UNIVERSITY
OF OSLO

Master's thesis

Cryptographic hardware resistant to Leakage-Based Differential Power Analysis Attacks

Magnus Amble

Electronics, Informatics and Technology
60 ECTS study points

Department of Informatics
Faculty of Mathematics and Natural Sciences

Spring 2023

Magnus Amble

Cryptographic hardware resistant to Leakage-Based Differential Power Analysis Attacks

Abstract

As cryptographic encryption algorithms have progressed in complexity, adversaries have looked to other means of accessing encrypted secret data than traditional cryptanalysis. One such alternative method of intrusion is through side-channel analysis, where physical properties of a cryptographic circuit are analyzed to reveal sensitive data.

This thesis presents analyses of countermeasures against side-channel analysis based on variations in the power consumption of a circuit. A trend seen in the advancement of Complementary Metal–Oxide–Semiconductor(CMOS) technology is an increase in static power consumption. In the context of cryptographic circuits, this aspect of the power consumption has proven to contain information about sensitive data processed by the circuit, making it a potential attack vector for an adversary.

Three gate-level countermeasures against leakage power analysis attacks are proposed in this thesis. Two of which are logic styles based on the equalization of a logic gate's power consumption between input combinations, coined Octuple logic and Quadruple Dual-Pulldown Logic (QDPL). Both of these countermeasures expand on the concept of a countermeasure from the literature called Exhaustive Logic Balancing (ELB). The third proposed countermeasure is a current-masking scheme that introduces input-independent noise in the power consumption through periodically randomizing the voltage on the bulk-terminals of the logic gate. Combining such a logic style with the bulk-voltage masking scheme lowers the possible signal-to-noise ratio obtainable by an adversary by both adding additional noise to, and by lowering the data-dependency in, a circuits power consumption. Preliminary experiments showed that the masking scheme is able to modulate the static current of a ELB NAND gate within an interval where the highest current is six times larger than the lowest current. Subsequently, rough estimates show that if signal averaging analyses are conducted on a register implemented in QDPL, the mask could make the analyses require over 800000 additional signal measurements, compared to when no mask is applied, to find a one-bit hamming weight difference in the register's static current.

Preface

This thesis was completed as the final project of the master's program Electronics, Informatics and technology at the Department of Informatics, University of Oslo. The work for this thesis started in September 2021, and was completed in May 2023.

I would like to express my sincere gratitude to my supervisors Snorre Aunet, Dag T. Wisland and Kristian G. Kjelgård for their guidance, feedback and productive discussions through all the work that went into this thesis.

I would also like to thank my family and friends for all their continued encouragement and support through the writing of this thesis.

Magnus Amble

Oslo, May 2023

Contents

List of Figures

List of Tables

Acronyms and abbreviations

BSPL Balanced Static Power Logic.

CMOS Complementary Metal–Oxide–Semiconductor.

CPA Correlation Power Analysis.

DAC Digital to Analog Converter.

DDPL Delay-Based Dual-rail Precharge Logic.

DPA Differential Power Analysis.

ELB Exhaustive Logic Balancing.

FDSOI Fully depleted silicon-on-insulator.

LDPA Leakage-Based Differential Power Analysis.

LFSR Linear Feedback Shift Register.

LHPA Leakage-Based Hamming Weight Power Analysis.

LPA Leakage power analysis.

MDPL Masked Dual-rail Pre-charge Logic.

PAA Power Analysis Attacks.

QDPL Quadruple Dual-Pulldown Logic.

RSD Relative Standard Deviation.

SABL Sense Amplifier Based Logic.

SCA Side-Channel Analysis.

SDRL Symmetric Dual-Rail Logic.

SNR Signal-to-noise-ratio.

SPA Simple Power Analysis.

TPDL Three-phase Dual-rail Pre-charge Logic.

TRNG True Random Number Generator.

WDDL Wave Dynamic Differential Logic.

1 Introduction

Encryption technology has a broad spectrum of applications including various forms of electronic communication, e-commerce transactions and electronic banking to name a few [1, 2]. The aim of cryptography is to ensure that confidential information remains private by limiting access to only those who have been authorized to use it. In light of this, security is of utmost importance in most implementations of encryption technology. Advanced encryption algorithms have been developed where breaking the encryption through cryptanalysis poses a monumental challenge. However, when such algorithms are implemented in hardware they can leak information through so-called side-channels[3]. Side-Channel Analysis (SCA) can then be used by an adversary where physical properties of the cryptographic device such as for example power consumption, timing variations or electromagnetic radiation is analyzed to obtain sensitive data about the system[4]. In this regard SCA effectively allows adversaries to circumvent traditional cryptanalysis and obtain sensitive data through other means.

One of these side-channels that has proven to make cryptographic circuits vulnerable to SCA is the power consumption of the circuit. Attacks that utilize this side-channel analyze variations in the power consumption of a cryptographic circuit to reveal secret data that is being processed by the circuit[5], this category of attacks is commonly referred to as Power Analysis Attacks (PAA). The variations in the power consumption can be traced back to the individual logic gates that make up the circuit. An undesirable property of traditional Complementary Metal–Oxide–Semiconductor (CMOS) logic gates is that their power consumption varies depending on the input pattern that is being applied to its input pins. This makes their current consumption directly dependent on their input pattern which will subsequently make any larger system consisting of such logic gates have a data-dependent current consumption.

Generally the term PAA serves as an umbrella term that encompasses any form of attack that is based on analyses of a circuit's power consumption. The categorization of PAA is primarily based on what aspect of the power consumption that is being analyzed. Specifically PAA can be divided into two main subcategories: leakage power analysis and dynamic power analysis. It has been shown that both the dynamic power[5] and the leakage power[6] exhibit data dependence in CMOS implementations, and thus both present an entry point for PAA . The forms of attacks within a given category exhibit varying degrees of complexity. For instance, in the dynamic current based category there exists attacks such as Simple Power Analysis (SPA) and Differential Power Analysis (DPA)[5]. Similarly in the leakage based category, often referred to as Leakage power analysis (LPA)[7, 8, 9], there exists attacks such as Leakage-Based Differential Power Analysis (LDPA)[10].

Traditionally the dynamic current consumption of the circuits was the subject of such analyses[5, 6, 10] but as transistor technology continues to decrease in size the leakage current is becoming an increasingly large fraction of the total current consumption. Successful power analysis attacks have been conducted using the leakage power[10, 7], showing that variations in the leakage power contains information about the data that is being processed and can be exploited to gain access to secret data[6]. This thesis will primarily focus on the input dependence of the static power consumption when considering different logic styles, but a truly PAA resistant logic style will have to be resistant to both static and dynamic power analysis. And thus, the dynamic power consumption can not be ignored.

One approach to solving the issue of information leakage is to take a ground up approach by stopping the leakage at the lowest abstraction level possible. Where ideally each individual building block of a system would have zero side-channel information leakage. The idea being that any subsequent larger system that is created with such secure building blocks would inherit the zero-leakage properties from the individual building blocks.

In the case of PAA resistant cryptographic circuits, such a ground-up approach can be taken by building the circuit from individual logic gates that are free from side-channel information leakage. PAA resistant logic gates are normally split up into two main categories. The first is through decoupling the input pattern to the logic gate from the current it consumes, where a common approach is to design logic gates that consume the same amount of power regardless of the input pattern. These approaches are often referred to as hiding mechanisms. The second main category is called masking. When done at the gate level, it usually entails combining the input and output signals of the logic gate with a randomly generated mask value. This is done through, for example, an XOR operation, and it obfuscates the correlation between the logic gate's power consumption and the original input signals[11]. There can be benefits to combining countermeasures from each main category, but when doing so one must take care to verify that one countermeasure does not weaken the effects of the other[12].

This thesis presents an analysis of a logic style in the hiding category called Exhaustive Logic Balancing (ELB) introduced in [13] which aims to have an input-independent static current. This logic style exhibits complete input-independence in its static current in ideal simulations, and this behavior is even maintained across different temperatures, but imbalances are observed in its dynamic current consumption.

Two new hiding logic-styles are proposed that expand on the ELB logic style. These two logic styles aim to keep the solid input-independence observed in the static current consumption of ELB logic, while expanding on its dynamic-current balancing capabilities. These two logic styles are coined Octuple logic and Quadruple Dual-Pulldown Logic (QDPL).

However, through experiments it will be shown that all the logic styles analyzed in this thesis lose their input-independence in the presence of intra-die mismatch, leading to a visible data-dependency in their static current. A masking scheme is then introduced in an attempt to address this. Unlike traditional masking countermeasures, which involve masking the input and output signals of a logic gate through a mathematical operation with a random mask value, this scheme utilizes periodically randomized voltages at the bulk nodes of the logic gate's internal transistors. This adds a layer of random and input-independent noise to the static current consumption of a logic gate. Experiments show that the added noise causes much larger variations in the static current than the input-dependency that is introduced by mismatch in logic styles like ELB or QDPL. A combination of such a logic style with the masking scheme therefore worsens an adversary's Signal-to-noise-ratio (SNR) both from the low data-dependency in the underlying hiding mechanism as well as with the noise from the mask. A combination of this kind of mask together with a hiding logic style like QDPL is therefore presented as a countermeasure against PAA. Alongside the work done in this thesis, a paper was written and submitted to a scientific conference included in appendix A, presenting the combination of the masking scheme together with ELB as a countermeasure.

2 Methods

Among the chapters in this thesis is a chapter presenting an evaluation of the input-dependency in the current consumption in multiple different logic styles. The same procedure was used in analyzing all the different logic styles to assess their robustness as countermeasures. This chapter contains an overview of the methods used when conducting these analyses.

2.1 Tools and simulation software

To be able to conduct accurate analyses of the logic gates based on simulation results, it is of utmost importance to use simulation software that provides accurate and realistic results. To achieve this, the state of the art design platform Cadence Virtuoso System Design Platform[14] version 6.1.7-64b was used for all circuit design as well as every simulation with the BSIM4[15] transistor model. All logic styles that simulations were performed for is implemented in 65 nm bulk CMOS technology. Throughout all the tests that were conducted for this thesis a supply voltage of $V_{DD} = 1.2\,\mathrm{V}$ was used. The logic levels '0' and '1' correspond to 0 V and 1.2 V respectively.

2.2 Transistor sizing

A transistor length of $L = 2 \cdot L_{min} = 120\,\mathrm{nm}$ is used for every transistor. When selecting widths for the transistors, simulations were conducted to ensure the logic gates had a symmetrical switching point of approximately $V_{switch} \approx \frac{V_{DD}}{2}$. Starting with a NMOS width of 120 nm, the width of the PMOS transistors were gradually increased until both input pins of the logic gate exhibited a roughly symmetric switching point. The switching point analyses were carried out using DC analysis, with one input signal swept from V_{DD} to GND at a time, and the simulation results were compared at each PMOS width to determine the optimal ratio between the widths of the NMOS and PMOS transistors for the most symmetric switching point for both inputs. The resulting PMOS transistor width to achieve this was 370 nm. If no other sizes are explicitly mentioned in the following sections in this thesis, these are the sizes that were used in the simulations.

2.3 Choice of logic gate

Wherever a single logic gate was analyzed for a given logic style, NAND gates were used as they have the ability to implement any other logic function through NAND-only logic. While this approach is far from optimal in terms of area and power consumption, a side-channel leakage-free NAND gate could theoretically serve as a fundamental building block with which larger systems free of side-channel leakages could be created.

2.4 Test benches for logic gates

In all tests that were conducted on individual logic gates, an ideal load of 100 fF was used on the logic gate's output. The size of this capacitive load was set to simulate the different logic gates performance in a high fan-out environment. Specifically what level of fan-out a 100 fF load corresponds to depends on the logic style in question. Ideal voltage sources were used as the input signals to the logic gates. These ideal components were chosen so that the performance of a given logic style could more easily be isolated. Another method that was used to isolate the performance of some of the logic styles that was analyzed in this thesis required both input signals and their inverses to be supplied to the logic gate. The inverse voltage sources in these test benches was set to a voltage of $\overline{V_x} = V_{DD} - V_x$ where V_x is either V_A or V_B. The reason why this method of supplying input signals results in isolation of the logic gates performance will be discussed further in the following chapter.

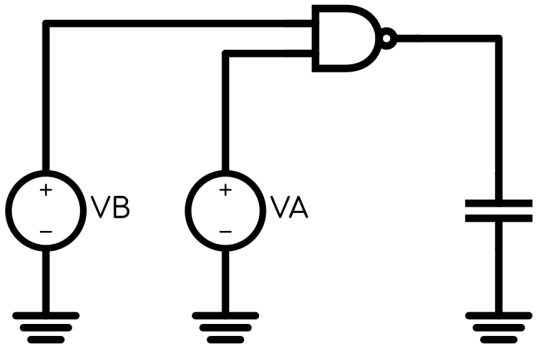

Figure 1: Test bench setup for a single NAND gate

For every logic style that is analyzed in the next chapter, the same experiments are repeated for all of them. The following sections gives an explanation of how those experiments were conducted and what test bench was used in them.

2.4.1 Obtaining the static current consumption of a logic gate for further analyses

When results that involve the static current of a logic style are presented in this thesis, it is often referred to a set of currents. The set that is referred to is a set containing the static current consumption of a logic gate for every possible input combination, and since only two-input logic gates are analyzed in this thesis, the sets contain four currents. To obtain the sets of currents, the test bench in fig. 1 was used to conduct four DC analyses where each input combination was given to the NAND gate while its current consumption was recorded. The resulting set of currents could then be used to calculate different metrics to evaluate the input-dependency in the current consumption of the logic gate.

This same test bench and procedure was utilized when conducting experiments involving temperature.

2.4.2 Evaluating input-dependency in the current consumption through a transition

Many aspects of a logic gate's current consumption will be analyzed to give a more thorough overview of any input-dependency that is found. One of these aspects was how the current consumed by a logic gate differs through a transition depending on which input pin triggered the transition. The test bench in fig. 1 could be used for this analysis as well. This experiment involved keeping one of the input-voltage sources static at V_{DD} while the other was taken through a sweep from GND to V_{DD} so as to ensure that the NAND gate underwent a transition. Additional experiments were conducted where both input-voltage sources are swept simultaneously to assess how the current consumption in such a scenario differs from a single-input transition, as a large difference could leak information of the state of the input signals through the power consumption.

Both DC and transient analyses were utilized to obtain these results as the current is evaluated in both the voltage domain and the time domain.

2.4.3 Evaluating data-dependency in the presence of mismatch

As will be discussed later in this thesis, intra-die mismatch can have a large impact on the input-dependency of a logic gate. The performance of a logic style in the presence of mismatch is therefore an important aspect to evaluate, to get a better understanding of the level of protection against information-leakage the logic style provides. Simulations that included mismatch were performed through Monte Carlo analyses. For reproducibility sake, the same Monte Carlo seed of 12345 and a starting point of 1 was used for every logic style where such analysis was conducted.

At points in the different analyses, a single iteration of the Monte Carlo simulation is picked out to act as a single instance of mismatch. This sort of analyses can give insight into how a real-world implementation of a logic gate would act, which is not the case when only dealing with averages of entire simulations. Wherever a single iteration of a Monte Carlo simulation is used as an example, iteration number five was used if otherwise is not explicitly stated. This iteration was picked out at random since there were no specific requirements any given iteration had to satisfy.

Monte Carlo simulations with 1000 iterations were used when statistical averages were gathered. It was observed that the average values of the recorded parameters had reached a point of stagnation by the 1000th iteration. As such, it was determined that running simulations beyond 1000 iterations was unlikely to yield any significant benefits. Therefore simulations of 1000 iterations were deemed adequate for the purpose of gathering statistical averages for the different logic styles.

Evaluating the variation in current for a given input combination

Two different experiments were conducted when evaluating the effects of mismatch. One utilized the test bench in fig. 1 where all 1000 iterations of the Monte Carlo iterations were repeated for every possible input combination to a NAND gate. The results of these experiments shows how much the current can change from one instance of mismatch to another at each input combination. The averaged metrics from this experiment gives an overview of the average magnitude of variation that is seen at each input combination. However, this analysis does not give an indication of how much the current changes between input combinations at a given instance of mismatch. So when the current consumed at each input combination is viewed as a set, this analysis does not give a measure of the variation within each set. To evaluate that aspect of mismatch, another experiment had to be conducted which is explained in the next section.

Evaluating the intra-set variation for a set of static currents

The other mismatch analysis that was conducted aimed to evaluate the variation that is seen within a set of currents containing the static current consumption of every input combination for a given case of mismatch. This gives an indication of how much the input-dependency changes from one instance of mismatch to another. And averages from all iterations can be calculated to give an indication of what level of input-dependency can be expected once mismatch is introduced.

A new test bench was required to be able to calculate metrics to evaluate input-dependency for a whole set of currents, at every iteration of the Monte Carlo simulation, and is shown in fig. 2. Here, four NAND gates of the logic style under evaluation were instantiated in the same test bench, each receiving a unique input combination so that they together make up a complete set of currents.

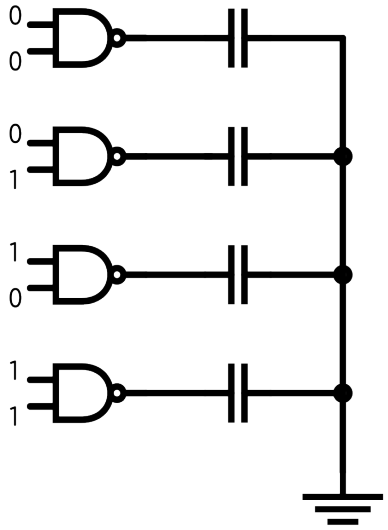

Figure 2: Test bench setup used for intra-set calculations for static current data sets

3 Gate level hiding countermeasures

This section presents the results of power-consumption analyses on NAND gates that are implemented in different logic styles. The analyses are conducted on multiple different aspects of the current consumption of the different logic gates to assess their input-dependency. First, the analysis results for standard CMOS logic are presented, followed by a presentation of the results for ELB. Subsequently, the logic styles Octuple logic and QDPL are introduced and analyzed. After these logic styles are analyzed comes a section discussing the issue of generating inverted input signals internally in a logic gate, and results are presented of an implementation an internal input generation scheme in a QDPL NAND gate. Finally, a comparison of all the logic styles is presented at the end of the section.

3.1 Standard CMOS logic

To get some baseline values for the variability of the static power consumption of a logic gate, a few simulations were conducted on a standard CMOS NAND gate. The topology of a standard CMOS NAND gate is depicted in fig. 3.

3.1.1 Static input DC analysis

Initial analyses were conducted to assess the extent to which the static current consumption this NAND gate topology would fluctuate depending on the input pattern that was applied to it. These metrics were gathered through the procedure described in section 2.4.1. The results of the simulation revealed significant variations in the current between each input combination as illustrated in fig. 4.

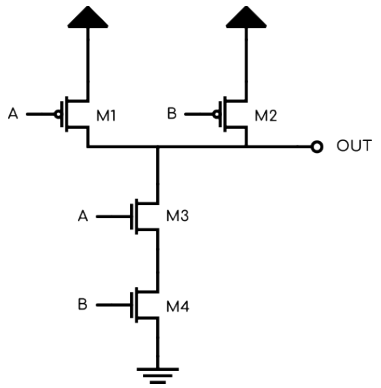

Figure 3: Standard CMOS NAND gate

The highest current is seen when the input is at a '11', at which point the current is 11.61 times higher than the lowest current which occurs at an input of '00'. The fluctuations in static current are mainly attributed to the varying number of transistors in cut-off mode for each input combination. This leads to a different number of transistors contributing their source-to-drain subthreshold leakage current to the total static current of the logic gate depending on the input combination[16]. The subthreshold current is mentioned specifically since in current technology it constitutes the largest fraction of the leakage current in a transistor[7].

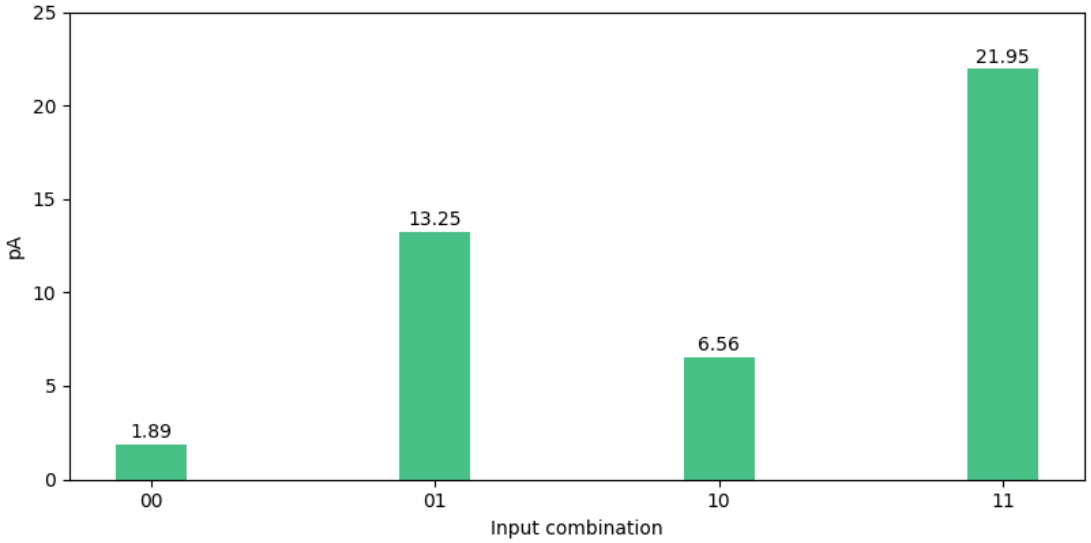

Figure 4: Static power consumption of a standard CMOS NAND gate for all input combinations

The static currents depicted in fig. 4 can be treated as a set of currents, which includes the static current at every input combination. Several metrics can be calculated from this set of currents and used as a benchmark for comparison later in this thesis when other logic styles are introduced. First, an interval of current values can be established, which in this instance was found to be:

$$1.89\,\text{pA} \leq I_{static} \leq 21.95\,\text{pA} \tag{1}$$

The mean of the set was calculated to be $\overline{I}_{static} = 10.91\,\text{pA}$. When considering the static current for each input combination as a set, the standard deviation of that set can serve as an indication of the input-dependency in the current[10]. The standard deviation in this analysis was found to be $\sigma = 7.54\,\text{pA}$. Another metric calculated for future comparison was the Relative Standard Deviation (RSD) which is derived from the standard deviation and the mean and provides a measure of the relative change within the set of currents compared to its mean. The currents from this experiment yielded an RSD of 69.13 %. Meaning that the variations in the static current was of such magnitude that the standard deviation equals 69.13 % of the mean value of the set of currents.

3.1.2 Input transition DC analysis

Most, if not all aspects of the current consumption of a standard CMOS NAND exhibits some input-dependency. The previous section established the current's dependence on the input signals when the logic gate is in a steady state. Next, the input-dependence in the current consumption through a transition of the input signals is examined. The procedure described in section 2.4.2 was used. This analysis was repeated two times, one time for each input signal. The resulting current consumption traces are plotted in fig. 5.

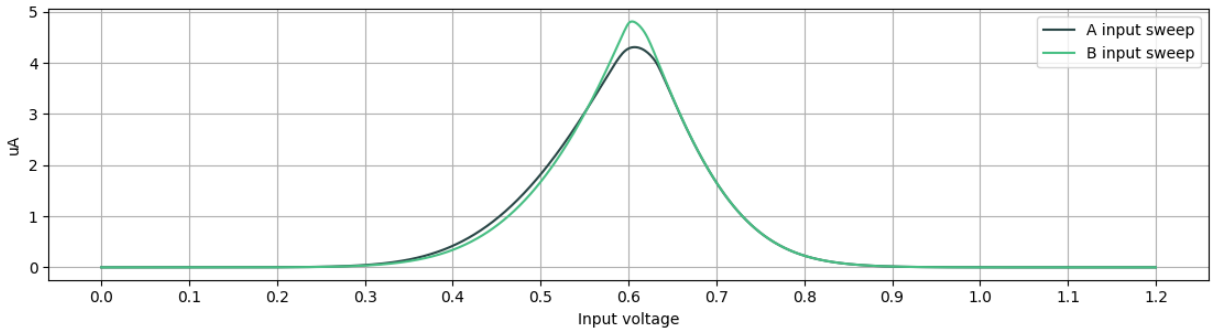

Figure 5: Transfer current of a standard CMOS NAND gate

As can be seen in fig. 5, there is a noticeable difference in the peak amplitude of the transition currents depending on which input signal causes a transition in the logic gate. The two input signals are labeled as A and B, fig. 3 shows which transistors of the NAND topology corresponds to what input signal.

The lowest peak current amplitude occurs during the sweep of the A input voltage. The results of these analyses revealed that the current in the A input sweep had a peak amplitude of 4.31 µA and that the B input sweep had a peak amplitude of 4.81 µA. When the peak amplitudes are compared, the NAND gate's current consumption is 1.117 times greater at its peak when the B input signal is swept compared to when the A input signal is swept. This difference stems from how the transistors in the logic gate are biased through the two input sweeps, mainly the two stacked NMOS transistors.

3.1.3 Internal biasing analysis

First, the simulation where input B is set to a constant voltage of V_{DD} while input signal A is swept from GND to V_{DD} is considered. Some biasing conditions of the stacked NMOS transistors through this input sweep is shown in fig. 6.

Transistor M_4 is shown to have very little change in its biasing through this analysis. This is due to the fact that input B, which is connected to the gate terminal of M_4, is statically set to V_{DD}. Given that the source terminal of M_4 is directly connected to GND, this results in an constant V_{GS} of V_{DD}. Additionally, these conditions lead to the node between the two NMOS transistors M_3 and M_4 effectively being grounded, as M_4 is in an constant on-state throughout the sweep. This is further demonstrated by the fact that M_4 has a V_{DS} of $\approx 0\,\mathrm{V}$ through the whole analysis.

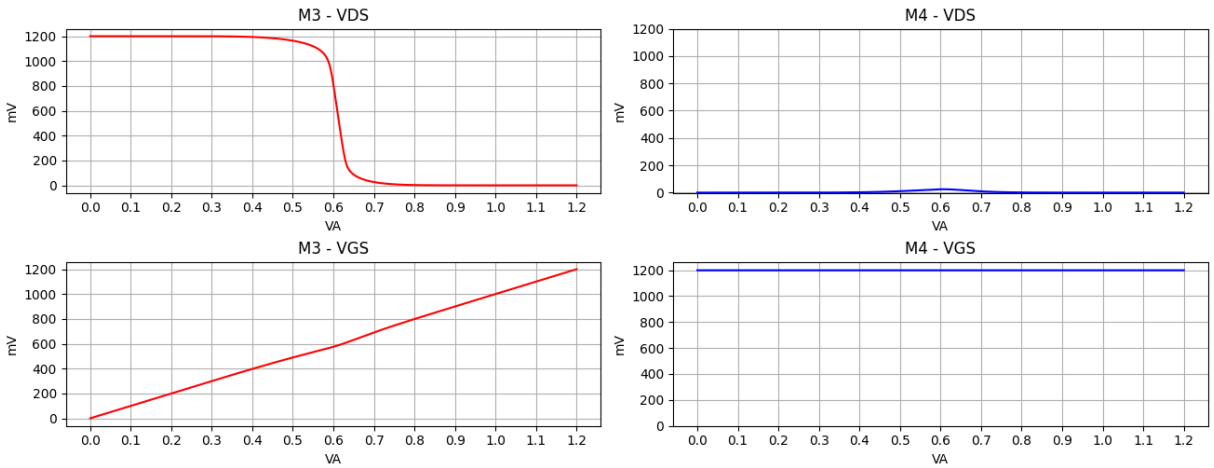

Figure 6: NMOS stack biasing through voltage sweep of VA

When the opposite simulation is performed and input B is swept while input A is kept at a constant voltage of V_{DD}, the biasing of transistor M_4 is not dormant as was the case in the prior test. In this simulation, M_3 will have a constant voltage applied to its gate terminal, while M_4 will have its gate voltage swept from GND to V_{DD}. More activity in the biasing conditions of both NMOS transistors are observed throughout the B voltage sweep compared to the A voltage sweep. Some of the biasing voltages are plotted in fig. 7.

As the V_{GS} of M_4 increases, the source of M_3 is pulled down to ground. Since the voltage applied to the gate of M_3 is constant during this sweep, this will cause the V_{GS} voltage of M_3 to increase as its source voltage decreases. Transistor M_3 will see changes to its V_{DS} voltage as well during the first part of the analysis since its source terminal is being pulled down to ground while its drain terminal is at a constant voltage potential of $\approx 1.2\,\mathrm{V}$.

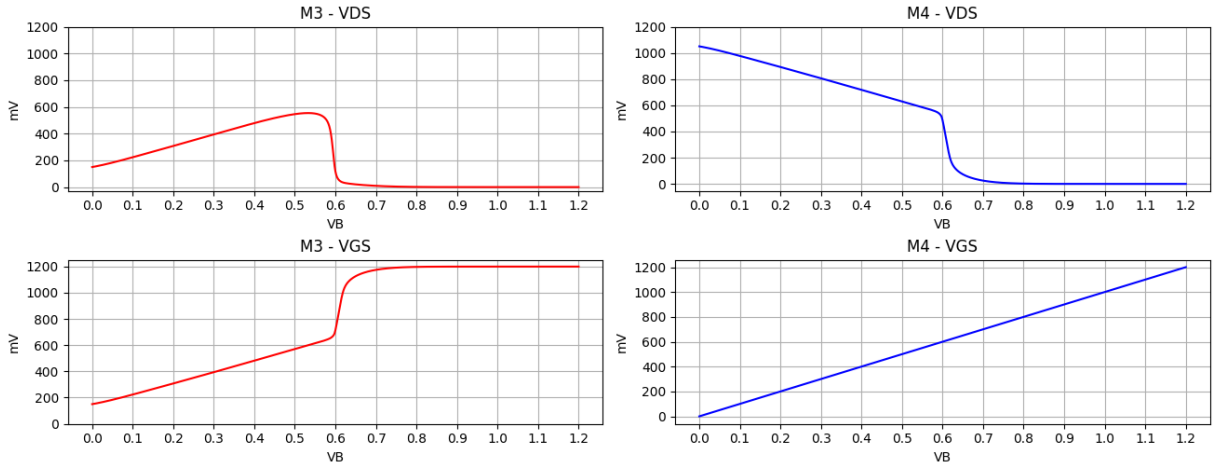

Figure 7: NMOS stack biasing through voltage sweep of VB

The difference in the peak transition current that was observed between the two voltage sweeps in fig. 5 can be attributed to the different biasing conditions experienced by the stacked NMOS transistors in each voltage sweep. Having such a clear difference in the transition current in DC simulations will translate to the dynamic current when tested in a transient simulation, as will be shown later in this thesis. Information containing the state of the input signals could therefore be leaked through analysis of the transient current waveforms as is done in for example DPA attacks[5]. The following chapter will explore how this difference in transition current can still reveal itself in other logic styles that expands on the standard CMOS logic topologies.

3.2 Exhaustive Logic Balancing

This section will present the results of a variety of tests that was conducted on a logic style in the hiding category called ELB.

This logic style works by duplicating a standard CMOS logic gate 2^n times, where n is the number of inputs to the logic gate. In the case of a two input logic gate, which will be considered for the rest of this thesis, this means constructing the top level ELB cell out of four internal standard CMOS logic gates. Each internal CMOS logic gate receives a unique combination of inverted or non-inverted input signals until all possible combinations of the top-level input signals are present within the ELB logic gate. By doing so, every input combination will be processed somewhere in the ELB logic gate regardless of the state of the top-level input signals which should result in a balanced static current consumption. Only one of the outputs from the internal logic gates will be used in subsequent circuitry, specifically the logic gate that has no inverted inputs so as to keep the logic gate's original logic function. A schematic of an ELB NAND gate is shown in fig. 8. All tests discussed in this section were conducted with ideal voltage sources as input signals for both A and B signals as well as their inverses.

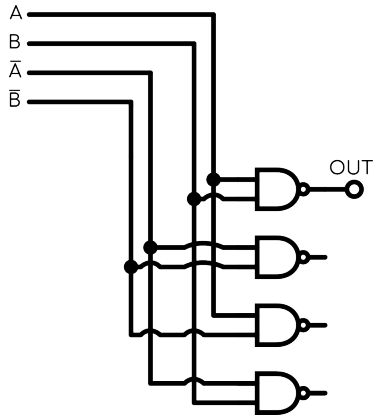

Figure 8: ELB NAND gate

3.2.1 Obtaining static currents and transition currents through DC analyses

Initial DC simulations was performed to assess the extent of input-independence in the static current of an ELB NAND gate. The simulations were carried out with the procedure described in section 2.4.1. The resulting static currents are shown in fig. 9.

The plot shows that before mismatch is introduced, an ELB NAND gate exhibits identical current consumption for all input combinations. For comparison, the static current consumption of a standard CMOS NAND gate is also presented in the plot. Furthermore, it is observed that the summation of all the currents from the standard CMOS gate is equivalent to the current consumption of the ELB gate at any input combination. This is a natural consequence, considering that the working mechanism of an ELB NAND gate is based on four inner standard CMOS NAND gates each processing a unique input combination.

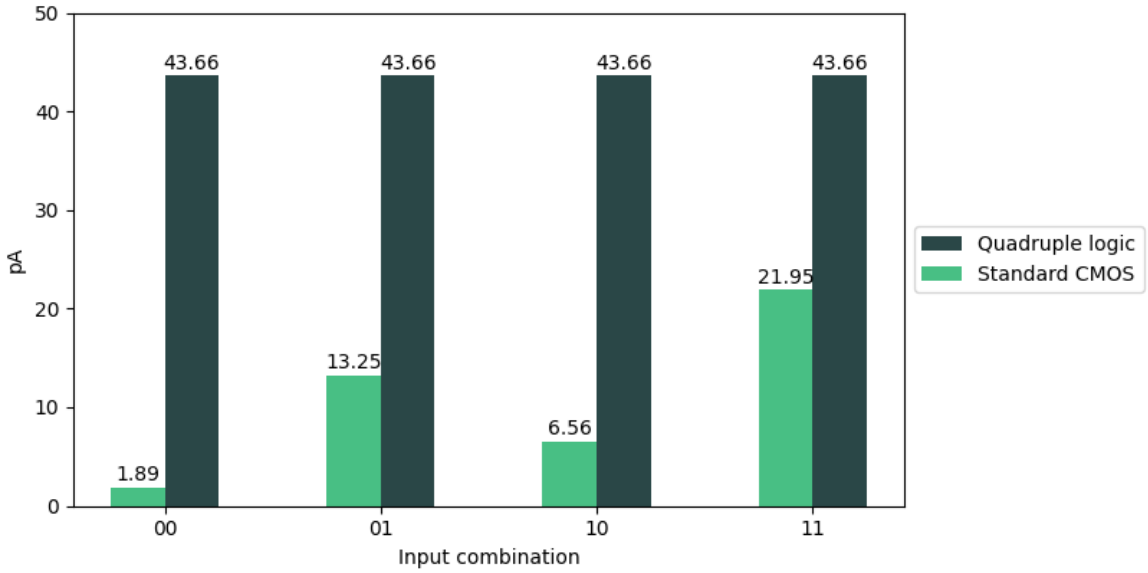

Figure 9: Static current consumption of an ELB NAND gate compared to a standard CMOS NAND gate

Although ELB logic effectively balances the static current consumption, it fails to address the imbalance in current consumption during the transition of an input signal. As demonstrated in section 3.1 for standard CMOS logic, sweeping the two input signals separately then comparing the resulting current consumption curves reveals a disparity in peak current consumption. The same difference in transition current is seen in ELB logic.

