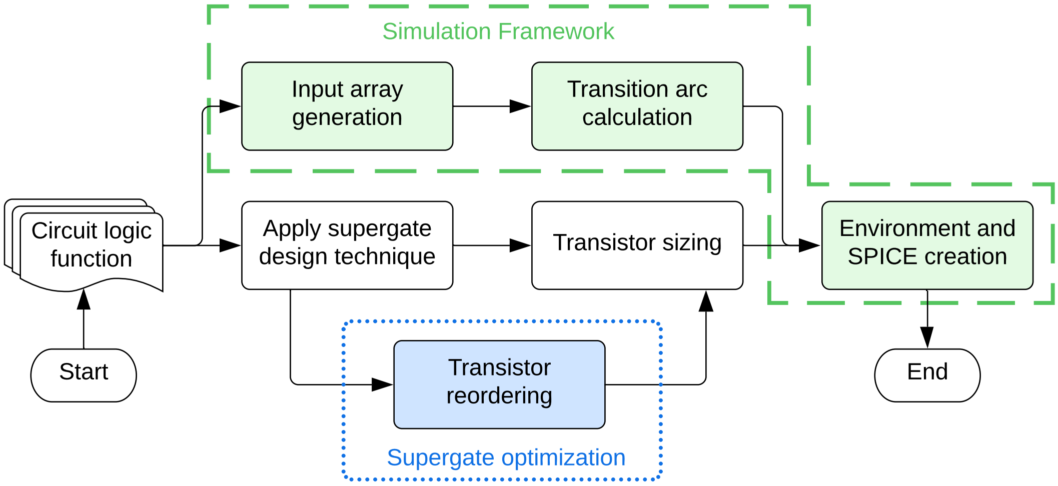


Figure 1 – Proposed methodology for an automatic comparison of multiple supergate design strategies. The dashed green box contains the steps taken on our proposed simulation framework. The dotted blue box is a proposed optional step to optimize a supergate design by reordering its transistors.

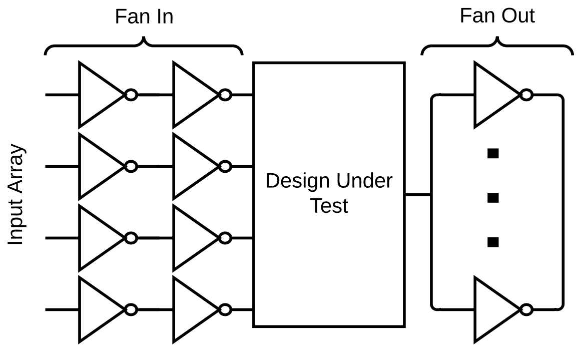


Figure 2 – Generic representation of the simulation environment. The input inverter size, and the number of inverters in the output change according to the chosen environment.

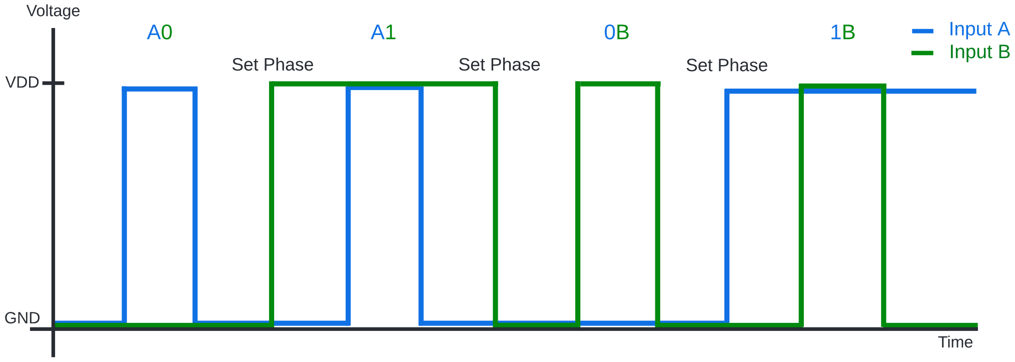
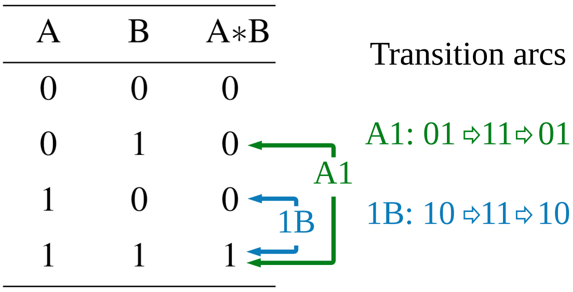


Figure 3 – All input values for a logic function with 2 inputs. Non-overlapping lines are illustrative. The description "A0" indicates that the input "A" is changing values while the other inputs are stable at the specified value.

2.1 Simulation Framework

The environment in which the design under test (DUT) is submitted is illustrated in Fig. 2. The input signals are built as rectangular pulses, however, to simulate a realistic input slope, two inverters associated in series are added before each of the gate's inputs. Regarding the circuit load, minimum sized inverters associated in parallel are connected to the DUT's output. The drive strength of the inverters connected to the inputs (FIN) and the number of inverters connected to the outputs (FO) vary in the different studies presented as the simulation environment may affect the results.

The input stimuli applied to the design under test contain all possible input state transitions of the implemented logic function. This choice was made to assure the capture of all relevant electrical behavior of the DUT. Once defined the circuit input array, for each literal in the logic function, a piece-wise linear (PWL) voltage source reproduces the input array. This PWL voltage source changes value every one nanosecond (1 GHz). Fig. 3 illustrates the input array made given a two-input NAND gate. The interval of one nanosecond is a conservative value to ensure that all logic gates correctly propagate their signal, allowing the chosen metrics to be evaluated. An arc consists of a low-high and a high-low transition of a single input. To achieve this behavior, a Set



A	B	A∗B
0	0	0
0	1	0
1	0	0
1	1	1

Figure 4 – Transition arcs for a NAND2.

phase is required. This step prepares all variables accordingly to the next arc and it is not utilized to electrically characterize the DUT.

Measurements regarding the circuit delay are created when the input value is identified as a transition arc. A transition arc occurs when a single bit flip in the input causes a change in the logic gate output. These arcs are identified by sweeping the truth table of the respective implemented logic function. Fig. 4 shows the transition arcs for a NAND2, changing the value of 'A' while 'B' is fixed on '1' changes the logic function output, 'A1' is identified as a transition arc. The delay measurement is built using the logic function, therefore allowing to validate if the logic gate is properly designed since the circuit output simulation result has to match the transition arc value.

The propagation delay, the static, and total power dissipation are measured in the transient simulation. As Fig. 5 illustrates, the propagation delay trigger is at the DUT input (after the buffer) when 50% of the supply voltage is reached. The propagation delay target is at the DUT output (before the load) also at 50% of the supply voltage, using this methodology only the DUT delay is considered. The static power is computed as the power dissipation after the circuit output is stable, and the average of the entire simulation is referred to as the average power. Both power measurements are made exclusively on the VDD voltage source, thus considering only the DUT power dissipation. Fig. 6 illustrates the described measures, propagation delay is the time between the input and the output to reach 50% of the nominal voltage, the average power is the power dissipation during the arc (does not include the Set phase), and static power is power dissipation while the DUT is stable.

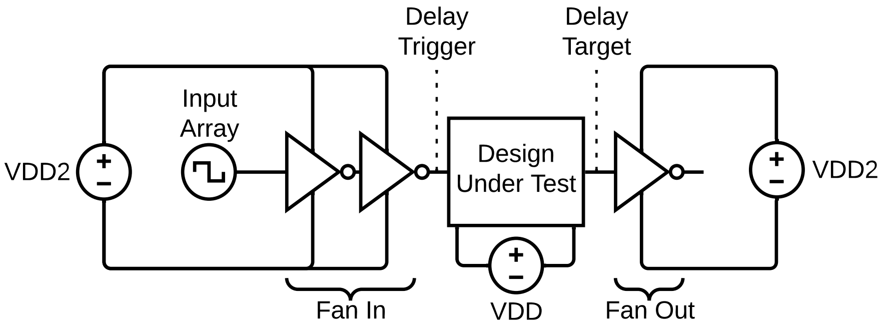


Figure 5 – Detailed simulation environment, containing voltage sources and delay measurements.

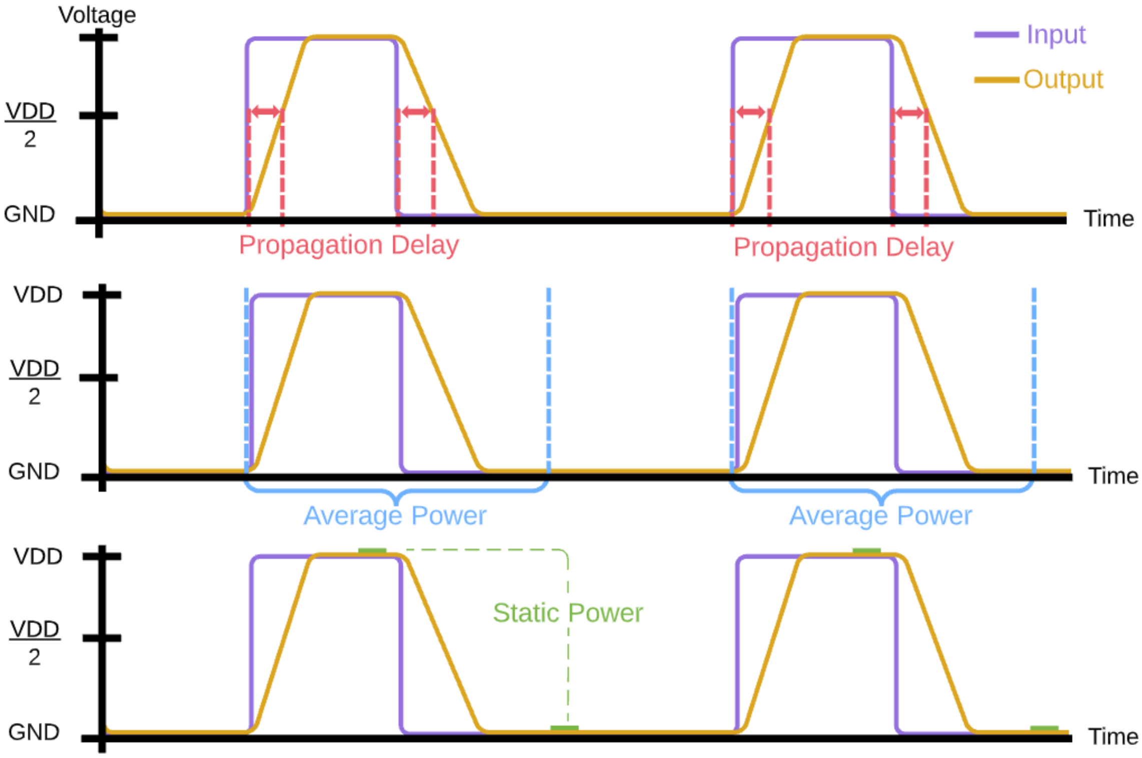


Figure 6 – Visual representation of the SPICE measurements made. Propagation delay is defined as the time the output takes to reach half VDD from the input change. Average power measures the power dissipation during input change and static power is the power dissipation while the circuit is stable.

3 SWITCH NETWORK DESIGN METHODS

A logic function with 4 inputs can generate up to 65 536 logic functions. It is possible to obtain a reduced set taking advantage of equivalent functions, where a possible operation is the permutation of inputs (P-Class), downsizing this set to 3982 logic functions (SASAO, 2012). An instance of two equivalent functions is exemplified in Table 1. It can be seen that each minterm on the output is the same, turning both F_1 and F_2 equivalent by input permutation. The 4 input P-Class was chosen due to being a considerably large set of logic functions allowing the study on multiple cases for each design strategy. Classes with more than 4 inputs may not be suitable for an efficient single supergate design due to the loss of electrical characteristics as the transistor stack grows (transistors associated in series). Increasing the number of stacked transistors, increases its delay, requiring an over-sizing, increasing the circuit area cost and tends to increase the power-dissipation.

3.1 SwitchCraft

SwitchCraft (CALLEGARO et al., 2010) is a framework that contains a collection of algorithms for an automatic generation of transistor netlist. In this chapter, three design strategies are evaluated.

Table 1 – Example of two *P-equivalent* functions.

AB	F_1	BA	F_2
00	0	00	0
01	0	10	**1**
10	**1**	01	0
11	0	11	0

$$F_3 = ab + de + ace + bcd \tag{1}$$

$$\downarrow$$

$$F_3 = a(b + (ec)) + (d(e + (bc))) \tag{2}$$

The first approach creates the gate directly from the logic function, an implementation based on Good Factor (BRAYTON, 1987). This method aims to reduce the number of literals present on the logic function, therefore, reducing the numbers of transistors; an example is shown in the optimization of (1) to (2). The resulting logic gates use the factorized expression to create the pull-down network. The pull-up network is built to be topologically and logically complementary, an example of a resulting transistor network is illustrated in Fig. 7 (b). This approach, also referred to as FAC-PD in this document, is implemented in SwitchCraft with the command *direct network from expression*.

The second approach uses the built-in factorization method from SwitchCraft. This approach aims to reduce the number of transistors in the logic gate. Both pull-up and pull-down planes are optimized individually, which results in a logically complementary gate but not necessarily topologically complementary. Fig. 7 (c) illustrates a resulting transistor network. This approach, also referred to as FAC in this document, is implemented with the command *factorization method*.

The third approach is built to reach a network with the minimum theoretical transistor stack for the logic function. This is achieved through a specific way to construct a binary decision diagram (BDD), as presented in (ROSA et al., 2009). Gates created with this procedure can hold a NSP association of transistors (bridge-networks) and also optimizes both pull-up and pull-down switch networks. An example of a bridge network is illustrated in Fig. 7 (d). This approach, also referred to as BDD in this document, is implemented in SwitchCraft with the command *from BDD - minimum stacks*.

All these previously cited algorithms are employed to create logic gate instances on the SwitchCraft framework. From each logic gate, data — such as transistor count, shortest and longest paths, short-circuit analysis, and the logic function implemented on each plane — is extracted.

