

DESIGN AND OPTIMIZATION OF SENSOR INTERFACE CIRCUITRY AND
ALL-DIGITAL PLL IN 22 nm TECHNOLOGY WITH
 g_m/I_D METHODOLOGY BY USING PRECALCULATED LOOK-UP TABLES

AMIRHOSSEIN MOHAMMADPANAH

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE

GRADUATE PROGRAM IN
ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

JANUARY 2026

© Amirhossein Mohammadpanah, 2026

Abstract

Designing high-performance analog and mixed-signal circuits in advanced CMOS nodes like 22nm FD-SOI faces significant challenges due to low supply voltages, complex device physics, and the inaccuracy of traditional design models. This thesis presents a systematic design methodology based on the transconductance-to-current ratio (g_m/I_D) and pre-computed lookup tables (LUTs) to address these challenges, enabling efficient and accurate design across all transistor operating regions.

The methodology is first validated through comprehensive characterization of the 22nm FD-SOI process, demonstrating the robustness of the normalized design space, particularly its invariance to back-gate biasing. This framework is then applied to two demanding circuit designs. First, a low-noise, high-bandwidth single-ended folded cascode Operational Transconductance Amplifier (OTA) is systematically optimized. System-level analysis addresses the stability challenges associated with its target application in a Capacitive Transimpedance Amplifier (CTIA) front-end for nanopore sensing, specifically handling large input capacitances while maintaining high bandwidth (> 1 MHz) and low noise (< 20 pA_{rms} system target). Second, a low-jitter, wide-range All-Digital Phase-Locked Loop (ADPLL) is designed, leveraging state-of-the-art digital architectures, including a bang-bang phase-frequency detector and a digitally controlled oscillator, targeting multi-GHz operation with sub-picosecond jitter for SoC clocking applications.

Simulation results verify the effectiveness of the g_m/I_D methodology, showing close agreement between the design targets and the final performance of the OTA achieved using the full BSIM-IMG device models. The work demonstrates a practical and efficient approach for designing complex analog and mixed-signal circuits in advanced FD-SOI technology, facilitating the development of next-generation sensing and communication systems.

Table of Contents

Abstract	ii
Table of Contents	iii
List of Tables	vi
List of Figures	vii
1 Introduction	1
1.1 Motivation and Background	1
1.2 Design Challenges in Modern CMOS	2
1.3 Proposed Solution and Research Objectives	3
1.4 Thesis Organization	4
1.5 Core Contributions	4
1.6 Summary	5
2 A $\frac{g_m}{I_D}$-Based Design Methodology for 22nm FD-SOI	7
2.1 Motivation for a Systematic Approach	7
2.2 The Lookup Table and Inversion Coefficient Framework	9
2.3 Technology Characterization and Validation	10
2.3.1 Simulation Environment & Models	11
2.3.2 Device Selection	11
2.3.3 The 4D Characterization Testbench	11
2.3.4 Parameter Extraction	12
2.3.5 Sanity Checks of the Device Models	12
2.3.6 Core Design Charts	14

2.3.7	FD-SOI Specific Characterization	22
2.4	Summary	25
3	Design of a High-Performance Folded Cascode OTA for a Low-Noise Charge-Integrating Amplifier	26
3.1	Introduction and Design Specifications	26
3.1.1	System-Level Analysis of the Transimpedance Amplifier	26
3.1.2	System-Level Requirements for Nanopore Readout	30
3.1.3	OTA Design Specifications	31
3.1.4	Summary of Prior TIA Approaches	31
3.2	Systematic Design of the Folded Cascode OTA	32
3.2.1	Architecture and Analytical Model	32
3.2.2	Systematic Sizing Procedure[1]	33
3.2.3	OTA Simulation and Verification	36
3.3	Capacitive Transimpedance Amplifier (CTIA) Implementation and Analysis . .	37
3.3.1	TIA Principle of Operation	38
3.3.2	Noise Analysis of the TIA	39
3.3.3	TIA Simulation and Verification	39
3.4	Chapter Summary and Transition	41
4	Design of a Low-Jitter All-Digital Phase-Locked Loop (ADPLL)	43
4.1	Introduction and Design Goals	43
4.2	ADPLL System-Level Architecture	43
4.3	Circuit-Level Implementation and Analysis	45
4.3.1	Phase and Frequency Detector (PFD)	45
4.3.2	Digital Loop Filter (DLF)	50
4.3.3	Digitally Controlled Oscillator (DCO)	54
4.3.4	Clock Divider	58
4.3.5	DCO Simulation and Verification	60
4.4	Closed-Loop System Verification	61
4.5	Performance Summary	62

5	Conclusion and Future Directions	65
5.1	Conclusion	65
5.2	Statement on Contributions	66
5.3	Future Work	67
	Bibliography	68
	Appendices	74
Appendix A	$\frac{g_m}{I_D}$ extraction and mapping MATLAB code	74
Appendix B	Folded-cascode OTA Design and optimization MATLAB code	87

List of Tables

3.1	Target Capacitive TIA System Specifications	30
3.2	OTA Target Specifications	31
3.3	W and L ratios of the Folded Cascode OTA	32
3.4	OTA Performance Summary	38
3.5	Performance comparison with state-of-art nanopore arrays and amplifiers	42
4.1	ADPLL Target Specifications	44
4.2	Performance Summary and Comparison with State-of-the-art Ring PLLs	63

List of Figures

2.1	Framework for script-based sizing using precomputed lookup tables, adapted from[2]	10
2.2	Transconductance efficiency (g_m/I_D) and transit frequency (f_T) versus V_{GS} for the eguslvt NMOS and PMOS devices.	13
2.3	Transit frequency (f_T) versus g_m/I_D ratio for NMOS (a) and PMOS (b) devices with various channel lengths (L).	15
2.4	Intrinsic gain versus g_m/I_D ratio for NMOS and PMOS devices.	16
2.5	Current density versus g_m/I_D ratio for NMOS and PMOS devices.	17
2.6	Excess noise factor (γ) versus g_m/I_D ratio for NMOS and PMOS devices with minimum channel lengths.	19
2.7	Flicker noise crossover frequency (f_{co}) versus g_m/I_D ratio for NMOS and PMOS devices with various channel lengths (L).	20
2.8	Flicker noise crossover frequency (f_{co}) versus g_m/I_D ratio for NMOS and PMOS devices with $L = 6 \mu m$	21
2.9	Example of a total noise PSD for a representative device, showing the flicker noise ($1/f$) region, the thermal noise floor, and the crossover frequency (f_{co}). . .	22
2.10	Effect of back-gate bias (V_{SB}) on g_m/I_D and f_T versus V_{GS} characteristics for an NMOS device.	23
2.11	Threshold voltage (V_{th}) versus channel length (L) for different back-gate biases, illustrating V_{th} tuning and short-channel effects.	24
2.12	Threshold voltage (V_{th}) versus channel length (L) for different drain-source voltages (V_{DS}), showing minimal DIBL effect in both NMOS and PMOS devices. . .	24

3.1	Inverse feedback factor ($1/\beta$) versus frequency for different values of the compensation capacitor (C_F), shown alongside the OTA's open-loop gain (A_{OL}). . .	28
3.2	Closed-loop transimpedance gain of the TIA for different values of the feedback compensation capacitor (C_F).	29
3.3	Normalized time-domain step response of the TIA for different values of the compensation capacitor (C_F).	29
3.4	Schematic of the single-ended folded cascode OTA.	32
3.5	Contour plot of DC loop gain as a function of NMOS and PMOS cascode channel lengths.	34
3.6	(a) Input pair current (I_{D1}) and (b) self-loading ratio (r_{self}) versus $(g_m/I_D)_1$ for different channel lengths.	35
3.7	Convergence of (a) self-loading ratio (r_{self}) and (b) required input pair current (I_{D1}) over several design iterations.	35
3.8	Simulated open-loop gain and phase response of the folded cascode OTA. . . .	36
3.9	Simulated input-referred noise spectral density of the folded cascode OTA. . .	37
3.10	Schematic of the Transimpedance Amplifier with feedback capacitor C_F and reset switch S_R	38
3.11	Transient simulation of the TIA showing the output voltage ramp in response to a constant input current and the reset phase.	40
3.12	Output and Input noise power spectral densities.	40
3.13	Running integral of the power spectral density.	41
3.14	Physical Layout of the TIA in 22nm FD-SOI.	41
4.1	Top-level block diagram of the All-Digital PLL [3].	44
4.2	BB-PFD schematic.	45
4.3	Mutex schematic.	46
4.4	PFD testbench schematic.	49
4.5	BB-PFD transfer characteristics to show how it is working as P2D.	49
4.6	BB-PFD phase noise.	50
4.7	BB-PFD Layout.	50
4.8	Digital Loop Filter block diagram - PID.	51

4.9	Digital Loop Filter block's simulation results.	51
4.10	Synthesized Digital Loop Filter block	51
4.11	MASH Block Diagram.	53
4.12	Tri-state Inverter used in the DCO	56
4.13	Array of inverters showing the three-phase, interleaved column structure and 17 rows.	56
4.14	Block diagram of the DCO inverter array formation and its control inputs. . . .	57
4.15	DCO controller's simulation results	58
4.18	Synthesized Clock Divider	58
4.16	Synthesized DCO controller	59
4.17	Clock Divider simulations	59
4.19	DCO simulation showing the effect of different control configurations on the output oscillation frequency.	60
4.20	Simulated output phase noise spectrum at 5.2 GHz.	61
4.21	Loop Gain frequency simulation response	62
4.22	Phase transfer function = $\frac{\Phi_{out}(f)}{\Phi_{in}(f)}$	62
4.23	operating frequency ranges under different supply voltages	63
4.24	Layout of the designed ADPLL core in 22nm FD-SOI.	64

Chapter 1

Introduction

1.1 Motivation and Background

Analog and mixed-signal integrated circuits (ICs) remain indispensable components in modern electronic systems. The relentless scaling of CMOS technology brings performance benefits but also challenges for analog design, particularly reduced supply voltages and the inaccuracy of traditional models in moderate/weak inversion[4]. Fully-Depleted Silicon-On-Insulator (FD-SOI) technology, such as the 22nm process used in this work, offers advantages like back-gate tuning but requires systematic design methodologies.

A key application driving the need for high-performance, low-noise analog front-ends in our lab is nanopore-based DNA sequencing. This third-generation sequencing (TGS) technology identifies nucleotide sequences by electrically measuring the modulation of ionic current as a DNA strand translocates through a nanometer-sized pore[5]. Nanopore sequencing has revolutionized genomics by enabling long reads and, most importantly to our electronic engineering goals, portable, real-time analysis. An underlying goal of our lab is to build chips that enable the broad deployment of this technology in portable settings.

However, the readout circuitry of nanopore-based instruments must accurately measure very small currents (pA-nA range) with high bandwidth (>1 MHz) in the presence of significant sensor capacitance (1-10 pF)[6, 7]. Achieving high throughput density (sequencing speed per unit chip area) requires pushing the limits of noise, bandwidth, and power consumption in the front-end amplifier, often implemented as a Transimpedance Amplifier (TIA)[7].

Downstream processing of nanopore signals derived from TIAs presents another challenge

that is also being addressed by our lab. In particular, we are developing systems-on-chip (SoCs) capable of carrying out the complex bioinformatics needed to interpret nanopore measurements in mobile devices. Among the many engineering aspects of this challenge is efficient clock generation. Of course this concerns many complex SoCs in modern computing, including the embedded low-power Internet-of-Things (IoT) and AIoT domains which subsume our nanopore processing interests as well.

These heterogeneous SoCs often combine standard processor cores with specialized accelerators, as is the case in our work. In systems aiming to push performance and power-efficiency envelopes, using multiple clock domains operating at different frequencies is especially critical[8]. All-Digital Phase-Locked Loops (ADPLLs) have emerged as a compelling solution, offering advantages over analog PLLs in area, scalability, portability across process nodes, and ease of integration into digital design flows[8]. Designing ADPLLs in mature yet advanced nodes like 22nm FD-SOI strikes a good cost-performance balance for AIoT systems as ours, enabling high clock rates while leveraging the technology’s benefits like low-voltage operation and back-gate biasing[8]. Achieving wide tuning ranges, low jitter, and high power efficiency are key goals for ADPLLs in these applications[8].

1.2 Design Challenges in Modern CMOS

Designing high-performance analog circuits in advanced nodes like 22nm FD-SOI presents several significant hurdles:

Inaccuracy of Traditional Models: The classic “square-law” model, while intuitive, is inaccurate outside the strong inversion region[4]. This discrepancy between simple hand calculations and complex simulator models (like BSIM-IMG, the industry standard for FD-SOI[9]) makes intuitive design difficult.

Inefficiency of Traditional Design Flows: Relying solely on iterative SPICE simulations (“tweaking”) is time-consuming[2], yields non-optimal results, and produces designs that are difficult to reuse or port to other technologies. SPICE-in-the-loop optimization is often too computationally expensive for thorough design space exploration[2].

Circuit-Specific Complexity:

TIA Design: Achieving stability while driving the large input capacitance (C_{in}) typical of

nanopore sensors is a major challenge[10]. There is a fundamental trade-off between noise, bandwidth, and power consumption that must be carefully managed to maximize throughput density[6, 7, 10]. The high flicker noise in advanced nodes like 22nm further complicates low-noise design.

ADPLL Design: Key challenges include designing wide-range, low-noise Digitally Controlled Oscillators (DCOs), high-resolution Time-to-Digital Converters (TDCs), and stable Digital Loop Filters (DLFs). Achieving low jitter requires careful management of noise sources and quantization effects within the digital loop[3, 8].

Process Variation and Mismatch: As device dimensions shrink, the relative impact of random variations in doping, oxide thickness, and lithography increases, leading to greater mismatch between nominally identical devices. Accurately predicting and mitigating the effects of mismatch is crucial for yield, but often requires lengthy Monte Carlo simulations[11].

1.3 Proposed Solution and Research Objectives

To address these challenges, this thesis adopts and applies a systematic, script-based design methodology centered on the transconductance-to-current ratio (g_m/I_D). This approach utilizes pre-computed lookup tables (LUTs) generated from accurate SPICE models to characterize transistor performance across all operating regions[1, 2]. By using g_m/I_D as the primary design knob, the methodology provides an intuitive yet precise way to navigate complex design trade-offs and bridge the gap between hand analysis and simulation[4]. It enables rapid design space exploration and facilitates the creation of optimized, well-documented, and reusable circuit designs.

The primary research objectives of this thesis are:

1. To develop and validate a comprehensive g_m/I_D -based design framework for the GlobalFoundries 22nm FD-SOI technology.
2. To apply this methodology to the systematic design of a high-performance folded cascode OTA optimized for a nanopore CTIA, targeting state-of-the-art throughput density by achieving a specific balance of noise ($< 20 \text{ pA}_{rms}$), bandwidth ($> 1 \text{ MHz}$), and power ($< 500 \text{ } \mu\text{W}$ [7, 12]).
3. To design and implement a low-jitter, wide-range ADPLL in 22nm FD-SOI, leveraging

digital techniques and targeting performance metrics suitable for SoC clocking (e.g., > 4 GHz range, $< 500 f_s$ jitter, < 2 mW power)[8].

1.4 Thesis Organization

The remainder of this thesis is organized as follows:

Chapter 2 details the development of the g_m/I_D -based design methodology, presents the technology characterization results for the 22nm FD-SOI process, and validates the framework.

Chapter 3 describes the systematic design of the folded cascode OTA using the proposed methodology and verifies its performance through simulation. It then demonstrates the integration of this OTA into a CTIA circuit for nanopore readout.

Chapter 4 presents the architecture, circuit-level implementation, and simulation results for the All-Digital PLL.

Chapter 5 concludes the thesis, summarizing the key contributions and suggesting directions for future work.

1.5 Core Contributions

This thesis presents the design and implementation of a monolithic System-on-Chip (SoC) front-end for Nanopore-based DNA sequencing in 22nm FD-SOI technology. While traditional analog design often relies on iterative geometry tuning to match small-signal parameters, this work establishes a systematic “Inverse Design” framework to streamline the sizing process in advanced nodes. The core contributions of this research are:

1. A Systematic “Inverse Design” Methodology via g_m/I_D

This work introduces a design flow that fundamentally reverses the traditional analog sizing approach. Instead of guessing transistor geometry (W, L) to achieve target small-signal parameters, this methodology starts with high-level system specifications (e.g., Bandwidth, Gain) and maps them to normalized design parameters such as Transconductance Efficiency (g_m/I_D), Transit Frequency (f_T), and Mismatch coefficients. To support this, a comprehensive 4D Lookup Table (LUT) database was generated based on the BSIM-IMG

model, the industry-standard compact model for FD-SOI devices developed by UC Berkeley and utilized within Cadence Virtuoso. The characterization specifically targets Super Low Threshold Voltage (slvt) and Extended Gate (eguslvt) PMOS and NMOS devices from the GlobalFoundries 22FDX library. By prioritizing these low-threshold devices, the framework maximizes voltage headroom and enables the hierarchical synthesis of sub-modules (such as Gain Stages and Input Pairs) directly from system specs, ensuring the design is optimal by construction without reliance on iterative SPICE tweaking.

2. High-Bandwidth TIA Design for Large Parasitic Sensors

A critical bottleneck in portable DNA sequencing is the massive parasitic capacitance ($C_p \approx 5$ pF) of the solid-state nanopore, which typically limits bandwidth. Using the proposed inverse design flow, this thesis demonstrates a Transimpedance Amplifier (TIA) capable of achieving a 1 MHz bandwidth and 5 M Ω transimpedance gain while driving this large capacitive load. The design leverages the forward body-biasing (FBB) capability of the 22nm node to maximize voltage headroom and linearity in a low-voltage (0.8 V) environment.

3. All-Digital PLL (ADPLL) for System Synchronization

To enable a fully integrated digital sequencing system, this work implements a wide-range All-Digital Phase-Locked Loop (ADPLL). The ADPLL is designed to support the diverse clocking requirements of the digital signal processing core and high-speed I/O. By utilizing a MASH 1-1-1 Sigma-Delta Modulator and a high-resolution DCO, the system achieves a simulated phase noise of -84 dBc/Hz at a 1 MHz offset, providing the low-jitter timing infrastructure required for high-precision ADC sampling.

1.6 Summary

This chapter introduced the challenges facing analog and mixed-signal circuit design in advanced CMOS technologies like 22nm FD-SOI, particularly the limitations of traditional design models and flows under low supply voltages. It motivated the adoption of a systematic, g_m/I_D -based design methodology using pre-computed lookup tables as a robust solution. The chapter also outlined the specific application contexts for this work: the design of a low-noise readout circuit for nanopore sequencing and a low-jitter clock generation system using an All-Digital

PLL. Finally, it established the limitations of current design approaches and the opportunities presented by the proposed methodology to address the demanding performance requirements of these applications.

Chapter 2

A $\frac{g_m}{I_D}$ -Based Design Methodology for 22nm FD-SOI

2.1 Motivation for a Systematic Approach

The continuous scaling of CMOS technology has brought significant performance improvements, but has also introduced profound challenges for analog circuit design. As supply voltages in advanced nodes such as 22nm FD-SOI are reduced to 0.8V and below, transistors are increasingly forced to operate outside of the traditional strong inversion region[4]. This trend means that the moderate and weak inversion regions are central to low-power design and that the strong inversion region is becoming less accessible. Consequently, the conventional “square-law” transistor model is no longer accurate, which invalidates many hand-calculation techniques and creates a significant gap between hand analysis and simulation results. This discrepancy often forces designers into a time-consuming, iterative loop of SPICE simulations and manual tweaking[1, 2], a process that is inefficient and yields designs that are difficult to document, reuse, or port to new technologies.

The choice of 22nm FD-SOI technology for this work introduces both unique opportunities and specific challenges. FD-SOI is an excellent platform for low-power and high-frequency circuits, primarily due to its ability to dynamically modulate the threshold voltage (V_{th}) by applying a bias to the back gate[9]. This feature allows for an effective trade-off between power and performance, a critical capability for modern energy-efficient circuits. However, designing for this technology also requires a methodology that can account for the complex electrostatic

coupling between the front and back gates. Furthermore, inherent process variations and device mismatch are significant concerns in advanced nodes[11], and traditional validation using lengthy Monte Carlo simulations becomes a major bottleneck in the design cycle.

To address these issues, this thesis adopts a script-based systematic design methodology centered on the transconductance-to-current ratio ($\frac{g_m}{I_D}$) and precomputed lookup tables (LUTs)[2]. This script-based approach effectively bridges the gap between high-level design equations and complex device physics by replacing simplified analytical models with accurate, simulation-based data. By using the Inversion Coefficient (IC) or the $\frac{g_m}{I_D}$ ratio as the primary design parameter, this framework provides a unified and continuous view of transistor performance across all regions of operation[4], from weak to strong inversion. This framework enables a fast, accurate, and repeatable design flow, which is well suited for advanced technologies like 22 nm FD-SOI.

Although the back-gate bias significantly alters the threshold voltage, research has shown that the normalized transconductance efficiency ($\frac{g_{ms}}{IC}$) curve remains remarkably consistent[13]. This powerful finding implies that a single, unified set of design charts can be used to guide optimization across a wide range of back-gate biasing conditions, making the design methodology both robust and highly efficient for FD-SOI technologies. This chapter therefore presents the development and validation of this systematic methodology, demonstrating its unique suitability for exploiting the full potential of 22 nm FD-SOI technology while providing a rapid and accurate framework for designing high-performance analog circuits.

Alternative g_m/I_D workflows based on analytical compact models such as EKV have been widely explored in the context of bulk CMOS technologies. In contrast, this thesis applies the same g_m/I_D and inversion-coefficient concepts directly to operating-point data obtained from the BSIM-IMG compact model, which is the foundry-supported FD-SOI model in the GlobalFoundries 22FDX PDK. This choice preserves the intuitive, physics-based g_m/I_D design space while remaining fully consistent with the industrial device models used in Cadence, avoiding the need for separate EKV-style model implementations.

2.2 The Lookup Table and Inversion Coefficient Framework

The proposed design methodology is centered on a script-based framework that utilizes precomputed lookup tables (LUTs). This approach is designed to bridge the gap between high-level analytical design equations and the complex behavior of advanced transistor models by replacing simplified device models with accurate, simulation-based data[2]. The LUTs are generated once for a given technology by scripting a comprehensive set of simulations to extract key device parameters and performance metrics across a range of bias conditions and geometries.

Central to this framework is the use of the Inversion Coefficient (IC) as the primary design parameter, which replaces the traditional gate-overdrive voltage ($V_{OV} = V_{gs} - V_{th}$). The IC is defined as the drain current in saturation normalized to a so-called technology-specific current ($IC = \frac{I_{Dsat}}{I_{spec}}$) and serves as a continuous measure of the transistor's inversion level[4]. The specific current (I_{spec}) is a fundamental parameter in this methodology. It is not just an arbitrary normalizing factor; it is a technology-dependent current defined as:

$$I_{spec} = I_{spec\Box} \cdot \frac{W}{L} = (2n\mu_0 C_{ox} U_T^2) \cdot \frac{W}{L}$$

where n is the subthreshold slope factor, μ_0 is the low-field mobility, C_{ox} is the gate oxide capacitance per unit area, and U_T is the thermal voltage. This equation shows that I_{spec} combines the key process parameters (n , μ_0 , C_{ox}) and temperature (U_T) into a single value. $I_{spec\Box}$, the specific current per square, is the fundamental technology parameter for the designer.

The value of I_{spec} serves as a physical boundary. As shown in [4, 13], the inversion coefficient $IC = 1$ (where $I_D = I_{spec}$) marks the approximate transition point between the weak inversion and strong inversion regions. The IC therefore provides a technology-independent measure of the inversion level, allowing the operating point to be defined consistently across weak ($IC < 0.1$), moderate ($0.1 < IC \leq 10$), and strong ($IC > 10$) inversion regions[4].

Directly related to the IC is the g_m/I_D ratio, or transconductance efficiency. In weak inversion, a transistor operates most efficiently, and the g_m/I_D ratio approaches its theoretical maximum of approximately $1/(nU_T)$. As the device moves into strong inversion, its efficiency

decreases. The complete workflow of this script-based sizing approach is illustrated in fig. 2.1. The use of LUTs ensures that the device parameters used in the script’s calculations accurately reflect the behavior captured by the full BSIM-IMG compact models, bypassing the complexities of the full model equations. As shown in the framework, the designer provides the desired

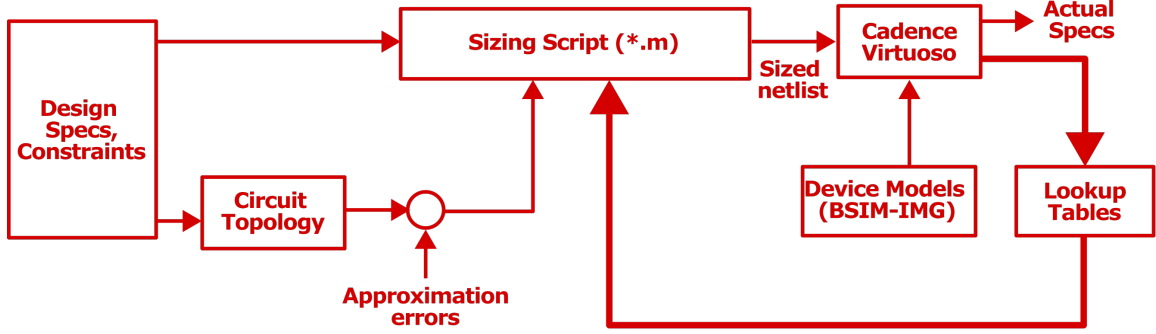


Figure 2.1: Framework for script-based sizing using precomputed lookup tables, adapted from[2]

specifications and constraints to a sizing script. This script leverages a set of circuit design equations along with pre-computed lookup tables (LUTs) to determine the appropriate device sizes, generating a sized netlist. The power of this approach lies in its use of normalized design ratios, such as transconductance efficiency ($\frac{g_m}{I_D}$) and transit frequency (f_T). These ratios allow for design trade-offs to be evaluated independently of specific bias currents or device dimensions. The LUTs themselves are organized as a multi-dimensional data structure where, for each simulated operating point, both raw parameters and these key design ratios are stored. The use of LUTs ensures that the device parameters used in the script’s calculations accurately reflect the behavior captured by the full BSIM-IMG compact models, enabling a fast and accurate design flow[9].

2.3 Technology Characterization and Validation

The purpose of this section is to present the characterization results for the target 22nm FD-SOI process. This involves validating the underlying device models and presenting the extracted data in a set of design charts that will be used to guide the circuit design process in the subsequent chapters.

2.3.1 Simulation Environment & Models

The characterization was performed using the Cadence Spectre circuit simulator within the Virtuoso Analog Design Environment (ADE). The transistor behavior was simulated using the BSIM-IMG (Independent Multi-Gate) compact model, which is the industry-standard model for FD-SOI technologies. Unlike bulk CMOS models (BSIM4), BSIM-IMG explicitly captures the coupling between the front and back gates, allowing for accurate modeling of the threshold voltage modulation via Back-Gate Biasing (V_{BG}), which is central to this thesis.

The entire characterization flow is fully scripted. A MATLAB driver program invokes Spectre in batch mode, configures the device testbench, and sweeps the four-dimensional bias and geometry space without any manual interaction. For each operating point, Spectre reports the relevant BSIM-IMG operating-point quantities, ac, and noise simulation results, which are then post-processed in MATLAB to construct the normalized gm/ID-based design charts used throughout this work.

2.3.2 Device Selection

Two specific device flavors from the GlobalFoundries 22FDX Process Design Kit (PDK) were targeted for characterization:

1. **Super Low Threshold Voltage (slvt):** Selected for high-speed signal paths (e.g., TIA Input Pair, DCO switches) where maximizing Overdrive Voltage (V_{OV}) is critical given the 0.8 V supply.
2. **Extended Gate Under-drive (eguslvt):** Selected for the I/O interfaces and bias circuitry requiring higher output impedance. Both NMOS and PMOS variants were characterized to enable complementary design structures.