This difference will first be displayed through a DC analysis following the procedure explained in section 2.4.2, with one simulation for each input. The resulting current consumption curves are shown in fig. 10. The results of these tests shows a clear difference in current consumption depending on which input pin causes the logic gate to transition. However, there was no noticeable difference in current consumption based on the direction of the voltage sweep applied to the input signal. Therefore, the current consumption will be consistent for a given pin, regardless of whether the input signal undergoes a transition from high to low or from low to high.

The test revealed that the A input sweep had the lowest peak current consumption, recorded at 8.56 µA. Meanwhile, the B input sweep had a peak current consumption of 9.54 µA, resulting in a ratio of 1.11 between the two. This ratio is quite similar to the one discovered between the current consumption curves for the standard CMOS logic, as presented in section 3.1.

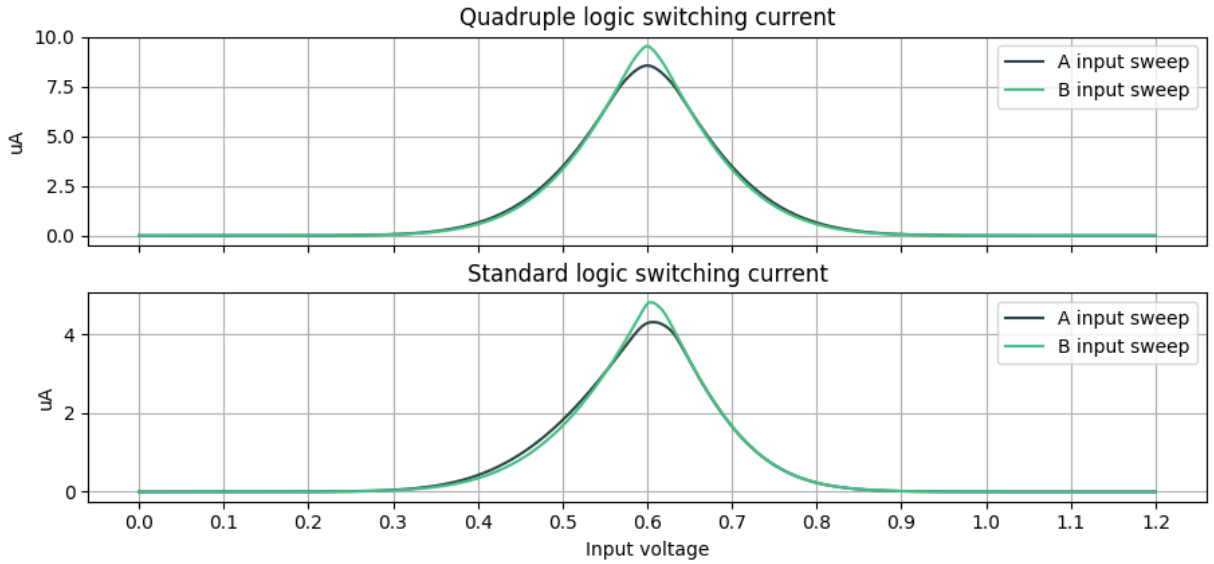

Figure 10: DC switching current consumption of an ELB NAND gate and a standard CMOS NAND gate

The discrepancy in transition currents can be attributed to the internal CMOS NAND gates within the ELB logic. The total current consumption of an ELB NAND gate is the sum of the current drawn by its four internal NAND gates. As the main A and B inputs of the ELB NAND gate is only connected to the corresponding A and B inputs of its internal CMOS NAND gates, the characteristics of the ELB logic gate are reflected in its internal gates. As outlined in section 3.1, a standard CMOS NAND gate exhibits a higher peak switching current during a sweep of input voltage on pin B. Therefore, a higher sum of internal currents is obtained when sweeping the input voltage on pin B of the ELB NAND gate than when sweeping the voltage on input A.

The figure presented in fig. 11 illustrates the transition current of the internal CMOS NAND gates obtained by a single-input sweep of an ELB NAND gate. The plot depicts the internal switching current for both an A input voltage sweep and a B input voltage sweep. The labels above each subplot contains a number for each internal NAND gate where $NAND1$ refers to the upper most NAND gate, and the following numbers refers to the internal NAND gates in descending order, in reference to fig. 8. It is evident from the plot that the internal currents during a B input voltage sweep exhibit a higher peak amplitude as compared to those during an A input sweep.

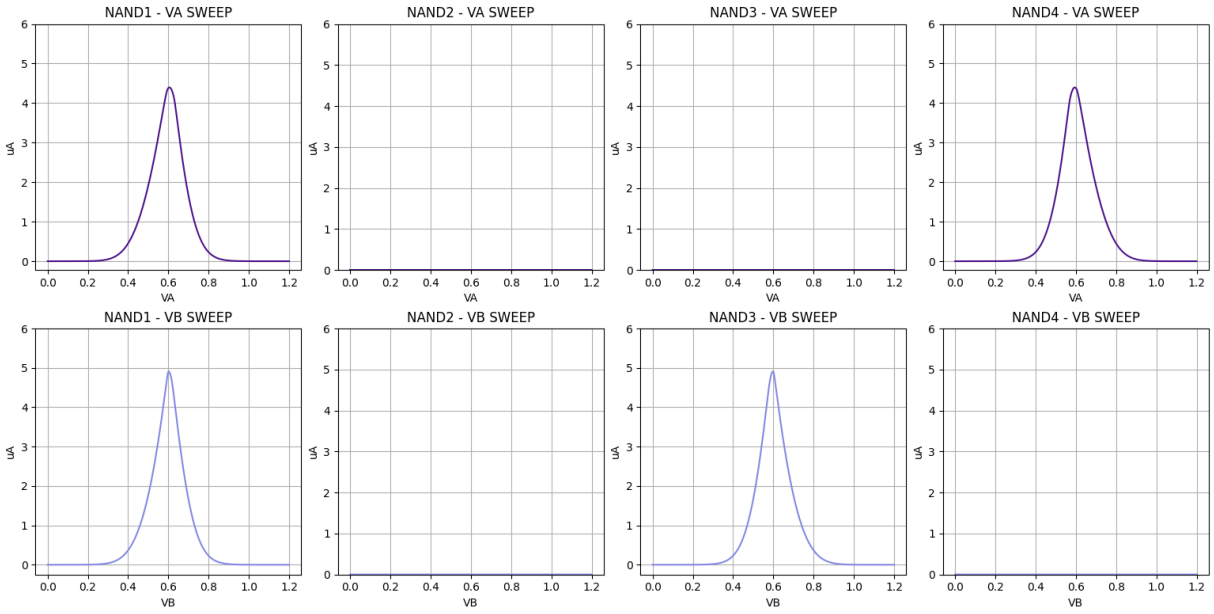

Figure 11: DC switching current consumed by the internal NAND gates of an ELB NAND gate

3.2.2 Analyses of transition currents in the time domain

Even though ELB logic was presented as a method of balancing a logic gate's static current consumption, some analyses were conducted to evaluate this logic style's performance in terms of dynamic current as well. To balance the dynamic current using the same methodology as the static current balancing in ELB logic, each transition of the input signals should result in identical dynamic current waves. This requires each internal logic gate to drive an equally sized load.

All possible transitions of the input signals were then tested on a ELB NAND gate, where all internal NAND gates were driving identical loads. The results of this test revealed three distinct dynamic current curves, which are depicted in fig. 12.

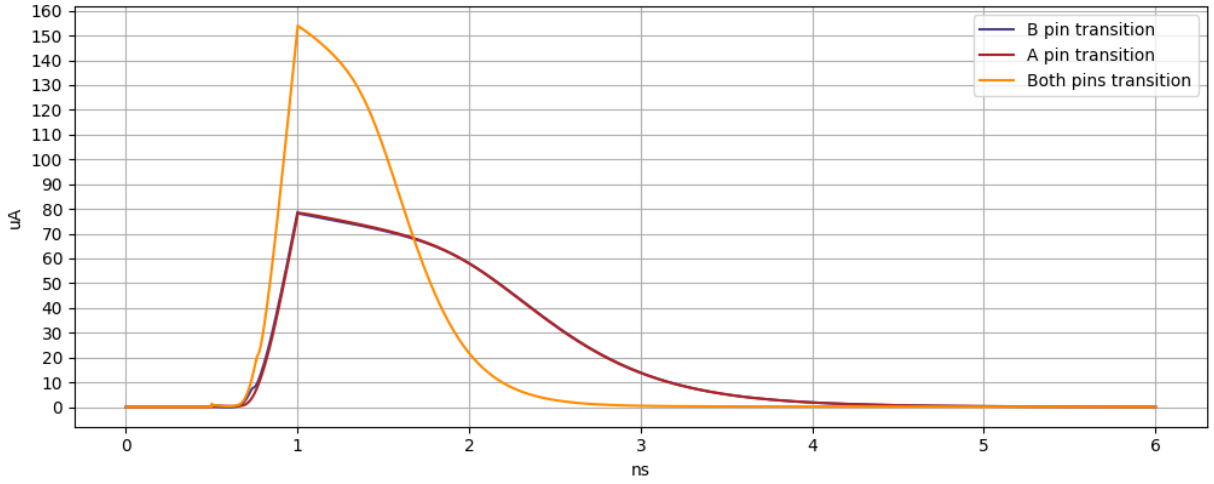

Figure 12: The three possible current traces of an ELB NAND gate with equal loads driven by the internal NAND gates

When any transition of the input signals occurs, one of the three possible dynamic current waves will be consumed by the ELB NAND gate. The three dynamic current traces correspond to the changes in the A input, B input, or both inputs simultaneously. The most notable difference

in the current waveforms is observed between the current waveform when both inputs change simultaneously and either one of the single-input transition current curves. Nevertheless, there is still a slight variation between the two single-input current waveforms. For a change on a given input pin, the current waveform will remain the same if the input value changes from high to low or vice versa. While there is no difference in peak amplitude between the single-input transition currents, there is a dissimilarity in the early stages of the transition. Figure 13 shows a plot that exhibits these two current traces in more detail, as well as a plot that displays the difference between the two current traces.

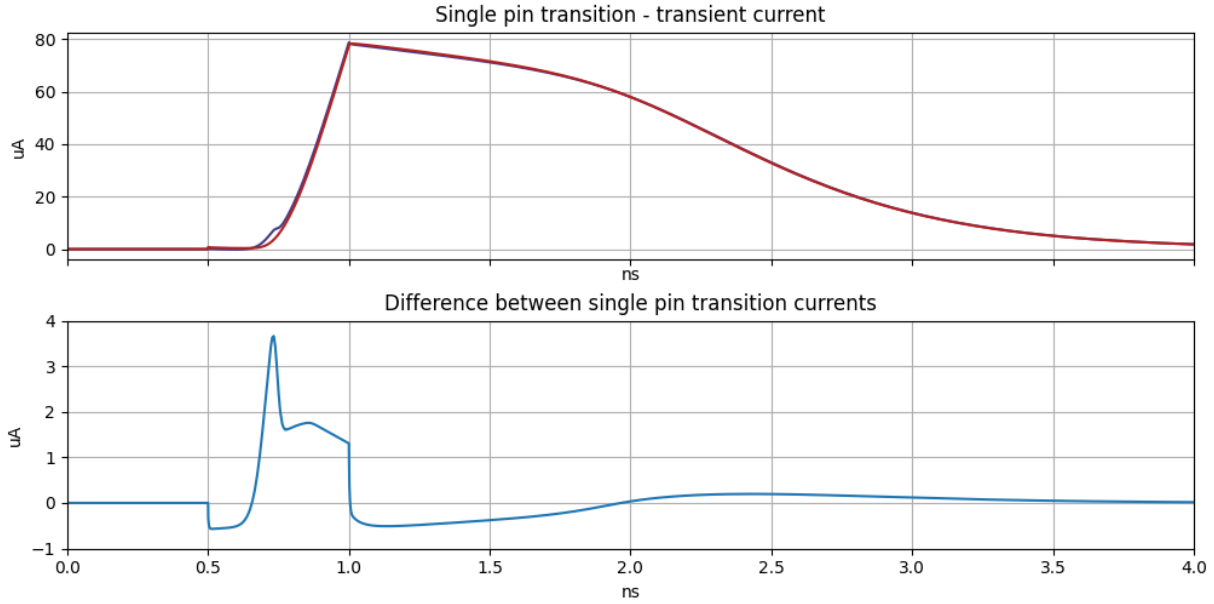

Figure 13: The two different single-input transitions and the difference between them

In fig. 13, the maximum difference between the two single-input transition currents is 3.66 µA, occurring at the 0.73 ns mark of the transient analysis. At this point, the A-pin transition current is 3.59 µA and the B-pin transition current is 7.25 µA, showing that the B-pin current is 2.01 times greater than the A-pin current at the time of the highest difference.

3.2.3 The impact of temperature on the static current

The static current consumption of a circuit can be significantly affected by the temperature at which the circuit. To ensure that the balance achieved by the ELB logic style is not compromised when the circuit temperature is manipulated, simulations were conducted with varying temperature parameters. In this section, the results of a DC analysis conducted on an ELB at temperatures of $-40\,°C$, $27\,°C$, and $80\,°C$ will be presented to observe how the static current changes. This analysis only considers variations in temperature and does not take into account mismatch or other non-ideal effects.

The results of this analysis showed that temperature alone is not able to disturb the balance achieved by ELB. Although the absolute value of the current was significantly impacted by temperature changes, there was no observable data-dependency in the logic gate's static current within a given temperature. A plot showing the static current consumption at each input for all three temperatures is depicted in fig. 14.

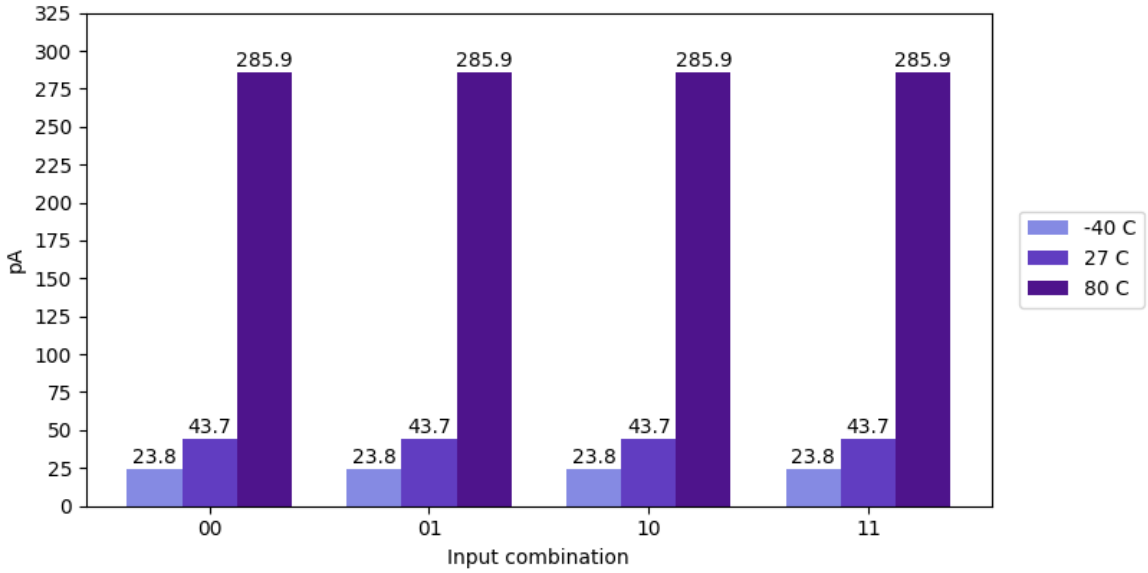

Figure 14: Static current from a ELB NAND gate for three different temperatures

3.2.4 Imbalances in the static current due to intra-die mismatch

In the previous sections, it was demonstrated that ELB logic demonstrates robust input-independence in its static current consumption in ideal simulations. However, an aspect that has not been taken into account so far is the impact of intra-die mismatch. This section will present the results of a DC analysis that was carried out with mismatch introduced into the circuit, and how it influenced the current consumption of the logic style.

To simulate the effects of mismatch, a Monte Carlo simulation was conducted. To get an idea of the level of variation that can be expected in the static current after mismatch is introduced, the Monte Carlo simulation was conducted with 1000 iterations. The average current and standard deviation in the static current at each input combination for all 1000 iterations were calculated and are plotted in fig. 15.

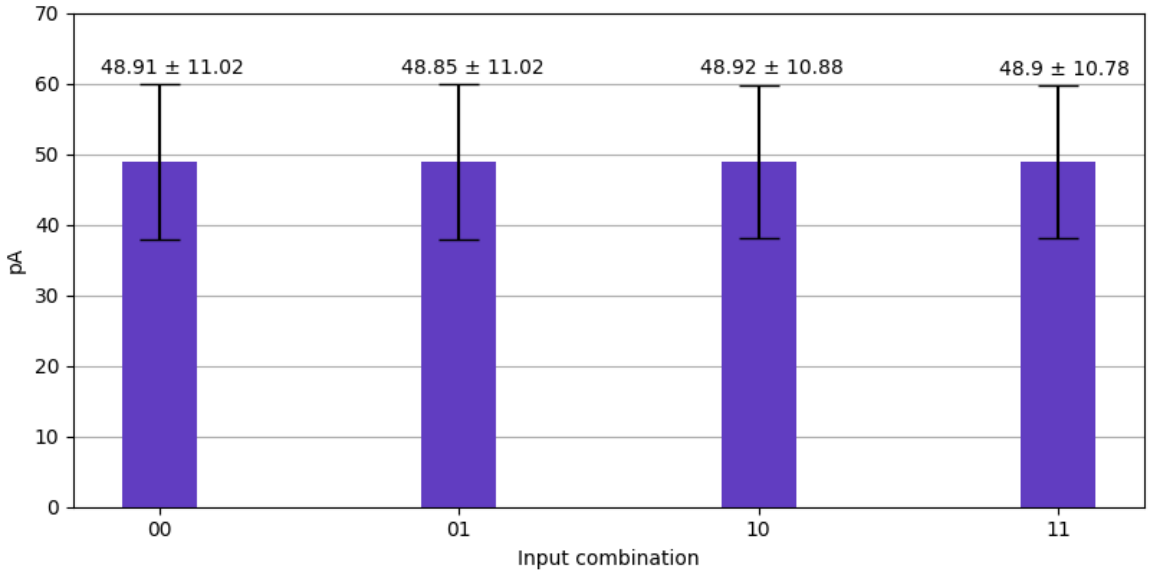

Figure 15: Average static current from a ELB NAND gate from the Monte Carlo simulation

The underlying balancing scheme of the logic style is apparent in the data when the average

currents are considered. The plot demonstrates that the static current and the standard deviation found in this simulation are similar in magnitude across all input combinations. Over a large number of iterations, the average static current of each input combination converges to the same value, with variations due to mismatch occurring around this shared mean current.

However, studying the averaged current consumption of 1000 cases of mismatch is not a realistic representation of a single circuit's behavior in a real-world scenario. For this reason, another Monte Carlo simulation was performed with only 10 iterations. The number of iterations was significantly reduced because the aim of this simulation was not to obtain a statistical distribution of the currents, but rather to provide some examples of instances of mismatch. One of the resulting Monte Carlo iterations was selected and analyzed further to represent one instance of mismatch in the logic gate and to illustrate the extent to which input-dependency could be introduced through mismatch. The static currents from the selected Monte Carlo run were plotted along with those from an ideal simulation, and the results are presented in fig. 16.

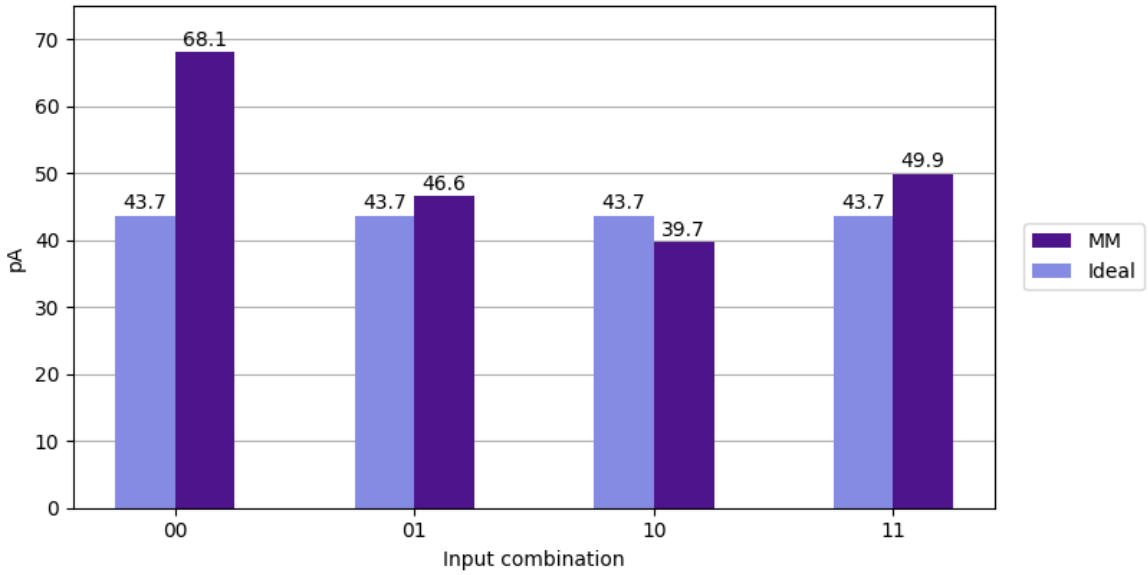

Figure 16: Example static current from a ELB NAND gate from one of the Monte Carlo runs

The introduction of mismatch leads to an imbalance in the static current between input combinations, and the obtained set of currents has an interval like the one shown in eq. (2). A standard deviation of $\sigma = 10.49\,\text{pA}$ is observed in the set of currents, yielding a RSD of 20.54 % relative to the mean value of $\overline{I}_{static} = 51.06\,\text{pA}$.

$$39.71\,\text{pA} \leq I_{static} \leq 68.08\,\text{pA} \tag{2}$$

This particular iteration from the Monte Carlo simulation was chosen for further analysis because it clearly illustrates that an input-dependency can be found in the logic gates static current after mismatch is introduced. But this iteration did not demonstrate a uniquely large input-dependency when compared to all other iterations although varying degrees of variation was seen in the sets of currents.

To get a better overview of the impact that intra-die mismatch can have on any given instance of a logic gate, another simulation was conducted where the goal was to gather averaged metrics about the variations in current consumption that is seen within a set of currents for any given case of mismatch. Whereas the plot in fig. 15 shows the average degree of change for each input combination across the whole Monte Carlo simulation, it does not consider the degree of change within each set of currents. To gather these metrics the test bench described in fig. 2 was used,

and the average current and standard deviation were calculated for the set in every iteration. Then the average of these intra-set metrics was calculated from all 1000 iterations of the Monte Carlo simulation.

This resulted in a average standard deviation of $\sigma = 7.99\,\mathrm{pA}$ and the average mean value for all runs was $\overline{I}_{static} = 48.89\,\mathrm{pA}$. The resulting RSD is 16.34 % indicating that some amount of input-dependency can be expected when the effects of mismatch are taken into account. The highest and lowest static currents from all iterations were averaged out to get the average interval as can be seen in eq. (3).

$$31.92\,\mathrm{pA} \leq I_{static} \leq 140.02\,\mathrm{pA} \tag{3}$$

A summary of the mismatch analyses is provided in table 1.

Simulation	Mean current	Standard deviation	RSD
Single instance	51.06 pA	10.49 pA	20.54 %
Average	48.89 pA	7.99 pA	16.34 %

Table 1: ELB mismatch summary

3.3 Octuple logic

As was explored in section 3.2, there are unaddressed imbalances in the current consumption of ELB logic. Specifically, there is a difference in current consumption through a single-input transition of the input signals depending on which of the input signals changes states. There are multiple ways of addressing this issue, one of which is introduced in this thesis as Octuple Logic. This approach is based on duplicating an ELB logic gate and inverting the order of the input pins of the second ELB gate as shown in fig. 17 which depicts an octuple logic NAND gate.

This approach requires eight internal standard CMOS logic gates as opposed to the four that is used by an ELB logic gate. The first column of NAND gates are connected in the same way as a regular ELB NAND gate while the second column is connected with the order of the A and B signals inverted. The remainder of this section will present the results of taking an octuple style NAND gate through the same tests that was done for ELB logic in section 3.2.

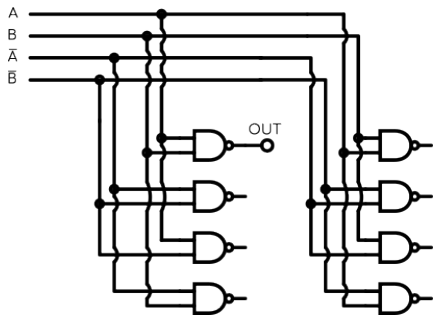

Figure 17: Octuple logic NAND gate

3.3.1 Obtaining static currents and transition currents through DC analyses

Initially a few DC analyses were conducted on this logic style to determine its current consumption characteristics. Ideal voltage sources were used for both the A and B input signals and their inverses and the static current consumption of the logic gate was recorded for every input combination, as described in section 2.4.1. As depicted in fig. 18, this style exhibits the same input-independent static current as the ELB logic does in ideal simulations. However, the octuple logic style's circuitry is twice that of an ELB gate, resulting in twice the static current consumption.

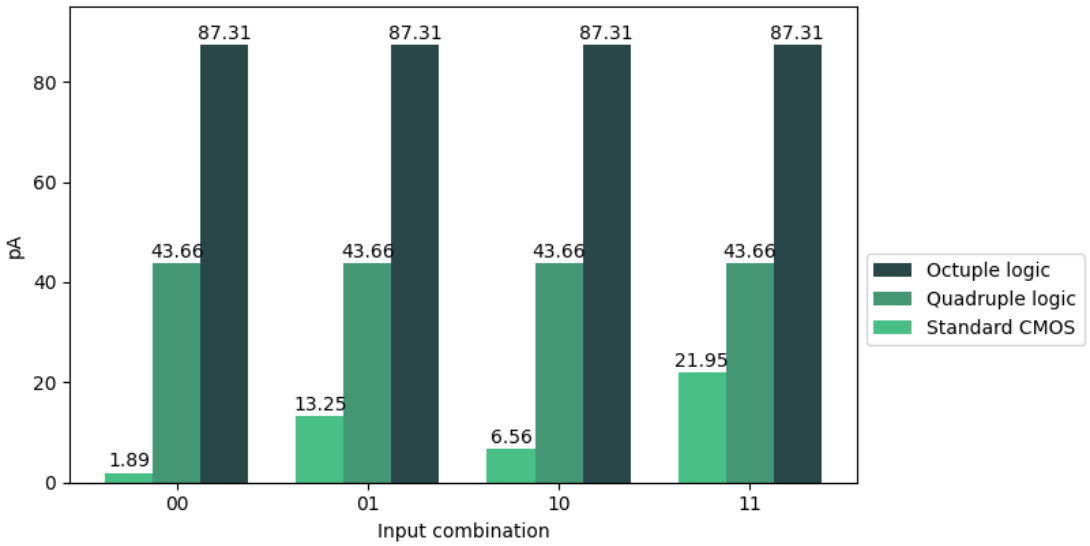

Figure 18: Static current consumption of octuple logic, quadruple logic and standard CMOS logic

In contrast to ELB logic however, further DC analysis of the octuple logic style reveals that any discrepancy in transition current during a single-input transition, regardless of which input pin initiates the transition, is effectively eliminated. This is shown in fig. 19 where the two current traces for the octuple logic are completely overlapping. These currents were obtained through the procedure described in section 2.4.2.

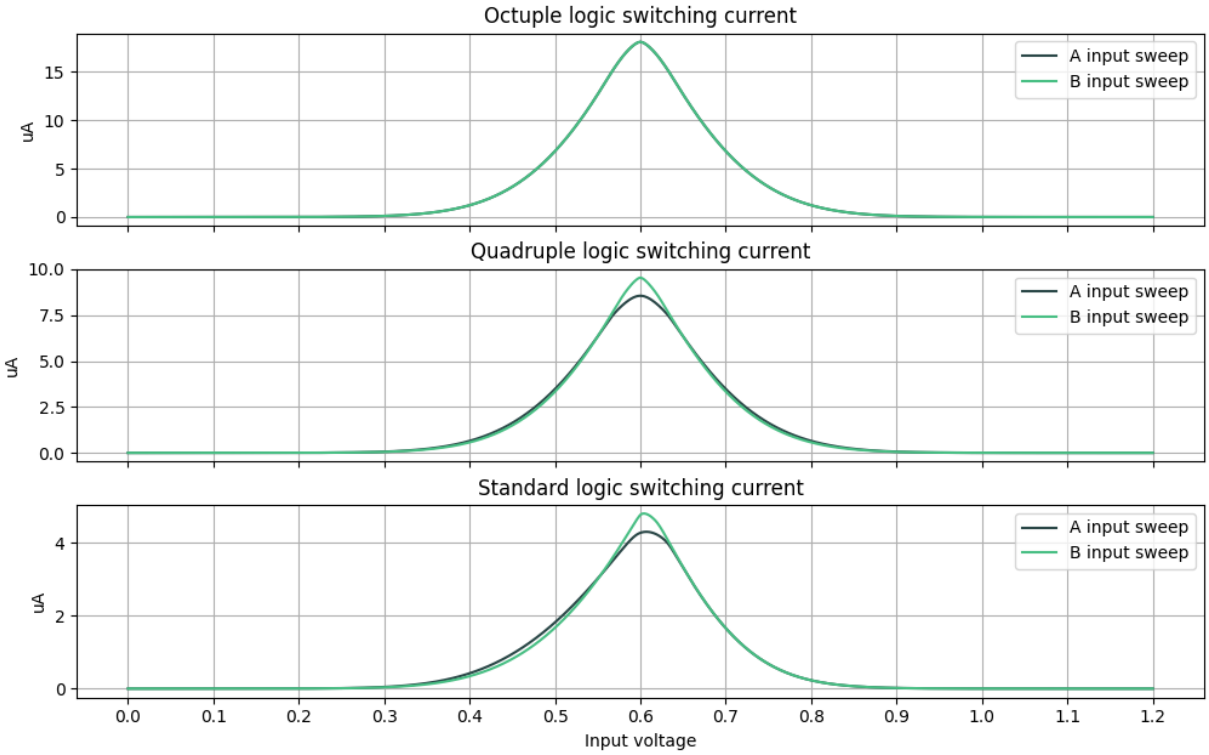

Figure 19: DC transition current for octuple logic, quadruple logic and standard CMOS logic

3.3.2 Analyses of transition currents in the time domain

In the transient analysis of ELB logic conducted in section 3.2.2, it was necessary to place identical loads on all four internal NAND gates to achieve uniform current consumption waves. One might assume that the same methodology applies to the octuple logic style and that eight identical loads would be required for such a logic gate. However, this is not the case for a NAND gate. Since the output of a NAND gate only transitions to a zero when both input pins are one, the second column of NAND gates, with inverted inputs, is functionally equivalent to the first column. Thus, the octuple NAND gate can be seen as having four pairs of internal NAND gates, each with the same combination of inverted and non-inverted inputs, where the two NAND gates in each pair respond identically to any input combination. As a result, the outputs of the two NAND gates in each pair can be connected and drive the same load, requiring only four equally sized loads for the octuple NAND gate. Figure 20 shows the internal topology of an octuple NAND gate, with all inputs

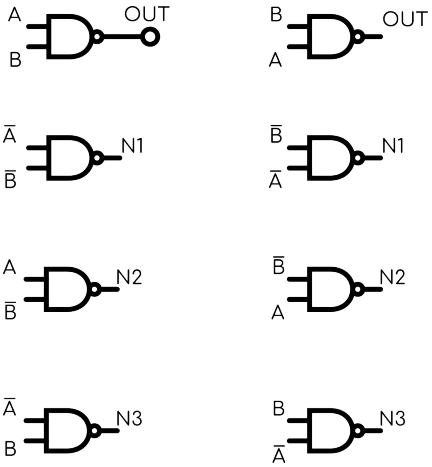

Figure 20: Octuple logic NAND gate with node names

labeled, and node names indicating which outputs are connected.

After connecting identical loads to each of the four NAND-pair outputs as explained above, transient analyses were conducted. Initially, all possible transitions were tested, revealing that the octuple NAND gate only has two distinct current consumption curves compared to the three found for an ELB gate in fig. 12. The reason for this difference is that there is practically no variation between the two single-input current consumption curves. The plot in fig. 21 displays the current consumption from both single-input transitions triggered by the A and B pins overlayed, with a separate plot showing the difference between them. It is worth noting that the difference plot is in units of attoampere on its y-axis.