3.2 Kernel Finder

Using a graph-based method, Kernel Finder (KF) (POSSANI et al., 2015) is currently the state-of-art regarding transistor count reduction, Fig. 7 (d) illustrates a NSP network when using KF. Using an Irredundant Sum-of-Products (ISOP) as input creates an undirected graph. Using this graph two modules are employed, kernel identification and the switch network composition. During the kernel identification, it searches for

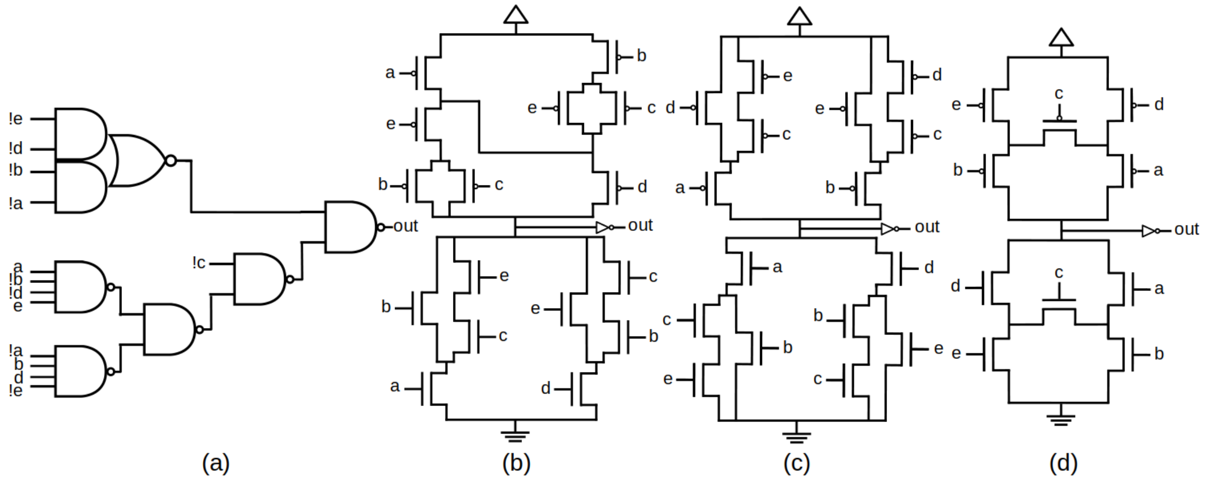


Figure 7 – Logic function implementation for (1) using four different approaches, (a) standard cell based, supergate designed with (b) pull-down optimization using FAC-PD technique, (c) individually factorization in both planes with FAC method, and (d) using non-series-parallel structures.

specific patterns on the graph. If the pattern (kernel) is logically equivalent and satisfies the rules for non-series-parallel or series-parallel kernels it is saved for the next step.

Once the kernels are identified, the switch network composition module optimizes the kernels by sharing their graphs edges. This module also allows the user to define a design rule such as a maximum number of stacked transistors. Two design approaches are recommended by the authors, the Bounded Mode with 4 stacked transistors for performance and the Unbounded Mode for maximum transistor count reduction and smaller circuit area. In this chapter, the Unbounded Mode is evaluated.

3.3 Results and Discussions

In this section, the results from applying the four previously defined logic network generation methods using the simulation framework are detailed and discussed. All logic gates in this study were designed with the 45nm CMOS TSMC TT process design kit (PDK). Transistor sizing is made through the Logical Effort technique using a PMOS/NMOS ratio of 1.32. Transistor order is as designed by the technique (unchanged). The simulation environment consists of FIN8 and FO4.

Table 2 summarizes the results obtained in the experiments with the 4 input P-Class. The FAC logic gates were used as a baseline so that the metrics obtained for a given gate were normalized by its corresponding gate designed with FAC. Values above 1.000 represent a higher (worse) value for the metric, e.g. 1.050 delay means 5.0% higher delay and 0.950 means an improvement of 5.0%. The values presented in the table are the mean of the normalized metric over the entire 4 input P-Class gates.

Table 2 shows that FAC methodology achieves better timing, power, and area re-

Table 2 – Electrical and topological metrics for the 4 input P-Class set of functions with results normalized to FAC method.

	FAC	FAC-PD	BDD	KF
Average delay	1.000	1.249	1.051	1.057
Maximum delay	1.000	1.224	1.028	1.055
Minimum static power	1.000	1.291	1.223	1.000
Average static power	1.000	1.239	1.122	1.138
Average power	1.000	1.248	1.060	1.169
Power-Delay-Product	1.000	2.320	1.106	1.248
Power-Delay-Area-Product	1.000	2.557	1.143	1.234
Nº of transistors	1.000	0.999	1.012	0.937
Area estimation*	1.000	1.094	1.029	0.969
Pull-up stack	1.000	1.021	0.980	-
Pull-down stack	1.000	1.073	0.980	-

* Sum of transistors width.

sults compared to the other methodologies, while FAC-PD presented the worst results in all electric measurements. Also, reducing the number of transistors in a gate or having smaller transistors stacks does not imply better electrical characteristics. This situation can be seen with the number of transistors reduced from the KF and BDD to the FAC and with the stack reduction from the FAC to the BDD. The results show that complex cells that optimize both planes individually (FAC, BDD, and KF) exhibit significantly better electrical characteristics.

In order to estimate the area cost of each design strategy, the sum of the transistor width is utilized. Comparing the techniques, KF achieves the smallest circuit area, as expected from using the Unbonded Mode. BDD uses the minimum transistor stack, to achieve these criteria it may abuse redundant parallel-associated transistors, resulting in a slight increase in area. It is important to note that FAC, BDD, and KF logic gates optimize their planes individually. Such cells may have an increased cost in the layout as one cannot assure the presence of the same Euler Path for both pull-down and pull-up networks (UEHARA et al., 1979). The layout implications of the transistor network are out of the scope of this thesis, however, works, such as (SMANIOTTO et al., 2017; CARDOSO et al., 2018, 2020), aim to automatically generate complex gate layouts. The techniques presented in these works can find or generate Euler paths to improve the logic gate area efficiency.

Fig. 8 shows the relationship between average propagation delay and power (closer to the origin is better) for the 4 input P-class set of functions. The two axis are presented in a log-log format for better visualization. Also, the results for each function are normalized to the same function obtained using the FAC technique. The quantity of cases in each quadrant is detailed on their labels, please note that the sum of gates in each quadrant does not necessarily result in the 3982 logic functions studied, this hap-

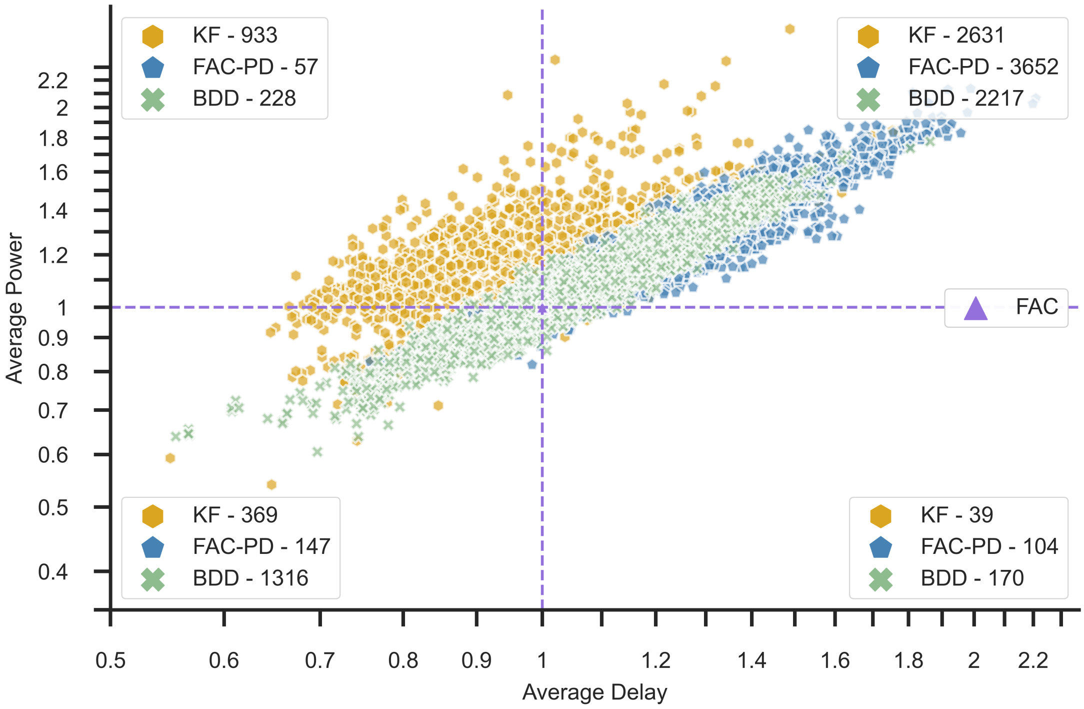


Figure 8 – Power and average delay scatter plot for the 4 input P-Class set. All results are normalized to FAC logic gates. Each marker represents a single gate designed with the labeled supergate design method.

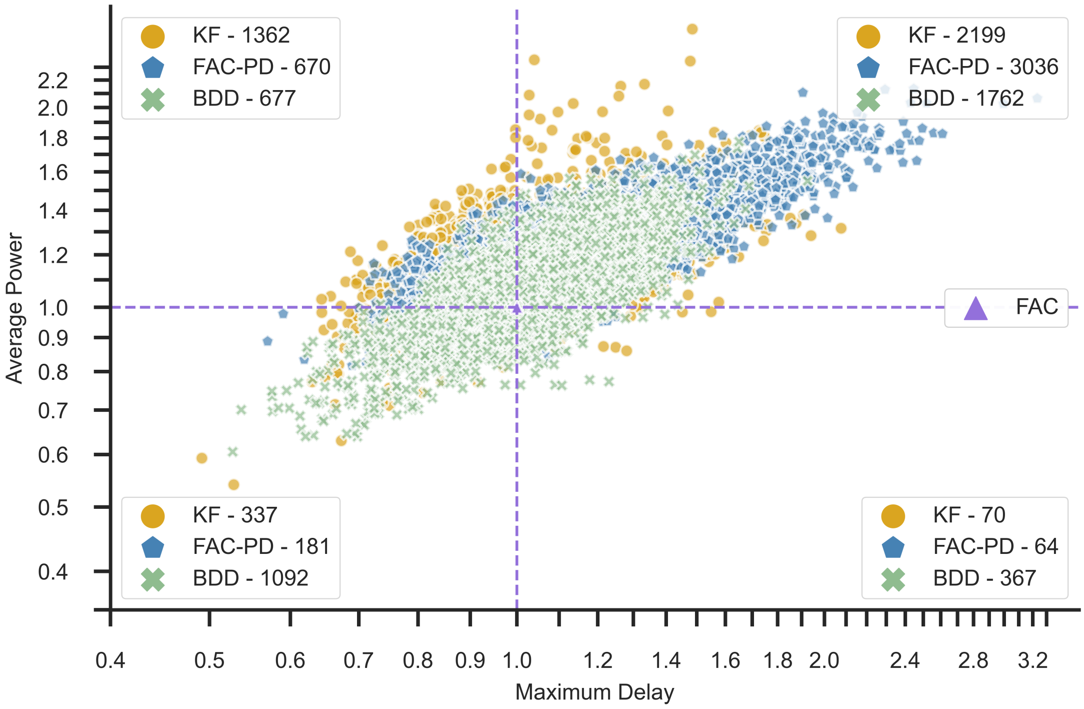


Figure 9 – Power and critical delay scatter plot for the 4 input P-Class set. All results are normalized to FAC logic gates. Each marker represents a single gate designed with the labeled supergate design method.

pens because two methodologies may lead to the same gate. Observing the results for FAC-PD logic gates only 7.73% had an improvement in at least one metric, where the vast majority (91.71%) achieves worse values in both average delay and average power. As seen in Table 2, FAC provides more consistent results across the whole set, however, Fig. 8 shows that BDD logic gates can achieve significant improvements in specific cases. In 33.04% of the logic functions, a gate designed with BDD achieved better results in both average delay and average power, moreover, in a particular gate, the BDD methodology reached a power-delay-product reduction of over 65%. KF gates achieve similar delay results to BDD, however, at an increased power dissipation.

Fig. 9 presents the results for maximum delay and power. Comparing the results obtained from the logic gate critical delay and average delay, no significant change from the average delay scatter plot is observed. BDD and KF can improve both metrics in 27.4% and 8.5% of the studied cases, respectively.

It is evident both in Fig. 8 and 9 that the transistor network generation method will have a major impact on a supergate design, which can be seen in the spread of the power and delay on these figures ranging from below 0.6 to above 2.0. Therefore, despite FAC logic gates achieving better results on average, a significant amount of the studied cases (27.4%) reached better designs with BDD when observing critical delay and power dissipation. The large spread of results, above mentioned, indicates that using a single supergate design approach (as they are presented) may yield logic gates with a performance that could be significantly improved by another design methodology. This observation suggests that developing a design selection criteria to choose which methodology would provide the best design to a targeted logic function. Other factors besides the transistor network generation method also contribute to the large spread of the electrical characteristics observed, which will be discussed next.

Delving deeper in a specific logic gate, the logic equation F_{3405} (3) has the highest average delay difference across all logic gates when comparing both FAC and BDD techniques, where BDD achieves significantly worst metrics. The transistor network for BDD is illustrated in Fig. 10 (a) and FAC on (b), both transistor networks are identical, however, FAC swaps V$_{dd}$ and the circuit output on the pull-up plane, likewise for V$_{ss}$ and output in the pull-down. This change does not affect transistor sizing and increases the logic gate average delay in 87%, critical delay in 65%, power dissipation in 89%, and average static power in 52%. This specific case study shows the major impact of transistor ordering on the electrical characteristics of the logic gate. Both FAC and BDD designs do not order their transistors with a specified technique and could take advantage of transistor reordering algorithms. Further discussion on this topic is presented in Chapter 4.

$$F_{3405} = \bar{a}\bar{b}\bar{d} + \bar{a}\bar{c}\bar{d} + bcd + \bar{b}\bar{c}\bar{d} + acd + abd \tag{3}$$

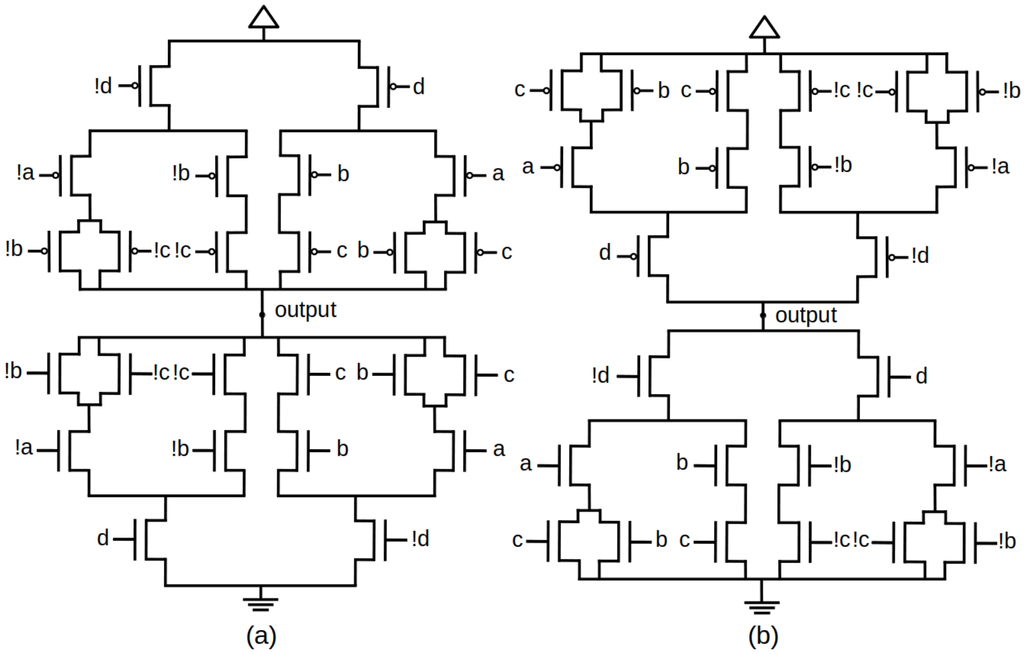


Figure 10 – Logic gate implementation for F_{3405} (3) designed with BDD (a) and FAC (b).