2.3.3 The 4D Characterization Testbench

A schematic testbench was constructed consisting of a single Device Under Test (DUT) driven by ideal DC voltage sources at the Gate (V_G), Drain (V_D), and Back-Gate (V_{BG}), with the Source (V_S) grounded. A nested DC sweep was executed across four dimensions to cover the complete design space:

- **Gate Voltage (V_{GS}):** Swept from 0 V to 0.8 V (Step: 5 mV) to capture sub-threshold, moderate, and strong inversion regions.
- **Drain Voltage (V_{DS}):** Swept from 0 V to 0.8 V (Step: 20 mV) to capture the triode-to-saturation transition.
- **Channel Length (L):** Swept from L_{min} (20 nm for slvt, 70 nm for eguslvt) to $10\mu m$ to analyze Short Channel Effects and intrinsic gain variations.
- **Back-Gate Voltage (V_{BG}):** Swept from -2.0 V to +2.0 V (Step: 0.2 V) to characterize the Forward and Reverse Body Bias (FBB/RBB) sensitivities.

2.3.4 Parameter Extraction

For every point in this 4D space, key small-signal parameters—including Transconductance (g_m), Output Conductance (g_{ds}), and parasitic capacitances (C_{gg}, C_{dd}, C_{dg}), were extracted directly from the BSIM-IMG operating point parameters. These raw data points were then exported and structured into HDF5/Mat format for rapid querying during the synthesis phase.

The complete extraction and LUT construction flow is provided in Appendix A as MATLAB code. This code can be re-run to regenerate all characterization tables for the same Global-Foundries 22FDX PDK, or adapted with minimal changes to other FD-SOI technologies that provide BSIM-IMG models, ensuring that the methodology is reproducible beyond this specific design.

2.3.5 Sanity Checks of the Device Models

Before building the design charts, it is essential to first verify that the foundry-provided SPICE models exhibit physically correct behavior. This step ensures that the lookup tables are built upon a reliable foundation by checking the model’s behavior against well-understood device physics[2].

Figure 2.2 presents this validation for the eguslvt¹ transistors. The transconductance efficiency, or g_m/I_D ratio, is a central parameter in this methodology. For low-power design, a primary objective is to achieve a large transconductance (g_m) while consuming the minimum possible bias current (I_D), making a high g_m/I_D ratio desirable. In weak inversion (low V_{GS}),

¹A 22nm device type intended for analog/RF applications from 1.2-V nominal supplies.

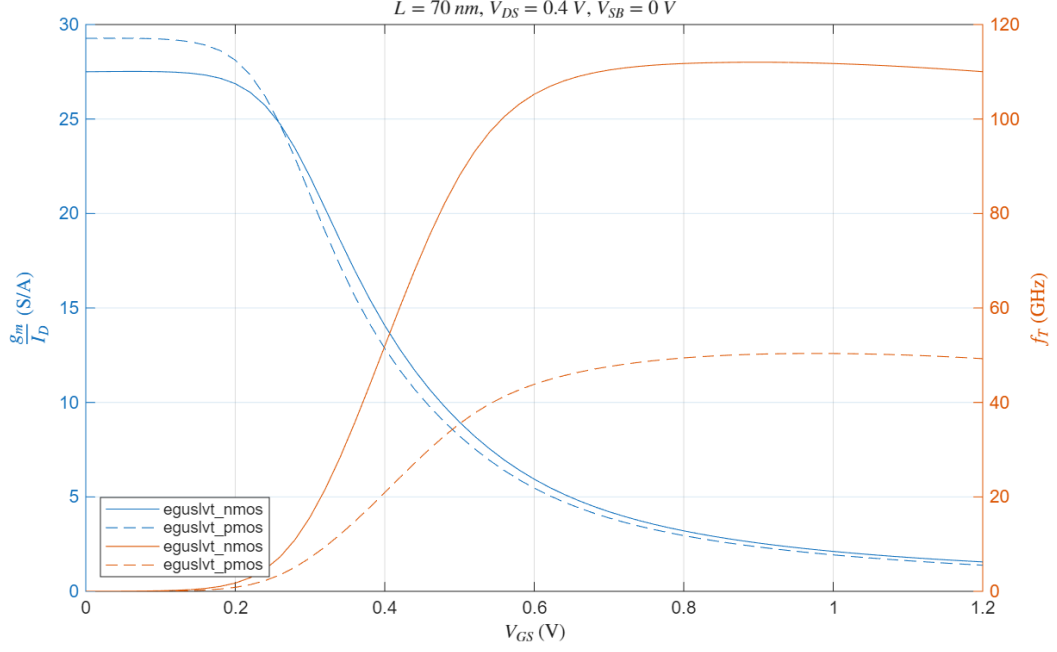


Figure 2.2: Transconductance efficiency (g_m/I_D) and transit frequency (f_T) versus V_{GS} for the eguslvt NMOS and PMOS devices.

the g_m/I_D ratio approaches its theoretical maximum of $\frac{1}{nU_T}$, where n is the subthreshold slope factor and U_T is the thermal voltage. For the classical square-law model in strong inversion, the ratio is given by $\frac{g_m}{I_D} = \frac{2}{V_{OV}}$, where V_{OV} is the overdrive voltage as noted earlier.

The plot reveals the fundamental trade-off between power efficiency and device speed, which is characterized by the transit frequency (f_T). The peak of the both NMOS and PMOS g_m/I_D curves, representing maximum efficiency, occurs at a low gate-source voltage ($V_{GS} < 0.4V$). In contrast, the highest speed, represented by the peak in f_T , occurs at a much higher bias of approximately $V_{GS} = 1.2V$. An ideal operating point with both maximum efficiency and maximum speed does not exist. It is also noteworthy that the nominal threshold voltage of 0.3 V does not represent a special point in this continuous design space.

The peak transit frequency (f_T) of the eguslvt NMOS and PMOS devices are approximately 110 GHz and 50 GHz, respectively, which is sufficient for building radio-frequency circuits. The plots also show that f_T saturates and begins to decrease at around $V_{GS} = 1V$. This roll-off is attributed to second-order effects such as carrier velocity saturation, which limits the increase of g_m at high electric fields. Since the models for both NMOS and PMOS devices exhibit these well-understood physical trends, they are considered reliable for generating the design lookup tables.

2.3.6 Core Design Charts

The extracted characterization data is organized into a set of design charts that use the g_m/I_D ratio as the primary design knob on the x-axis. These charts are essential design tools for visualizing key figures of merit (FOMs) and making quantitative trade-offs.

Figure 2.3 plots the transit frequency (f_T) against the g_m/I_D ratio for several different channel lengths (L) for both NMOS and PMOS devices. This chart directly illustrates the fundamental trade-off between device speed and power efficiency. The plots clearly show that the maximum device speed is achieved at a low g_m/I_D ratio (in the moderate-to-strong inversion region) and with the shortest available channel length. Conversely, operating at a high g_m/I_D for maximum power efficiency results in a lower intrinsic speed.

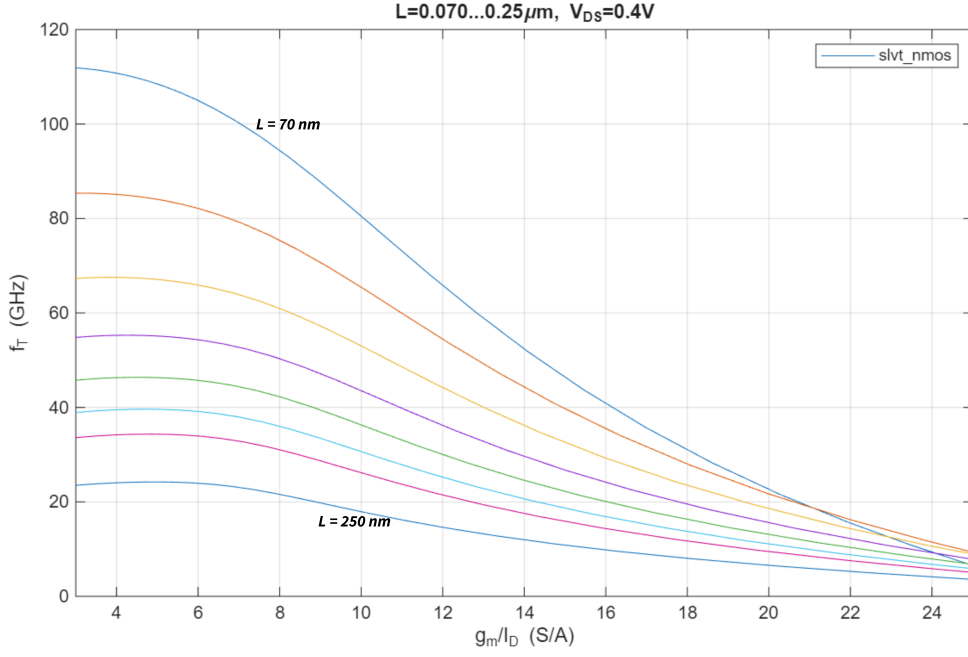
This characterization is performed with the MOSFET held in the saturation region by maintaining a drain-source voltage of $V_{DS} = V_{DD}/2$. Furthermore, the source-bulk voltage is kept at $V_{SB} = 0$ V, meaning the bulk and source terminals are connected. When $V_{SB} \neq 0$ V, the body effect occurs.

While often considered an unwanted parasitic effect in bulk CMOS, this effect is a key feature in FD-SOI technology, where the bulk terminal is more accurately described as a back gate. Applying a voltage to this back gate can be used intentionally as an additional design knob to modulate device characteristics, most notably the threshold voltage. The impact of back-gate biasing on the normalized design space will be explored in the following section.

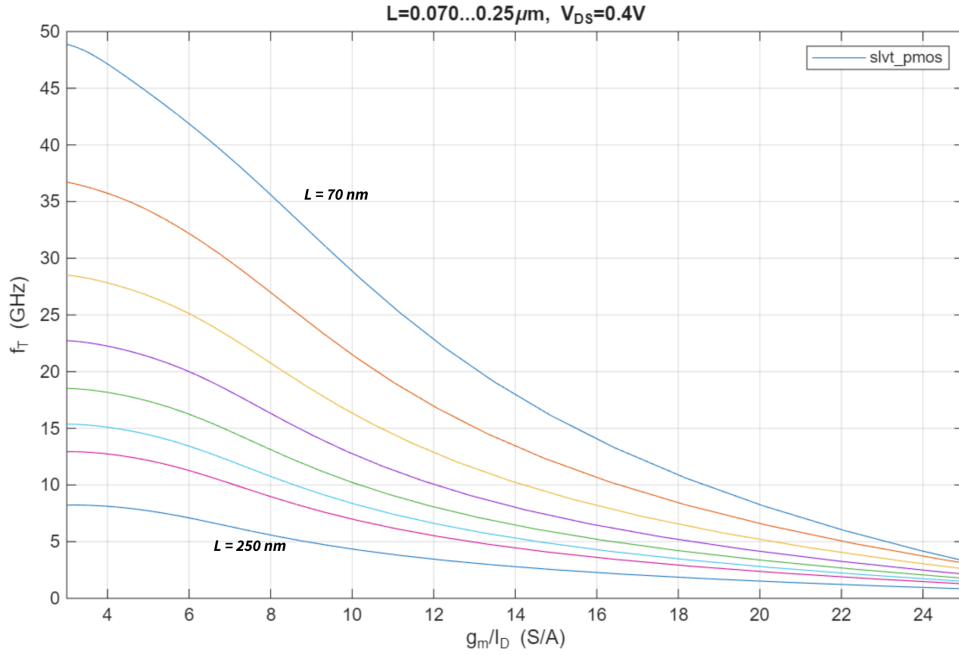
Another critical figure of merit for amplifier design is the intrinsic gain (g_m/g_{ds}), also known as self-gain. This ratio represents the maximum theoretical voltage gain achievable from a single transistor configuration. Figure 2.4 plots the intrinsic gain as a function of the g_m/I_D ratio for various channel lengths.

The chart demonstrates that intrinsic gain is highest at large g_m/I_D values (weak inversion) and increases significantly with longer channel lengths (L). This reveals a key design trade-off, as increasing L to achieve higher gain results in a slower transistor (lower f_T). This plot is the primary tool for selecting an appropriate channel length and operating point to meet a specific gain requirement.

During the characterization process, a modeling anomaly was observed in the NMOS data. For long-channel NMOS devices, the extracted intrinsic gain reached huge, non-physical values



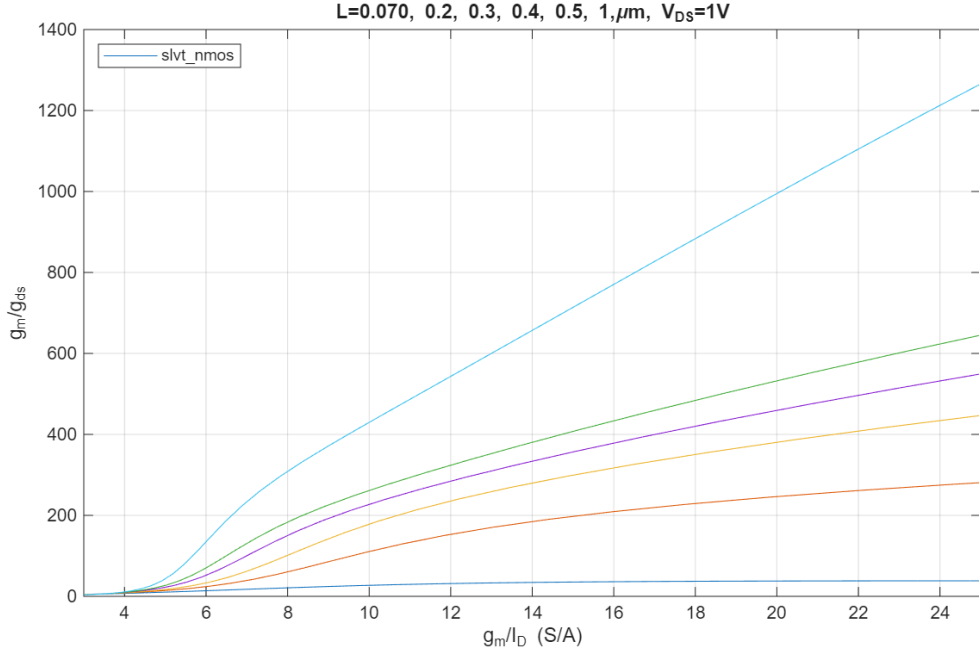
(a)



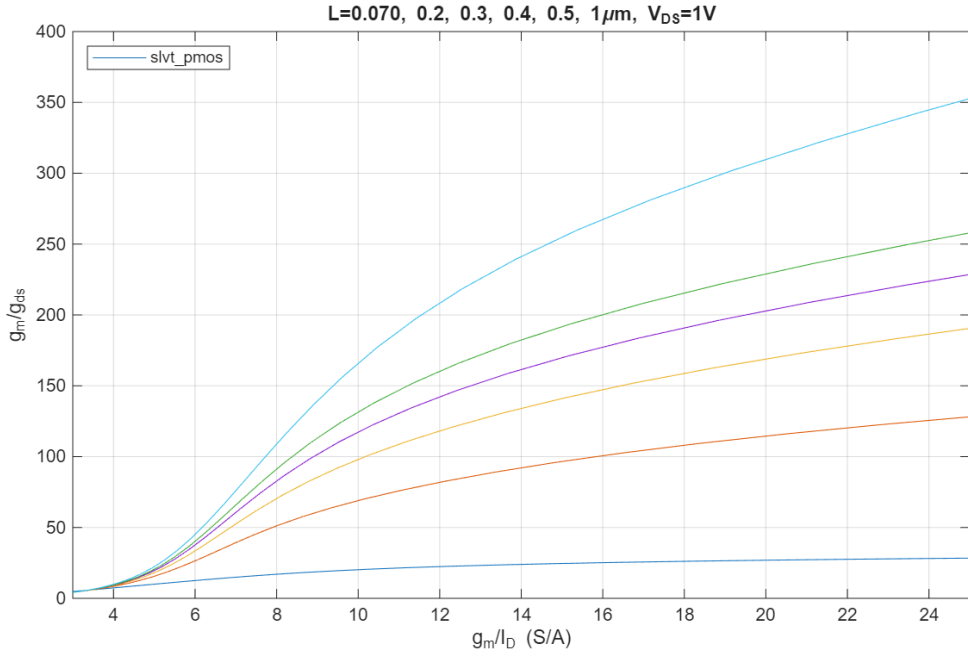
(b)

Figure 2.3: Transit frequency (f_T) versus g_m/I_D ratio for NMOS (a) and PMOS (b) devices with various channel lengths (L).

(e.g., in excess of 1000 V/V) that continued to increase without the expected saturation behavior seen in the PMOS data. This indicates a potential issue in the PDK model's parameterization of the output conductance (g_{ds}), which was likely returning unphysically small values. This example underscores the importance of critically benchmarking device models when adopting a new technology, as model parameterization is a complex task and artifacts can exist. Given



(a)



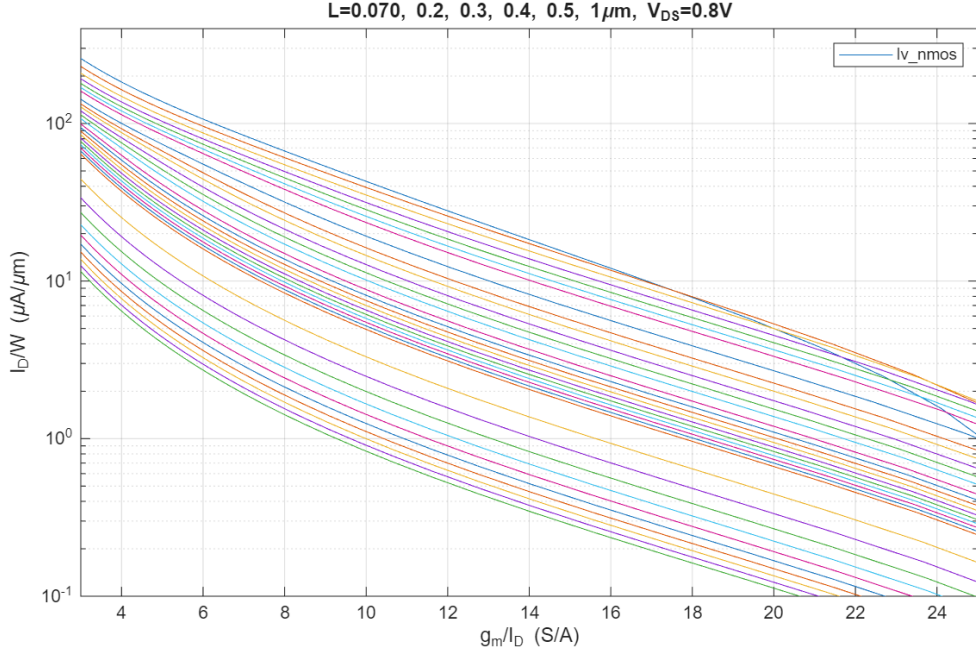
(b)

Figure 2.4: Intrinsic gain versus g_m/I_D ratio for NMOS and PMOS devices.

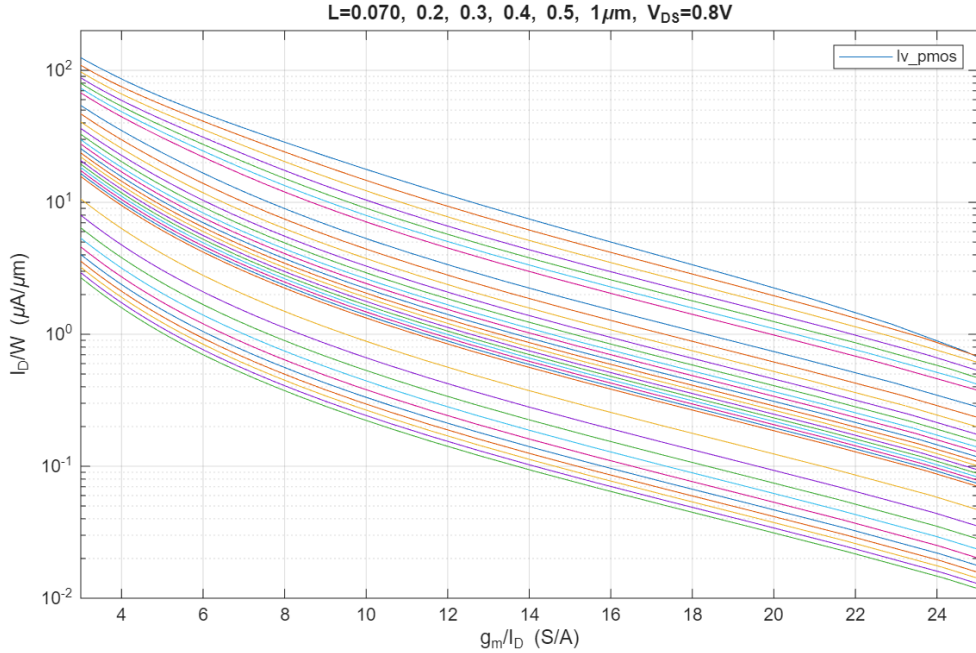
this discrepancy, a conservative approach is adopted for this work: in design regions where the NMOS model provides unreliable data, the corresponding, well-behaved PMOS intrinsic gain values are used as a proxy.

The drain current density (I_D/W) chart, shown in fig. 2.5, is the primary tool used for sizing the width (W) of a transistor. In a MOSFET, the total drain current (I_D) is directly

proportional to the device width (W). By normalizing the current by its width, we get the current density (I_D/W), a metric that is independent of W and is fundamentally set by the chosen operating point (g_m/I_D) and channel length (L).



(a)



(b)

Figure 2.5: Current density versus g_m/I_D ratio for NMOS and PMOS devices.

This chart is therefore used to find the required W for a target bias current (I_D). Once the optimal g_m/I_D and L are selected based on gain and speed requirements, the corresponding

I_D/W value is read from the chart. The final device width is then calculated as:

$$W = \frac{I_D}{I_D/W}$$

This systematic step completes the link between the high-level performance specifications and the physical dimensions of the transistor.

Noise performance is a critical consideration in analog circuit design. While several noise sources exist in a MOSFET, the primary source of thermal noise is the random motion of charge carriers in the channel. This is modeled as a current noise source between the drain and source terminals. For this first-order analysis, other sources such as induced gate noise are not considered, and flicker ($1/f$) noise will be addressed separately.

The power spectral density (PSD) of the drain current thermal noise is given by the well-established formula:

$$S_{id}(f) = \frac{\bar{i}_d^2}{\Delta f} = 4kT\gamma g_m$$

Here, k is the Boltzmann constant, T is the absolute temperature, and g_m is the transconductance of the device. The parameter γ is the excess noise factor. While classical long-channel theory suggests a constant value for γ ($2/3$ in saturation), this is not accurate for modern short-channel devices, where γ can be significantly larger and is a function of the device's operating point and geometry.

To enable accurate noise calculations within the g_m/I_D framework, the excess noise factor γ has been characterized across the design space. Figure 2.6 presents this characterization for both NMOS and PMOS devices. The chart in fig. 2.6 plots the extracted γ value as a function of the g_m/I_D ratio for minimum channel lengths (70 nm). This plot serves as a direct design tool for noise analysis. When calculating the thermal noise contribution of a transistor, a designer can first select an operating point (g_m/I_D) and channel length (L) based on gain and speed requirements. Then, the corresponding γ value can be found from this chart and used in the noise equation to achieve an accurate, simulation-based prediction of the device's drain current noise.

In addition to the frequency-independent thermal noise, MOSFETs exhibit a significant low-frequency noise component known as flicker noise, or $1/f$ noise. Its physical origin is largely

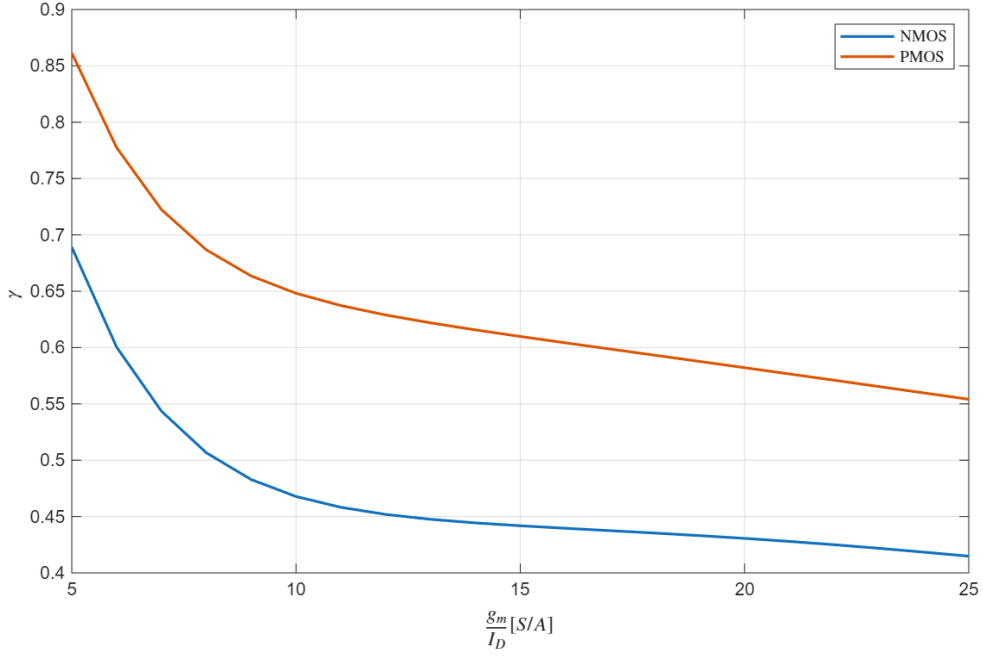
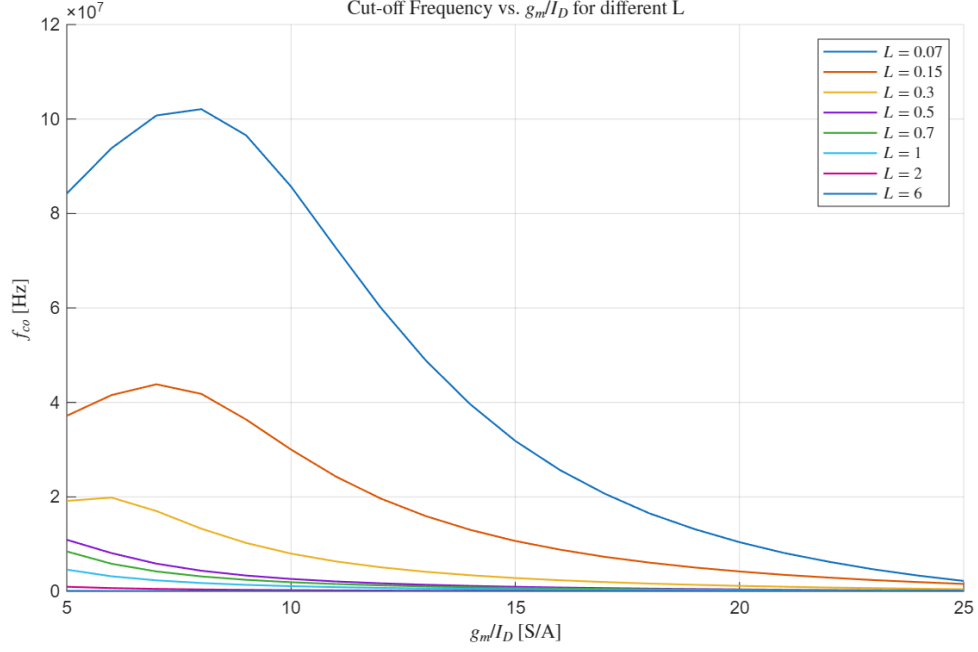


Figure 2.6: Excess noise factor (γ) versus g_m/I_D ratio for NMOS and PMOS devices with minimum channel lengths.