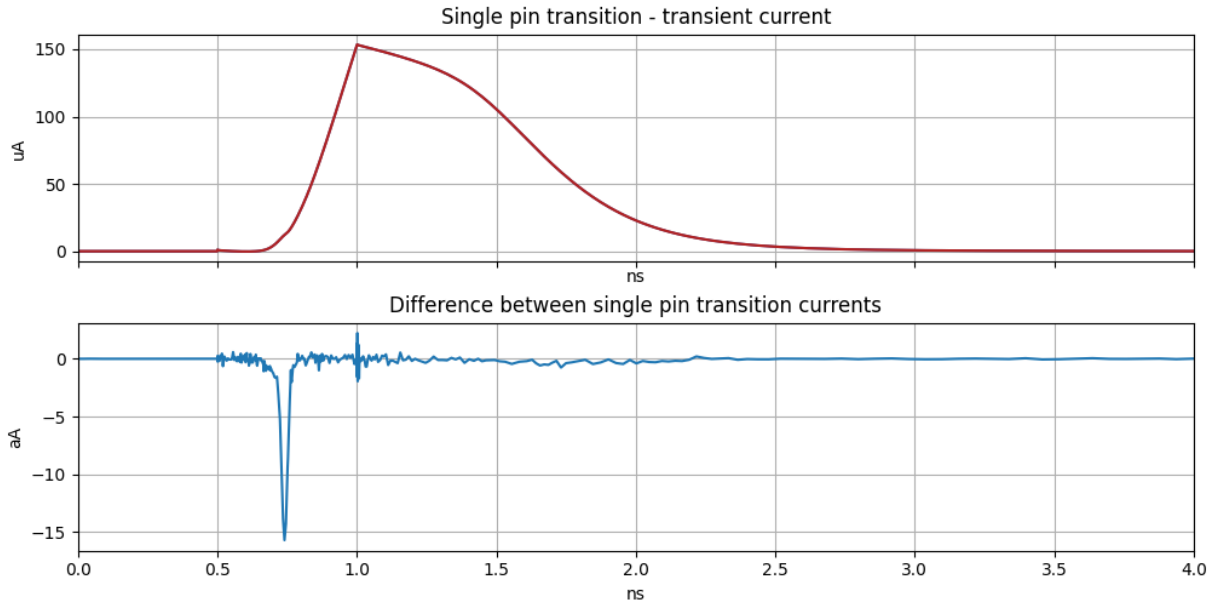

Figure 21: The two single-input transition currents and the difference between them seen in an octuple logic NAND gate

However, the dynamic current consumption of the octuple logic style is not completely balanced. Although the current consumed through a transition triggered by a single input is identical regardless of whether it was caused by the A or B pin, one imbalance remains. There is still an imbalance between a single-input transition and a transition caused by both input pins simultaneously, as was also observed in the ELB logic style, depicted in fig. 12. The two different dynamic current consumption curves that can occur in an equally loaded octuple NAND gate are displayed in fig. 22. One of the current-waveforms in the plot will emerge as long as there is a transition of the input signals, regardless of whether the inputs change from high to low or low to high. The single factor determining which of these two curves will occur is whether both input values change or only one of them does.

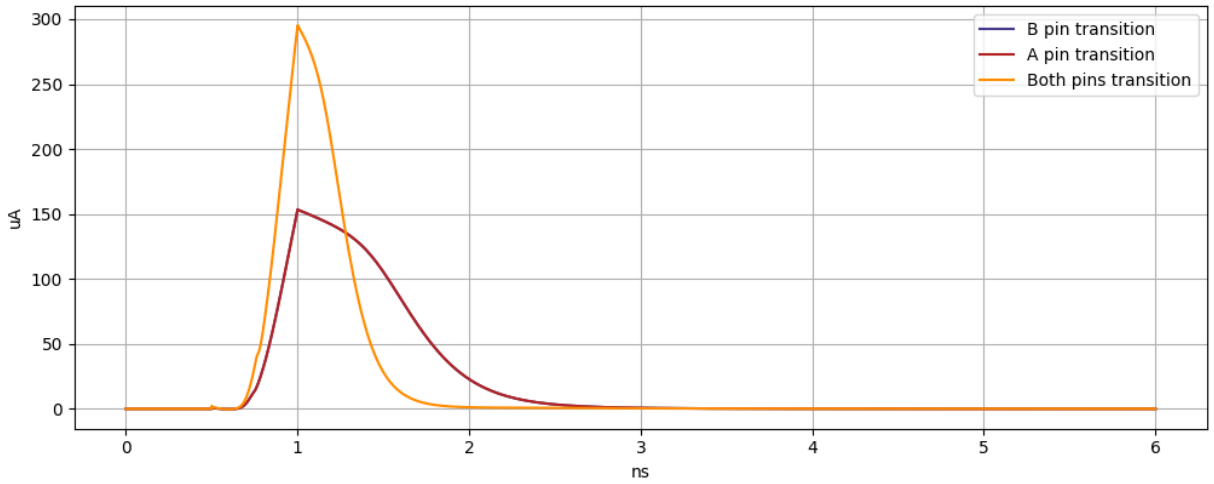

Figure 22: The two possible current consumption curves seen in an ideal octuple NAND gate

3.3.3 The impact of temperature on the static current

Similar to the analysis conducted for the ELB logic style in section 3.2.3, an assessment was carried out to examine the effects of temperature on the octuple logic style. The same temperature parameters were used for this analysis, namely −40 °C, 27 °C, and 80 °C. The results of testing an octuple NAND across the three different temperatures were found to be similar to those observed for the ELB logic style in section 3.2.3. Specifically, temperature can significantly impact the absolute value of the static current. However, no input-dependency is introduced to the static current when only the temperature is changed.

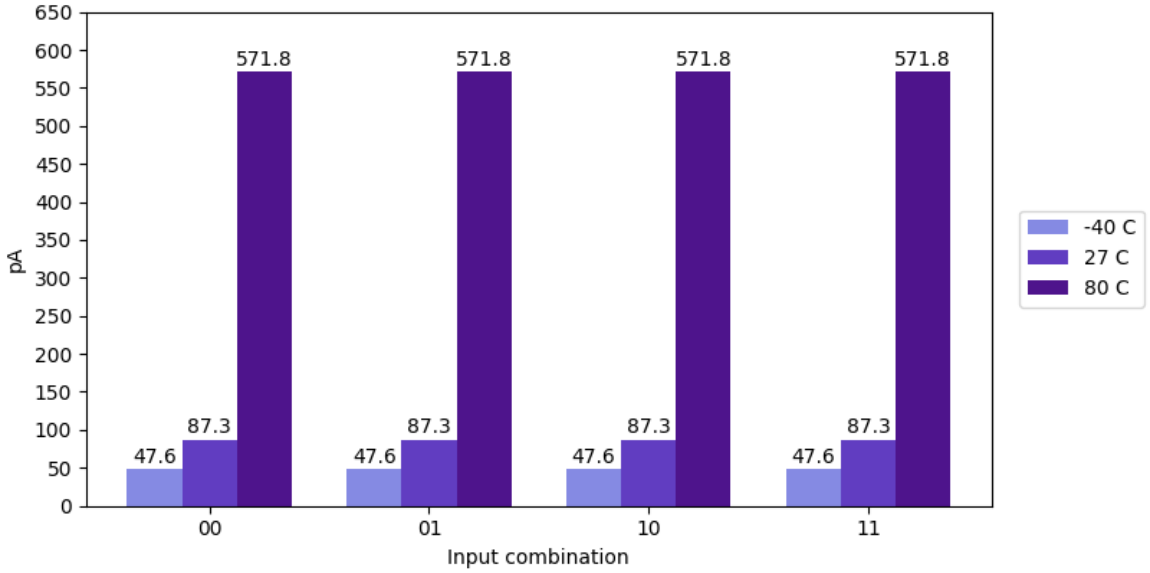

Figure 23: Static current from a octuple NAND gate in three different temperatures

3.3.4 Imbalances in the static current due to intra-die mismatch

The previous sections demonstrated that the octuple logic style achieves input-independence in its static current in ideal simulations and in ideal simulations where temperature is taken into account. Nevertheless, as was seen for the ELB logic style in section 3.2.4, the octuple logic style also loses its input-independence when mismatch is introduced.

The same approach used to analyze the ELB logic in section 3.2.4 was followed to introduce mismatch into the simulations for the octuple logic style. Monte Carlo simulations were used to gather statistical parameters, with 1000 iterations conducted initially, followed by a smaller simulation with 10 iterations to illustrate single instances of mismatch. The average current and standard deviation for all input combinations from the larger simulation is plotted in fig. 24.

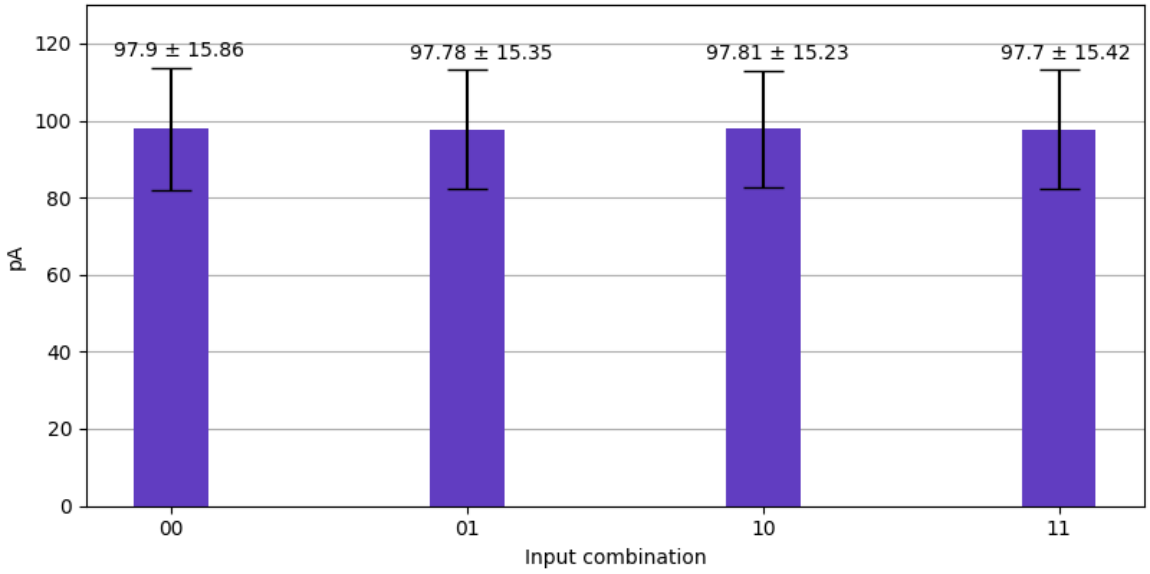

Figure 24: Static current average from Monte Carlo simulation

The octuple logic style demonstrates similar characteristics to those observed in the ELB logic, where the mean current for each input combination has similar amplitude and the standard deviation has comparable size for all input combinations. These results suggest that the fundamental logic style provides a balanced mean value around which variations due to mismatch occur.

A Monte Carlo simulation with 10 iterations was conducted next to be used as individual examples of mismatch, and one of the resulting current sets was picked out for further analysis. That current set is plotted together with the ideal currents in fig. 25.

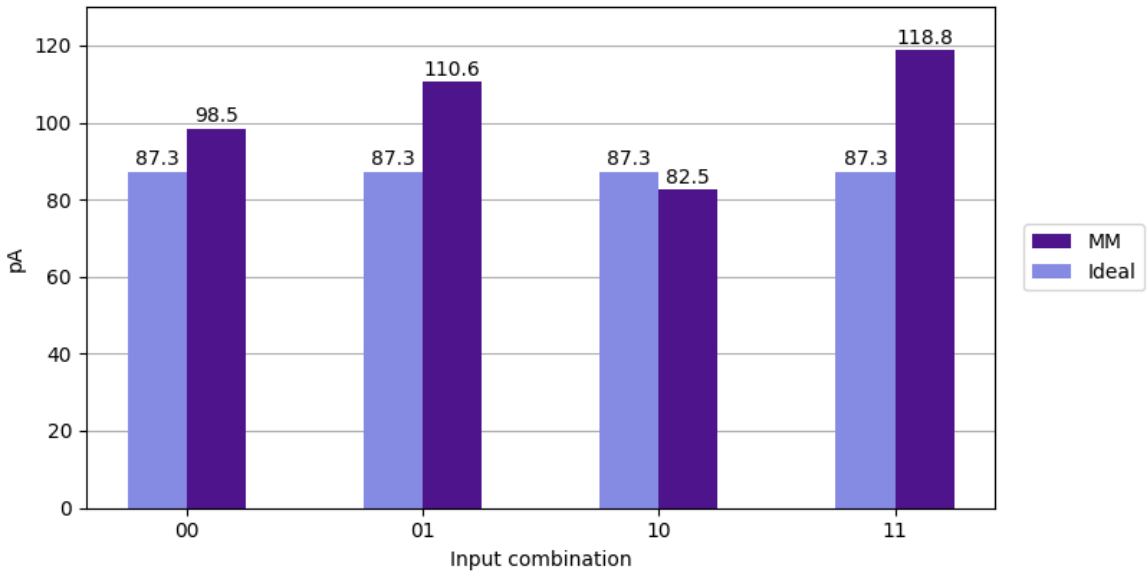

Figure 25: Static current from example Monte Carlo parameter set and ideal simulation

When looking at one instance of mismatch like the one in fig. 25 it is evident that mismatch causes an input-dependency to occur in the logic style. This set of currents has an interval like the one described in eq. (4). The standard deviation of the set is $\sigma = 13.66\,\text{pA}$ and the mean value is $\overline{I}_{static} = 102.58\,\text{pA}$. The resulting RSD is 13.32 %.

$$82.48\,\text{pA} \leq I_{static} \leq 118.78\,\text{pA} \tag{4}$$

To demonstrate that the single instance of mismatch shown in fig. 25 is not an outlier, the intra-set Monte Carlo analysis described in section 2.4.3 was conducted to find averages for the same metrics that was presented for the single iteration in the previous section.

A Monte Carlo simulation with 1000 iterations was conducted, and the standard deviation and average current were calculated for the set of currents at every iteration. The resulting average standard deviation for all of the 1000 sets of static currents was $\sigma = 11.79\,\text{pA}$, and the average mean value found for all the sets was $\overline{I}_{static} = 97.81\,\text{pA}$. This gives an average RSD of 12.05 %. The average interval of all the current sets is of the size described in eq. (5).

$$69.55\,\text{pA} \leq I_{static} \leq 205.58\,\text{pA} \tag{5}$$

A summary of the mismatch analyses is provided in table 2.

Simulation	Mean current	Standard deviation	RSD
Single instance	102.58 pA	13.66 pA	13.32 %
Average	97.81 pA	11.79 pA	12.05 %

Table 2: Octuple logic mismatch summary

3.4 Dual-pulldown logic

A NAND gate topology presented in [17], dubbed *perfectly symmetric 2-input NAND gate* by the authors of [17], is seen in fig. 26. This logic gate topology is intended to serve as the foundation for a new logic style that will be introduced later in this thesis and will be referred to as dual-pulldown logic from this point onward. A few tests were conducted to establish a baseline performance of a single dual-pulldown NAND gate. The results of these baseline-tests will be presented in this section.

An alternative approach to achieving the benefits of an octuple NAND gate in terms of balancing the transition current, as demonstrated in section 3.3, without requiring eight internal logic gates, is by utilizing non-standard topologies for the internal logic gates. One such alternative topology duplicates either the pull-down or pull-up network, depending on the logic gate's function. In the case of a NAND gate, the pulldown branch is duplicated, and the second pulldown branch has its inputs flipped, as depicted in fig. 26. This approach effectively addresses the biasing issue of the stacked NMOS transistors identified in standard CMOS logic, as discussed in section 3.1, by simultaneously presenting both biasing conditions for any single-input transition, regardless of whether it is triggered by the A or B input signal. This balance in biasing conditions results in equal single-input transition currents regardless of the input signal that triggers the transition.

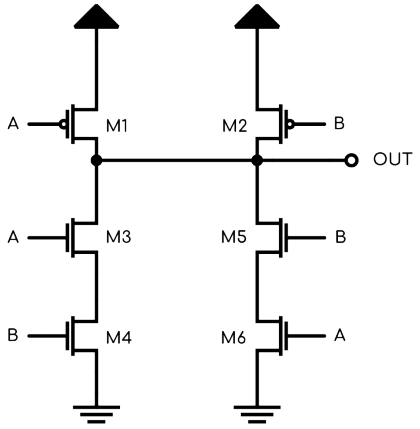

Figure 26: Dual pulldown NAND gate

3.4.1 Obtaining static currents and transition currents through DC analyses

To begin with, the static current corresponding to each input combination was recorded. The results revealed large variations between the static currents recorded for each of the four possible input combinations for this particular logic style, as depicted in fig. 27. The highest static current recorded was 33.65 pA, which is 9.64 times larger than the lowest static current value of 3.49 pA. It was also observed that the static current remained constant at 18.61 pA when only one of the inputs was active, regardless of which input was active. The set of currents had an interval as described in eq. (6), with a mean value of $\overline{I}_{static} = 18.59\,\text{pA}$. The standard deviation of the set was calculated to be $\sigma = 10.66\,\text{pA}$.

$$3.49\,\text{pA} \leq I_{static} \leq 33.65\,\text{pA} \tag{6}$$

When compared to the metrics that was found for standard CMOS logic in section 3.1, dual-pulldown logic has larger values for both standard deviation and a higher mean value. However the RSD is slightly lower at 57.36 % compared to the 69.13 % that was found for standard CMOS.

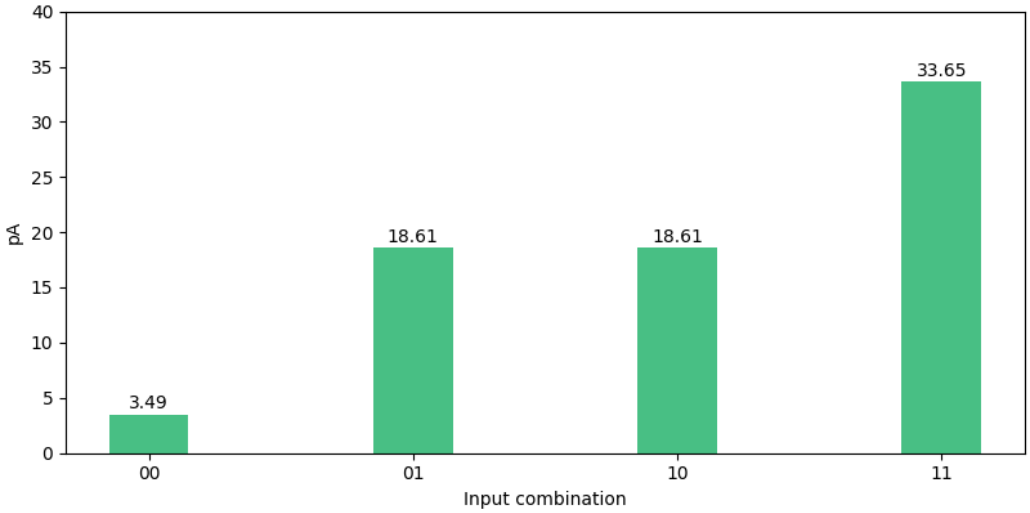

Figure 27: Static current consumption based on input combination for a single dual-pulldown NAND gate

Next, the DC transition current was simulated and it was observed that this logic style demonstrates equal single-input transition currents. This is shown by the overlapping current traces throughout the sweep in fig. 28. The tests resulted in a peak switching current of 6.6 µA.

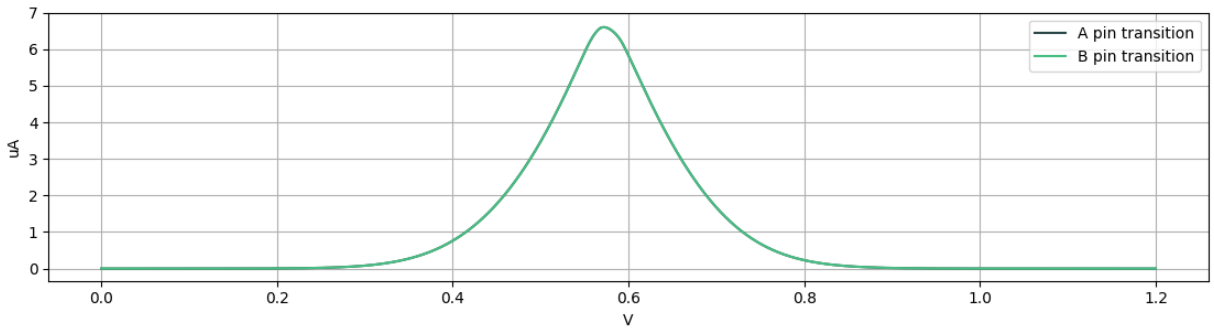

Figure 28: DC transition current for a single dual-pulldown NAND gate

3.4.2 Analyses of transition currents in the time domain

Transient analyses were conducted to explore if the dual-pulldown NAND gate would continue to demonstrate equal single-input transition currents in the time domain. Results from this analysis showed that the equality between the single-input transition currents persisted, as seen by the overlap between the current-traces in fig. 29. The upper plot in fig. 29 displays the single-input transition currents while the lower plot shows the difference between them. In this ideal analysis, the curves differ by a maximum of only 1 aA. Notably, the current traces in fig. 29 occur only when the output of the NAND gate transitions from 0 to 1, as this is the only transition that charges the single output load for this logic style. Therefore, a single-input transition current, like the ones seen in the plot, only takes place when the input signals change from '11' to either '10' or '01'. As opposed to the quadruple or octuple logic styles described in section 3.2 and section 3.3, where one of the four loads would be charged regardless of the changes made to the input signals.

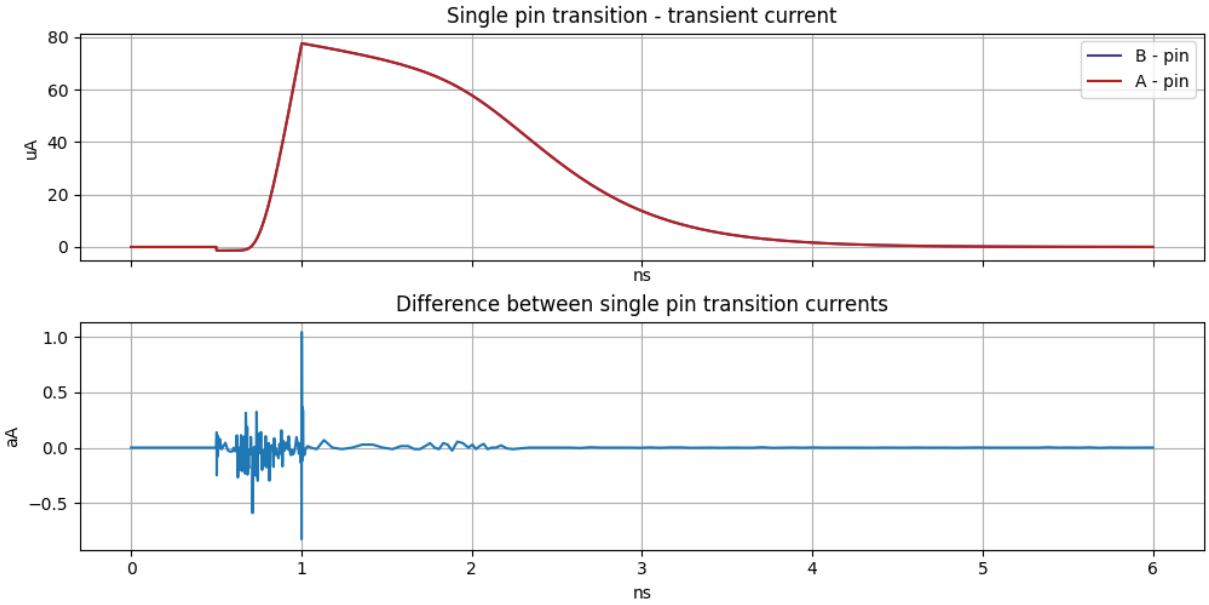

Figure 29: Transient single-input currents and difference for a single dual-pulldown NAND gate

Still only looking at the dynamic current curves that occur when a load is charged, there is a second possible current curve in addition to the single-input transition curves. That curve is, as was the case for the previous logic styles, when both input signals change at the same time. In the case of a single dual pulldown NAND gate such as the one that was tested here, that means that the inputs have to go from '11' to '00' for this second current curve to occur. The two single-input transition currents together with the double-pin transition currents are plotted together in fig. 30.

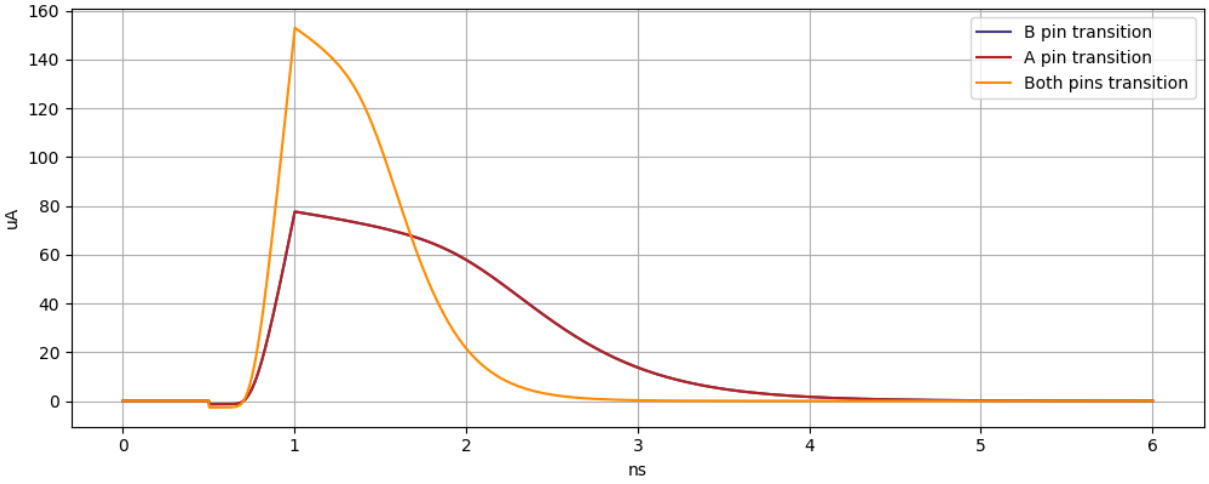

Figure 30: Transient single-input and double-pin currents for a single dual-pulldown NAND gate

3.4.3 The impact of temperature on the static current

Even though this logic style alone was not expected to achieve input-independence in its static current by itself, analysis was done to explore what impact could be expected from a varying temperature on the static current consumed by the logic style. As was done for previous logic styles, this was tested through a DC analysis with the temperatures −40 °C, 27 °C and 80 °C.

As was seen in fig. 27, there is no difference in static current when only one pin is active. However there are considerable differences when the two input combinations with one single active pin are compared to the other input vectors '00' and '11'. This relationship between the currents at each input combination remains consistent across the three temperatures.

At every temperature the ratio between the highest and the lowest current was calculated to explore how the relative difference in current between the input vectors is affected by temperature. At 80 °C, the highest current of 152 pA was 3.6 times greater than the lowest current of 42.3 pA, resulting in the lowest high-to-low ratio among the tested temperatures. At −40 °C the high-to-low ratio increases by a substantial amount where the highest current of 26.28 pA is 42.4 times greater than the lowest current of 0.62 pA. The currents at 27 °C are the same as the ones reported in section 3.4.1.

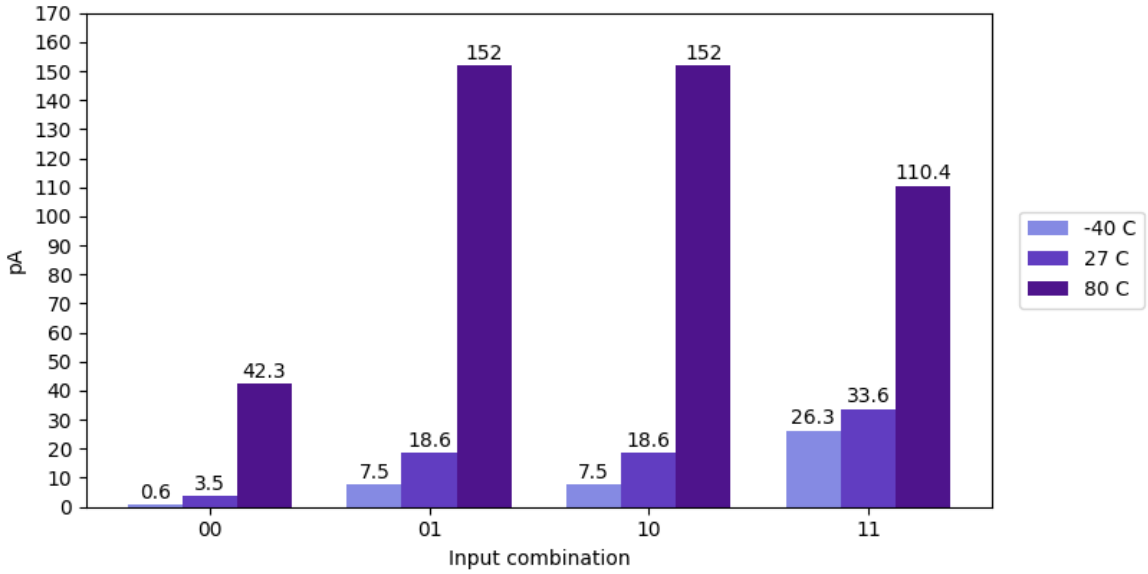

Figure 31: Static currents from a dual-pulldown NAND gates in three different temperatures

The currents across different temperatures were evaluated using the same metrics as was used earlier when other logic styles were presented. Analyses of the current-set obtained at 27 °C are presented in section 3.4.1.

At −40 °C, the interval in eq. (7) was found for the currents across each input combination with a mean value of $\overline{I}_{static} = 10.46\,\text{pA}$. The temperature has an impact on the mean current, however there is no significant change in the standard deviation of the current set, which is $\sigma = 9.55\,\text{pA}$. This means that there are larger relative changes in current within the −40 °C temperature.

$$0.62\,\text{pA} \leq I_{static_{-40}} \leq 26.28\,\text{pA} \tag{7}$$

Finally, the same calculations were carried out for the set of currents obtained at 80 °C. The resulting current-set has an interval as described in eq. (8), a mean value of $\overline{I}_{static} = 114.15\,\text{pA}$ and a standard deviation of $\sigma = 44.86\,\text{pA}$. The size of the standard deviation shows that the largest swings in static currents in terms of absolute value occurred at this temperature.

$$42.25\,\text{pA} \leq I_{static_{80}} \leq 152.01\,\text{pA} \tag{8}$$

To compare the performance of this logic style across different temperatures, the RSD was calculated for each set of static currents. As mentioned earlier, the largest relative change, and by extension, the largest RSD was observed at $-40\,^{\circ}\text{C}$ with a value of 91.23 %. The RSD was then calculated for the set of currents at $27\,^{\circ}\text{C}$ and found to be 57.34 %. Finally, the RSD was calculated for the current set obtained at $80\,^{\circ}\text{C}$, which was found to be 39.3 %. The metrics presented in this section reveal a trend of increasing absolute changes and decreasing relative changes in the static current as the temperature increases.

A table summarizing the results of the temperature analysis is shown table 3.

Temperature	Mean current	Standard deviation	RSD
$-40\,^{\circ}\text{C}$	10.46 pA	9.55 pA	91.23 %
$27\,^{\circ}\text{C}$	18.59 pA	10.66 pA	57.36 %
$80\,^{\circ}\text{C}$	114.15 pA	44.86 pA	39.3 %

Table 3: Summary of current-characteristics of a dual-pulldown NAND gate for different temperatures

3.5 Quadruple dual-pulldown logic

In an effort to develop a logic style that possesses the benefits seen in octuple logic in section 3.3, with balance in both its static current as well as its transition current, without the need for eight separate internal logic gates, Quadruple Dual-Pulldown Logic (QDPL) was created. This logic style is based on the same methodology as ELB logic where one QDPL logic gate consists of four internal logic gates. However, in QDPL the internal standard CMOS logic gates are substituted for dual-pulldown logic gates, like the ones analyzed in section 3.4. The internal connection of a QDPL NAND gate is shown in fig. 26, showing that it is connected in the same way as an ELB NAND gate.

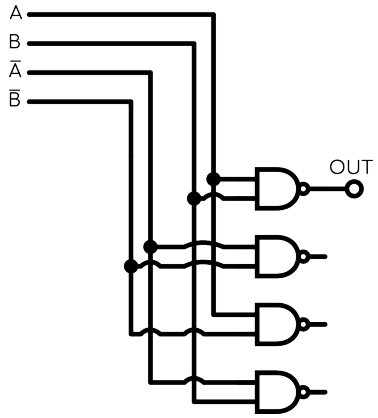

Figure 32: Quadruple dual-pulldown NAND gate

3.5.1 Obtaining static currents and transition currents through DC analyses

Initially, DC analyses were conducted to record the static current consumption of the QDPL NAND gate at every input combination. The currents are plotted in fig. 33. The plot demonstrates that quadrupling dual-pulldown NAND gates provides the same advantages as quadrupling standard CMOS NAND gates in terms of static current, as discussed for ELB logic in section 3.2. Meaning that QDPL achieves an input-independent static current in ideal simulations.

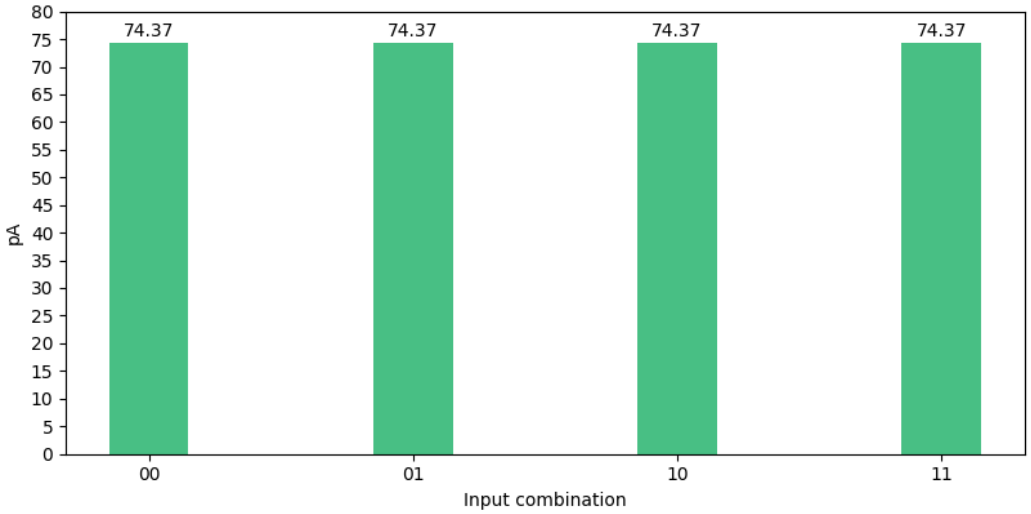

Figure 33: Static current consumption based on input combination for a QDPL NAND gate

Once the input-independent behavior of the QDPL NAND gate's static current was established, further analyses were conducted to investigate whether or not there was a difference in single-input transition current depending on if the A or the B input signal causes a transition of the logic gate. As depicted in fig. 34, the plot demonstrates that the single-input DC transition currents of a QDPL NAND gate completely overlap with each other. This indicates that the advantages of the internal dual-pulldown logic gates are successfully transferred into the quadrupled version.

