BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF MICROELECTRONICS

ÚSTAV MIKROELEKTRONIKY

ANALYSIS AND DESIGN OF DECOUPLING CAPACITORS IN THE 65 NM CMOS DIGITAL STANDARD LIBRARY

ANALÝZA A NÁVRH BLOKOVACÍCH KONDENZÁTORŮ V DIGITÁLNÍ STANDARDNÍ KNIHOVNĚ TECHNOLOGIE CMOS 65 NM

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR
AUTOR PRÁCE

Radek Kučera

SUPERVISOR
VEDOUCÍ PRÁCE

Ing. Vojtěch Král

BRNO 2024

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Mikroelektronika a technologie**

Ústav mikroelektroniky

Student: Radek Kučera ***ID:*** 240855

Ročník: 3 ***Akademický rok:*** 2023/24

NÁZEV TÉMATU:

Analýza a návrh blokovacích kondenzátorů v digitální standardní knihovně technologie CMOS 65 nm

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s digitálními standardními knihovnami v technologii CMOS 65 nm. Zvláště se zaměřte na blokovací kondenzátory v souvislosti s digitálně standardní knihovnou a jejich použitím na čipu. Vysvětlete účel a princip blokovacích kondenzátorů a charakterizujte jejich možné vlastnosti. Prozkoumejte různé topologie blokovacích kondenzátorů a navrhněte vlastní varianty, které následně porovnáte v jednoduché simulaci. Na základě těchto porovnání vyberte blokovací kondenzátory pro hlubší analýzu. Pro vybrané kondenzátory vytvořte layout a pomocí parazitní extrakce a detailní simulace zkoumejte jejich chování a vlastnosti. Prioritně se zaměřte na kapacitu vzhledem k ploše buňky, rychlost odezvy a s tím spojenou maximální frekvenci a svodový proud. Monitorujte tyto parametry v závislosti na teplotě, napětí a procesních variacích. Na závěr zhodnoťte všechny analyzované blokovací kondenzátory a doporučte nejvhodnější varianty pro začlenění do digitální standardní knihovny.

DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce.

Termín zadání: 5.2.2024 ***Termín odevzdání:*** 30.5.2024

Vedoucí práce: Ing. Vojtěch Král

doc. Ing. Pavel Šteffan, Ph.D.
předseda rady studijního programu

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

Bachelor's Thesis

Bachelor's study program **Microelectronics and Technology**

Department of Microelectronics

Student: Radek Kučera ***ID:*** 240855

Year of study: 3 ***Academic year:*** 2023/24

TITLE OF THESIS:

Analysis and design of decoupling capacitors in the 65 nm CMOS digital standard library

INSTRUCTION:

Familiarize yourself with digital standard libraries in 65 nm CMOS technology. Specifically, focus on decoupling capacitors in relation to the digital standard library and their application on chips. Explain the purpose and principle of decoupling capacitors and characterize their possible properties. Investigate various topologies of decoupling capacitors and propose your own variants, which you will then compare in a simple simulation. Based on these comparisons, select decoupling capacitors for deeper analysis. For the selected capacitors, create a layout and examine their behavior and properties using parasitic extraction and detailed simulation. Focus primarily on the capacitance relative to the cell area, response time, associated maximum frequency, and leakage current. Monitor these parameters depending on temperature, voltage, and process variations. Finally, evaluate all analyzed decoupling capacitors and recommend the most suitable variants for integration into a digital standard library.

RECOMMENDED LITERATURE:

Podle pokynů vedoucího práce.

Date of project specification: 5.2.2024 ***Deadline for submission:*** 30.5.2024

Supervisor: Ing. Vojtěch Král

doc. Ing. Pavel Šteffan, Ph.D.
Chair of study program board

ABSTRACT

This work deals with the design of decoupling capacitors in the form of CMOS transistors for stabilizing the supply voltage in a digital standard library, focusing on 65 nm technology. The introduction provides an overview of CMOS technology and a description of MOS structures. The analysis compares four topologies of decoupling capacitors (NMOS, PMOS, CMOS, Cross-Coupled) and identifies Cross-Coupled as the best choice. The last part focuses on the design and optimization of the layout of decoupling capacitors. Four different layouts were created, optimized for capacity, quality factor, leakage current, and a compromise between these factors, to be integrated into standard digital libraries according to specific applications.

KEYWORDS

CMOS, MOSFET, decoupling capacitors, leakage, noise, voltage drops, integrated circuit, digital standard cells, 65 nm technology

ABSTRAKT

Tato práce se zabývá návrhem blokovacích kondenzátorů ve formě CMOS tranzistorů pro stabilizaci napájecího napětí v digitální standardní knihovně se zaměřením na 65 nm technologii. Úvod poskytuje přehled CMOS technologie a popis MOS struktur. Analýza porovnává čtyři topologie blokovacích kondenzátorů (NMOS, PMOS, CMOS, Cross-Coupled) a identifikuje Cross-Coupled jako nejlepší volbu. Poslední část se zaměřuje na návrh a optimalizaci layoutu blokovacích kondenzátorů. Byly vytvořeny čtyři různé layouty, optimalizované pro kapacitu, činitel jakosti, svodový proud a kompromis mezi těmito faktory, aby mohly být integrovány do standardních digitálních knihoven podle specifických aplikací.

KLÍČOVÁ SLOVA

CMOS, MOSFET, blokovací kondenzátor, šum, úbytky napětí, integrovaný obvod, digitálně standardní buňky, 65 nm technologie

ROZŠÍŘENÝ ABSTRAKT

Trh s polovodiči se neustále vyvíjí, tranzistory se zmenšují a integrované obvody jsou stále složitější. Tento pokrok umožňuje vytvářet zařízení, která jsou výkonnější a energeticky úspornější. Nicméně toto neustálé zmenšování CMOS technologií přináší nové výzvy, včetně problémů se svodovým proudem, udržováním konstantního napájecího napětí a šumem.

Tato bakalářská práce se zaměřuje na podrobnou analýzu a návrh blokovacích kondenzátorů v rámci technologie CMOS 65 nm (Complementary Metal Oxide Semiconductor), které jsou klíčovým prvkem pro potlačení rušení napájecího napětí na čipu a vykrytí výkonových špiček. Úvodní část práce je věnována technologii CMOS, kde jsou vysvětleny základní principy. Je zde uveden podrobný popis tranzistoru MOSFET (Metal Oxide Semiconductor Field-Effect Transistor), který je nezbytný pro pochopení funkce blokovacích kondenzátorů v této technologii. Tato část se rovněž zaměřuje na kapacity spojené s tranzistory MOSFET a stručně se zabývá různými typy svodových proudů.

Následně je představen koncept digitálních standardních buněk, jejich význam v procesu návrhu čipů a typické složení. V práci je rozebrána problematika fyzického návrhu těchto buněk a diskutována omezení, která mohou při návrhu nastat.

Hlavní část práce se soustředí na analýzu čtyř různých topologií blokovacích kondenzátorů (NMOS, PMOS, CMOS a Cross-Coupled). Byly sledovány změny klíčových parametrů (činitel jakosti Q, efektivní odpor R_{ef}, efektivní kapacita C_{ef} a svodový proud I_l) v závislosti na změně základních charakteristik tranzistorů (šířka W, délka L a počet hřebínků NF). Pro simulaci těchto parametrů byl navržen obvod a byly stanoveny vztahy pro jejich výpočet. Na tyto simulace byly použity pouze netlisty ze schématu, tudíž bez parazitních prvků. K simulacím byly vytvořeny skripty v jazyce Python, které umožnily automatizaci tohoto procesu. Výsledky simulací byly normalizovány z důvodu podepsaného NDA (Non Disclosure Agreement) ohledně technologie 65 nm.

Z výsledků simulací vyplývá, že PMOS topologie vykazuje nejnižší svodový proud a dosahuje nejvyššího činitele jakosti, i když její kapacita je menší ve srovnání s NMOS topologií o stejné ploše. U NMOS topologie činitel jakosti dosahuje nejvyšších hodnot při vyšších frekvencích. CMOS topologie kombinuje vlastnosti NMOS a PMOS a nabízí vyvážené vlastnosti. Cross-Coupled topologie má shodnou kapacitu a svodový proud s CMOS, ale dosahuje nejhorších hodnot činitele jakosti. Z analýzy také vyplynulo, že s rostoucími parametry délky a šířky klesá činitel jakosti a frekvence při které dosahuje maxima. Naopak, zvyšování počtu hřebínků vedlo k vyššímu činiteli jakosti a frekvenci jeho maxima, ale mělo minimální vliv na kapacitu a svodový proud. Nižší prahové napětí zlepšilo frekvenci maxima činitele jakosti, ale zvýšilo svodový proud.

V závěrečné části práce byla pro podrobnou analýzu vybrána Cross-Coupled topologie. Byla vybrána primárně kvůli svým výrazně lepším ESD (Electrostatic Discharge) vlastnostem ve srovnání s ostatními topologiemi a také, že dosahuje stejných hodnot kapacity a svodového proudu jako CMOS topologie. Ostatní topologie mají připojené napájecí napětí přímo na hradle, což je v technologii 65 nm nebezpečné kvůli nízkému průraznému napětí oxidu křemíku. Cílem bylo vytvoření čtyř layoutů v digitálně standardní buňce, z nichž se každý zaměřoval na optimalizaci určitých parametrů, konkrétně kapacity, činitele jakosti, svodového proudu a jeden zaměřený na vyvážení těchto tří parametrů. Tyto layouty byly vytvářeny iterativně, počínaje počátečními návrhy, které byly postupně upravovány na základě zjištění ze simulací s parazitní extrakcí z layoutu v kombinaci se zjištěními z jednoduchých simulací z netlistu. V průběhu bylo navrženo více než šedesát různých variant. Výsledky simulací byly opět normalizovány.

Byly navrženy různé layouty buněk, které by mohly být integrovány do digitálních standardních knihoven na základě jejich specifických aplikací. Například buňka navržená pro minimalizaci svodového proudu by vyhovovala knihovně zaměřené na použití v aplikacích s nízkou spotřebou energie, kde je minimalizace ztráty energie prioritou. Dále by buňka optimalizovaná pro činitel jakosti byla přínosná v knihovnách vyžadujících vysoké spínací rychlosti a nízké rušení spojené s vysokými spínacími frekvencemi. Navíc by univerzálně optimalizovaná buňka byla ideální pro knihovnu, která se snaží optimalizovat různé parametry, jako je rychlost, plocha a svodový proud. Kromě toho by buňka optimalizovaná pro kapacitu vzhledem k ploše mohla být zahrnuta do všech knihoven a použita v případě potřeby zvýšení kapacity bez zvětšení plochy. Nakonec je možné přidat všechny tyto buňky do jedné knihovny, což umožní návrháři vybrat vhodnou buňku na základě konkrétní aplikace.

V pracích navazujících na tuto analýzu by bylo možné provést více simulací modelujících realističtější aplikace těchto buněk. Například by bylo možné prozkoumat, jak se chování navržených buněk mění, když jsou obklopeny jinými buňkami s různými funkcemi nebo vyššími kovovými vrstvami. Nakonec by jejich chování mohlo být fyzicky ověřeno, pokud by byly vyrobeny na testovacím čipu a simulované výsledky by mohly být porovnány se skutečným chováním. Případně by bylo možné použít tyto buňky v již vyráběném návrhu, který by se vyrobil na testovacím čipu, a následně by se tyto dvě implementace mohly fyzicky porovnat.

KUČERA, Radek. *Analysis and design of decoupling capacitors in the 65 nm CMOS digital standard library*. Bachelor's Thesis. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Microelectronics, 2024. Advised by Ing. Vojtěch Král

Author's Declaration

Author:	Radek Kučera
Author's ID:	240855
Paper type:	Bachelor's Thesis
Academic year:	2023/24
Topic:	Analysis and design of decoupling capacitors in the 65 nm CMOS digital standard library

I declare that I have written this paper independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the paper and listed in the comprehensive bibliography at the end of the paper.

As the author, I furthermore declare that, with respect to the creation of this paper, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll. of the Czech Republic, Section 2, Head VI, Part 4.

Brno .....

.....
author's signature*

*The author signs only in the printed version.

ACKNOWLEDGEMENT

I would like to express my gratitude to my thesis supervisor, Ing. Vojtěch Král, for his valuable advice and invaluable assistance in writing this thesis. I would also like to thank Ing. Jiří Šolc, whose support allowed me to work on this project. My thanks also go to all my colleagues who provided me with advice and inspiration.
Furthermore, I would like to thank my family and friends for their constant support throughout my studies. Last but not least, I would like to thank my classmates for the joint projects and mutual support that greatly helped me during my studies.

Contents

List of Figures

List of Tables

Listings

Introduction

The semiconductor market is constantly evolving, transistors are getting smaller and integrated circuits are getting more complex. These advances are making it possible to create devices that are both more powerful and more energy-efficient. However this continual shrinking poses challenges, including problems with leakage current, noise and the complexity of heat management.

The focus of this work will primarily be on addressing issues related to noise and, to a lesser extent, leakage. A key aspect of this work is the investigation of decoupling capacitors in 65 nm technology within digital standard cells. Decoupling capacitors, which can occupy up to 20 % of the chip area [1], play a crucial role in noise mitigation. Because of this large area, care must be taken to ensure that leakage is considered when designing these decoupling capacitors.

The initial part of the work is dedicated to CMOS (Complementary Metal Oxide Semiconductor) technology, where the fundamental principles and operations are explained. A detailed description of the MOSFET (Metal Oxide Semiconductor Field-Effect Transistor) transistor, essential for understanding the function of decoupling capacitors in this technology, is provided. This section is focused on the capacitances associated with MOSFET transistors and briefly discuss various types of leakage currents.

The next section describes digital standard digital cells and introduces how the complexity of digital circuit design is currently handled. All the important views for understanding circuit design and layout are also explained there.

In the preceding section, various topologies of decoupling capacitors that can be implemented in standard cells are introduced, along with a description of their basic properties. The specific parameters of these capacitors, crucial for assessment and evaluation, are clearly defined and outlined. A circuit for the simulations are designed to monitor these parameters, and the methodology for performing the simulations will be described. Finally, the results are evaluated.

In the final part, based on the results from the preceding sections, one topology is selected for deeper analysis. For this chosen topology, four layouts are created, each optimized for capacity, quality factor, leakage, and one balanced between these three parameters. These layouts are designed within the context of a digital standard cell. The design process is iterative, incorporating insights gained from initial simulations. The methodology for both design and simulation is thoroughly described.

1 CMOS

In 1958, Jack Kilby built the first integrated circuit with two transistors [2]. Just a few years later in 1963 Frank Wanlass invented CMOS (Complementary Metal Oxide Semiconductor) while he was working for Fairchild Semiconductor [3]. CMOS slowly became the favored choice over bipolar transistors because of its low power consumption, speed, greater noise immunity, and smaller on-chip area. This technological change has enabled the rapid development of integrated circuits.

Now Apple's latest M2 Ultra processor contains 134 billion transistors [4]. That's a 47% annual growth in the number of transistors over the last 65 years. This exponential growth was predicted by Gordon Moore in 1965 when he noticed that the number of transistors was doubling every year. This growth was adjusted to doubling every 2 years in 1975. This prediction is called Moore's law and is still valid to this day [5]. In Figure 1.1 the green line indicates Moore's prediction, which was calculated by doubling the first data point every two years, while the blue line is a regression of the actual transistor counts. Such exponential growth is possible only because of the CMOS scalability.

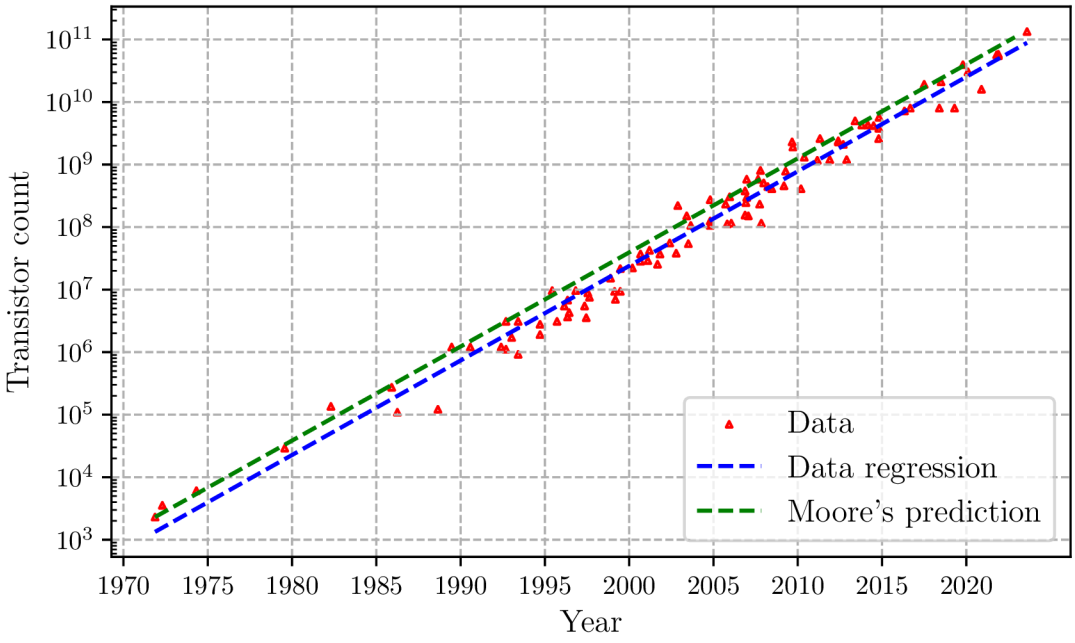

Figure 1.1: Moore's law: The number of transistors per microprocessor [6].

1.1 Technology Nodes

A technology node refers to a specific generation of chip manufacturing technology. The progression from one technology node to the next meant a reduction in the physical dimensions of the transistor. This meant higher performance, lower power consumption, and more efficient use of space, it also introduces several challenges. As transistors become smaller, issues like increased leakage current, susceptibility to electronic noise, and thermal management problems become more pronounced.

The technology node originally referred to a specific physical dimension in a chip design, namely the length of a transistor, measured first in micrometers (µm) and later in nanometers (nm). When the 32 nm node was reached around 2009, the term "technology node" started being used more for marketing than for precise technical measurement [7]. However, the trend of scaling down transistors in nanometer terms remained, though not directly linked to gate length. This shift meant smaller, yet not precisely defined, transistor sizes that continued to improve chip performance and efficiency.

This work will be focused on 65 nm node technology. At this specific node technology, transistors with three different threshold voltages (low, standard, and high) are available. Different threshold voltages can be used to optimize between performance or power consumption. Additionally, in static timing analysis (STA), different threshold voltages are employed to meet timing constraints. In this technology, silicon oxide is used as the gate dielectric. With this material, at this scale, challenges arise, specifically quantum mechanical effects become noticeable.

1.2 Principle of CMOS

The CMOS technology utilizes a pair of transistors to implement logic functions. These transistors are of two types: n-type (NMOS) and p-type (PMOS) MOSFETs (Metal Oxide Semiconductor Field-Effect Transistor). For example, the full name is NMOSFET, but it is often shortened to NMOS. MOSFET is a four-pin component consisting of the source, drain, gate, and bulk terminals. CMOS circuits are fabricated through multiple layers both inside and on the surface of a single silicon wafer, in this case, a p-type wafer. This wafer is then doped with donor or acceptor impurities, to form n-type regions and p-type regions, respectively. In manufacturing, NMOS transistors are fabricated directly on the substrate, and PMOS transistors are fabricated in the n-well. N-well is formed by diffusion of donors into the substrate [2, 3].

In Figure 1.2 the cross-sectional view of the CMOS pair can be seen. On the left side is PMOS, and on the right side is NMOS, with the corresponding schematic symbol below them. PMOS is formed in the n-well, and the drain and source are formed by p+ diffusion regions, where the "+" indicates they are heavily doped. The NMOS transistor is formed directly on the substrate, with the drain and source formed by n+ diffusion. It is important to mention that the drain and source are interchangeable because the transistor is symmetric. The gate is directly connected to a polysilicon. Polysilicon is a form of silicon made from small crystals and serves as a conductive layer. Between the substrate and this polysilicon there is a really thin layer of SiO_2, which is called gate oxide. The substrate or n-well is also called the body or bulk of the MOSFET. The colors used to describe the different parts in this Figure 1.2, are going to be same throughout this work.

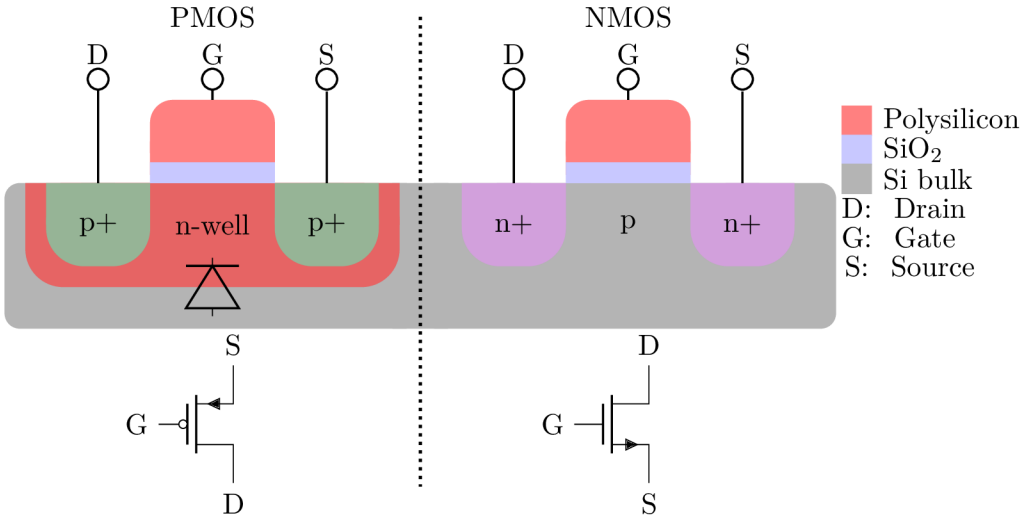

Figure 1.2: CMOS: Cross-sectional view and transistor symbols

The boundary between the n-well and the p-substrate together forms a PN junction that acts as a diode. This diode is illustrated in Figure 1.2. The area of this boundary is quite large, therefore, the p-substrate is usually connected to the lowest potential and the n-well to the highest potential in the circuit, so that the diode is reverse biased. MOSFET is a four-pin component, and the fourth pin is already mentioned bulk, and since in this case it is connected to the lowest potential or the highest potential, it is not shown in the schematic symbols. In this work, the lowest potential will be referred to as V_{SS}, which represents the logical zero, and the highest potential will be called V_{DD}, represents the logical one.