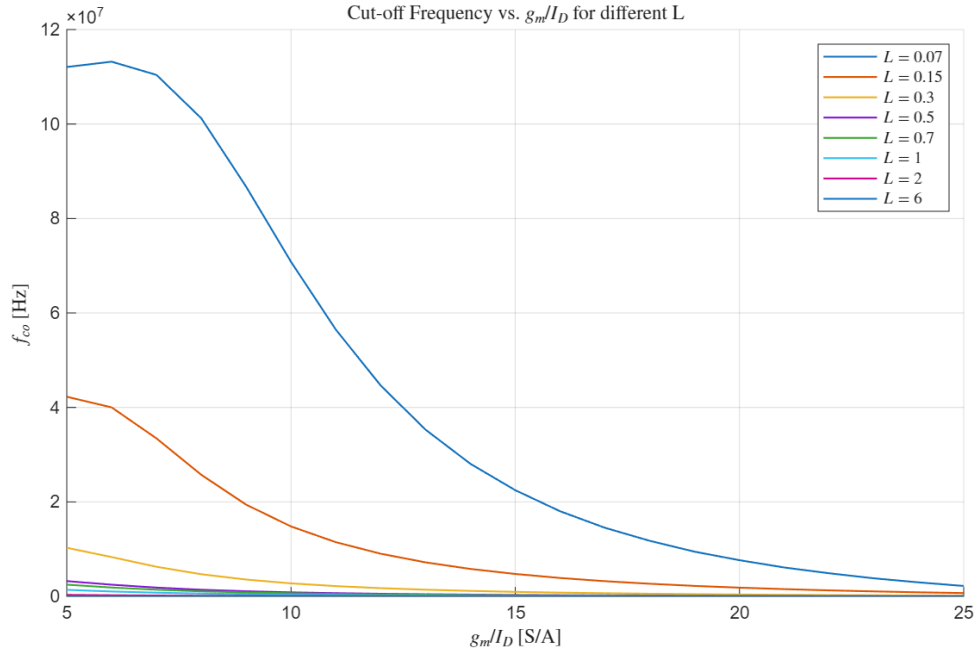
attributed to the trapping and de-trapping of charge carriers by defects at the silicon (Si) and silicon dioxide (SiO_2) interface. This behaviour is highly dependent on the manufacturing process, meaning its magnitude can differ significantly between technologies. The power spectral density (PSD) of flicker noise is characterized by its inverse relationship with frequency (f):

$$S_{fl}(f) = \frac{K_f}{f}$$

where K_f is a process-dependent flicker noise coefficient. To characterize the frequency range over which this noise is dominant, the crossover frequency (f_{co}) is used. This is the frequency at which the flicker noise power equals the thermal noise power. Figure 2.7 plots this important figure of merit against the g_m/I_D ratio for various device geometries. As shown in the plot, the crossover frequency is a strong function of both the operating point (g_m/I_D) and the channel length (L). For short-channel devices, f_{co} can reach well into the couple of megahertz, significantly degrading the noise performance of a circuit in that frequency band. The magnitude of flicker noise is generally inversely proportional to the device area (WL). For a fixed aspect ratio and g_m/I_D , this implies that flicker noise is inversely proportional to L^2 . This relationship creates a critical trade-off between flicker noise and speed. Increasing the channel length is an effective way to reduce flicker noise, but as shown previously, it dramatically reduces the transit



(a)

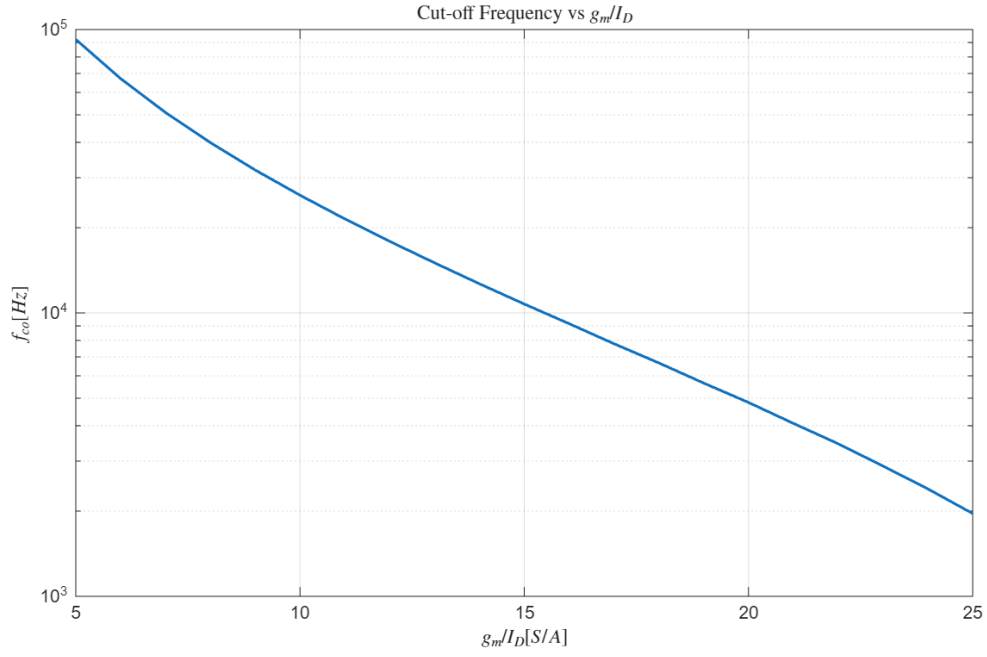


(b)

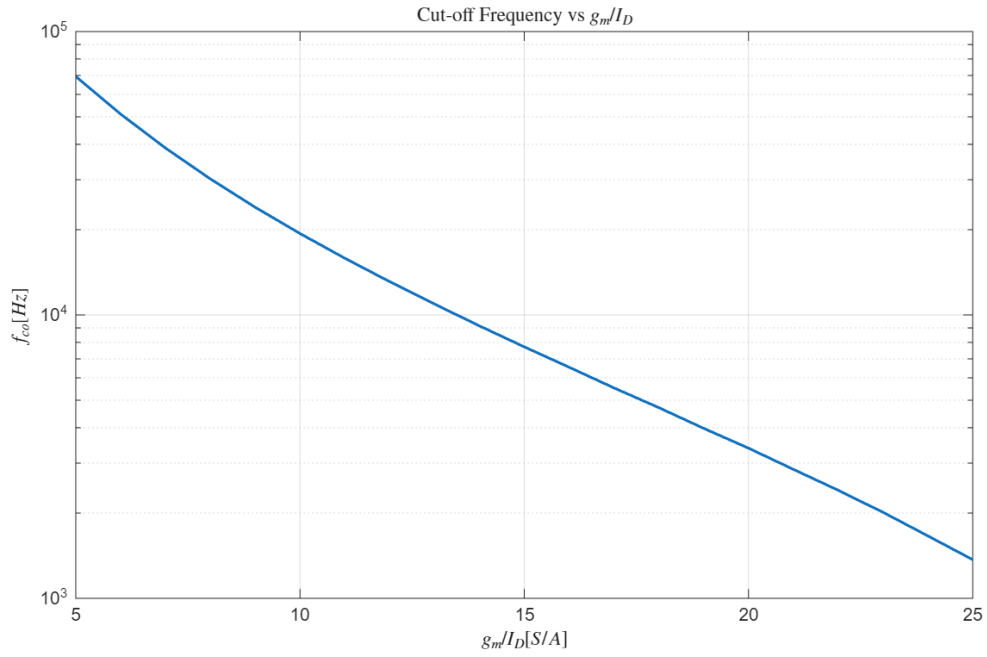
Figure 2.7: Flicker noise crossover frequency (f_{co}) versus g_m/I_D ratio for NMOS and PMOS devices with various channel lengths (L).

frequency (f_T).

The total noise spectrum of the device is the sum of the thermal and flicker noise components. This is illustrated in fig. 2.9, which shows an example of a total noise PSD. At low frequencies, the total noise is dominated by the $1/f$ characteristic of flicker noise. As frequency increases, the flicker noise diminishes, and the noise floor is set by the white thermal noise of the channel.



(a)



(b)

Figure 2.8: Flicker noise crossover frequency (f_{co}) versus g_m/I_D ratio for NMOS and PMOS devices with $L = 6 \mu m$

The crossover frequency, f_{co} , is clearly visible as the "corner" where the two noise sources have equal magnitude.

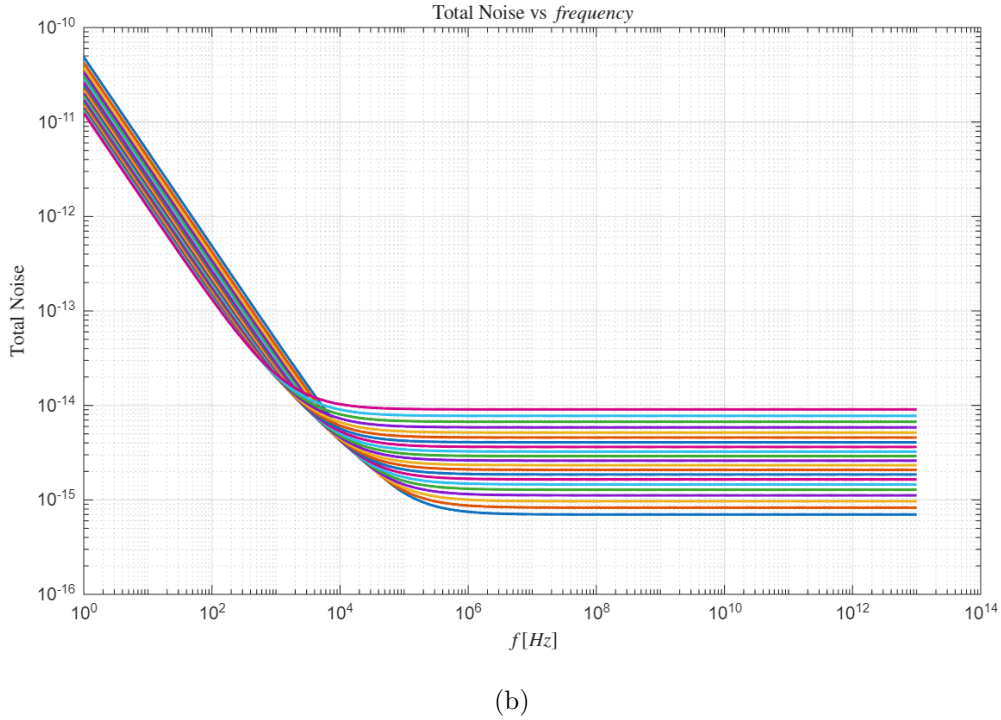
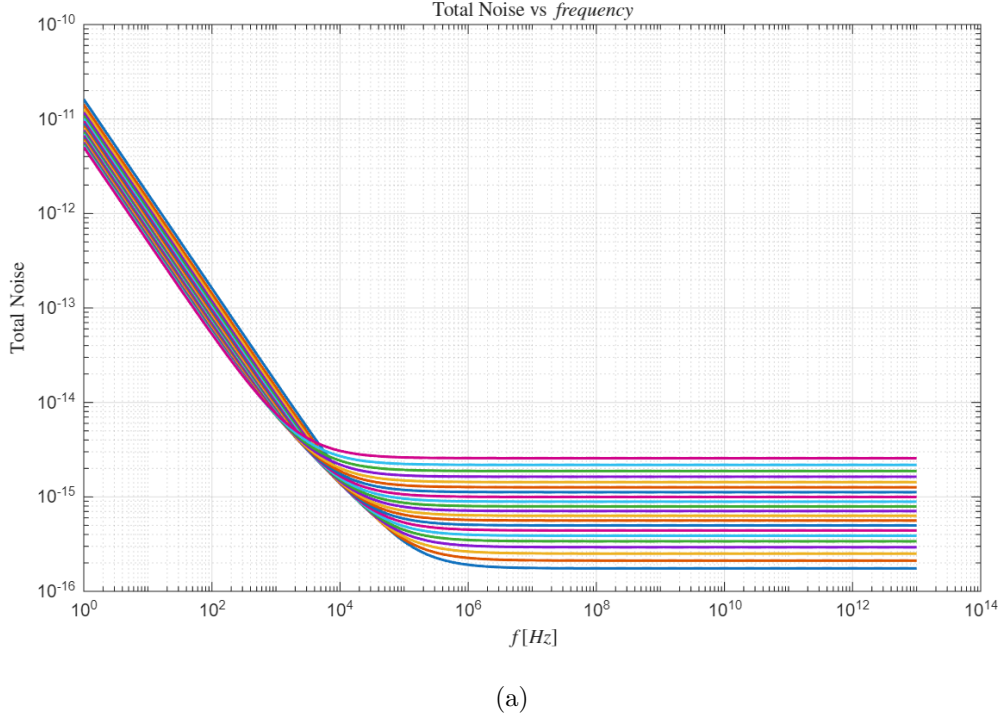


Figure 2.9: Example of a total noise PSD for a representative device, showing the flicker noise ($1/f$) region, the thermal noise floor, and the crossover frequency (f_{co}).

2.3.7 FD-SOI Specific Characterization

As introduced previously, the back gate in FD-SOI technology provides a powerful mechanism for tuning transistor performance. The primary effect of applying a back-gate bias ($V_{SB} = V_{bg}$) is the modulation of the front-gate threshold voltage (V_{th}). This results in a lateral shift of the

transistor's transfer characteristics, as shown in fig. 2.10 for an NMOS.

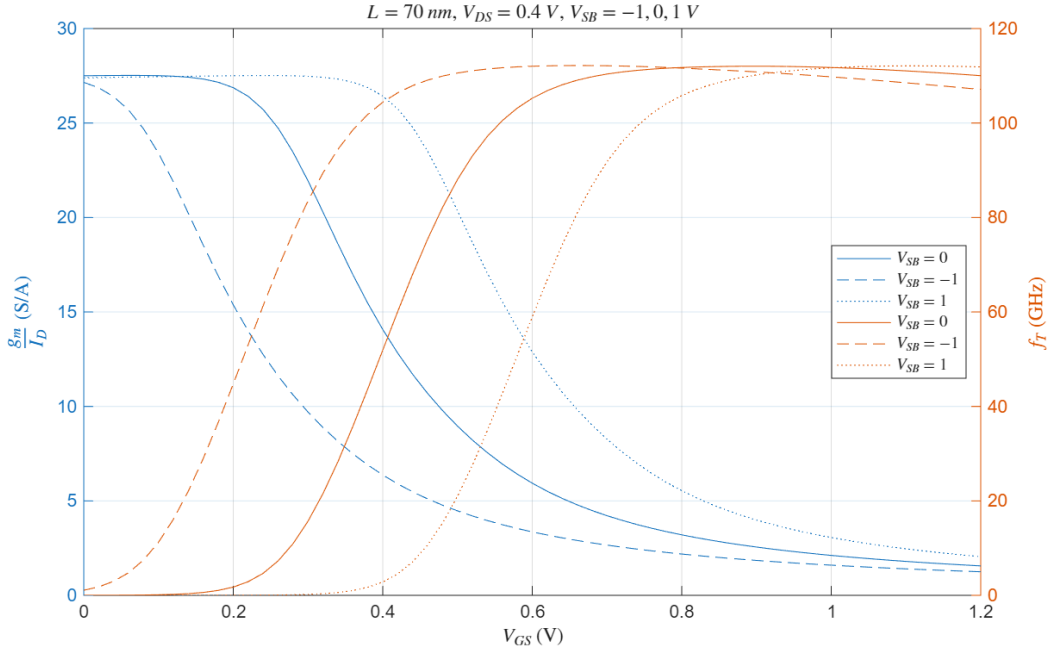


Figure 2.10: Effect of back-gate bias (V_{SB}) on g_m/I_D and f_T versus V_{GS} characteristics for an NMOS device.

The plot demonstrates that applying a forward body bias (FBB) shifts the curves to the left (corresponding to a lower V_{th}), while a reverse body bias (RBB) shifts them to the right (higher V_{th})[13]. Importantly, the shape and peak values of the g_m/I_D and f_T curves remain largely unchanged. This reinforces the conclusion from the previous section: the normalized design space is robust, and the back gate acts as an independent tuning knob.

This threshold voltage modulation can be quantified by plotting the extracted V_{th} as a function of channel length (L) for different back-gate biases, as presented in fig. 2.11. The plot clearly shows the vertical shift in V_{th} achieved through back-gate tuning. For a long-channel device, the threshold voltage can be shifted from approximately 460 mV with RBB to 40 mV with FBB. This plot also illustrates the impact of Short-Channel Effects (SCE). For all bias conditions, the threshold voltage begins to decrease, or “roll off,” at shorter channel lengths. This behaviour is a classic manifestation of SCE.

Another critical short-channel effect is Drain-Induced Barrier Lowering (DIBL), where the threshold voltage becomes dependent on the drain-source voltage (V_{DS}). Figure 2.12 investigates this effect by plotting V_{th} versus channel length for different values of V_{DS} while keeping the back-gate bias constant. If DIBL were significant, the curves for different V_{DS} values would

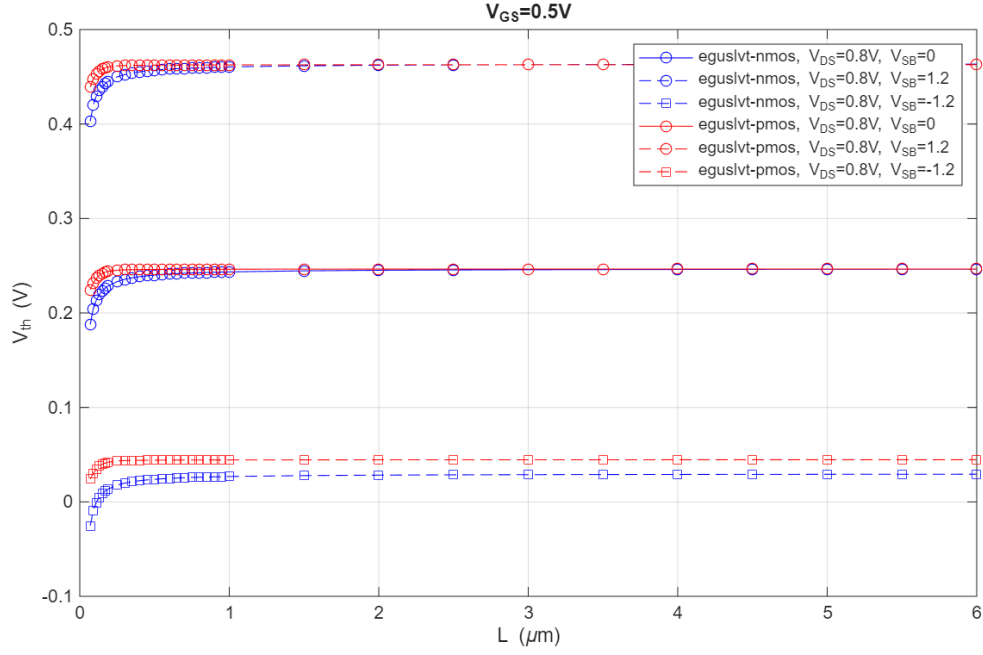


Figure 2.11: Threshold voltage (V_{th}) versus channel length (L) for different back-gate biases, illustrating V_{th} tuning and short-channel effects.

be clearly separated. As shown in the plot, the curves for both NMOS and PMOS devices are nearly overlapping. This demonstrates that the DIBL effect is minimal in this 22nm FD-SOI technology, which is a highly desirable characteristic for analog design as it contributes to a higher intrinsic gain (lower output conductance, g_{ds}).

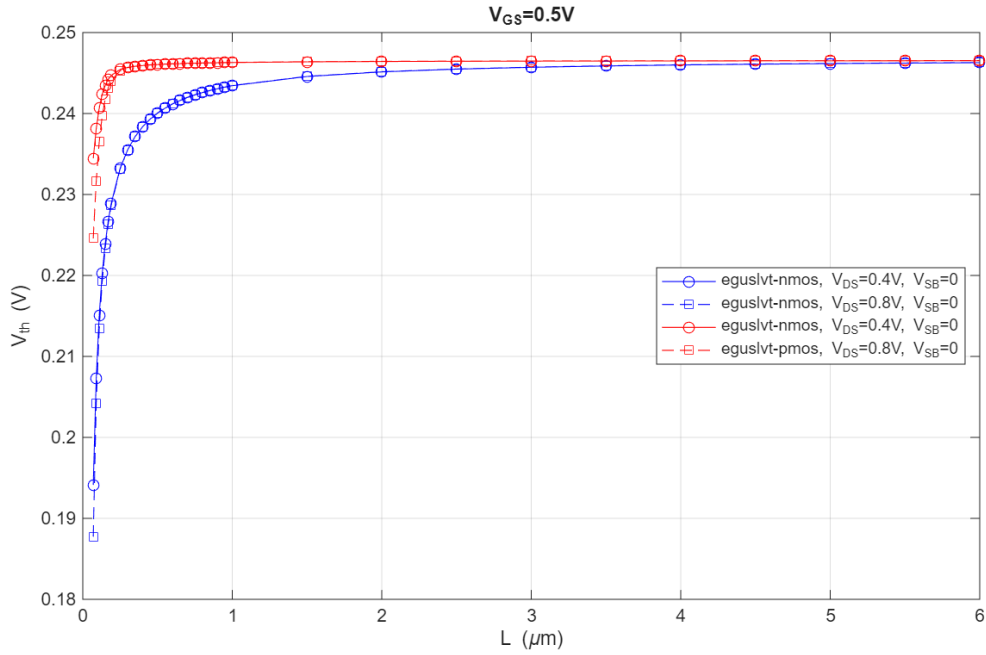


Figure 2.12: Threshold voltage (V_{th}) versus channel length (L) for different drain-source voltages (V_{DS}), showing minimal DIBL effect in both NMOS and PMOS devices.

2.4 Summary

This chapter has established a systematic design methodology based on the g_m/I_D ratio, tailored for the 22nm FD-SOI technology. The framework, which utilizes pre-computed lookup tables, was presented as a robust solution to the challenges of designing in advanced, low-voltage nodes where traditional analytical models are no longer accurate.

A comprehensive characterization of the target technology was performed. Through a series of design charts, the fundamental trade-offs between key performance metrics—such as intrinsic gain (g_m/g_{ds}), transit frequency (f_T), and power efficiency—were quantified. The noise behavior, including both thermal and flicker noise, was also characterized and integrated into the design framework. Furthermore, the unique features of FD-SOI technology were explored, demonstrating the utility of the back gate as an independent design knob for tuning the threshold voltage. It was shown that despite these large shifts in V_{th} , the normalized g_m/I_D design space remains remarkably consistent, confirming the versatility of the proposed methodology.

With the device models validated and the technology thoroughly characterized, the foundation is now set to apply this methodology to a practical circuit design. The following chapter will focus on the design of a high-performance Operational Transconductance Amplifier (OTA). The chapter will begin by calibrating the specific analytical equations for the chosen OTA topology, thereby completing the validation of the entire design framework. Subsequently, the design charts and sizing script developed in this chapter will be used to systematically size the transistors to meet a set of performance specifications.

Chapter 3

Design of a High-Performance Folded Cascode OTA for a Low-Noise Charge-Integrating Amplifier

3.1 Introduction and Design Specifications

This chapter details the practical application of the g_m/I_D methodology to the design of a low-noise current readout circuit. The chosen topology is a Charge-Integrating Amplifier (CIA), also known as a Capacitive Transimpedance Amplifier (CTIA), which consists of an operational amplifier with a feedback capacitor and a periodic reset switch. This architecture is widely used for interfacing with sensors that produce a current output.

The core of the TIA is a high-performance Operational Transconductance Amplifier (OTA). This chapter is therefore divided into two main parts. First, a single-ended folded cascode OTA is systematically designed and sized to meet the stringent requirements of a TIA, including high gain, high bandwidth, and low noise. Second, this designed OTA is integrated into the TIA topology, and the performance of the complete readout circuit is analyzed and verified.

3.1.1 System-Level Analysis of the Transimpedance Amplifier

Before defining the detailed specifications for the OTA, it is crucial to understand the system-level challenges inherent in transimpedance amplifier design, particularly when dealing with a

large sensor capacitance. While this work focuses on a Capacitive TIA (CTIA), analyzing the more common Resistive TIA (RTIA) first provides valuable insight into the fundamental issues of stability and bandwidth.

A primary challenge in TIA design is the large input capacitance (C_{in}), which is the sum of the sensor capacitance and the OTA's own input capacitance. This capacitance interacts with the feedback resistor (R_F) to create a low-frequency pole in the feedback network's transfer function[10], given by:

$$f_p = \frac{1}{2\pi R_F C_{in}}$$

A target transimpedance gain of $5\text{ M}\Omega$ is chosen, as a high gain in the mega-ohm range is required to convert the picoampere-level currents typical of nanopore sensors into a measurable voltage. When combined with a challenging input capacitance of 5 pF —a realistic value representing the parasitic load from the solid-state nanopore, packaging, and amplifier, which often falls in the $1\text{--}10\text{ pF}$ range—this creates a pole at a very low frequency of only $\approx 6.4\text{ kHz}$. This feedback pole introduces significant phase shift into the loop gain. The effect is best visualized by plotting the open-loop gain of the OTA (A_{OL}) next to the inverse of the feedback factor ($1/\beta$). As shown in fig. 3.1, the low-frequency pole in the feedback network causes the $1/\beta$ curve to rise at 20 dB/decade , leading to a 40 dB/decade rate of closure where it intersects the A_{OL} curve. This steep closure results in a phase margin near zero, guaranteeing oscillation.

To stabilize the circuit, a small compensation capacitor (C_F) is added in parallel with the feedback resistor. This capacitor introduces a zero into the feedback network that counteracts the phase shift from the input pole, restoring stability. The value of this capacitor can be calculated to achieve a target phase margin. For a critically damped response with a phase margin of approximately 65° , the optimal compensation capacitor is given by:

$$C_F = \sqrt{\frac{C_{in}}{2\pi R_F \text{ GBW}}}$$

The choice of C_F directly controls the trade-off between the closed-loop bandwidth and the stability of the TIA[10]. Figure 3.1 shows the impact of three different C_F values on the $1/\beta$ curve. A small capacitor results in a high-frequency zero and an unstable intersection, while a large capacitor ensures stability but forces a low-frequency intersection, reducing bandwidth.

An optimal capacitor ensures a stable 20 dB/decade closure at the highest possible frequency.

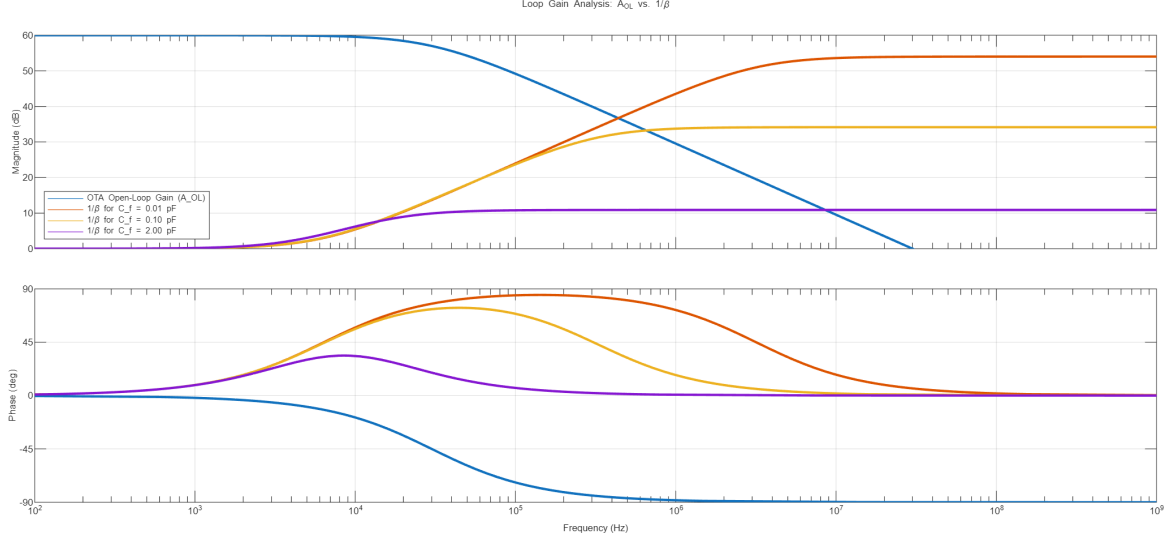


Figure 3.1: Inverse feedback factor ($1/\beta$) versus frequency for different values of the compensation capacitor (C_F), shown alongside the OTA’s open-loop gain (A_{OL}).

This frequency-domain behavior directly translates to the closed-loop transimpedance gain and the time-domain step response, as shown in fig. 3.2 and fig. 3.3, respectively.

- The underdamped case (small C_F) exhibits significant peaking in the frequency response and ringing in the step response, indicating instability.
- The overdamped case (large C_F) has no peaking but suffers from a severely reduced bandwidth and a slow, sluggish step response.
- The optimally damped case provides a maximally flat frequency response and a fast step response with minimal overshoot, representing the ideal trade-off for accurately amplifying the nanopore signal.