Table 3 – Transistor network for F_{220} (4) obtained from FAC (a) and BDD (b).

	FAC	BDD
Average delay	1.000	1.051
Maximum delay	1.000	1.006
Minimum static power	1.000	1.474
Average static power	1.000	1.073
Average power	1.000	1.006
Power-delay-product	1.000	1.057
Nº of transistors	1.000	0.933
Area estimation*	1.000	0.854

* Sum of transistors width.

$$F_{220} = \bar{a}\bar{b}c + \bar{a}b\bar{c}\bar{d} + \bar{a}cd + a\bar{b}\bar{c}d \tag{4}$$

Considering the logic equation F_{220} (4), Fig 11 illustrates the designs obtained from FAC (a) and BDD (b). Concerning the number of transistors, the FAC gate had an increase in comparison to the BDD design, from 28 to 30 (20 to 22 disregarding input inverters). The FAC gate also presented a larger transistor stack count on the pull-down plane from 3 to 4. Observing these metrics could indicate that the BDD logic gate would have better electrical behavior, however, this is not true after doing the SPICE simulation. Table 3 shows that BDD had an increase of 5% regarding the circuit average delay, a minor increase in the maximum delay, and the average power. Despite having a reduction of 14.6% when adding all transistors width the BDD gate achieves 5.7% worse power-delay-product.

The electrical characteristics of supergates are greatly impacted by the gate struc-

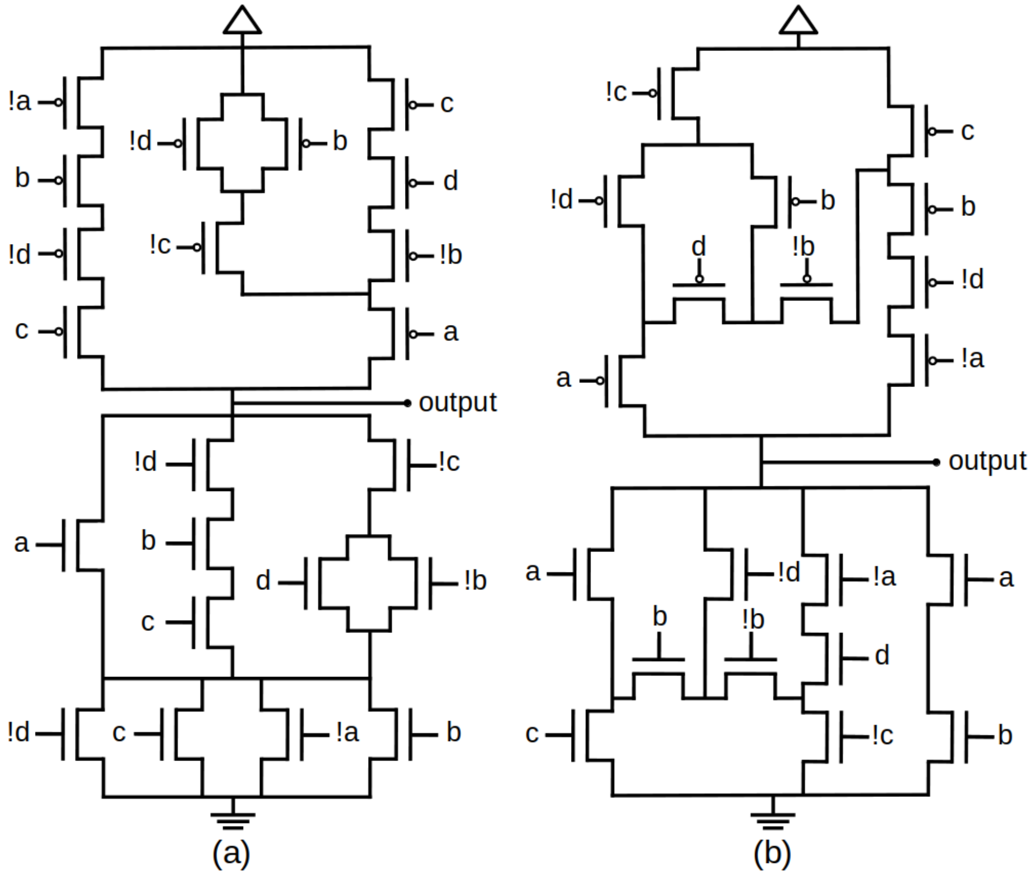


Figure 11 – Results for F_{220} (4) using logic networks generated by FAC (a) and BDD (b).

ture. In this chapter, an electrical comparison between logically equivalent logic gates designed through three supergate techniques contained in the SwitchCraft (CALLEGARO et al., 2010) and Kernel Finder (POSSANI et al., 2015). Regarding the electrical characteristics, the FAC logic network generation method achieved better results across all metrics in the 4 input P-Class set of functions when compared to FAC-PD, BDD, and KF logic gates, while small improvements in circuit area were found in KF gates. These results suggest that despite the FAC technique providing better results on average, an alternate design strategy may have significantly better characteristics. Also, none of the approaches has a defined transistor ordering priority, punishing the circuit optimization. The main conclusions of this chapter are summarized as follows.

1. We experimentally demonstrate that supergate design based solely on topological metrics is not strict enough to guide a design.
2. Improvements in metrics, such as transistor count, transistor stack, or gate area, do not necessarily lead to electric gains in a supergate, therefore, they should be avoided when comparing design methodologies.
3. No single supergate design strategy delivers consistently better results, resulting in supergates that may vary from being the best up to twice as worst, indicating that design selection criteria are of great importance.
4. Transistor ordering on supergates is of crucial importance for electrical charac-

teristics. Further investigation on reordering algorithms is encouraged for both improvements in electrical behavior.

The study presented in this chapter was published in JICS 2021 (Appendix A.1.1).

4 SUPERGATE TRANSISTOR REORDERING

The attention to pin selection and transistor reordering is long studied. Marek (MAREK-SADOWSKA; LIN, 1990) focused on the selection of input pins to reduce the logic gate delay and power. Further work is done by Carlson (CARLSON; LEE, 1995), instead of only working with input pins, he also explored the reordering of transistors inside logic gates. Soon, the potential for a reduction in dynamic power through transistor reordering was noticed and many works attempted to take advantage of this resource. These works target the reduction of charging/discharging events in the internal capacitance of the gate. Shen (SHEN; LIN; WANG, 1995) proposed the transistor placement in a network so that the transistors with a higher chance of being active would be closer to the power supply and as a tie-breaker, the transistors with a lower transition density would be closer to the output. Musoll's (MUSOLL; CORTADELLA, 1996) approach optimizes the gate through the permutation transistors and blocks connected in series; a set of all possible gate designs is built and exhaustively tested; finally selecting the gate structure which minimizes the charging/discharging events. Prasad's (PRASAD; ROY, 1996) algorithm targets the minimization of power under a delay constraint exhaustively permuting transistors. In Glebov's (GLEBOV; BLAAUW; JONES, 1995) approach, the entire network is re-designed with a BDD strategy and uses switch-level simulations, evaluating all possible structures.

The above-mentioned solutions have a high computational cost and are not adequate for gates with high complexity or for those which would be designed on-the-fly. As an alternative, Hossain's approach (HOSSAIN; ZHENG; ALBICKI, 1996) follows the same principle of reducing the activity on internal capacitances, however, it does so without the need to perform simulations exhaustively. A *component* is defined as a collection of transistors that are in series with any other collection of transistors. The algorithm follows two steps, (i) arrange the components in all the serial paths between the output and power supply node in ascending order of their weighted switching activity; if two components are tied, place the component with the larger number of transistors closer to the power supply; begin with components which are not contained in any other components and (ii) repeat the process for all components which are contained

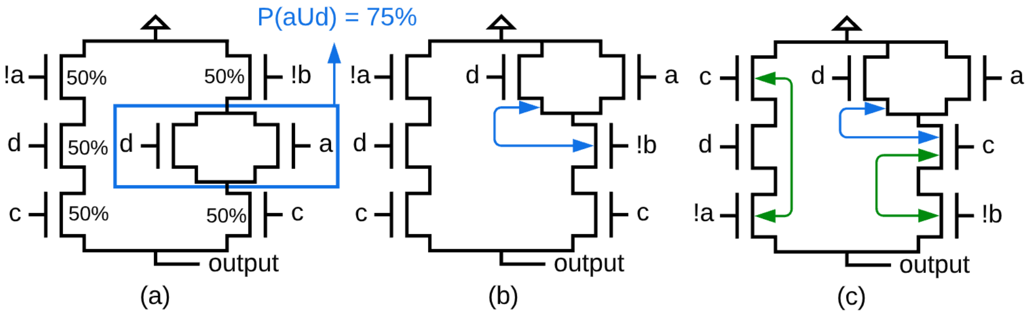


Figure 12 – Transistor network components probabilities, (a) illustrates an unordered transistor network, (b) reordered network with Hossain's algorithm and (c) proposed transistor reordering.

in the components already considered until all components have been so arranged. This solution is suitable for large complex gates due to its small computational cost and competitive results.

Besides delay and dynamic power reduction, the transistor reordering problem has also been studied to minimize static power (CHUN; CHEN, 2016), layout-related challenges where the Euler path was sought either exhaustively (SMANIOTTO et al., 2017) or through Boolean Satisfiability (CARDOSO et al., 2020). Other reordering applications include the mitigation of BTI (BUTZEN et al., 2012), HCI (KIAMEHR; FIROUZI; TAHOORI, 2012), and soft errors (REIS et al., 2020).

4.1 Hossain's Transistor Reordering

To further investigate the electrical impact of a transistor reordering method applied to supergates, a study using the algorithm presented in (HOSSAIN; ZHENG; ALBICKI, 1996) is presented. Hossain's technique aims to reduce the activity on internal capacitances with a computationally inexpensive algorithm, allowing the use of an on-the-fly design. This methodology reorders transistors based on a weighted switching activity, transistors with higher activity are positioned near the VDD on the pull-up plane or GND on the pull-down plane. However, the proposed algorithm does not handle non-series-parallel associations, thus only FAC supergates have been reordered.

Fig. 12 presents an example of the reordering made by the strategy in (HOSSAIN; ZHENG; ALBICKI, 1996) (b) over an unordered transistor network (a). Assuming all inputs with equal chances of being 1 or 0 and statistically independent, the probability of the transistor being active is 50% and by association, the parallel is calculated to be 75%. On Fig. 12 (b) using the components probabilities, the algorithm reorders blocks with the highest probability, indicated by the blue arrow, to be closer to the power supply (HOSSAIN; ZHENG; ALBICKI, 1996).

Preliminary results on a small selection of logic gates without inverted inputs presented significant electrical improvement, thus an experiment on the 4 input P-Class was conducted. Fig. 13 shows the electrical results for FAC supergates on the 4 in-

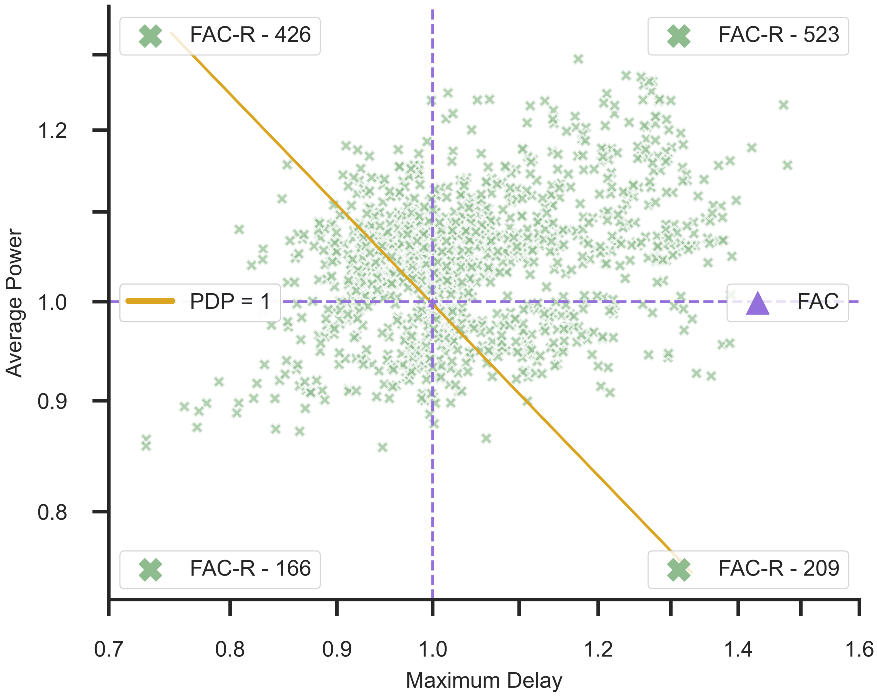


Figure 13 – Power and critical delay scatter plot for the 4 input P-Class set. Comparison between original FAC supergates and FAC-R reordered with (HOSSAIN; ZHENG; ALBICKI, 1996). Each marker represents a single reordered supergate.

put P-Class with original FAC and reordered (FAC-R) (HOSSAIN; ZHENG; ALBICKI, 1996) transistors. Across the 3982 logic gates, 1324 (33.25%) designs had a change in their transistor ordering. Observing the supergate critical delay 55.28% (732 from 1324) achieved the worst values and 71.67% (949) had a higher power dissipation. Considering the power-delay-product 67.97% (900) resulted in inferior results. The results show that the majority of the supergates designed by FAC do not benefit from the reordering proposed in (HOSSAIN; ZHENG; ALBICKI, 1996).

Table 4 details the results for the 4 input P-Class with non-ordered FAC supergates as the baseline. This experiment shows that on average the FAC-R achieved the worst electrical behavior in all delay and power measurements. It is not clear to the author why the algorithm fails to improve the electrical characteristics of the supergates. The experiments have shown a strong dependency on the logic function and the supergate design, generating data that does not provide an easily identifiable pattern.

Delving deeper in Fig. 12 (b) as an example, it contains two inputs with inverters. On the logic gate level, all inputs have a similar signal delay and slope, however, once that signal goes through an inverter, an additional delay is incurred. This situation made the reordering less efficient, thus a study on transistors with inverted inputs is made in the next section.

Table 4 – Electrical results for the 4 input P-Class set of functions with results normalized to FAC method.