The main principle of CMOS is the use of complementary pairs to implement logic functions. Complementary means that NMOS and PMOS transistors are used together, where the NMOS passes a strong logic zero and the PMOS passes a strong logic one, unlike other logic families. Passing strong logic zero or one means that the output voltage is almost V_{SS} or V_{DD} [2]. Importantly, only one of these two transistors is open at a time, which means that there are practically no static power losses, and the only power loss is during the switching phase. Compared to bipolar transistors, which are current-controlled, CMOS is voltage-controlled, which further reduces static losses.

Many combinational logic gates, such as inverter (NOT), AND, OR, NAND, and others, as well as sequential circuits like the D flip-flop, can be created through the right combination of complementary pairs. More of these functions will be explored in Chapter 2. The principle of operation for these gates, starting with the simplest one, the inverter (NOT), can be explained to facilitate understanding.

1.2.1 Inverter

An inverter is a circuit that outputs inverse of its input logic value. In Figure 1.3(A), the schematic of the inverter is shown, where 'X' is the input that is connected to the gates of the transistors and 'Y' is output. The transistors labeled QP and QN are PMOS and NMOS respectively.

Before the operation of the inverter is explained, it is necessary to clarify the basic operation of a transistor. For the purposes of the initial explanation, the transistor can be thought of as a switch that is controlled by a voltage on the gate which determines whether the switch is 'on' or 'off.' For a PMOS transistor, a logic one turns it 'off', while for an NMOS, a logic one turns it 'on'.

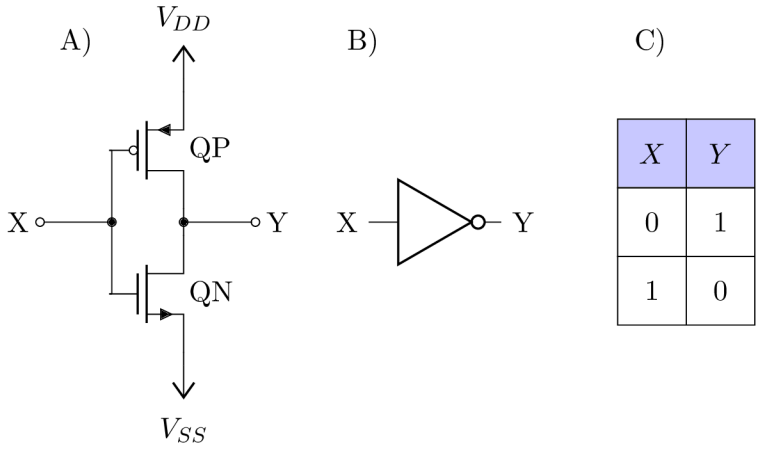

Figure 1.3: Schematic diagram (A), symbol (B) and truth table (C) of the CMOS inveter.

The operation of the inverter is such that when voltage V_{DD} is applied to an input (X), the transistor QP turns off (closes) and QN turns on (opens), thus connecting V_{SS} to output (Y). When voltage V_{SS} is applied to an input (X) transistor, QP turns on (opens) and QN turns of (closes), thus connecting V_{DD} to output (Y). If voltage V_{DD} and V_{SS} represents logical one and zero, respectively, then this circuit performs logical inversion.

1.3 MOSFET

MOSFET is a majority-carrier device; for NMOS, the majority carriers of electric charge are electrons, and for PMOS, the major carriers are holes. It is important to note that, when NMOS and PMOS transistors are the same size, NMOS is faster than PMOS because the electron mobility is approximately three times higher than the hole mobility. When a voltage is applied between the source and gate, the magnitude of the current flowing through the channel can be controlled by applying voltage to the gate. The channel is the area between the drain and the source. Beyond a certain voltage level on the gate, this channel becomes conductive, and the transistor starts to open. This voltage is called the threshold voltage. As mentioned in Section 1.1 the 65 nm technology used in this work allows the use of transistors with three different threshold voltages (low, standard, and high).

The dimensions of a transistor channel are usually described by two parameters: width and length. Another important parameter is the number of fingers. Number of fingers means that a single transistor is split into multiple parallel transistors with the same channel length, but their widths are divided by the number of fingers.

1.3.1 Layout

Before proceeding further, it is important that the method of creating a NMOS transistor in the layout design environment is explained, so that several frequently mentioned parameters are understood. In this section, many aspects, especially the production process, will be simplified as they are not essential for this work. Figure 1.4 illustrates a transistor in its drawn form, showing the active layer that outlines an opening in the silicon oxide. This opening indicates the physical location of the transistor. The n-select layer, in conjunction with the active layer, defines where the silicon is going to be doped with n-type donors. During the fabrication process, polysilicon is first laid down, and then the region outlined by the active layer is exposed and doped with donors, resulting in the formation of an NMOS transistor. This is illustrated in the cross-section below the drawn layers in Figure 1.4. Additionally, the drawn length and width, as seen in the figure, will slightly differ in the fabricated transistor, due to manufacturing variabilities.

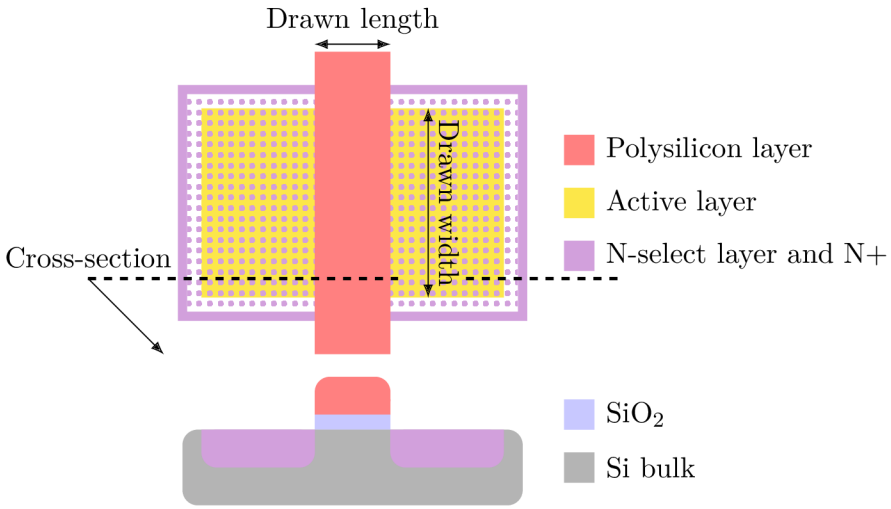

Figure 1.4: Layout of a NMOS and its cross-section

1.3.2 Capacitance

Capacitor

Capacitance is the ability of an object to store an electrical charge. The basic component that has capacitance is the capacitor. In its basic form, a capacitor are two conductive plates close together separated by an insulator (dielectric). Assuming there is a charge Q and $-Q$ on the plates and a voltage V between them, then the capacitance is given by:

$$C = \frac{Q}{V} \quad [\mathrm{F}] \tag{1.1}$$

where C is the capacitance in farads (F), Q is charge in coulombs (C), and V is voltage in volts (V). The capacitance of a capacitor can also be calculated from the physical dimensions by this equation:

$$C = \epsilon_0 \cdot \epsilon_r \cdot \frac{A}{d} \quad [\mathrm{F}] \tag{1.2}$$

where ϵ_0 is the permittivity of the vacuum, ϵ_r is the relative permittivity of the insulating material between the two plates. The A is area and d is the distance of the two plates.

MOS Capacitor

Looking again at Figure 1.2, it can be seen that the polysilicon and substrate is separated by an SiO_2. Which are two conducting surfaces separated by an insulator, thus forming a capacitor. The capacitance of this structure is based on the same relationship as equation 1.2. Due to the constant thickness of the oxide, the C'_{ox} constant is added for ease of calculation. The C'_{ox} is given by:

$$C'_{ox} = \frac{\epsilon_r \cdot \epsilon_0}{t_{ox}} = \frac{\epsilon_{ox}}{t_{ox}} \quad [\mathrm{F\,m^{-2}}] \tag{1.3}$$

where $\epsilon_0 = 8.85 \times 10^{-12}\,\mathrm{F\,m^{-1}}$ and the relative dielectric constant ϵ_r of SiO_2 is 3.9. The t_{ox} is the thickness of the SiO_2, which is also a constant. Then the capacitance C_{ox} of the structure can be calculated by:

$$C_{ox} = C'_{ox} \cdot A = C'_{ox} \cdot l_{drawn} \cdot w_{drawn} \quad [\mathrm{F}] \tag{1.4}$$

where A is the area between the polysilicon and the substrate. The l_{drawn} is the length and the w_{drawn} is the width of this transistor. Why the area is a multiplication of the length and width can be better seen in the Figure 1.4. The reason for using the drawn width and length is that the C_{ox} capacitance consists of the capacitance between the gate and drain/source overlaps and the capacitance between the gate and substrate. By using the drawn dimensions, these capacitances are accounted for.

The value of this capacitance will depend not only on the area and distance between the substrate (bulk) and the polysilicon (gate), but also on the voltage between the gate and bulk, due to the behavior of the semiconductor. This capacitance relationship can be explained by looking at the isolated structure only with gate and substrate body without a source and drain, which can be seen in Figure 1.5. Despite the earlier statement in Section 1.2 that V_{SS} represents the lowest potential, this figure, for consistency in labeling, presents an exception. Here, in the specific scenario where ($V_g < V_{SS}$), the V_{SS} is not the lowest potential.

In the figure the body is p-type so that means the carriers are holes. In Figure 1.5 (A) a negative voltage V_g is applied to gate and the holes are attracted to the region under the gate oxide, this mode is called accumulation. In this mode the value of capacitance is given by the equations 1.3 and 1.4

In Figure 1.5 (B) the voltage V_g is greater than V_{SS} but less than the threshold voltage V_t. This mode is called a weak inversion. In this case there is a positive charge on the polysilicon and the electrons from the substrate are repelled away, thus creating a depletion region under the silicon oxide. In this region, a small negative charge remains, which is built into the lattice of the crystal. Examining the capacitance in this mode reveals two capacitors connected in series. The first one is the oxide capacitor with the same capacitance as in accumulation mode. The second one is the capacitance between the depleted region and substrate.

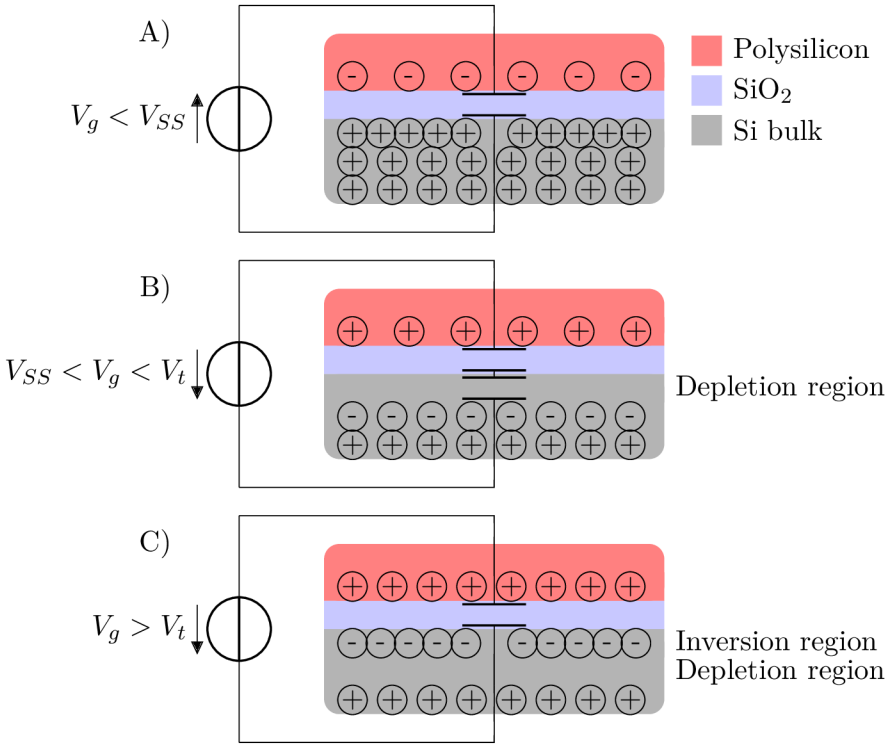

Figure 1.5: CMOS: Cross-sectional view and transistor symbols

In Figure 1.5 (C) is the voltage V_g bigger than the threshold voltage V_t. At this voltage, the free electrons are drawn directly under the region below the oxide and forms a conductive layer, called inversion layer. A transistor with this inversion layer formed is referred to as being in inversion mode. In this mode the resulting capacitance is again given by the equations 1.3 and 1.4 as the one operating in accumulation mode.

The behavior of capacitance C'_{ox} as a function of the voltage on the gate can be seen in Figure 1.6, where curve A) shows the behavior at low frequencies and curve B) at high frequencies. The reason why this capacitance is small at high frequencies is that the inversion channel does not have enough time to form. Therefore, the capacitance remains low.

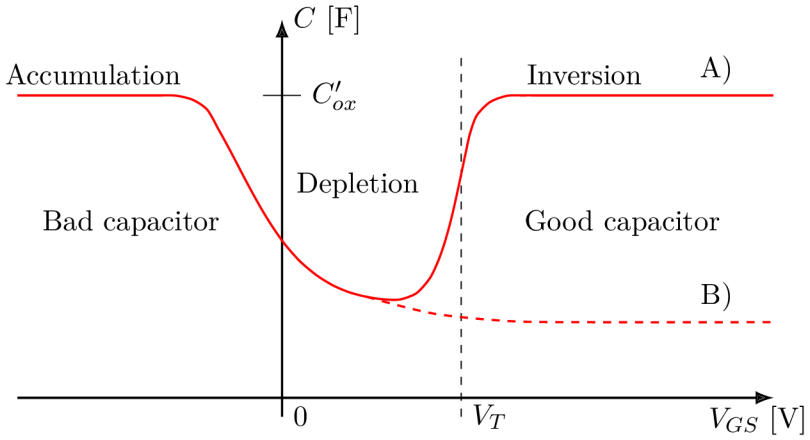

Figure 1.6: MOS capacitance-voltage curves (A) low frequency, (B) high frequency [3].

Further in 1.6 it can be seen, that in accumulation the MOS forms a bad capacitor and in inversion a good capacitor. This can be explained when looking at Figure 1.7. In this figure, there is NMOS transistor with the source, drain, and bulk connected to V_{SS} in two modes of operation. To save space, only half of NMOS in each mode is shown. On the left side, the NMOS is connected to operate in accumulation, so the gate voltage is $V_g < V_{SS}$. On the right side, the NMOS is connected to operate in inversion, so the gate voltage is $V_g > V_t$. When the transistor operates in accumulation, the closest V_{SS} is at the substrate connection and thus the substrate resistance is connected to the capacitor in series. This resistance significantly slows down the charging and discharging of the capacitor and thus degrades the frequency response. On the contrary, in inversion, a channel is formed between the drain and source, and the nearest path to V_{SS} is through this channel, where its resistance is negligible.

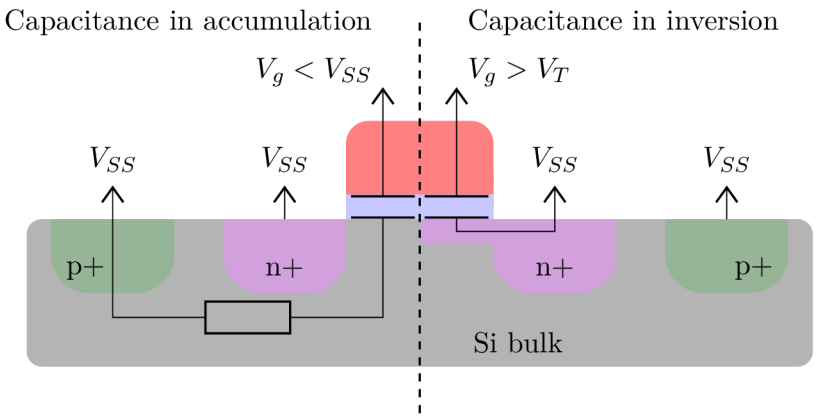

Figure 1.7: Comparison of capacitance and series resistance with transistor in accumulation (left) and in inversion (right)

Other Capacitance

MOSFET has many other parasitic capacitances. However, they are not so essential for this work and a detailed explanation would be lengthy and, moreover, not entirely necessary. So a few further parasitic capacitances will be roughly mentioned here.

In the Figure 1.8, different parasitic capacitances on an NMOS transistor can be seen. For the oxide capacitance described in the previous section, it is assumed that half of it is between gate and drain and the other half is between gate and source. Therefore, this capacitance is referred to as C_{gd} and C_{gs}. For this transistor there are also capacitances between the bulk and the drain, gate and source. The values of these capacinatces will vary depending on how the transistor is connected.

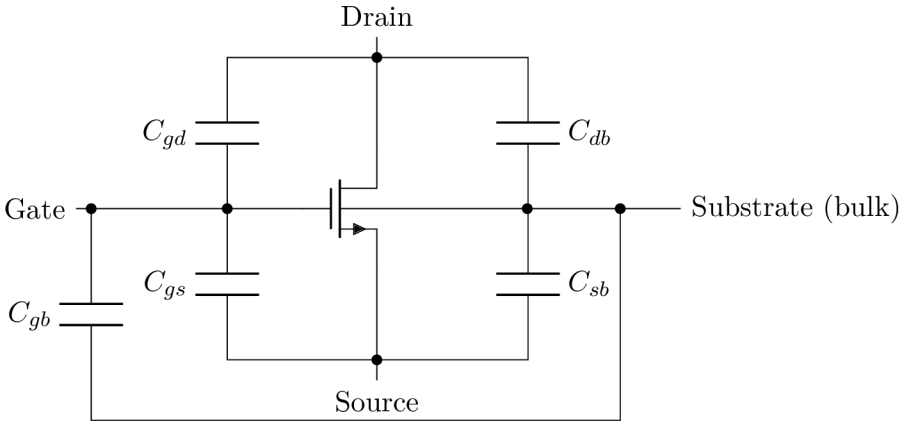

Figure 1.8: MOSFET capacitances [3].

1.3.3 Leakage

Unfortunately, the gradual shrinking of transistors has a negative effect, which is an increase in leakage current. This leakage current has a significant impact on power consumption, and in many technologies, it accounts for 30-50 % of total power losses [8].

Leakage current refers to the undesirable flow of current through a circuit when it is supposed to be off or in a non-conductive state. In MOSFET transistors, several mechanisms can cause this unwanted leakage current [8]. Leakage is dependent on the technology used; for instance, in 65nm technology, specific factors become the primary contributors [9].

Gate Tunneling Leakage

Gate tunneling leakage current is a quantum mechanical effect where electrons pass through a thin gate oxide [10]. The solution to this problem is the use of high permittivity gate dielectrics, which greatly reduce quantum tunneling, while maintaining a thin layer [11]. In this research, the focus is on the 65 nm technology node, where silicon oxide is still used as the gate dielectric. At this scale, gate tunneling poses a significant challenge. And given the topologies that will be used, which will be mentioned in Section 3.2.1, this leakage will be the biggest contributor.

Sub-threshold Leakage

Sub-threshold current flows between drain and source when the device is in weak inversion. This weak inversion layer allows a small current to flow. As the transistors get smaller, the threshold voltages get smaller. This results in it being easier for the inversion layer to form. As the threshold voltage shrinks, the sub threshold current grows exponentially [8].

Reverse Bias Leakage

As mentioned in Section 1.2, a diode is formed in the CMOS structure and is connected to be reverse biased. When the diode is in reverse bias, there is still some current flowing through it. This is due to the thermal generation of minority carriers that can drift across the depletion region.

2 Standard Cells

In this day and age when integrated circuits contain billions of transistors, as mentioned in Section 1, cost-effective and rapid development of a new chip is necessary. Given the huge cost to produce a few samples of chips, correct functionality on the first production run is absolutely crucial.

To manage the complexity and scale of chip design, a circuit is first described with a system specification, which might be described in text or a system specification language. This is then converted by designers into Register-Transfer Level (RTL) written in Hardware Description Language (HDL) such as VHDL or Verilog. Next step is to synthesize the RTL description, this includes several steps, with the most important being the conversion of RTL into a generic gates and registers and optimization of the logic for speed and area. The final step is mapping these generic gates to a specific set of cells [2]. These cells are called standard cells, and they are a predefined and pre-laid out function blocks.

The most important property of standard cells is their fixed height, while their width can vary. With this fixed height, they can be connected side by side and will align. They can also be flipped along the y-axis and still align. Additionally, when flipped around the x-axis, they can be connected directly above each other.

Standard cells are designed and optimized either for low power, high speed, or minimal layout area [12]. Additionally, they can be fine-tuned to achieve a compromise among these three key parameters, balancing the requirements based on specific design needs. The standard cell library contains a large number of cells with different functions, a few basic ones can be seen in the Table 2.1.

Table 2.1: List of common cells in the digital standard cell library [13].