This analysis highlights a critical design requirement: to achieve a high closed-loop bandwidth, the OTA’s Gain-Bandwidth Product (GBW) must be significantly larger than the target signal bandwidth. The required GBW is approximately[10]:

$$\text{GBW} \geq \frac{C_{in} + C_F}{C_F} \cdot f_{-3\text{dB}}$$

A system-level bandwidth of 1 MHz is chosen for this design. While some commercial nanopore systems control the translocation speed to a slower rate (e.g., ≈ 450 bases/sec) to improve accuracy, a key challenge for next-generation, high-throughput systems is to accurately capture

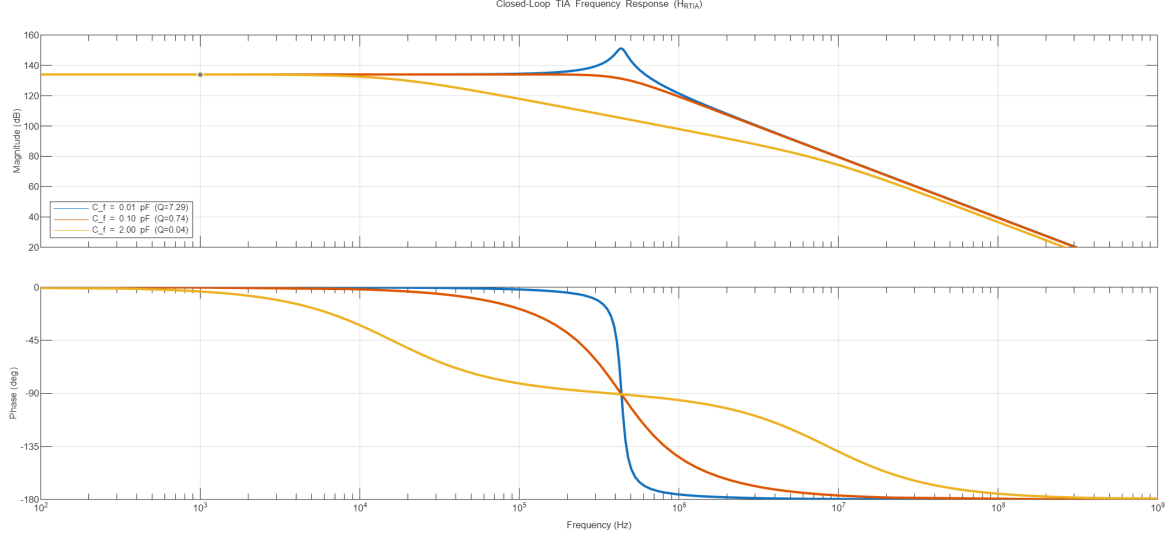


Figure 3.2: Closed-loop transimpedance gain of the TIA for different values of the feedback compensation capacitor (C_F).

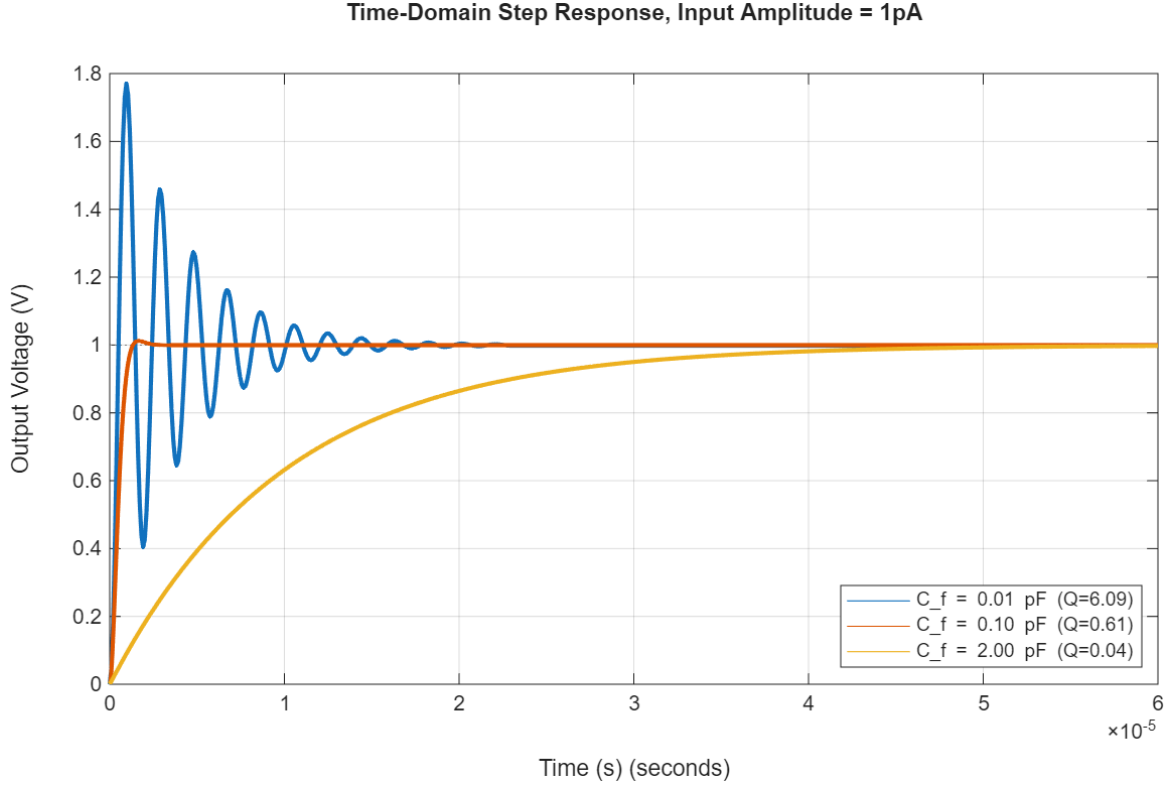


Figure 3.3: Normalized time-domain step response of the TIA for different values of the compensation capacitor (C_F).

the signal from uncontrolled molecular translocation. These events are much faster, occurring on the order of 1–3 $\mu\text{s}/\text{base}$, which requires a readout circuit bandwidth of at least 1 MHz [6, 10]. To achieve a 1 MHz bandwidth—a specification chosen to accurately capture the fast translocation events of molecules passing through the nanopore— with a typical capacitance ratio, the OTA’s GBW must be 20 to 30 times higher than the signal bandwidth. These system-

Table 3.1: Target Capacitive TIA System Specifications

Parameter	Specification	
Input Current Range	10 pA – 5 nA	Covers typical ionic currents for nanopore sensing.
Closed-Loop Bandwidth	≈ 1 MHz	Necessary to capture fast molecular translocation events.
Input Capacitance (C_p)	$= 0.5$ pF	Represents a realistic parasitic load from a solid-state nanopore[7].
Input-Referred Noise	< 20 pA _{rms}	A competitive target for high-resolution nanopore signal acquisition.
Power Consumption	< 500 μ W	Aims for high efficiency to enable scalability into large arrays.

level constraints directly inform the specifications for the core OTA, which are detailed in the following section.

3.1.2 System-Level Requirements for Nanopore Readout

Nanopore sensors generate small current signals, typically in the range of 10 pA to a few nA. To accurately capture the fast translocation events of molecules like DNA passing through the pore, the readout circuit requires a high bandwidth, often exceeding 1 MHz. A significant challenge in designing such an interface is the presence of a large parasitic capacitance at the input (C_p), which is characteristic of the nanopore setup and can be in the range of 1 – 10 pF. The readout circuit must remain stable and maintain low noise performance despite this large input capacitance.

A critical challenge in 22 nm technology is the 3 – 10 $\times$ higher flicker noise compared to older nodes, which can dominate the noise budget for low-frequency nanopore signals. The g_m/I_D methodology is essential in this context, as traditional square-law models fail completely in advanced nodes. Based on these application requirements and a review of state-of-the-art implementations, the system-level specifications for the complete TIA are defined in table 3.1.

3.1.3 OTA Design Specifications

Table 3.2: OTA Target Specifications

Parameter	Specification	
Technology	GF 22 nm FD-SOI	The chosen advanced node for low-power, high-speed design.
Supply Voltage	0.8 V	A low supply voltage suitable for the 22 nm process to minimize power.
DC Gain (A_V)	> 60 dB	Ensures accurate current integration by maintaining less than 0.1% gain error at the virtual ground.
Gain-Bandwidth (GBW)	> 25 MHz	Required to achieve 1 MHz closed-loop bandwidth and ensure stability with a 5 pF input capacitance.
Phase Margin (PM)	$> 60^\circ$	Guarantees a stable, non-oscillating response in the feedback configuration.
Input-Referred Noise	$< 5 \frac{\text{nV}}{\sqrt{\text{Hz}}}$	The dominant OTA noise must be low to meet the total TIA noise budget.
Load Capacitance (C_L)	0.5 pF	Represents the expected load from subsequent stages.
Power Consumption	$< 400 \mu\text{W}$	The majority of the TIA power budget is allocated to the core amplifier.

Achieving such system-level goals requires a carefully sized OTA. The specifications in table 3.2 are derived from the overall TIA requirements, paying special attention to the relationship between the OTA’s open-loop performance and the final closed-loop behavior of the TIA. For instance, to achieve a 1 MHz closed-loop bandwidth with a large input capacitance, the OTA’s Gain-Bandwidth Product (GBW) must be significantly higher, typically 20-30 times the target bandwidth.

3.1.4 Summary of Prior TIA Approaches

The nanopore and bio-sensor front-ends surveyed in this section demonstrate a range of Resistive TIA and Capacitive TIA architectures, each optimized for particular combinations of bandwidth, input capacitance, and power in older bulk or partially depleted nodes. However, these designs typically rely on traditional geometry-tuning and iterative SPICE refinement, rather than on a normalized g_m/I_D -plus-LUT framework calibrated to FD-SOI compact models.

Table 3.3: W and L ratios of the Folded Cascode OTA

W_1	$a,b/L_1$	$76.76\text{ }\mu\text{m}/0.2\text{ }\mu\text{m}$
W_2	$a,b/L_2$	$24.175\text{ }\mu\text{m}/0.15\text{ }\mu\text{m}$
W_3	$a,b/L_3$	$11.366\text{ }\mu\text{m}/0.15\text{ }\mu\text{m}$
W_4	$a,b/L_4$	$40.23\text{ }\mu\text{m}/0.235\text{ }\mu\text{m}$
W_5	$a,b/L_5$	$41.62\text{ }\mu\text{m}/0.235\text{ }\mu\text{m}$

3.2 Systematic Design of the Folded Cascode OTA

3.2.1 Architecture and Analytical Model

The folded cascode topology, shown in fig. 3.4, is selected for this design. This architecture offers high DC gain in a single stage, which is advantageous for stability, and provides good high-frequency performance, making it well-suited for a TIA application[10].

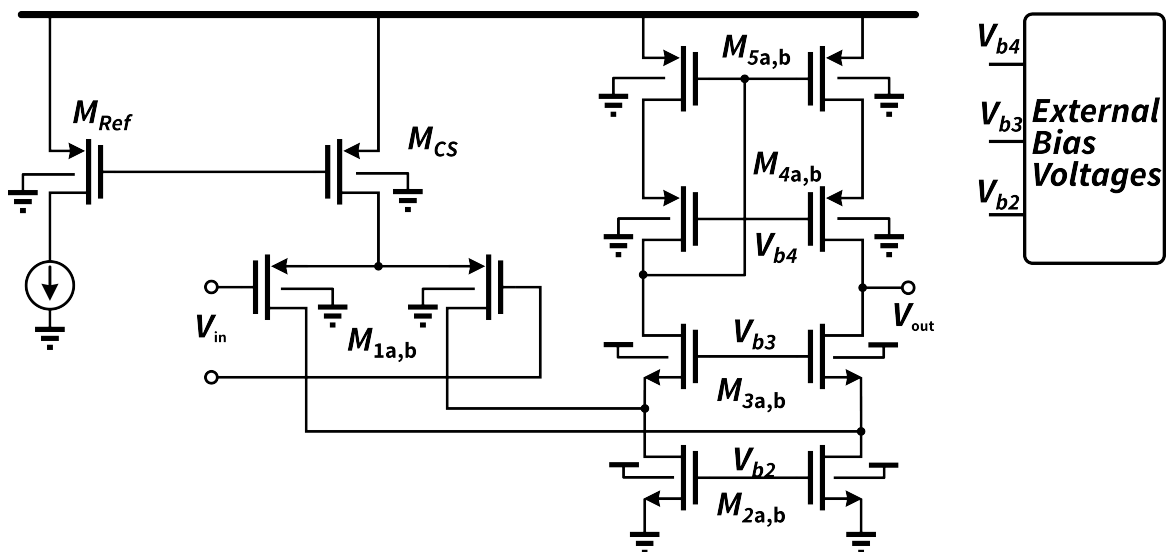


Figure 3.4: Schematic of the single-ended folded cascode OTA.

The key performance metrics of this OTA are determined by the small-signal parameters of its constituent transistors. The DC gain (A_{v0}) is set by the transconductance of the input pair (g_{m1}) multiplied by the total output resistance (R_{out}). The output resistance is the parallel combination of the resistance looking down into the NMOS cascode stack ($R_{out,n}$) and up into the PMOS cascode stack ($R_{out,p}$), where each is boosted by the intrinsic gain of the cascode devices. The DC loop gain (L_0) for the amplifier in a feedback configuration can be expressed as:

$$L_0 = \beta \cdot A_{v0} = \beta \cdot g_{m1} \cdot (R_{out,p} || R_{out,n})$$

where β is the feedback factor. These refined equations are used within the systematic sizing script to calculate the expected gain.

3.2.2 Systematic Sizing Procedure[1]

The OTA is sized using a systematic, script-based procedure that leverages the g_m/I_D lookup tables from Chapter 2. The design flow is implemented in a MATLAB script that automates the sizing procedure.

Step1: Sizing Cascode Lengths for DC Gain The first step is to determine the required channel lengths for the cascode transistors (M_2, M_3, M_4, M_5) to meet the DC gain requirement. The low-frequency loop gain (L_0) is a function of the intrinsic gains (g_m/g_{ds}) of these devices. The sizing script performs a two-dimensional sweep, varying the lengths of the NMOS cascodes ($L_{2,3}$) and PMOS cascodes ($L_{4,5}$) while keeping their operating points (g_m/I_D) fixed in moderate inversion ($15 [\frac{S}{A}]$). The resulting DC gain is calculated for each combination, generating the contour plot shown in fig. 3.5. From this plot, channel lengths of $L_{2,3} = 0.15 \mu m$ and $L_{4,5} = 0.235 \mu m$ are selected to achieve the required DC gain while keeping the lengths reasonably short to minimize parasitic capacitance.

Step2: Selection of Input Pair Geometry for Speed, Power, and Self-Loading With the cascode lengths fixed, the next step is to size the input differential pair (M_1). The required unity-gain frequency (f_u) of the amplifier is determined by the settling time (t_s) and dynamic error (e_d) specifications:

$$f_u = \frac{\ln(1/e_d)}{2\pi \cdot t_s}$$

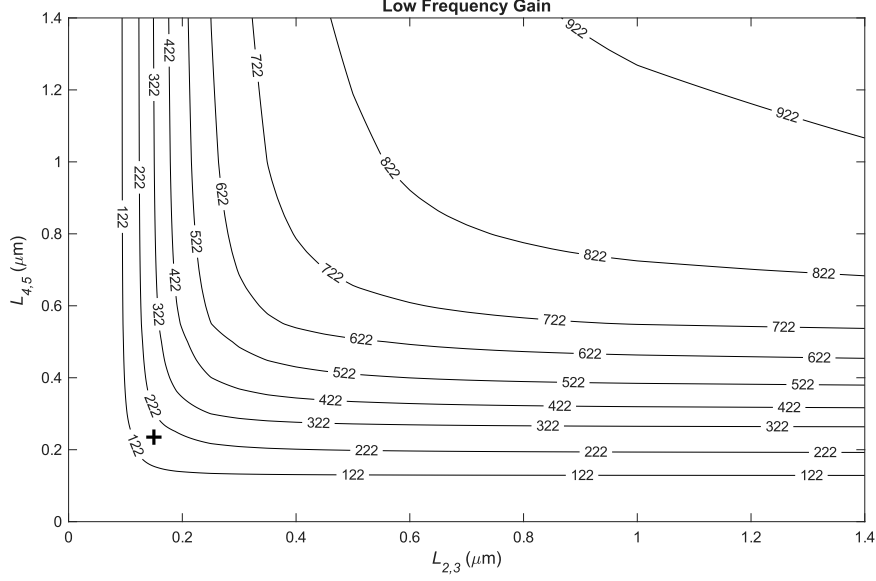


Figure 3.5: Contour plot of DC loop gain as a function of NMOS and PMOS cascode channel lengths.

The required transconductance of the input pair (g_{m1}) to achieve this speed is given by:

$$g_{m1} = 2\pi \cdot f_u \cdot \frac{C_{L,tot}}{\beta \cdot \kappa}$$

where $C_{L,tot}$ is the total load capacitance, β is the feedback factor, and κ accounts for pole-splitting effects. A key challenge is self-loading, where the parasitic capacitance of the input stage itself (C_{self}) contributes to the total load capacitance. The ratio $r_{self} = C_{self}/C_{L,tot}$ quantifies this effect. The sizing script explores this trade-off by sweeping the operating point of the input pair, $(g_m/I_D)_1$, for various channel lengths, as shown in fig. 3.6. Figure 3.6(a) shows that a higher $(g_m/I_D)_1$ requires a lower bias current, but fig. 3.6(b) shows this comes at the cost of a higher self-loading ratio.

Step3: Iterative Sizing for Self-Loading The self-loading capacitance depends on the final transistor sizes, which in turn depend on the total load, creating a circular dependency. The sizing script resolves this using an iterative convergence loop. It begins with an initial guess for self-loading ($r_{self} = 0$), identifies the design point that satisfies bandwidth constraints with minimum current, calculates the resulting parasitic capacitance at that point, and updates the load value for the next iteration. This process is repeated until the solution converges (i.e., the relative error between iterations falls below a defined tolerance), as shown in fig. 3.7.

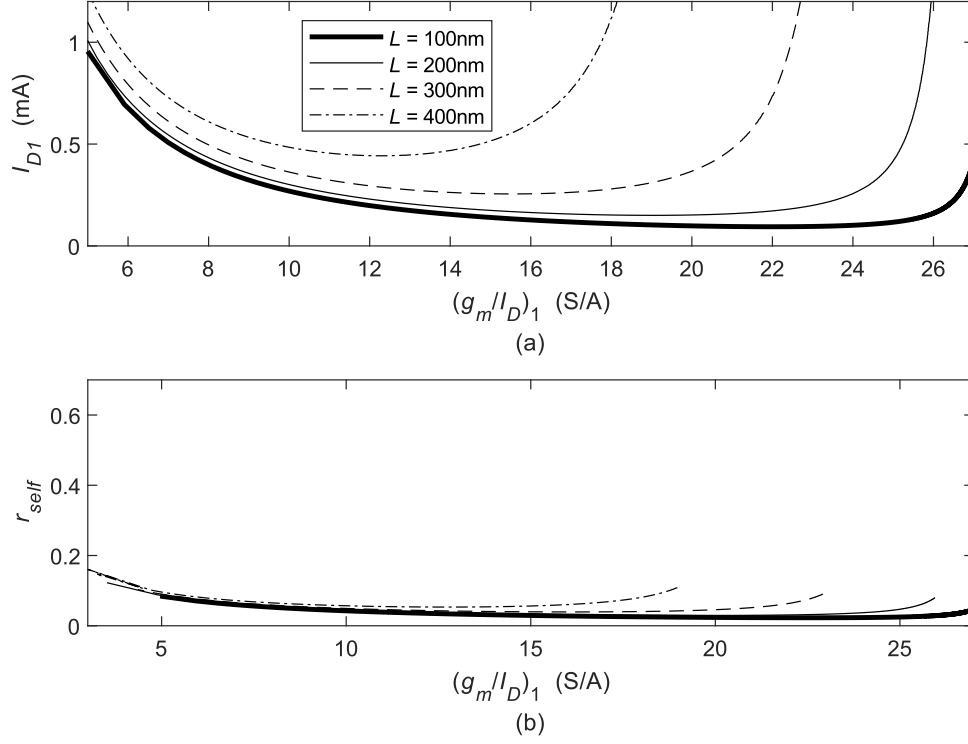


Figure 3.6: (a) Input pair current (I_{D1}) and (b) self-loading ratio (r_{self}) versus $(g_m/I_D)_1$ for different channel lengths.

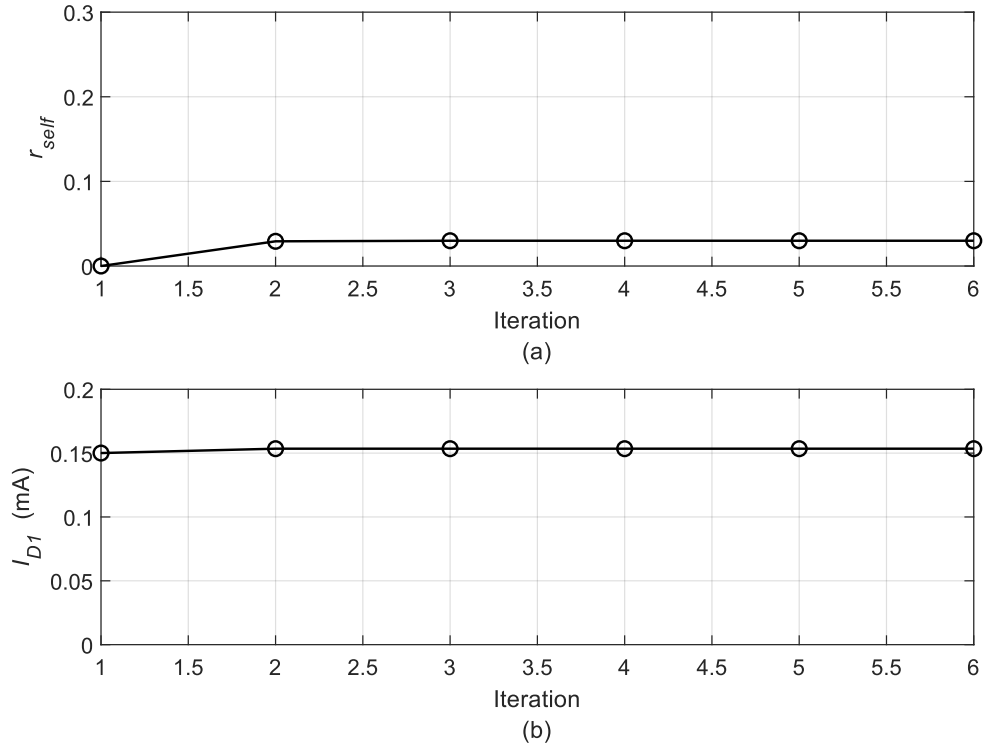


Figure 3.7: Convergence of (a) self-loading ratio (r_{self}) and (b) required input pair current (I_{D1}) over several design iterations.

Step4: Final Device Sizing Once the iterative procedure converges to an optimal operating point for the input pair— $(g_m/I_D)_1$, I_{D1} , and L_1 —the script performs the final sizing

calculations. It uses the current density (I_D/W) values from the lookup tables at the chosen operating points for all transistors to determine their required widths (W).

3.2.3 OTA Simulation and Verification

The final, fully-sized OTA schematic, derived from the systematic sizing procedure, is implemented in Cadence and verified using SPICE simulations with the BSIM-IMG model[9] for the 22 nm FD-SOI process. This verification step is essential to confirm that the performance predicted by the gm/id-based methodology translates accurately to a full transistor-level simulation.

Frequency Response

An AC analysis is performed to characterize the open-loop gain and phase response of the OTA. The resulting Bode plot is shown in fig. 3.8. From this simulation, the key frequency-domain performance metrics are extracted. The DC gain is 68 dB, which meets the target of >60 dB. The unity-gain frequency, or Gain-Bandwidth Product (GBW), is 43 MHz, exceeding the >25 MHz specification. Finally, the phase margin is measured to be 89 °, ensuring the amplifier is stable in a closed-loop configuration.

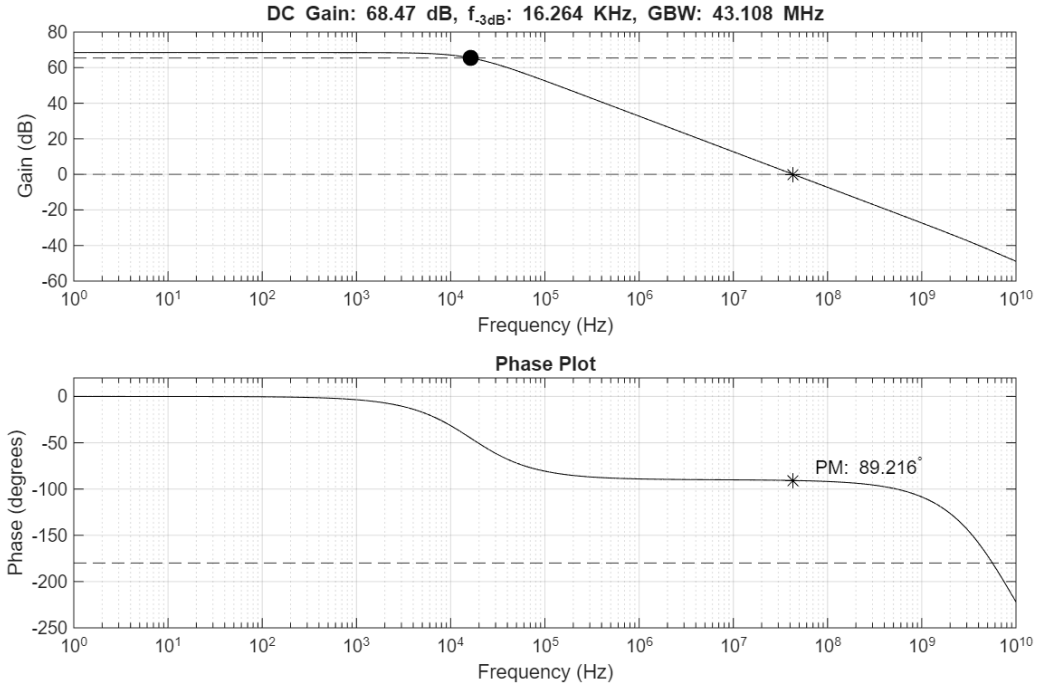


Figure 3.8: Simulated open-loop gain and phase response of the folded cascode OTA.

Noise Performance

A noise analysis is conducted to determine the input-referred noise of the OTA, with

the resulting spectrum shown in fig. 3.9. The simulation shows a thermal noise floor of approximately $0.4 \text{ [nV}/\sqrt{\text{Hz}}]$, which is below the target specification of $5 \text{ [nV}/\sqrt{\text{Hz}}]$. The plot also reveals a flicker noise corner frequency (f_{co}) of approximately 400 Hz. This confirms that the design choices made to mitigate flicker noise, such as using longer channel lengths for the active load, were effective.

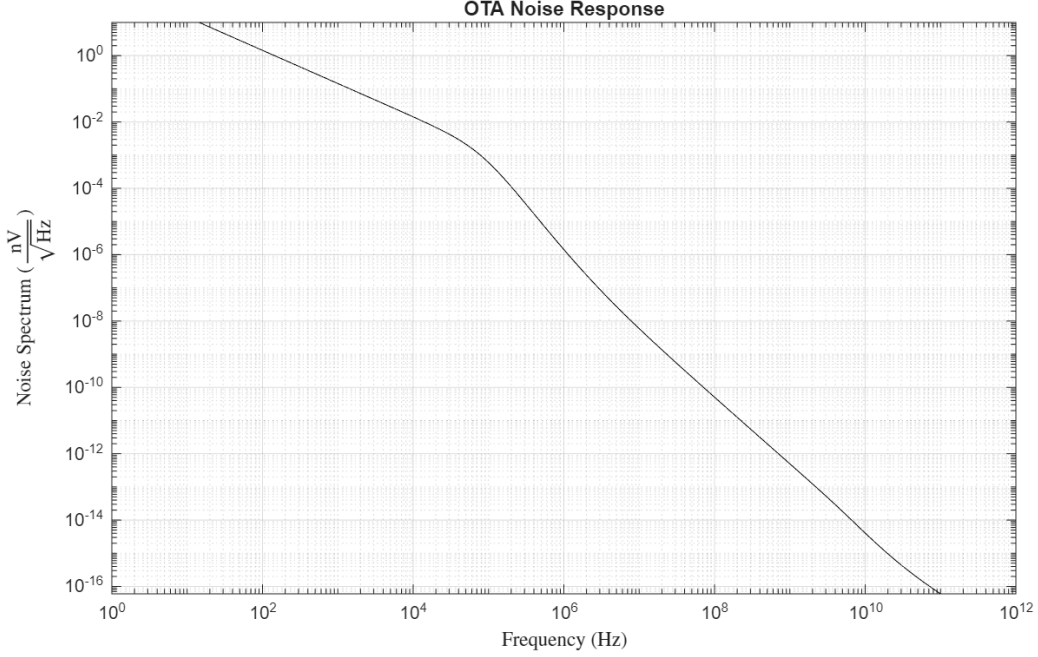


Figure 3.9: Simulated input-referred noise spectral density of the folded cascode OTA.