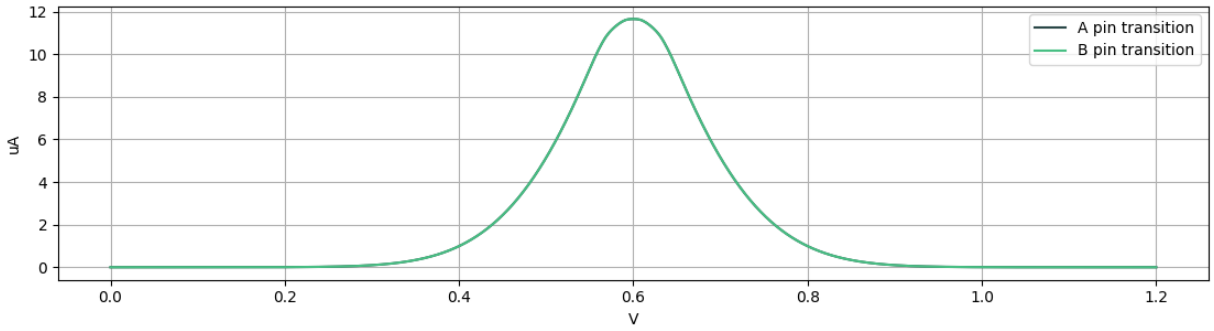

Figure 34: DC transition current of a QDPL NAND gate

3.5.2 Analyses of transition currents in the time domain

To achieve equal transient transition currents for every possible transition of the input signals, all four internal logic gates require an equal load. Similar transient analyses that have been conducted for the previously discussed logic styles were repeated for QDPL . Specifically, every possible transition of the input signals was simulated to inspect the resulting current consumption curves. Then the difference in current consumption between a transition of the A input signal and the B input signal was calculated. The plots in fig. 35 show that there is practically no difference between the single-input transition currents from the transient analyses. When the difference between them is calculated, the resulting peak value is around 0.5 aA.

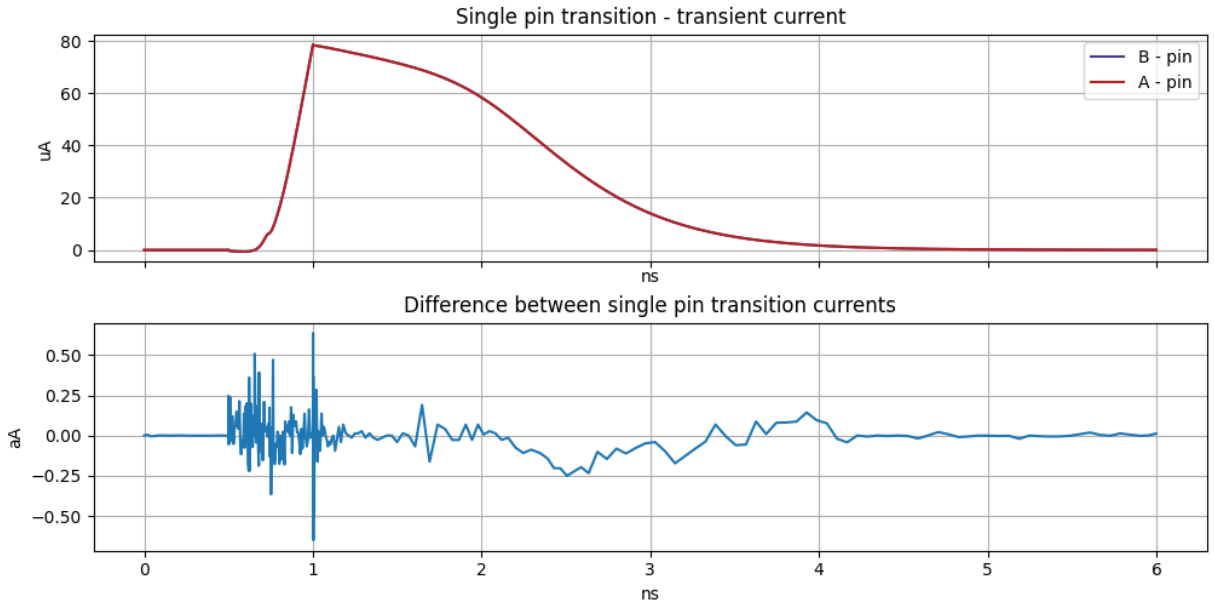

Figure 35: Transient single-input currents and the difference between them for a QDPL NAND gate

Although the single-input transition currents are balanced for this logic style, there are still other possible current waveforms when it comes to the dynamic current. Similar to the previously discussed logic styles, QDPL also has a different current waveform when both inputs change simultaneously, as shown in fig. 36. When analyzing every possible transition of the input signals to a QDPL NAND gate with four identical loads in an ideal simulation, the logic gate will display one of the two current traces depicted in the plot below. Regardless of whether the input signals change from a high to a low or vice versa, the logic gate will show one of the two current curves in ideal simulations, and the only deciding factor as to which of the two curves it will consume is

whether both input pins are changed or only one of them changes.

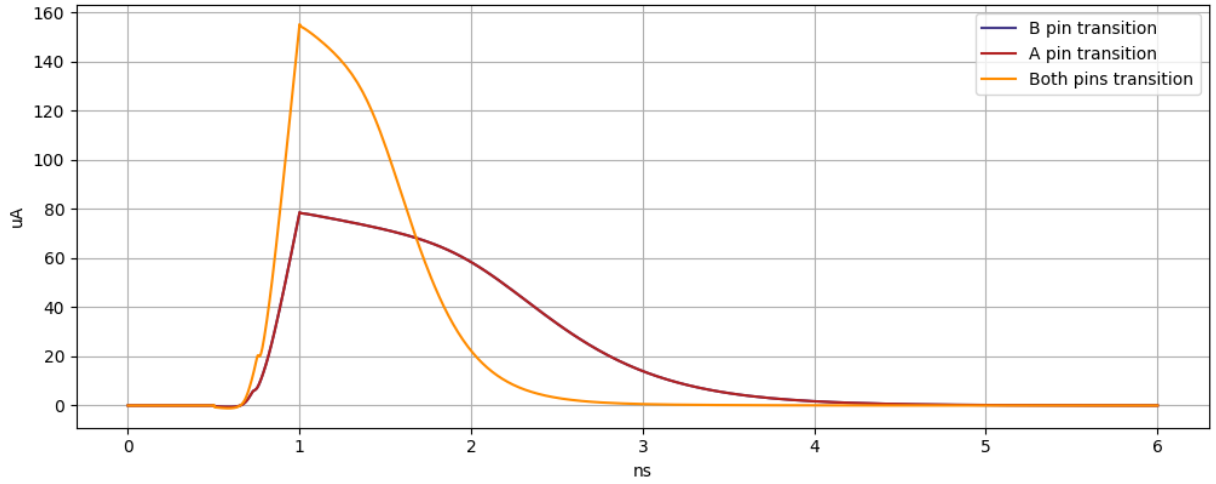

Figure 36: The two possible dynamic current traces for a QDPL NAND gate

3.5.3 The impact of temperature on the static current

Similar to the analysis done on the previously discussed logic styles, the impact of temperature on the static current of this logic style was examined. As with the other logic styles that involve a set of internal logic gates to process all input combinations simultaneously, the static current of this logic style remained input-independent at each temperature that was tested. The temperatures used in this analysis was $-40\,^{\circ}\mathrm{C}$, $27\,^{\circ}\mathrm{C}$ and $80\,^{\circ}\mathrm{C}$ and the resulting static currents are plotted in fig. 37.

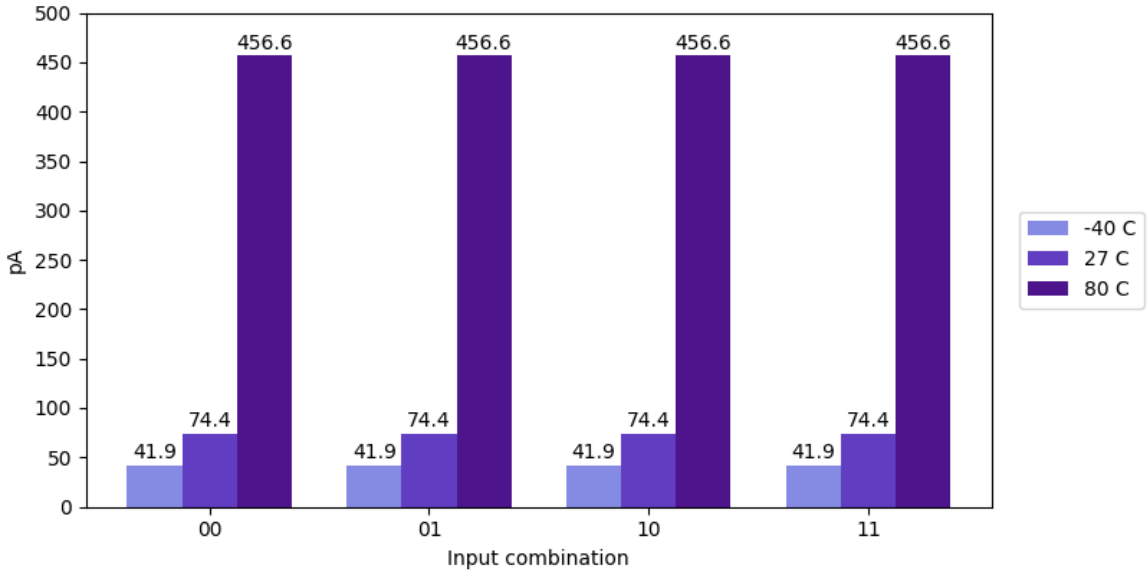

Figure 37: QDPL static current consumption across different temperatures

3.5.4 Imbalances in the static current due to intra-die mismatch

Similar to the analysis conducted previously to study the performance of different logic styles in the presence of mismatch, a Monte Carlo simulation with 1000 iterations was carried out to determine the average current and standard deviation for each input combination of a QDPL NAND gate. The result of that simulation is plotted in fig. 38.

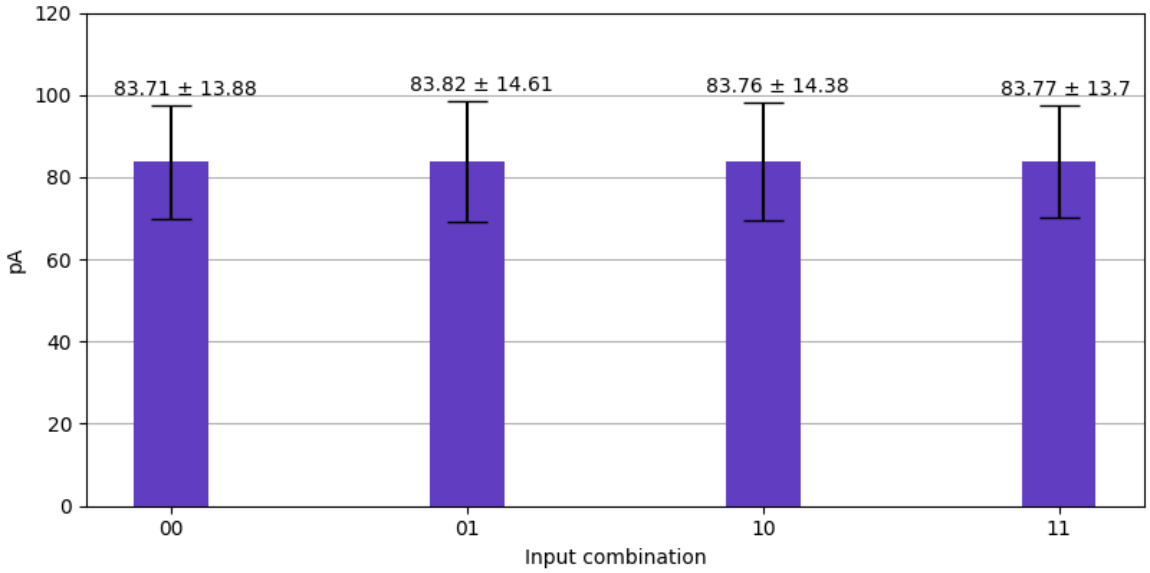

Figure 38: QDPL average static currents from Monte Carlo simulation

The findings from this simulation is similar to the findings of the other logic styles that was based on internal logic gate duplication to achieve a balanced static current. As can be seen in the plot above both the average currents and the standard deviation are very similar between the input combinations.

Next, a smaller Monte Carlo simulation of 10 iterations was conducted to generate individual examples of instances of mismatch. One of the resulting parameter sets was picked out and is plotted together with the ideal current of a QDPL NAND gate in fig. 39.

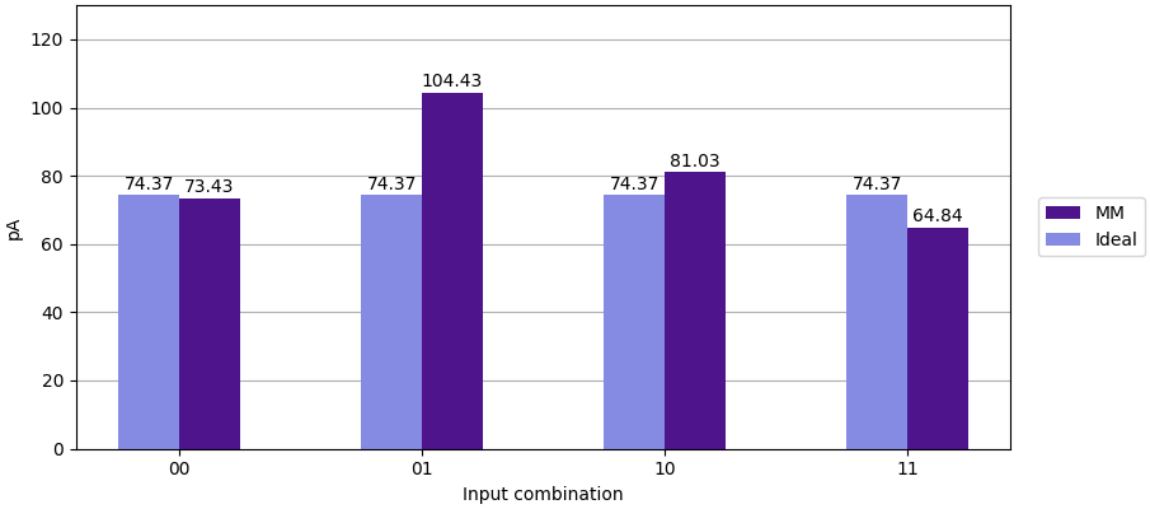

Figure 39: QDPL example instance of input-dependence due to mismatch

The resulting set of currents caused by the mismatch has an interval like the one described in eq. (9). The standard deviation of the set is $\sigma = 14.73\,\text{pA}$ which in relation to the mean of $\overline{I}_{static} = 80.93\,\text{pA}$ equals an RSD of 18.2 %

$$64.84\,\text{pA} \leq I_{static} \leq 104.43\,\text{pA} \tag{9}$$

Then a Monte Carlo simulation with 1000 iterations was used to gather averaged intra-set metrics using the procedure and test bench described in section 2.4.3. Obtaining averages gives a more general view of what level of input-dependency can be expected when mismatch is introduced, and this analysis will also demonstrate that the single iteration that was analyzed earlier in this section does not demonstrate a particularly drastic case of data-dependency.

The analysis revealed that the average interval across all sets of currents was of the size described in eq. (10). The average mean value of the sets was calculated to be $\overline{I}_{static} = 83.75\,\text{pA}$, and the average standard deviation was found to be $\sigma = 10.64\,\text{pA}$. As a result, the average RSD from this simulation was 12.7 %.

$$55.17\,\text{pA} \leq I_{static} \leq 166.75\,\text{pA} \tag{10}$$

A summary of the mismatch analyses is provided in table 4.

Simulation	Mean current	Standard deviation	RSD
Single instance	80.93 pA	14.73 pA	18.2 %
Average	83.75 pA	10.64 pA	12.7 %

Table 4: QDPL mismatch summary

3.6 QDPL with internal input generation

Until this point, all tests conducted on logic styles that require the inversion of input signals were done using ideal voltage sources for both input signals and their inverses. However when creating larger circuitry with such logic gates, the need to supply a signal and its inverse to every logic gate becomes an obstacle very quickly. One attempt to overcome this obstacle is to generate the inverse of the input signals internally in the logic gate. The authors of [13] did this for ELB by using four inverters as depicted in fig. 40. This section presents the results of implementing such an input generation scheme in a QDPL NAND gate.

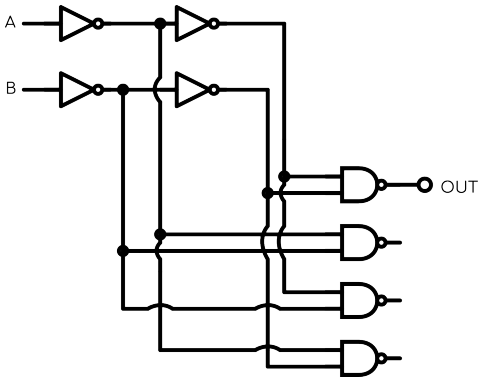

Figure 40: Quadruple NAND gate with internal input signal generation

3.6.1 Choice of inverter

When adding circuitry to the logic gate to generate the inverted input signals, it is important to do so in a way that does not disturb the balance that the logic style itself achieves. The choice of inverter is therefore an important one. To establish a baseline for the static current characteristics that could be expected from an inverter, a DC analysis was conducted where the static current was recorded for a standard CMOS inverter with the transistor dimensions mentioned in section 2.2, like seen in fig. 42, for both possible input values and is plotted in fig. 41.

The plot clearly demonstrates that a standard CMOS inverter's static current consumption is not balanced, as it draws more current when the input is at a logic high compared to a logic low. The ratio between the highest and lowest current is 1.384. To reduce this ratio, an alternative inverter topology was used, known as the Symmetric Dual-Rail Logic (SDRL) inverter, which was introduced in [8]. The SDRL inverter connects a second inverter to the output of the first inverter, ensuring that one inverter processes a logic '1' and the other processes a logic '0', regardless of the top-level input to the inverter.

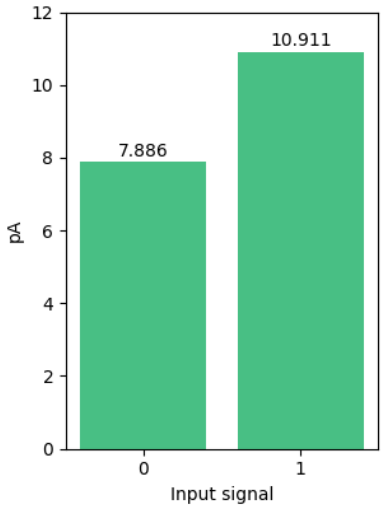

Figure 41: Static current consumption of a standard CMOS inverter

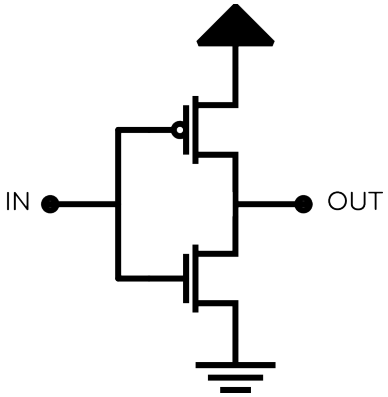

Figure 42: Schematic of a standard CMOS inverter

In fig. 44, a schematic of a SDRL inverter is presented. A DC analysis was conducted on this topology to record the static current based on the input signal. The results indicated that, while not perfectly balanced, the SDRL inverter had a smaller ratio between its two different static current consumptions compared to the standard CMOS inverter. The static current consumption was highest when the input was logic '0' and slightly lower when it was logic '1', as shown in fig. 43. During the tests, the highest current consumption was 1.028 times greater than the lowest current. This was achieved by keeping the transistor lengths at 120 nm but changing the widths of both the NMOS and PMOS transistors to 170 nm as opposed to the widths mentioned in section 2.2, which was used for the standard CMOS inverter. Given the closeness between the two static currents, it was decided to use SDRL inverters when proceeding to construct a QDPL NAND gate with internal input generation.

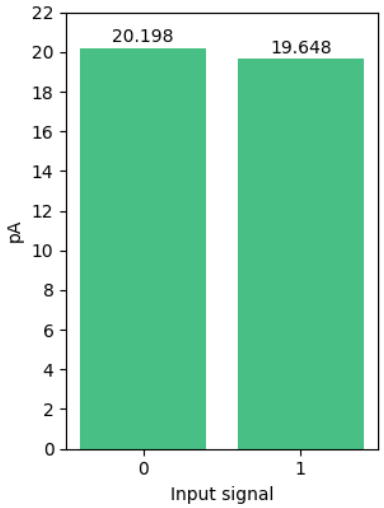

Figure 43: Static current consumption of a SDRL inverter

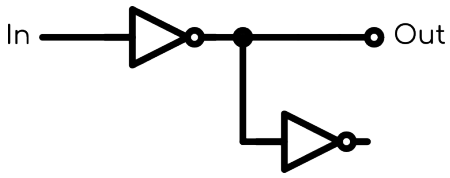

Figure 44: SDRL inverter

3.6.2 Implementation in the NAND gate

Analysis was conducted on both ELB logic and QDPL with internal SDRL inverters to explore the effects of the extra circuitry on the input-dependence in the static current of the logic styles. The current was recorded for every input combination and the highest and lowest currents that was found for each logic style is plotted in fig. 45 and fig. 46. The lowest static current was seen when a '00' was present on the inputs, and the highest current was seen when the inputs was '11'. This was true for both logic styles in the tests that were conducted. Neither one of the logic styles shows a substantial difference in current consumption between the highest and the lowest current values. Which is to be expected since none of the underlying logic styles has any difference in their current, therefore all imbalances seen in fig. 45 and fig. 46 are solely caused by the four internal inverters. This is indicated further by the fact that the same difference of 2.92 pA is seen in both cases. Given the additional benefits seen in QDPL compared to ELB , it was decided to conduct further analyses on QDPL with internal input generation, the results of which are presented in the following sections.

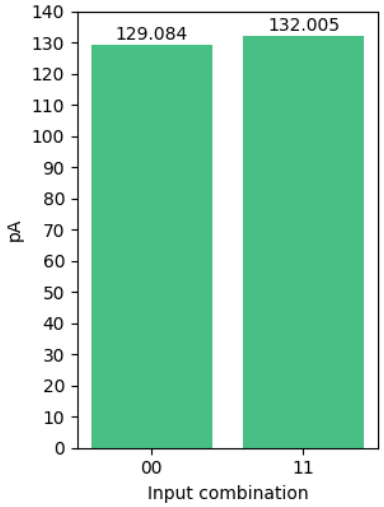

Figure 45: Static current consumption of an ELB gate with internal input generation

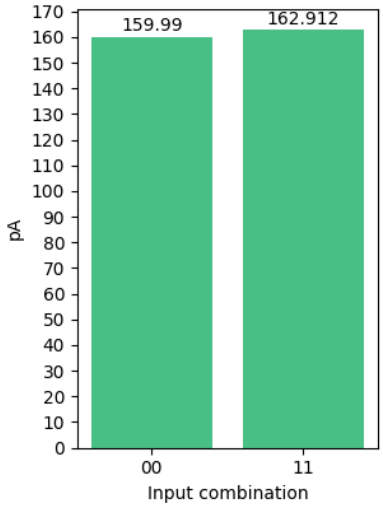

Figure 46: Static current consumption of a QDPL gate with internal input generation

Further analysis on QDPL with internal input generation will be conducted later in this thesis in a case where such logic gates are used to create a larger system.

3.6.3 Obtaining static currents and transition currents through DC analyses

It was found through DC analyses where the static current was recorded for every input combination that the addition of the internal inverters causes a slight input-dependency in the logic gate. This is displayed in fig. 47 where a small difference can be seen in the static current between the inputs '00', '11' and the inputs where only one input is active. From the results of this analysis the interval of static currents can be expressed as eq. (11).

$$159.99\,\text{pA} \leq I_{static} \leq 162.91\,\text{pA} \tag{11}$$

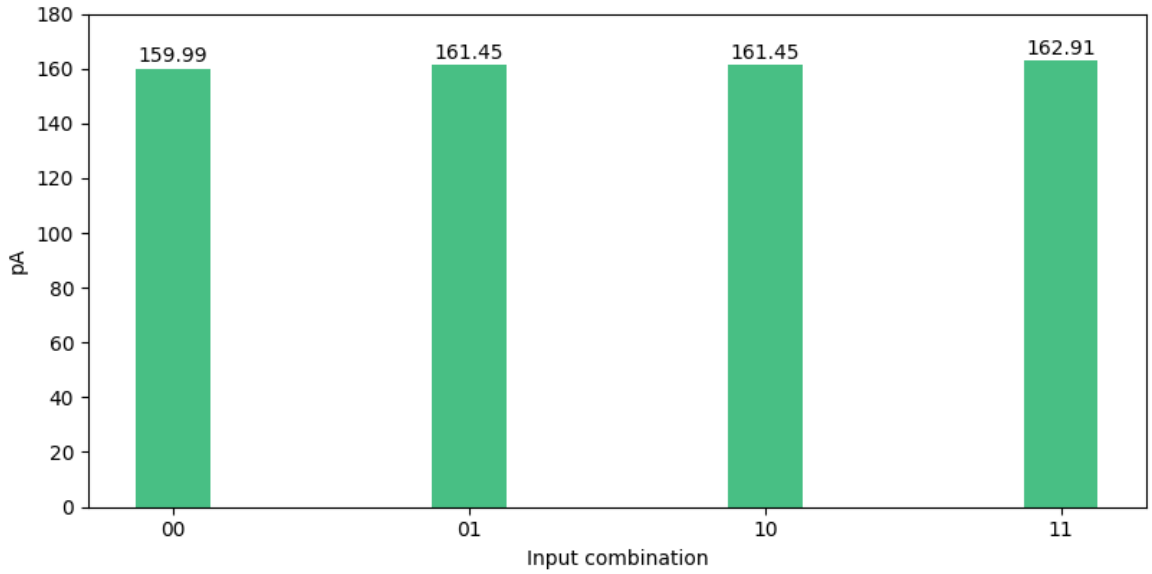

Figure 47: QDPL NAND with internal input generation static current

As the addition of internal inverters introduced some variations in current between different input combinations, metrics can be calculated to quantify the input-dependency. The set of currents has a mean value of $\overline{I}_{static} = 161.45\,\text{pA}$ and a standard deviation of $\sigma = 1.03\,\text{pA}$. The magnitudes of the mean and the standard deviation results in a RSD of 0.64 %.

With the added inverter circuitry inside of the QDPL NAND, the DC transition current takes on a different shape. While there is still no difference in the single-input transition currents between the A pin or B pin, there is an extra spike added to the current waveform. This spike is generated from the collective peak switching currents of all the internal inverters that are activated during the input sweep.

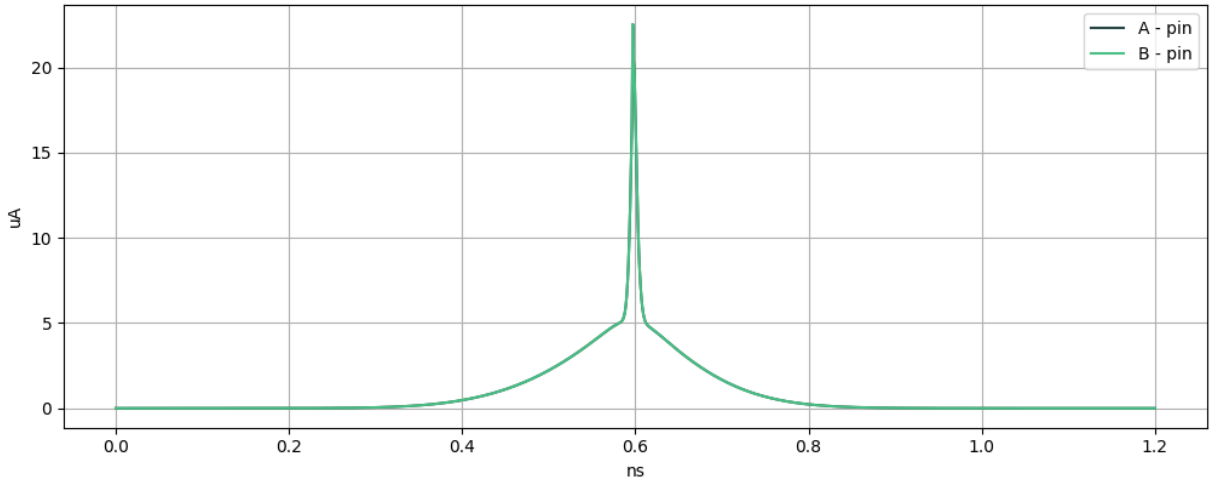

Figure 48: QDPL NAND with internal input generation DC switching current

3.6.4 Analyses of transition currents in the time domain

When internal input generation is included in a QDPL NAND gate, there are no longer only two dynamic current curves when every possible transition is inspected in a transient analysis. To illustrate this, the logic gate was taken through every possible transition of the input signals and the resulting current waveforms is plotted in fig. 49.

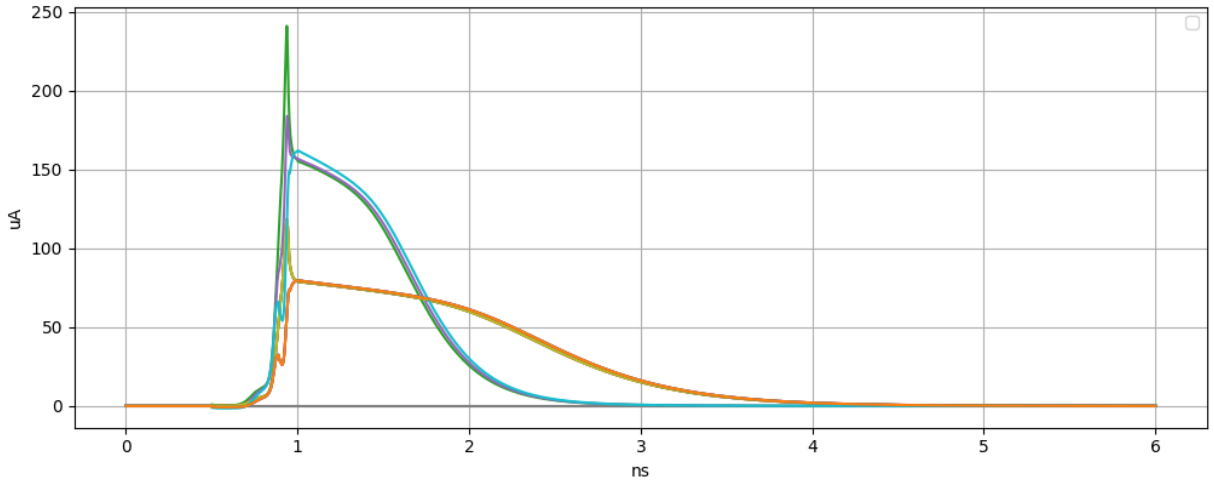

Figure 49: QDPL NAND with internal input generation dynamic transition currents

It is observed that the logic style is no longer indifferent to the direction in which the input voltages are swept after implementing the input generation. When two single-input sweeps are

compared where one is swept from V_{DD} to GND and the other is swept in the opposite direction, a visible difference is seen in the early parts of the transition as seen in fig. 50. However when two single-input transitions are compared where the input signals are swept in the same direction, the difference between them is negligible. This means that if the A input is swept from GND to V_{DD}, the resulting current waveform will be almost identical to the waveform obtained when doing the same with the B input signal. A plot showing all single-input transitions as well as the difference in current consumption between the A and B input sweeps of the same direction is shown in fig. 50. When the input voltage is swept in the same direction, the difference in the resulting current consumption is in the realm of tenths of femtoamperes.

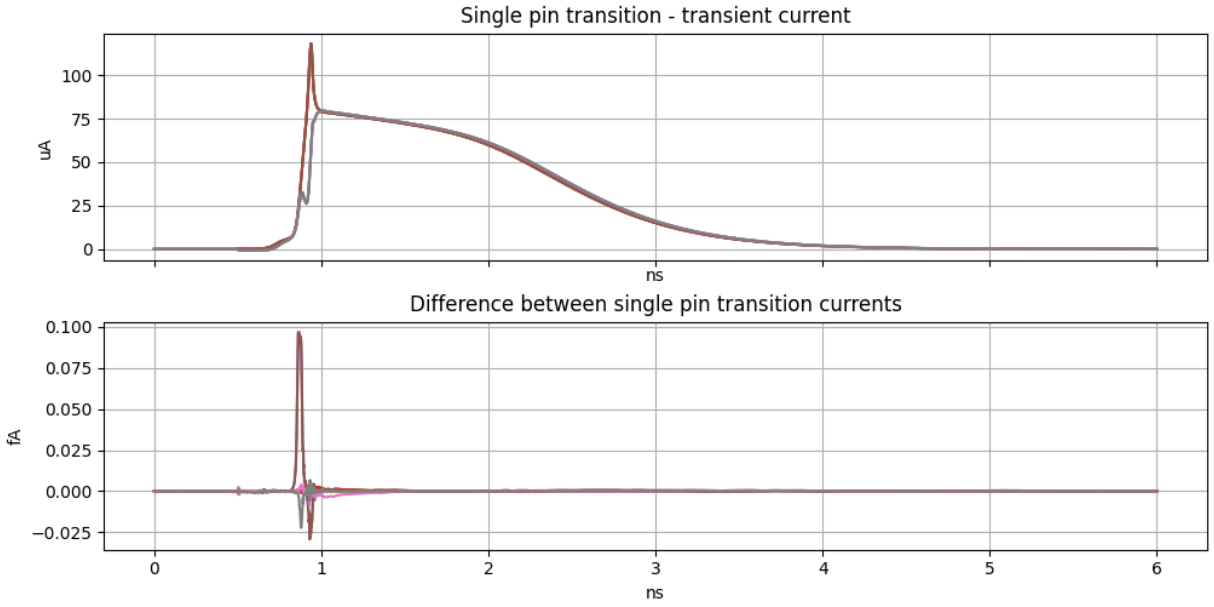

Figure 50: Single-input transition currents from a QDPL NAND with internal input generation

To further illustrate the difference that is observed between two input sweeps of opposite directions, a plot was made of the resulting current traces of two simulations, one where the A input was swept from GND to V_{DD} and one where it was swept in the opposite direction. Before the implementation of the internal inverters this would have resulted in two identical current waveforms, but as can be seen in fig. 51 this is not the case after the implementation of the inverters.