Name of cell	**Description of the cells**
BUF, INV, AND, OR, NAND, NOR, XOR, XNOR	Simple logical functions with multi-inputs and different output strengths
HALF / FULL ADDER	2-bit half or full adder with different output strengths
MUX / DEMUX	Multiplexer or demultiplexer with different output strengths
ECO CELLS	Universal cells that can be tuned in case of need
AOI / OAI	Multi-input AND/OR or AND/OR logical combination
FLIP-FLOPS / SCAN FLIP-FLOPS	Flip-flops (plain, reset, set) with different output strengths
LATCHES	Flip-flop controlled with level
FILLER / FILL CAP	The cell can connect power rails or can be used as decoupling capacitances
CLOCK GATING CELLS / DELAYS	Used to synchronize the clock signal, delays are used for compensation of STA violations

2.1 Physical View

In the context of a standard cell library, the term "view" is used to describe different representations or models of the standard cell. In this work, the physical view will be the most important.

2.1.1 Symbol View

The view symbol serves as a view of the cell at the highest level of abstraction showing only the most important elements such as inputs and outputs. An example of such a symbol can be seen in the Figure 2.1.

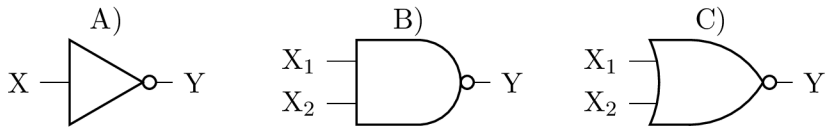

Figure 2.1: Symbol view of an inverter (A), NAND (B) and NOR (C).

2.1.2 Schematic View

The schematic view shows a detailed electrical representation of the standard cell. It displays how transistors and other passive components are internally connected and how such a standard cell works internally. In the Figure 2.2 the NAND (A) and NOR (B) circuit schematic can be seen.

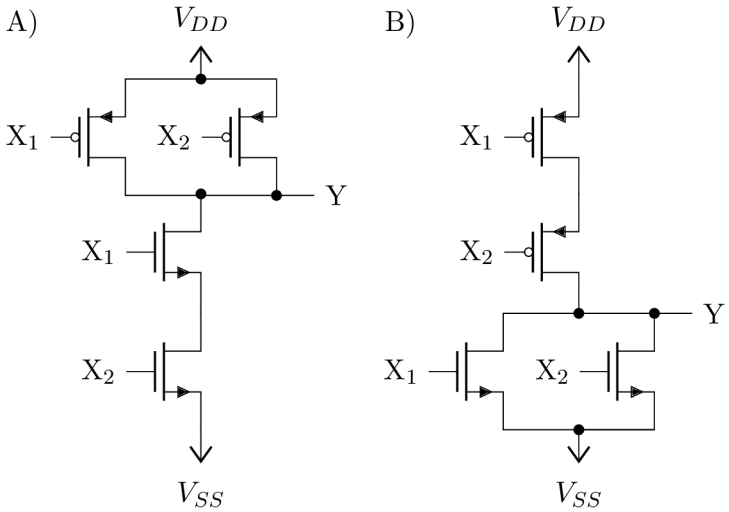

Figure 2.2: Schematic view of a NAND (A) and NOR (B).

2.1.3 Layout View

In Figure 2.3 standard cell template is shown. In this cell, there is a designated space for NMOS at the bottom with layers prepared for its formation. The same applies to PMOS, for which there is a reserved space at the top. This figure does not show the actual proportions of the individual layers so that all layers are easily visible. The reason why the space allocated for PMOS is larger than NMOS is that PMOS is slower than NMOS. Next, there are power rails, V_{DD} at the top and V_{SS} at the bottom. With all these pre-prepared layers to form a transistor, the designer only needs to outline the active layer and the polysilicon layer, as detailed in Section 1.3.1. Then the transistors can be connected to the power rails and to each other using metals and contacts. The metal layer is a conductor that is used to connect components to each other. The designation "Metal1" for the layer suggests that multiple metal layers are available, with their number being dependent on the specific manufacturing technology and the chosen metallization option. These layers are crucial for forming interconnections between the various components of the device. The next layer is the contact layer, which is used to connect the metal layers to each other or to connect the metal directly to the substrate. The active layer can be seen around the contacts on the power rails, indicating that the contacts are connected to the bulk of the transistors.

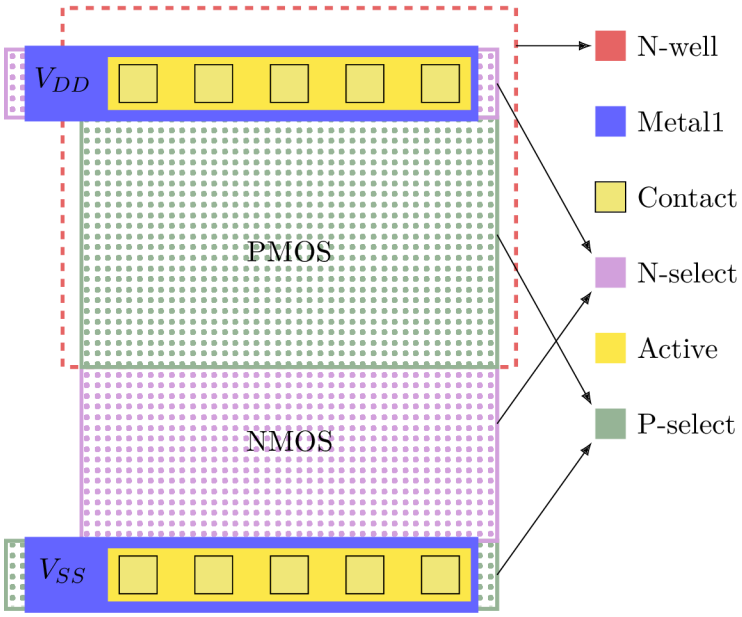

Figure 2.3: Standard cell template.

3 Decoupling Capacitors

In this chapter, the primary sources of power supply noise, primarily voltage drops, are introduced, along with potential solutions using decoupling capacitors. Various topologies are also presented, followed by a simple netlist simulation that allows for the determination of parameters important for these capacitors. The results of the simulation are then evaluated.

3.1 Power Supply Noise

With the increase in chip sizes and the reduction in transistor sizes, the corresponding wires become narrower while extending in length. This combination, with increasing switching speed and decreasing power supply voltage, causes a serious noise immunity problem, and maintaining good quality of the power supply becomes an issue [14].

Ripple

Ripple in the context of power supply noise refers to the small, unwanted residual periodic variation of the DC voltage within a power supply after being rectified. The magnitude of the ripple is typically influenced by the design of the power supply and its filtering. A higher ripple can lead to increased noise within the circuit, affecting signal integrity and potentially leading to incorrect operations, especially in sensitive digital circuits.

EMI

EMI is caused by a change in a current over time, which is referred to as $\Delta I/\Delta t$ noise [15]. This change in current causes electromagnetic emissions, and conversely external electromagnetic emissions can also cause a change in current in the circuit. In integrated circuits such a change can be caused by a fast rising or falling edge of a clock signal or a data signal, thus emitting an electromagnetic emission. On the contrary, these external emissions can be received by the integrated circuit and cause unintended logic switching.

Voltage Drops

In the Figure 3.1, a simple power delivery system is depicted. Such a system contains a power supply and a power load. The wiring between the power supply and the power load is not ideal and consists of parasitic resistance R_{dd} and R_{ss}, and inductance L_{dd} and L_{ss}. When the current $I(t)$ is drawn into the load, there is a

voltage loss in these parasitic elements. On the wiring there is resistive voltage drop $\Delta V_R(t)$ and inductive voltage drop $\Delta V_L(t)$ described by the following equations:

$$\Delta V_R(t) = I(t) \cdot R \quad [\mathrm{V}], \tag{3.1}$$

$$\Delta V_L(t) = L \cdot \frac{\mathrm{d}I(t)}{\mathrm{d}t} \quad [\mathrm{V}]. \tag{3.2}$$

The symbols R and L refer to the parasitic resistance and inductance on the wiring, with units in ohms (Ω) and henries (H), respectively. Therefore, at the terminals to which the load is connected, the voltage changes by these losses. As can be seen in Figure 3.1, the voltage at the first terminal decreases to:

$$V_{DL}(t) = V_{DD} - \Delta V_R(t) - \Delta V_L(t) \quad [\mathrm{V}]. \tag{3.3}$$

Conversely, the voltage at the second ground terminal increases to:

$$V_{SL}(t) = V_{SS} + \Delta V_R(t) + \Delta V_L(t) \quad [\mathrm{V}] \tag{3.4}$$

In these expressions, $V_{DL}(t)$ and $V_{SL}(t)$ represent the voltages at the terminals where the load connects to the power supply's V_{DD} and V_{SS} potentials, respectively.

This power supply uncertainty caused by voltage drop is referred to in literature as "power supply noise" [16]. Power supply noise negatively affects the behavior of circuits. It causes uncertainty in clock and timing signals, on-chip clock jitter, lowering noise margin, and degrading gate oxide reliability [16]. When designing a load circuit, it is assumed that the supply voltage is within a certain range, this range is called the noise margin and is typically 10 % of V_{DD}. Therefore, the main objective of a power supply system is to deliver sufficient current to each circuit and not to exceed the noise margin. One potential solution to mitigate power supply noise is the use of decoupling capacitors [14, 16, 17].

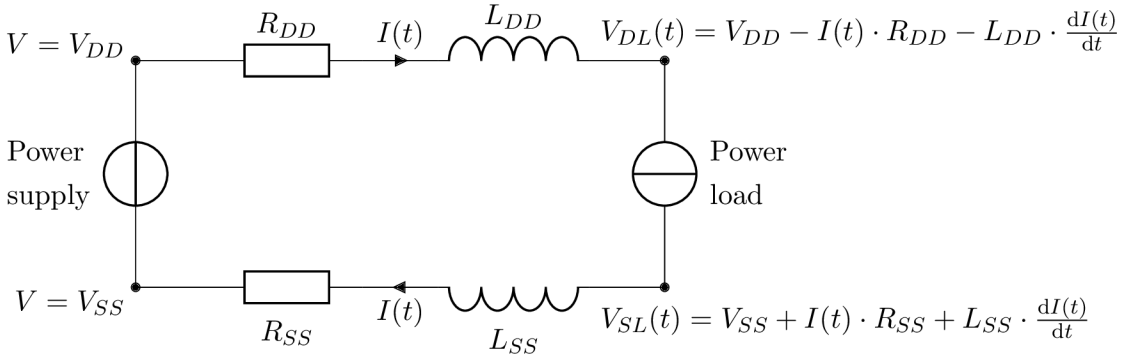

Figure 3.1: Power delivery system, with non-ideal interconnection lines [16].

3.2 Power Supply Decoupling

Decoupling capacitors are used to keep the supply voltage in the noise margin to ensure signal integrity while reducing electromagnetic interference (EMI) radiated noise. From a physical point of view, the decoupling capacitor serves as a storage of charge and energy. The energy stored in a capacitor is given by

$$E_C = \frac{1}{2} \cdot C \cdot V^2 = \frac{1}{2} \cdot \frac{Q^2}{C} = \frac{1}{2} \cdot V \cdot Q \quad [\text{J}], \tag{3.5}$$

where C is capacity, V is the voltage and Q is the charge stored in capacitor. Since the voltage in a chip is fixed, it's desirable to have as large a capacitance as possible [16]. If the equation 1.4 is recalled, a larger capacity can be achieved either by enlarging the width or length, or by reducing the gate oxide thickness. However, several limitations are faced: the area is constrained by the cell size and primarily by the available area on the chip, and the thickness of the oxide is determined by the manufacturing technology. Even if variation in the oxide thickness were possible, an increase in the leakage current caused by tunneling would start to be observed, as described in section 1.3.3.

In a design these capacitors are placed between the power supply and the power load, ideally as close to the load as possible. When the load switches it needs energy, which is supplied by the decoupling capacitor that is nearby, and does not need to take energy from the power supply. The wiring path between the load and a capacitor is much shorter, then that from power supply, thus resistive and inductive voltage drop is reduced.

3.2.1 Topologies

In a Figure 3.2 four possible topologies in CMOS 65 nm are presented [1]. The first two topologies 3.2(A) and (B) are NMOS and PMOS transistors connected in inversion mode between the power supply rails. The third topology is the CMOS seen in Figure 3.2(C), which is just a combination of the previous two. And the fourth last topology in the Figure 3.2(D) is Cross-Coupled.

As can be observed, the first two topologies, NMOS and PMOS, consist of only one transistor each. This will be reflected in the simulation results, as their capacitance will be smaller due to their smaller area. In all simulations, the same parameters will be used for all transistors to ensure that the results can be reasonably compared and processed. It would be possible to use two transistors in parallel for the simulation and this would solve this problem, but from a layout point of view two transistors of the same type would not fit in a standard cell because of the layer spacing rules. For this reason they will be simulated as a single transistor

For these four topologies, three different devices will be used, each with a different threshold voltage. As mentioned in 1.1, there are three different threshold voltages available in 65 nm technology, namely low, standard and high. In total, four different topologies with three different threshold voltages will be examined, which makes a total of twelve combinations. The Table 3.1 shows a list of these combinations and appropriate labels, which will subsequently be used in this work.

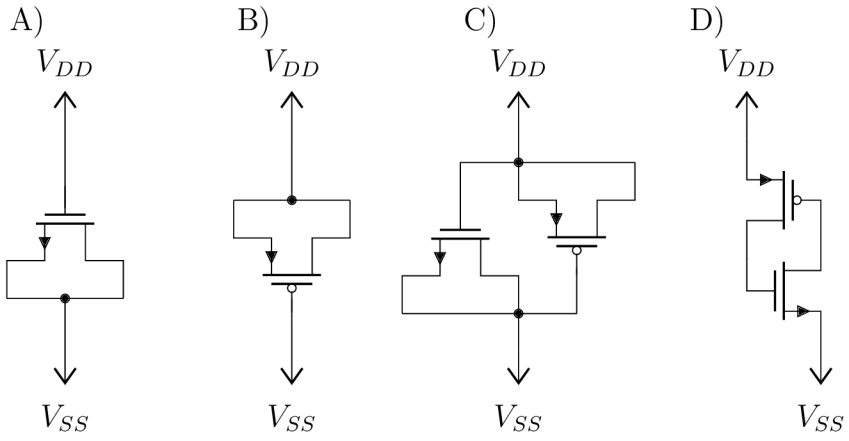

Figure 3.2: CMOS decoupling capacitors topologies: (A), NMOS (B), PMOS (C), CMOS and (D) Cross-Coupled

Table 3.1: Threshold voltage and topology labeling.

	Topologies			
Threshold	NMOS	PMOS	CMOS	Cross-Coupled
Low	n_l	p_l	cmos_l	cp_l
Standard	n_s	p_s	cmos_s	cp_s
High	n_h	p_h	cmos_h	cp_h

Furthermore, all combinations of the following parameters will be used for all topologies:

- **Length values:** 1λ, 2λ, 4λ, 5λ, 7λ
- **Width values:** 1μ, 2μ, 4μ
- **Number of fingers values:** 1, 3, 6

That's five different values for the length, three different values for the width, and three different values for the number of fingers, for a total of forty-five different combinations. Together these forty-five combinations with four different topologies and three threshold voltages make five hundred and forty different combinations. However, not all of these combinations will be valid, because for some combinations

of width and number of fingers, the division will result in a smaller width than the minimum allowed. These invalid combinations will be filtered out during processing. Unfortunately, due to the NDA, all potentially sensitive information will presented in a normalized form. The channel lengths and widths will be normalized to values where 1λ will corresponds to the lowest used length value and 1μ for lowest used width. The number of fingers remains the same. This means that if the lowest width of 420 nm was used for the simulation, this width would be marked here as 1μ and then the 840 nm width would be marked here as 2μ and so on.

3.3 Simulation

To determine the impact of various topologies and changes in their parameters, specific parameters need to be monitored. The key parameters of interest are effective capacitance, resistance, and quality factor. These parameters are chosen for analysis because the topologies can be modeled as a series connection of a resistor and a capacitor, as illustrated in Figure 3.3.

Obviously, the true model of a device is more complex than a mere resistor and capacitor; hence it is represented as a lumped RC circuit for the sake of simplification and practical analysis.

Looking again at the Figure 3.3 it can be seen, that the R_{ef} and C_{ef} are functions of a frequency. This is due to the behaviour of the channel, and how at higher frequencies the inversion layer fails to form as mentioned in the section 1.3.2.

The quality factor in this context is a measure of efficiency in terms of energy loss. It is defined as the ratio of energy stored to energy dissipated per cycle.

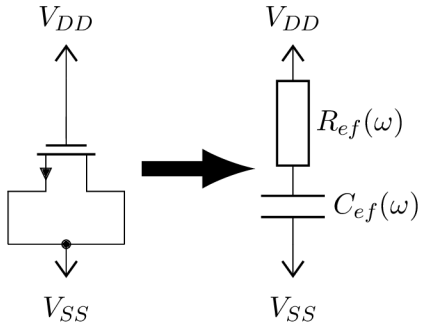

Figure 3.3: NMOS modeled as lumped RC circuit.

To simulate the selected parameters, the following circuit was chosen, which can be seen in Figure 3.4. In this circuit, the topology is connected to a DC voltage source with voltage $V_{dc} = V_{DD}$ in series with AC voltage source with $V_{ac} = 10\,\%$ of the V_{DD} voltage. This simulation will represent a topology connected to a power

rail, with an AC noise of a specific frequency superimposed on the voltage. The reason why it is 10 % of V_{DD} is because it is the noise margin limit. A complex current I_c with phase $\phi°$ will flow through this circuit. This current will be used to calculate the studied parameters.

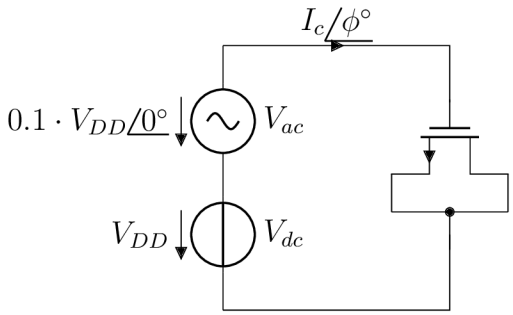

Figure 3.4: Circuit for simulation of selected parameters.

The following calculation equations are intended for a specific frequency only. The parameters will be calculated using the fact that the complex current $\hat{I}_c$ flows through the circuit. From this current, the complex impedance $\hat{Z}$ of the topology can be determined using equation 3.6.

$$\hat{Z} = \frac{\hat{V_{ac}}}{\hat{I_c}} \tag{3.6}$$

Furthermore, it can be utilized that R_{ef} and C_{ef} are in series, so the real component of the complex impedance will be R_{ef} and the complex component can be used to calculate C_{ef}. It can be seen in equation 3.7 that the R_{ef} is given by the real part of the complex impedance $\mathrm{Re}(\hat{Z})$.

$$R_{ef} = \mathrm{Re}(\hat{Z}) \tag{3.7}$$

The following equation will be used to calculate C_{ef}

$$C_{ef} = -\frac{1}{2\pi \cdot f \cdot \mathrm{Im}(\hat{Z})} \tag{3.8}$$

where f is the frequency of V_{ac} and $\mathrm{Im}(\hat{Z})$ represents the imaginary component of the impedance $\hat{Z}$. For the calculation of the quality factor Q the equation 3.9 will be used.

$$Q = \frac{\mathrm{Im}(\hat{Z})}{\mathrm{Re}(\hat{Z})} \tag{3.9}$$

The next parameter to be monitored will be leakage. For the sake of clarity it is not shown in the Figure 3.4, but it is the DC current flowing from the source V_{dc}

3.3.1 Simulation Methodology

Spectre simulator from Cadence [18] was chosen for the simulations. In Listing 3.1 an example of the netlist that was used for the simulations can be seen. Under the comment "Parameters", all the necessary parameters are defined. The most important ones are "Length, Width, nf_p", which are used in subcircuits. Under the comment "netlist", the actual circuit configuration can be seen, which corresponds to the configuration from the Figure 3.4.

From the comment "simulation", two simulations are defined, necessary for simulating the examined parameters. The AC analysis was conducted from a frequency of 1 MHz to 10 GHz with a step of 30 points per decade. And a DC analysis was performed to determine leakage.

Under the comment "Topology example", you can see an example of a subcircuit. For each topology, a separate subcircuit was created. Of course, not all are listed in the actual netlist, but they are included from another file, which can be seen in appendix A.5. And the entire netlist that was used for the simulations can be seen in appendix A.1

Listing 3.1: Example of a Spectre netlist used for simulations with subcircuit example

```
global 0
//Parameters
parameters Length=λ Width=μ DC_voltage=1.2 \
Amplitude=0.1*DC_voltage nf_p=1 worktmp=25
//netlist
VDC(v_ac 0) vsource dc=DC_voltage type=dc
VAC(decap v_ac) vsource type=sine ampl=Amplitude
Xdecap(decap 0) {topology} //one of different topologies
//simulations
ac1 ac start=1M stop=10G dec=30 annotate=sweep
dc1 dc
//Topology example
subckt cp_s (in_s 0)
    Tp_cp_s (in_s 1 2 in_s) pfet l=Length w=Width nf=nf_p
    Tn_cp_s (1 2 0 0) nfet l=Length w=Width nf=nf_p
ends
```

To facilitate simulation, due to the large number of combinations, the Spectre MDL (Measurement Description Language) was used. Part of the code used for simulation is presented in Listing 3.2. It can be seen that there are three nested "foreach" loops in this code, which steps through all combinations of parameters and calls two functions inside. These functions "ac_sim_file" and "dc_sim_file"

execute the given AC and DC simulation respectively and calculate the monitored parameters and save them to a file. Their full functionality can be seen in the appendix A.3.