	FAC	FAC-R
Average delay	1.000	1.001
Maximum delay	1.000	1.013
Minimum static power	1.000	1.015
Average static power	1.000	1.001
Average power	1.000	1.014

4.2 Proposed Transistor Reordering

During the design of a supergate, it may be required to add inverters to assure the transistor network satisfies the Boolean function. Applying the transistor reordering proposed by (HOSSAIN; ZHENG; ALBICKI, 1996) successfully improved the electrical characteristics whenever no inverter was added. Taking into consideration this effect, this thesis proposes to place transistors with inverted inputs near the gate output. Fig. 12 (c) illustrates an example of this reordering. Note that this does not change the order of transistor blocks, only transistors associated in-series, thus the extra step indicated by the green arrow is applied.

The reordering algorithm and the proposed reordering algorithm are evaluated through SPICE simulations of the 4-input P-Class set (SASAO, 2012). Transistors were sized according to the Logical Effort (SUTHERLAND et al., 1999) method using a 1.32 PMOS/NMOS ratio. All logic gates were designed with the 45nm CMOS TSMC technology. All supergates on this chapter were designed with SwitchCraft using the *factorization method* (FAC). It is important to note that, despite generating the transistor network, the SwitchCraft design does not have a defined ordering technique. Supergates reordered by Hossain's technique are referred to as "FAC-R" and, supergates reordered by the proposed method are referred to as "FAC-PN", from probability-based and negated inputs.

Fig. 14 shows the average power and the maximum delay for each of the gates generated with the proposed algorithm (FAC-PN) normalized by its counterpart generated with FAC-R (HOSSAIN; ZHENG; ALBICKI, 1996). The reordering algorithms resulted in different designs in 3387 of the 3982 logic gates designed from the 4 input P-Class. The proposed algorithm achieved gains in both power dissipation and critical delay in 2146 gates (63.36% of the gates with different designs) while obtaining worse results in both metrics in only 262 (7.74%). The spread in Fig. 14 displays that most of the reordered supergates achieved a reduction in both critical delay and power dissipation. Table 5 details the average results for the 4 input P-Class normalized to FAC-R supergates.

The proposed reordering algorithm leads to improvements in the electrical charac-

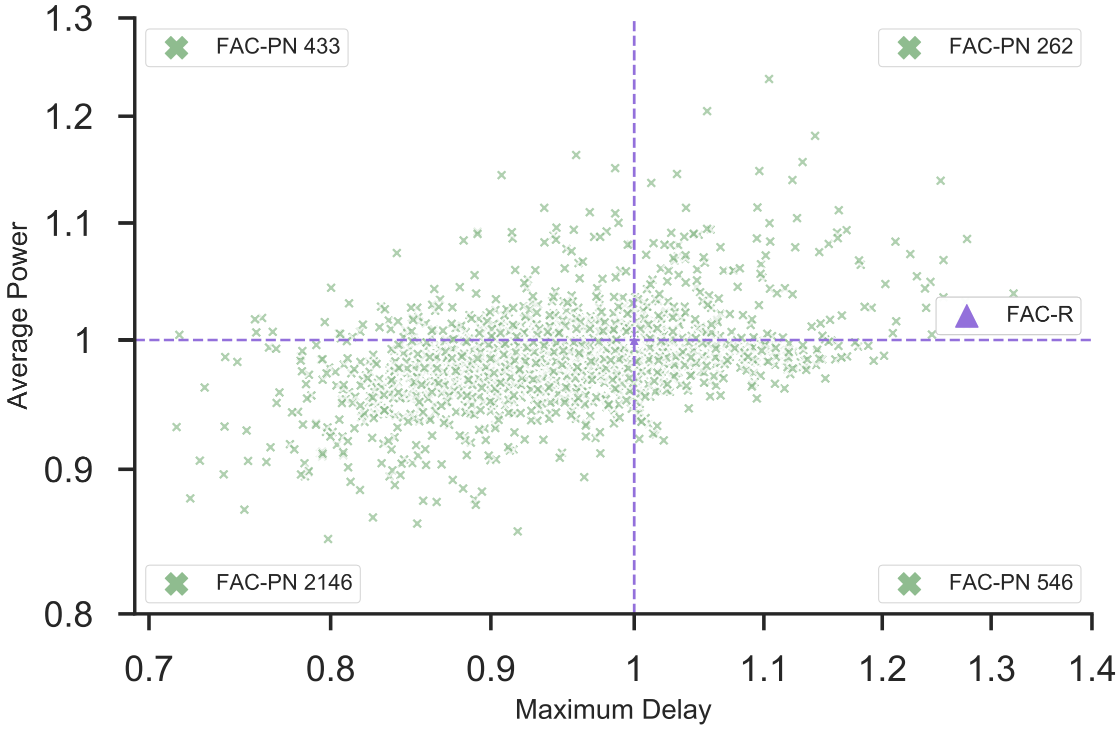


Figure 14 – Comparison between the FAC-R (HOSSAIN; ZHENG; ALBICKI, 1996) and for the proposed reordering (FAC-PN) supergates. Each marker represents a single gate designed with the proposed algorithm normalized by its counterpart designed with (HOSSAIN; ZHENG; ALBICKI, 1996) for the 4 input P-Class.

Table 5 – Normalized results to the Hossain reordering using FAC supergates on the 4-input P-Class.

	FAC-R	FAC-PN
Average Delay	1.000	0.997
Critical Delay	1.000	0.963
Power-Delay-Product	1.000	0.954
Average Static Power	1.000	0.991
Average Power	1.000	0.990

teristics of the supergates, on average it improved all studied metrics. The major advantage of the proposed reordering is the critical delay improvement. It targets inputs with inverted literals, which have a larger chance of being the ones with a worst-case delay. Even though it leads to improvements in the majority of cases, the drawbacks from the reordering algorithm (HOSSAIN; ZHENG; ALBICKI, 1996) are still in place and are not addressed in this work. The next section will further investigate issues from using the probability of series-parallel associations for transistor reordering.

4.3 Transistor Reordering Drawbacks

In this section, a case study on the logic function F_{1088} (5) will be presented discussing where the reordering of components based on the probability of being active fails to improve the logic gate electrical behavior.

$$F_{1088} = ((a + (bd)) * (!a + (!d + (!bc)))) * (!c + d) \tag{5}$$

Fig. 15 (a) illustrates the transistor network of the logic function F_{1088} with an arbitrary ordering of the components. Three critical blocks and their probability of being active are highlighted. While evaluating the PMOS plane, the blue highlighted component has a smaller probability to propagate the signal when compared to the single "!a" transistor. Conducting probability-based algorithms, such as the one studied in this work, will swap them, as shown in Fig. 15 (b), however, this change by itself does not improve the gate's electrical characteristics. The reordering of components approach aims to put internal nodes with a higher probability to be charged closer to the supply nodes. Nonetheless, comparing a single transistor with a group of transistors fails this premise due to series associations inside the block reducing the probability. Due to the recursive nature of the reordering algorithm, the internal nodes of a component are lost, thus being susceptible to these non-intended swaps. In (HOSSAIN; ZHENG; ALBICKI, 1996), there is an attempt to address this issue with the number of transistors in a component serving as a tie-breaker when the conducting probabilities are equal. Such a measure is not sufficient to avoid bad swaps like the one presented here.

In regards to the logic gate critical propagation delay, the non-ordered gate obtains a better result. The input array for this situation is "1011" for "a", "b", "c", "d" respectively. The swap of the highlighted blocks in the NMOS plane of Fig. 15 increases the output load for this specific input array. This can be seen on the red arrow across the logic gate netlist, on (a) the orange block is not active, after the swap on (b) the signal propagates further into the pull-down plane, increasing the load and leading to a larger propagation delay.

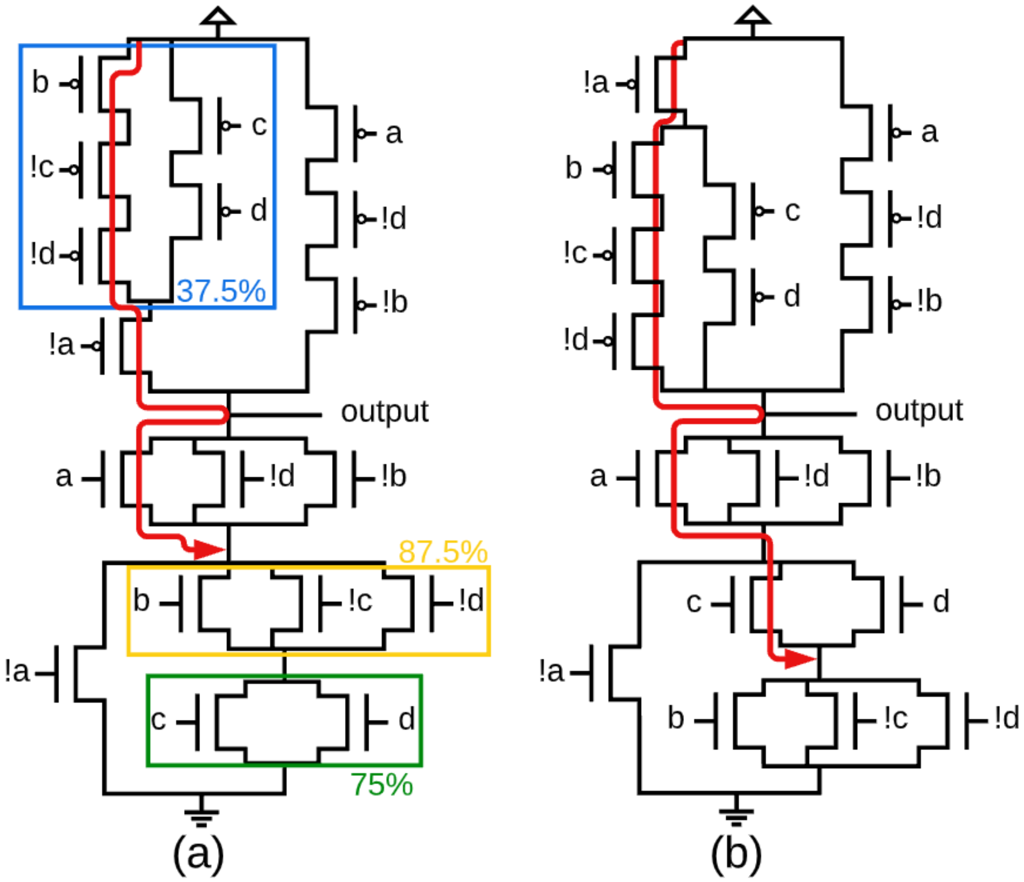


Figure 15 – Transistor networks for F_{1088}, (a) unordered, (b) probability based reordering.

Algorithms that use conduction probability of components to reorder will, in some cases, face one of the situations here presented, and may fail to improve the critical delay of a gate.

This chapter evaluated the impact of a transistor reordering algorithm on the electrical characteristics of supergate. The drawbacks of transistor reordering based on conducting probabilities of components were presented and detailed in a case study. Despite such drawbacks, the algorithm can improve the electrical characteristics of a supergate by reordering the transistors. This technique can be used in post-processing or for the automatic design of logic gates, with low computational cost.

5 SUPERGATE TRANSISTOR SIZING

During the logic synthesis of a digital IC in a standard-cell-based project, the design is mapped to the technology by the selection of the logic gates. However, each chosen logic gate has multiple sizes (drive strengths), thus the problem of gate sizing (gate selection). Power-performance trade-off is significantly affected by gate sizing and it is an NP-Hard problem to be optimally solved (NING, 1994).

Several articles in the literature tackle this problem. The proposed solutions could be broadly divided into two main areas: continuous and discrete sizing. Continuous sizing focus on optimizing the transistors or the logic gate parameters (RAHMAN; TENNAKOON; SECHEN, 2013; LEVI; BELENKY; FISH, 2013; GUPTA et al., 2019). Aiming to optimize the circuit area, (RAHMAN; TENNAKOON; SECHEN, 2013) uses a look-up table delay model based on Logical Effort. A modified Logical Effort is proposed by (LEVI; BELENKY; FISH, 2013) for dual-mode logic gates for high-performance or low-power. Considering process variations, supply voltage, and temperature, (GUPTA et al., 2019) optimally sizes transistors for power or delay.

Currently, the modern design flow mostly uses only discrete cell sizes, which may introduce overheads when using continuous optimizations. Discrete methods, on the other hand, focus on optimizing the use of the fixed cell sizes in the standard cell library (FLACH et al., 2014; HU et al., 2012; FATEMI et al., 2019; HESHAM; NASSAR; MOSTAFA, 2020). For high-performance circuits, (FLACH et al., 2014) uses a Lagrangian-Relaxation-based approach and (HU et al., 2012; FATEMI et al., 2019) approaches the problem by applying metaheuristics. Also for energy-efficient design, (HESHAM; NASSAR; MOSTAFA, 2020) operates the library in near-threshold voltage and proposes body bias to increase performance.

The on-the-fly gate generation allows the implementation of supergates. Works that take advantage of supergates, to the best of the author’s knowledge, do not experiment with transistor sizing techniques, applying an unmodified Logical Effort (SUTHERLAND et al., 1999) sizing (CALLEGARO et al., 2010; POSSANI et al., 2015; CARDOSO et al., 2016) or uses minimum-sized transistors (OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019; ALBINAGORTA et al., 2019).

Table 6 – Occurrences for the best Logical Effort multiplier coefficient for power-delay-product in the 3 input NPN-Class.

Coefficient	1.00	0.95	0.90	0.85	0.80	0.75	0.70	0.65	0.60	0.55	0.50	0.45	0.40	0.35
Occurrences	1	0	1	0	0	0	0	0	1	0	9	1	0	1

Considering the pivotal role of transistor sizing in the logic gates power-performance trade-off, this chapter presents a simple calibration technique to the Logical Effort method for a better power-delay-product while not increasing the computational cost for an automated supergate design.

5.1 Logical Effort Calibration Methodology

In the original Logical Effort method, the delay of a logic gate is represented by the arrangement of transistors and the rate of output/input load (SUTHERLAND et al., 1999). This method is a technology-independent model due to a calibration of the delay to a minimum-sized inverter. The Logical Effort's approach aims to optimize the logic gate delay and usually presents good results. However, focusing only on the gate delay may not be adequate for low-power applications. With that in mind, this chapter presents a simple coefficient multiplier to adjust the transistor sizing for a better power-performance trade-off.

5.1.1 The Premise

The approach aims to optimize the transistor sizing made by the Logical Effort technique, focusing on the reduction of the power-delay-product of automatically designed supergates while not increasing the algorithm complexity (computational cost). To achieve this goal, a multiplier is applied to the width obtained from the Logical Effort sizing. A sweep with a step of 0.05 from numbers 0.35 to 1.00 was simulated in the 3 input NPN-Class (SASAO, 2012) of logic functions. In regards to the simulation environment, FIN4 and FO4 were used.