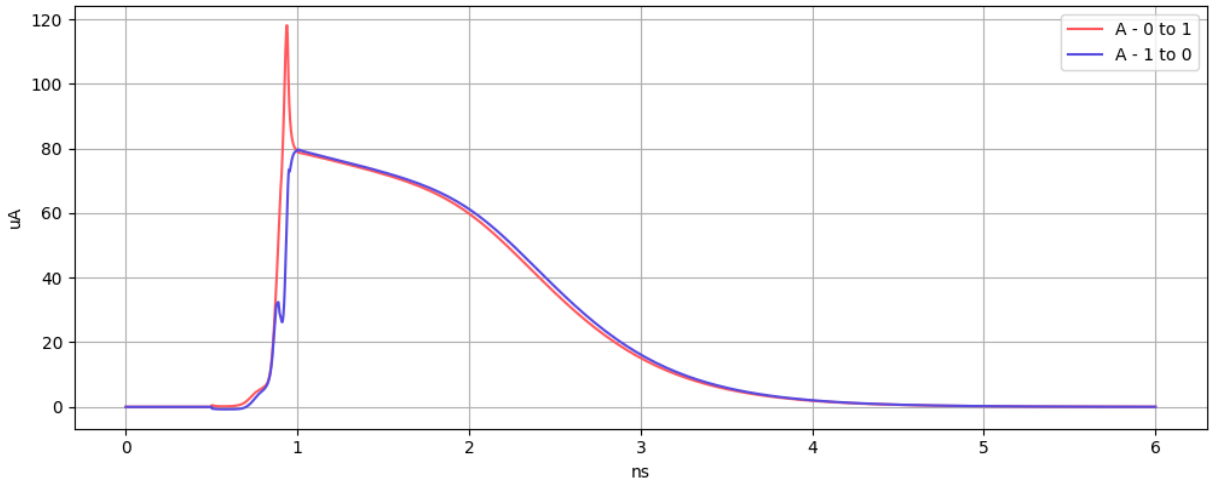

Figure 51: Dynamic current trace for different voltage directions in a QDPL with internal input generation

3.6.5 The impact of temperature on the static current

As shown in the previous sections, the static current of this logic style is found to be slightly input-dependent, raising the question of how this input-dependency is affected by changes in the circuit's temperature. To investigate this further, a DC simulation was conducted where the static current consumption of the logic gate was recorded for all input combinations at three different temperatures, −40 °C, 27 °C and 80 °C. The plot in fig. 52 shows the resulting currents. The results of these simulations can be used to calculate a few metrics.

First, the difference between the highest and lowest recorded current was calculated for each temperature. This revealed that the absolute value of the largest difference within each temperature stayed roughly the same. The difference that was found for −40 °C, 27 °C and 80 °C was 2.76 pA, 2.92 pA and 3.06 pA respectively.

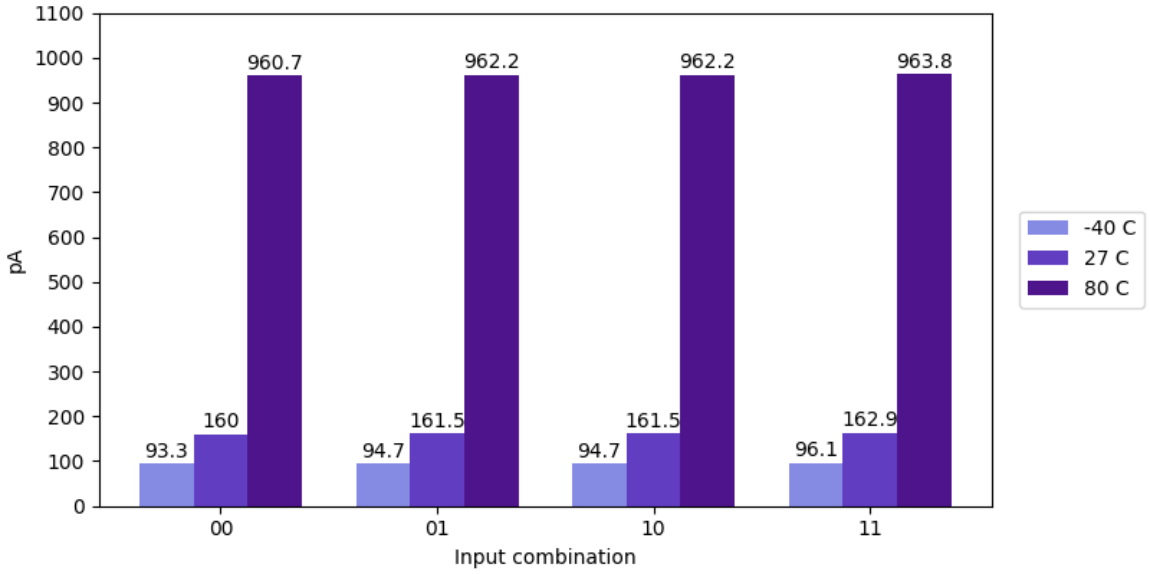

Figure 52: Static current of QDPL with internal input generation across temperatures

More metrics can be calculated to further explore the impact of temperature, the metrics for 27 °C are already presented in section 3.6.3. In this section, the set of currents obtained from the test at −40 °C is analyzed first. At this temperature the static currents fell within the interval seen in eq. (12). The set has a mean value of $\overline{I}_{static} = 94.69\,\text{pA}$ with a comparatively low standard deviation of $\sigma = 0.98\,\text{pA}$ which results in a RSD of 1.03 %. This shows that the relative difference in static current increases as temperature decreases given that the RSD increased from 0.64 % which was the case at 27 °C.

$$93.31\,\text{pA} \leq I_{static_{-40}} \leq 96.06\,\text{pA} \tag{12}$$

On closer inspection of the current set at 80 °C, it is apparent that the mean value of this set is substantially higher than that of the lower temperature sets. The currents at this temperature fall within the interval described in eq. (13), with a mean value of $\overline{I}_{static} = 962.24\,\text{pA}$ and a standard deviation of $\sigma = 1.08\,\text{pA}$. Although the standard deviation remains roughly constant across all three temperatures, the mean current value increases significantly with temperature, resulting in a reduction in the RSD. For this set, the RSD was calculated to be 0.11 %.

$$960.71\,\text{pA} \leq I_{static_{80}} \leq 963.77\,\text{pA} \tag{13}$$

A summary of the temperature analyses is presented in table 5.

Temperature	Mean current	Standard deviation	RSD
−40 °C	94.69 pA	0.98 pA	1.03 %
27 °C	161.45 pA	1.03 pA	0.64 %
80 °C	962.24 pA	1.08 pA	0.11 %

Table 5: Summary of temperature analyses of QDPL with internal input generation

An interesting finding with this logic style is that the standard deviation stayed roughly equal across all three temperatures that was tested. This means that the absolute value of the change in current was not significantly impacted by changing the temperature in either direction. The change in current relative to the mean will decrease with temperature as a result of the mean value increasing while the standard deviation stays almost static.

As a last point of comparison, an ELB NAND gate was constructed with the same internal input generation mechanism and tested for the same temperatures. Identical values was found for standard deviation at every temperature, indicating that the deviation in currents is caused solely by the inverter circuits.

3.6.6 Imbalances in the static current due to intra-die mismatch

This logic style was of particular interest regarding the effects of mismatch, as even ideal simulations showed an imbalance in the static currents. To analyze this further, the same procedure as in earlier sections was repeated. A set of averaged currents and standard deviations for each input combination was collected from a Monte Carlo simulation with 1000 iterations, and is plotted in fig. 53.

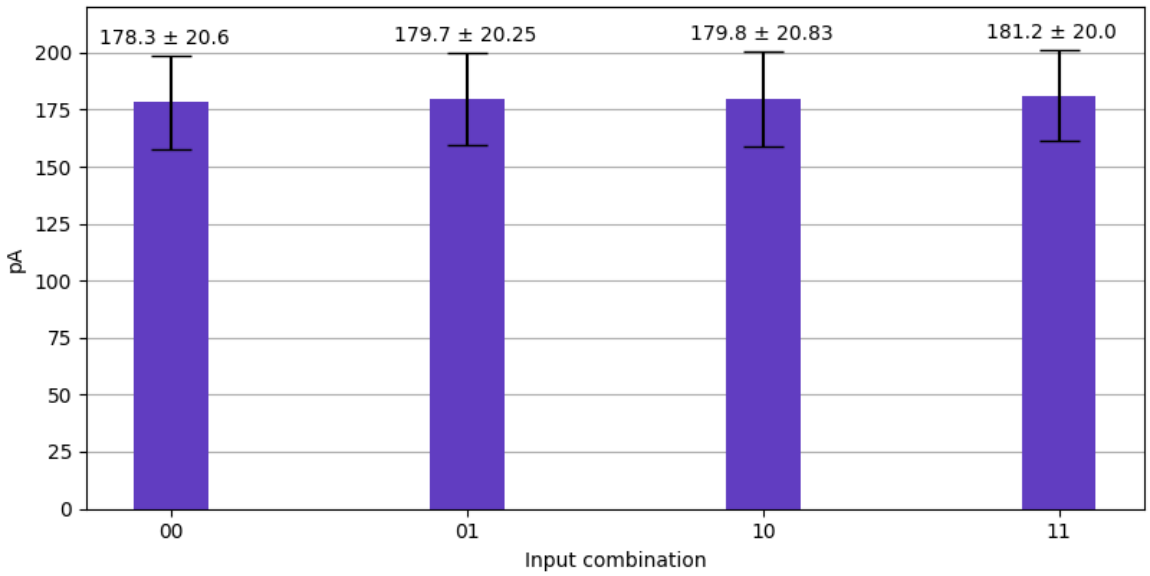

Figure 53: Average static current of QDPL from a Monte Carlo simulation

The same trend is seen in the average currents as was seen in ideal simulations of QDPL , where the current consumed at an input of '00' is the lowest, the current at an input of '11' is the highest and the current for the two input combinations with one active pin is in between and are very similar in size.

Next, a smaller Monte Carlo simulation of 10 iterations was conducted to generate some examples of the logic styles performance when operating in the presence of mismatch. An iteration was

chosen as an example and is plotted together with the currents from an ideal simulation in fig. 54.

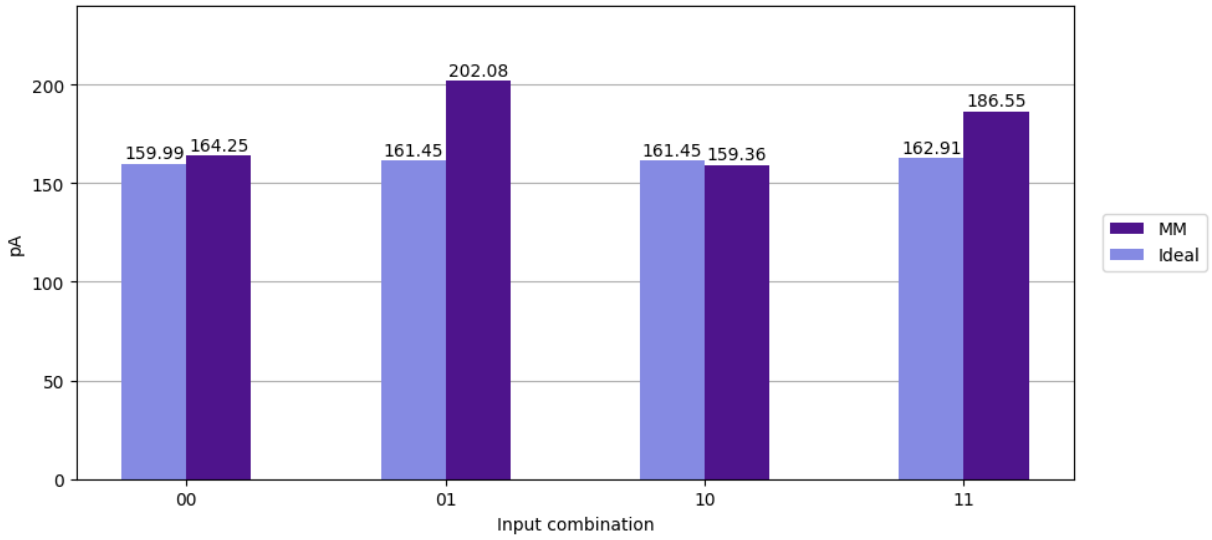

Figure 54: Example static current of QDPL with mismatch effects

This set of currents has a mean value of $\overline{I}_{static} = 178.06\,\text{pA}$ and a standard deviation of $\sigma = 17.24\,\text{pA}$. This results in a RSD of 9.68 %. The set had an interval like the one described in eq. (14).

$$159.36\,\text{pA} \leq I_{static} \leq 202.08\,\text{pA} \tag{14}$$

The degree of variation between input combinations within a set of currents was evaluated using a Monte Carlo simulation with 1000 iterations to get averaged metrics in addition to the single example iteration. To gather this information, the intra-set analysis procedure described in section 2.4.3 was used. The average mean current from this analysis was found to be $\overline{I}_{static} = 179.6\,\text{pA}$, while the average standard deviation was $\sigma = 15.58\,\text{pA}$, with a resulting RSD of 8.67 %. The average interval of currents that was found is described by eq. (15).

$$137.58\,\text{pA} \leq I_{static} \leq 294.68\,\text{pA} \tag{15}$$

A summary of the analyses on the effects of mismatch within a set of currents is presented in table 6.

Simulation	Mean current	Standard deviation	RSD
Single instance	178.06 pA	17.24 pA	9.68 %
Average	179.6 pA	15.58 pA	8.67 %

Table 6: QDPL with internal input generation mismatch summary

3.7 Comparison of logic styles

The previous sections where a logic style is analyzed have all used the same metrics to quantify the logic style's performance. This section aims to summarize and compare the performance of the different logic styles directly to analyze them in relation to each other, and will be divided into two parts: results with and without mismatch.

First, the metrics calculated from the static currents obtained from ideal simulations are shown in table 7. These results show that the logic styles that are based on a set of internal logic gates to continually process every input combination demonstrate no variation in their static current across input combinations, and thus a standard deviation of 0 is seen for all of them. The methodology behind such countermeasures demonstrates solid performance in ideal simulations and even across temperatures.

However once internal input generation is introduced to QDPL , marked (IIG) in table 7, some variation is reintroduced to the logic style. The two other logic styles that show some variation in their static currents are CMOS and dual-pulldown logic, which is to be expected given that they are not designed to be countermeasures by themselves.

Logic style	Mean current	Standard deviation	RSD
Standard CMOS	10.91 pA	7.54 pA	69.13 %
Dual-pulldown logic	18.59 pA	10.66 pA	57.36 %
ELB	43.66 pA	0	0
Octuple logic	87.31 pA	0	0
QDPL	74.37 pA	0	0
QDPL (IIG)	161.45 pA	1.03 pA	0.64 %

Table 7: Logic gate performance metrics from ideal simulations

After the introduction of mismatch, the input-dependency of the logic styles underwent visible changes. The analyses with mismatch were performed exclusively on the logic styles designed as countermeasures against PAA . Table 8 shows the averaged metrics of the variation within a set of currents for each of those logic styles.

Out of the tests conducted on logic with ideal input sources, Octuple logic exhibited the highest standard deviation and mean current, whereas ELB showed the lowest values for the same metrics. The metrics for QDPL were found to be in between those found for ELB and octuple logic.

Comparing the RSD of the different logic styles reveals that ELB shows larger variations in its static current in relation to its mean current compared to the other two logic styles, while the RSD of octuple logic and QDPL are very close in value.

Its clear from the table that both the mean current and the standard deviation increases after internal input generation is added to QDPL, marked (IIG) in table 8.

Logic style	Mean current	Standard deviation	RSD
ELB	48.89 pA	7.99 pA	16.34 %
Octuple logic	97.81 pA	11.79 pA	12.05 %
QDPL	83.75 pA	10.64 pA	12.7 %
QDPL (IIG)	179.60 pA	15.58 pA	8.67 %

Table 8: Average metrics for logic gate performance in the presence of mismatch

4 Masking current consumption through bulk voltage variations

In this section, a method for introducing input-independent noise to the current consumption of logic gates by periodically randomizing the bulk voltages of their internal transistors is presented. This technique can be applied to any logic style where the bulk terminals of the transistors are not tied up in any functionally crucial connection, but they are instead accessible to be connected to the periodically changing voltage. However, the mask is presented here in combination with hiding logic styles like ELB and QDPL. Combining the mask with such a logic style can take advantage of the underlying hiding mechanism while obscuring the input-dependency caused by mismatch with additional input-independent noise.

The masking scheme works by adding dynamically changing voltages to the bulk terminal of the transistors that make up the logic gate. Two voltages are used in the implementation of the mask, one shared by all NMOS transistors and one shared by all PMOS transistors. If proper ranges are chosen for these voltages, they can be randomized at any time without disturbing the logic function of the logic gate. Once those criteria are met, voltage generators can be designed to periodically randomize the bulk voltages asynchronously from the main system where the logic gates operate. This masking scheme alters the majority of the various aspects of the circuit's current consumption. First, let's consider how the current consumption of the logic gate is affected by the body effect.

Due to the body effect, the threshold voltages of the transistors in a circuit with a bulk-mask applied will be altered as the bulk voltages are randomized. Where, for a given source potential, an increase in the bulk voltage corresponds to a decrease in threshold voltage. This has implications for multiple different aspects of a transistor's current consumption. When considering the sub-threshold current of a transistor that is in an off-state, an increase in threshold voltage results in a decrease in sub-threshold current [18]. This is also shown by the term V_{SB} in eq. (16), which is the source-to-body voltage for a given transistor, where an increase in bulk voltage while the source voltage is static results in a decrease of the V_{SB} term and subsequently the sub-threshold current.

$$I_{subth} = Ae^{\frac{q}{nkT}\cdot(V_{GS}-V_{TH0}-\gamma V_{SB}+\eta V_{DS})} \cdot (1 - e^{\frac{qV_{DS}}{kT}}) \tag{16}$$

A changing threshold voltage can also be seen to have effects on a transistor's current through the square law model seen in eq. (17)[19]. The threshold voltage V_{th} is a significant factor in the model, as it appears in several locations, including an exponential term.

$$I_D = \frac{1}{2}\mu C_{OX}\frac{W}{L}(V_{GS} - V_{th})^2 \cdot [1 + \lambda(V_{DS} - (V_{GS} - V_{th}))] \tag{17}$$

The following sections presents the results of applying this masking scheme on a single ELB NAND gate to begin with, then subsequently on an eight-bit register made of QDPL NAND gates. The current consumption of the circuits is analyzed and different metrics are presented.

4.1 Generating voltages for the bulk-voltage mask

In order to conduct transient analyses with an active mask, it was necessary to generate two dynamically changing voltages to be used as the mask-voltages. While there are numerous methods to achieve this, for the purposes of this thesis, only a proof of concept was needed so that the degree of modulation that could be achieved in a circuit's current consumption when utilizing the mask could be explored. For these reasons, simplicity was prioritized over other important aspects such as security and efficiency.

The voltage generation circuit used in this work consists of a five-bit Linear Feedback Shift Register (LFSR) connected to a five-bit Digital to Analog Converter (DAC), as shown in fig. 55. One such circuit was used for each of the two mask-voltages. The amplifier in the DAC was adjusted to supply voltages only within a given range and new voltages are generated every time the LFSR is clocked. The five bits of the LFSR results in 32 different possible voltage levels on the output of the DAC . Given the pseudo-random nature of the LFSR , this circuit will also give an indication of how the current of a circuit can be expected to be modulated from a truly randomly generated mask.

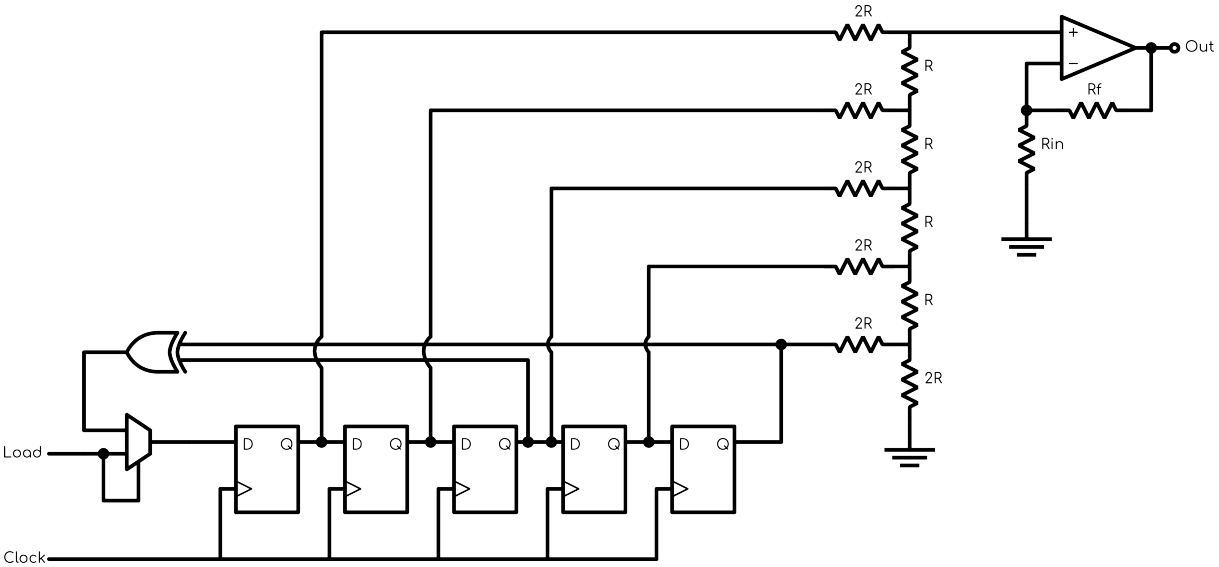

Figure 55: The circuit that was used to generate the bulk-voltage mask

4.2 Experiments on a single logic gate

Initial DC simulations were conducted on an ELB NAND gate to observe the degree of change that could be achieved in its static current depending on the voltages being applied to the circuits bulk terminals. These initial tests were done on an ELB NAND gate without internal input generation. Ideal voltage sources were used for the mask-voltages in these initial DC simulations.

4.2.1 Determining the degree of modulation in the static current

A range needed to be chosen for the bulk voltages for both the NMOS and PMOS transistors such that the circuit would keep its function while still being able to modulate the logic gate's static current as much as possible. The range in which the bulk voltages can be changed without causing a breakdown of the circuit's function depends on multiple different factors such as transistors sizing and the load that is being driven by the logic gate. To find the appropriate voltage ranges in this instance, a DC simulation was conducted where the ELB NAND gate was taken through a transition while a parametric sweep was conducted on both the NMOS and PMOS bulk voltages from GND to V_{DD}. The output voltage levels as well as the current consumption of the gate was observed at every iteration of the parametric sweep. Then two intervals could be picked out where no breakdown of the logic gate's function occurred, which in this instance was found to be the following:

$$0\,\mathrm{V} \leq V_{nb} \leq 0.4\,\mathrm{V} \tag{18}$$

$$0.7\,\mathrm{V} \leq V_{pb} \leq 1.2\,\mathrm{V} \tag{19}$$

A new parametric sweep of the bulk voltages was conducted within those ranges and the static current was sampled at each combination, the following plot could then be made to illustrate the degree of change that was achievable in the static current:

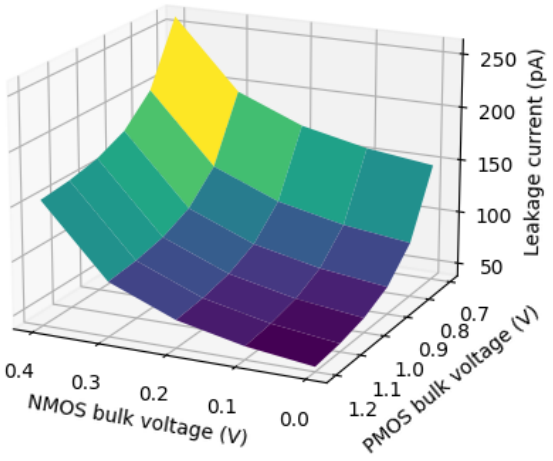

Figure 56: Static current consumption plotted against PMOS and NMOS bulk voltages

The plot shows that manipulating the bulk voltages of an ELB NAND gate within the ranges mentioned above results in a modulation of the current that is uniformly distributed between the two voltages, and the current can be modulated between values in the following interval:

$$43.66\,\mathrm{pA} \leq I_{static} \leq 275.16\,\mathrm{pA} \tag{20}$$

This means that it is theoretically possible to achieve a maximum current that is 6.30 times higher than the minimum current. The maximum current is drawn at:

$$(V_{nb}, V_{pb}) = (0.4\,\mathrm{V}, 0.7\,\mathrm{V}) \tag{21}$$

And the minimum static current is drawn at GND and V_{DD} such that:

$$(V_{nb}, V_{pb}) = (0\,\mathrm{V}, 1.2\,\mathrm{V}) \tag{22}$$

4.2.2 Utilizing the mask in the presence of mismatch

This masking scheme was presented as a way to mask input-dependency that is re-introduced in hiding logic styles from mismatch. When looking at the intervals of static currents caused by mismatch that was presented for ELB logic in section 3.2.4, the one Monte Carlo run that was picked out as an example of a singular instance of mismatch showed a highest-to-lowest current ratio of 1.71 and the corresponding ratio from the average values obtained from the larger Monte Carlo analysis showed a ratio of 1.43. The sizes of these ratios indicate that the bulk mask has the capability to cover the input-dependency with noise given that the mask has the power to modulate the current by a ratio that is several times larger, as the highest-to-lowest ratio seen from the interval in eq. (20) is 6.3.

In order to demonstrate the effect of the masking scheme in the presence of mismatch, a Monte Carlo simulation was repeated under identical conditions as the example instance of mismatch presented for ELB logic in section 3.2.4. However, in this instance, the simulation was conducted with the mask voltages set to those resulting in maximum current modulation, as shown in eq. (21). This will demonstrate the maximum amount of noise that is possible to add at every input combination. A bar graph displaying both the set of currents from when the mask is at its minimum modulation level, which is equal to the unmasked conditions that was presented in section 3.2.4, and from when the mask is at its maximum modulation level is shown in fig. 57.

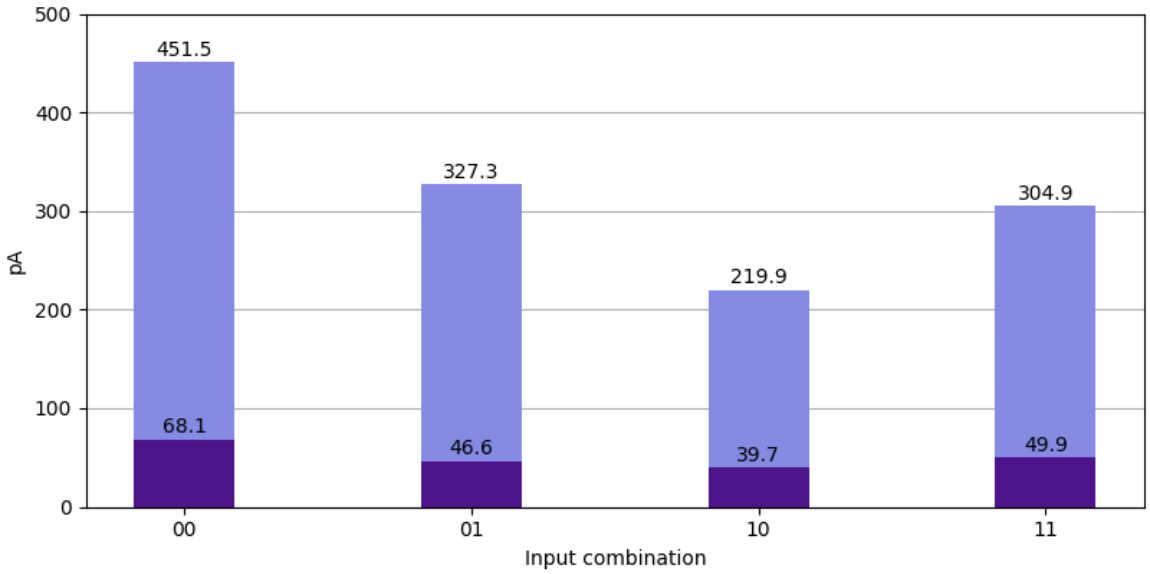

Figure 57: Static currents with mismatch with and without mask

In this plot, the lighter upper part of the bars represents the maximum amount of noise that can be added to the current for a given input combination. The amount of noise added to the current can fall anywhere within the lighter part of the bar depending on how the bulk voltages are adjusted. The mask consistently demonstrates its ability to multiply the unmasked current by a factor of around six. When the highest-to-lowest ratios are calculated for all input combinations seen in fig. 57, they are 6.63, 7.03, 5.54, and 6.11 for the inputs 00, 01, 10, and 11, respectively. Common to all these ratios is that they are comparatively large when looked at in relation to the variation caused by mismatch, which has a high-to-low current ratio of 1.71 in this instance of mismatch. This implies that the mask has the ability to add noise to the current consumption of the logic gate that is over three times larger in amplitude than the input-dependency caused by mismatch.

4.2.3 Exploring the mask's impact on dynamic current consumption

The investigation of the bulk-voltage mask's impact has so far focused on the static current. However, in order to assess its effects further, additional testing was conducted to explore the impact of the mask on the dynamic current consumption of the logic gate as well. To do this, the current consumption of an ELB NAND gate was simulated as it went through the same transition while different values of the bulk mask were applied. Throughout the transition, the mask voltages were held stable. The plot in fig. 58 illustrates the current traces that exhibited the highest and lowest peak amplitudes after recording the transition current of the logic gate for every combination of the mask voltages.

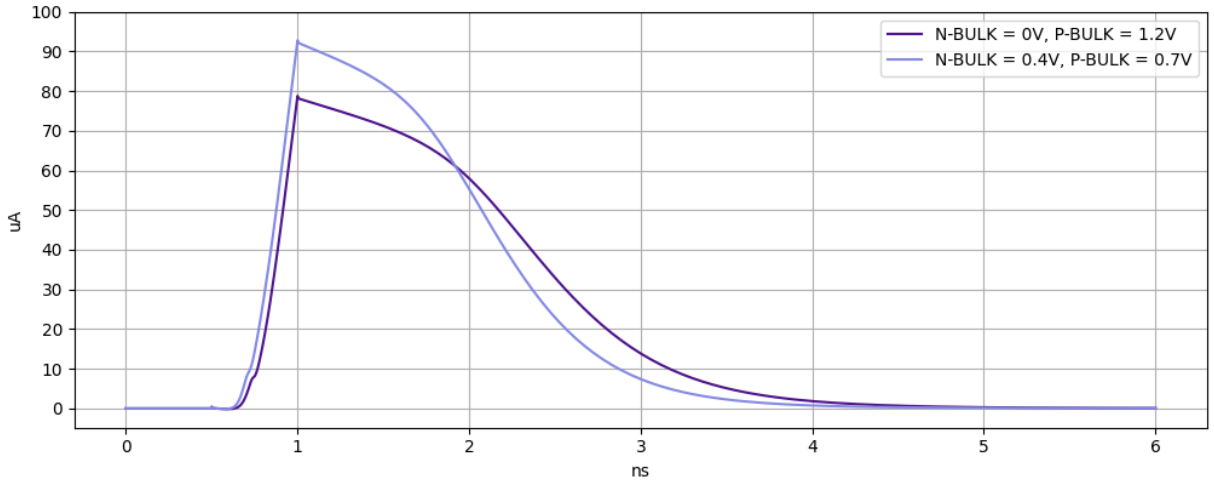

Figure 58: Highest and lowest dynamic currents in an ELB NAND gate with bulk-voltage mask applied

The bulk voltages that produced the highest and lowest peak current for the dynamic current consumption of the ELB NAND gate were found to be the same as those seen in the static current tests, shown in eq. (22) and eq. (21). As shown in fig. 58, the peak currents of the resulting waveforms were 78.8 µA when the bulk voltages were set to $(0\,\mathrm{V}, 1.2\,\mathrm{V})$ and 92.8 µA when set to $(0.4\,\mathrm{V}, 0.7\,\mathrm{V})$. Therefore, by utilizing the bulk mask, the range of the peak dynamic current in this case can be described as:

$$78.8\,\mathrm{pA} \leq I_{dyn-peak} \leq 92.8\,\mathrm{pA} \tag{23}$$

4.2.4 Utilizing dynamically changing bulk-voltages

To finally conduct a transient analysis where the mask-voltages are dynamically changing over time, a LFSR based voltage generator as described in section 4.1 was used to generate voltages in the correct ranges for the NMOS and PMOS bulk terminals. In this way, the system could be tested in a transient analysis where the bulk voltages are manipulated while the ELB NAND gate is operating. In this first analysis, the same simulation was repeated four times, one time for each unique input value of the NAND gate. The input values were then held constant through the whole simulation. The LFSRs had the same initial conditions in all four simulations so that they would generate the same mask-voltage sequence in every simulation. This was done to compare the current traces from each simulation and make sure that the balance of the ELB logic in ideal simulations is not undone by applying the bulk-voltage mask. As shown in fig. 59, the traces from each input combination are identical when this simulation is done on an ideal circuit. The generated bulk-voltage sequences and the resulting current are shown in the plot in fig. 59.

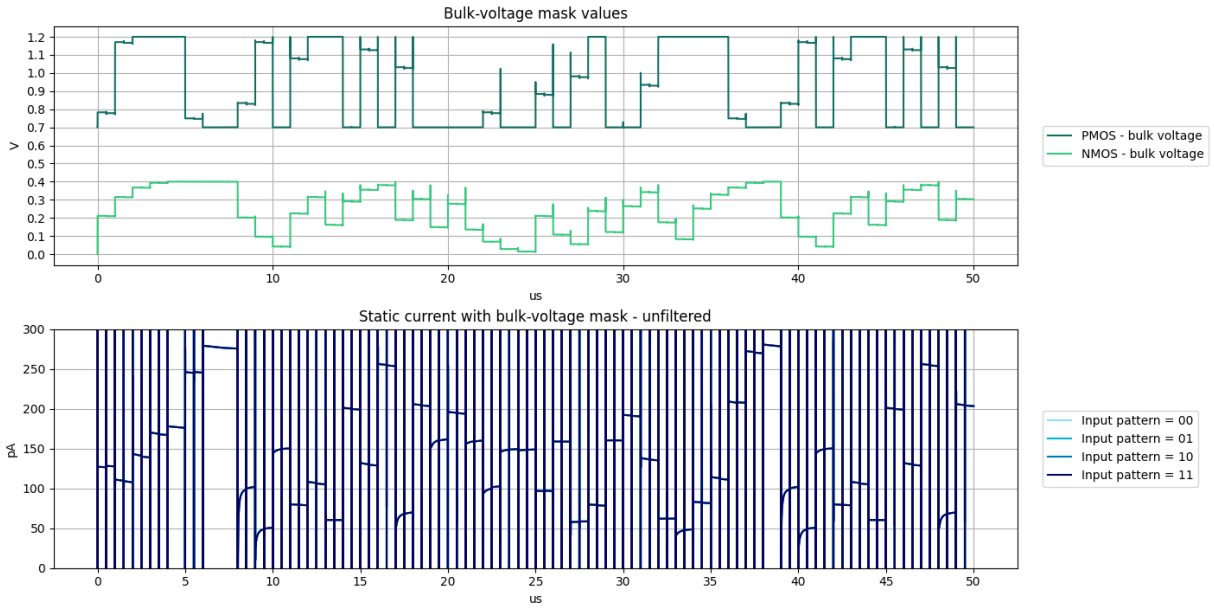

Figure 59: Unfiltered static current consumption of a ELB NAND gate with bulk-voltage mask

It is apparent that the resulting current is quite noisy because of the spikes that occur in the current every time the bulk mask switches to new voltages. The magnitude of these spikes is likely due to the use of ideal voltages sources in the simulation, and would be limited in a real-world scenario. However it is still possible to see the current levels in between the spikes and that the static part of the current is being altered by the bulk-voltage mask. To make the static part of the current clearer, the current was sampled once it had settled to a static level after every new mask voltage value then the samples was interpolated into a new plot and is presented in fig. 60. This sample and interpolate filter will be applied to every transient analysis where the static current is the primary focus from now on in this thesis.