Unfortunately, Spectre does not allow changing the topology during the run of a simulation[19]. Therefore, a Python script was created that runs the simulation, changes the topology, and then runs the simulation again. In this way, it simulates all twelve topologies in twelve simulation runs. For better clarity, the script also saves error and warning messages, which are then printed out after all the simulations are completed. The entire code can be seen in the appendix B.2.

Listing 3.2: Example MDL script for parameter sweep in Spectre simulations

```
foreach Length from {1λ,2λ,4λ,5λ,7λ} {
  foreach Width from {1μ,2μ,4μ} {
    foreach nf_p from{1,3,6}{
      run ac_sim_file(out=fmt("data/cp_l_l=%S_w=%S_nf=%S.raw"
   , Length, Width, nf_p))
      run dc_sim_file(out=fmt("data_leak/Leak_cp_l_l=%S_w=%
   S_nf=%S.raw", Length, Width, nf_p))
      }
  }
}
```

Also, all the simulation results, including effective capacitance, resistance, and quality factor, will be normalized. The Cross-Coupled topology with standard threshold voltage, length 1λ, width 1μ and 1 finger will be reference topology, and all the data will be normalized to it. Maximal values from results of this reference topology will be taken and all data from all topologies are going to be normalized by dividing their values by those maximal values. This means, that the reference topology will have their maximal values equal to one after the normalization, and others will be linked to this value. The python script used for this normalization can be seen in appendix B.1. In Table 3.2 it can be seen how the observed values look after normalization for the reference topology.

Table 3.2: Maximal observed values for Cross-Coupled topology with l=1λ, w=1μ and nf=1 aftter the normalization.

f_{Qmax} $[Hz]$	Q_n $[-]$	f_{Rmax} $[Hz]$	R_n $[-]$	f_{Cmax} $[Hz]$	C_n $[-]$	I_{ln} $[-]$
1.359E+07	1	1E+06	1	1E+06	1	1

3.4 Simulation Results

In this section it will be investigated how variations in parameters such as length (L), width (W), number of fingers (NF), and threshold voltages (V_t) impact the quality factor (Q), effective resistance (R_{ef}), capacitance (C_{ef}), and leakage (I_l). Furthermore, it is important to mention again that the NMOS and PMOS topologies are modeled as if they occupy only half of the standard cell, and therefore their results are not entirely comparable with those of CMOS and Cross-Coupled topologies. However, they will be compared here since this section evaluates their behavior in response to parameter changes. This will be most significant in the case of capacitance, where their capacitance will be roughly half that of CMOS and Cross-Coupled.

3.4.1 Quality Factor

In this section, the effects of varying width, length, and number of fingers on the quality factor will be investigated. The results will be evaluated on the threshold voltage where the influence can be best observed, and then at the end of this section, the impact among various threshold voltages will be assessed.

Effect of Width Variation

The impact of varying width on quality factor for all topologies with standard threshold voltage can be seen in Figure 3.5. It is evident, that an increase in width causes both the quality factor and its maximum frequency point f_{Qmax} to decrease. Similar trends are observed in other topologies, although the effects are less pronounced.

Differences between the various topologies are also illustrated in Figure 3.5. It is apparent that the Cross-Coupled topology has the smallest quality factor of all, and its values are lower at all frequencies than those of the other topologies. The PMOS topology, on the other hand, reaches the highest overall quality factor values. The NMOS topology demonstrates its peak quality factor values at the highest frequencies. And CMOS is somewhere in between, because it is a combination of both PMOS and NMOS.

Effect of Length Variation

Variation in length also affects quality factor across all topologies. An increase in length results in a decrease in quality factor at f_{Qmax}, similar to the trend observed with changes in width. This influence can be seen in Figure 3.6 where all topologies with a standard V_t are presented.

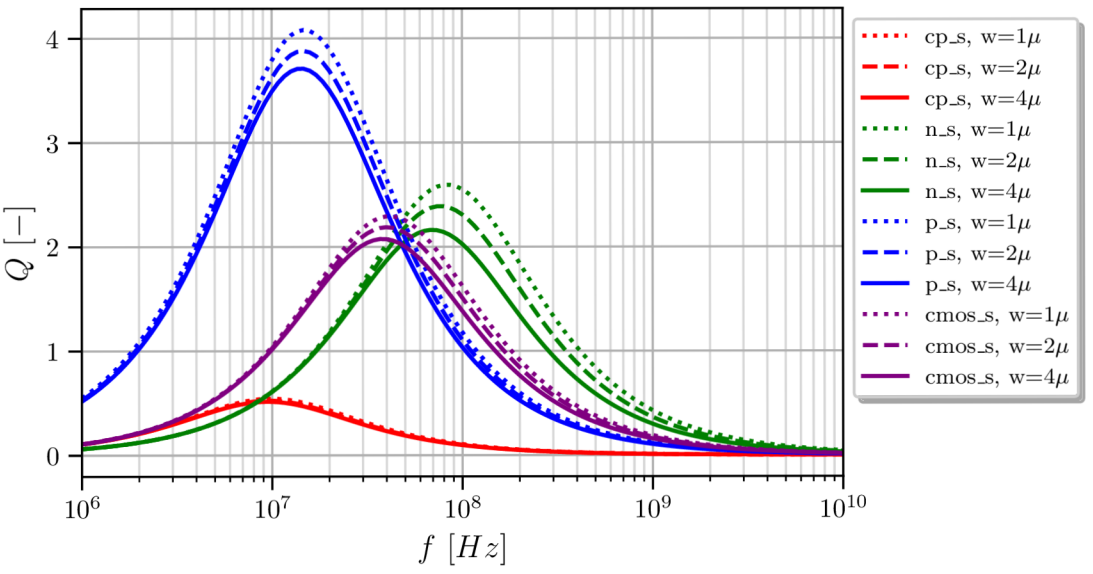

Figure 3.5: Effect of width variation on the quality factor for all topologies at standard threshold voltage and at a constant length l=2λ and number of fingers nf=1.

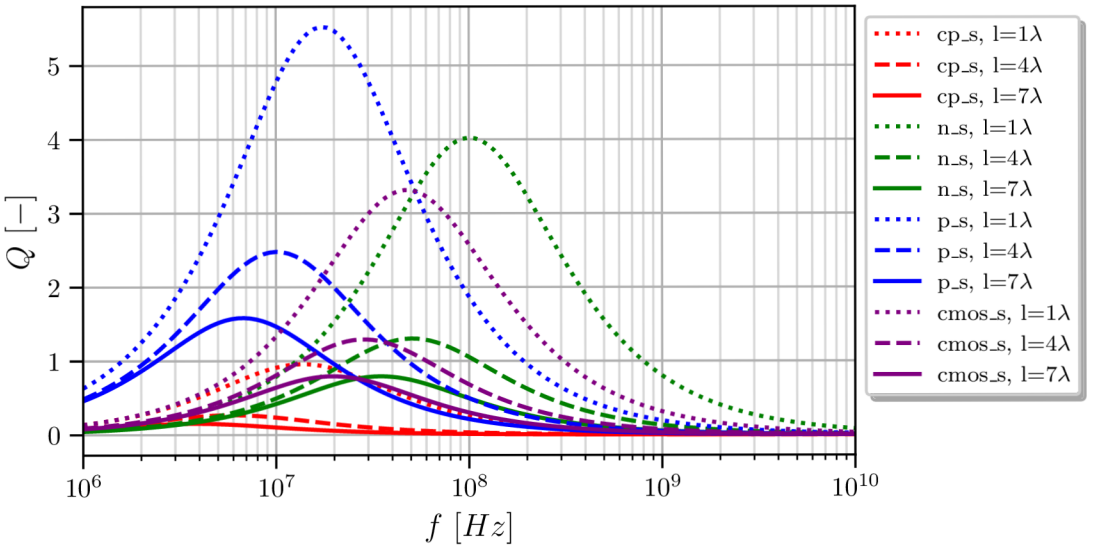

Figure 3.6: Effect of length variation on the quality factor for all topologies at standard threshold voltage and at a constant width w=2μ and number of fingers nf=1.

Effect of Number of Fingers Variation

The impact of changing number of fingers has the opposite effect than changing width or length. With a higher number of fingers, both quality factor and f_{Qmax}

increase, which can be seen in Figure 3.7. The increase in f_{Qmax} is not very noticeable in this figure, therefore it can be seen for NMOS topology in the Table 3.3.

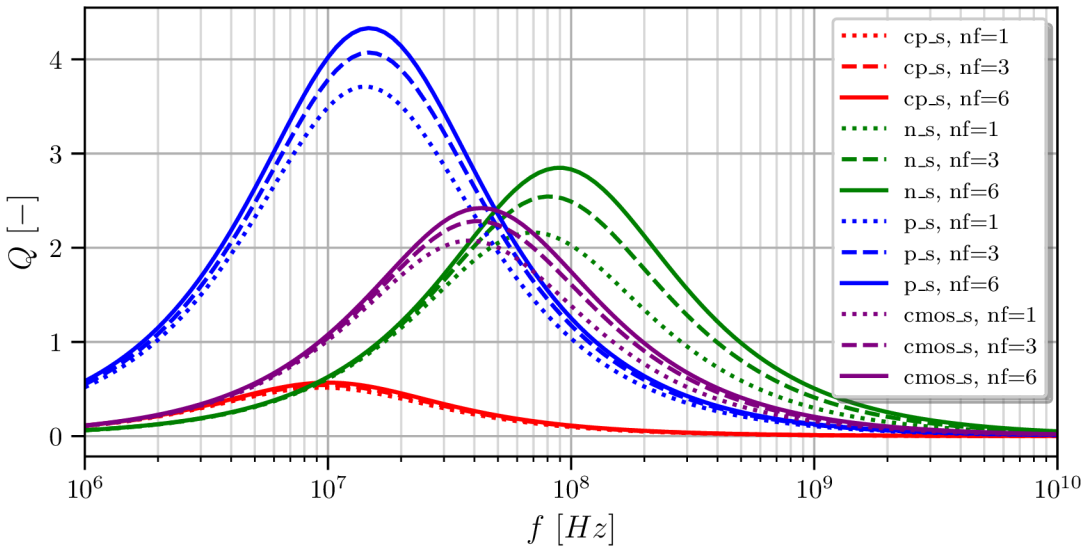

Figure 3.7: Effect of changing the number of fingers on the quality factor for all topologies at standard threshold voltage and at a constant length l=7λ and width w=2μ.

Table 3.3: Maximal quality factor values for NMOS topology corresponding to the data presented in Figure 3.7

Topology	nf [−]	f_{Qmax} [Hz]	Q_{max} [−]
n_s	1	6.813E+07	2.162
n_s	3	7.943E+07	2.544
n_s	6	9.261E+07	2.848

Effect of Threshold Voltage Variations

Changes in the threshold voltage for different widths in the NMOS topology can be seen in Figure 3.8. From this figure, it can be noticed that with lower threshold voltage the frequency f_{Qmax} increases. It can also be noticed that the curves covers the largest area at low threshold voltage. The increase in f_{Qmax} applies to NMOS, Cross-Coupled, and CMOS topologies for all the different width, length and number of fingers combinations.

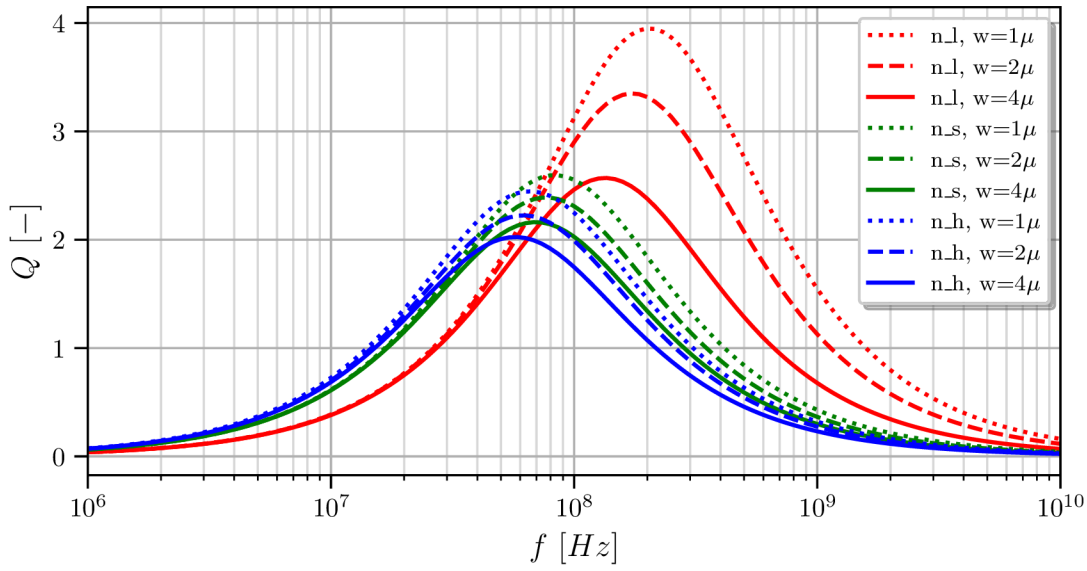

Figure 3.8: Effect of width variation on the quality factor for NMOS topology with all threshold voltages and at a constant length l=2λ and number of fingers nf=1.

The increase in f_{Qmax} is not same for the PMOS topology. Because this influence is not entirely apparent from a graph, it will be demonstrated in the Table 3.4. The table is aligned from the lowest frequency where quality factor peaks. From the table, it can be seen that for low and high threshold voltage the peak is at the same frequency, and that the standard threshold voltage peaks at higher voltage.

Table 3.4: Effect of threshold voltage variation on the f_{Qmax} and Q for PMOS topology at a constant length l=2λ, width w=2μ, and number of fingers nf=1.

Topology	w [nm]	f_{Qmax} $[Hz]$	Q_{max} $[-]$
p_l	2μ	1.359E+07	4.009
p_h	2μ	1.359E+07	2.900
p_s	2μ	1.468E+07	3.881

This behavior cannot be described as simply as with the other topologies, but when looking at Table 3.5, it can be noticed that for $l = 7\lambda$ and $l = 4\lambda$, this topology behaves like the other topologies. However, at $l = 1\lambda$, it behaves differently. And given that the previous Table 3.4 was for $l = 2\lambda$, it could be said that for the PMOS topology, the influence of threshold voltage will also be a function of length.

Table 3.5: Effect of length variation on the maximal values of quality factor for PMOS topology with all threshold voltages and at a constant width w=2μ and number of fingers nf=1.

Topology	l [nm]	f_{Qmax} $[Hz]$	Q_{max} $[-]$
p_h	7λ	5.412E+06	1.303
p_s	7λ	6.813E+06	1.584
p_l	7λ	7.943E+06	1.766
p_h	4λ	8.577E+06	1.982
p_s	4λ	1.000E+07	2.479
p_l	4λ	1.080E+07	2.690
p_l	1λ	1.585E+07	5.549
p_s	1λ	1.711E+07	5.522
p_h	1λ	1.848E+07	3.631

3.4.2 Effective Resistance

This section explores how changes in width, length, and number of fingers affect the effective resistance.

Effect of Width Variation

The impact of changing width on effective resistance is best observed in the Cross-Coupled topology, as shown in Figure 3.9. This figure shows that from about frequency $f = 10^7$ Hz onwards the effective resistance decreases with decreasing threshold voltage and increasing width. And for frequencies below $f = 10^7$ Hz the effective resistance decreases with increasing threshold voltage. This is true for all other topologies.

Furthermore, differences between individual topologies are best observed at low threshold voltage, as illustrated in Figure 3.10. This figure shows that below the frequency of $f = 10^7$ Hz, the NMOS topology has the highest effective resistance, followed by Cross-Coupled and CMOS with approximately the same resistance, and PMOS with the lowest. This order reverses after the values stabilize, approximately from the frequency of $f = 5 \cdot 10^8$ Hz, with the highest to lowest resistance in the following order: Cross-Coupled, PMOS, NMOS, and CMOS. A similar pattern of behavior with the same order is also observed for the other two threshold voltage values.

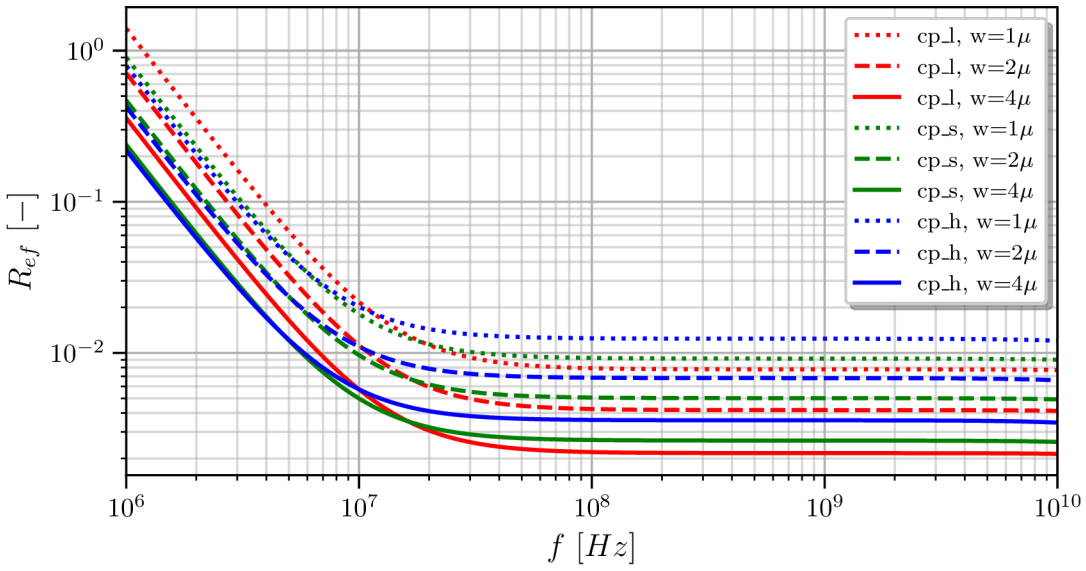

Figure 3.9: Effect of width variation on resistance for cross-coupled topology with all threshold voltages and at a constant length l=2λ and number of fingers nf=1.

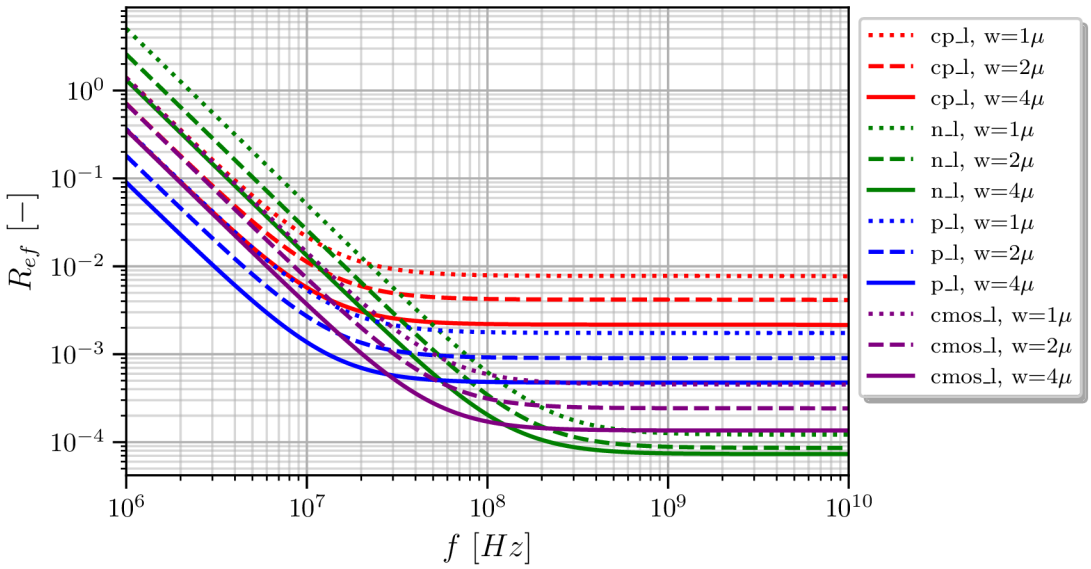

Figure 3.10: Effect of width variation on resistance for all topologies at low threshold voltage and at a constant length l=2λ and number of fingers nf=1.

Effect of Length Variation

The influence of the variation in length on effective resistance is most clearly observed in the Cross-Coupled topology, as can be seen in Figure 3.11. It can be noted that up to a certain frequency, in this case around $f = 10^7$ Hz, the effective resistance

decreases with increasing length and higher threshold voltage, which is similar to the effect observed with the variation in width. However, from approximately the frequency of $f = 10^7$ Hz onwards, both variations have the opposite effect, meaning that with an increasing length and higher threshold voltage, the effective resistance increases. Also, a further decrease in effective resistance can be observed from the frequency of $f = 3 \cdot 10^9$ Hz for length l=7λ.