Table 6 presents the number of occurrences for each multiplier value that achieved the best PDP value. Most of the cases use 0.5 the multiplier to the transistor width resulting from the Logical Effort approach. Only a single case was found to be 1.00 which is the unmodified Logical Effort method (the inverter gate). This result indicates that the sizing made by the Logical Effort is oversized for an optimal PDP (for this technology and process), thus further evaluation should be considered to find a better-suited sizing.

5.1.2 Proposed Logical Effort Calibration

Considering the study on the 3 input NPN-Class, the multiplier of 0.5 was the value that achieved better PDP results in most of the cases. However, the 3 input NPN-Class consists of only 14 logic functions, which is a small sample considering that supergates can be designed with more than 3 inputs. Therefore, this chapter evaluates the use of this multiplier on the 4 input P-Class. In this chapter, logic gates sized with the unmodified Logical Effort technique will be referred to as *LE*. Logic gates with the 0.5 the multiplier will be referred to as *Half*.

However, it is important that the on-the-fly designed supergate is compatible with a standard cell library, where the cell height is fixed and transistors sizes are multiples of that height (SHAHOOKAR; MAZUMDER, 1991). Targeting such compatibility, this work will round down the value obtained from *Half* when the value is not a multiple-sized transistor. Logic gates sized with this rounded-down value will be referred to as *Lower*.

Fig. 16 shows an example of a supergate using each transistor sizing approach. Values by each transistor consists of the minimum transistor width multiplied by each method, top value is resulted from *LE*, middle from *Half* and bottom from *Lower*. Both *Half* and *Lower* result in the same sizing when the multiplier from *LE* is even, when the size is odd, e.g. 3, *Half* results in 1.5 and *Lower* in 1.

In this study, two fan-in sizes were used, Fan-in 1 (FIN1) represents the input array connected with two minimum sized inverters in series, Fan-in 4 (FIN4) uses transistors four times the minimum width. As circuit output load, three values were used, a single minimum sized inverter, Fan-out 1 (FO1), four minimum sized inverters associated in parallel (FO4), and eight minimum sized inverters (FO8). All possible combinations of FIN and FO were simulated in this chapter. As for the supergate design methodology the SwitchCraft framework was used (CALLEGARO et al., 2010). In this chapter, logic gates were created using the *factorization method* command (FAC) and *from BDD - minimum stacks* command (BDD), Fig. 16 illustrates a resulting FAC supergate and the three evaluated sizing values.

5.2 Results and Discussions

This section presents the experimental results for logic gates sized through Logical Effort and both proposed adaptations *Half* and *Lower*. All gates were implemented with TSMC 45nm technology process design kit using the value of 1.32 as a PMOS/NMOS ratio. No transistor reordering technique is applied.

Table 7 summarizes the results obtained for the 4 input P-Class set for FAC supergates. Each metric for each logic gate was normalized by its *LE* counterpart. The columns represent the fan-in strength and the sizing method while the rows represent

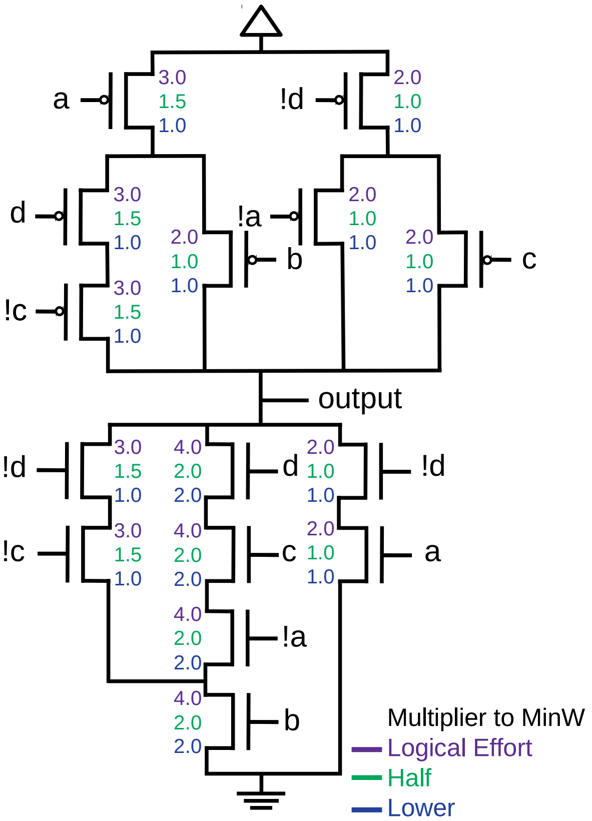


Figure 16 – Supergate sizing example using the three approaches, from top-down, Logical Effort, Half, and Lower. The values at the side of each transistor represent the transistor width multiplied by the minimum width allowed by the technology.

each metric with the varying output load. As power-delay-area-product (PDA) the sum of the logic gate transistors' width multiplies the values obtained in the power-delay-product (PDP). PDP was significantly improved by both *Half* and *Lower*. These results are achieved due to the substantial reduction in power dissipation, even at a cost in propagation delay. As *LE* focuses solely on delay, it leads to a worse power-delay trade-off for supergates design.

Considering the sum of the transistors width, *Lower* and *Half* reduces the value by *LE* gates in 52.6% and 44.5%, respectively. However, taking into account that a supergate in a fixed height cell layout does not allow a non-multiple size in its transistor width, a more realistic area cost for *Half* is a reduction of 36.5%. These results indicate a better PDA trade-off can be achieved by applying both sizing methods. By varying the input strength, the results show that increasing the input drive strength benefits *LE* sizing the most in regards to propagation delay, this can be seen by the worsening of both delay metrics of *Half* and *Lower*. *LE* gates use bigger transistors, increasing their input load and slew rate, which explains why a stronger drive-in improves the delay. In regards to the power dissipation no significant impact can be seen (less than 0.1%).

Changes to the logic gate load, also benefit *LE* the most, as the fan-out load is at or above four minimum sized inverters, *LE* starts to have a smaller delay than the other

Table 7 – Normalized results to Logical Effort technique for the FAC supergates on the 4 input P-Class set of functions.

		Fan-In 1			Fan-In 4		
		LE	Half	Lower	LE	Half	Lower
Fan Out 1	Average Propagation Delay	1.000	0.915	0.920	1.000	0.941	0.952
	Critical Propagation Delay	1.000	0.930	0.976	1.000	0.951	1.005
	Average Power	1.000	0.599	0.524	1.000	0.598	0.523
	Power-Delay-Product	1.000	0.558	0.512	1.000	0.570	0.526
	Power-Delay-Area-Product	1.000	0.313	0.246	1.000	0.319	0.253
Fan Out 4	Average Propagation Delay	1.000	1.012	1.058	1.000	1.042	1.097
	Critical Propagation Delay	1.000	1.012	1.102	1.000	1.034	1.134
	Average Power	1.000	0.646	0.581	1.000	0.646	0.580
	Power-Delay-Product	1.000	0.655	0.640	1.000	0.668	0.657
	Power-Delay-Area-Product	1.000	0.367	0.306	1.000	0.374	0.315
Fan Out 8	Average Propagation Delay	1.000	1.112	1.206	1.000	1.144	1.250
	Critical Propagation Delay	1.000	1.106	1.261	1.000	1.130	1.298
	Average Power	1.000	0.697	0.641	1.000	0.697	0.640
	Power-Delay-Product	1.000	0.772	0.808	1.000	0.788	0.830
	Power-Delay-Area-Product	1.000	0.432	0.386	1.000	0.441	0.396

	Logical Effort	Half	Lower
Sum of transistors width	1.000	0.555	0.474

approaches. These results indicate that if performance is the priority to the design, *LE* is better suited, which shows that the technique's main objective is accomplished.

While not being the scope of this study to electrically evaluate multi-level circuits, using smaller transistors dimensions decreases the input node capacitance, thus requiring less drive strength from previous logic gate levels. The overall capacitance increase in the circuit is smaller when using *Half* and *Lower*.

Fig. 17 presents the results for all logic gates in the 4 input P-Class for critical propagation delay and power, normalized to the *LE* sizing with FIN4 and FO4 as the environment with FAC supergates. It shows that all logic gates by *Half* and *Lower* achieved lower power dissipation (the inverter gate is equally sized, thus reproduces the same results as *LE*). Regarding the maximum delay of each logic gate, only 1040 (26.1%) from *Half* and 857 (21.5%) from *Lower* achieved a better results in comparison to *LE*. However, each point under the yellow line represents an improvement in the power-delay-product, only 9 (0.22%) gates from *Half* and 26 (0.65%) from *Lower* achieved worse results.

Considering the best-case scenario for *LE*, using FIN4 and FO8, *Half* gates improves PDP on 97.5% and *Lower* on 88.2% of the studied cases. These results further indicate that power-performance trade-off from *LE* can be significantly improved.

Regarding BDD supergates, no notable difference from FAC supergates is found on the electrical characteristics, the impact on both design strategies was similar. Table 8

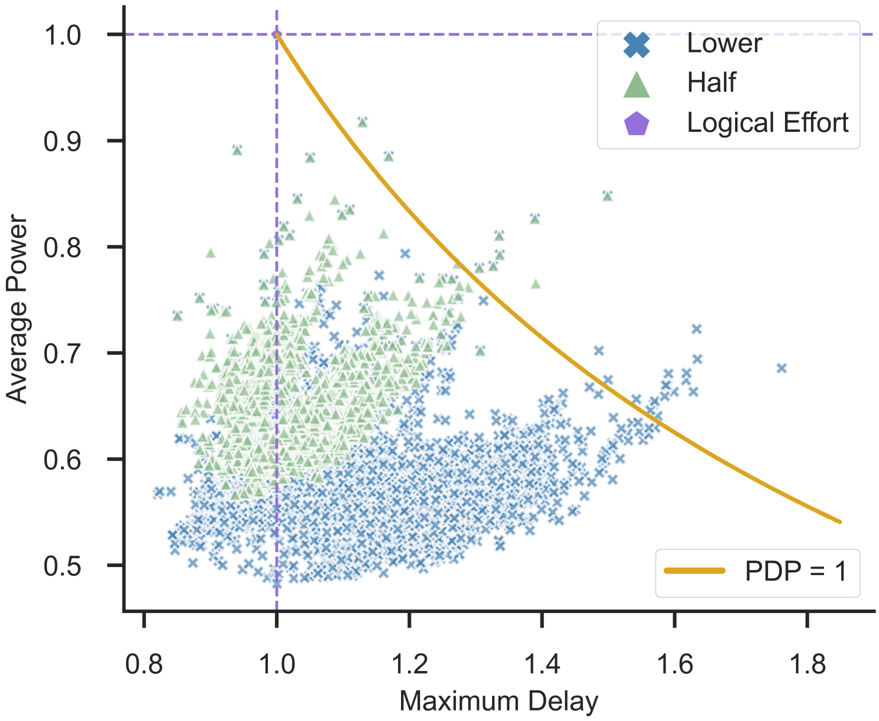


Figure 17 – Average Power and Maximum delay scatter plot for each function of the 4 input P-Class set of functions designed with the *Lower* and *Half* methods and normalized to the *LE* sized gates. The simulation environment used for these results is FIN4 and FO4 using FAC supergates.

describes the average result across all functions for BDD supergates normalized to the Logical Effort method. Fig. 18 illustrates the scatter plot for 4 input P-Class using FIN4 and FO4 for BDD supergates, similar distribution to the FAC supergates is achieved when sizing BDD supergates.

A substantial amount of research is presented for gate sizing (gate selection) given a library of cells, however, the sizing of automatically generated static CMOS complex gates is not sufficiently discussed. This chapter shows that using the unmodified *LE* technique results in a worse power-performance trade-off for supergates design. By applying a simple calibration methodology for a coefficient multiplier, significant gains are achieved in up to 99.9% of the studied cases. The results presented in this chapter also show that proper sizing is critical for an efficient supergate design. The study presented in this chapter was published in LASCAS 2022 (Appendix A.1.2).

Table 8 – Normalized results to Logical Effort technique for the BDD supergates on the 4 input P-Class set of functions.

		Fan-In 1			Fan-In 4		
		LE	Half	Lower	LE	Half	Lower
Fan Out 1	Average Propagation Delay	1.000	0.911	0.908	1.000	0.937	0.941
	Critical Propagation Delay	1.000	0.938	0.971	1.000	0.960	1.000
	Average Power	1.000	0.594	0.515	1.000	0.599	0.515
	Power-Delay-Product	1.000	0.557	0.500	1.000	0.570	0.514
	Power-Delay-Area-Product	1.000	0.311	0.318	1.000	0.280	0.288
Fan Out 4	Average Propagation Delay	1.000	1.006	1.043	1.000	1.035	1.081
	Critical Propagation Delay	1.000	1.018	1.094	1.000	1.042	1.126
	Average Power	1.000	0.640	0.570	1.000	0.639	0.570
	Power-Delay-Product	1.000	0.652	0.622	1.000	0.666	0.640
	Power-Delay-Area-Product	1.000	0.364	0.372	1.000	0.348	0.358
Fan Out 8	Average Propagation Delay	1.000	1.105	1.191	1.000	1.137	1.234
	Critical Propagation Delay	1.000	1.112	1.249	1.000	1.137	1.284
	Average Power	1.000	0.690	0.630	1.000	0.689	0.607
	Power-Delay-Product	1.000	0.766	0.784	1.000	0.783	0.806
	Power-Delay-Area-Product	1.000	0.428	0.437	1.000	0.438	0.450

		Logical Effort	Half	Lower
	Sum of transistors width	1.000	0.554	0.470

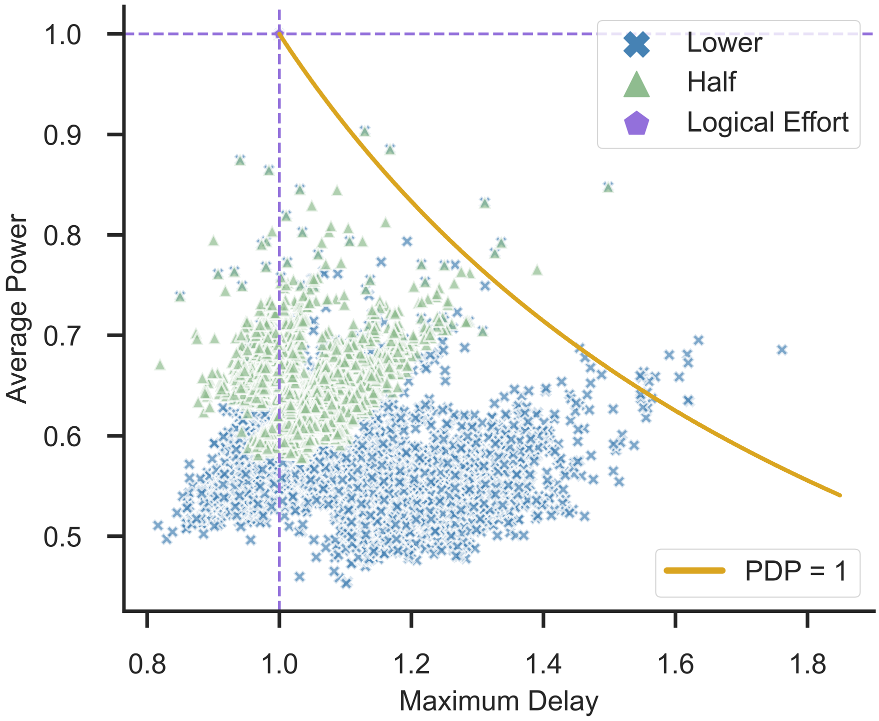


Figure 18 – Average Power and Maximum delay scatter plot for each function of the 4 input P-Class set of functions designed with the *Lower* and *Half* methods and normalized to the *LE* sized gates. The simulation environment used for these results is FIN4 and FO4 using BDD supergates.