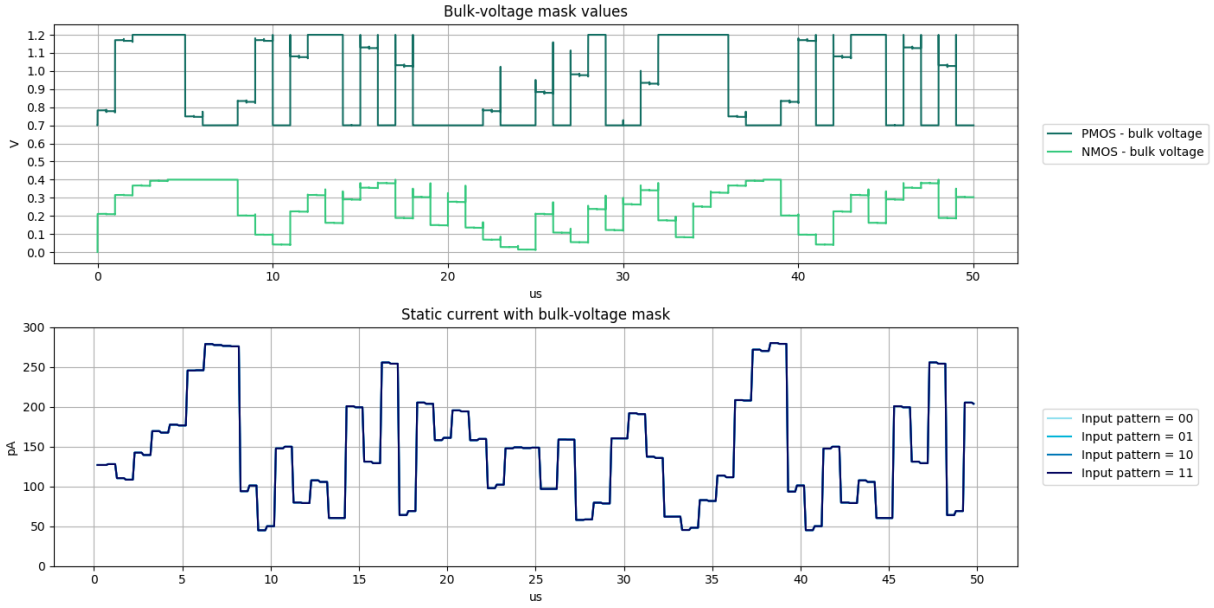

Figure 60: Filtered static current consumption of a ELB NAND gate with bulk-voltage mask

The current traces in fig. 60 shows that there is no deviation between the simulated currents depending on what values was present on the logic gates inputs when tested in an ideal simulation, meaning that the mask itself does not cause any input dependency to be introduced to the logic style. In these tests the highest current in the trace was found to be 279.80 pA and the lowest current was found to be 44.75 pA. Resulting in a maximum current that is 6.25 times higher than the minimum current.

4.2.5 Visualizing a dynamically changing mask in the presence of mismatch

The same simulation was repeated in a small Monte Carlo simulation of 10 iterations to introduce mismatch. The current traces for every input from one of the resulting Monte Carlo iterations were plotted and is shown in fig. 61. When the bulk-voltages are modulated between their maximum ranges shown in eq. (18) and eq. (19) the resulting modulation in the current is several times larger than the input-dependent variation that is seen between the input combinations.

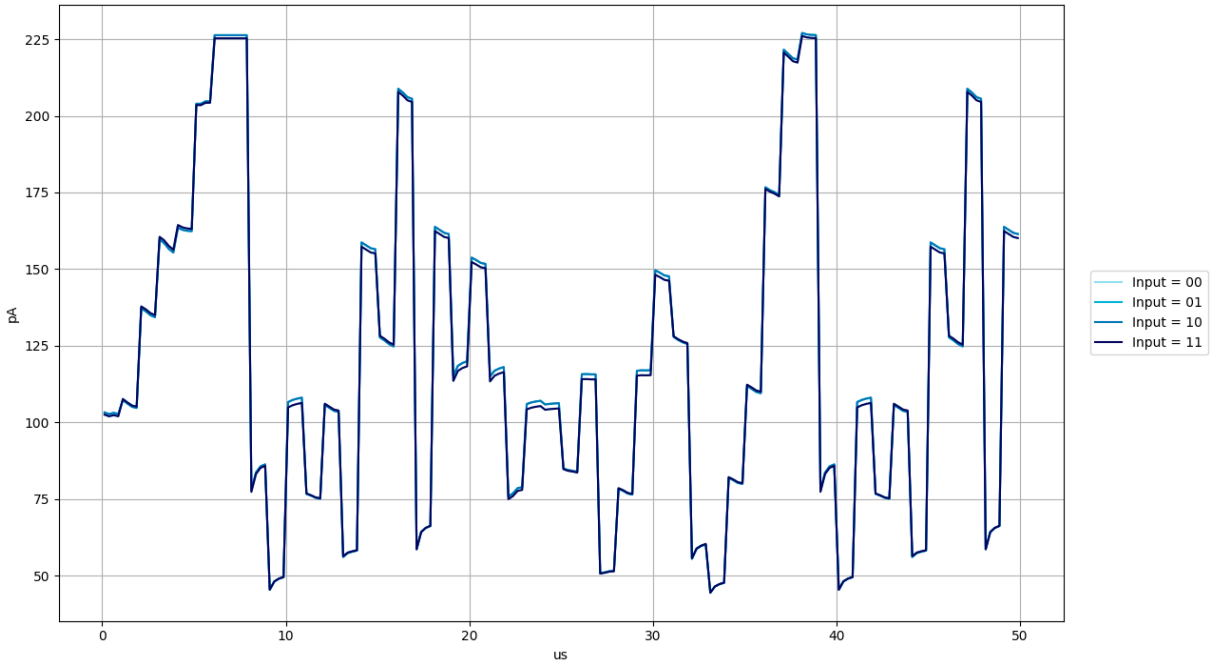

Figure 61: Filtered static current consumption of a ELB NAND gate with bulk-voltage mask

4.3 Experiments on an eight-bit register

One of the main goals with this bulk-voltage mask is that it should be able to operate completely independently and asynchronously in relation to the circuit it is supplying bulk voltages for. Naturally for this to be true, the function of the main circuit can not be altered by the application of the mask. To advance from tests conducted on single logic gates to tests conducted on larger systems, an eight-bit register was constructed from QDPL logic gates like the ones that was discussed in section 3.6. Conducting tests on a register also allows for the exploration of how a changing Hamming weight impacts the circuit's current consumption. The Hamming weight of a register is an important aspect when it comes to PAA as it can be used in Correlation Power Analysis (CPA) attacks, that can be based on either the dynamic power[20] or the leakage power[7], dubbed Leakage-Based Hamming Weight Power Analysis (LHPA) in [9], of the register.

A D flip-flop was constructed out of QDPL NAND gates in an effort to create a register with no data-dependence in its static current, the topology that was chosen for this is shown in fig. 62. This flip-flop topology was chosen for its NAND-only nature, since only two-input QDPL NAND gates were created and thus available for constructing systems. Then eight such D flip-flops were used to construct an eight-bit shift register as seen in fig. 63.

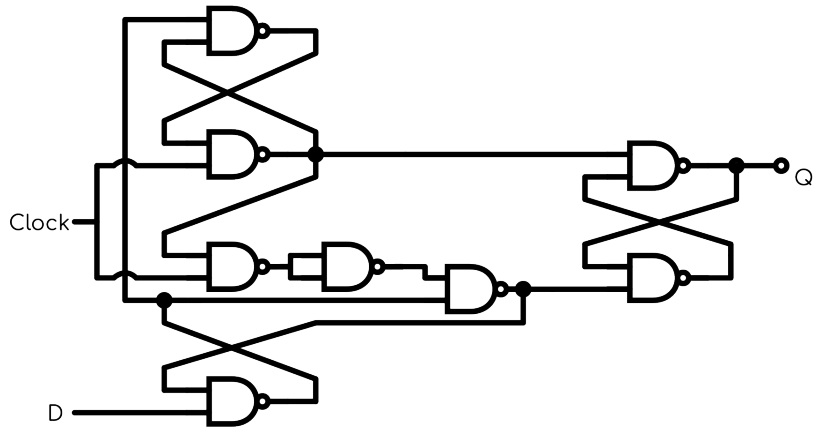

Figure 62: Edge triggered D flip-flop constructed out QDPL NAND gates

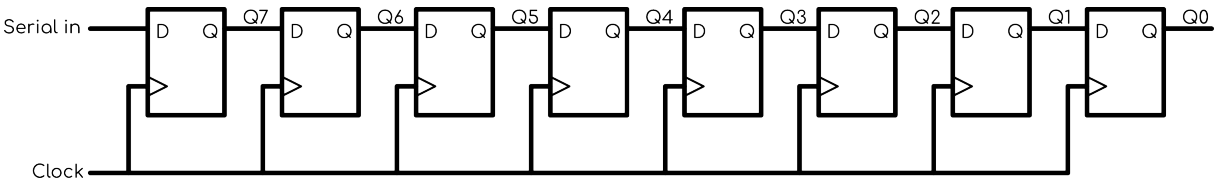

Figure 63: Eight bit shift register made from QDPL D flip-flops

4.3.1 Increasing Hamming weight test

Initially, two transient analyses were carried out on the eight-bit register. One simulation used a LFSR -based generator to actively produce the bulk mask, as previously discussed, while the other had stable bulk voltages of $(V_{nb}, V_{pb}) = (0\,\mathrm{V}, 1.2\,\mathrm{V})$. During this initial test, the serial-in voltage of the eight-bit register was raised, and a logic '1' was clocked through the register. This experiment provided several data points, including an assessment of the QDPL NAND gates' current-hiding capability in a larger system as the Hamming weight of the register increased. The results of this test could also be used to verify that the register circuit exhibited identical

behavior with and without the bulk-voltage mask and that its functionality was not affected by utilizing the mask.

The following plot shows the clock, serial-input and the parallel output voltages of the register while it is operating with an active bulk-voltage mask. The voltages demonstrate that the register functions as expected after the application of the mask. An important detail about these analyses is that the bulk voltage generators were clocked four times faster than the register as a way of controlling that the mask can run independently from the main system.

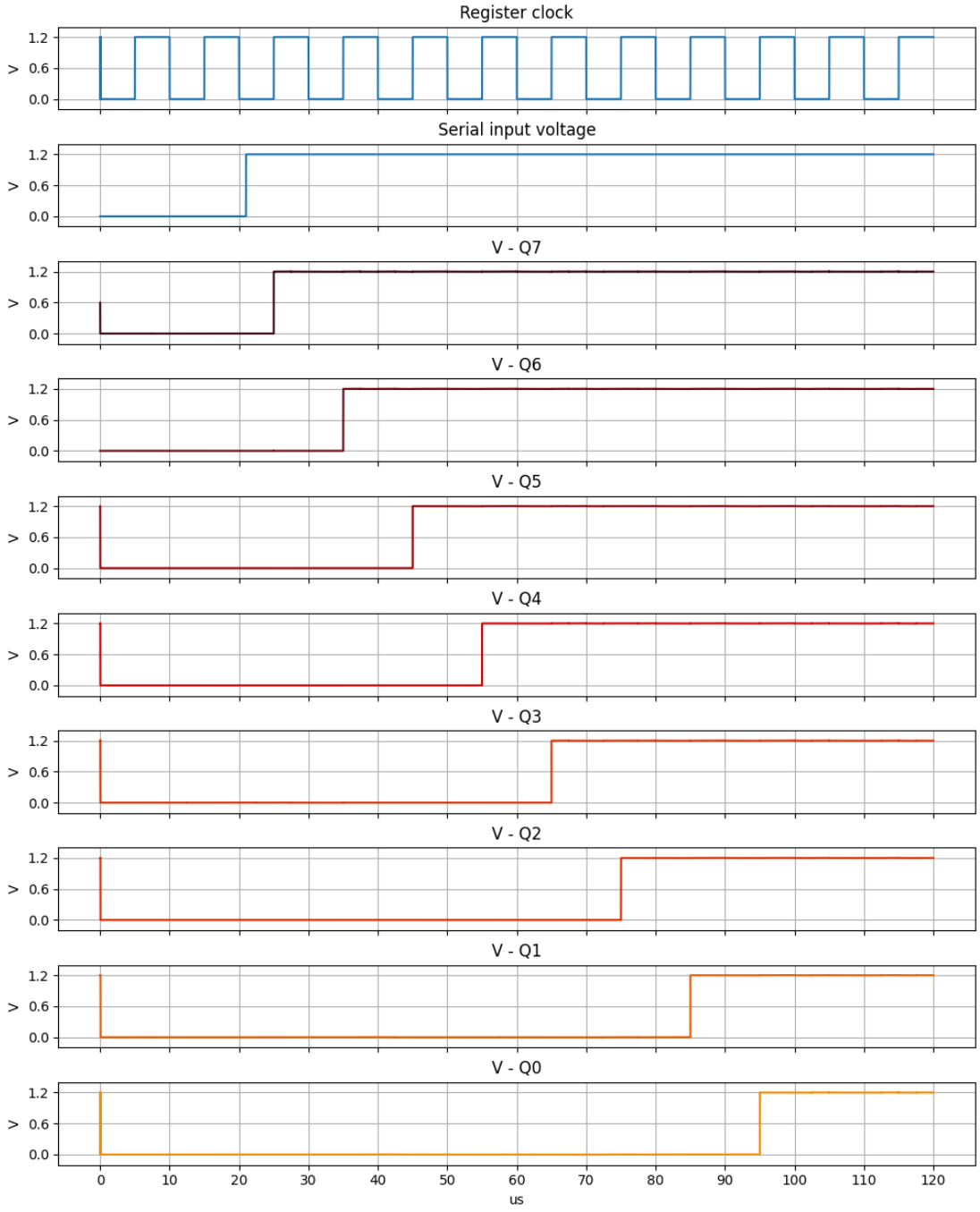

Figure 64: Register voltages while operating with the bulk-voltage mask

The voltages of the bulk mask that were utilized in the simulation were recorded and plotted alongside the respective clock signal of their generators. As shown in fig. 65 and fig. 64, the results of this analysis demonstrated that the mask was able to be randomized by an asynchronous and independent clock signal, with regard to the main system, without disturbing the function of the main system.

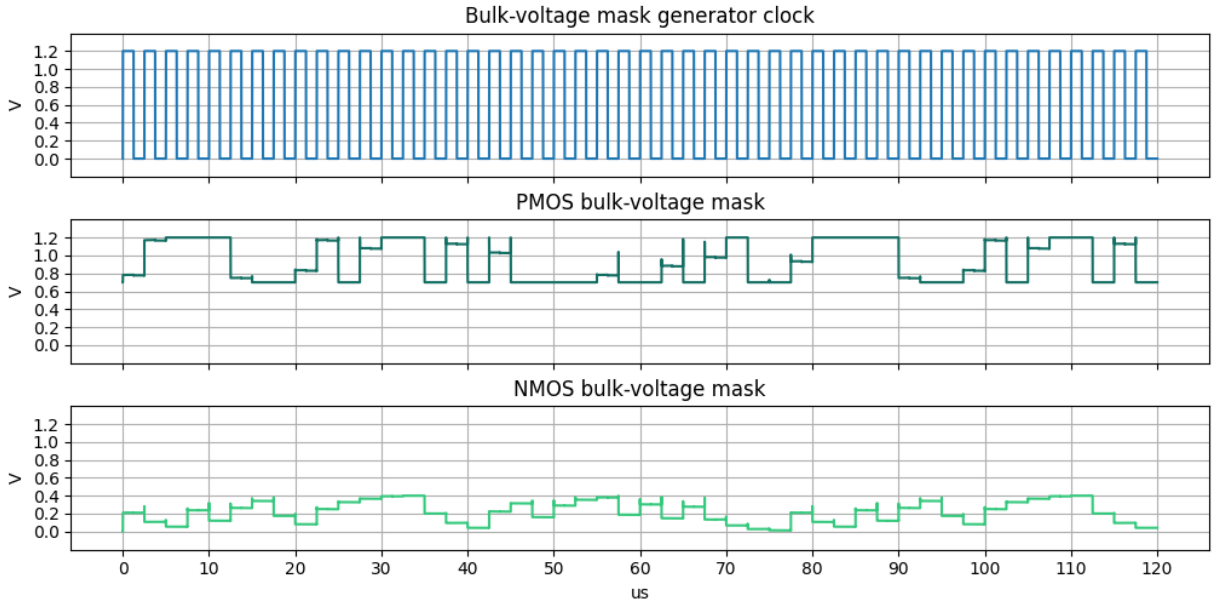

Figure 65: Bulk-voltage mask that was used during the register experiments

The static current consumption of the register was recorded for both analyses and both are depicted in fig. 66. The plot shows the resulting current with an active bulk mask, as well as the current when the mask voltages are statically set to $(0\,\mathrm{V}, 1.2\,\mathrm{V})$.

After further inspection of the static current consumption of the register without the mask, it was observed that the Hamming weight of the register had a negligible impact on its static current consumption. However, there were slight fluctuations in the current due to the clock signal of the register, where the register consumed slightly more current every half-period when the clock signal was low. These fluctuations are believed to stem from the internal input generation of the QDPL NAND gates utilized in the D flip-flops, as discussed in section 3.6. The highest and lowest current values recorded during the transient analysis were 10.4926 nA and 10.4447 nA, respectively, resulting in a maximum unmasked current 1.00459 times higher than the minimum current. The range of the unmasked current can be expressed as:

$$10.4447\,\mathrm{nA} \leq I_{unmasked} \leq 10.4926\,\mathrm{nA} \tag{24}$$

The other trace depicted in fig. 66 illustrates the static current drawn by the register when the bulk mask is active. It is observed that the use of the mask results in large fluctuations in the current consumption. Notably, these fluctuations are uncorrelated to any data present in the register and are solely caused by the independently running bulk mask. As shown in fig. 64, one bit was clocked into the register every 10 µs, while the observed fluctuations occur at a faster rate. The interval of the static current that was observed throughout the trace after the application of the mask is presented eq. (25). The range of currents demonstrates that the highest recorded static current was 2.42 times higher that the lowest recorded current.

$$10.95\,\mathrm{nA} \leq I_{masked} \leq 26.56\,\mathrm{nA} \tag{25}$$

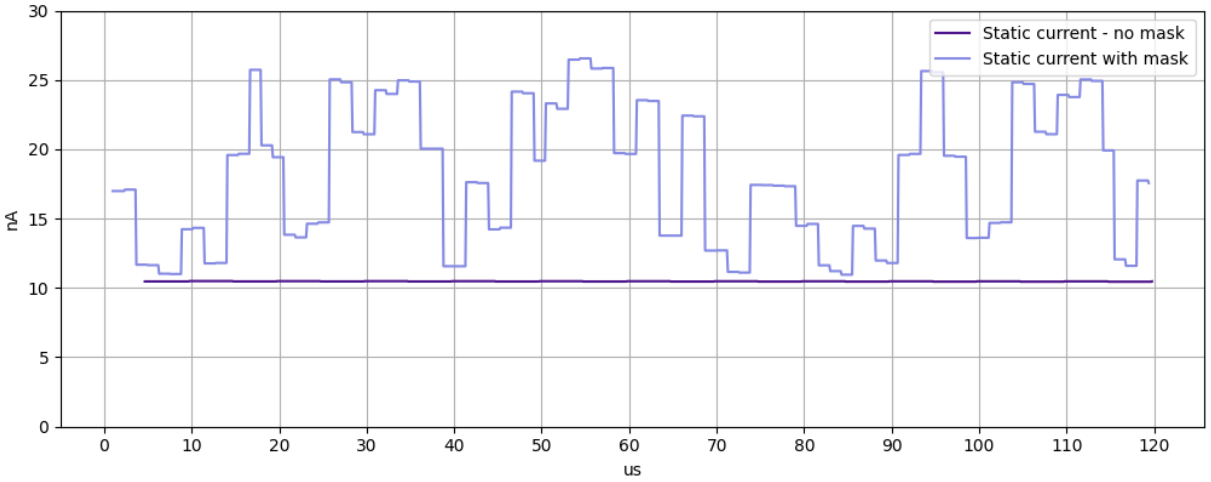

Figure 66: Static current consumption of the eight-bit register with and without the bulk-voltage mask

4.3.2 Arbitrary input sequence test

In the next analyses of the register, an arbitrary bit pattern was supplied through the serial-in port during a simulation period of 30 ms. The clock period for the register was set to 100 µs to let its current settle to a steady state before the next clock edge, then static current was sampled shortly before every rising clock edge. This allowed for the same Hamming weight to be observed in various bit orientations within the register, and the static current was recorded for each. The test was carried out twice, once without the bulk mask applied, and once with the bulk mask activated. There was no need to clock the bulk voltage generators faster than the system clock in this test, as the current was only sampled once per system clock period.

To visualize the results of this test, a scatter plot was created and is shown in fig. 67. The leftmost plot shows the currents that were sampled during the test without the bulk mask. From a visual inspection, the plot demonstrates that there is no obvious correlation between the Hamming weight of the register and its static current. These characteristics are also seen numerically as the correlation coefficient between the current and Hamming weight was calculated to be $r = -0.14$, showing a weak inverse correlation. The entire test showed an interval of static currents like the one described in eq. (26). QDPL demonstrates good qualities in terms of current-hiding characteristics in this test given that the range of the static current is only around 40 pA.

$$10.403\,\text{nA} \leq I_{unmasked} \leq 10.443\,\text{nA} \tag{26}$$

The set of samples had a mean value of $\overline{I}_{\text{unmasked}} = 10.418\,\text{nA}$ and a standard deviation of $\sigma = 8.84\,\text{pA}$. Such a comparatively small standard deviation in relation to the mean current results in a low RSD of only 0.085 %. Given that this set of samples was recorded from an unmasked register, these metrics indicate a low data dependency in the static current.

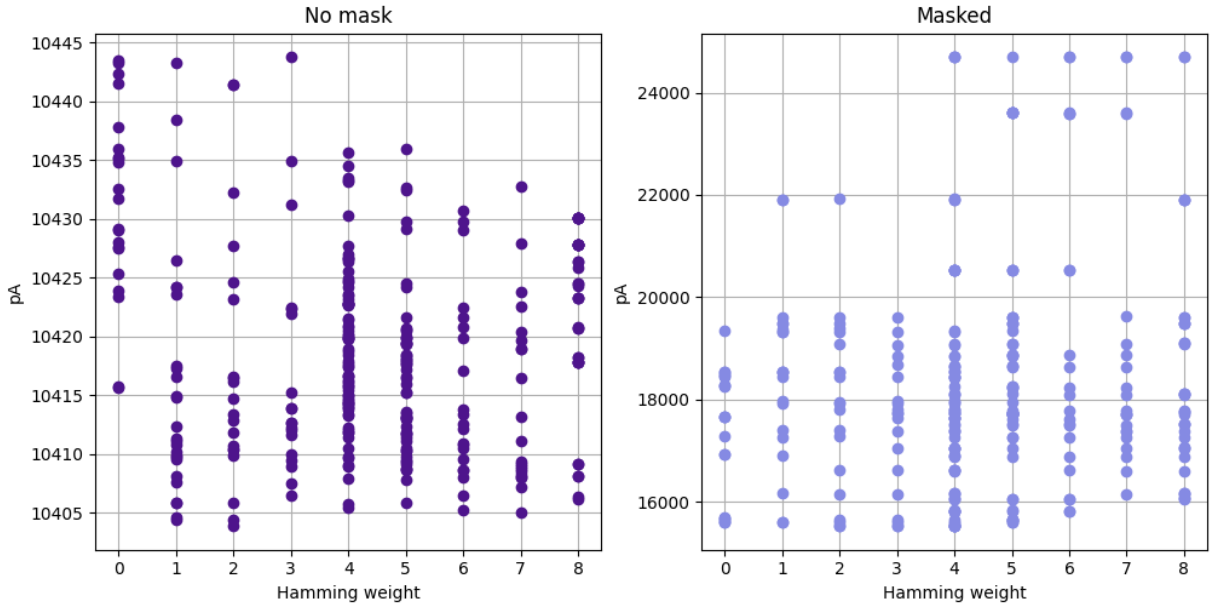

Figure 67: Static current vs Hamming weight in a QDPL register

Since these simulations were conducted with ideal conditions and ideal source for the input signal, the majority of any variations in the unmasked current can be attributed to the changing data in the register. In such a case, the standard deviation of the current can be used as an indication of the data dependency demonstrated by the circuit. However, when metrics are calculated for the register's current after the mask is applied, the standard deviation is no longer useful as an indication of data dependency but rather a measure of variations in the current caused by the mask.

The rightmost plot in fig. 67 shows the currents sampled when the arbitrary-input test was repeated with an active bulk-mask. A significantly wider interval of static currents was observed in this test, and is described by eq. (27). It is possible that this interval could have been even larger, however, the slow clock speed of the bulk-voltage generators, combined with the initial seeds in the LFSR, prevented the setting of every combination of bulk voltages within the simulation time. This is evident from the fact that the lowest current in the interval is higher than that of the unmasked circuit's interval, as seen in eq. (26).

$$15.530\,\text{nA} \leq I_{masked} \leq 24.702\,\text{nA} \tag{27}$$

From a visual inspection of the scatter plot of the masked currents in fig. 67, there is still no obvious correlation between Hamming weight and static current. The correlation coefficient was calculated to be $r = 0.15$, showing that the mathematical correlation is also weak. The set of sampled currents has a mean value of $\overline{I}_{masked} = 18.242\,\text{nA}$, and the standard deviation increased to $\sigma = 2140\,\text{pA}$ after the application of the mask. This results in an Relative Standard Deviation (RSD) of $11.73\,\%$. When these metrics are considered in comparison with those found for the unmasked currents, it is clear that a large amount of data-independent noise can be added by this masking scheme in larger systems, and its effectiveness is not limited to being used only on single logic gates.

4.3.3 Comparison of a corresponding standard CMOS register

As a point of comparison, the arbitrary-input sequence experiment was repeated for an eight-bit register implemented in standard CMOS logic with the same D flip-flop topology as before, shown in fig. 62. This comparison is made to get insight into both the effectiveness of the hiding mechanism of QDPL as well as the impact of the bulk mask on a logic style without any underlying hiding mechanism. The resulting current samples are plotted in fig. 68.

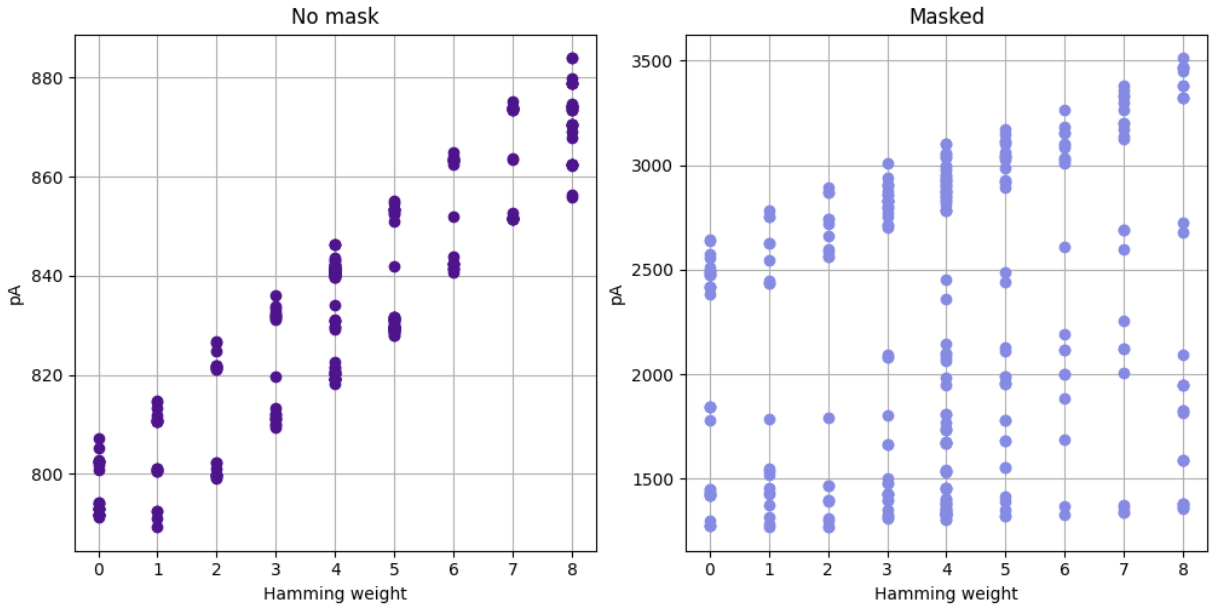

Figure 68: Static current vs Hamming weight in a standard cmos register

The leftmost plot shows the samples that were taken of the unmasked current consumed by the CMOS register. A visual inspection of the plot reveals a strong correlation between Hamming weight and static current is evident. This correlation is further supported by the calculated correlation coefficient of $r = 0.9$. The sampled currents have an interval as described in eq. (28) and a mean value of $\overline{I}_{unmasked} = 834.51\,\text{pA}$. The standard deviation of the set is $\sigma = 23.25\,\text{pA}$, resulting in a RSD of 2.78 %.

$$789.31\,\text{pA} \leq I_{unmasked} \leq 884.02\,\text{pA} \tag{28}$$

The rightmost plot in fig. 68 shows the sampled currents when the test was repeated with an active bulk mask. It is clear that the mask added a large amount of noise to the set of samples, but there is still a visible data dependency in the scatter plot. Even though noise is introduced to the current, the point around which the noise occurs increases linearly with the Hamming weight of the register. It is likely that with enough data points, a regression line could be found by averaging every data point at each Hamming weight. The correlation coefficient was reduced down to $r = 0.22$ after the mask was applied. And the interval of currents was increased to the interval seen in eq. (29).

$$1269.96\,\text{pA} \leq I_{masked} \leq 3514.13\,\text{pA} \tag{29}$$

The mean of the masked samples is $\overline{I}_{masked} = 2213.5\,\text{pA}$ and the standard deviation was increased to $\sigma = 704.44\,\text{pA}$ after the application of the mask, resulting in a RSD of 31.8 %.

4.3.4 Introducing mismatch to the QDPL register

Next, analyses were carried out to observe the impact of mismatch on the data-dependency of the register. For this experiment, logic '1' values were clocked into the register until it was full, followed by clocking in '0' values into the register until it was emptied. This sequence was then repeated several times to create a pattern that resulted in the Hamming weight going through systematic cycles of increase and decrease. A clock period of 100 µs was used to let the current settle to a static state before the next clock edge. The static current was then sampled a little before every rising clock edge to create a trace of the static current during the test sequence.

Before introducing mismatch, the analysis was conducted in an ideal case to establish a baseline. The resulting current trace, along with a plot of the Hamming weight of the register, is presented in fig. 69.

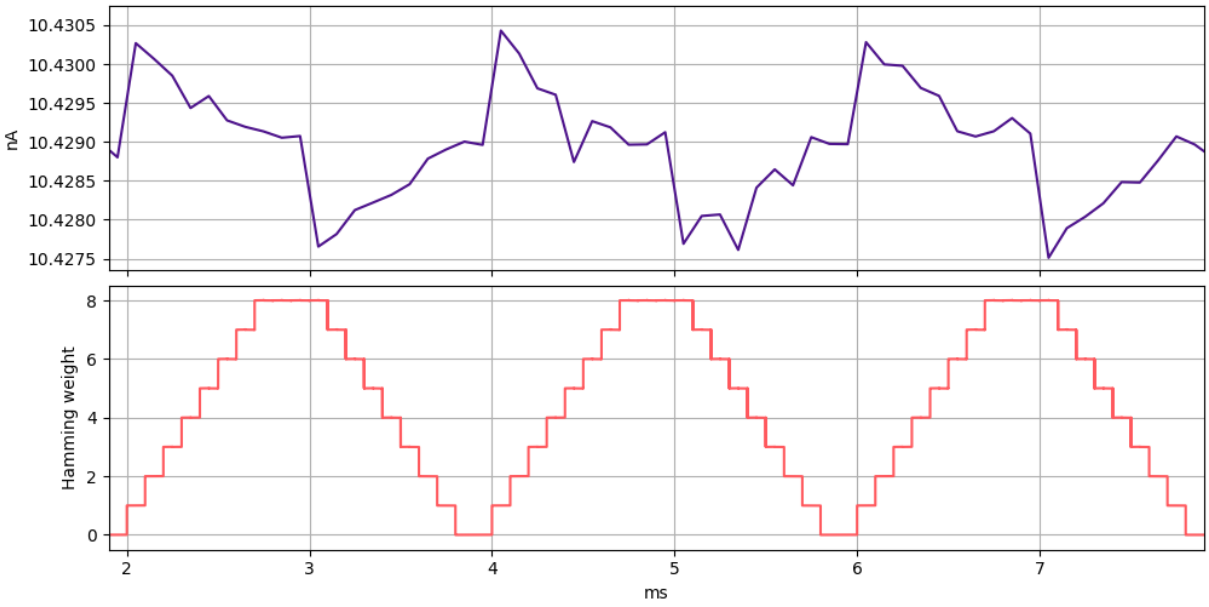

Figure 69: Static current and Hamming weight in a QDPL register from an ideal simulation

The graph shows a small inverse correlation between the static current consumption of the register and its Hamming weight even in an ideal simulation. This correlation is likely due to the imbalances caused by the internal input generation in the QDPL NAND gates that make up the register, as discussed in section 3.6. It is important to note that the difference in static current is only around 3 pA when the highest and lowest recorded currents from the analysis are compared.

When the same test is repeated with mismatch introduced, the correlation between static current and Hamming weight becomes much clearer. To introduce mismatch into the simulation, a Monte Carlo simulation was conducted, and one of the resulting iterations was picked out as an example. In an effort to save time, the first iteration of the Monte Carlo simulation was used as the example in this analysis. The sampled static current is plotted together with the Hamming weight in fig. 70.