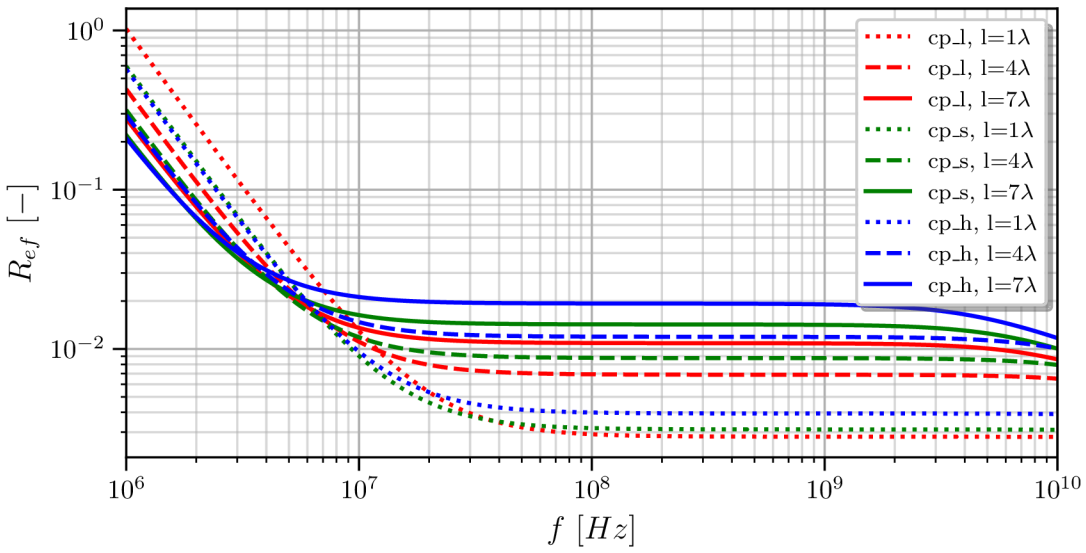

Figure 3.11: Effect of length variation on resistance for cross-coupled topology with all threshold voltages and at a constant width w=2μ and number of fingers nf=1.

Effect of Number of Fingers Variation

The influence of the number of fingers on effective resistance is not as pronounced as it was for variations in length and width. This can be seen in Figure 3.12, which shows that with an increased number of fingers, the effective resistance decreases across all topologies. Furthermore, the impact of threshold voltage is the same as it was for the variation in width.

3.4.3 Effective Capacitance

This section details how effective capacitance is influenced by variations in width, length, and number of fingers.

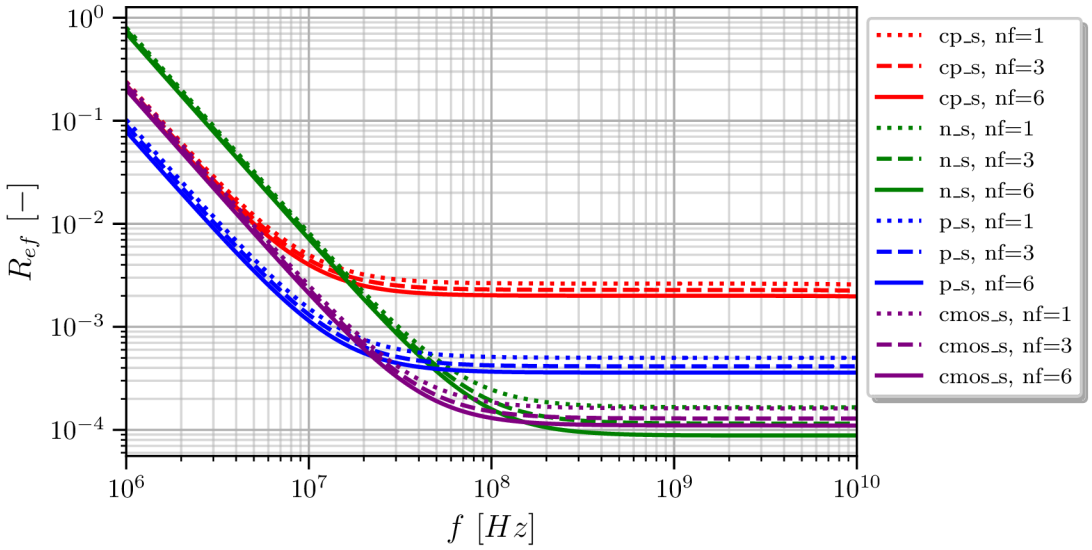

Figure 3.12: Effect of changing the number of fingers on resistance for all topologies at standard threshold voltage and at a constant length l=7λ and width w=2μ.

Effect of Width and Length Variation

The influence of changing width and length on the effective capacitance is as expected: with greater width and length, the area increases, leading to an increase in effective capacitance. This can be seen for width variation in Figure 3.13 and for length variation in Figure 3.14.

From these two figures, it can be observed that, up to a certain frequency, the CMOS and Cross-Coupled topologies have the same capacitance when the area is the same. Beyond a certain frequency, the capacitance of the Cross-Coupled topology starts to decrease sharply for greater lengths and widths. This decrease is much more pronounced for longer channel lengths than for transistor widths. Also, a similar decrease can be noticed for the CMOS topology in Figure 3.14 at length l=7λ. Additionally, from both of these figures, it can be seen that the NMOS topology has a higher capacitance than the PMOS topology for the same area.

Effect of Number of Fingers Variation

The effect of the number of fingers can be clearly observed in Figure 3.15. From the figure, it can be seen that with a higher number of fingers, the capacitance slightly increases.

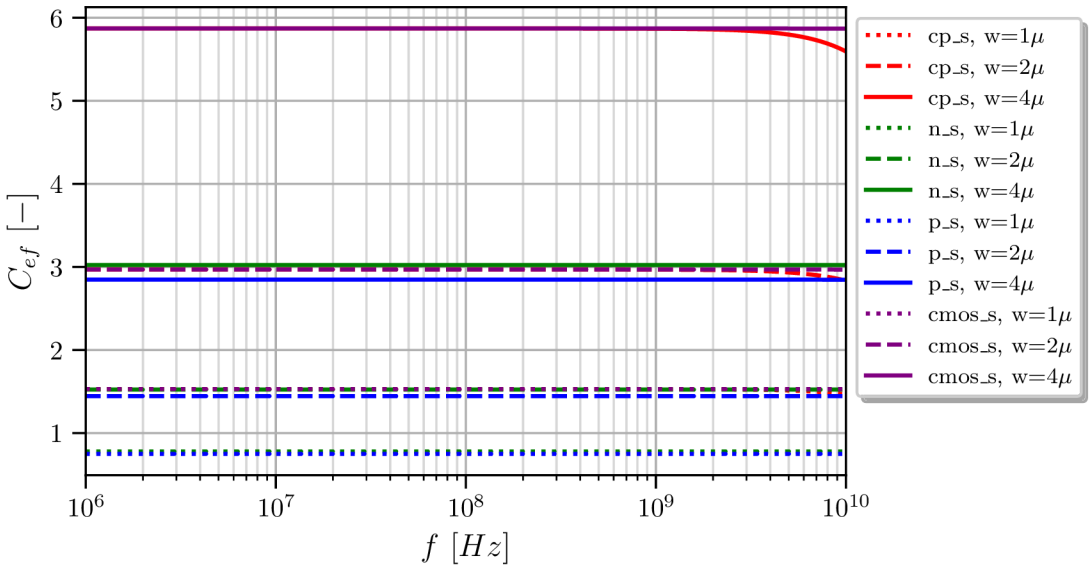

Figure 3.13: Effect of width variation on capacity for all topologies at standard threshold voltage and at a constant length l=2λ and number of fingers nf=1.

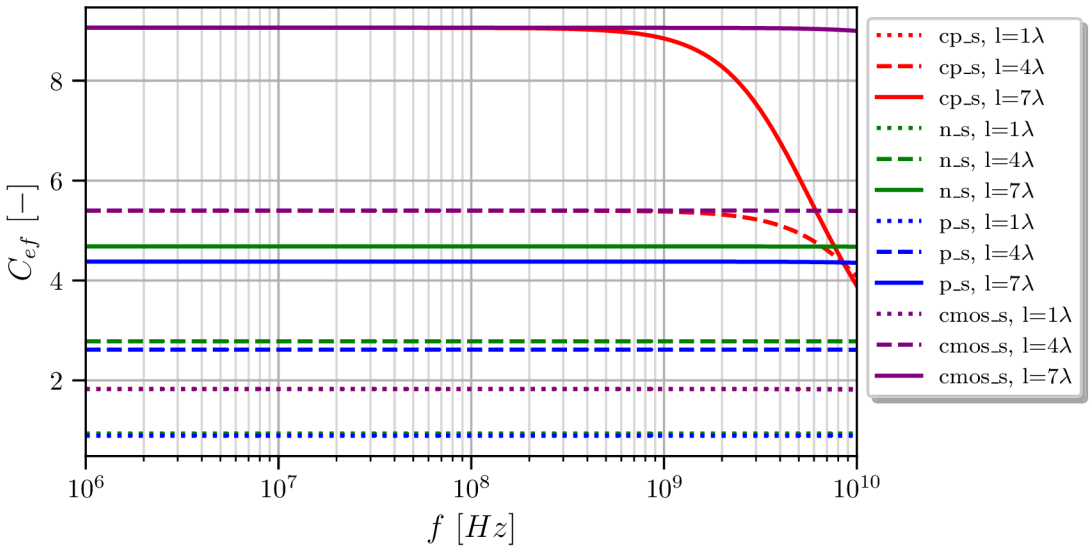

Figure 3.14: Effect of length variation on capacity for all topologies at standard threshold voltage and at a constant width w=2μ and number of fingers nf=1.

Effect of Threshold Voltage Variations

The effect of different threshold voltages on this capacitance is minimal, but it can still be seen that capacitance increases with higher threshold voltage. This can be observed by looking at the Figure 3.16.

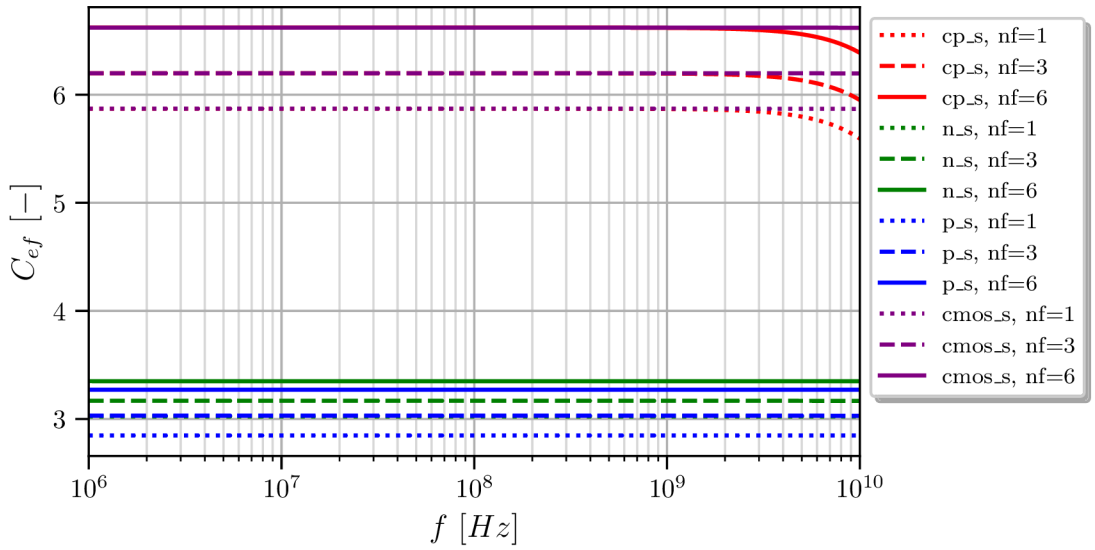

Figure 3.15: Effect of changing the number of fingers on capacity for all topologies at standard threshold voltage and at a constant length l=7λ and width w=2μ.

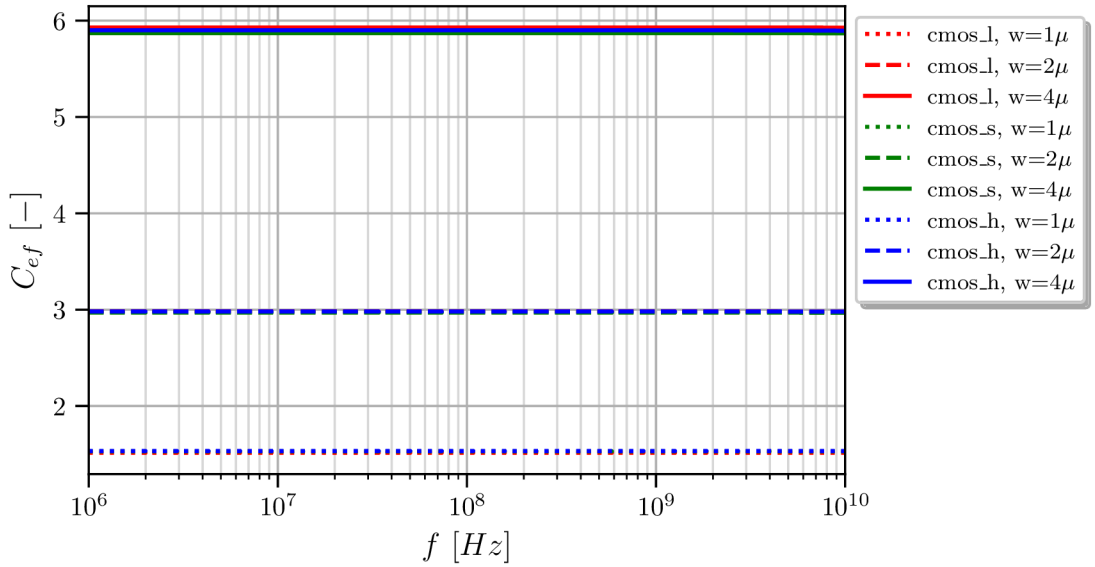

Figure 3.16: Effect of width variation on capacity for CMOS topology with all threshold voltages and at a constant length l=2λ and number of fingers nf=1.

3.4.4 Leakage

The effects of varying width, length, and number of fingers on leakage current leakage are examined in this section.

Effect of Width and Length Variation

The effect of changing width and length on leakage for all topologies at standard threshold can be seen in figures 3.17 and 3.18, respectively. From these figures, it

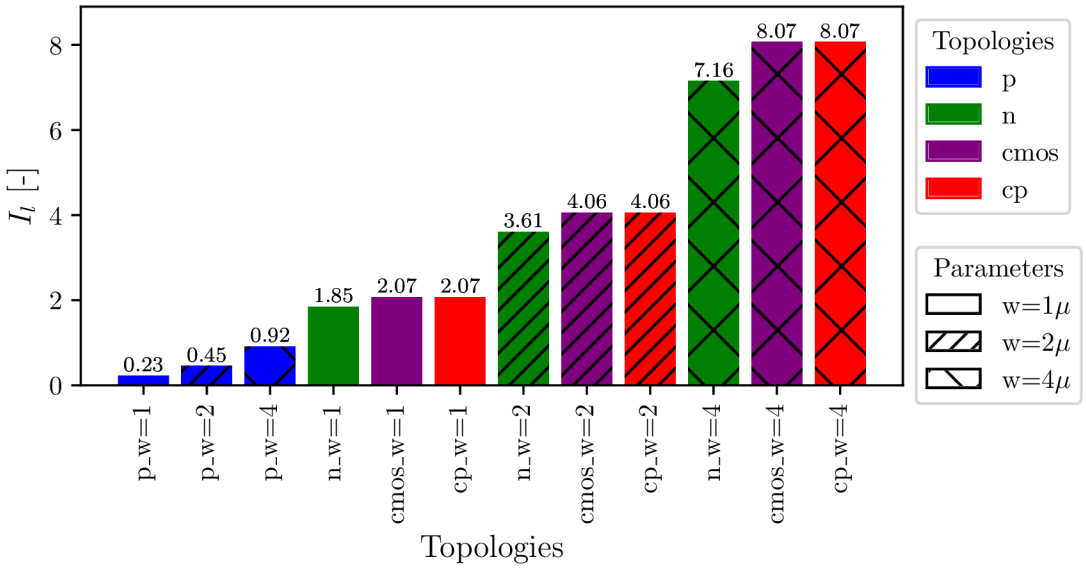

Figure 3.17: Effect of width variation on leakage for all topologies at standard threshold voltage and at a constant length l=2λ and number of fingers nf=1.

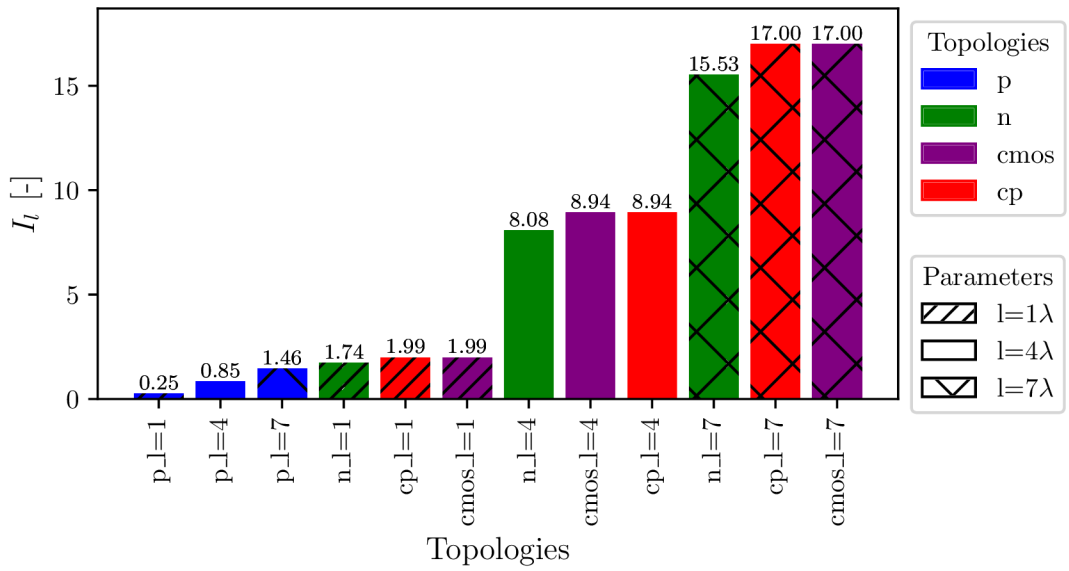

Figure 3.18: Effect of length variation on leakage for all topologies at standard threshold voltage and at a constant width w=2μ and number of fingers nf=1.

can be seen that for variations in widths and lengths, the PMOS topology has the

absolutely lowest leakage, while CMOS and Cross-Coupled have the highest leakage. However, it's important to remember that NMOS and PMOS are modeled as half-sized compared to CMOS and Cross-Coupled. Furthermore, it is evident that with larger areas, leakage increases for all topologies.

Effect of Number of Fingers Variation

The influence of changing the number of fingers on leakage can be observed for the standard threshold voltage in Figure 3.19. This figure makes it evident that leakage slightly increases with a larger number of fingers, but not significantly.

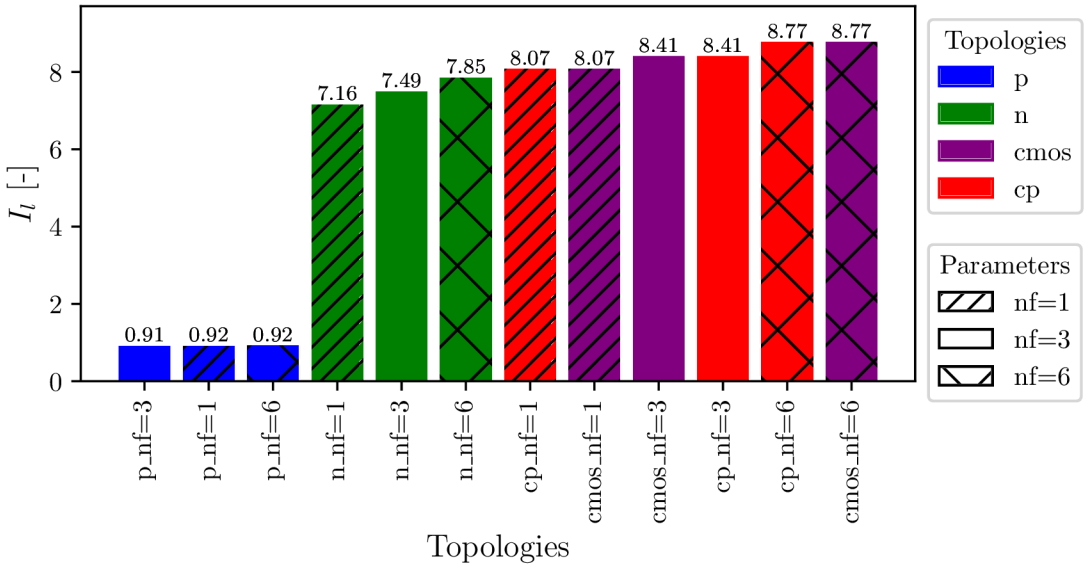

Figure 3.19: Effect of changing the number of fingers on leakage for all topologies at standard threshold voltage and at a constant length l=2λ and width w=4μ.

Effect of Threshold Voltage Variations

The differences among various threshold voltages are clearly visible for the NMOS topology with length variation in Figure 3.20. It can be seen that with a lower threshold voltage, the leakage increases, which is attributed to the increase in sub-threshold leakage.

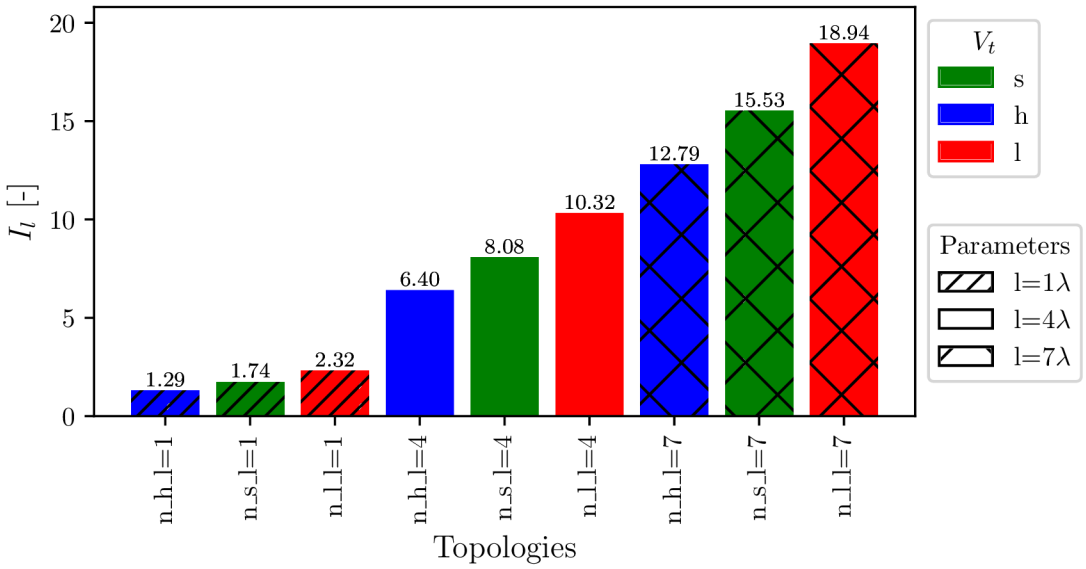

Figure 3.20: Effect of length variation on leakage for NMOS topology with all threshold voltages and at a constant width w=2μ and number of fingers nf=1.