6 EXAMPLE OF TECHNOLOGY MAPPING

This chapter presents the steps taken to design the technology-mapped circuits and the cell library. Fig. 19 illustrates the methodology employed in this chapter, which uses a logic function as input to generate a SPICE file environment ready for simulation. All designs in this chapter, the cell library, and the supergates, uses the Logical Effort sizing approach at the gate-level only, delay paths on multi-level circuits were not considered. All circuits were implemented with TSMC 45nm technology node using the value of 1.32 PMOS/NMOS ratio. No reordering technique is applied.

6.1 ABC

The ABC tool from Berkeley (BRAYTON; MISHCHENKO, 2010) is a framework that uses And-Inverter-Graphs (AIG) and contains a collection of algorithms for logic synthesis, mapping, and formal verification. In this chapter, the ABC technology mapping is used to design the library mapped circuits with the MCNC genlib from the SIS distribution (SENTOVICH et al., 1992). Table 9 presents the logic functions contained in the MCNC set.

The steps taken to design the circuits with ABC are presented by the flowchart in Fig. 19 and are detailed as follows.

1. Read the logic function in hexadecimal format using *read_truth* and the MCNC genlib with *read_library*;
2. Execute the *map* command ten times, on each map a *balance* is performed on the AIG;
3. Write the mapped circuit in Verilog format using *write_verilog* command;

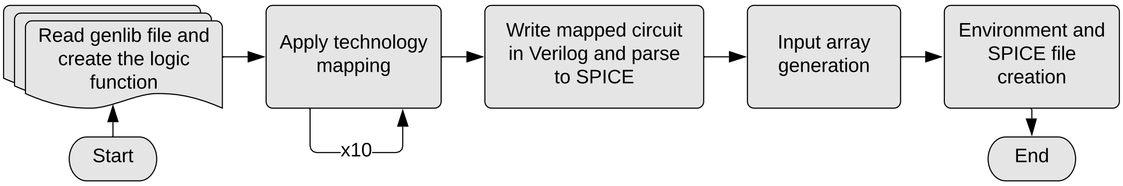

Read genlib file and create the logic function
Apply technology mapping
Write mapped circuit in Verilog and parse to SPICE
Input array generation
Environment and SPICE file creation
Start
x10
End

Figure 19 – Methodology for electrical evaluation of technology mapped circuits using ABC.

Table 9 – List of Logic Gates in MCNC genlib.

INV	AND2	NAND2/3/4	
OR2	NOR2/3/4	XOR2	XNOR2
AOI21	AOI22	OAI21	OAI22

The command *map* is used multiple times due to the incremental optimization done by ABC. This value is selected empirically, most of the circuits already converged to the minimal solution after three to five executions, ten is used as a form to obtain the best possible result from the ABC synthesis. The execution of *balance* balances the AIG, reducing the number of levels and, if possible, obtaining a smaller delay. Fig 7 (a) illustrates a design done by this methodology.

6.2 Library Mapped Circuits vs Supergates

This section presents the experimental results for both ABC mapped circuits and the supergates applied in the 4 input P-class (SASAO, 2012) set of functions. The choice to evaluate logic functions with up to 4 inputs is due to the belief that supergates with more than 4 transistors associated in series may require a bigger transistor sizing without a sufficient electrical behavior improvement resulting in a less efficient design. All supergates on this chapter were designed with SwitchCraft using the *factorization method* (FAC).

Table 10 summarizes the results obtained for the 4 input P-class set. Each metric for each logic gate was normalized by its ABC circuit counterpart. On average, the obtained results indicate that supergate-based designs achieve better power dissipation and average propagation delay at the cost of critical delay. Considering the power-delay-product (PDP) supergates also show better efficiency.

In regards to the total transistor count, only ten circuits from ABC obtained fewer transistors. A reduction of 43.5% in the number of transistors is achieved by FAC gates, however, this metric by itself is somewhat misleading since transistors in supergates usually are set on a bigger chain, which results in a bigger transistor sizing. A more realistic comparison could be done by adding all the transistors widths resulting in a reduction of 12.9%. Yet, as the most commonly used design methodology consists of fixed cell heights (SHAHOOKAR; MAZUMDER, 1991) and gate-matching during the layout is the most area-efficient approach (UEHARA et al., 1979), supergates that are not topologically complementary may present area overheads. Whenever an Euler path is not found for both pull-up and pull-down networks an insertion of a dummy transistor or a diffusion break is necessary. Evaluation of the layout is out of the scope of this thesis, however, works, such as (SMANIOTTO et al., 2017; CARDOSO et al., 2018, 2020), are capable of automatically generating supergates layouts, while also

Table 10 – Normalized results to ABC mapped circuits for the 4 input P-Class set of functions.

	ABC	FAC	BDD
Average Propagation Delay	1.000	0.895	0.931
Critical Propagation Delay	1.000	1.058	1.072
Average Static Power	1.000	0.818	0.915
Average Power	1.000	0.788	0.827
Power-Delay-Product	1.000	0.863	0.915
Transistor Count	1.000	0.565	0.568
Area estimation*	1.000	0.871	0.884

* Sum of transistors width.

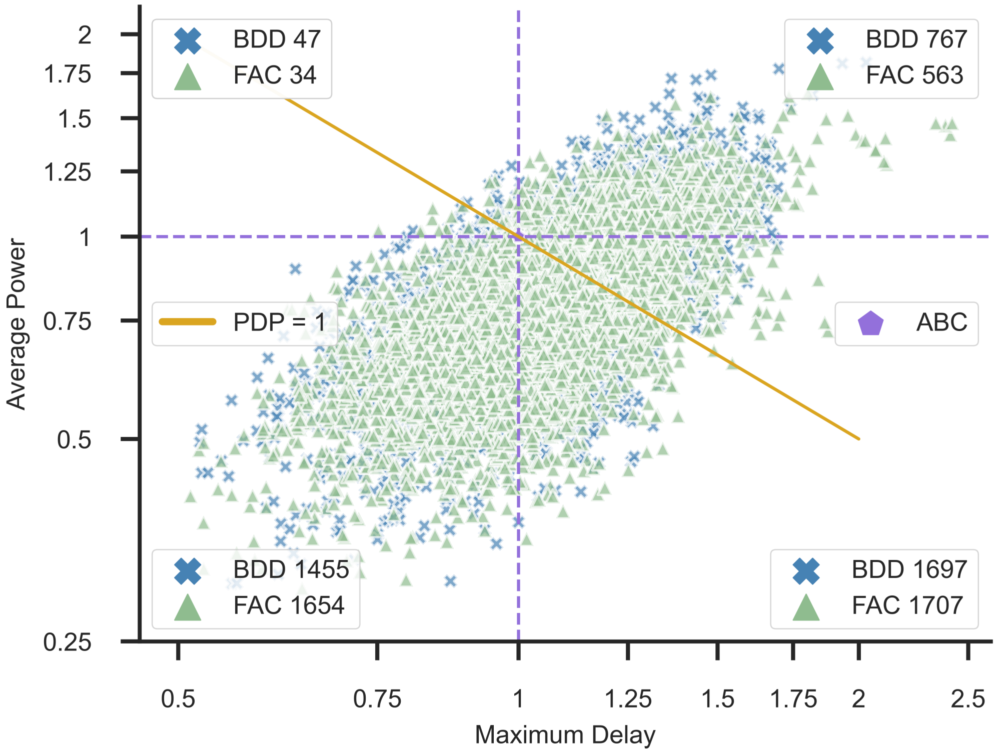


Figure 20 – Average power and critical delay scatter plot for each function of the 4 input P-Class.

being capable of finding or generating Euler paths to improve the gate layout efficiency.

Fig. 20 presents the results for all logic gates in the 4 input P-Class for critical propagation delay and power dissipation designed with the FAC and BDD methods and normalized to ABC mapped circuits. The number of logic gates on each quadrant is detailed on their labels, please note that the sum of gates does not add up to 3982 due to some designs being identical to the ABC circuits, e.g. the logic functions on Table 9. Table 11 details the number of cases that supergates achieved better or worse power-delay-product compared to ABC circuits. Supergates under the yellow line in Fig. 20 display that FAC and BDD gates obtain better PDP in 70.9% and 64.5%, respectively. If a design selection technique is employed to pick the best of both supergates methodologies, 77.4% of the cases achieve better PDP, indicating that supergates may be suitable for a more efficient power-delay tradeoff.

Table 11 – Supergate power-delay-product comparison to ABC circuits.

	Power-Delay-Product Improvement	Power-Delay-Product Decrease
FAC	2823 (70.9%)	1145 (28.8%)
BDD	2569 (64.5%)	1409 (35.4%)
Best FAC / BDD	3080 (77.4%)	898 (22.6%)

Table 12 – Ratio of better power-delay trade-off for FAC supergates compared to ABC circuits for the 4 input P-Class.

Logic Levels	Highest Transistor Stack Size			
	2	3	4	5
1	75% 3 - 1	0% 0 - 2	0% 0 - 1	- 0 - 0
2	54% 6 - 11	56% 27 - 21	43% 10 - 13	0% 0 - 1
3	100% 10 - 0	84% 624 - 113	64% 460 - 259	14% 4 - 23
4	- 0 - 0	89% 285 - 33	57% 437 - 331	12% 10 - 68
5	- 0 - 0	99% 295 - 3	71% 506 - 201	19% 10 - 41
6	- 0 - 0	100% 5 - 0	85% 119 - 21	71% 5 - 2

ABC-mapped circuits achieve worse electrical characteristics in either power or delay in 85% of the studied cases when compared to FAC, and 80% to BDD. Also, FAC and BDD reduce the power dissipation in at least 80% of the studied cases. With that in mind, it still is not clear to the author how to identify when a supergate will achieve better electrical behavior prior to an electrical simulation. No clear correlation was found when observing the reduction in the count of transistors with improvements to electrical characteristics.

Table 12 presents the ratio FAC supergates achieve better PDP compared to the ABC counterpart across the 4 input P-Class. Each case is classified according to the number of logic levels from the ABC design and the highest transistor stack in the FAC supergate. For instance, 84% of the circuits that have 3 logic levels and a transistor stack of 3 (Fig. 21 fits this case) obtains a better power-delay trade-off when using a supergate design. These results indicate that a FAC supergate is more likely to achieve a better PDP when the number of logic levels is higher than the supergate stack.

An indication of smaller power dissipation is the reduction of logic levels in a circuit. Using complex gates applied to the 4 input P-Class results in single-level circuits (disregarding input/output inverters), however, when using a technology mapping approach usually more than a single level is needed. Considering that not all inputs cause a change on the circuit output, multi-level circuits will result in a significantly higher power

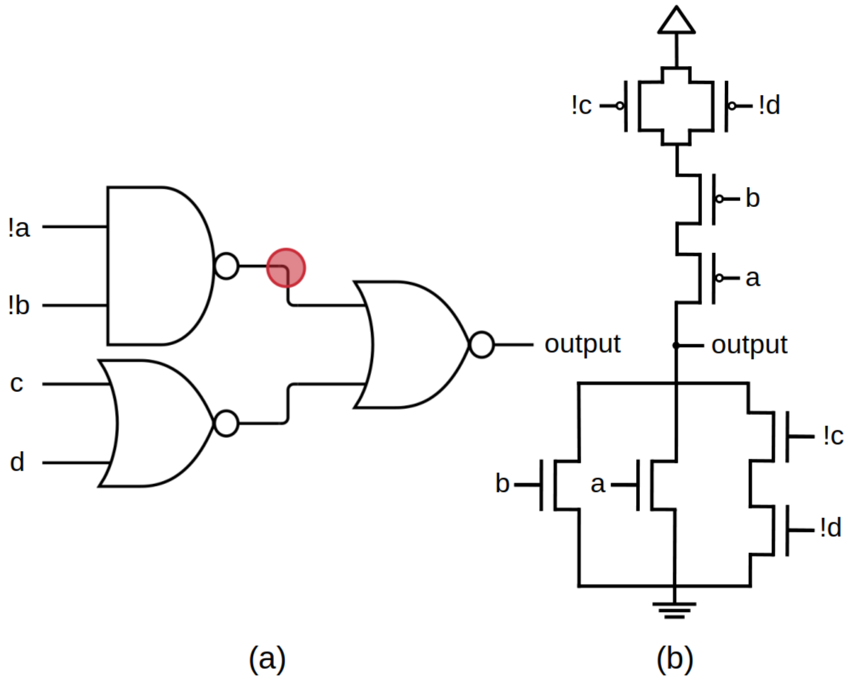


Figure 21 – Circuit design for (6) using ABC technology mapping (a) and supergate design by SwitchCraft (b).

dissipation due to spurious transitions (glitches), which causes logic gates switches despite not changing the circuit output.

Fig. 21 illustrates a technology mapped circuit (a) and an equivalent supergate (b) for F_{10} (6). Considering all inputs as '0' and changing the value of 'a', creates a spurious signal at the NAND gate output (node in red). Fig. 22 shows a waveform representation for both circuits, where no output switching is present on the supergate dissipating significantly less power. To reduce the power dissipation of glitches on standard cell logic circuits, works propose path balancing through gate resizing (WANG et al., 2011) or buffer insertion (KIM; KIM; HWANG, 2001). Supergates on the other hand may reduce the impact of glitches on targeted blocks that could be replaced by a single supergate.

$$F_{10} = \bar{a}(\bar{b}(c + d)) \tag{6}$$

The results show that supergates achieve better power-delay efficiency (70.9% studied cases) due to a considerable reduction in power dissipation across 84.4% logic functions while reducing the circuit area by 12.9%. Considering critical delay, supergates achieved, on average, 5.8% worst metrics, indicating that supergates may be better suited for low-power applications. The author believes that the reduction of logic levels may benefit a system design due to the glitch power reduction. It still is, however, an open challenge on how a technology mapper can handle supergates efficiently in a system design. This chapter's results indicate that a technology mapper with this capability could lead to circuits with a significantly smaller area and power.