Performance Summary

Table 3.4 provides a comprehensive summary of the simulated OTA performance compared against the initial design specifications. The simulation results confirm that the designed OTA meets or exceeds all key performance targets. This successful outcome validates the effectiveness of the systematic g_m/I_D design methodology for implementing complex analog circuits in an advanced 22nm FD-SOI process.

3.3 Capacitive Transimpedance Amplifier (CTIA) Implementation and Analysis

With the folded cascode OTA designed and verified, the final step is to implement it as the core amplifying block in a low-noise readout circuit for the nanopore sensing application. The chosen topology is a TIA, which is well-suited for measuring the small, high-frequency ionic

Table 3.4: OTA Performance Summary

Parameter	Achieved
Technology	GF 22 nm
Supply Voltage	0.8 V
DC Gain (A_V)	68 dB
Gain-Bandwidth (GBW)	43 MHz
Phase Margin (PM)	89°
Input-Referred Noise	0.4 $\frac{\text{nV}}{\sqrt{\text{Hz}}}$
Load Capacitance (C_L)	0.5 pF
Power Consumption	200 μW

currents characteristic of nanopore sensors.

3.3.1 TIA Principle of Operation

The schematic of the TIA is shown in fig. 3.10. The circuit consists of the high-gain OTA, a feedback integration capacitor (C_F), and a periodic reset switch (S_R).

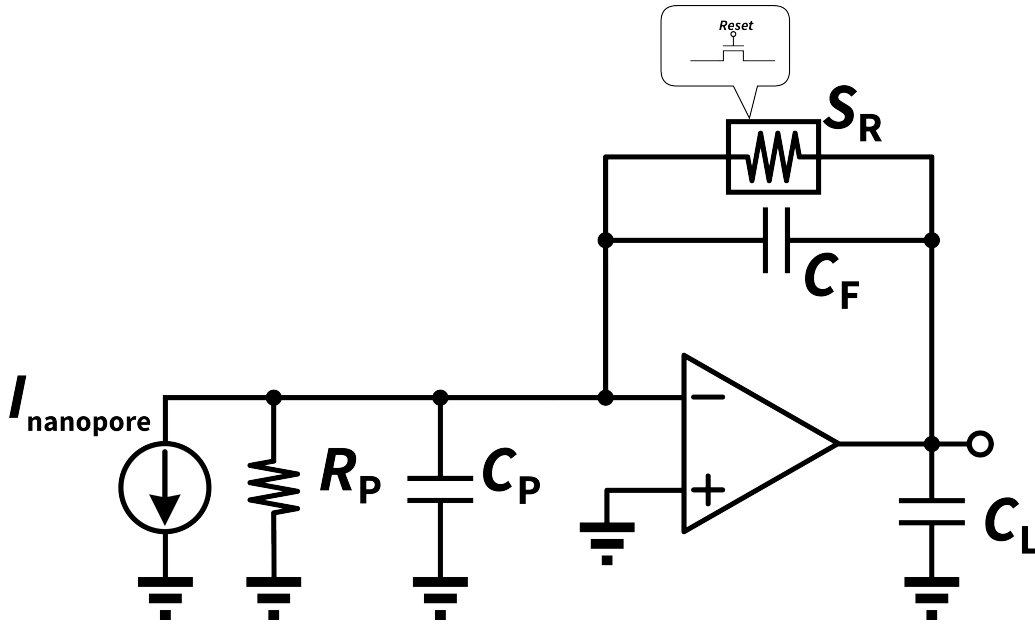


Figure 3.10: Schematic of the Transimpedance Amplifier with feedback capacitor C_F and reset switch S_R .

The circuit operates by integrating the input current from the nanopore sensor (I_{in}) onto the feedback capacitor (C_F). Because the inverting input of the OTA is a virtual ground, the input current flows directly onto C_F , causing the output voltage to ramp linearly according to

the equation:

$$V_{out}(t) = -\frac{1}{C_F} \int_0^t I_{in} dt$$

After a fixed integration time (T_{int}), the reset switch (S_R) is briefly closed, discharging the capacitor and resetting the output voltage to its initial state, preparing the circuit for the next integration cycle. The primary advantage of this topology is that by choosing an appropriate integration time, even very small currents can produce a large, easily measurable output voltage swing.

3.3.2 Noise Analysis of the TIA

The noise performance of the TIA is a critical parameter, as nanopore signals can be very small. The total output noise is primarily determined by two dominant sources[10]:

1. **OTA Input-Referred Voltage Noise:** The OTA has its own input-referred voltage noise ($v_{n,OTA}$), which is integrated by the feedback network along with the signal, contributing to the output noise.
2. **Reset (kT/C) Noise:** A fundamental noise source in this topology is the thermal noise associated with the reset switch. When the switch (S_R) is opened, the thermal noise from its channel resistance leaves a random charge on the feedback capacitor. This results in an initial random voltage offset across the capacitor at the start of each integration cycle. The variance of this noise is given by $\overline{v_{n,reset}^2} = kT/C_F$. This $\frac{kT}{C}$ noise sets a fundamental limit on the minimum detectable signal and can often be the dominant noise source in the circuit.

3.3.3 TIA Simulation and Verification

The complete TIA circuit, including the designed OTA and the reset switch, is simulated to verify its functionality and performance.

A transient simulation is performed to demonstrate the expected output voltage ramp in response to a constant input current, followed by a successful reset. The simulation, shown in fig. 3.11, confirms that the circuit correctly integrates the input current and can be periodically reset.

A final noise analysis is conducted to characterize the total output-referred noise of the

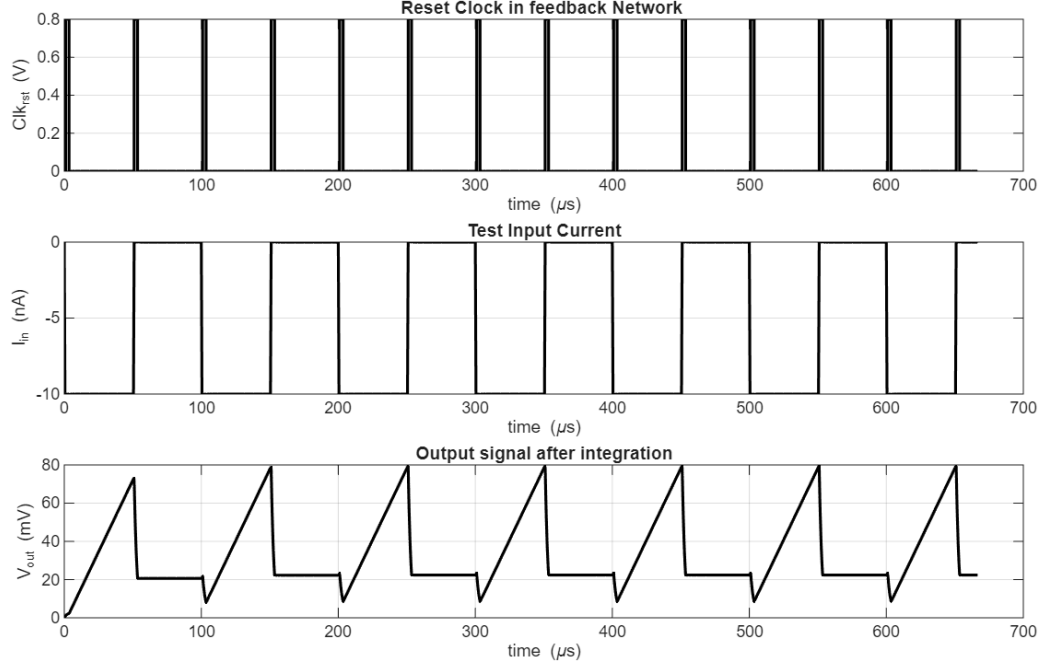


Figure 3.11: Transient simulation of the TIA showing the output voltage ramp in response to a constant input current and the reset phase.

complete TIA. The results are compared against the theoretical calculations to validate the noise model. This final verification confirms the suitability of the designed folded cascode OTA for this low-noise nanopore current readout application. The physical layout of the TIA can be seen in fig. 3.14.

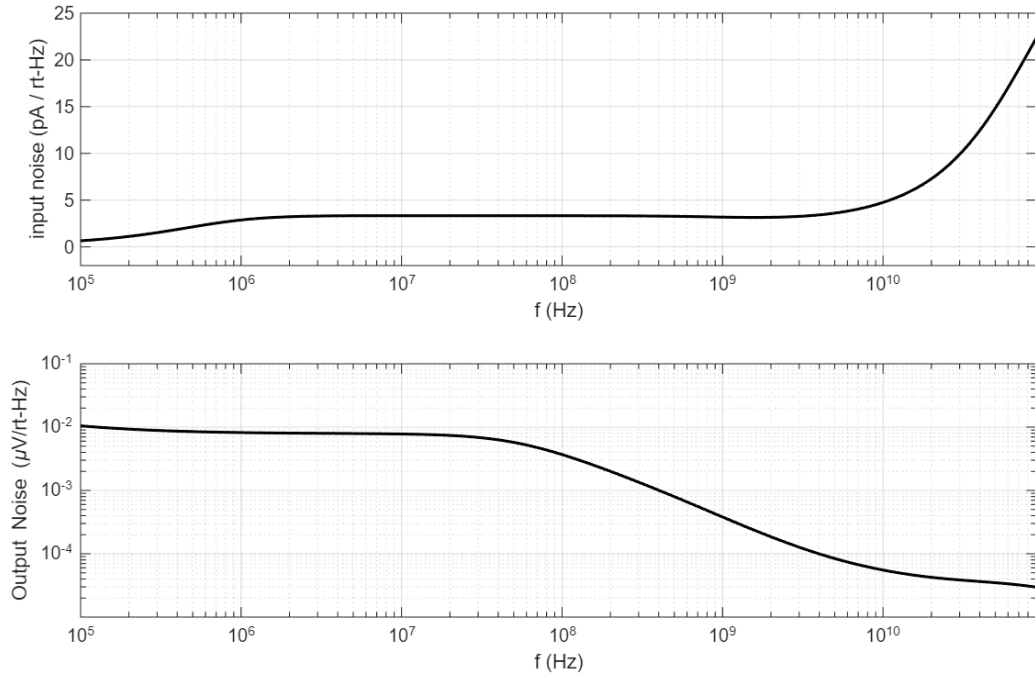


Figure 3.12: Output and Input noise power spectral densities.

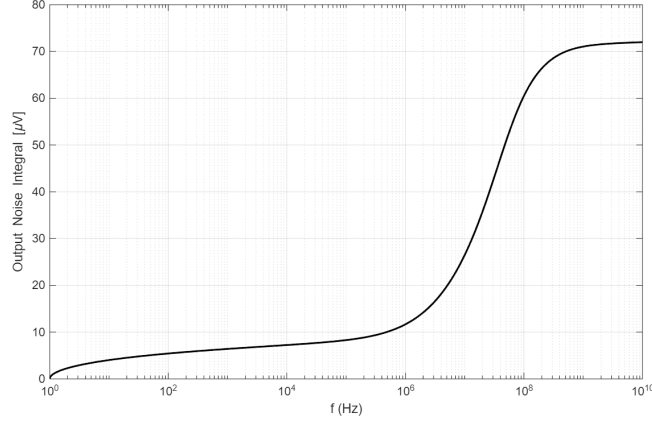


Figure 3.13: Running integral of the power spectral density.

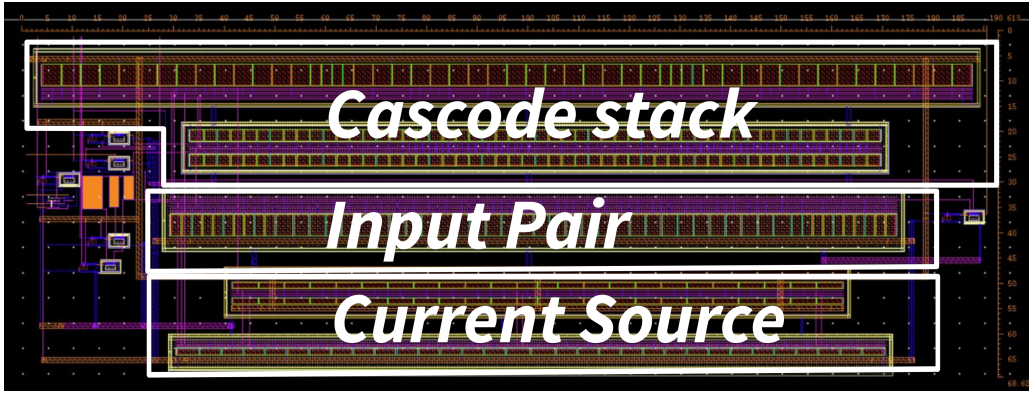


Figure 3.14: Physical Layout of the TIA in 22nm FD-SOI.

3.4 Chapter Summary and Transition

This chapter has demonstrated the practical application of the systematic g_m/I_D design methodology to a complete, high-performance analog subsystem. The design process began with a system-level analysis of the stability and bandwidth challenges inherent in transimpedance amplifiers, justifying the stringent specifications required for a low-noise nanopore readout circuit.

Following this, a single-ended folded cascode OTA was systematically designed to meet these specifications. By leveraging the g_m/I_D design charts and an iterative, script-based sizing procedure, a circuit was sized for the competing requirements of high gain, wide bandwidth, low noise, and power efficiency. The final transistor-level design was verified with SPICE simulations, which confirmed that the OTA successfully met all performance targets, validating the effectiveness of the design methodology.

Finally, the verified OTA was implemented as the core block in a capacitive TIA. The principle

Table 3.5: Performance comparison with state-of-art nanopore arrays and amplifiers

	This Work	Sync	MinION,'14	[14], '12	[15], '16	[16], '14	[17], '12 ¹	[18], '09	[19], '16
Technology	22 nm	0.13 μm -	0.18 μm	0.18 μm	0.35 μm	0.5 μm	0.35 μm	.18 μm	
Supply Voltage	0.8 V	1.2 V	-	1.5 V	1.5 V	3.3 V	3.3 V	3.3 V	1.8 V
No. of Channels	1	106	512	4	25	1	2	1	1
Power/Channel	200 μW	340 μW	2 mW	4.5 mW	5.1 mW	40 mW	1.5 mW	-	5.22 mW
Bandwidth	1 MHz	0-7.87 kHz	0-4 kHz	0-1 MHz	0-1MHz	0-10 kHz	0-10 kHz	0-1 MHz	0-10 kHz
Channel Area	0.001 59 mm ²	0.054 mm ²	0.5 mm ²	0.25 mm ²	0.16 mm ²	-	0.13 mm ²	0.35 mm ²	0.0912 mm ²
Input-Ref. Noise	~ 0.8 pArms	~ 1.5 pArms	~ 1.5 pArms	~ 155 pArms	~ 129.4 pArms	1.2 pArms	2.44 pArms	-	1.25 pArms
Input Capacitance	0.5 pF	2 pF	2 pF	3 pF	10 pF	3 pF	-	0 pF	0 pF
Integrated ADC	No	Yes	Yes	No	No	Yes	No	No	No

of operation, including key performance limiters such as kT/C noise, was discussed. The simulation of the complete TIA confirmed its functionality, demonstrating its suitability for the target nanopore sensing application.

With the design of this critical analog front-end complete, the focus of the thesis will now shift to another essential mixed-signal block. The following chapter will detail the design and implementation of a low-jitter All-Digital Phase-Locked Loop (ADPLL).

Chapter 4

Design of a Low-Jitter All-Digital Phase-Locked Loop (ADPLL)

4.1 Introduction and Design Goals

This chapter details the design and implementation of a low-jitter All-Digital Phase-Locked Loop (ADPLL). ADPLLs are critical components in modern SoCs for clock generation and frequency synthesis, offering significant advantages over their analog counterparts in terms of scalability, reconfigurability, and process portability[8]. The architecture presented in this chapter is based on the robust, all-static CMOS design described in [3, 8], which has demonstrated a wide tuning range and a wide power supply range, making it a versatile and power-efficient topology.

The design is implemented in the target 22nm FD-SOI process and is optimized to meet the specifications for a high-performance, low-power clocking system. The target specifications, derived from the state-of-the-art, are summarized in table 4.1.

4.2 ADPLL System-Level Architecture

The top-level architecture of the proposed ADPLL, shown in fig. 4.1, follows the all-digital approach presented in [3]. The loop consists of four main digital or digitally-controlled blocks: a Phase Frequency Detector (PFD), a Time-to-Digital Converter (TDC), a Digital Loop Filter (DLF), and a Digitally Controlled Oscillator (DCO).

Table 4.1: ADPLL Target Specifications

Parameter	Specification	
Technology	GF 22 nm FD-SOI	The chosen advanced node for low-power, high-speed design.
Supply Voltage	0.5 V – 0.9 V	A wide range is targeted for flexibility and low-power operation capability.
Frequency Range	≈ 100 MHz – 5 GHz	A wide tuning range suitable for diverse SoC clocking needs.
Reference Frequency	100 MHz	A standard reference clock frequency.
RMS Jitter	<0.2 ps	Targeting ultra-low jitter for high-performance applications.
Phase Noise @ 1 MHz	< -85 dBc/Hz (@ 5.2 GHz)	A competitive target for spectral purity at high frequencies.
Power Consumption	< 4 mW (@ 0.9 V, ≈ 5 GHz)	Aims for high power efficiency across the operating range.
Area	< 0.003 mm ²	Minimizing area is crucial for SoC integration.

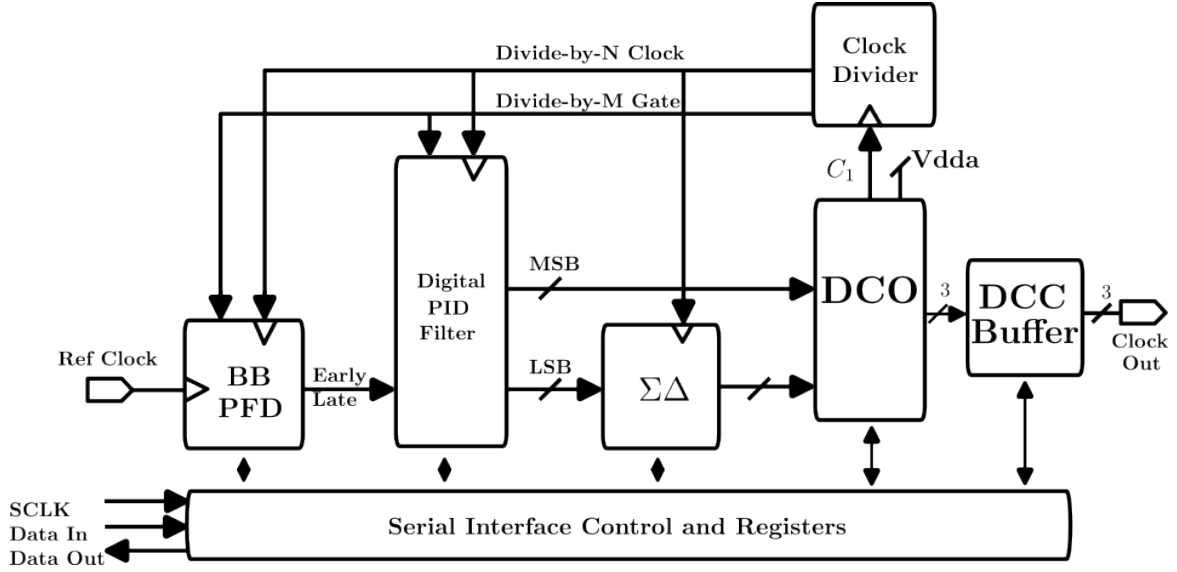


Figure 4.1: Top-level block diagram of the All-Digital PLL [3].

The PFD detects the phase and frequency difference between the input reference clock (REF) and the divided DCO output clock (DIV). The TDC then digitizes this time difference into a digital code. This code, representing the phase error, is processed by the DLF, which implements a proportional-integral control law to generate a multi-bit control word for the DCO. Finally, the DCO adjusts its output frequency based on this control word, completing the negative feedback loop and driving the phase error towards zero.

4.3 Circuit-Level Implementation and Analysis

4.3.1 Phase and Frequency Detector (PFD)

The core of the ADPLL's phase comparison mechanism is a Bang-Bang Phase and Frequency Detector (BB-PFD). This architecture represents a departure from conventional linear PFDs, which output a signal proportional to the phase error. Instead, the BB-PFD provides a binary, "bang-bang" output, indicating only whether the reference clock is leading or lagging the feedback clock. This binary phase detection approach, pioneered for clock recovery circuits[20, 21], provides only sign information but offers significant advantages in digital implementation[22]. Most importantly, it provides an effectively infinite dynamic range for frequency capture[23], eliminating the cycle-slipping problem that limits linear PFDs to a $\pm 2\pi$ phase detection range. The entire circuit is self-timed and asynchronous, drawing heavily on principles of delay-insensitive and quasi-delay-insensitive design[24], making it robust against process, voltage, and temperature (PVT) variations[3].

4.3.1.1 Circuit Architecture and Operation

The operational flow of the BB-PFD, shown in the fig. 4.2, can be broken down into four key stages: asynchronous edge detection, mutual exclusion arbitration, metastability handling, and a self-timed reset.

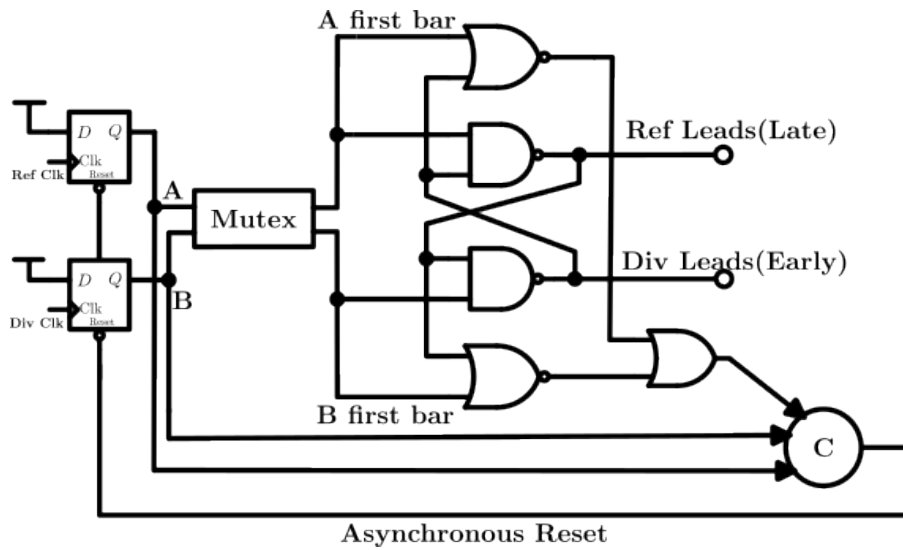


Figure 4.2: BB-PFD schematic.

- **Asynchronous Edge Detection:**

The PFD's two inputs, one for the reference clock (REF) and one for the divided DCO clock (DIV), are monitored by independent, event-driven input latches. Upon the arrival of a rising edge on its respective input, each latch transitions from an idle to an active state, setting a flag that indicates the edge has been detected.

- **Mutual Exclusion (Mutex) Arbiter:**

The decision of which clock edge arrived first is made by a mutual exclusion element, or mutex shown in fig. 4.3, implemented with cross-coupled NOR gates. This circuit acts as an arbiter[25, 26]. When an edge arrives first on one input, its corresponding NOR gate's output is driven high. This high signal is cross-coupled to the other NOR gate, forcing its output to remain low, even when the second, later edge eventually arrives. The mutex effectively locks in its "winner-take-all" decision and provides an unambiguous determination of the lead/lag relationship.

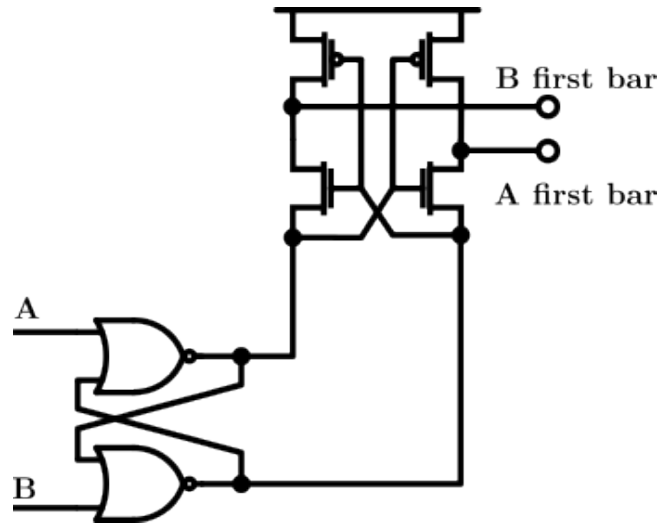


Figure 4.3: Mutex schematic.

- **Metastability Handling:**

When the REF and DIV edges arrive nearly simultaneously, the mutex can enter a metastable state[27] where its outputs hover at an indeterminate voltage level. To prevent the propagation of this uncertain state, a metastability filter is placed after the mutex. This filter waits until the two outputs of the mutex differ by at least a transistor threshold voltage before allowing the signal to pass. This ensures that the metastable state has resolved into valid logic levels[28], guaranteeing that only clean, regenerated signals reach the subsequent flip-flop. The importance of modeling metastability in bang-bang

detectors is highlighted in [29]

- **Self-Timed Reset Mechanism:**

After the mutex decision is captured by a set-reset (SR) flip-flop, the circuit must reset itself to prepare for the next comparison cycle. This is coordinated by a Muller C-element[30], a fundamental block in asynchronous design[31]. The C-element's output transitions only when all of its inputs agree. It monitors three conditions: the reference edge has been detected, the divided clock edge has been detected, and the SR flip-flop has successfully latched the result. Only when all three conditions are met does the C-element generate a reset pulse. This pulse clears the input latches, which in turn causes the C-element's output to return low, ending the reset. This self-timed loop ensures that a reset never occurs prematurely [32, 33].

4.3.1.2 Performance Advantages

The self-timed, bang-bang architecture provides several fundamental performance advantages that are critical for high-performance ADPLL design.

- **Infinite Dynamic Range and Unified Detection:** The most significant advantage is its infinite dynamic range. Unlike linear PFDs, the BB-PFD always correctly identifies which clock is faster, regardless of the magnitude of the frequency or phase error. This eliminates cycle-slipping and allows the PLL to achieve lock from a much wider range of initial frequency offsets[23] without requiring auxiliary frequency acquisition loops. The same circuit seamlessly transitions from coarse frequency detection (where the output remains in one state for multiple cycles) to fine phase detection (where the output provides cycle-by-cycle lead/lag information) without any mode switching, similar in concept to digital PFDs designed for fast acquisition[34] or phase-domain ADPLLs [23].
- **Dead-Zone Elimination:** The BB-PFD architecture has no dead zone. The mutex arbiter will eventually resolve a decision for any non-zero phase difference, no matter how small[22]. This ensures that the loop receives continuous correction information even when very close to the lock point, preventing the limit-cycle oscillations that degrade jitter performance in PFDs with a dead zone.
- **Inherent PVT Robustness:** The self-timed nature of the reset loop provides inherent

robustness to PVT variations[35]. Circuit delays may change with process, voltage, or temperature, but the handshaking protocol ensures that the reset operation only occurs after the current comparison is definitively complete[32]. The circuit’s timing adapts automatically, eliminating the need for the large design margins required in synchronous designs to guarantee timing closure across all corners.