4 Analysis of Selected Decoupling Capacitor Topologies

In this chapter, the simulation methodology will be described, followed by a discussion of the proposed layouts that meet specific selection criteria. These layouts are designed to optimize four critical parameters: capacity, quality factor, leakage, and a balanced consideration of the previous three. Serial resistance will not be examined separately in this section, as it is essentially incorporated in the quality factor.

For the practical layout design, a standard cell width of thirty-two grids was chosen. This uniform size ensures that all designed layouts occupy the same area, facilitating a more straightforward comparison of their performance. A standard threshold was also selected for all transistors so that the layouts and examined parameters could be compared with each other.

Based on the results from the previous section, the CP topology was selected for deeper analysis. This topology achieved the same results in terms of capacity and leakage as CMOS, but had a much worse quality factor and the capacity decreases at a lower frequency. However, a significant advantage over CMOS is ESD protection [20]. In CMOS topology, the gates of transistors are directly connected to the power supply voltage, and with a gate thickness of only 2 nm of SiO_2, the breakdown voltage is around five volts, which increases the risk of breakdown [21].

4.1 Simulation

This section will describe the simulation settings. The principle of obtaining the monitored parameters is the same as it was for a simple simulation from the netlist, as it was described in section 3.3. However, the simulation itself will be different. A transient analysis will be used instead of AC anylisis. This is due to the fact that the results from the AC analysis with a complete netlist with parasitic extraction were significantly inaccurate. In addition, process, voltage and temperature (PVT) variations were included in the analysis. The inclusion of PVT is critical to ensure that the simulation results accurately reflect the behavior of the circuit under various operating conditions. The PVTs that were used can be seen in Table 4.1.

In the Table 4.1 it can be seen that the PVTs were chosen to represent the extremes and normal conditions. The normal condition is represented by the TYP process, 1.2 V voltage, and 25 °C temperature. The extreme conditions are represented by the 'FFF5' process, 1.3069 V voltage, and −55 °C temperature. The other extreme conditions are represented by the 'SSF5' process, 1.0910 V voltage, and

Table 4.1: List of PVTs used in the simulation.

Config name	Process	Voltage [V]	Temperature [°C]
fff5s_1p3069_m55c	FFF5	1.3069	-55
fff5s_1p3069_200c	FFF5	1.3069	200
tt_1p2000_m55c	TYP	1.2000	-55
tt_1p2000_25c	TYP	1.2000	25
tt_1p2000_200c	TYP	1.2000	200
ssf5s_1p0910_m55c	SSF5	1.0910	-55
ssf5s_1p0910_200c	SSF5	1.0910	200

200 °C temperature. The extreme conditions were chosen to represent the worst-case scenario, where the normal conditions were chosen to represent the most common operating conditions. The 'FFF5' process designation represents a contidion in manufacturing where the transistors deviates towards faster performance, while the 'SSF5' process designation represents a condition where the transistors deviates towards slower performance. The '5' in both designations refers to a 5 sigma deviation from the mean. That means that the parameters fell well outside from the normal distribution of the process parameters.

As can be seen from Table 4.1, the extreme values of voltage do not exactly deviate by ±10 %. This is because it is necessary to superimpose an AC voltage on the DC voltage supply. If the values were precisely within ±10 %, they would exceed this limit with the added AC voltage. Therefore, these values were set to approximately ±9 %. Furthermore, the V_{dc} voltage was adjusted from the original 10 %, as shown in Figure 3.4, to ±1 %, which is evident in Figure 4.1. This change in amplitude does not affect the results, as this voltage is canceled out in the formulas later on. With this adjustment, the voltage does not exceed the voltage limits.

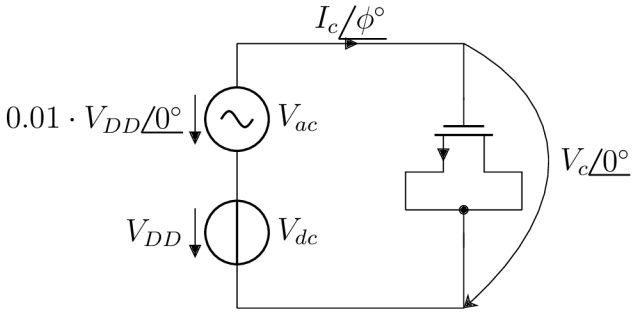

Figure 4.1: Circuit for simulation of selected parameters.

With the change from AC analysis to transit analysis, the way of finding and calculating the monitored parameters has changed a bit. The main difference is

that it is not possible to work directly with complex numbers, but the phase shift between current and voltage has to be determined first. First, it is necessary to find out at what point in time the voltage and current pass through the mean value, from which the phase shift between these waveforms can then be calculated:

$$\phi = 2 \cdot \pi \cdot f_{ac} \cdot (t_{I_{cm}} - t_{V_{cm}}), \tag{4.1}$$

where ϕ is the phase shift, f_{ac} is the frequency at which the measurement is made, $t_{I_{cm}}$ and $t_{V_{cm}}$ are the time points of current and voltage passage through the mean value respectively. Then from this phase shift, the complex impedance $\hat{Z}$ can be easily calculated by:

$$\hat{Z} = \frac{V_c \angle 0°}{I_c \angle \phi°} \tag{4.2}$$

then the rest of the monitored parameters can be calculated using the equations from section 3.3.

4.1.1 Simulation Methodology

For the simulation, the Spectre simulator was used. The same methodology was used as in the previous chapter in section 3.3. The spectre netlist remained almost the same, but the simulation was changed to transient analysis. The modified netlist can be seen in Listing 4.1. Complete netlists can be found in appendix A.2.

Listing 4.1: Example of a Spectre netlist used for simulations

```
global 0
//Parametres
parameters DC_voltage=1.2 \
AC_magnitude=0.01*DC_voltage \
Amplitude=0.01*DC_voltage \
temp=25 \
freq_ac = 1M \
//netlist
VDC(v_ac 0) vsource dc=DC_voltage type=dc
VAC(decap v_ac) vsource dc=0 mag=AC_magnitude type=sine ampl=
   Amplitude freq=freq_ac
Xdecap(decap 0) {pex_topology}
//simulations
dc1 dc
tran1 tran stop=2m errpreset=conservative
```

In Listing 4.2, the MDL code that was used for simulations can be seen. This code was utilized for simulations of all topologies. It is possible to observe in this

code that all PVTs described in Table 4.1 were used. Subsequently, a file is created in which data will be stored. This is followed by foreach loops for each PVT, in which simulations are conducted. For each PVT, a dc analysis "leak_dc" is performed, and then a transient analysis "tran_sim" is conducted in the loop for frequencies from 100 kHz to 10 THz. The complete code including definitions of the analyses can be found in Appendices A.4 and A.6. Finally, all the data, as in the previous chapter, was normalized, therefore it is possible to compare them only relatively. The normalization was carried out by dividing all the data by some value known only to the author of the work.

Listing 4.2: Eample of MDL scrip used for simulations.

```
analysis corner,pvts[]={ {List of PVTS from 4.1 }
print fmt("%s,%s,%s,%s\n", "F", "R", "C", "Q") to="data/{
    topology_name}.raw"
foreach corner from pvts onerror=continue {
    run corner
    run leak_dc(out=fmt("data/leak_all.raw"))
    foreach freq_ac from swp(start=100k, stop=10T, log=720) {
        run tran_sim(leak=leak_dc->leakdc ,out=fmt("data/{
    topology_name}.raw"))
    }
    print fmt("%s,%s,%s,%s\n", "F", "R",  "C", "Q") addto="
    data/{topology_name}.raw"
}
```

4.2 Iterative Layout Design

In this section, the methods used to design the layouts will be described. Figure 4.2 shows the methodology that was used for the layout designs. Initially, the goal was set to optimize the layouts for capacity, quality, leakage current, and balanced optimization. Then, the first version of the layout was created, which underwent parasitic extraction and simulation. If the layout met the expectations, it was added to the final layouts. If not, it was modified and the entire process was repeated. Several such iterations were carried out until results were achieved that were considered satisfactory. Overall, more than sixty different layouts were designed, and their description would be too extensive for this work. For this reason, only the layouts selected for the final comparison will be described.

Several tools were used in the design process. Cadence Virtuoso, a tool commonly used for integrated circuit design, was used for the design. The Siemens Calibre

xACT 3D tool, was used to extract parasitic element. Furthermore, the design was based on the simulation results that were performed in the previous chapter 3.4.

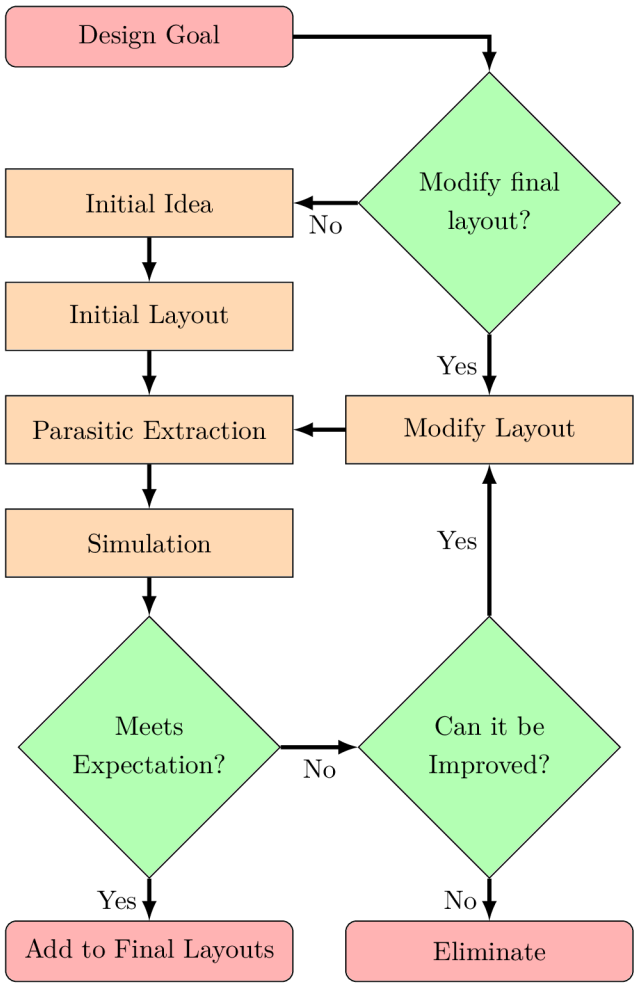

Figure 4.2: Methodology for iterative layout design.

4.3 Final Layouts

In this section, the layouts selected for the final comparison will be described. Each layout will be discussed in terms of its optimization and the results of the simulations. Given the nearly unlimited possibilities in layout design, it would be possible to optimize the cells almost indefinitely, thereby achieving results that could be better than those described in this work. However, due to limited time and resources, the layouts were optimized only to a certain extent. Further optimizations would be possible, but their improvements would be negligible compared to the demands on time and resources. In this work, due to NDA, the final layouts have been redrawn

Table 4.2: Results of Simulations for Selected Layouts.

Layout	C_{max} [−]	$f_{C_{1/2}}$ [GHz]	Q_{max} [−]	$f_{Q_{max}}$ [MHz]	I_L/C_{max} [−]
Capacity	1.67	10.76	4.73	1.61	19.8
Q-factor	0.80	152.76	11.79	12.25	40.6
Leakage	0.92	7.59	4.2	1.53	16.2
Balanced	1.29	46.13	7.51	5.55	33.9

from Virtuoso and thus do not precisely replicate the original simulated layouts. Nevertheless, the main ideas are maintained, and the differences are minimal.

The Table 4.2 presents the results of simulations for selected layouts, including the following values: C_{max}, which is the maximum capacity, $f_{C_{1/2}}$, the frequency at which the capacity drops to half, Q_{max}, which is the maximum quality factor, and $f_{Q_{max}}$, the frequency at which the quality factor is highest. Additionally, the ratio of leakage to capacity I_L/C_{max} is provided, which better characterizes the leakage current than the leakage current itself and allows for comparison among layouts, as layouts with higher capacity may also exhibit higher leakage currents. The colors in the table indicate which layout is best for a given parameter. Green signifies the best value, red the worst, and yellow an intermediate value. It is evident that the specific layouts excel in the parameters for which they were optimized. The layout with the best capacity is optimized for capacity, the layout with the best results in frequency parameters is optimized for quality factor, and the layout with the best leakage current is optimized for leakage current. The layout optimized for balance shows intermediate values in all parameters except for the ratio of leakage current to capacity, where it has the worst outcome.

The graphs displayed in figures from 4.3 to 4.4 show the simulation results for all layouts. These graphs illustrate how capacity and quality factor vary with frequency. The values depicted in these graphs correspond to the data presented in Table 4.2.

4.3.1 Optimized for Capacity

The Figure 4.5 shows a layout that has been optimized for capacity. This layout was designed to maximize capacity while also focusing on maintaining a frequency at which the capacity falls to half. In this work, this frequency will be referred to as the cutoff frequency. To achieve these parameters, three minimal NMOS transistors were used along with three minimal PMOS transistors above them, enabling efficient space utilization. These transistors are crucial for improving the cutoff frequency. The number three was determined iteratively; adding another transistor no longer had a significant impact. The rest of the cell was filled with large PMOS transistors.

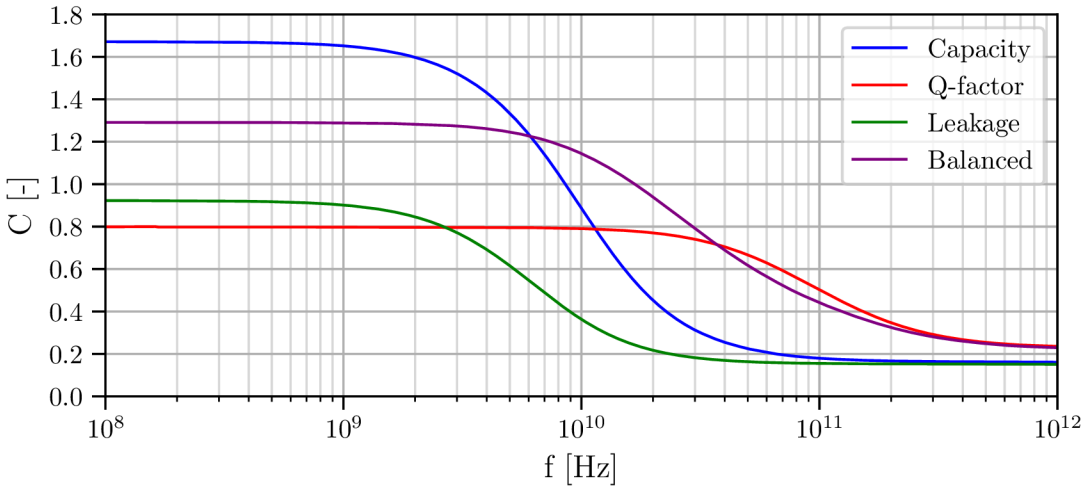

Figure 4.3: Trend of capacity as a function of frequency for all layouts.

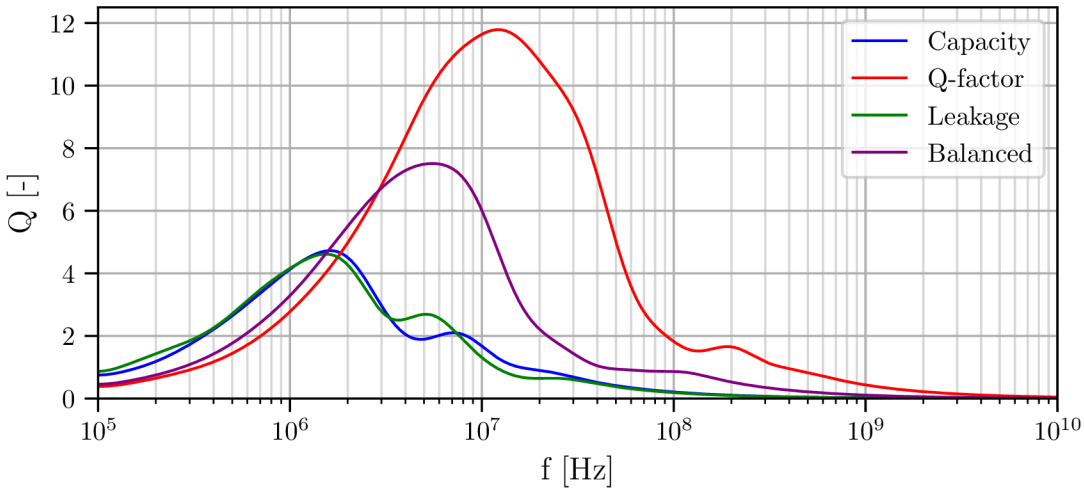

Figure 4.4: Trend of quality factor as a function of frequency for all layouts.

In cases where higher capacity is needed and frequency dependence is not a concern, it is possible to use just one minimal NMOS instead of three ones and also replace the minimal PMOSs with a single large one. This approach increases the capacity but at the cost of worse frequency dependence. Alternatively, reducing the number of fingers increases capacity as it frees up space that would otherwise be occupied by contacts. With the required width between transistors for contacts being χ nm, the area decreases according to the following equation:

$$A_{nf_x} = A_{nf_0} - \chi \cdot (NF - 1), \tag{4.3}$$

where A_{nf_x} represents the area with x fingers, A_{nf_0} is the area with zero fingers, and NF denotes the number of fingers.

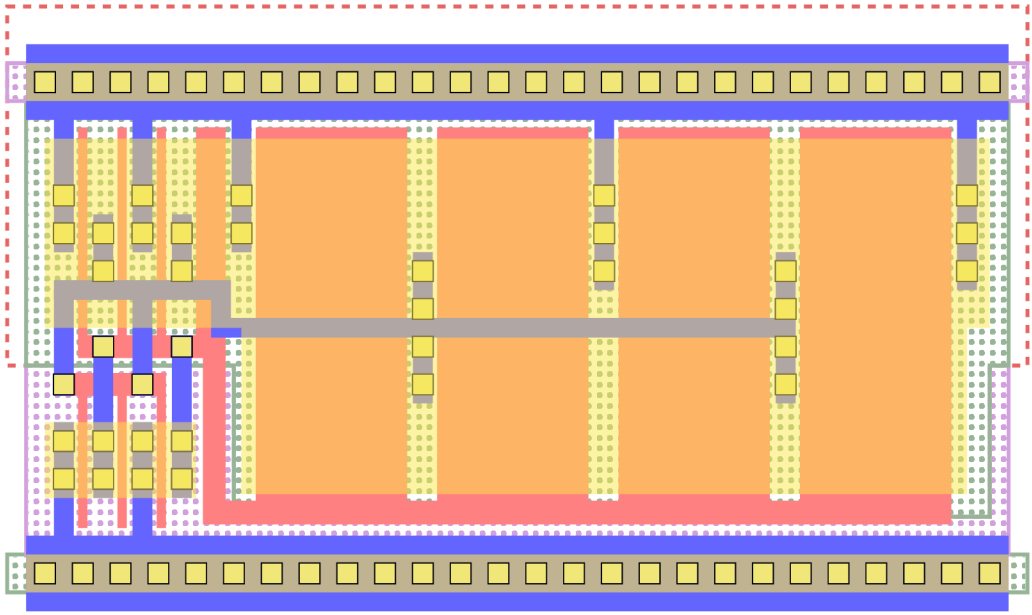

Figure 4.5: Layout optimized for capacity.

The values of C_{max} and $f_{C_{1/2}}$ for various PVTs are displayed in Table 4.3. From this table, it is evident that the layout achieves higher capacity at lower temperatures and with fast transistors at higher voltages. The dependency of capacity on frequency for various PVTs, from which these values were obtained, can be seen in Figure 4.6

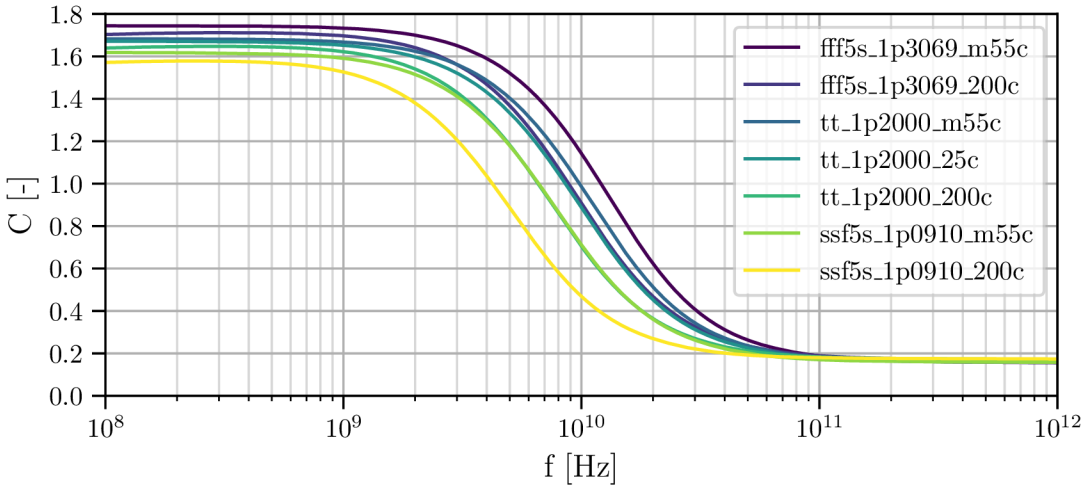

Figure 4.6: Dependency of capacity on frequency for a layout optimized for capacity, under various PVT conditions.