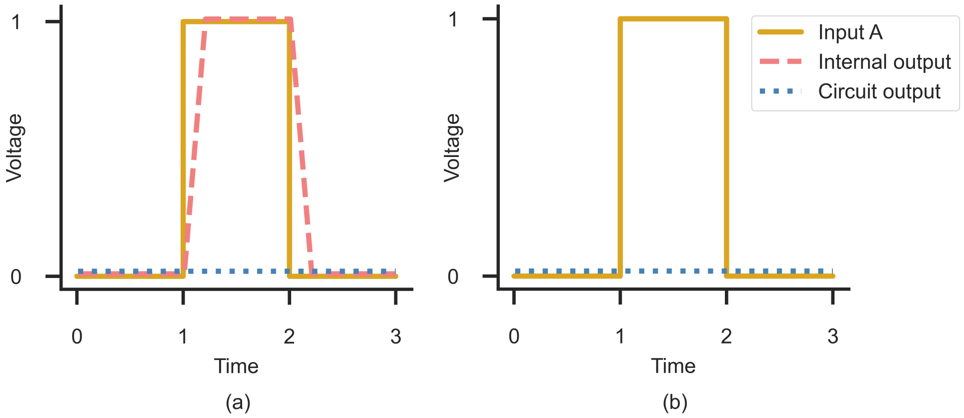


Figure 22 – Voltage waveform for Fig. 21, (a) illustrates the NAND gate output change on Fig. 21 (a) and (b) no node switching is present on the supergate design Fig. 21 (b).

6.3 AES S-BOX

This section presents a study on an Advanced Encryption Standard (AES) circuit on the substitution box (S-BOX) module. This block has 8 inputs and outputs and it is responsible for scrambling the message with the key. The circuit was designed with the TSMC 45nm standard-cell library using the Genus framework for Cadence. The simulation environment is FIN8 and FO4, and transistor sizing is made through Logical Effort at the logic gate level using the value of 1.32 as PMOS/NMOS ratio.

The resulting VERILOG file contains 874 logic gates and it is transformed in a SPICE file format with an in-house developed tool. Given the SPICE file format, the following steps are taken to identify logic gate candidates to be replaced by a supergate.

1. Count the occurrences of each node;
2. For each node, if occurrences equals 2, add node to usable node list;
3. For each logic gate, if usable node list contains all inputs, add logic gate to usable gate list;
4. For each usable gate, design a supergate using FAC method and replace both inputs and the logic gate;

In other words, if all inputs in a logic gate are unitary (all inputs are only connected to the specific logic gate), it becomes a candidate for a supergate replacement. If the designed supergate is an in-fact a supergate (it may result in an inverter, NAND, or NOR) all input gates and the logic gate are replaced by a supergate. Fig. 23 illustrates both examples of a usable and a non-usable logic gate. In the green solid line box all inputs are unitary, thus being a usable logic gate. On the other hand, the red dashed line box contains one input with a non-unitary connection, thus being a non-usable logic gate. Table 13 describes all the logic gates and their equations replaced by the proposed methodology. Fig. 23 case is described by the NAND-NAND-NOR line in Table 13.

Applying the proposed supergate insertion methodology to the AES S-BOX circuit,

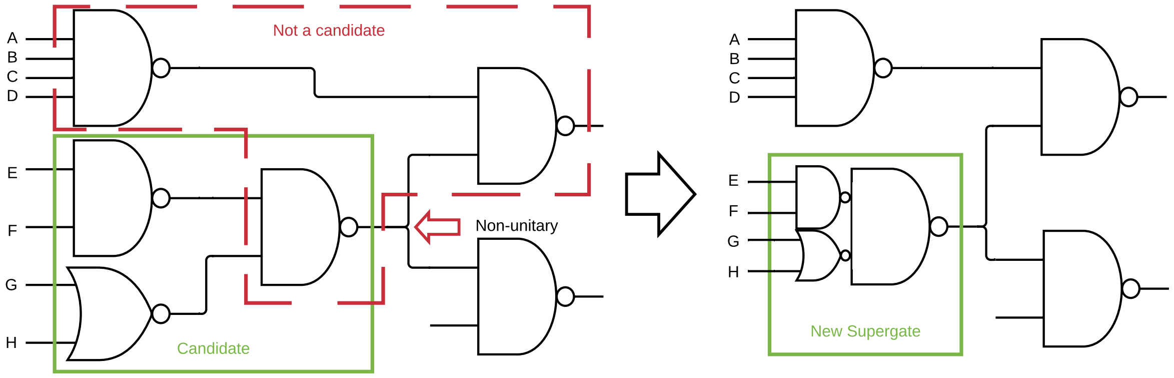


Figure 23 – Example of a usable (green box) and a non-usable (red dashed box) sub-circuit for a supergate replacement.

Table 13 – List of logic gates and inputs replaced for an equivalent supergate.

Logic Gate	Input 1	Input 2	Equation
OR	NAND	NOR	(!(a*b)+!(c+d))
AND	NOR	NAND	(!(a+b)*!(c*d))
NOR	AND	NOR	!((a*b)+!(c+d))
NOR	OR	NOR	!((a+b)+!(c+d))
NOR	NAND	NOR	!(!(a*b)+!(c+d))
NOR	NOR	NOR	!(!(a+b)+!(c+d))
NAND	NAND	AND	!(!(a*b)*(c*d))
NAND	NAND	NAND	!(!(a*b)*!(c*d))
NAND	NAND	NOR	!(!(a*b)*!(c+d))

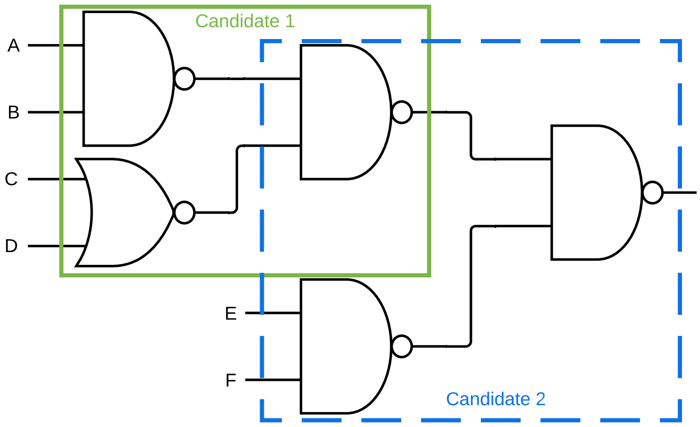


Figure 24 – Example of two overlapping supergates candidates. Candidate 1 (green box) is closer to the primary input, thus it has priority over Candidate 2 (blue dashed box).

Table 14 – List of occurrences of usable supergates and their impact on the electrical characteristics.

Complex Gate	Better	Worst
OR-NAND-NOR	0	1
AND-NOR-NAND	1	0
NOR-AND-NOR	1	0
NOR-OR-NOR	1	0
NOR-NAND-NOR	4	8
NOR-NOR-NOR	28	0
NAND-NAND-AND	0	3
NAND-NAND-NAND	15	24
NAND-NAND-NOR	3	10

resulted in 120 replacement candidates. Each replacement was evaluated individually, resulting in minor improvements in 69 (57.5%) cases in PDP and minor worsening in 51 (42.5%). It is important to note that not all individually designed supergates can be inserted in the circuit due to overlapping. Fig. 24 illustrates an example of two overlapping supergates candidates. In the proposed methodology we used a greedy solution to this situation, the first candidate (closer to the primary inputs) will always be chosen, resulting in 99 from 120 possible sub-circuits replacements.

Table 14 describe the count of each supergate occurrence using the proposed methodology (only non-overlapping supergates). If the inserted supergate improved the circuit PDP, it counts as one occurrence in column "Better", otherwise, the count goes to "Worst". A total of 53 (53.54%) replacements improved the circuit PDP against 46 (46.46%). The supergate that stands out the most is the NOR-NOR-NOR achieving better results in 28 of 28 occurrences. An indication of this result is the not-great electrical characteristics of the standard-cell NOR2 gate.

Fig. 25 illustrates all the results obtained from replacing the AES S-BOX sub-circuits with supergates. Each quadrant label contains the number of occurrences for itself. Analyzing the supergates results, most of the cases did not change the circuit critical delay (72 cases). As for those that changed, 30 improved, and 18 achieved worst results. Observing the power dissipation a split is achieved, the same amount of cases (60) achieved an improvement and a worsening of the power dissipation. This result indicates that blindly replacing sub-circuits for supergates without any electrical guidance may not improve the circuit electrical behavior.

As the last experiment, two more approaches to the AES S-BOX were simulated. Table 15 describes the electrical characteristics normalized to the standard-cell design. Using all usable (99 cases) supergates found by the proposed methodology, identified by "All", and only supergates that improved the circuit PDP (53 cases), identified by "Best". Results show that using all possible supergates despite reducing the transistor count by 5.7% it achieved a slightly worse PDP while increasing the area cost by 1.2%,

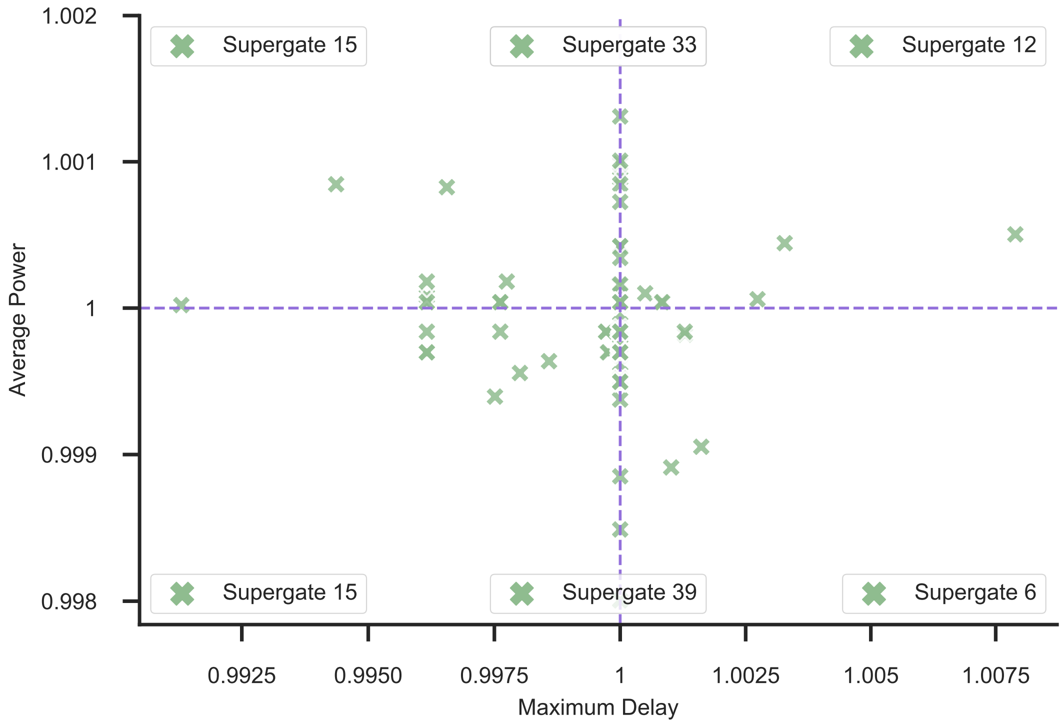


Figure 25 – Average Power and Maximum Delay scatter plot for each sub-circuit replacement for a supergate on the AES S-BOX circuit. The results are normalized to the standard-cell circuit.

Table 15 – AES S-BOX results normalized to the Standard Cell version.

	AES	All	Selected*
Critical Propagation Delay	1.000	1.006	0.997
Average Power	1.000	0.996	0.987
Power-Delay-Product	1.000	1.002	0.984
Transistor Count	1.000	0.943	0.969
Area estimation**	1.000	1.012	1.001

* Supergates that improved the PDP when inserted alone.

** Sum of transistors width.

on the other hand, "Best" supergates improved all electrical characteristics with an area increase of 0.01%. It is important to note that this area cost is not the circuit layout area. Supergates reduce the number of logic levels, thus decreasing the number of interconnections. As the work from Conceição (OLIVEIRA CONCEIÇÃO; LUZ REIS, 2019) stated, reducing the number of logic gates will decrease the circuit wirelength, increasing the circuit routability and it may yield a more area-efficient design.

The results indicate that if a methodology capable of using the electrical behavior of supergates as a metric, further improvement on the circuit could be achieved. In this chapter, only 9 supergates were designed, if these were characterized, an STA (Static Timing Analysis) approach could be applied, guiding if a specific sub-circuit replacement would improve the electrical characteristics.

The proposed supergate insertion methodology is rudimentary and limited, it considers only two logic levels. As discussed in Chapter 6.2, supergates gains are significantly bigger when replacing sub-circuits with a higher number of logic levels. The design freedom of on-the-fly supergates is not explored. Designing supergates as a post-processing step on circuits that are mostly NAND, AND, NOR, OR gates limits the design space and still electrical improvement can be achieved. This chapter uses Logical Effort sizing, supergates, in particular, should benefit significantly from other sizing values (as discussed in Chapter 5). However, to circumvent this problem, in the author's point of view, the whole logic synthesis must get along with supergates. This is a massive challenge, the logic synthesis by itself has to be competitive with commercial tools, both in quality of the design and execution time, while considering the possibility of using supergates. The author believes that using supergates to reduce the circuit power dissipation is probably easier (not easy) than delay optimization. Applying a proper transistor sizing to supergates on non-critical paths is a possible insertion location to achieve smaller power dissipation. The study presented in this chapter was published in ISCAS 2022 (Appendix A.1.3).

7 CONCLUSIONS

The benefits of using supergates for the design of VLSI are still an open question. From the technology mapping to the supergate design, it is a challenge to improve the quality of a resulting circuit using the design on-the-fly approach. This thesis provided an electrical analysis to guide works that would use supergates in their design. In this work, a comparison of supergate design strategies, transistor reordering, and transistor sizing was presented. To achieve the best possible supergate design, all of these aspects must be considered.

The design of a supergate is complex and requires optimizations during the network generation, reordering, sizing, and other variants not included in this thesis. All of these aspects are strongly related to the gate's electrical characteristics. Regarding the transistor network generation method, the most all-around technique is the FAC contained in SwitchCraft. However, the performance of the design technique is strongly related to the logic function. A single design strategy is very unlikely to always design the best supergate. To obtain the most suitable supergate, multiple strategies should be explored. Transistor reordering may cause a supergate to be as twice as slow, using a transistor reordering algorithm, major improvements can be achieved. BDD supergates are heavily impaired due to their ordering, however, due to its usage of non-series-parallel association, the studied reordering algorithm is not effective. The role of transistor sizing in supergates is especially critical due to its design freedom. As the supergate stack grows, it may require a bigger transistor width to keep up with timing requirements. However, as presented in this work, this approach does not achieve a good power-delay trade-off.