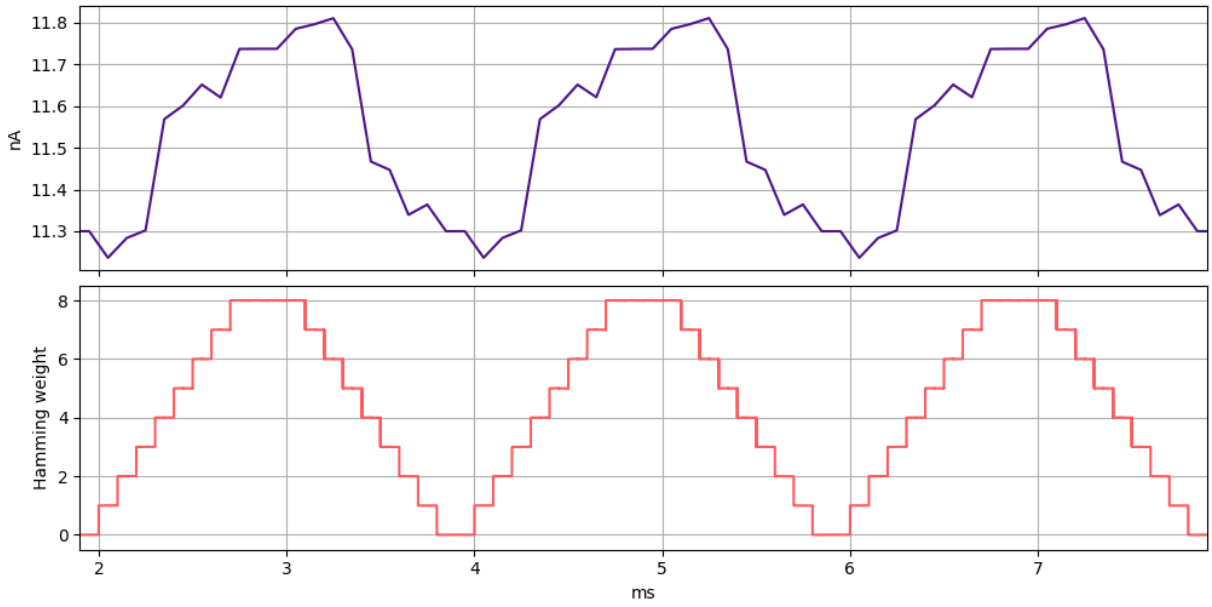

Figure 70: Static current and Hamming weight in a QDPL register with mismatch

From a visual inspection it is clear that there is a much stronger correlation between the static current and the register's Hamming weight in this instance of mismatch. The correlation is no longer inverse and the amplitude change of the current is considerably larger. The difference in current consumption between the Hamming weight of zero and eight is approximately 575 pA. Meaning that in this test, it was found that introducing mismatch lead to a 191.6 times larger difference in static current between the highest and lowest Hamming weight.

Next, the same instance of mismatch was simulated once more but this time with the bulk mask applied. The resulting current consumption is plotted together with the Hamming weight cycles in fig. 71. The bulk-voltage generators was only clocked once per system clock cycle in this simulation since the static current was only sampled once per system clock cycle.

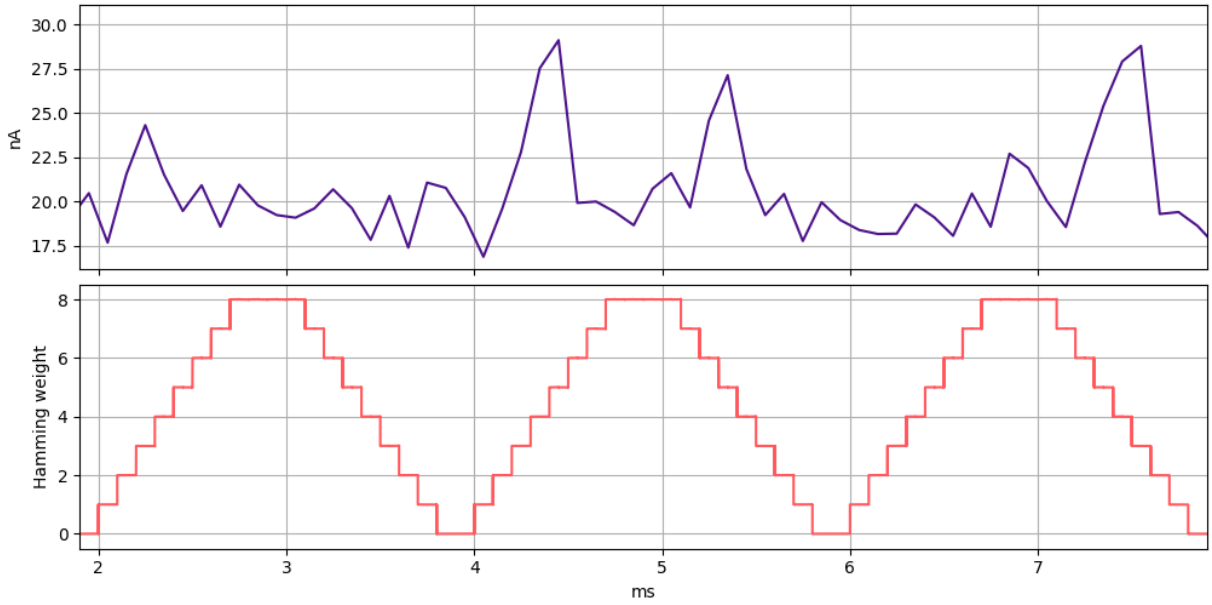

Figure 71: Static current and Hamming weight in a QDPL register with mismatch and bulk mask

After the application of the bulk mask, the static current is seen fluctuating by a maximum value of 13.54 nA. Which means that the mask has the power to modulate the static current by an amplitude that is 23 times larger than the differences caused by the input-dependency alone in this particular instance of mismatch.

The next analysis that was conducted on the register explored the effects of increasing the temperature in the presence of mismatch. To do this, the same Monte Carlo simulation that was used in the previous experiments was repeated while the temperature was increased to 80 °C, and the same iteration of the simulation was picked out for further analysis. The resulting current consumption is plotted in fig. 72.

Increasing the temperature resulted in a drastic increase in data-dependency in the register. This analysis demonstrated that when the register changes from a Hamming weight of zero to eight, the static current increases from 67.54 nA to 73.66 nA. Which corresponds to a difference of 6.12 nA in the register's static current between its highest and lowest Hamming weight. These results show that increasing the temperature of the circuit from 27 °C to 80 °C, amplifies the difference in current between the highest and lowest Hamming weights by a factor of around ten.

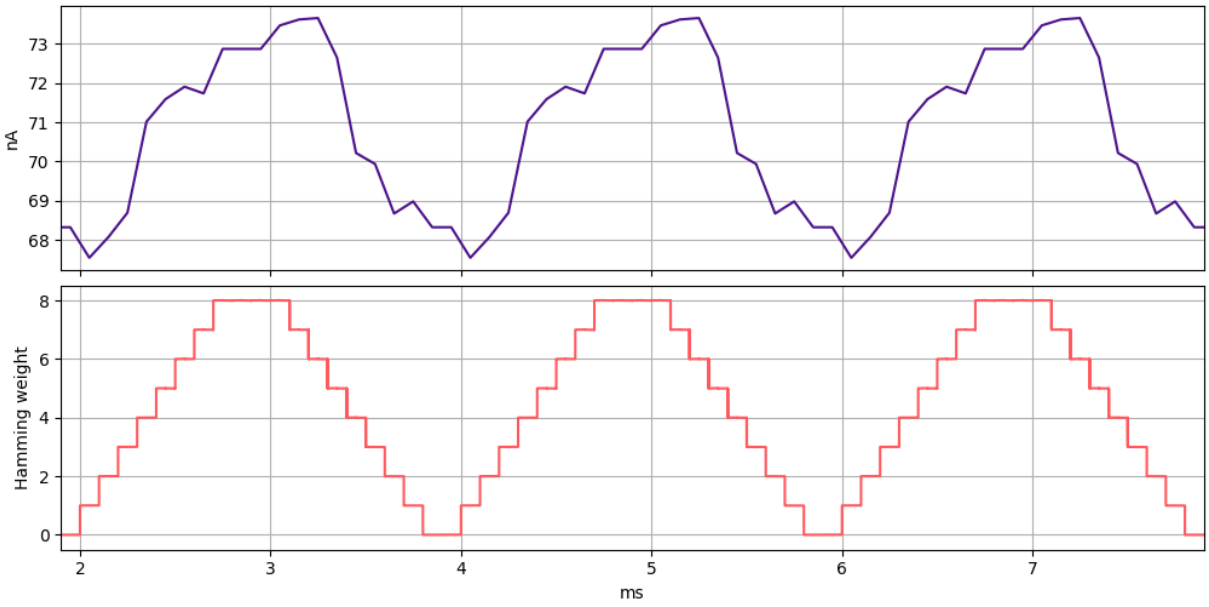

Figure 72: Static current and Hamming weight in a QDPL register with mismatch in 80 °C

One more test was conducted where the same iteration of the Monte Carlo simulation was once again analyzed at the temperature of 80 °C, but this time with the mask applied. An interesting result from this analysis was that after the temperature was increased, the mask's ability to modulate the current increased with it. The difference between the highest and lowest recorded static current from this analysis was 578.06 nA. This result indicates that even though the data-dependency increases with temperature, the mask can still produce noise that is several times larger than the fluctuations due to data-dependency. Specifically, this test showed that the mask can manipulate the current by an amount that is 94.45 times larger than the variations caused by mismatch. The current from this analysis is plotted in fig. 73.

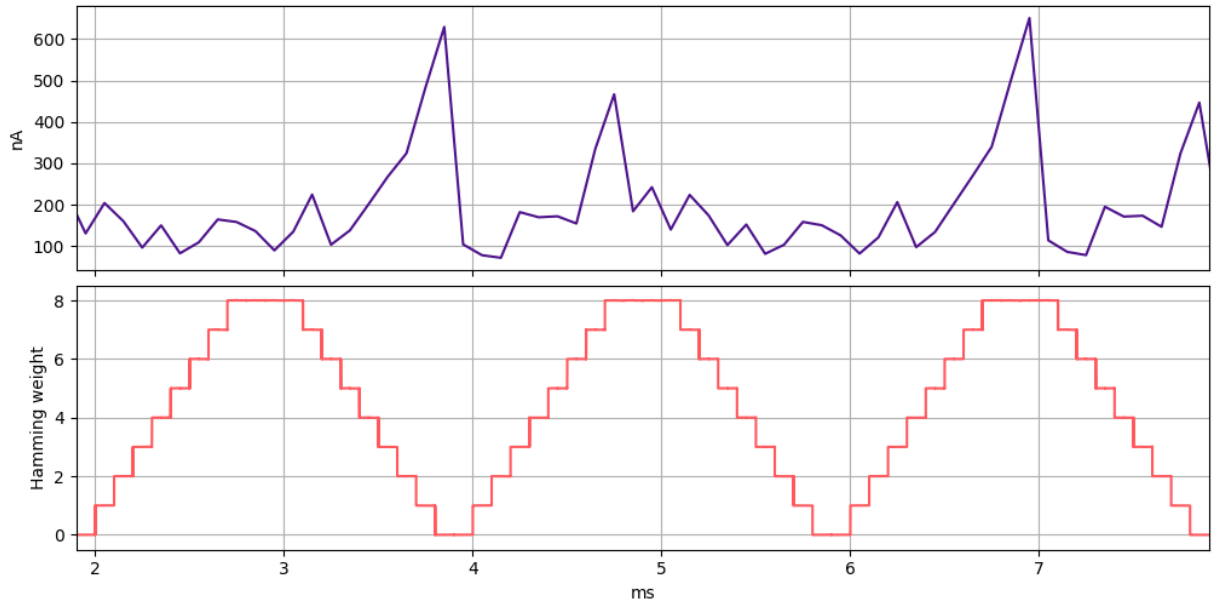

Figure 73: Static current and Hamming weight in a QDPL register with mismatch and mask in 80 °C

4.3.5 Signal averaging calculations

In this work, a lot of emphasis has been placed on the ability of the mask to be randomized independently from the system it is masking. However, this independence makes the noise added by the mask approximate white noise. A consequence of this is that the masked signal can be uncovered through signal averaging. And as pointed out in [7], it is not uncommon for signal averaging to be utilized in LPA attacks. This section will present analyses exploring how the bulk-mask can increase the difficulty of signal averaging.

When measuring a signal in a noise-filled environment, any measurement will include the amplitude of the signal of interest in addition to some amount of noise. If each of the captured signals is synchronized in time, then averaging them causes the result of the calculation to converge towards the amplitude of the signal[21]. This can be applied to the masked current-traces looked at in the previous section to unveil the underlying unmasked current.

First, the Signal-to-noise-ratio (SNR), which in this case is the ratio between the underlying current-trace and the noise in the measurement, is defined as in eq. (30). N is the number of collected current-traces, A is the signal amplitude, and σ^2 is the variance of the noise in the measurement.

$$SNR = N\frac{A^2}{\sigma^2} \tag{30}$$

When this calculation is done with the added noise from the mask, the noise contributions can be split up into the initial noise and the mask noise, as seen in eq. (31).

$$SNR = N\frac{A^2}{\sigma_{init}^2 + \sigma_{mask}^2} \tag{31}$$

Finally, the equation can be solved for N so that it can be used to calculate the number of current-traces needed for a given combination of noise, amplitude, and SNR , as seen in eq. (32).

$$N = \frac{\sigma_{init}^2 + \sigma_{mask}^2}{A^2} \cdot SNR \tag{32}$$

With the formula described in eq. (32), a visualization of the increase in needed traces due to the mask noise can be created. To conduct this analysis, data from the previously discussed QDPL register with mismatch was used. First, the data from the analysis with mismatch in 27 °C seen in fig. 70 was used in the calculations.

When choosing values for the equation, the initial noise was an unknown factor. However, since the point of this analysis was not to give accurate metrics from a specific case, but rather to show the increase in the amount of needed current-traces due to the mask in a more general sense, an arbitrary value of $\sigma_{init}^2 = 1000\,\text{pA}^2$ was chosen. Then to find the amplitude to be used in the formula, the average difference in current between two subsequent values of Hamming weight was calculated and found to be 64.8 pA. This value represents the value that has to be detected through analysis to identify the smallest change in Hamming weight. Then lastly, to find the variance in the noise caused by the mask, the variance in the masked current trace in fig. 71 was calculated and found to be $\sigma_{mask}^2 = 8.096\,\text{nA}^2$.

With these values established, the number of needed current-traces was calculated for SNR values one to five. The plot in fig. 74 shows a graph displaying the increase in needed current-traces for all five different SNR values as the mask noise-variance is increased from $0\,\text{nA}^2$ to $8.096\,\text{nA}^2$. As seen in the plot, to reach a SNR of five when the mask-noise contribution is $8.096\,\text{nA}^2$, an additional 79,220 current-traces are needed in the signal averaging analysis.

An important note is that the noise-variance from the mask can be increased by randomizing the mask-voltages more often. When the clock period of the LFSR based generators was decreased from 100 µs, as was the case in the previously discussed analysis, to 5 µs, the noise-variance from the mask increased to $27.068\,\text{nA}^2$. When the same calculations are repeated with this level of noise, it is found that an additional 873625 current-traces are needed to reach an SNR of five.

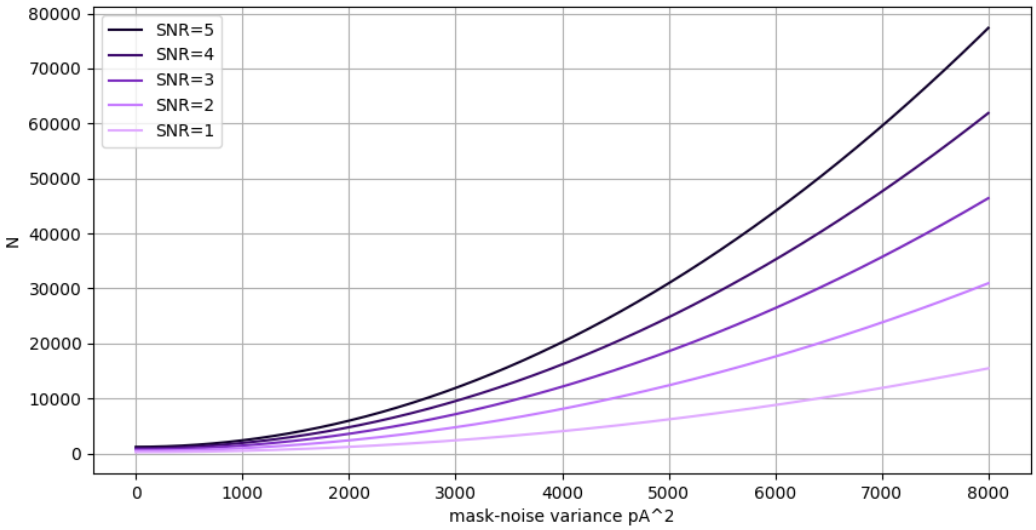

Figure 74: Needed current-traces vs mask noise variance for different values of SNR

The experiment was repeated using the data from the analysis performed at 80 °C, as shown in fig. 72 and fig. 73. Following the same approach to determine the change in current when the Hamming weight is changed by one to be used as the amplitude in eq. (32), it was determined that the average change in the unmasked current between each Hamming weight was 686.0 pA. When calculating the noise-variance in the masked current seen in fig. 73, the large swings in current due to the increased temperature resulted in an massive increase in noise contribution from the mask without decreasing the randomization period from 100 µs. The resulting variance was calculated to be 16 372 nA^2. Using these metrics, the number of current-traces needed was calculated and is presented in fig. 75. Based on this analysis, it was found that 2.848 billion traces would be required to achieve an SNR of five in this case. In fact to achieve an SNR as low as one, 569 million current-traces are required.

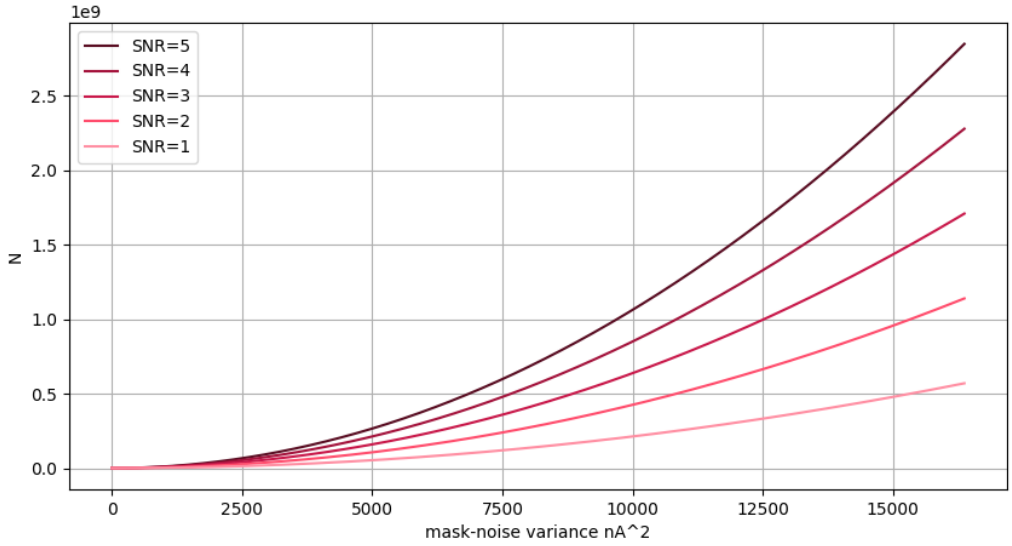

Figure 75: Needed current-traces vs mask noise variance for different values of SNR in 80 °C

5 Discussion

5.1 Transistor lengths

At the beginning of the development and analyses of the different logic styles discussed in this thesis, a transistor length of $2 \cdot L_{min} = 120\,\text{nm}$ was chosen. In other implementations of logic gates, the developer might want to realize the circuit with minimum transistor lengths to minimize the total area usage. In such a case, the effects of mismatch might have a larger impact on the input-dependency in a logic gate. This can be seen in eq. (33)[19], where ΔP is the difference in a given transistor parameter between two devices that are placed a distance of D apart from each other. The term $\sigma^2(\Delta P)$ describes the statistical variance in the device parameter-difference ΔP from random variations caused by mismatch. The relationship shown in this equation indicates that increasing the transistor length L results in a lower variance for a given difference between some parameter of two transistors.

$$\sigma^2(\Delta P) = \frac{A_P^2}{WL} + S_P^2 D^2 \tag{33}$$

5.2 Resource utilization

Currently, improving the resistance of a circuit against side-channel information leakage often includes an increase in its area, power consumption, and overall complexity. This holds true for all countermeasures discussed in this thesis. However, a substantial increase in area and power consumption may render some solutions difficult to implement in practical applications where resources are limited. Therefore, as IC technology progresses, an important area of research is solutions that enhance the security of a circuit while avoiding significant expenses in terms of resources.

5.3 Countermeasures against leakage based attacks

When examining the current literature on gate-level countermeasures, it becomes evident that the majority of the available research has focused on countermeasures against PAA attacks that are based on the dynamic current consumption of a circuit. Comparatively, there are much fewer countermeasures that specifically address LPA attacks. This thesis analyzed one such countermeasure introduced in the literature, namely ELB as presented in [13]. ELB was of special interest among the LPA oriented countermeasures as it was able to achieve zero input-dependence in its static current when tested in ideal simulations in 65 nm CMOS. Therefore, ELB was chosen as a starting point to develop Octuple logic and QDPL that adds further balancing in their dynamic current consumption. Another countermeasure against LPA attacks found in the literature is the logic style called Balanced Static Power Logic (BSPL), which was introduced in [16]. In the introduction of BSPL , the authors of [16] presented simulation results that demonstrated that this logic style could achieve a balanced static current consumption for all input combinations. However, these results were obtained when the logic gates were implemented in 180 nm CMOS technology. When reviewing LPA countermeasures for the work done in this thesis, preliminary experiments were conducted on a BSPL NAND gate in 65 nm CMOS technology, and no such balance was found in the current between input combinations. A plot showing these results is presented in appendix B. This suggests that the BSPL logic style may be obsolete for more modern CMOS technology, and therefore, no further analysis was conducted on this logic style.

Studies have evaluated whether countermeasures against dynamic power based PAA provide resistance against LPA attacks, including research conducted in [22] and [23]. These works analyzed DPA resistant logic styles such as Sense Amplifier Based Logic (SABL)[24], Wave

Dynamic Differential Logic (WDDL)[25], Masked Dual-rail Pre-charge Logic (MDPL)[26], and Three-phase Dual-rail Pre-charge Logic (TPDL)[27], all of which were found to be vulnerable to LPA attacks. Notably, [23] pointed out that the logic style known as Delay-Based Dual-rail Precharge Logic (DDPL)[28] showed promising resistance to LPA attacks. This logic style differs from the other logic styles by encoding output values in the time domain rather than voltage values, suggesting that such encoding schemes may be a promising domain for future research.

The limited number of countermeasures specifically designed to protect against LPA attacks, coupled with the fact that existing countermeasures against dynamic power based PAA provides inadequate protection against LPA attacks, points to the necessity for research in this field and the development of new and effective LPA attack countermeasures.

5.4 Internally multiplied logic gates

In this thesis, all LPA attack-countermeasure logic styles that are analyzed are based on the principle of using a top-level cell made up of a set of duplicated internal logic gates. This principle aims to achieve an equal static current consumption across all input combinations, which can theoretically be achieved when the internals of the cell include 2^n logic gates each processing a unique combination of the input signals, where n is the number of inputs to the top-level cell. This approach guarantees that all variations of the top-level input signals are processed at all times, regardless of the values of the input signals to the top-level cell, theoretically resulting in the same current being drawn at every input combination.

This principle is demonstrated in the logic style called ELB, originally introduced in [13]. The analysis conducted on ELB logic in this thesis is presented in section 3.2. This logic style achieves an equal static current consumption for all input combinations in ideal simulations and even across different temperatures. However, ELB does not achieve a balanced dynamic current when the currents from the different possible transitions of the logic gates were compared. While it was never claimed that ELB can achieve a balanced dynamic current in its introduction, it can still be perceived as a drawback when attempting to create a comprehensive countermeasure. When conducting experiments on ELB logic, the sizing of the transistors in the logic gates were done in such a way as to achieve a symmetrical switching point. It might be possible to tweak transistor sizes to equalize an ELB logic gate's transition currents, but this could potentially result in an unsymmetrical switching point, making the logic gate vulnerable to other attack vectors such as timing attacks. Additionally, if the current balancing relies on precise transistor sizing, the security that the gate aims to achieve could quickly be lost due to uncontrollable factors such as inconsistencies in manufacturing.

Two logic styles were presented in this thesis that expanded on the internal multiplication concept to achieve further balance between the transition currents while keeping the balance in the static current. One of these logic styles is called octuple logic and is presented in section 3.3. This logic style improves on ELB logic by having equal single-input transition currents while keeping the balance in the static current. This logic style accomplishes the additional current balancing while only utilizing standard cells.

QDPL was the other logic style introduced in this thesis. Compared to octuple logic, QDPL reduces the number of internal logic gates from eight to four by using non-standard cells internally. This approach provides every benefit of octuple logic while using fewer transistors per top-level logic gate. Although using non-standard cells adds some complexity to each internal cell, the end result is a less resource-intensive top-level cell compared to octuple logic. By continually processing every input combination through an internal set of logic gates, like ELB logic, QDPL achieves a balanced static current in ideal simulations.

One apparent drawback to all these styles, and internally multiplied logic styles in general, is that

their area is guaranteed to increase by a factor of at least 2^n when compared to standard CMOS logic gates. In some cases, the increase will be even larger, as is the case for octuple logic, where any given octuple logic gate will be at least eight times larger than a corresponding CMOS logic gate. Or in the case of QDPL , the internal logic cells contain two additional transistors compared to an equivalent CMOS logic gate. This means that the internal logic gates already require more area than standard CMOS logic cells, and then four such gates have to be instantiated to realize one QDPL logic gate. Considering all this, it is clear that these logic styles are not well-suited for circuits where the total area is under some constraint.

Ideally, one would want a logic style to have identical dynamic current consumption curves for every transition of the inputs while also having an input-independent static current consumption to consider the hiding mechanism of the logic style to be complete. None of the logic styles presented in this thesis accomplishes this goal. Although octuple logic and QDPL has identical current consumption curves for every single-input transition, there is still a difference in current between the single-input transitions and when both inputs transition simultaneously. The current curves in a double-input transition have roughly twice the peak amplitude compared to the current curve of a single-input transition for all the analyzed logic styles. Further research is required to evaluate the extent to which this difference impacts side-channel information leakage. However, it should be emphasized that each input transition does not result in the same current consumption.

When transient analyses were conducted on the different internally multiplied logic styles to gather data about the given logic style's performance in terms of dynamic current, all three logic styles presented in this thesis required four identical capacitive loads to equally load the internal logic gates and thereby attempt to achieve equal dynamic-current consumption curves. In a real world application this would require both the size of the load and the routing to each of the internal outputs to be identical, which places very difficult, or even impossible, requirements on the design flow of a system utilizing such logic gates. This is clearly a highly unrealistic requirement to expect from every logic gate in a system and a more efficient way of loading the internal gates of these logic styles should be researched.

The performance of the three different internal multiplication-based logic styles were tested while operating with mismatch, and results showed that intra-die mismatch has the potential to introduce input-dependency back into all of the different logic styles that were tested. Given the fact that there was quite a large difference between the absolute value of the static currents consumed by the three logic styles, the metric RSD was used. This metric measures the standard deviation in relation to the mean of a set of values, resulting in a measure of the relative variation within a set. This is a convenient metric to use when the variation within two sets of significantly different sizes are compared. However, it should be emphasized that the absolute value of the variations in the current is an important metric. Signal processing to remove the DC offset could potentially be conducted on the current consumption signals, leaving only the absolute value of the variations behind and rendering the RSD metric obsolete.

5.5 Internal input generation

In the evaluations conducted on various logic styles in this thesis, a distinction was made between the logic style's performance when both the main input signals together with their inverses were supplied with ideal voltage sources versus when the inverses of the input signals were generated internally in the logic cell. Utilizing ideal voltage sources for all four signals allowed the analysis to more easily isolate the current-balancing properties of the logic style in question. However, from a practical point of view, supplying both the main signals and their inverse to every logic gate in a system becomes a burden very quickly when developing a system.

This is why there is a need to generate the inverse of the input signals internally in the logic gate so that only two input signals are required by the logic cell. The type of inverter that is used for this task, and the way the inverters are connected, can have an impact on the information-leakage of the logic gate even if the underlying logic style itself has robust leakage-protection properties. As shown in section 3.6, the choice of logic style for the inverters used in this work is called SDRL and is introduced in [8].

When internal input generation was included in the QDPL NAND gate, some of the current balancing properties that the logic style inherently possesses were lost. Before the implementation of internal input generation, the logic style ensured that the single-input transition currents were equal even during transient analyses as long as the internal logic gates were equally loaded. However, as seen in fig. 51, this is no longer true after internal input generation is implemented. Instead, multiple current traces with varying shapes are observed, as shown in fig. 49. More research is required to determine how these differences impact the feasibility of a successful PAA in realistic scenarios with proper circuit layout and realistic loads on the internal logic gates.

5.6 Bulk mask

A masking scheme was proposed in this thesis where randomized voltages were applied to the bulk terminals of the transistors in a logic gate to randomize the logic gate's power consumption. Promising results were found in terms of how much the current of a circuit could be modulated using this method. In a case where the bulk voltages are generated in a properly randomized fashion, this masking scheme has the possibility to add an extra layer of protection against PAA to logic styles in the hiding category. However, the generation of these voltages is not necessarily trivial and the security provided by a bulk-voltage mask is heavily dependent on the method chosen to generate the bulk voltages. If generated in a deterministic way, the mask could essentially be undone with post-processing. For that reason, the proof-of-concept generators based on LFSR circuits that were used earlier in this thesis are likely not a sufficiently secure method of voltage generation given the pseudo-random nature of an LFSR .

Many alternatives of CMOS True Random Number Generator (TRNG) circuits exist that could potentially be used for the generation of the two mask-voltages. One such TRNG is first described in [29], then later implemented and tested on chip in [30]. This circuit is able to produce numbers at high speeds, which could increase the amount of noise the bulk-mask is able to produce. Furthermore, this TRNG was developed to be used for encryption applications, and analyses to assess the circuit's vulnerability to power analysis were conducted in [31] where it demonstrated robust properties in terms of information leakage. However, it should be noted that the information leakage characteristics were gathered from a TRNG implemented in 0.35 µm CMOS technology, and the circuit would very likely demonstrate vastly different power consumption characteristics if implemented in 65 nm CMOS technology.

Another aspect of the voltage generation that is essential is that it should not be possible to alter or stop the randomization process from the outside of the IC it is working within. If an adversary has the option to stop the randomization of the mask, it could be done at a point where the data-dependency in the current is amplified, potentially simplifying the process of a LPA attack. This is seen in fig. 57 where the mask was tested on a single ELB NAND gate, that if the mask was only added in a static and non-randomized fashion, the result would be an amplified input-dependency in the static current. For that same reason, the randomization of the bulk mask should not be controlled by the system clock of the targeted system. Certain LPA attacks depend on minimizing the noise in power measurements, such as by pausing the system clock to allow the steady-state current to stabilize, in order to gather the intended confidential data more precisely[32, 33]. If the bulk-voltage generators are also clocked by the system clock signal or a derivative of it, one would run into the problem of potentially amplifying the data-dependency in

the current.

However, it is possible to introduce a substantial amount of noise into an adversary's power measurements by using a clock signal for the generators that is not affected by any tampering to the system clock. If, for instance, the generators are signaled for randomization by an internally generated signal that is constantly oscillating as long as the system is powered on, there will always be a layer of data-independent noise on top of the static current, regardless of any manipulation of the system clock.

The security of such a system might benefit from implementing the mask-voltage generators with some countermeasures against side-channel information leakage. The constantly changing mask voltages add a substantial dynamic power component to the system, and further research should be conducted to explore if this potentially results in additional information-leakage through poorly implemented generators.

When employing a bulk-voltage mask one should take care to explore the effects it has on the logic style used in the targeted system. The masking scheme is versatile and offers a simple implementation given that it is simply providing bulk voltages to a circuit, but care should be taken to explore that it is not working to the detriment of the protection offered by the underlying logic style. The ranges of voltages that the mask operates within that was presented in section 4 are not universally applicable and can be adjusted to the needs of any given system. The voltages can also be randomized within a shorter interval to have a lesser impact on the driving capabilities of each transistor, although this will reduce the modulation seen in the current.

As the results of this thesis shows, the bulk-mask is able to provide noise to a circuit with a larger amplitude than the variations due to data-dependency in an average instance of mismatch. However, to thoroughly quantify the effectiveness of the masking scheme, additional tests should be conducted where it is utilized to mask the current of a cryptographic system, and compare the difficulty of a successful PAA with and without the mask.

5.6.1 Similar works

Some countermeasures based on manipulating back-gate voltages of transistors have been presented in the literature before. Namely, the three countermeasures presented in [34], [35] and [36]. All these countermeasures were presented as being implemented in Fully depleted silicon-on-insulator (FDSOI) technology to take advantage of the improved dynamic range these transistors exhibit through back-gate voltage manipulation. The countermeasures in [34] and [36], similarly to the work presented in this thesis, are based on randomization of the back-gate voltages. However, the method of randomization is different in that the voltages are randomized once per encryption process of the cryptographic system receiving the back-gate voltages. Such a solution requires synchronization between the randomization and the encryption process, as opposed to the solution presented in this thesis which is proposed as running independently from the main system which offers more versatility in the implementation of the mask.

Another point of difference is seen in [36] where the two voltages for the NMOS and PMOS transistors are set in a symmetrical fashion so that the back-gate voltage for the PMOS transistors are always the negative of the NMOS back-gate voltage. This biasing scheme was chosen by the authors of [36] so that there would be no difference in the driving driving capabilities of the PMOS and NMOS transistors. For the work that was done in this thesis, it was chosen to not set the voltages in a symmetrical fashion but instead have the two voltages be generated independently from each other to achieve a higher degree of randomization from the larger number of mask-voltage combinations. Through the analysis done in this work, no immediate fault was found in the logic functions of the masked circuits from this biasing scheme. However, further research should be conducted to analyze if an unsymmetrical biasing scheme can make the circuit

vulnerable to timing attacks from the uneven PMOS and NMOS driving capabilities.

The work done in [35] also utilizes back-gate voltages to manipulate a circuit's current consumption, but this countermeasure does not do so to introduce noise but rather to equalize the current consumption between register states.

6 Conclusion

The physical implementation of cryptographic circuits is an issue that requires careful consideration. Careless implementations of such circuits can lead to easily exploitable attack vectors for adversaries to gain access to secret data through side-channel information leakage. This is a very undesirable outcome given the confidential nature of cryptography.

Most of the effort in this thesis has been devoted to one such side-channel, specifically the static power consumption of the circuit. Since the discovery of side-channel channel analysis through power consumption, extensive research has gone into developing countermeasures against PAA that are based on the dynamic power consumption such as DPA . However, as CMOS technology is continually scaled down, its static power consumption is rapidly increasing. The static power has been proven to be an exploitable side-channel through successful LPA attacks in prior research[10], making this a threat of increasing importance.