Table 4.3: The effect of different PVTs on the capacity of the layout optimized for capacity.

PVT	C_{max} [−]	$f_{C_{1/2}}$ [GHz]
fff5s_1p3069_m55c	1.74	14.45
fff5s_1p3069_200c	1.71	10.76
tt_1p2000_m55c	1.68	12.25
tt_1p2000_25c	1.67	10.76
tt_1p2000_200c	1.65	8.47
ssf5s_1p0910_m55c	1.62	8.79
ssf5s_1p0910_200c	1.58	5.86

4.3.2 Optimized for Leakage

In Figure 4.7, the layout optimized for leakage current can be seen. At first glance, it appears very similar to the layout optimized for capacitance. However, it includes only one minimum-sized NMOS transistor, as NMOS transistors have significantly higher leakage. The main idea of these PMOS transistors is to maximize the contribution of fringing capacitance, as leakage does not increase with this capacitance. This is because a larger area means greater capacitance but also greater leakage through the gate.

Figure 4.8 shows the dependence of leakage current on frequency for all PVTs. The values of I_L and I_L/C_{max} for various PVTs can be seen in Table 4.4. From this table, it is apparent that the layout exhibits the highest leakage current at higher temperatures and with faster transistors at higher voltages. It is also evident that the leakage current increases sharply with rising temperature. The lowest leakage is at PVT "ssf5s_1p0910_m55c", which represents slow transistors at low temperatures. It can be noted that this value is 1, because all leakage values were normalized by this value.

4.3.3 Optimized for Quality Factor

Figure 4.9 illustrates a layout optimized for the highest quality factor, accommodating the maximum number of NMOS and PMOS transistors with the shortest possible channel length within the cell. As mentioned in section 3.4.1, PMOS transistors typically provide higher quality factor values, whereas NMOS transistors achieve these values at higher frequencies. Although results from this section might suggest that using predominantly NMOS transistors and only one PMOS transistor would be feasible, such an approach would significantly increase the series resistance

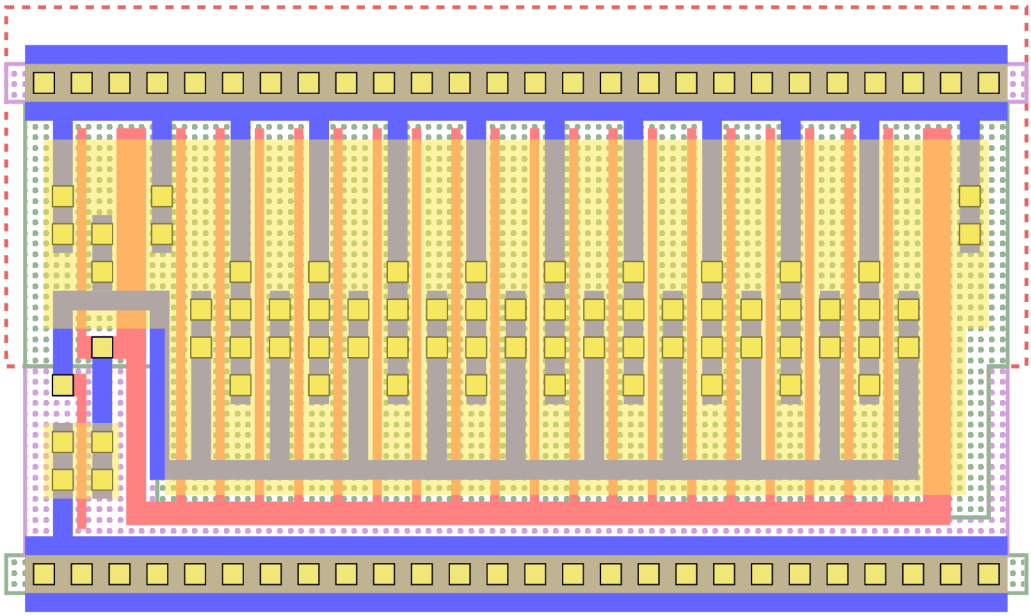

Figure 4.7: Layout optimized for leakage.

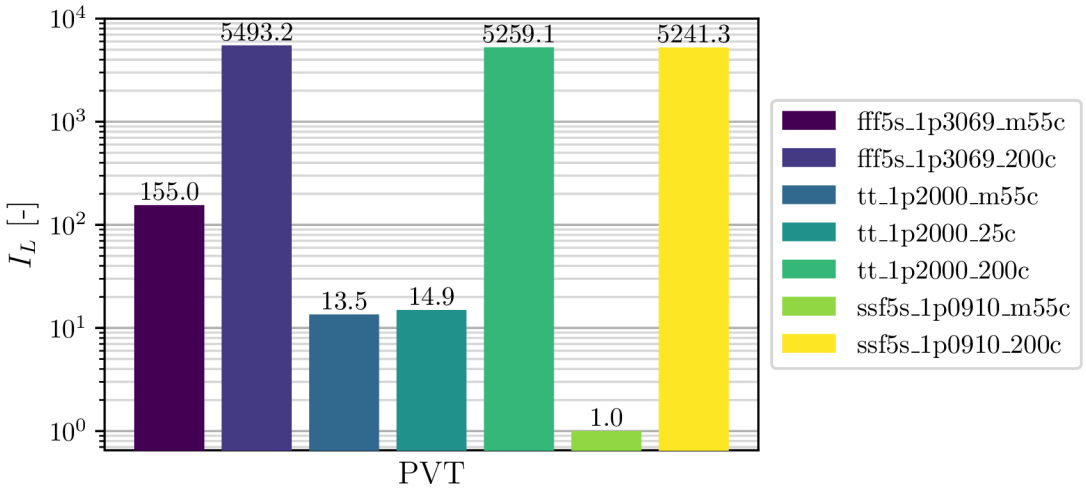

Figure 4.8: Leakage for a layout optimized for leakage, under various PVT conditions.

and decrease the overall quality factor value. Therefore, the best decision was to use an equal number of NMOS and PMOS transistors.

Figure 4.10 shows how the quality factor varies with frequency for different PVT conditions. From Table 4.5, it is evident that the layout achieves the highest quality factor values at lower temperatures and with slow transistors at lower voltages, but with the lowest frequency $f_{Q_{max}}$. Conversely, the highest values of $f_{Q_{max}}$ occur at

Table 4.4: The effect of different PVTs on the leakage of the layout optimized for leakage.

PVT	C_{max} [−]	I_L [−]	I_L/C_{max} [−]
fff5s_1p3069_m55c	0.94	155.0	165.0
fff5s_1p3069_200c	0.94	5493.2	5874.6
tt_1p2000_m55c	0.92	13.5	14.6
tt_1p2000_25c	0.92	14.9	16.2
tt_1p2000_200c	0.92	5259.1	5739.2
ssf5s_1p0910_m55c	0.90	1.0	1.1
ssf5s_1p0910_200c	0.89	5241.3	5896.6

lower temperatures with fast transistors at higher voltages. A similar trend can also be observed for the frequency $f_{C_{1/2}}$ at which the capacity drops to half.

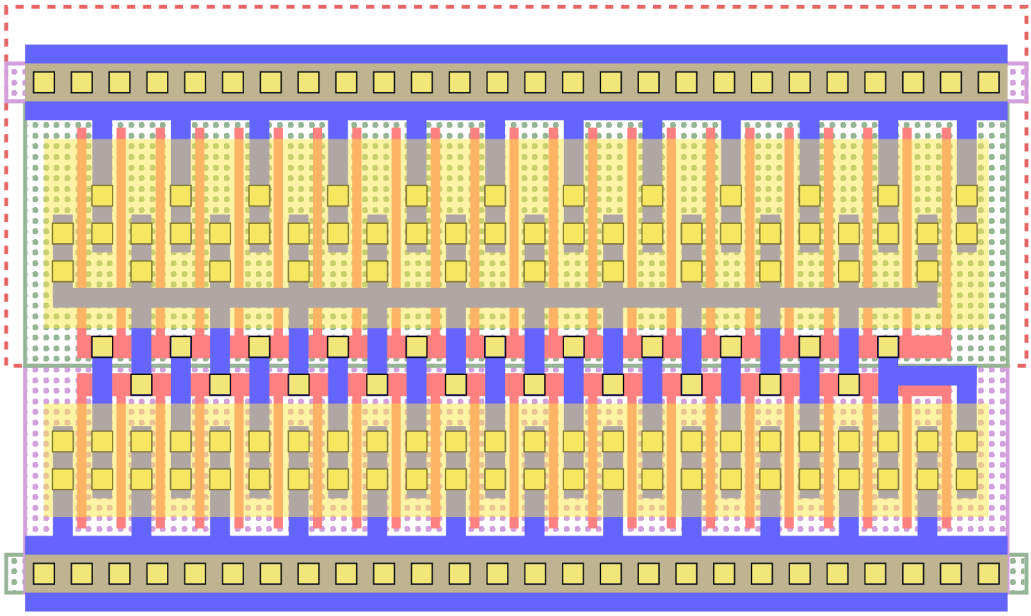

Figure 4.9: Layout optimized for quality factor.

4.3.4 Balanced Optimization

The last layout to be discussed is the balanced optimization layout. Shown in Figure 4.11, this layout was designed to achieve medium values for all parameters, resembling a combination of the other layouts. It includes the maximum number of NMOS transistors with the minimum channel length, which unfortunately worsens the leakage but significantly enhances the frequency dependencies. It also consists

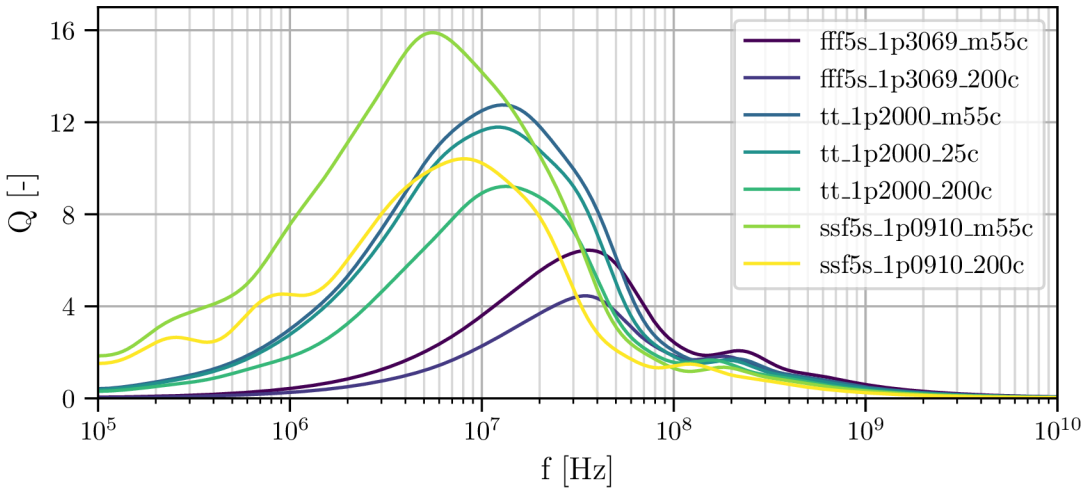

Figure 4.10: Dependency of quality factor on frequency for a layout optimized for quality factor, under various PVT conditions.

Table 4.5: The effect of different PVTs on the quality factor of the layout optimized for quality factor.

PVT	C_{max} [−]	$f_{C_{1/2}}$ [GHz]	Q_{max} [−]	$f_{Q_{max}}$ [MHz]
fff5s_1p3069_m55c	0.81	212.81	6.44	36.31
fff5s_1p3069_200c	0.81	164.44	4.46	34.36
tt_1p2000_m55c	0.80	173.78	12.75	12.94
tt_1p2000_25c	0.80	152.76	11.79	12.25
tt_1p2000_200c	0.80	124.74	9.21	13.43
ssf5s_1p0910_m55c	0.78	115.88	15.90	5.55
ssf5s_1p0910_200c	0.78	84.72	10.42	8.02

of five PMOS transistors with minimum channel lengths and three large PMOS transistors. As seen in Table 4.2, this layout presents median values across all parameters except for the ratio of leakage current to capacitance, where it ranks as the second worst.

The dependence of parameters on frequency and PVT can be seen in Figures 4.12, 4.13, and 4.14. Furthermore, Tables 4.6 and 4.7 provide values for all the monitored parameters across different PVT settings. The trend of parameter changes with respect to PVT is consistent with that observed in other layouts.

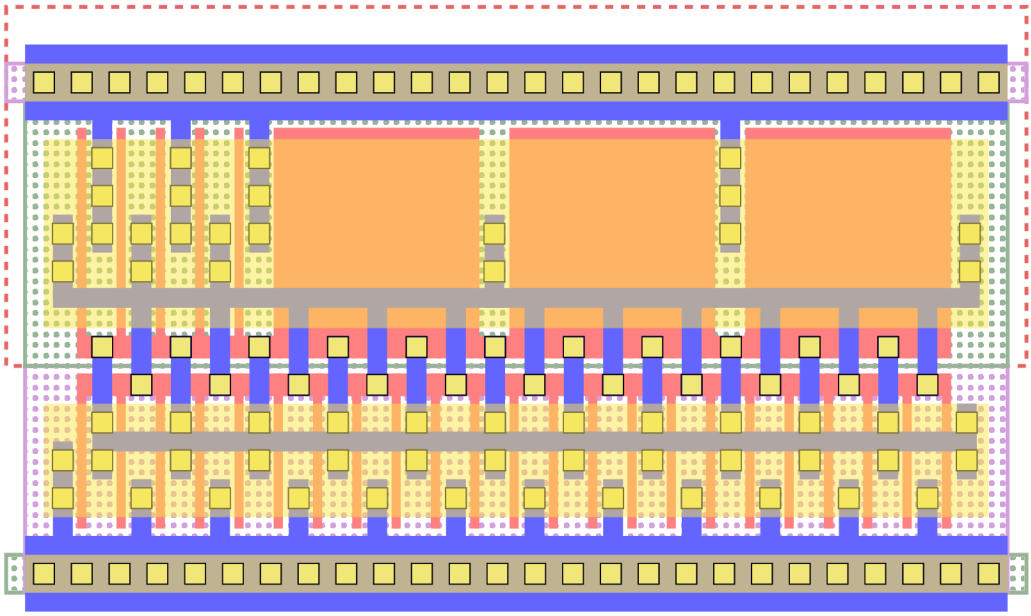

Figure 4.11: Layout optimized for balance between capacity, quality factor, and leakage.

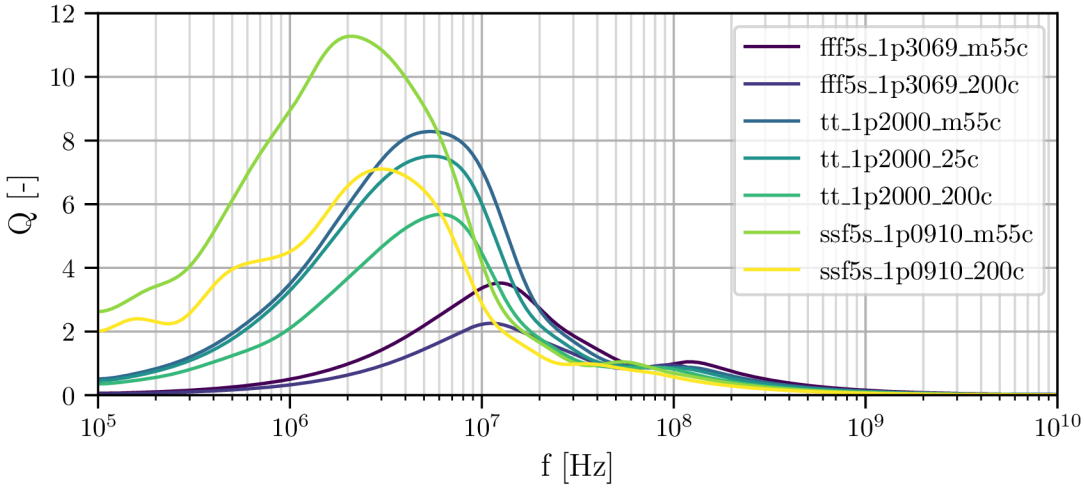

Figure 4.12: Dependency of quality factor on frequency for a layout optimized for balance, under various PVT conditions.

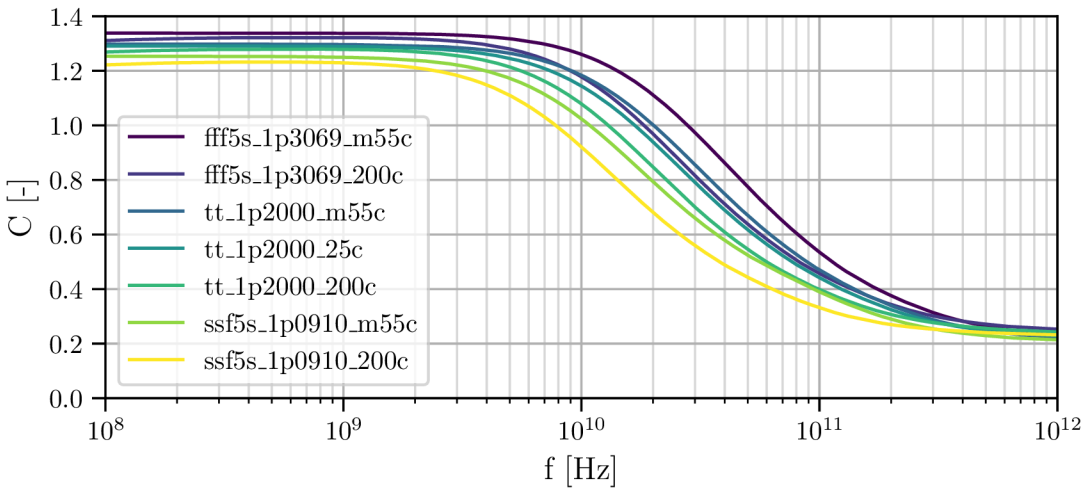

Figure 4.13: Dependency of capacity on frequency for a layout optimized for balance, under various PVT conditions.

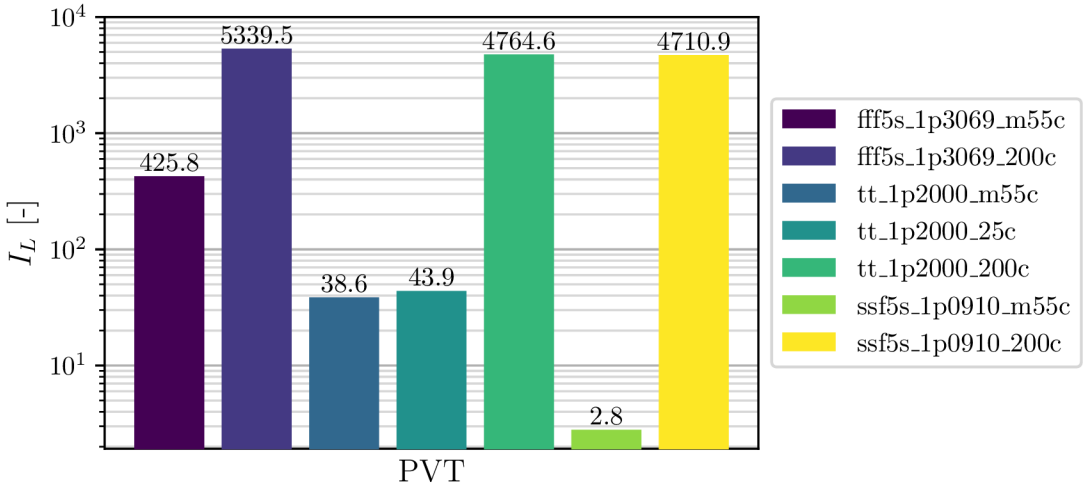

Figure 4.14: The effect of different PVTs on the leakage of the layout optimized for balance.

Table 4.6: The effect of different PVTs on the monitored parameters of the layout optimized for balance (fist part).

PVT	$f_{C_{1/2}}$ [GHz]	Q_{max} [−]	$f_{Q_{max}}$ [MHz]
fff5s_1p3069_m55c	66.68	3.52	12.47
fff5s_1p3069_200c	46.99	2.26	11.17
tt_1p2000_m55c	53.46	8.28	5.35
tt_1p2000_25c	46.13	7.51	5.55
tt_1p2000_200c	36.31	5.68	6.08
ssf5s_1p0910_m55c	33.73	11.28	2.09
ssf5s_1p0910_200c	24.66	7.11	3.02

Table 4.7: The effect of different PVTs on the monitored parameters of the layout optimized for balance (second part).

PVT	C_{max} [−]	I_L/C_{max} [−]
fff5s_1p3069_m55c	1.34	318.0
fff5s_1p3069_200c	1.32	4037.3
tt_1p2000_m55c	1.30	29.8
tt_1p2000_25c	1.29	34.0
tt_1p2000_200c	1.28	3722.9
ssf5s_1p0910_m55c	1.25	2.2
ssf5s_1p0910_200c	1.23	3822.0

Conclusion

The study commenced by examining 65 nm CMOS technology, with a focus on MOS structures and MOSFET transistors, which are essential for understanding decoupling capacitors. It also detailed the properties of digital standard cells and their role in chip design, including their physical design aspects.