4.3.1.3 PFD Operational Modes: Frequency and Phase Detection

The BB-PFD seamlessly handles both large frequency differences and small phase differences without requiring separate modes of operation.

- **Frequency Acquisition (Large Error):** When there is a significant frequency difference, edges from the faster clock will consistently arrive before the PFD has finished resetting from the previous cycle. This causes the PFD’s output to remain latched in one state (e.g., “reference leads”) for many consecutive cycles. The digital loop filter integrates this constant signal, causing the DCO to rapidly slew its frequency up or down until it is close to the reference.
- **Phase Lock (Small Error):** Once the frequencies are nearly matched, the PFD provides cycle-by-cycle lead-lag information. In this regime, the PFD completes a full detect-arbitrate-reset cycle for every clock period. The output toggles based on the small phase error, and the loop filter integrates these binary decisions to maintain a precise phase lock. The absence of a “dead zone” ensures that even infinitesimally small phase errors generate a corrective signal, preventing limit-cycle oscillations around the lock point.

4.3.1.4 PFD Simulation and Verification

To validate the theoretical operation and performance of the BB-PFD, extensive circuit-level simulations were performed. A dedicated testbench was created to sweep the static time difference between the REF and DIV input edges and measure the average PFD output fig. 4.4.

The resulting PFD characteristic curve is presented in fig. 4.5. This simulation confirms the PFD’s intended “bang-bang” operation. The output is a binary ‘low’ state (0) for any negative time difference (DIV leads) and a binary ‘high’ state (1) for any positive time difference (REF leads). The transition is extremely sharp and centered precisely at zero, which demonstrates the absence of a dead zone. This high sensitivity is critical as it prevents the limit-cycle oscillations

that can degrade jitter performance in PFDs with a dead zone.

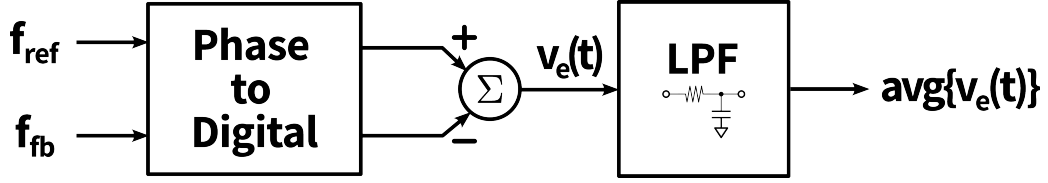


Figure 4.4: PFD testbench schematic.

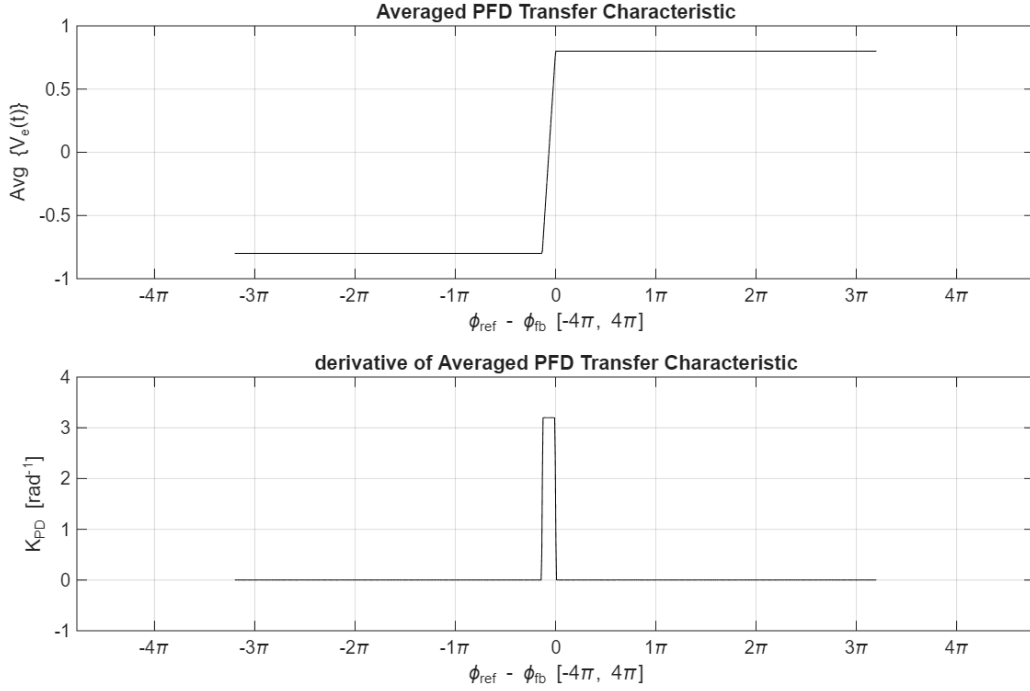


Figure 4.5: BB-PFD transfer characteristics to show how it is working as P2D.

A noise analysis was also conducted to characterize the noise contribution of the PFD block itself. While the PFD's binary output is inherently robust to amplitude noise, the timing uncertainty (jitter) of its internal logic, particularly in the mutex and reset path, can contribute to the overall PLL noise floor. The simulation results in fig. 4.6 confirmed that the PFD's intrinsic noise contribution is minimal and well below the noise generated by the DCO, ensuring that the PFD does not become a limiting factor in the PLL's jitter performance.

Finally, the physical layout of the BB-PFD was completed in the 22nm FD-SOI process, as shown in fig. 4.7. Careful layout techniques were employed, particularly for the asynchronous and self-timed components, to ensure symmetrical routing and minimize timing skew between the critical REF and DIV paths.

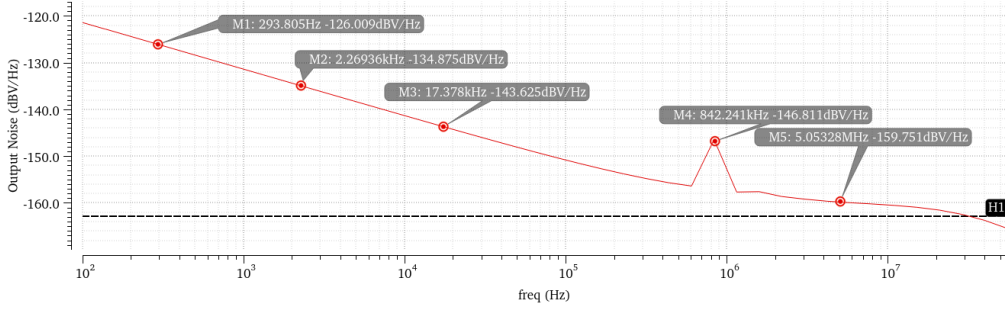


Figure 4.6: BB-PFD phase noise.

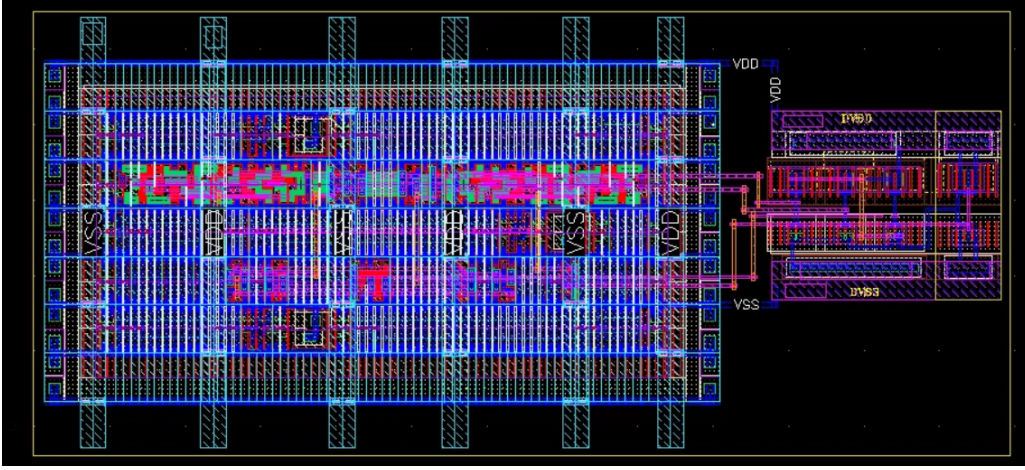


Figure 4.7: BB-PFD Layout.

4.3.2 Digital Loop Filter (DLF)

The Digital Loop Filter (DLF) is the core computational block of the ADPLL. Its primary function is to convert the binary phase error signal from the PFD into a multi-bit, high-precision control word that steers the DCO's frequency. The DLF implements the loop dynamics necessary for stable locking and optimal performance.

4.3.2.1 PID Controller Architecture

The architecture is a fully synchronous, pipelined Proportional, Integral, Differential (PID) controller[3, 36], as shown in the block diagram in fig. 4.8, which provides robust control over the loop's transient and steady-state behavior.

The DLF consists of three parallel paths that collectively determine the DCO control word:

- **Integral (I) Path:** This path is the heart of the loop filter, providing long-term frequency control. It accumulates the PFD error signal over time, multiplied by a programmable integral constant (K_i). This integration ensures that any static (DC) phase offset is driven

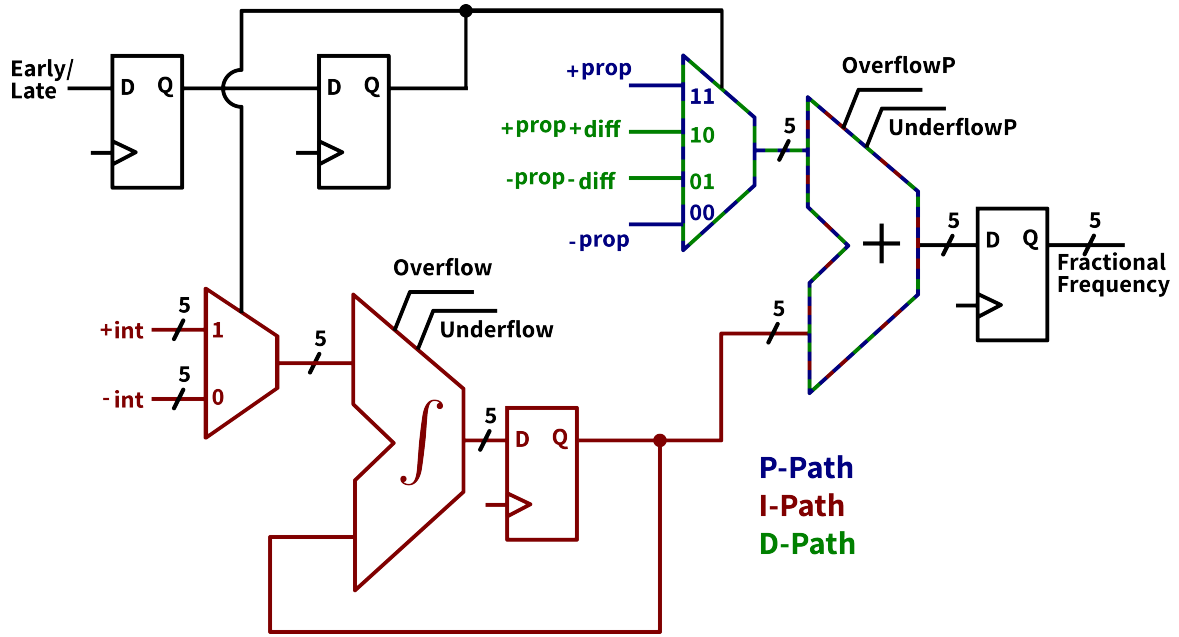


Figure 4.8: Digital Loop Filter block diagram - PID.

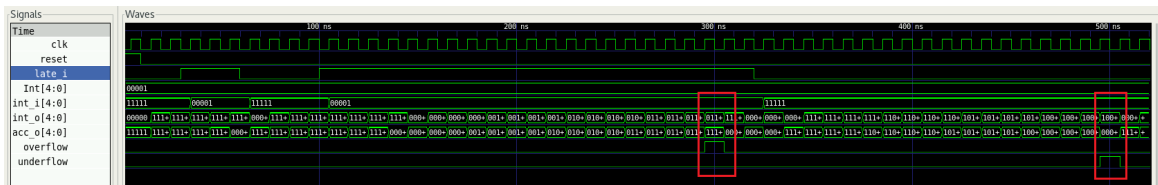


Figure 4.9: Digital Loop Filter block's simulation results.

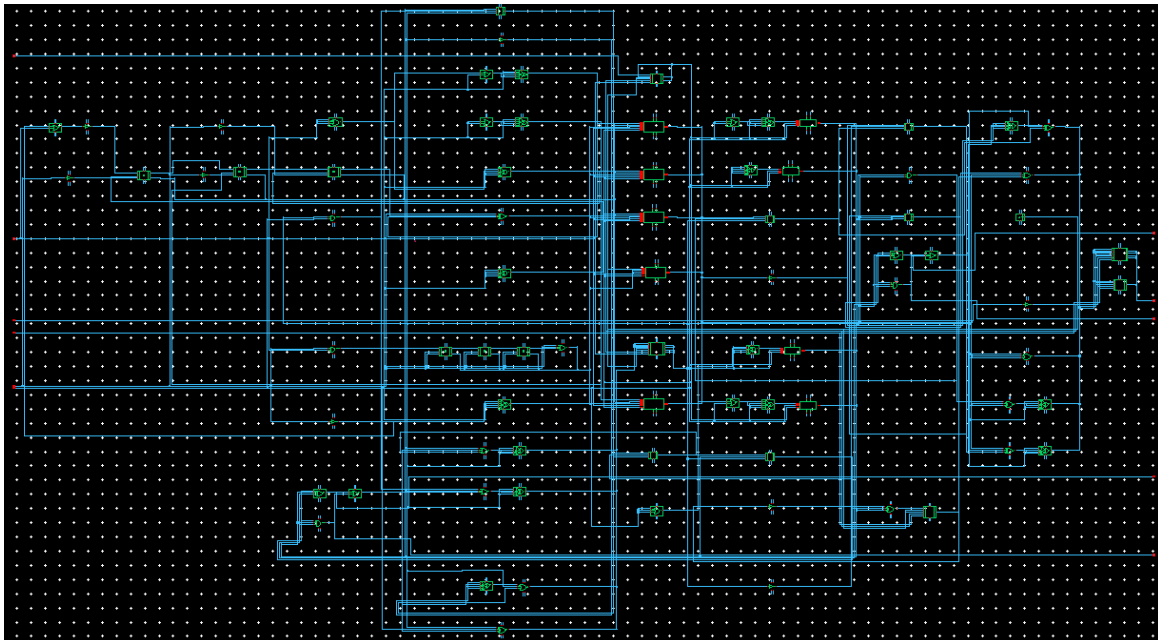


Figure 4.10: Synthesized Digital Loop Filter block

to zero[37], guaranteeing a precise and accurate phase lock. The accumulator's output represents the main frequency setting for the DCO.

- Proportional (P) Path: This path provides an immediate correction that is proportional

to the current PFD error signal, scaled by a proportional constant (K_p). Unlike the integral path, it has no memory of past errors. Its purpose is to improve the loop's transient response, reducing the settling time after a phase or frequency step.

- **Differential (D) Path:** This path approximates a derivative, responding to the rate of change of the phase error. It provides predictive correction, adding damping to the loop[37]. This damping is critical for suppressing overshoot and ringing during the locking process, leading to a more stable response.

The digital logic for this PID controller was implemented in Verilog. Functional verification was performed through simulation, as shown in fig. 4.9, to ensure the filter's output correctly responds to various input stimuli according to the programmed proportional, integral, and differential gains.

Finally, the verified Verilog code was synthesized into a physical layout for the 22nm FD-SOI process, resulting in the compact, standard-cell-based implementation shown in fig. 4.10.

4.3.2.2 Implementation and Operation

The DLF is clocked at the divided output clock frequency (f_{DIV}), which matches the update rate of the PFD. This ensures computational efficiency, as a new DCO control word is calculated only when new phase information is available.

A key architectural feature is the two-stage accumulation process that achieves high frequency resolution. The DLF itself implements the five least significant bits of the frequency control word using binary arithmetic. This "fractional frequency" command represents a fine adjustment within the range of a single DCO inverter step. When this 5-bit accumulator overflows or underflows, it sends an increment or decrement command to the DCO controller. The DCO controller, which manages the 768 parallel inverters, acts as the most significant part of the accumulator. This creates an effective accumulator precision of 14.5 bits (5 bits from the DLF + $\log_2(768) \approx 9.5$ bits from the DCO), achieving high resolution without the complexity of a large, single-stage multi-bit adder.

The proportional and differential corrections are applied after the integral path. This reduces the latency of the fast proportional path, allowing it to respond more quickly to high-frequency phase disturbances and improving the overall jitter performance.

4.3.2.3 Interface to the DCO and Sigma-Delta Modulation

The 5-bit fractional frequency command from the DLF cannot directly control a physical inverter, as each inverter can only be fully ON or fully OFF. This discrepancy creates quantization error at the DCO output. Therefore, a third-order MASH Sigma-Delta Modulator ($\Sigma\Delta M$) is used to convert this fractional value into a time-averaged sequence of binary control signals[38, 39]. For example, a fractional command of 0.25 would be converted into a dithering pattern that enables a specific inverter for, on average, one out of every four clock cycles. This process, known as dithering, provides the DCO with a sub-inverter frequency resolution[3]. The specific architecture, shown in fig. 4.11, is a MASH 1-1-1 $\Sigma\Delta M$ [40]. It cascades three first-order accumulator stages to produce a 7-bit dithering pattern. This 7-bit output is composed of the carry-out bits from each stage, combined through digital error cancellation logic. As detailed in the reference design [3], the output is structured as 1 bit from the first stage (C_1), 2 bits from the second (C_2), and 4 bits from the third (C_3).

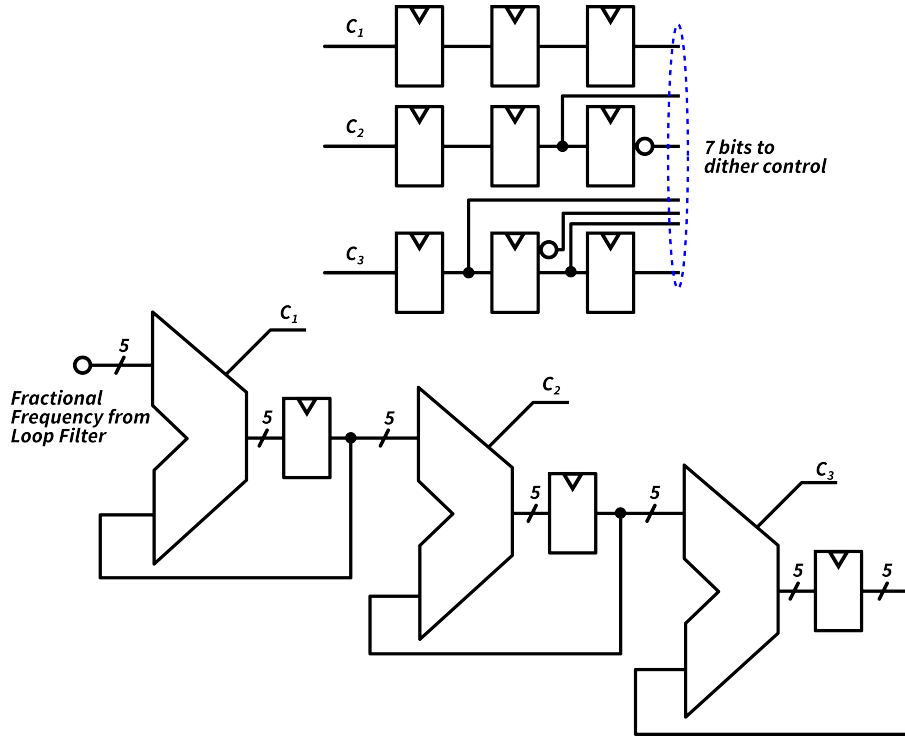


Figure 4.11: MASH Block Diagram.

This architecture uses delay elements (implemented as D flip-flops, $D = z^{-1}$) to perform two critical functions:

- **Synchronization:** They align the carry-out signals from the sequential accumulator stages so that they arrive at the output logic simultaneously.

- **Differentiation:** They perform z-domain subtraction to implement the error cancellation logic, (e.g., $C_2 \cdot (D^2 - D^3)$), which cancels the lower-order quantization errors from the first and second stages.

The result of this MASH architecture is that the first and second-order quantization errors are cancelled, leaving only a residual third-order quantization error. This error is “noise shaped” with a third-order high-pass characteristic, proportional to $(1 - z^{-1})^3$ [40].

This high-pass shaping is critical as it pushes the quantization noise power to high frequencies, well outside the PLL’s loop bandwidth, where it is effectively filtered out by the loop’s inherent low-pass characteristic. This achieves the fine resolution required for low in-band phase noise (e.g., -112 dBc/Hz at 1 MHz offset) and minimizes the spurious tones that would otherwise occur.

4.3.3 Digitally Controlled Oscillator (DCO)

The Digitally Controlled Oscillator (DCO) is the core frequency-generating element of the ADPLL. Its output frequency is directly controlled by the 14-bit digital word from the Digital Loop Filter (DLF). The 9 most significant bits (MSBs) of this word are used for coarse frequency tuning via the pseudo-thermometer control, while the 5 least significant bits (LSBs) provide the fractional frequency command to the MASH $\Sigma\Delta$ M for fine resolution. The design philosophy adopted for this work is a fully digital approach, utilizing a ring oscillator composed entirely of standard digital logic gates, rather than a hybrid architecture employing an analog Voltage-Controlled Oscillator (VCO) driven by a Digital-to-Analog Converter (DAC).

4.3.3.1 Design Philosophy: Fully Digital vs. Hybrid DAC + VCO

The choice of a fully digital DCO offers significant advantages aligned with modern VLSI trends[41, 42]. Firstly, it eliminates the need for a high-resolution, low-noise DAC, which is a complex analog block sensitive to process variations and difficult to scale across technology nodes. Secondly, the DCO, being constructed from digital inverters and control logic, scales naturally with process technology, offering better portability and predictability compared to analog VCOs. Digital circuits are also easier to model and simulate accurately using standard digital design flows. Furthermore, the power consumed by the DCO directly contributes to the

oscillation energy, potentially leading to better intrinsic phase noise performance compared to a hybrid approach where the DAC represents power overhead. While the area of a large inverter-based ring oscillator may exceed that of a compact LC-tank VCO core, the overall system area is often comparable or smaller when considering the area required for a high-performance DAC and associated analog circuitry.

4.3.3.2 DCO Architecture: Three-Stage Ring Oscillator

The DCO is implemented as a three-stage ring oscillator, the minimum number required for oscillation in CMOS. This configuration provides the highest oscillation frequency for a given inverter delay ($f_{osc} = 1/(6 \cdot t_{delay})$). The fundamental building block for each stage is the tri-state inverter, shown in fig. 4.12, which can be digitally enabled or disabled to place its output in a high-impedance state. Each stage consists of 271 parallel tri-state inverters. The oscillation frequency is tuned by digitally enabling or disabling these inverters. Enabling N inverters in parallel provides approximately N times the drive current for a relatively constant capacitive load (dominated by the next stage's input capacitance), thereby reducing the stage delay (t_{delay}) and increasing the oscillation frequency. The inverters are physically arranged in a 17-row by 48-column array, as one array of it is illustrated in fig. 4.13. Columns are interleaved across the three oscillator phases to ensure symmetry, minimize local loading, and distribute heat evenly. Rows 0-15 implement the primary frequency control, while row 16 contains specialized inverters. Figure 4.14 further details how the control logic is formed and connected to this array to manage the individual inverters.

4.3.3.3 Frequency Control: Pseudo-Thermometer Code (PTC)

To control the 768 tunable inverters (distributed across rows 0-15), a Pseudo-Thermometer Code (PTC) scheme is employed[3]. This approach provides a compromise between the control logic complexity of pure binary encoding and the large bit-width requirement of pure thermometer encoding. PTC uses a coarse thermometer code for row selection and a fine thermometer code for column selection within the active row. The key advantage is that incrementing or decrementing the frequency requires changing the state of only one inverter at a time, ensuring glitch-free frequency transitions[43], similar to pure thermometer control, but with significantly reduced control logic overhead. The row and column control logic uses latches and shifters,

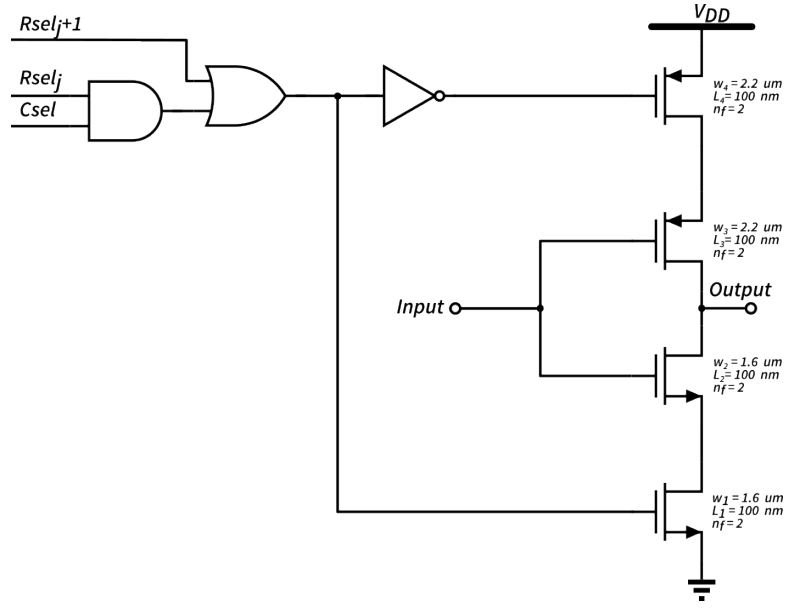


Figure 4.12: Tri-state Inverter used in the DCO

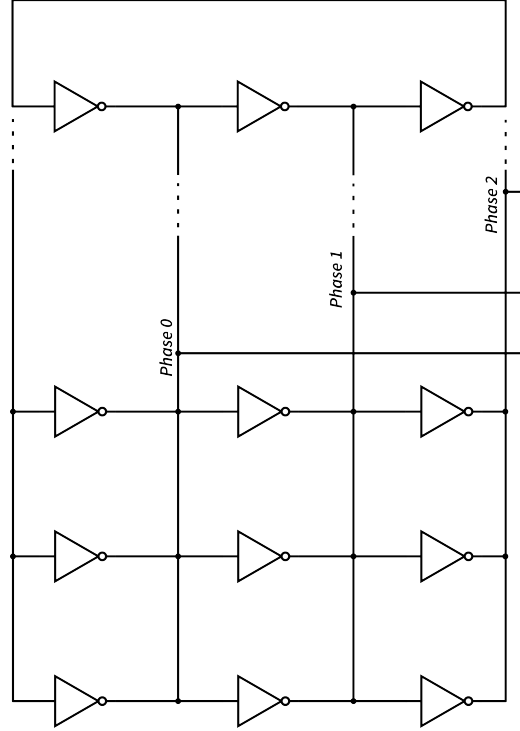


Figure 4.13: Array of inverters showing the three-phase, interleaved column structure and 17 rows.

with adjacent rows employing opposite polarity column control signals to simplify row-to-row transitions. Intelligent clock gating disables the control logic when no frequency change is requested[44, 45], reducing dynamic power consumption and switching noise.