A comparison between supergates and technology-mapped circuits on small logic functions presented that supergates can achieve significantly better electrical characteristics. In this thesis, a study on an AES S-BOX, synthesized by a commercial tool, was performed to investigate the viability of supergates on larger circuits. The usage of supergates was limited due to the proposed insertion methodology only finding 9 different designs, and even in that scenario, electrical gains could be achieved. As their potential was not fully explored, the author believes that further study on a more sophisticated supergate insertion would increase the electrical benefits.

7.1 Direction for Supergates Onward

During this thesis, several topics regarding supergates were studied and evaluated, however, none is as deep as it can get. This section aims to direct future works regarding supergates.

Supergates Design Strategies: Two other logic function optimization techniques could be used to design supergates. Using Read-Once or disjoint-support decomposition optimizations from (CALLEGARO, 2016) and Function Composition (MARTINS et al., 2010) to optimize the logic function of each CMOS plane individually, may result in supergates similar to FAC. The author believes that they may achieve competitive electrical characteristics.

Transistor Reordering: A technique capable of reordering a transistor network with non-series-parallel should greatly benefit BDD and KF supergates. This reordering should consider a critical path analysis, it must not optimize both pull-up and pull-down networks individually and should aim to reduce the overall internal capacitance on slower paths.

Transistor Sizing: Sizing is a delicate subject due to its restrictions based on the design rules. A discrete sizing for improved power-delay trade-off for supergates would greatly boost their performance. A study on non-equal sizes for transistors on the same non-critical-path may reduce the internal capacitances on the output node, thus improving the supergate power and delay.

Supergate Compound Evaluation: SPICE simulations are time-consuming, running multiple simulations is computationally expensive. This work has not evaluated the compound modifications here studied. Simulating all supergates strategies, with their transistors properly reordered and sized is the path to find the best supergate possible.

Supergate Logic Synthesis: As discussed before, it is a massive challenge. Using the electrical characteristics may be the key to finding a spot for supergates during the logic synthesis of VLSI circuits. Using topological metrics may not be strict enough to guide a design. The results in this thesis indicate that a possible niche for supergates is to reduce the circuit power dissipation. This could be achieved if supergates are inserted in non-critical paths, where even if a loss of delay is obtained, it will not decrease the circuit operating frequency (clock).

Supergate Layout Characteristics: Having the supergate layout would increase the accuracy of the simulations as it provides parasitics. Running the same experiments here studied with a tool capable of automatically designing supergates layouts would further validate if the directions for their design were efficient.

Supergates on Multigate Devices: This work uses planar-bulk CMOS 45nm transistors, a study using, for example, FinFET technology, could further boost or decrease the gains of supergates.

If the presented future works were done, the niche for supergates maybe finally found.

REFERENCES

ALBINAGORTA, K. A. C.; CONCEIÇÃO, C.; CARDENAS, C. S.; REIS, R. Exploring area and total wirelength using a cell merging technique. In: IFIP/IEEE 27TH INTERNATIONAL CONFERENCE ON VERY LARGE SCALE INTEGRATION (VLSI-SOC), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p.329–334.

BRAYTON, R. K. Factoring logic functions. **IBM Journal of research and development**, [S.l.], v.31, n.2, p.187–198, 1987.

BRAYTON, R.; MISHCHENKO, A. ABC: An academic industrial-strength verification tool. In: INTERNATIONAL CONFERENCE ON COMPUTER AIDED VERIFICATION, 2010. **Anais...** [S.l.: s.n.], 2010. p.24–40.

BUTZEN, P. F.; DAL BEM, V.; REIS, A. I.; RIBAS, R. P. Design of CMOS logic gates with enhanced robustness against aging degradation. **Microelectronics Rel.**, [S.l.], v.52, n.9-10, p.1822–1826, 2012.

CALLEGARO, V. **Minimização ótima de classes especiais de funções booleanas**. 2016. Tese (Doutorado em Ciência da Computação) — Tese (Programa de Pós-Graduação em Microeletrônica)–Instituto de Informática . . . .

CALLEGARO, V. et al. SwitchCraft: a framework for transistor network design. In: INTEGRATED CIRCUITS AND SYSTEM DESIGN, 23., 2010. **Proceedings...** [S.l.: s.n.], 2010. p.49–53.

CARDOSO, M. et al. Transistor Placement for Automatic Cell Synthesis through Boolean Satisfiability. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (ISCAS), 2020., 2020. **Anais...** [S.l.: s.n.], 2020. p.1–5.

CARDOSO, M. S. et al. Physical design of supergate cells aiming geometrical optimizations. In: IEEE 59TH INTERNATIONAL MIDWEST SYMPOSIUM ON CIRCUITS AND SYSTEMS (MWSCAS), 2016., 2016. **Anais...** [S.l.: s.n.], 2016. p.1–4.

CARDOSO, M. S. et al. Libra: an automatic design methodology for CMOS complex gates. **IEEE Trans. Circuits Syst., II, Exp. Briefs**, [S.l.], v.65, n.10, p.1345–1349, 2018.

CARLSON, B. S.; LEE, S.-J. Delay optimization of digital CMOS VLSI circuits by transistor reordering. **IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.**, [S.l.], v.14, n.10, p.1183–1192, 1995.

CHUN, J. W.; CHEN, C. R. Transistor and pin reordering for leakage reduction in CMOS circuits. **Microelectronics J.**, [S.l.], v.53, p.25–34, 2016.

DUAN, S.; ZWOLINSKI, M.; HALAK, B. Lifetime Rel.-Aware Digital Synthesis. **IEEE Trans. Very Large Scale Integr. (VLSI) Syst**, [S.l.], v.26, n.11, p.2205–2216, Nov 2018.

FATEMI, H. et al. Enhancing sensitivity-based power reduction for an industry IC design context. **Integration**, [S.l.], v.66, p.96–111, 2019.

FLACH, G. et al. Effective method for simultaneous gate sizing and v th assignment using lagrangian relaxation. **IEEE transactions on computer-aided design of integrated circuits and systems**, [S.l.], v.33, n.4, p.546–557, 2014.

GLEBOV, A. L.; BLAAUW, D.; JONES, L. G. Transistor reordering for low power CMOS gates using an SP-BDD representation. In: INT. SYMP. ON LOW POWER DES., 1995., 1995. **Proceedings. . .** [S.l.: s.n.], 1995. p.161–166.

GUPTA, P.; GOURISHETTY, S.; MANDADAPU, H.; ABBAS, Z. PVT Variations Aware Robust Transistor Sizing for Power-Delay Optimal CMOS Digital Circuit Design. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (ISCAS), 2019., 2019. **Anais. . .** [S.l.: s.n.], 2019. p.1–5.

HESHAM, A.; NASSAR, A.; MOSTAFA, H. Energy-Efficient Near-Threshold Standard Cell Library for IoT Applications. In: NOVEL INTELLIGENT AND LEADING EMERGING SCIENCES CONFERENCE (NILES), 2020., 2020. **Anais. . .** [S.l.: s.n.], 2020. p.567–572.

HOSSAIN, R.; ZHENG, M.; ALBICKI, A. Reducing power dissipation in CMOS circuits by signal probability based transistor reordering. **IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.**, [S.l.], v.15, n.3, p.361–368, 1996.

HU, J. et al. Sensitivity-guided metaheuristics for accurate discrete gate sizing. In: INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, 2012. **Proceedings. . .** [S.l.: s.n.], 2012. p.233–239.

KESSLER, H. et al. Evaluation of Non-Series-Parallel Structures for BTI-Aware Automated Design Methodologies. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (ISCAS), 2020., 2020. **Anais...** [S.l.: s.n.], 2020. p.1–5.

KIAMEHR, S.; FIROUZI, F.; TAHOORI, M. B. Input and transistor reordering for NBTI and HCI reduction in complex CMOS gates. In: SYMP. ON VLSI, 2012. **Proceedings...** [S.l.: s.n.], 2012. p.201–206.

KIM, S.; KIM, J.; HWANG, S.-Y. New path balancing algorithm for glitch power reduction. **IEE Proceedings-Circuits, Devices and Systems**, [S.l.], v.148, n.3, p.151–156, 2001.

LEVI, I.; BELENKY, A.; FISH, A. Logical effort for CMOS-based dual mode logic gates. **IEEE Transactions on very large scale integration (VLSI) systems**, [S.l.], v.22, n.5, p.1042–1053, 2013.

MACHADO, L. et al. KL-cut based digital circuit remapping. In: NORCHIP 2012, 2012. **Anais...** [S.l.: s.n.], 2012. p.1–4.

MACHADO, L. et al. Iterative remapping respecting timing constraints. In: IEEE COMPUTER SOCIETY ANNUAL SYMPOSIUM ON VLSI (ISVLSI), 2013., 2013. **Anais...** [S.l.: s.n.], 2013. p.236–241.

MAREK-SADOWSKA, M.; LIN, S. P. Pin assignment for improved performance in standard cell design. In: INT. CONF. ON COMPUT. DES.: VLSI IN COMPUT. AND PROCESSORS, 1990. **Proceedings...** [S.l.: s.n.], 1990. p.339–342.

MARQUES, F. S. et al. DAG based library-free technology mapping. In: ACM GREAT LAKES SYMPOSIUM ON VLSI, 17., 2007. **Proceedings...** [S.l.: s.n.], 2007. p.293–298.

MARTINS, M. G. et al. Boolean factoring with multi-objective goals. In: IEEE INT. CONF. ON COMPUTER DES., 2010., 2010. **Anais...** [S.l.: s.n.], 2010. p.229–234.

MATOS, J. M.; CARRABINA, J.; REIS, A. Efficiently mapping VLSI circuits with simple cells. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, [S.l.], v.38, n.4, p.692–704, 2018.

MUSOLL, E.; CORTADELLA, J. Optimizing CMOS circuits for low power using transistor reordering. In: ED&TC EUROPEAN DES. AND TEST CONF., 1996. **Proceedings...** [S.l.: s.n.], 1996. p.219–223.

NING, W. Strongly NP-hard discrete gate-sizing problems. **IEEE transactions on computer-aided design of integrated circuits and systems**, [S.l.], v.13, n.8, p.1045–1051, 1994.

OLIVEIRA CONCEIÇÃO, C. M. de; LUZ REIS, R. A. da. Transistor count reduction by gate merging. **IEEE Transactions on Circuits and Systems I: Regular Papers**, [S.l.], v.66, n.6, p.2175–2187, 2019.

POSSANI, V. N. et al. Graph-based transistor network generation method for supergate design. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems**, [S.l.], v.24, n.2, p.692–705, 2015.

PRASAD, S.; ROY, K. Transistor reordering for power minimization under delay constraint. **ACM Trans. on Des. Automation of Electronic Syst. (TODAES)**, [S.l.], v.1, n.2, p.280–300, 1996.

RAHMAN, M.; TENNAKOON, H.; SECHEN, C. Library-based cell-size selection using extended logical effort. **IEEE transactions on computer-aided design of integrated circuits and systems**, [S.l.], v.32, n.7, p.1086–1099, 2013.

REIS, R. et al. Circuit Level Design Methods to Mitigate Soft Errors. In: IEEE LATIN-AMERICAN TEST SYMP. (LATS), 2020., 2020. **Anais...** [S.l.: s.n.], 2020. p.1–3.

RICCI, A.; DE MUNARI, I.; CIAMPOLINI, P. An evolutionary approach for standard-cell library reduction. In: ACM GREAT LAKES SYMPOSIUM ON VLSI, 17., 2007. **Proceedings...** [S.l.: s.n.], 2007. p.305–310.

ROSA JUNIOR, L. S. da et al. Fast disjoint transistor networks from BDDs. In: INTEGRATED CIRCUITS AND SYSTEMS DESIGN, 19., 2006. **Proceedings...** [S.l.: s.n.], 2006. p.137–142.

ROSA, L. S. da; SCHNEIDER, F. R.; RIBAS, R. P.; REIS, A. I. Switch level optimization of digital CMOS gate networks. In: INTERNATIONAL SYMPOSIUM ON QUALITY ELECTRONIC DESIGN, 2009., 2009. **Anais...** [S.l.: s.n.], 2009. p.324–329.

SASAO, T. **Switching theory for logic synthesis**. [S.l.]: Springer Science & Business Media, 2012.

SENTOVICH, E. M. et al. SIS: A system for sequential circuit synthesis. **University of California, Berkeley**, [S.l.], 1992.

SHAHOOKAR, K.; MAZUMDER, P. VLSI cell placement techniques. **ACM Computing Surveys (CSUR)**, [S.l.], v.23, n.2, p.143–220, 1991.

SHEN, W.-Z.; LIN, J.-Y.; WANG, F.-W. Transistor reordering rules for power reduction in CMOS gates. In: ASP-DAC'95/CHDL'95/VLSI'95 WITH EDA TECHNOFAIR, 1995. **Proceedings...** [S.l.: s.n.], 1995. p.1–6.

SMANIOTTO, G. et al. A post-processing methodology to improve the automatic design of cmos gates at layout-level. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRONICS, CIRCUITS AND SYSTEMS (ICECS), 2017., 2017. **Anais...** [S.l.: s.n.], 2017. p.42–45.

SUTHERLAND, I.; SPROULL, R. F.; SPROULL, B.; HARRIS, D. **Logical effort**: designing fast CMOS circuits. [S.l.]: Morgan Kaufmann, 1999.

UEHARA, T. et al. Optimal layout of CMOS functional arrays. In: DESIGN AUTOMATION CONFERENCE, 16., 1979. **Anais...** [S.l.: s.n.], 1979. p.287–289.

WANG, L. et al. A gate sizing method for glitch power reduction. In: IEEE INTERNATIONAL SOC CONFERENCE, 2011., 2011. **Anais...** [S.l.: s.n.], 2011. p.24–29.

Appendices

APPENDIX A – LIST OF PUBLICATIONS

A.1.1 Journal of Integrated Circuits and Systems

- KESSLER, Henrique.; MUÑOZ, Marcello. ; FINKENAUER, Plínio. ; ROSA JR, Leomar da. ; CAMARGO, Vinícius. (2021). Electrical Evaluation of Logic Network Generation Methods for On-the-Fly Supergate Design. Journal of Integrated Circuits and Systems, 16(3), 1–7. https://doi.org/10.29292/jics.v16i3.527

A.1.2 13th Latin American Symposium on Circuits & Systems (LASCAS 2022)

- KESSLER, Henrique.; BOHLKE, Murilo. ; PORTO, Marcelo. ; ROSA JR, Leomar da. ; CAMARGO, Vinícius., "Calibration of Logical Effort Transistor Sizing for On-the-Fly Low-Power Supergate Design," 13th Latin American Symposium on Circuits and Systems (LASCAS), 2022.

A.1.3 2022 IEEE International Symposium on Circuits & Systems (ISCAS)

- KESSLER, Henrique.; ROSA JR, Leomar da. ; PORTO, Marcelo. ; CAMARGO, Vinícius., "Standard Cell and Supergates Designs: an Electrical Comparison on 4-Input Logic Functions," 2022 IEEE International Symposium on Circuits & Systems (ISCAS), 2022.