Subsequently, various types of noise expected on the chip were introduced. Decoupling capacitors were suggested as a potential solution. Four suitable topologies for use within standard cells—NMOS, PMOS, CMOS, and Cross-Coupled—were presented. For each topology, the study investigated the effects of changes in basic transistor parameters such as width W, length L, and number of fingers NF, on quality factor Q, effective resistance R_{ef}, capacitance C_{ef}, and leakage I_L. A circuit for simulating these parameters was proposed, along with formulas for their calculation. Python scripts were created to automate the simulation process, and the resulting data was normalized due to a signed NDA concerning the 65 nm technology. However, this did not impact their evaluation, as absolute numbers are not necessary for assessing the examined effects.

The analysis of the individual topologies revealed that the PMOS topology exhibits the lowest leakage while achieving the highest quality factor, although its capacitance is smaller compared to the NMOS topology of the same area. The NMOS quality factor reaches the highest values at higher frequencies. The CMOS topology, which combines the features of NMOS and PMOS, exhibits behavior lying between these two. The Cross-Coupled topology showed the lowest quality factor values, with its capacity comparable to the CMOS topology.

It was observed that increasing the length and width parameters decreases the quality factor and maximum frequency f_{Qmax} while increasing the capacitance. Conversely, a larger number of fingers led to a higher quality factor and maximum frequency but had minimal effect on the capacitance and leakage. A lower threshold voltage was found to improve the maximum frequency of f_{Qmax} but increase the leakage current.

In the final part, the Cross-Coupled topology was selected for in-depth analysis due to its significantly better ESD properties compared to other topologies, which is particularly crucial for 65 nanometer technology given the low breakdown voltage of the gate oxide. The goal was to create four layouts in a standard digital cell, each focusing on the optimization of specific parameters, namely capacitance, quality factor, leakage current, and one aimed at balancing these three parameters. These layouts were created iteratively, starting with initial designs that were gradually modified based on findings from simulations with parasitic extraction from the layout combined with findings from simple netlist simulations. Over the course of the thesis,

more than sixty different variants were proposed. The simulation results were then normalized again.

Various cell layouts have been designed for integration into standard digital libraries based on their specific applications. For example, a cell designed to minimize leakage would suit a library focused for use in low power applications, where minimizing power loss is a priority. Next, a cell optimized for a high quality factor would be beneficial in libraries requiring high switching speeds and low interference associated with high switching frequencies. Additionally, a universally optimized cell would be ideal for a library that aims to optimize various parameters such as speed, area, and leakage. Furthermore, a cell optimized for capacity relative to area could be included in all libraries and used when there is a need to increase capacity without increasing the area. Ultimately, it is possible to add all these cells to a single library, allowing the designer to select the appropriate cell based on the specific application.

In studies following this analysis, it would be possible to conduct more simulations modeling more realistic applications of these cells. For instance, it could be explored how the behavior of the designed cells changes when they are surrounded by other cells with different functions or higher metal layers. Ultimately, their behavior could be physically validated if they were manufactured on a test chip and the simulated results were compared with real behavior. Alternatively, it would be possible to use these cells in an already produced design, which would be manufactured on a test chip, and subsequently, these two implementations could be physically compared.

Bibliography

[1] CHARANIA, T.; OPAL, A.; SACHDEV, M. Analysis and Design of On-Chip Decoupling Capacitors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. 2013, vol. 21, no. 4, pp. 648–658. Available from DOI: 10.1109/TVLSI.2012.2198501.

[2] WESTE, N.; HARRIS, D. *CMOS VLSI Design 4e: A circuits and systems perspective*. Paperback. PEARSON INDIA, 2010. ISBN 978-9332542884.

[3] BAKER, R.J. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Hardcover. Wiley-IEEE Press, 2019. ISBN 978-1119481515.

[4] *Apple introduces M2 Ultra — apple.com*. 2023. Available also from: https://www.apple.com/uk/newsroom/2023/06/apple-introduces-m2-ultra/. [Accessed 21-10-2023].

[5] KELLEHER, A.B. Celebrating 75 years of the transistor A look at the evolution of Moore's Law innovation. In: *2022 International Electron Devices Meeting (IEDM)*. IEEE, 2022. Available from DOI: 10.1109/iedm45625.2022.10019538.

[6] RUPP, K. *Microprocessor Trend Data* [https://github.com/karlrupp/microprocessor-trend-data]. 2023. [Accessed 25-10-2023].

[7] SHUKLA, P. *A Brief History of Process Node Evolution*. 2021. Available also from: https://www.design-reuse.com/articles/43316/a-brief-history-of-process-node-evolution.html. [Accessed 06-12-2023].

[8] NARENDRA, S. G.; CHANDRAKASAN, A. Kluwer Academic Publishers, 2006. ISBN 0387257373. Available from DOI: 10.1007/0-387-28133-9.

[9] SHAULY, E. N. CMOS Leakage and Power Reduction in Transistors and Circuits: Process and Layout Considerations. *Journal of Low Power Electronics and Applications*. 2012, vol. 2, no. 1. ISSN 2079-9268. Available from DOI: 10.3390/jlpea2010001.

[10] ROY, K.; PRASAD, S. *Low-power CMOS VLSI circuit design*. New York: Wiley, 2000. ISBN 9780471114888.

[11] MAITY, N. P.; MAITY, R.; BAISHYA, S. (eds.). *High-k gate dielectric materials: applications with advanced metal oxide semiconductor field effect transistors (MOSFETs)*. First edition. Palm Bay, Florida: Apple Academic Press, 2020. ISBN 9781771888431.

[12] FISHER, C.; BLANKENSHIP, R.; JENSEN, J.; ROSSMAN, T.; SVILICH, K. Optimization of standard cell libraries for low power, high speed, or minimal area designs. In: *Proceedings of Custom Integrated Circuits Conference*. 1996, pp. 493–496. Available from DOI: 10.1109/CICC.1996.510604.

[13] KRAL, V. Design of Multi-bit Pulsed Latches with Scan Input in CMOS ONK65 Technology. *Radioengineering*. 2023, vol. 32, no. 4, pp. 557–567. Available from DOI: 10.13164/re.2023.0557.

[14] CHEN, H.H.; SCHUSTER, S.E. On-chip decoupling capacitor optimization for high-performance VLSI design. In: *1995 International Symposium on VLSI Technology, Systems, and Applications. Proceedings of Technical Papers*. IEEE, [n.d.]. Available from DOI: 10.1109/vtsa.1995.524641.

[15] ARCHAMBEAULT, B.; BRENCH, C.; RAMAHI, O.M. *EMI/EMC Computational Modeling Handbook*. 2nd ed. Springer US, 2001. The Springer International Series in Engineering and Computer Science 630. ISBN 978-1-4613-5613-4, ISBN 978-1-4615-1557-9.

[16] VAISBAND, I.P. et al. *On-Chip Power Delivery and Management*. Springer International Publishing, 2016. Available from DOI: 10.1007/978-3-319-29395-0.

[17] MENG, X. Design and analysis of active and passive decoupling capacitors for on-chip power supply noise management. 2009. Available from DOI: 10.14288/1.0068219.

[18] CADENCE DESIGN SYSTEMS. *Spectre Circuit Simulator* [Software]. 2020. Version 20.1. Available also from: https://www.cadence.com/.

[19] CADENCE DESIGN SYSTEMS. *Spectre Circuit Simulator User Guide*. 2020. Version 20.1. Available also from: https://www.cadence.com/. [Accessed 15-12-2023].

[20] MENG, X.; SALEH, R.; ARABI, K. Novel Decoupling Capacitor Designs for sub- 90nm CMOS Technology. In: *7th International Symposium on Quality Electronic Design (ISQED'06)*. IEEE, [n.d.]. Available from DOI: 10.1109/isqed.2006.93.

[21] AMERASEKERA, A.; DUVVURY, Ch. *ESD in Silicon Integrated Circuits*. Wiley, 2002. ISBN 9780470846056. Available from DOI: 10.1002/0470846054.

Symbols and abbreviations

CMOS	Complementary Metal Oxide Semiconductor
MOSFET	Metal Oxide Semiconductor Field-Effect Transistor
NMOSFET	N-type Metal Oxide Semiconductor Field-Effect Transistor
NMOS	N-type MOSFET
PMOS	P-type MOSFET
PVT	Process, Voltage, Temperature
HDL	Hardware Description Language
RTL	Register-Transfer Level
EMI	Electromagnetic Interference
EST	Electrostatic Discharge
NDA	Non Disclosure Agreement
MDL	Measurement Description Language
STA	Static Timing Analysis

List of appendices

A Spectre

A.1 Spectre netlist

Listing A.1: Spectre netlist used for simulations

```
//Topology sim
simulator lang=spectre
include "path_to_models"
include "path_to_subciurkits - appendix A.5"
//Global variables
global 0
//Parameters
parameters Length=λ Width=μ \
DC_voltage=1.2 Amplitude=0.1*DC_voltage \
nf_p=1 temp =25
//netlist
VDC(v_ac 0) vsource dc=DC_voltage type=dc
VAC(decap v_ac) vsource type=sine ampl=Amplitude
Xdecap(decap 0) {topology} //one of different topologies
//Simulation settings
simopts options save=all audit=full \
    narrate=no error=yes info=yes \
    reltol=1e-3 vabstol=1e-6 iabstol=1e-12 \
    tnom=25 temp=temp scalem=1.0 \
    scale=1.0 gmin=1e-21 rforce=1 \
    maxnotes=5 maxwarns=5 digits=5 \
    cols=80 pivrel=1e-3 ckptclock=1800 \
    rawfmt=psfascii dochecklimit=no \
    dc_pivot_check=yes
simOptions options useprobes=yes
//simulations
ac1 ac start=1M stop=10G dec=30 annotate=sweep
dc1 dc
```

Listing A.2: Spectre netlist used for layout simulation with parasitic extraction.

```
//Layout sim
simulator lang=spectre
//models
include "path_to_models"
include "path_to_subciurkits_layouts"
```

```
include "PVT_definitions - appendix ??"
//Global variables
global 0
//Parameters
parameters DC_voltage=1.2 \
AC_magnitude=0.01*DC_voltage \
Amplitude=0.01*DC_voltage \
temp =25 \
freq_ac =1M \
//netlist
VDC(v_ac 0) vsource dc=DC_voltage type=dc
VAC(decap v_ac) vsource dc=0 mag=AC_magnitude type=sine ampl=
   Amplitude freq=freq_ac
Xdecap(decap 0) {topology_layout}
//Simulation settings
simopts options save=allpub audit=full \
    narrate=no error=yes info=yes \
    reltol=1e-3 vabstol=1e-6 iabstol=1e-12 \
    tnom=25 temp=temp scalem=1.0 \
    scale=1.0 gmin=1e-18 rforce=1 \
    maxnotes=5 maxwarns=5 digits=5 \
    cols=80 pivrel=1e-3 ckptclock=1800 \
    rawfmt=psfascii dochecklimit=no \
    dc_pivot_check=yes
simOptions options useprobes=yes
//simulations
dc1 dc
tran1 tran stop=2m errpreset=conservative
```

A.2 Spectre MDL

Listing A.3: MDL script for parameter sweep in Spectre simulations

```
alias measurement ac_sim_file{
    input string out="myfile.out"
    print fmt("%s,%s,%s,%s,%s\n", "F_{topology}", "Z_r_{
   topology}", "Z_i_{topology}",  "C_{topology}", "Q_{
   topology}") to=out
    run ac1
    print fmt("%g,%g,%g,%g,%g\n", xval(re(V(decap))), \
    re(V(decap)/I(Xdecap:1)), \
    im(V(decap)/I(Xdecap:1)), \
    abs(1/(im(V(decap)/I(Xdecap:1))*2*pi*xval(re(V(decap))))
   , \
    abs(im(V(decap)/I(Xdecap:1))/re(V(decap)/I(Xdecap:1))) \
     ) addto=out
}

alias measurement dc_sim_file{
    input string out="myfile.out"
    print fmt("%s\n", "Leakage") to=out
    run dc1
    print fmt("%g\n", I(Xdecap:1) ) addto=out
}

foreach Length from {1λ,2λ,4λ,5λ,7λ} {
    foreach Width from {1μ,2μ,4μ} {
        foreach nf_p from{1,3,6}{
            run ac_sim_file(out=fmt("data/{topology}=%S_w=%
   S_nf=%S.raw", Length, Width, nf_p))
            run dc_sim_file(out=fmt("data_leak/{topology}=%
   S_w=%S_nf=%S.raw", Length, Width, nf_p))
        }
    }
}
```

Listing A.4: MDL script for layout simulation with parasitic extraction.

```
alias measurement leak_dc{
    input string out="myfile.out"
    run dc1
```

```
    print fmt("%s,%g\n", "{topology_name}", I(Xdecap:1) )
   addto=out
    export real leakdc = I(Xdecap:1)
}
alias measurement tran_sim{
    input real leak
    input string out="myfile.out"
    real stop_time = 3/freq_ac
    run tran1(stop=stop_time)

    real current = -I(Xdecap:1)
    real magcur =  ( abs(max(current))+abs(min(current))/2 )
   - leak
    real vph = cross(sig=V(decap), dir=fall, n=1, thresh=
   DC_voltage, start=0)
    real ith = (max(current)+min(current))/2
    real iph = cross(sig=current, dir=fall, n=1, thresh=ith,
   start=0.01/freq_ac)
    real ph_diff = 2*pi*freq_ac*abs(iph-vph)
    real Z_r = -(0.01*DC_voltage * cos(ph_diff)) / magcur
    real Z_i = (0.01*DC_voltage * sin(ph_diff)) / magcur
    real C = 1/(Z_i * 2 * pi * freq_ac)
    real Q = Z_i/Z_r
    print fmt("%g,%g,%g,%g\n",freq_ac, Z_r, C, Q) addto = out
}

analysis corner,pvts[]={ alter_fff5s_1p3069_m55c,
   alter_fff5s_1p3069_200c, alter_tt_1p2000_m55c,
   alter_tt_1p2000_25c, alter_tt_1p2000_200c,
   alter_ssf5s_1p0910_m55c, alter_ssf5s_1p0910_200c }
print fmt("%s,%s,%s,%s\n", "F", "R", "C", "Q") to="data/{
   topology_name}.raw"
foreach corner from pvts onerror=continue
{
    run corner
    run leak_dc(out=fmt("data/leak_all.raw"))
    foreach freq_ac from swp(start=100k, stop=10T, log=720) {
        run tran_sim(leak=leak_dc->leakdc ,out=fmt("data/{
   topology_name}.raw"))
    }
```

```
35      print fmt("%s,%s,%s,%s\n", "F", "R",  "C", "Q") addto="
    data/{topology_name}.raw"
36  }
```

A.3 Topology subcircuits

Listing A.5: Spectre subcircuits used for simulations

```
////////////////////////////////CMOS
//LVT
subckt cmos_l (in_cmos_l 0)
    Tp_cmos_l (in_cmos_l 0 in_cmos_l in_cmos_l) lvtpfet l=
   Length w=Width \
    Tn_cmos_l (0 in_cmos_l 0 0) lvtnfet l=Length w=Width
ends
//SVT
subckt cmos_s (in_cmos_s 0)
    Tp_cmos_s (in_cmos_s 0 in_cmos_s in_cmos_s) pfet l=Length
    w=Width
    Tn_cmos_s (0 in_cmos_s 0 0) nfet l=Length w=Width
ends
//HVT
subckt cmos_h (in_cmos_h 0)
    Tp_cmos_h (in_cmos_h 0 in_cmos_h in_cmos_h) hvtpfet l=
   Length w=Width \
    Tn_cmos_h (0 in_cmos_h 0 0) hvtnfet l=Length w=Width
ends
////////////////////////////////CP
//LVT
subckt cp_l (in_lvt 0)
    Tp_cp_l (in_lvt net6 net4 in_lvt) lvtpfet l=Length w=
   Width
    Tn_cp_l (net6 net4 0 0) lvtnfet l=Length w=Width
ends
//SVT
subckt cp_s (in_s 0)
    Tp_cp_s (in_s net7 net2 in_s) pfet l=Length w=Width
    Tn_cp_s (net7 net2 0 0) nfet l=Length w=Width
ends
//HVT
subckt cp_h (in_hvt 0)
    Tp_cp_h (in_hvt net9 net8 in_hvt) hvtpfet l=Length w=
   Width
    Tn_cp_h (net9 net8 0 0) hvtnfet l=Length w=Width
ends
////////////////////////////////N
```

```
//LVT
subckt n_l (in_n_l 0)
    T_n_l (0 in_n_l 0 0) lvtnfet l=Length w=Width
ends
//SVT
subckt n_s (in_n_s 0)
    T_n_s (0 in_n_s 0 0) nfet l=Length w=Width
ends
//HVT
subckt n_h (in_n_h 0)
    T_n_h (0 in_n_h 0 0) hvtnfet l=Length w=Width
ends
////////////////////////////////P
//LVT
subckt p_l (in_p_l 0)
    T_p_l (in_p_l 0 in_p_l in_p_l) lvtpfet l=Length w=Width
ends
//SVT
subckt p_s (in_p_s 0)
    T_p_s (in_p_s 0 in_p_s in_p_s) pfet l=Length w=Width
ends
//HVT
subckt p_h (in_p_h 0)
    T_p_h (in_p_h 0 in_p_h in_p_h) hvtpfet l=Length w=Width
ends
```

A.4 PVT definitions

Listing A.6: PVT definitions for the MDL script

```
simulator lang=spectre
alter_fff5s_1p3069_m55c altergroup {
include "{path_to_models}" section=FFF5
parameters temp = -55
parameters DC_voltage = 1.3069
}

alter_fff5s_1p3069_200c altergroup {
include "{path_to_models}" section=FFF5
parameters temp = 200
parameters DC_voltage = 1.3069
```

```
}

alter_tt_1p2000_m55c altergroup {
include "{path_to_models}" section=typ
parameters temp = -55
parameters DC_voltage = 1.2000
}

alter_tt_1p2000_25c altergroup {
include "{path_to_models}" section=typ
parameters temp = 25
parameters DC_voltage = 1.2000
}

alter_tt_1p2000_200c altergroup {
include "{path_to_models}" section=typ
parameters temp = 200
parameters DC_voltage = 1.2000
}

alter_ssf5s_1p0910_m55c altergroup {
include "{path_to_models}" section=SSF5
parameters temp = -55
parameters DC_voltage = 1.0910
}

alter_ssf5s_1p0910_200c altergroup {
include "{path_to_models}" section=SSF5
parameters temp = 200
parameters DC_voltage = 1.0910
}
```

B Python scripts

B.1 Data normalization

Listing B.1: Script for data normalization

```
import os
import pandas as pd
import numpy as np
normaliying_name_length ={"1λ": "1" ,"2λ": "2", "4λ": "4", "5λ":"5", "7λ":"7"}
normaliying_name_width ={"1μ": "1" ,"2μ": "2", "4μ": "4"}
normal_data = pd.read_csv("cp_s_l=1λ_w=1μ_nf=1.raw")
columns_normal = ["Z_r_cp_s", "Z_i_cp_s", "C_cp_s", "Q_cp_s"]
lengths = ["1λ","2λ","4λ","5λ", "7λ"]
widths = ["1μ", "2μ", "4μ"]
nfs = ["1", "3", "6"] #will remain same
topologies = ["cp_l","cp_s","cp_h","n_l","n_s","n_h","p_l","p_s","p_h","cmos_l","cmos_s","cmos_h"]
for topology in topologies:
    for length in lengths:
        for width in widths:
            for nf in nfs:
                filename = f"{topology}_l={length}_w={width}_nf={nf}.raw"
                columns_to_normalize = ["Z_r_"+topology, "Z_i_"+topology, "C_"+topology, "Q_"+topology]
                data = pd.read_csv(filename)
                #all max values for normalization from reference topology
                max_values = normal_data[columns_normal].max()
                for column, max_value in zip(columns_to_normalize, max_values):
                    data[column] = data[column] / max_value
                data.to_csv(f"{topology}_l={normaliying_name_length[length]}_w={normaliying_name_width[width]}_nf={nf}.raw", index=False)
```

B.2 All topology simulation run

Listing B.2: Script for running Spectre for each topology

```
import os
import glob

class CircuitSimulator:
    def __init__(self, subcirkuit_find: list[str], sim_name: str):
        self.subcirkuit_find = subcirkuit_find
        self.subcirkuit_replace = self.subcirkuit_find[1:] + self.subcirkuit_find[0:1]
        self.sim_name = sim_name

    def _replace(self, find: str, replace: str, file_path: str) -> None:
        with open(file_path, 'r') as file:
            file_contents = file.read()
            file_contents = file_contents.replace(find, replace)
        with open(file_path, 'w') as file:
            file.write(file_contents)

    def _error_check(self) -> None:
        log_files = glob.glob('log/*.log')
        for filename in log_files:
            print(f"----------{filename}----------")
            with open(filename, 'r') as file:
                print("---WARNINGS----")
                for line in file:
                    if 'WARNING' in line:
                        print(f'{line.strip()}')
                print("----ERRORS----")
                for line in file:
                    if 'ERROR' in line:
                        print(f'{line.strip()}')
            print("\n\n")

    def run_simulation(self) -> None:
        sim_name = self.sim_name
        for i in range(len(self.subcirkuit_find)):
```

```
35              command = f"bsub -Is -q spectre spectremdl -batch
    {sim_name}.mdl" \
36              f"-design {sim_name}.scs +l log/{sim_name}{self.
    subcirkuit_find[i]}.log"
37              os.system(command)
38              self._replace(self.subcirkuit_find[i], self.
    subcirkuit_replace[i], f"{sim_name}.scs")
39              self._replace(self.subcirkuit_find[i], self.
    subcirkuit_replace[i], f"{sim_name}.mdl")
40          self._error_check()
41
42 if __name__ == "__main__":
43     subcirkuit_find = ["{List of all topologies}"]
44     sim = CircuitSimulator(subcirkuit_find, "sim")
45     sim.run_simulation()
```