4.3.3.4 Specialized Inverters (Row 16)

The top row (row 16) houses several specialized sets of inverters:

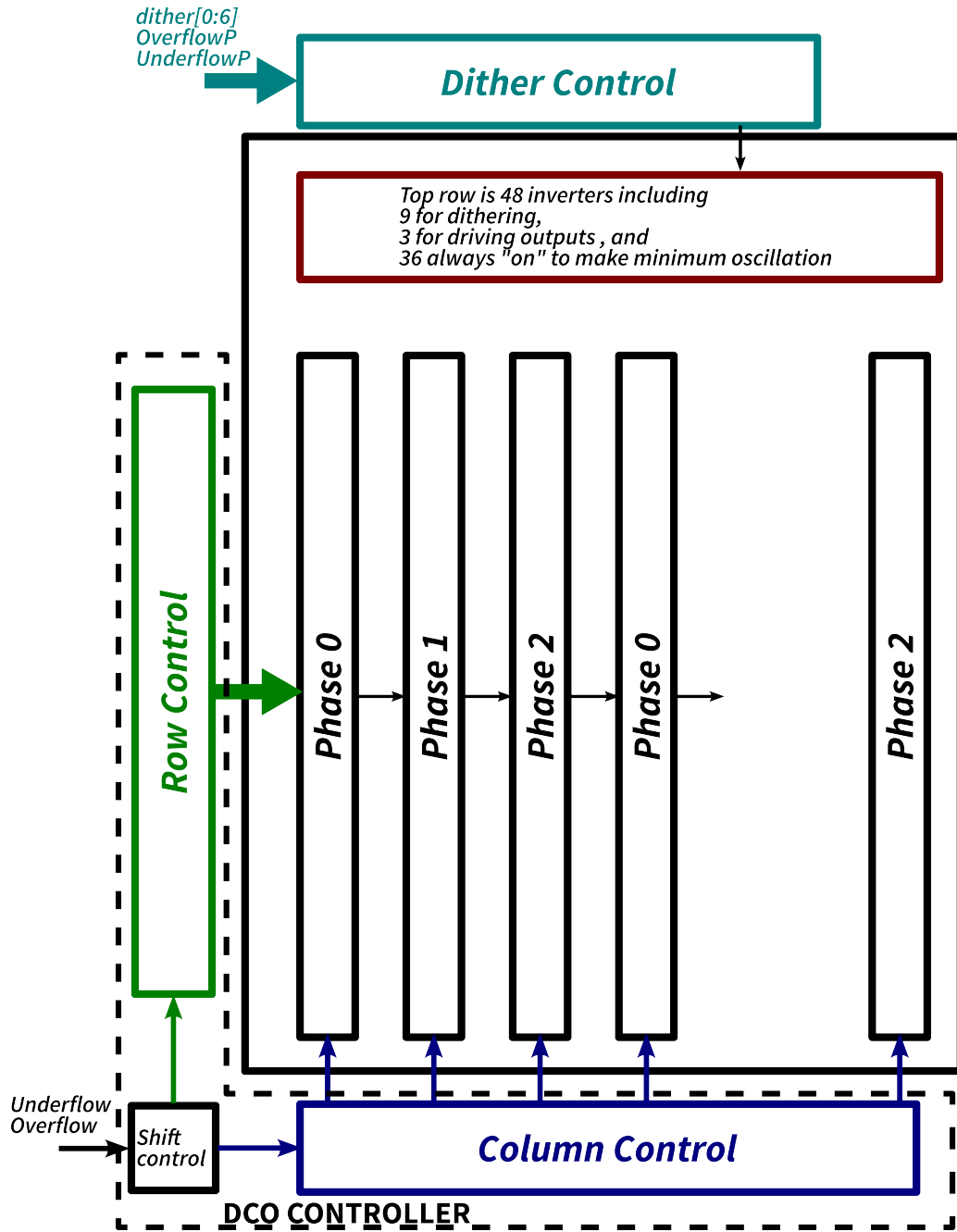


Figure 4.14: Block diagram of the DCO inverter array formation and its control inputs.

- Dithering Inverters (7): Controlled directly by the 7-bit output of the third-order MASH Sigma-Delta Modulator ($\Sigma\Delta M$). These inverters toggle rapidly to provide sub-inverter frequency resolution through time-averaging, enabling fractional-N synthesis.
- Latency Bypass Inverters: Controlled by the overflow/underflow signals from the DLF's proportional-differential path. These provide a fast, low-latency path for immediate phase corrections, bypassing the main PTC control logic to improve transient response and jitter.
- Output Drivers (3): One dedicated driver per phase ensures symmetrical loading on the oscillator core, preventing duty cycle distortion.

- Always-On Inverters (36): These ensure reliable oscillator startup under all conditions and define the minimum operating frequency.

The entire control logic was implemented in Verilog and functionally verified through simulation. Figure 4.15 shows a simulation waveform demonstrating the correct generation of control signals in response to test inputs. After verification, the control logic was synthesized into a standard-cell-based implementation for the 22nm FD-SOI process, as shown in fig. 4.16.

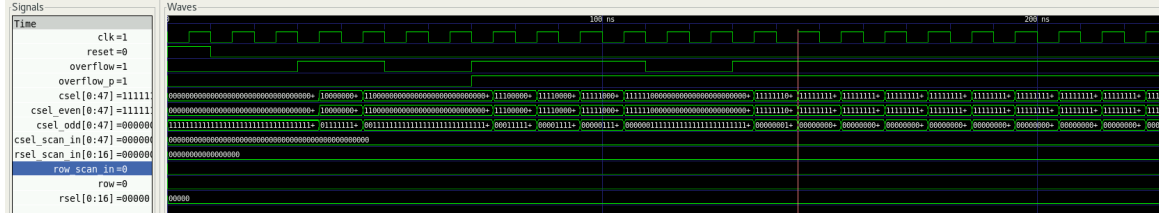


Figure 4.15: DCO controller's simulation results

4.3.4 Clock Divider

The final component in the ADPLL's feedback path is the clock divider. This fully digital block takes the high-frequency output of the DCO (f_{DCO}) and divides it down to match the input reference frequency (f_{REF}). This divided clock (f_{DIV}) is then sent to the PFD for phase and frequency comparison, closing the loop.

The divider was implemented in Verilog and its functionality was verified through simulation, as shown in fig. 4.17, which demonstrates the correct frequency division. The synthesized implementation of the clock divider is shown in fig. 4.18.

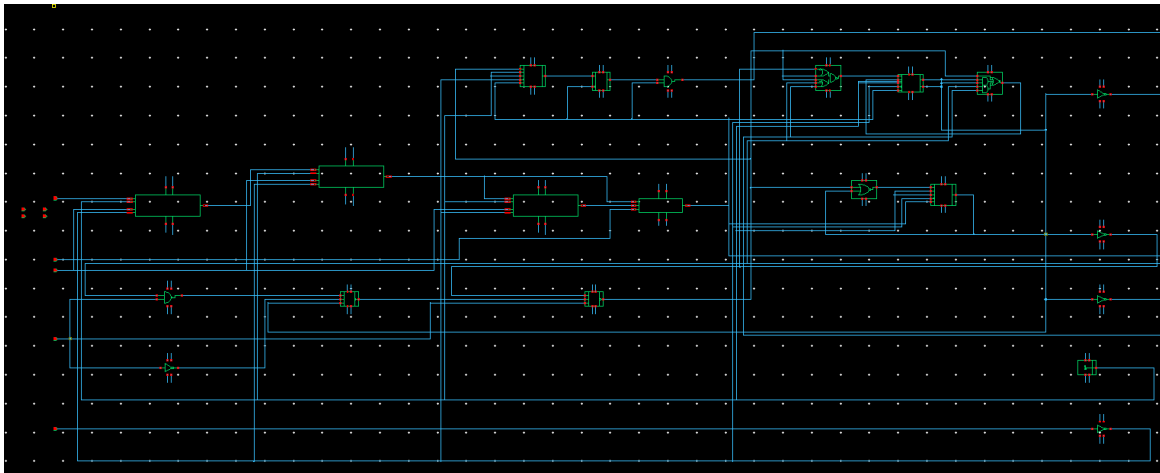


Figure 4.18: Synthesized Clock Divider

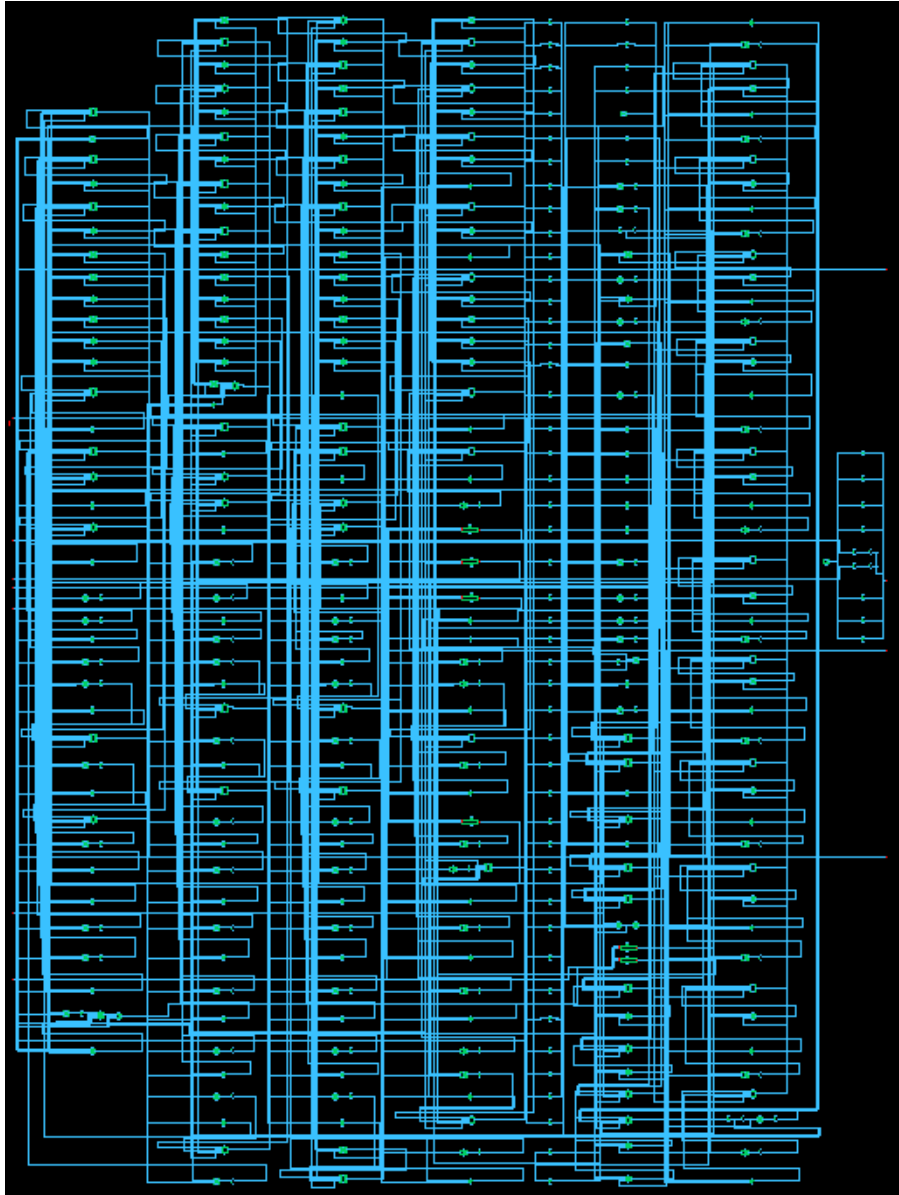


Figure 4.16: Synthesized DCO controller

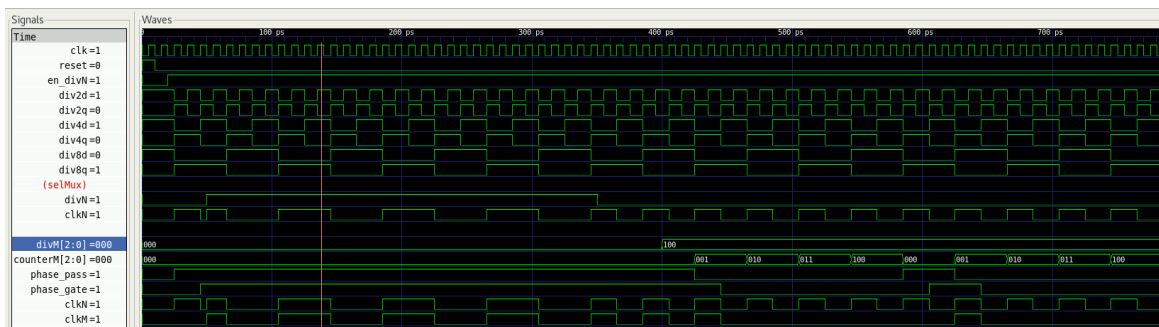


Figure 4.17: Clock Divider simulations

4.3.5 DCO Simulation and Verification

The functionality of the DCO and its various control mechanisms was verified through transient simulation. Figure 4.19 shows the DCO's operation, illustrating the impact of the different digital control inputs on the output frequency.

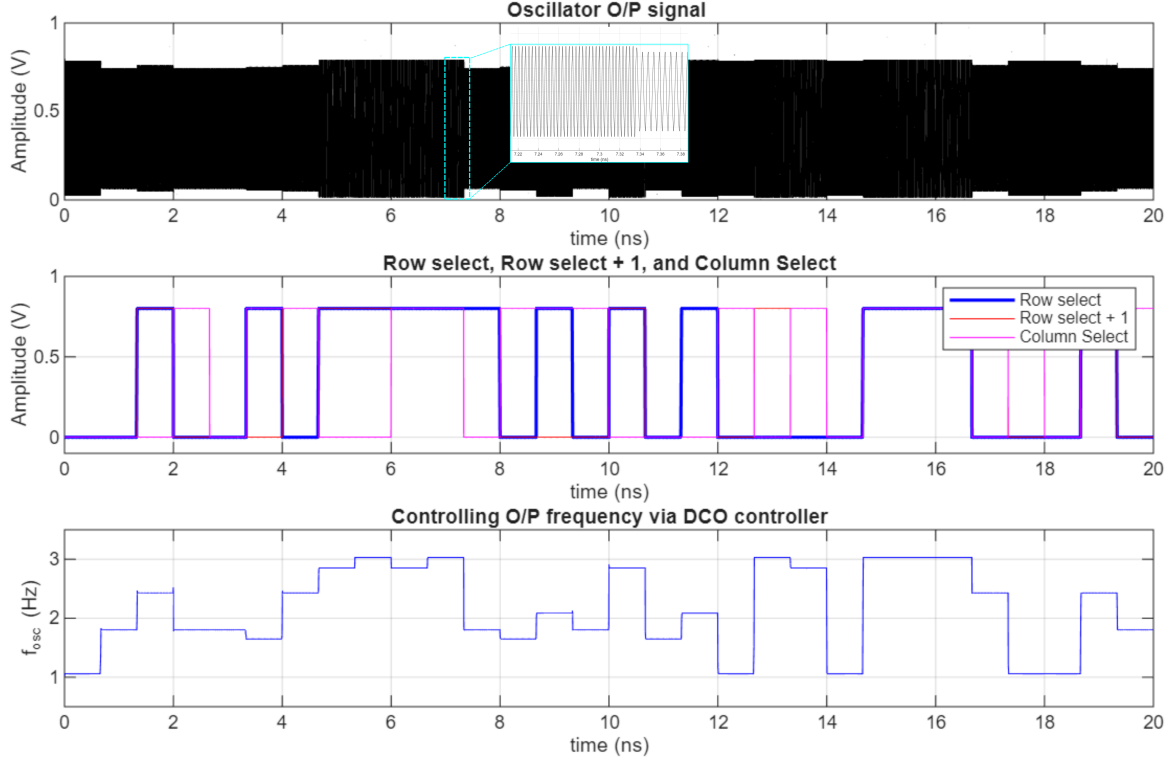


Figure 4.19: DCO simulation showing the effect of different control configurations on the output oscillation frequency.

The DCO achieves a wide total frequency tuning range (approximately 16:1). However, due to the uniform sizing of the inverters, the frequency step size is non-uniform, being relatively coarse at low frequencies and fine at high frequencies. This limits the effective dynamic range for optimal PLL performance (low jitter and spurs) to approximately 4:1 [3], typically within the upper portion of the total range. This range is sufficient to cover expected PVT variations without requiring complex band-switching mechanisms.

In summary, the DCO architecture represents an optimized design employing fully digital control techniques to achieve a wide tuning range, fine frequency resolution via dithering, glitch-free transitions via PTC, and robustness, all while leveraging the scalability and predictability of digital logic in the 22 nm FD-SOI process.

4.4 Closed-Loop System Verification

The performance of the complete ADPLL is verified through closed-loop simulations. Key metrics, including phase noise, jitter, loop stability, and frequency response, are characterized to confirm the design meets the target specifications.

The simulated output phase noise spectrum, measured with a 5.2 GHz output frequency, is presented in fig. 4.20. The plot demonstrates the spectral purity of the generated clock. The phase noise achieves -89 dBc/Hz at a 1 MHz offset, indicating strong suppression of noise within the loop bandwidth and low intrinsic noise from the DCO at higher frequencies. Integrating this phase noise spectrum from 1 kHz to 100 MHz yields an exceptionally low RMS jitter of 0.1 ps, confirming the design's suitability for high-performance clocking applications.

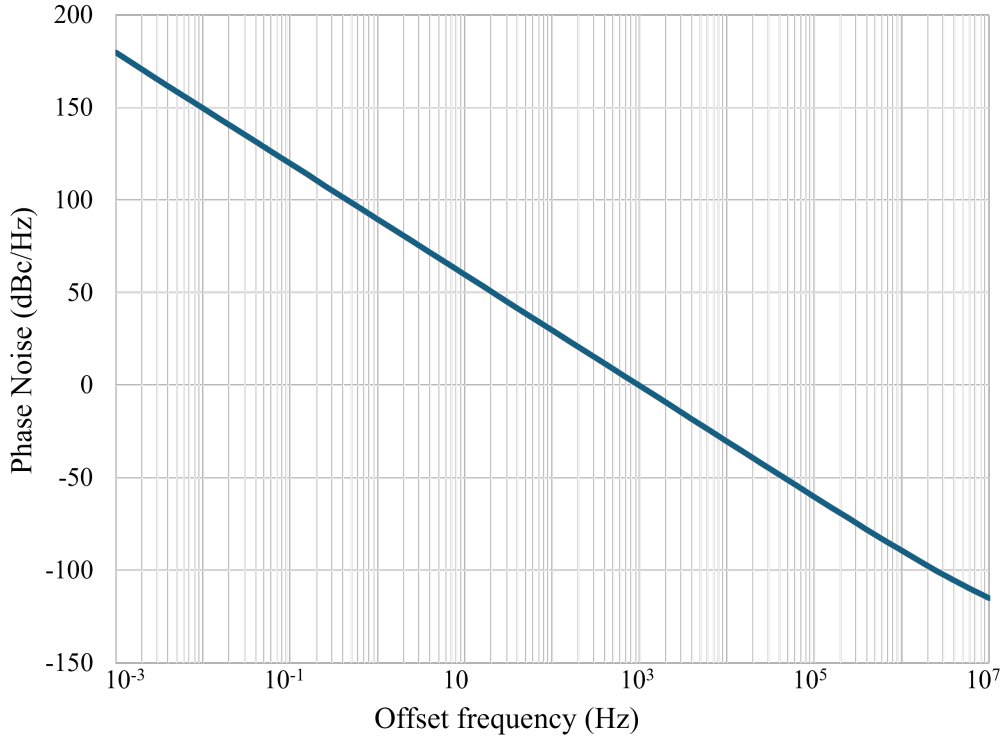


Figure 4.20: Simulated output phase noise spectrum at 5.2 GHz.

Further analysis characterizes the loop dynamics. Figure 4.21 shows the simulated loop gain frequency response, which is used to assess the stability of the ADPLL. The plot confirms adequate phase margin of 60° at $f_u = 1$ MHz, ensuring stable locking behavior. The closed-loop phase transfer function ($\Phi_{out}(f)/\Phi_{in}(f)$), shown in fig. 4.22, illustrates how reference phase noise is filtered by the loop up to $f_{-3db} = 1.36$ MHz. The low-pass characteristic indicates effective suppression of high-frequency reference noise, while the bandwidth reflects the loop's

tracking capability.

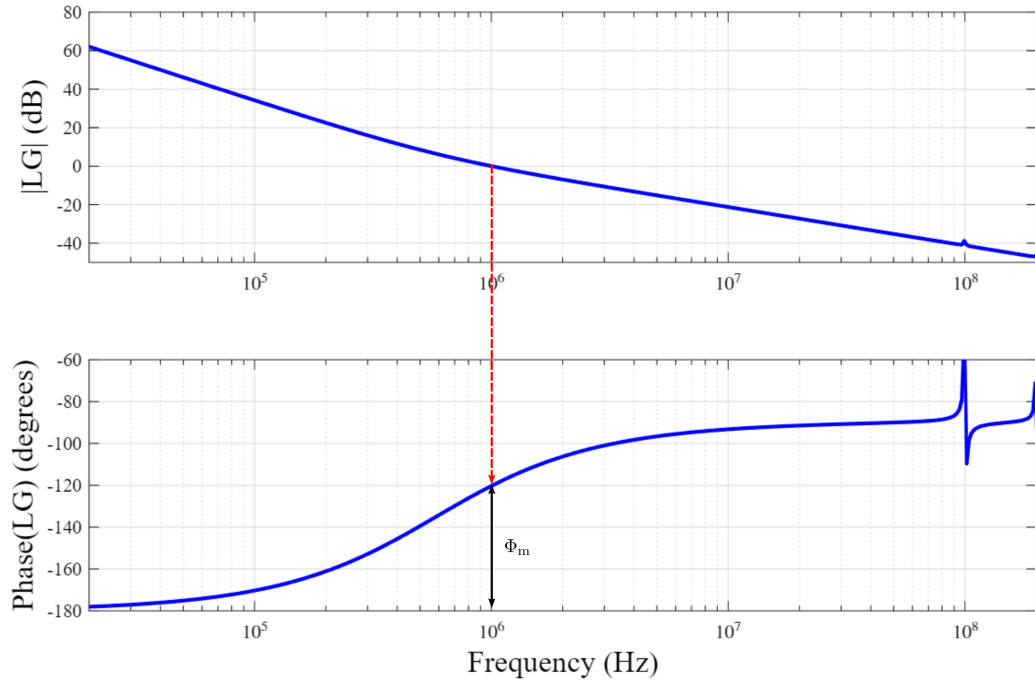


Figure 4.21: Loop Gain frequency simulation response

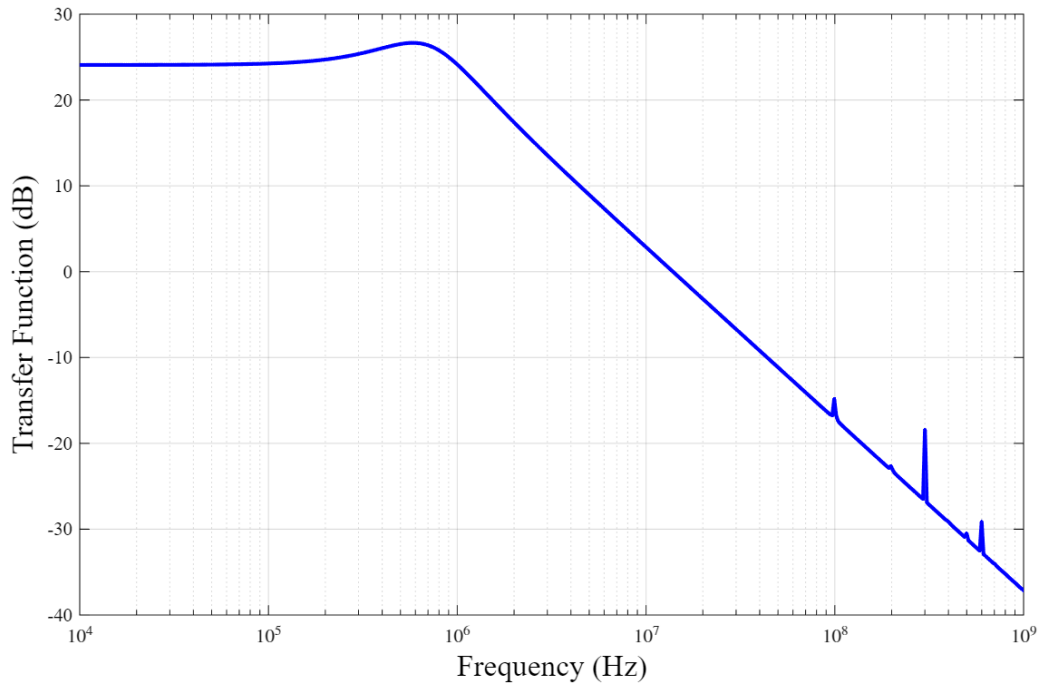


Figure 4.22: Phase transfer function = $\frac{\Phi_{out}(f)}{\Phi_{in}(f)}$

4.5 Performance Summary

The performance of the designed ADPLL, characterized through extensive simulations, meets or exceeds the target specifications. Table 4.2 provides a detailed summary of the key per-

Table 4.2: Performance Summary and Comparison with State-of-the-art Ring PLLs

	This Work	F. Ahmad ISSCC'16 [46]	M. Lee ISSCC'18 [47]	A.V. Rylyakov ISSCC'07 [3]	B. Xiang [48]
Type	Digital	Digital	Digital	Digital	Analog
Technology (nm)	22	16	28	65	22
Core Area (mm ²)	0.002	0.03	0.0043	150	0.015
Supply (V)	0.5-0.9	NA	0.3-1.2	0.5-1.3	0.5-0.9
RMS Integrated Jitter (ps)	0.1	NA	2.13	0.7	2.3
Power (mW)	3.4@0.9V	7.1	6.95	17.2@0.9V	0.682
Output Freq Range (GHz)	0.083-5.2	2.056@1V	2.056@1V	0.5-8	3.2@0.8V

formance metrics and compares this work against several state-of-the-art ADPLLs, including ring-oscillator-based designs implemented in similar technology nodes.

The designed ADPLL achieves an exceptionally wide frequency tuning range, spanning from 83 MHz to 5.2 GHz, as illustrated across different supply voltages in fig. 4.23. It operates reliably across a wide supply voltage range from 0.5 V to 0.9 V, demonstrating significant power efficiency. The power consumption is only 0.23 mW at 0.5 V and 3.4 mW at 0.9 V while operating at the maximum frequency of 5.2 GHz.

The phase noise performance is highly competitive, reaching -89 dBc/Hz at a 1 MHz offset

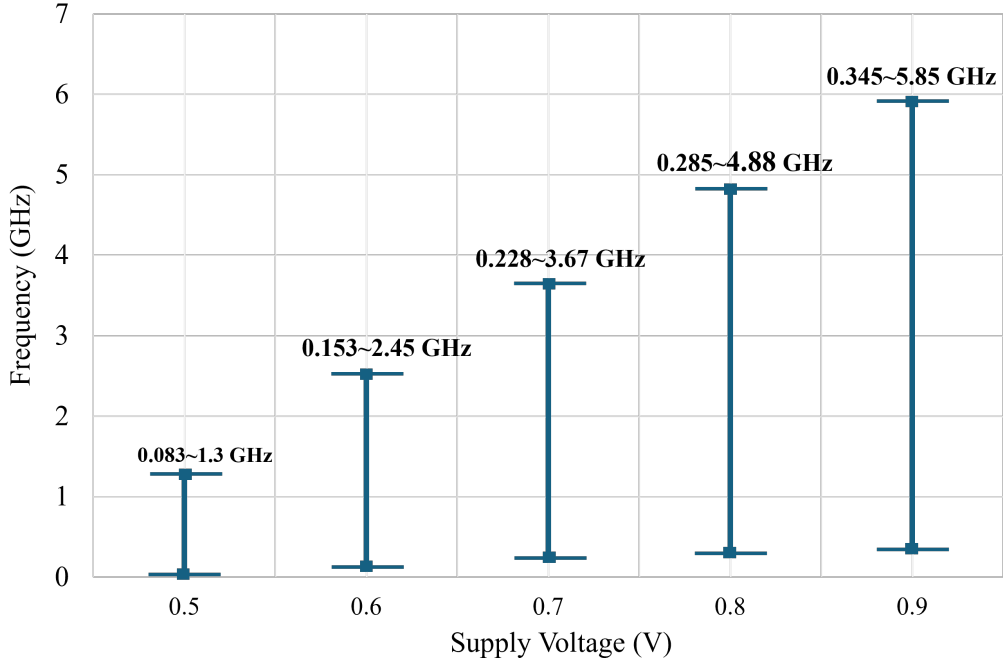


Figure 4.23: operating frequency ranges under different supply voltages

from a 5.2 GHz carrier. This translates to an integrated RMS jitter of only 0.1 ps (1 kHz - 100 MHz). Compared to the referenced works in table 4.2, this design achieves superior jitter performance and a very wide tuning range while maintaining excellent power efficiency. Furthermore, the implementation occupies a very compact core area of only 0.002 mm² (71 μ m $\times$ 32 μ m), highlighting the benefits of the all-digital approach and the 22 nm FD-SOI technol-

ogy for area-constrained SoC integration. The physical layout of the ADPLL core is shown in fig. 4.24.