Multiple LPA attack countermeasures based on the methodology of an internal set of logic gates have been analyzed in this thesis, where two of them, namely octuple logic and QDPL, are introduced for the first time to the best of the author's knowledge. Both of these new logic styles builds on the methodology of a countermeasure introduced in [13], called ELB. ELB demonstrated a balanced static current in ideal simulations, but further analysis showed an imbalance in its transition currents. To expand on the capabilities of ELB logic, octuple logic and QDPL are presented. Both these logic styles keep the balance in static current but achieve additional balance in their single-input transition currents.

Common to all these logic styles is that their balance in static current is lost when mismatch is introduced. Through analysis, it has been found that despite the input-dependency seen in the presence of a mismatch, the variations in currents occur around a point of balance that is common to every input combination. This implies that there are still benefits in using these logic styles even when mismatch is considered, since any variation in current between input combinations will be a deviation from the same point. When contrasted with standard CMOS logic gates, which demonstrate an unbalanced current even in ideal simulations, it is clear that any instance of a logic gate based on an internal set of logic gates has a higher likelihood of having a more balanced static current.

An approach to conceal this data-dependency was introduced in this thesis that introduces input-independent noise through randomized variations in the bulk voltage of the logic gate's internal transistors. Through simulations, this approach showed a consistent ability to introduce noise in the static current with a larger amplitude than the variations caused by mismatch. It is also shown that when the bulk voltages are modulated within proper ranges, they can be randomized without disturbing the function of the circuit they are supplying bulk voltages for. This makes it possible to develop voltage generators that can run independently from the main system to randomize its static current in an asynchronous way in relation to the operation taking place in the system. Given that side-channel analyses that are based on static power consumption often rely on gathering measurements with as little noise as possible, this masking scheme has the power to significantly increase the difficulty of a successful LPA attack by adding a large amount of uncorrelated noise.

References

[1] M. Bellare and P. Rogaway, "Introduction to modern cryptography," *Ucsd Cse*, vol. 207, p. 207, 2005.

[2] D. R. Stinson and M. Paterson, *Cryptography: theory and practice*. CRC press, 2018.

[3] B. Chevallier-Mames, M. Ciet, and M. Joye, "Low-cost solutions for preventing simple side-channel analysis: Side-channel atomicity," *IEEE Transactions on computers*, vol. 53, no. 6, pp. 760–768, 2004.

[4] D. Bellizia, G. Scotti, and A. Trifiletti, "Tel logic style as a countermeasure against side-channel attacks: Secure cells library in 65nm cmos and experimental results," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 11, pp. 3874–3884, 2018.

[5] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*. Springer, 1999, pp. 388–397.

[6] J. Giorgetti, G. Scotti, A. Simonetti, and A. Trifiletti, "Analysis of data dependence of leakage current in cmos cryptographic hardware," in *Proceedings of the 17th ACM Great Lakes symposium on VLSI*, 2007, pp. 78–83.

[7] M. Alioto, L. Giancane, G. Scotti, and A. Trifiletti, "Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 2, pp. 355–367, 2009.

[8] N.-h. Zhu, Y.-j. Zhou, and H.-m. Liu, "A standard cell-based leakage power analysis attack countermeasure using symmetric dual-rail logic," *Journal of Shanghai Jiaotong University (Science)*, vol. 19, pp. 169–172, 2014.

[9] B. Halak, J. Murphy, and A. Yakovlev, "Power balanced circuits for leakage-power-attacks resilient design," in *2015 Science and Information Conference (SAI)*. IEEE, 2015, pp. 1178–1183.

[10] Lang Lin and W. Burleson, "Leakage-based differential power analysis (LDPA) on sub-90nm CMOS cryptosystems," in *2008 IEEE International Symposium on Circuits and Systems*. Seattle, WA, USA: IEEE, May 2008, pp. 252–255. [Online]. Available: http://ieeexplore.ieee.org/document/4541402/

[11] S. Mangard, T. Popp, and B. M. Gammel, "Side-channel leakage of masked cmos gates." in *CT-RSA*, vol. 3376. Springer, 2005, pp. 351–365.

[12] E. Tena-Sánchez, F. E. Potestad-Ordóñez, C. J. Jiménez-Fernández, A. J. Acosta, and R. Chaves, "Gate-Level Hardware Countermeasure Comparison against Power Analysis Attacks," *Applied Sciences*, vol. 12, no. 5, p. 2390, Feb. 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/5/2390

[13] T. Moos and A. Moradi, "Countermeasures against Static Power Attacks: – Comparing Exhaustive Logic Balancing and Other Protection Schemes in 28 nm CMOS –," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 780–805, Jul. 2021. [Online]. Available: https://tches.iacr.org/index.php/TCHES/article/view/8992

[14] Cadence, "Virtuoso system design platform," 2019, [Accessed 02-May-2023]. [Online]. Available: \url{https://www.cadence.com/en_US/home/tools/ic-package-design-and-analysis/ic-package-design-flows/virtuoso-system-design-platform.html/}

[15] BSIM Group, "Bsim4 model," 2000, [Accessed 02-May-2023]. [Online]. Available: \url{http://bsim.berkeley.edu/models/bsim4/}

[16] B. Fadaeinia, T. Moos, and A. Moradi, "Bspl: Balanced static power logic," *Cryptology ePrint Archive*, 2020.

[17] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, 4th ed. USA: Addison-Wesley Publishing Company, 2010.

[18] K. Roy, A. Agarwal, and C. H. Kim, "Circuit Techniques for Leakage Reduction," *A*, p. 17, 2006.

[19] T. Carusone, D. Johns, and K. Martin, *Analog Integrated Circuit Design, 2nd Edition*. Wiley, 2011. [Online]. Available: https://books.google.no/books?id=GeobAAAAQBAJ

[20] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems-CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings 6*. Springer, 2004, pp. 16–29.

[21] R. G. Lyons, "Understanding digital signal processing," 2004.

[22] M. Djukanovic, L. Giancane, G. Scotti, A. Trifiletti, and M. Alioto, "Leakage power analysis attacks: Effectiveness on dpa resistant logic styles under process variations," in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*. IEEE, 2011, pp. 2043–2046.

[23] M. Alioto, S. Bongiovanni, M. Djukanovic, G. Scotti, and A. Trifiletti, "Effectiveness of leakage power analysis attacks on dpa-resistant logic styles under process variations," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 2, pp. 429–442, 2013.

[24] K. Tiri, M. Akmal, and I. Verbauwhede, "A dynamic and differential cmos logic with signal independent power consumption to withstand differential power analysis on smart cards," in *Proceedings of the 28th European solid-state circuits conference*. IEEE, 2002, pp. 403–406.

[25] K. Tiri and I. Verbauwhede, "A logic level design methodology for a secure dpa resistant asic or fpga implementation," in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, vol. 1. IEEE, 2004, pp. 246–251.

[26] T. Popp and S. Mangard, "Masked dual-rail pre-charge logic: Dpa-resistance without routing constraints," in *Cryptographic Hardware and Embedded Systems–CHES 2005: 7th International Workshop, Edinburgh, UK, August 29–September 1, 2005. Proceedings 7*. Springer, 2005, pp. 172–186.

[27] M. Bucci, L. Giancane, R. Luzzi, and A. Trifiletti, "Three-phase dual-rail pre-charge logic," in *Cryptographic Hardware and Embedded Systems-CHES 2006: 8th International Workshop, Yokohama, Japan, October 10-13, 2006. Proceedings 8*. Springer, 2006, pp. 232–241.

[28] M. Bucci, L. Giancane, R. Luzzi, G. Scotti, and A. Trifiletti, "Delay-based dual-rail precharge logic," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 7, pp. 1147–1153, 2010.

[29] F. Pareschi, G. Setti, and R. Rovatti, "A fast chaos-based true random number generator for cryptographic applications," in *2006 Proceedings of the 32nd European Solid-State Circuits Conference*. IEEE, 2006, pp. 130–133.

[30] ——, "Implementation and testing of high-speed cmos true random number generators based on chaotic systems," *IEEE transactions on circuits and systems I: regular papers*, vol. 57, no. 12, pp. 3124–3137, 2010.

[31] F. Pareschi, G. Scotti, L. Giancane, R. Rovatti, G. Setti, and A. Trifiletti, "Power analysis of a chaos-based random number generator for cryptographic security," in *2009 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2009, pp. 2858–2861.

[32] S. M. Del Pozo, F.-X. Standaert, D. Kamel, and A. Moradi, "Side-channel attacks from static power: When should we care?" in *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2015, pp. 145–150.

[33] T. Moos, A. Moradi, and B. Richter, "Static power side-channel analysis of a threshold implementation prototype chip," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*. Lausanne, Switzerland: IEEE, Mar. 2017, pp. 1324–1329. [Online]. Available: http://ieeexplore.ieee.org/document/7927198/

[34] B.-A. Dao, T.-T. Hoang, A.-T. Le, A. Tsukamoto, K. Suzaki, and C.-K. Pham, "Exploiting the back-gate biasing technique as a countermeasure against power analysis attacks," *IEEE Access*, vol. 9, pp. 24 768–24 786, 2021.

[35] K. Palma and F. Moll, "Current balancing random body bias in fdsoi cryptosystems as a countermeasure to leakage power analysis attacks," *IEEE access*, vol. 10, pp. 13 451–13 459, 2022.

[36] ——, "Analysis of random body bias application in fdsoi cryptosystems as a countermeasure to leakage-based power analysis attacks," *IEEE access*, vol. 9, pp. 114 977–114 988, 2021.

A Paper submitted for review at an IEEE conference

Randomized Bulk-Voltages: A Countermeasure to Mask Side-Channel Leakage of CMOS Logic Gates

Randomized Bulk-Voltages: A Countermeasure to Mask Side-Channel Leakage of CMOS Logic Gates

Magnus Amble*, Snorre Aunet†*, Dag T. Wisland*, Kristian G. Kjelgård*
* Department of Informatics, University of Oslo, Norway
†Department of Electronic Systems, Norwegian University of Science and Technology, Norway
magnuamb@uio.no

***Abstract*—Integrated Cryptographic Circuits implemented in Complementary Metal Oxide Semiconductor (CMOS) technology have proven to be vulnerable to static power analysis attacks. In this paper we explore the limitations of a countermeasure against leakage based power analysis attacks called Exhaustive Logic Balancing, which demonstrates complete input-independence in ideal simulations but loses it in the presence of mismatch. We propose a solution to address this input-dependency by introducing input-independent noise through randomized variations in the bulk-voltages of the logic gate's internal transistors. Simulations show that this method can conceal input-dependency in the power consumption by decreasing the signal-to-noise-ratio significantly.**

***Index Terms*—side-channel attacks, leakage power based analysis attacks, countermeasures, cryptographic circuits**

I. Introduction

The power consumption of a cryptographic circuit can be analyzed by an adversary to gain access to secret data that is being processed by the circuit [1]. Such attacks are commonly referred to as Power Analysis Attacks (PAA) and can target different aspects of the circuit's power consumption. In recent years, more attention has been given to PAA that target the static power consumption of the circuit under attack. These attacks, referred to as Leakage Power Analysis (LPA) attacks, are posing a progressively large threat to cryptographic circuits. As Complementary Metal Oxide Semiconductor (CMOS) technology progresses and transistor sizes continue to decrease, the static power consumption of CMOS circuits is becoming an increasing fraction of the total power consumed by the circuit. This trend indicates that the efficiency of LPA attacks is set to increase.

To prevent such attacks, countermeasures can be implemented at different levels of abstraction. However, implementing countermeasures at the system or algorithmic level can be specific to the encryption algorithm used, making the design process more time-intensive and more challenging to automate [2]. Recent research has focused on developing countermeasures at the gate level of a system, which can offer a more generic approach. The methodology behind such gate-level countermeasures is that a theoretical library of leakage-free logic gates can serve as a foundation for building larger systems while maintaining the information-leakage properties of the individual logic gates.

Prior works have split gate level countermeasures up into two broad categories, hiding countermeasures and masking countermeasures. Hiding countermeasures seek to make the current consumption of a logic gate independent of its input signals, for instance by ensuring an equal current consumption for all possible input combinations. Meanwhile, masking countermeasures introduce noise to the power consumption of a logic gate to mask the input-dependency in its power consumption [3].

Manipulating back gate voltages as a countermeasure to PAA has been proposed in [4], [5] and [6]. All these countermeasures are implemented in Fully depleted silicon-on-insulator (FDSOI) technology. Both countermeasures presented in [4] and [5] are based on randomizing the back gate voltages of the transistors, but differs from the work presented in this paper by only being randomized once per encryption process. This inherently requires some form of synchronization between the voltage-randomization and the encryption process. The countermeasure presented in [6] is not based on introducing noise through the back gate terminals, but rather to bias the back gate in way that equalizes the current consumed between different states of a register.

Among the contributions of this paper is an exploration of the properties of a logic style in the hiding category called Exhaustive Logic Balancing (ELB), presented in [7]. It will be demonstrated that an ELB NAND gate consumes a constant static current across all input combinations during testing in ideal simulations. However, in the presence of mismatch, this input-independence is lost. Subsequently, a combination of ELB and a new masking scheme that aims to add randomness to the logic gate's current consumption is introduced. The purpose of this combination is to take advantage of the low degree of information leakage in the underlying logic style then obscure the remaining input-dependency in the presence of mismatch effects with the new masking scheme.

The masking scheme functions by introducing randomness to the current consumption of a logic gate via periodically randomizing the bulk voltage of its internal transistors. It will be argued that if the range within which the voltages are randomized is chosen properly, this masking scheme can be applied to a circuit without disturbing its function. In such instances, new bulk voltages could be generated without regard to any ongoing process in the main system, which implies that numerous new bulk voltages could be generated during a single encryption operation in the case of a cryptographic circuit without disrupting the encryption. Utilizing the mask in this way adds a large amount of data-independent noise to

a circuits static current consumption which can significantly increase the difficulty of a successful LPA attack. To the best of the authors' knowledge, such a masking scheme implemented with standard bulk CMOS has not been presented in earlier literature.

II. Simulations on ELB logic

The primary objective of the ELB logic style is to achieve a static current consumption that is independent of the input. This objective is pursued by constructing a ELB logic gate out of 2^n internal logic gates, where n represents the number of inputs. A unique combination of inverted and non-inverted versions of the ELB input signals is then assigned to each internal logic gate. In fig. 1, the topology of a two-input ELB NAND gate is illustrated. Simulations was conducted to evaluate the performance of ELB logic with and without mismatch. To introduce mismatch to the simulations, Monte Carlo simulations were conducted. When conducting Monte Carlo simulations, a seed value of 12345 was used, with a starting point of 1. The evaluation of the logic style includes metrics presented from both individual iterations of a Monte Carlo simulation as well as averages of all iterations of the simulation. In the event that a singular iteration of the Monte Carlo simulation is selected for further analysis, iteration number five is arbitrarily chosen.

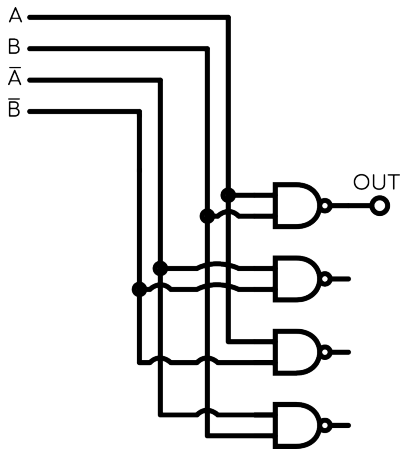

Fig. 1. ELB NAND gate topology

In [7], the ELB topology was presented with inverters internally in the logic gate so that only two input signals had to be supplied to the ELB logic gate and their inverse would be generated by the inverters. However, in such a configuration, the wrong choice of inverter topology can disturb the balance in static current that is achieved by the underlying logic style. In this work, such internal inverters were not used, and it was instead chosen to supply both the original input signals and their inverses from ideal voltage sources as four separate signals, as shown in fig. 1. This choice was made to isolate the performance of the core concept of the ELB logic style itself from any potential variations in current consumption that might arise from the chosen method of input signal generation.

In ideal simulations using the aforementioned input signal configuration, ELB logic successfully achieves a balanced and input-independent static current consumption. However, with the introduction of mismatch into the simulation, this balance is compromised, potentially leading to significant input-dependency in power consumption of the logic gate.

The masking scheme outlined in this paper addresses this issue by adding a layer of input-independent noise to the current consumption of a logic gate through periodically randomizing the body bias of the logic gate's internal transistors. Two separate voltages are needed to implement the mask, one voltage shared by all PMOS transistors and another voltage shared by all NMOS transistors.

For the mask to be able to run independently from the main system in which the logic gate operates, the bulk-voltages have to only be randomized within a range that does not impede the logic gate's operation. The ranges used in this work were determined by conducting parametric sweeps on both bulk voltages while the logic gate was taken through different transitions, then choosing the voltages that did not cause any impairment in its function. Based on this analysis, a range of voltages was selected that resulted in a uniformly distributed modulation of the logic gate's static current between the two voltages. The voltage ranges found to achieve this were:

$$0\,\mathrm{V} \leq V_{N-bulk} \leq 0.4\,\mathrm{V} \tag{1}$$

$$0.8\,\mathrm{V} \leq V_{P-bulk} \leq 1.2\,\mathrm{V} \tag{2}$$

A parametric sweep of the bulk voltages was then conducted within these ranges, and the resulting static current of an ELB NAND gate was recorded at every point and is plotted in fig. 2.

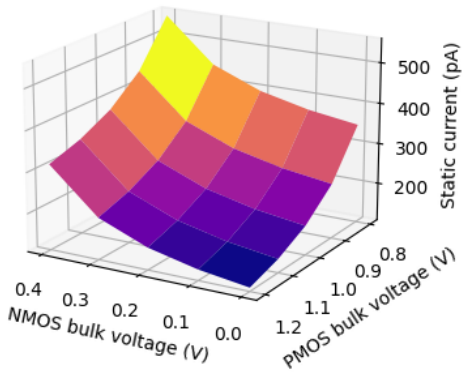

Fig. 2. Modulation of static current from the bulk mask

The plot in fig. 2 illustrates that by utilizing the previously specified bulk-voltage ranges, the mask can effectively modulate the static current of the logic gate within a

range of $116.41\,\mathrm{pA}$ to $550.82\,\mathrm{pA}$. This results in a ratio of $I_{max} = 4.73 \cdot I_{min}$ between the highest and lowest possible current values. In addition, the plot displays that the minimum current occurs at the normal bulk biasing conditions of $(V_{N-bulk}, V_{P-bulk}) = (0\,\mathrm{V}, 1.2\,\mathrm{V})$ and the maximum current is observed when both bulk voltages are tuned to the opposite end of their respective ranges, which is $(0.4\,\mathrm{V}, 0.8\,\mathrm{V})$.

To illustrate the impact of mismatch on the static current of an ELB NAND gate, a brief Monte Carlo simulation was performed to generate instances of the circuit's behavior in the presence of mismatch. For each iteration of the Monte Carlo simulation, the static current of the NAND gate was recorded for each possible current input combination. A particular iteration from the results was selected and is presented in fig. 3.

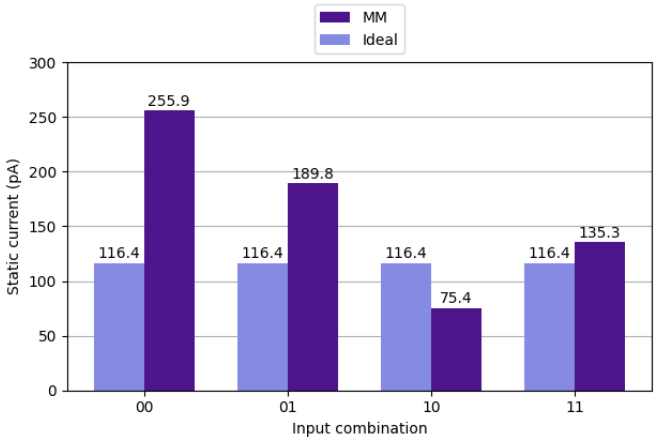

Fig. 3. Static currents of an ELB NAND gate with and without mismatch

As shown in the plot, the static current consumption that was previously identical for all input combinations, now has an interval as given in eq. (3). The resulting set of currents has a mean value of $\overline{I}_{static} = 164.1\,\mathrm{pA}$ and a standard deviation of $\sigma = 66.7\,\mathrm{pA}$. This yields a relative standard deviation of $40.6\,\%$. Earlier works such as [8], have employed standard deviation to indicate the degree of input-dependency in a current set. Based on this approach, the metrics presented above suggest a noteworthy level of input-dependency in this specific instance of mismatch.

$$75.4\,\mathrm{pA} \leq I_{static} \leq 255.9\,\mathrm{pA} \tag{3}$$

Input-dependency of this magnitude is not necessarily a rare occurrence in situations with mismatch. To demonstrate this, four ELB NAND gates were instantiated in a test bench where each gate received one combination of the input signals. The four resulting currents were then treated as one complete set of currents per input combination, with which different metrics could be calculated. Then a Monte Carlo simulation of 5000 iterations was conducted on this test setup.

The same metrics as was presented for the single instance of mismatch were calculated for all 5000 iterations of the simulation. To determine the interval, the average of the lowest currents from all four ELB NAND gates was first calculated, followed by the average of the highest currents. This analysis yielded a much larger interval of currents, as described in eq. (4). The average standard deviation of all 5000 sets was $\sigma = 45.3\,\mathrm{pA}$, while the average mean of all sets was $\overline{I}_{static} = 164.1\,\mathrm{pA}$. This results in an average relative standard deviation of $30.0\,\%$. Taken together, these metrics suggest that a significant amount of input dependency is to be expected when considering mismatch.

$$50.0\,\mathrm{pA} \leq I_{static} \leq 1373.2\,\mathrm{pA} \tag{4}$$

Next, the same example instance of mismatch that was depicted in fig. 3 was analyzed again, but this time with the bulk mask voltages that results in the largest modulation of the static current applied. From fig. 2 these voltages are shown to be:

$$(V_{N-bulk}, V_{P-bulk}) = (0.4\,\mathrm{V}, 0.8\,\mathrm{V}) \tag{5}$$

This test serves as an illustration of the upper limit of noise that can be introduced for each input combination. The corresponding static currents are depicted in fig. 4, where the darker segment of each bar represents the unmasked current and the lighter segment represents the current after applying the described mask. By modulating the mask voltages, any degree of noise within the lighter segment of each bar can be introduced to the current consumption.

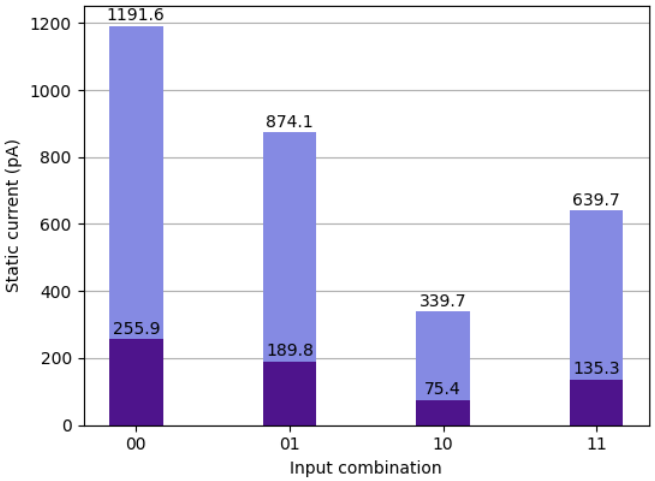

Fig. 4. Static currents of an ELB NAND gate with and without mask

Upon visual inspection, it is evident that the magnitude of the noise added by the bulk mask exceeds the magnitude of the input dependency caused by mismatch. This observation is supported numerically. Specifically, if the ratio between the highest and lowest unmasked current is calculated, a ratio of approximately 3.4 is found, indicating that the largest variation in static current that can arise due to input dependency in this instance of mismatch is a multiplication factor of 3.4. When the ratio between the masked current and the unmasked current is calculated for any input combination, it is found

to be approximately 4.6, indicating that the use of a bulk mask can introduce a higher degree of variation in the static current than the variations caused by mismatch. It should be noted that these calculations were performed using a single instance of mismatch, but this instance is fairly representative when compared the average metrics from the Monte Carlo simulation with 5000 iterations that was presented earlier.

Given the uncorrelated nature of the mask's noise, it is likely that an adversary can remove it through signal averaging with enough sampled power traces. Although, the mask has the power to increase the number of traces needed to achieve a given signal to noise ratio. This can be seen in eq. (6), eq. (7) and eq. (8), where SNR is the signal to noise ratio, N is the number of power traces, A is the signal amplitude and σ^2_{init} and σ^2_{mask} is the variance of the initial noise and the mask noise respectively.

If the signal to noise ratio is given by eq. (6), adding the noise contributions from the mask results in eq. (7). When solving for N as in eq. (8), it can be seen that the number of traces needed to achieve a given signal to noise ratio is exponentially proportional to the noise from the mask.

$$SNR = N\frac{A^2}{\sigma^2} \tag{6}$$

$$SNR = N\frac{A^2}{\sigma^2_{init} + \sigma^2_{mask}} \tag{7}$$

$$N = \frac{\sigma^2_{init} + \sigma^2_{mask}}{A^2} \cdot SNR \tag{8}$$

To visualize the increase in power traces needed to reach the same SNR after the noise from the mask is introduced, the plot in fig. 5 was made. To find the level of variance possible from this masking scheme, a random sequence of 200 numbers within the range $116\,\mathrm{pA}$ to $550\,\mathrm{pA}$ was generated. This sequence mimics a current being randomly modulated within the range seen in fig. 2. The variance was calculated for a few such sequences which all ended up with a value of $\approx 15\,000\,\mathrm{pA}^2$. Then the amplitude was set to the average mean value that was found in the Monte Carlo simulation which was $\approx 164\,\mathrm{pA}$, and an arbitrary value of $\sigma^2 = 1000\,\mathrm{pA}^2$ was set for σ^2_{init}. The plot shows the increase in the number of needed traces for five different values of SNR, all showing an increase of over 10000 traces except the plot when $SNR = 1$. The effects of the mask is much greater when calculated for lower signal amplitudes, where a signal amplitude of $A = 50\,\mathrm{pA}$ results in 452000 needed traces to reach an SNR of 5.

All simulations were performed using the Cadence Virtuoso System Design Platform. The simulated circuits were implemented in $65\,\mathrm{nm}$ bulk CMOS technology. All simulated circuits used a supply voltage of $1.2\,\mathrm{V}$. The minimum length of $L = 60\,\mathrm{nm}$ was used for all transistors. The width of the transistors was chosen by first setting all NMOS transistors to the minimum width of $W_N = 120\,\mathrm{nm}$, then choosing the width of all PMOS transistors such that the logic gate implementation achieved a symmetric switching point. The PMOS widths was set to $W_p = 270\,\mathrm{nm}$ to achieve this symmetry.

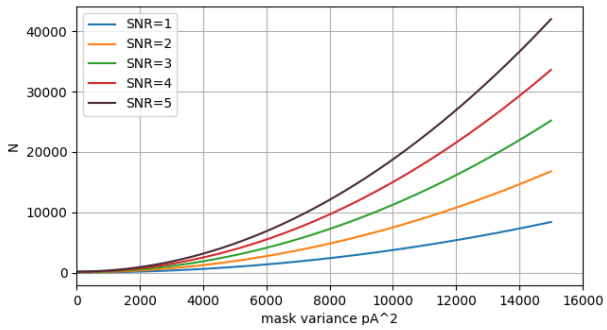

Fig. 5. Number of traces needed to reach a given SNR vs mask noise variance

III. Discussion

The effectiveness of the proposed bulk-voltage mask is highly dependent on the randomness of the bulk-voltages. If the generation of the bulk-voltages is done in a deterministic way, the mask could be undone by an adversary. Additionally, it is of utmost importance to ensure that the mechanism of clocking the mask for randomization cannot be accessed externally by a potential adversary. If such access were available, the bulk mask could be stopped at a point that amplifies the input-dependency of the logic gates. The proposed solution to circumvent this issue in this paper is be to implement an internal mechanism of oscillation that acts as a clock signal for the circuitry responsible for randomizing the bulk mask, that runs independently from the main system. The mask can be randomized as frequently as necessary, and the frequency of randomization does not have to be constant. In such case, a developer is left with a high degree of freedom in implementing the masking scheme.

IV. Conclusion

This paper presents a methodology to improve limitations to the usefulness of current-matching logic styles for resilience towards LPA attacks. It's shown that the recently published ELB fails to provide equal leakage currents under the inevitable presence of mismatch in CMOS technologies. Mismatch leads to undesirable input-dependent leakage currents, and thus vulnerability to LPA attacks. We propose a new method of randomly changing the voltages on the bulk nodes of transistors to mask the dependencies of leakage from mismatch and demonstrate through simulations that this should improve the circuit's robustness against undesirable leakage-based DPA attacks.

REFERENCES

[1] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*. Springer, 1999, pp. 388–397.

[2] E. Tena-Sánchez, J. Castro, and A. J. Acosta, "A methodology for optimized design of secure differential logic gates for dpa resistant circuits," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 4, no. 2, pp. 203–215, 2014.

[3] E. Tena-Sánchez, F. E. Potestad-Ordóñez, C. J. Jiménez-Fernández, A. J. Acosta, and R. Chaves, "Gate-level hardware countermeasure comparison against power analysis attacks," *Applied Sciences*, vol. 12, no. 5, p. 2390, 2022.

[4] K. Palma and F. Moll, "Analysis of random body bias application in fdsoi cryptosystems as a countermeasure to leakage-based power analysis attacks," *IEEE access*, vol. 9, pp. 114 977–114 988, 2021.

[5] B.-A. Dao, T.-T. Hoang, A.-T. Le, A. Tsukamoto, K. Suzaki, and C.-K. Pham, "Exploiting the back-gate biasing technique as a countermeasure against power analysis attacks," *IEEE Access*, vol. 9, pp. 24 768–24 786, 2021.

[6] K. Palma and F. Moll, "Current balancing random body bias in fdsoi cryptosystems as a countermeasure to leakage power analysis attacks," *IEEE access*, vol. 10, pp. 13 451–13 459, 2022.

[7] T. Moos and A. Moradi, "Countermeasures against static power attacks:–comparing exhaustive logic balancing and other protection schemes in 28 nm cmos–," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 780–805, 2021.

[8] L. Lin and W. Burleson, "Leakage-based differential power analysis (ldpa) on sub-90nm cmos cryptosystems," in *2008 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2008, pp. 252–255.

B Results of preliminary experiments on BSPL logic

Figure 76 shows a plot of the static currents of a BSPL NAND gate, implemented in 65 nm CMOS technology, recorded during the preliminary experiments that was done during the research phase of this thesis. The static currents were found through the DC analysis procedure explained in section 2.4.1.

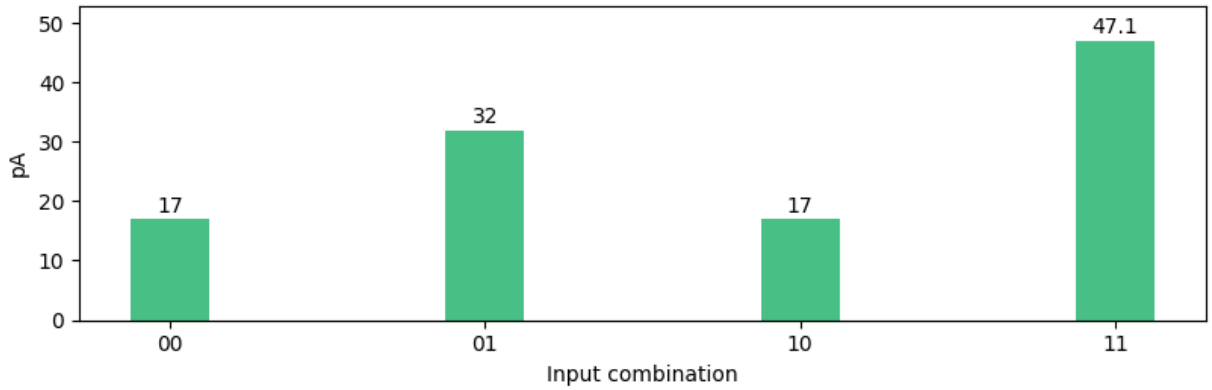

Figure 76: Static current consumption of a BSPL NAND gate

Figure 77 shows a recreation of the schematic of a BSPL NAND gate that the authors of [16] presented in their introduction of the BSPL logic style.

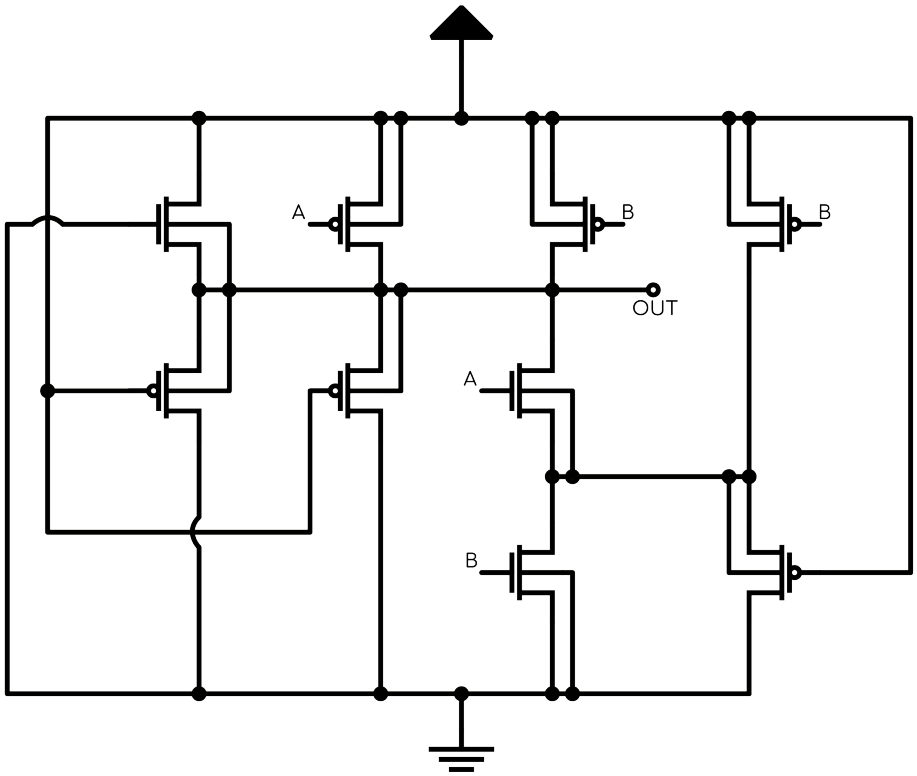

Figure 77: Schematic of a BSPL NAND gate