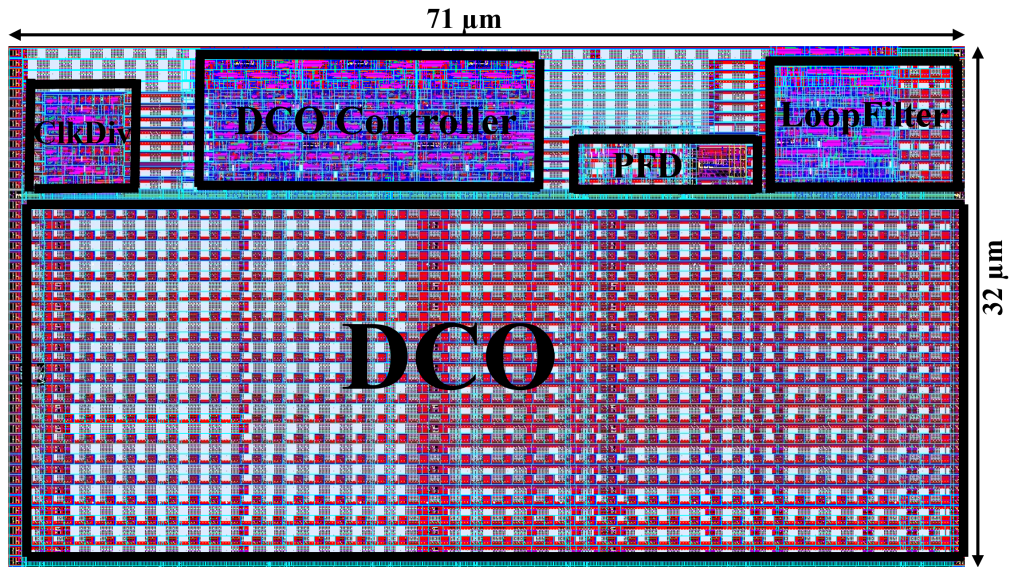


Figure 4.24: Layout of the designed ADPLL core in 22nm FD-SOI.

Chapter 5

Conclusion and Future Directions

5.1 Conclusion

This thesis addressed the growing challenges of designing high-performance analog and mixed-signal circuits in advanced CMOS technologies, specifically focusing on the 22nm FD-SOI process. Traditional design approaches face limitations due to reduced supply voltages, complex device physics, and the inaccuracy of simplified transistor models outside the strong inversion region. To overcome these hurdles, a systematic design methodology based on the transconductance-to-current ratio (g_m/I_D) and pre-computed lookup tables (LUTs) was developed and validated. This framework provides an accurate and efficient means to navigate complex design trade-offs across all transistor operating regions, bridging the gap between analytical design and detailed circuit simulation.

A comprehensive characterization of the 22nm FD-SOI technology was performed, generating detailed design charts for key figures of merit, including intrinsic gain, transit frequency, noise parameters, and parasitic capacitances. The unique characteristics of FD-SOI, particularly the effectiveness of back-gate biasing and its consistent behavior within the normalized g_m/I_D framework, were highlighted. The methodology was also extended to incorporate an efficient estimation technique for local mismatch, reducing reliance on time-consuming Monte Carlo simulations.

The practical utility of this systematic methodology was demonstrated through the design of two distinct, high-performance circuits:

1. A low-noise, high-bandwidth folded cascode OTA was designed for a Charge-Integrating

Amplifier front-end suitable for nanopore sensing. System-level analysis addressed the critical stability issues associated with large input capacitances, informing the OTA specifications. The OTA was systematically sized using the g_m/I_D framework, and simulations confirmed that it met demanding targets for gain, bandwidth, noise, and power consumption.

2. A low-jitter, wide-range All-Digital PLL (ADPLL) was designed, employing state-of-the-art digital techniques such as a bang-bang phase-frequency detector with metastability filtering and a digitally controlled ring oscillator with pseudo-thermometer coding and sigma-delta dithering. Simulation results demonstrated excellent performance, including a wide tuning range (83 MHz - 5.2 GHz), ultra-low jitter (0.1 ps_{rms}), high power efficiency ($3.4 \text{ mW @ } 5.2 \text{ GHz}$), and a compact area (0.0023 mm^2).

In conclusion, this work successfully established and applied a robust g_m/I_D -based design methodology for the 22 nm FD-SOI process. The close agreement between the systematic design procedure and final simulation results for both the analog front-end and the digitally-intensive ADPLL validates the effectiveness of this approach for tackling complex mixed-signal design challenges in advanced technology nodes.

5.2 Statement on Contributions

The key contributions of this thesis are:

- Development and comprehensive validation of a g_m/I_D -based design framework for the 22 nm FD-SOI technology, including detailed characterization charts and analysis of back-gate biasing effects.
- Systematic application of the g_m/I_D methodology to the design and optimization of a high-performance folded cascode OTA, meeting stringent specifications for low-noise, high-bandwidth TIA applications.
- Demonstration of system-level analysis for TIA stability and bandwidth, bridging the gap between OTA performance and closed-loop system requirements.
- Design and simulation verification of a state-of-the-art, low-jitter, wide-range ADPLL in 22 nm FD-SOI, achieving highly competitive performance metrics.

5.3 Future Work

While this thesis successfully demonstrated the design methodology and circuit implementations, several avenues for future research remain:

- **Fabrication and Experimental Verification:** The most critical next step is the physical fabrication of the designed OTA/TIA and ADPLL circuits in the 22 nm FD-SOI process. Experimental measurements are required to definitively validate the simulated performance and assess the accuracy of the g_m/I_D methodology against silicon results.
- **Methodology Enhancements:** The g_m/I_D framework could be further extended to incorporate more sophisticated modeling, such as distortion analysis or more detailed mismatch models. Automating the generation of behavioral models directly from the gm/id design point would also enhance the workflow.
- **Circuit Optimization:** Further optimization of the designed circuits could be explored. The CTIA noise performance could potentially be improved through techniques like chopper stabilization. The ADPLL could be further optimized using dynamic back-gate biasing.
- **System Integration:** The designed blocks could be integrated into larger systems, such as a multi-channel nanopore readout array or a complete SoC clocked by the ADPLL.
- **Development Towards Analog Generators:** The script-based sizing methodology presented in this thesis forms a foundation that could be extended towards building automated analog circuit generators. Following the direction of frameworks like the Berkeley Analog Generator (BAG) and OpenFASOC, the Python scripts could be enhanced to automate not only circuit sizing and simulation but potentially also layout generation and verification. This would enable the rapid creation of fully verified analog IP blocks based on high-level specifications, significantly improving design productivity, reuse, and reproducibility, aligning with trends towards open-source hardware design.

Bibliography

- [1] Paul G. A. Jespers and Boris Murmann. Systematic Design of Analog CMOS Circuits: Using Pre-Computed Lookup Tables. Cambridge University Press, 2017. doi: 10.1017/9781108125840.
- [2] Boris Murmann. Practical aspects of script-based analog design using precomputed lookup tables. In 2024 IEEE International Symposium on Circuits and Systems (ISCAS), pages 1–5, 2024. doi: 10.1109/ISCAS58744.2024.10558027.
- [3] JosÉ A. Tierno, Alexander V. Rylyakov, and Daniel J. Friedman. A wide power supply range, wide tuning range, all static cmos all digital pll in 65 nm soi. IEEE Journal of Solid-State Circuits, 43(1):42–51, 2008. doi: 10.1109/JSSC.2007.910966.
- [4] Christian Enz, Hung-Chi Han, and Simon Berner. Design of low-power analog circuits in advanced technology nodes using the g_m/i_D approach. In 2023 IEEE International Symposium on Circuits and Systems (ISCAS), pages 1–5, 2023. doi: 10.1109/ISCAS46773.2023.10181364.
- [5] Jerzy K. Kulski. Next-generation sequencing — an overview of the history, tools, and “omic” applications. Open Access pear reviewed chapter, 8(2):278–292, Jan 2016. ISSN 1932-4545. doi: 10.1109/TBCAS.2013.2262998.
- [6] Sepideh Asgari, Amirhossein Mohammadpanah, Ebrahim Ghafar-Zadeh, and Sebastian Magierowski. Cmos nanopore-based dna sequencing systems: Recent advancements and future prospects: A review. IEEE Sensors Journal, 25(9):14539–14556, 2025. doi: 10.1109/JSEN.2025.3548651.
- [7] Yunus Dawji, Mehdi Habibi, Ebrahim Ghafar-Zadeh, and Sebastian Magierowski. A scal-

- able discrete-time integrated cmos readout array for nanopore based dna sequencing. IEEE Access, 9:155543–155554, 2021. doi: 10.1109/ACCESS.2021.3129171.
- [8] Sepideh Asgari, Amirhossein Mohammadpanah, Ebrahim Ghafar-Zadeh, and Sebastian Magierowski. An all digital pll in 22-nm fd soi for hardware accelerated embedded systems. In 2024 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), pages 1–7, 2024. doi: 10.1109/CCECE59415.2024.10667180.
- [9] H. Agarwal, P. Kushwaha, A. Dasgupta, M. Y-Kao, T. Morshed, G. Workman, K. Shanbhag, X. Li, V. Vinothkumar, Y. S. Chauhan, S. Salahuddin, and C. Hu. Bsim-ing: Advanced model for fdsoi transistors with back channel inversion. In 2020 4th IEEE Electron Devices Technology Manufacturing Conference (EDTM), pages 1–4, 2020. doi: 10.1109/EDTM47692.2020.9117979.
- [10] Jonah Van Assche and Jens Anders. Low-noise current readout circuits: A tutorial overview. IEEE Solid-State Circuits Magazine, 17(3):48–60, 2025. doi: 10.1109/MSSC.2025.3579705.
- [11] Kareem H. Khattab, Muh-Dey Wei, and Renato Negra. Technology-independent local mismatch estimation using precomputed lookup tables. In 2025 21st International Conference on Synthesis, Modeling, Analysis and Simulation Methods, and Applications to Circuits Design (SMACD), pages 1–4, 2025. doi: 10.1109/SMACD65553.2025.11092134.
- [12] Aurojyoti Das, Qiuyang Lin, Yixiong Hu, Wim Sijbers, Eric Beamish, Juliette Gevers, Chris Van Hoof, Georges Gielen, and Nick Van Helleputte. A 376w per-channel, drift-tolerant translocation recording frontend with event detection for nanopore sensor arrays. In 2024 IEEE European Solid-State Electronics Research Conference (ESSERC), pages 420–423, 2024. doi: 10.1109/ESSERC62670.2024.10719545.
- [13] Hung-Chi Han, Antonio D’Amico, and Christian Enz. Comprehensive design-oriented fdsoi ekv model. In 2022 29th International Conference on Mixed Design of Integrated Circuits and System (MIXDES), pages 40–44, 2022. doi: 10.23919/MIXDES55591.2022.9838014.
- [14] J. K. Rosenstein, M. Wanunu, C. A. Merchant, M. Drndic, and K. L. Shepard. Integrated

- nanopore sensing platform with sub-microsecond temporal resolution. Nat. Methods, 9(5): 487–492, May 2012.
- [15] Siddharth Shekar, David J. Niedzwiecki, Chen-Chi Chien, Peijie Ong, Daniel A. Fleischer, Jianxun Lin, Jacob K. Rosenstein, Marija Drndić, and Kenneth L. Shepard. Measurement of dna translocation dynamics in a solid-state nanopore at 100 ns temporal resolution. Nano Letters, 16(7):4483–4489, 2016. doi: 10.1021/acs.nanolett.6b01661. URL <https://doi.org/10.1021/acs.nanolett.6b01661>. PMID: 27332998.
- [16] M. Crescentini, M. Bennati, M. Carminati, and M. Tartagni. Noise limits of cmos current interfaces for biosensors: A review. IEEE Transactions on Biomedical Circuits and Systems, 8(2):278–292, April 2014. ISSN 1932-4545. doi: 10.1109/TBCAS.2013.2262998.
- [17] B. Goldstein, D. Kim, J. Xu, T. K. Vanderlick, and E. Culurciello. Cmos low current measurement system for biomedical applications. IEEE Transactions on Biomedical Circuits and Systems, 6(2):111–119, April 2012. ISSN 1932-4545. doi: 10.1109/TBCAS.2011.2182512.
- [18] G. Ferrari, F. Gozzini, A. Molari, and M. Sampietro. Transimpedance amplifier for high sensitivity current measurements on nanodevices. IEEE J. Solid-State Circuits, 44(5): 1609–1616, May 2009.
- [19] S. Dai, R. T. Perera, Z. Yang, and J. K. Rosenstein. A 155-db dynamic range current measurement front end for electrochemical biosensing. IEEE Transactions on Biomedical Circuits and Systems, 10(5):935–944, Oct 2016. ISSN 1932-4545. doi: 10.1109/TBCAS.2016.2612581.
- [20] J.D.H. Alexander. Clock recovery from random binary signals. Electronics Letters, 11: 541–542, 1975. doi: 10.1049/el:19750415. URL <https://digital-library.theiet.org/doi/abs/10.1049/el%3A19750415>.
- [21] M. Verbeke, P. Rombouts, X. Yin, and G. Torfs. Inverse alexander phase detector. Electronics Letters, 52(23):1908–1910, 2016. doi: <https://doi.org/10.1049/el.2016.3368>. URL <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/el.2016.3368>.

- [22] Behzad Razavi. Designing BangBang PLLs for Clock and Data Recovery in Serial Data Transmission Systems, pages 34–45. 2003. doi: 10.1109/9780470545492.ch4.
- [23] J. Strzelecki and S. Ren. Near-zero dead zone phase frequency detector with wide input frequency difference. Electronics Letters, 51(14):1059–1061, 2015. doi: <https://doi.org/10.1049/el.2015.0653>. URL <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/el.2015.0653>.
- [24] Alain J. Martin. Compiling communicating processes into delay-insensitive VLSI circuits. Distributed Computing, 1(4):226–234, December 1986. ISSN 1432-0452. doi: 10.1007/BF01660034. URL <https://doi.org/10.1007/BF01660034>.
- [25] David Kinniment. Synchronization and arbitration in digital systems. pages 1–263, 03 2008. doi: 10.1002/9780470517147.
- [26] Yang Zhang, Leandro S. Heck, Matheus T. Moreira, David Zar, Melvin Breuer, Ney L.V. Calazans, and Peter A. Beerel. Design and analysis of testable mutual exclusion elements. In 2015 21st IEEE International Symposium on Asynchronous Circuits and Systems, pages 124–131, 2015. doi: 10.1109/ASYNC.2015.28.
- [27] T.J. Chaney and C.E. Molnar. Anomalous behavior of synchronizer and arbiter circuits. IEEE Transactions on Computers, C-22(4):421–422, 1973. doi: 10.1109/T-C.1973.223730.
- [28] Ran Ginosar. Metastability and synchronizers: A tutorial. IEEE Design Test of Computers, 28(5):23–35, 2011. doi: 10.1109/MDT.2011.113.
- [29] Jri Lee, K.S. Kundert, and B. Razavi. Analysis and modeling of bang-bang clock and data recovery circuits. IEEE Journal of Solid-State Circuits, 39(9):1571–1580, 2004. doi: 10.1109/JSSC.2004.831600.
- [30] David E. Muller and University of Illinois at Urbana-Champaign. Theory of asynchronous circuits. Number 1. 15 in Report ;no. 66. University of Illinois, Graduate College, Digital Computer Laboratory, Urbana, Illinois, 1955. URL <https://catalog.hathitrust.org/Record/100701423>.

- [31] M. Shams, J.C. Ebergen, and M.I. Elmasry. Modeling and comparing cmos implementations of the c-element. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 6(4):563–567, 1998. doi: 10.1109/92.736128.
- [32] I. E. Sutherland. Micropipelines. Commun. ACM, 32(6):720–738, June 1989. ISSN 0001-0782. doi: 10.1145/63526.63532. URL <https://doi.org/10.1145/63526.63532>.
- [33] F.U. Rosenberger, C.E. Molnar, T.J. Chaney, and T.-P. Fang. Q-modules: internally clocked delay-insensitive modules. IEEE Transactions on Computers, 37(9):1005–1018, 1988. doi: 10.1109/12.2252.
- [34] In-Chul Hwang, Sang-Hun Song, and Soo-Won Kim. A digitally controlled phase-locked loop with a digital phase-frequency detector for fast acquisition. IEEE Journal of Solid-State Circuits, 36(10):1574–1581, 2001. doi: 10.1109/4.953487.
- [35] A.J. Martin and M. Nystrom. Asynchronous techniques for system-on-chip design. Proceedings of the IEEE, 94(6):1089–1120, 2006. doi: 10.1109/JPROC.2006.875789.
- [36] J. Crowe and M.A. Johnson. Phase-Locked Loop Methods and PID Control, pages 259–296. Springer London, London, 2005. ISBN 978-1-84628-148-8. doi: 10.1007/1-84628-148-2_7. URL https://doi.org/10.1007/1-84628-148-2_7.
- [37] Loop Filters, chapter 11, pages 267–270. John Wiley Sons, Ltd, 2005. ISBN 9780471732693. doi: <https://doi.org/10.1002/0471732699.ch11>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471732699.ch11>.
- [38] B. Miller and R.J. Conley. A multiple modulator fractional divider. IEEE Transactions on Instrumentation and Measurement, 40(3):578–583, 1991. doi: 10.1109/19.87022.
- [39] T.A.D. Riley, M.A. Copeland, and T.A. Kwasniewski. Delta-sigma modulation in fractional-n frequency synthesis. IEEE Journal of Solid-State Circuits, 28(5):553–559, 1993. doi: 10.1109/4.229400.
- [40] Shanthi Pavan, Richard Schreier, and Gábor C. Temes. Understanding Delta-Sigma Data Converters. IEEE Press Series on Microelectronic Systems. John Wiley & Sons / IEEE Press, Hoboken, NJ, USA, 2nd edition, 2017. ISBN 9781119258308.

- [41] Robert Bogdan Staszewski. State-of-the-art and future directions of high-performance all-digital frequency synthesis in nanometer cmos. In Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pages 229–232, 2010. doi: 10.1109/ISCAS.2010.5537937.
- [42] Robert Bogdan Staszewski. Digital Deep-Submicron CMOS Frequency Synthesis for RF Wireless Applications. PhD thesis, 08 2002.
- [43] Ping Xiao, William W. Berez, Weiying Ding, and Mohsen Moussavi. Thermometer-code-to-binary encoders, March 2010. URL <https://patents.google.com/patent/US7675440B1/en>.
- [44] Arsalan Shahid, Saad Arif, Muhammad Qadri, and Saba Munawar. Power Optimization Using Clock Gating and Power Gating: A Review, pages 1–20. 07 2016. ISBN ISBN13: 9781522502876 — ISBN10: 1522502874. doi: 10.4018/978-1-5225-0287-6.ch001.
- [45] Juan P. Oliver, Juan Curto, Diego Bouvier, Manuela Ramos, and Eduardo Boemo. Clock gating and clock enable for fpga power reduction. In 2012 VIII Southern Conference on Programmable Logic, pages 1–5, 2012. doi: 10.1109/SPL.2012.6211782.
- [46] Fazil Ahmad, Greg Unruh, Amrutha Iyer, Pin-En Su, Sherif Abdalla, Bo Shen, Mark Chambers, and Ichiro Fujimori. 19.1 a 0.5-to-9.5ghz 1.2ps-lock-time fractional-n dpll with $\pm 1.25\%$ In 2016 IEEE International Solid-State Circuits Conference (ISSCC), pages 324–325, 2016. doi: 10.1109/ISSCC.2016.7418038.
- [47] Minseob Lee, Shinwoong Kim, Hwasuk Cho, Jahyun Koo, Kwang-Hee Choi, Jin-Hyeok Choi, Byungsub Kim, Hong-June Park, and Jae-Yoon Sim. A 0.3-to-1.2 v frequency-scalable fractional-n adpll with a speculative dual-referenced interpolating tdc. In 2018 IEEE International Solid-State Circuits Conference-(ISSCC), pages 122–124. IEEE, 2018.
- [48] Bo Xiang, Yongping Fan, James Ayers, James Shen, and Dan Zhang. A 0.5v-to-0.9v 0.2ghz-to-5ghz ultra-low-power digitally-assisted analog ring pll with less than 200ns lock time in 22nm finfet cmos technology. In 2020 IEEE Custom Integrated Circuits Conference (CICC), pages 1–4, 2020. doi: 10.1109/CICC48029.2020.9075897.

Appendix A

$\frac{g_m}{I_D}$ extraction and mapping MATLAB code

```
%% Sample MATLAB code
% Configuration for techsweep_spectre_run.m
% Modified By Amirhossein Mohammadpanah, York University_EMIL
    Group
% Tested with ICADVM20.10.320
% July 27, 2024

function c = techsweep_config_bsimimg_22_spectre_modified

% Models and file paths
c.modelfile = '/local/data0/PDK/22FDX-EXT/V1.0_4.1/Models/
    Spectre/models/design_wrapper.lib.scs' section=tt_pre';
c.modelinfo = 'GF 22nm, BSIMIMG';
c.corner = 'NOM';
c.temp = 300;
c.modeln = 'eguslvtnfet';
c.modelp = 'eguslvtpfet';
c.savefilen = '22nch_eguslvt_b';
```

```

c.savefilep = '22pch_eguslvt_b';

c.simcmd = 'set path=($path /cad/cadence/ICADVM20.10.320_lnx86/
tools/bin);/CMC/tools/cadence/SPECTRE21.10.460_lnx86/tools.
lnx86/bin/spectre +log techsweep.out ++aps +lqtimeout 900 -
maxw 5 -maxn 5 techsweep.scs';
c.outfile = 'techsweep.raw';
c.sweep = 'sweepvds_sweepvgs-sweep';
c.sweep_noise = 'sweepvds_noise_sweepvgs_noise-sweep';

% Sweep parameters
c.VGS_step = 20e-3;
c.VDS_step = 20e-3;
c.VSB_step = 0.1;
c.VGS_max = 0.8;
c.VDS_max = 0.8;
c.VSB_max = 2;
c.VGS = 0:c.VGS_step:c.VGS_max;
c.VDS = 0:c.VDS_step:c.VDS_max;
c.VSB = -c.VSB_max:c.VSB_step:c.VSB_max;
c.LENGTH = [(0.07:0.004:0.036) (0.04:0.02:0.2) (0.25:0.05:1)
(1.5:0.5:10)];
%c.LENGTH = [ (0.07:0.02:0.2) (0.25:0.05:1) (1.5:0.5:6)];
c.NFIN = 2;
%c.NF = 1;
c.WIDTH = 1;

% Variable mapping
c.outvars = {'ID','VT','IGD','IGS','GM','GMB','GDS','
CGG','CGS','CSG','CGD','CDG','CGB','CDD','CSS','CBB','CBS','

```



```

CDS'};

c.n{1}= {'mn:ids','A', [1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
]};

c.n{2}= {'mn:vth','V', [0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0
]};

c.n{3}= {'mn:ig','A', [0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0
]};

c.n{4}= {'mn:is','A', [0 0 0 -1 0 0 0 0
0 0 0 0 0 0 0 0
]};

c.n{5}= {'mn:gm','A_V', [0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0
]};

c.n{6}= {'mn:gmb's','A_V', [0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0
]};

c.n{7}= {'mn:gds','A_V', [0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0
]};

c.n{8}= {'mn:cgg','F', [0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
]};

c.n{9}= {'mn:cgs','F', [0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0
]};

c.n{10}={ 'mn:cgd','F', [0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0

```

```

    ]});
c.n{11}={ 'mn:cgb', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    1    0    0    0    0    0
    ]});
c.n{12}={ 'mn:cdd', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    0    1    0    0    0    0
    ]});
c.n{13}={ 'mn:cdg', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    1    0    0    0    0    0    0
    ]});
c.n{14}={ 'mn:css', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    0    0    -1    0    0    0
    ]});
c.n{15}={ 'mn:csG', 'F',    [0    0    0    0    0    0    0
    0    0    1    0    0    0    0    0    0    0    0
    ]});
c.n{16}={ 'mn:cbb', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    1    0    0
    ]});
c.n{17}={ 'mn:cbS', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    1    0
    ]});
c.n{18}={ 'mn:cdS', 'F',    [0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    0    -1
    ]});
%
%               {'ID','VT','IGD','IGS','GM','GMB','GDS','
CGG','CGS','CSG','CGD','CDG','CGB','CDD','CSS','CBB','CSS','
CDS'};
c.p{1}= {'mp:ids','A',    [-1    0    0    0    0    0    0

```

```

0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{2}= {'mp:vth','V',    [0      -1      0      0      0      0      0
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{3}= {'mp:ig','A',    [0      0      -1      0      0      0      0
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{4}= {'mp:is','A',    [0      0      0      1      0      0      0
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{5}= {'mp:gm','A_V',  [0      0      0      0      1      0      0
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{6}= {'mp:gmb','A_V', [0      0      0      0      0      1      0
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{7}= {'mp:gds','A_V', [0      0      0      0      0      0      1
0      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{8}= {'mp:cgg','F',    [0      0      0      0      0      0      0
1      0      0      0      0      0      0      0      0      0      0
    ]];
c.p{9}= {'mp:cgs','F',    [0      0      0      0      0      0      0
0      1      0      0      0      0      0      0      0      0      0
    ]];
c.p{10}={ 'mp:cgd','F',    [0      0      0      0      0      0      0
0      0      0      1      0      0      0      0      0      0      0
    ]];
c.p{11}={ 'mp:cgb','F',    [0      0      0      0      0      0      0

```

```

    0    0    0    0    0    1    0    0    0    0    0
  ]});
c.p{12}={ 'mp:cdd', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0    0    0   -1    0    0    0    0
  ]});
c.p{13}={ 'mp:cdg', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0   -1    0    0    0    0    0    0
  ]});
c.p{14}={ 'mp:css', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    1    0    0    0
  ]});
c.p{15}={ 'mp:csb', 'F', [0    0    0    0    0    0    0    0
    0    0   -1    0    0    0    0    0    0    0    0
  ]});
c.p{16}={ 'mp:cbb', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    1    0    0
  ]});
c.p{17}={ 'mp:csb', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    1    0
  ]});
c.p{18}={ 'mn:cds', 'F', [0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    0   -1
  ]});

%
c.outvars_noise = {'STH','SFL'};
c.n_noise{1}= {'mn:therm_sid', ''};
c.n_noise{2}= {'mn:flicker', ''};

%
c.p_noise{1}= {'mp:therm_sid', ''};
c.p_noise{2}= {'mp:flicker', ''};

```

```

% Simulation netlist
netlist = sprintf([...
'//techsweep.scs \n'...
'include %s\n'...
'include "/local/data0/Students/AM/temp/gmid22_scripts/Book-on-gm
-ID-design/starter_kit/techsweep_params.scs" \n'...
'save mn \n'...
'save mp \n'...
'parameters gs=0 ds=0 l=length*1e-6 w=%de-6 wireopt=19 \n'...
'vnoi      (vx  0)          vsource dc=0  \n'...
'vdsn      (vdn vx)         vsource dc=ds  \n'...
'vgzn      (vgn 0)          vsource dc=gs  \n'...
'vbsn      (vbn 0)          vsource dc=-sb \n'...
'vdsp      (vdp vx)         vsource dc=-ds \n'...
'vgzp      (vgp 0)          vsource dc=-gs \n'...
'vbsp      (vbp 0)          vsource dc=sb  \n'...
'\n'...
'//NOTE: YOUR MODELS SHOULD BE SET UP SUCH THAT THE STRESS PARAMS
      (SA, SB, etc.) ARE AUTOMATICALLY COMPUTED\n'...
'mn      (vdn vgn 0 vbn) %s w=w l=l nf=2 par=(1) par_nf=(1) * (1)
      m=1 plorient=0 acv_opt=-1 \\\ \n'...
'      ptwell=0 ngcon=1 nscon=1 ndcon=1 p_la=0 p_wa=0 ulp=0
      tie_orient=0 swrfmhc_local=0 analog=0 \n'...
'mp      (vdp vgp 0 vbp) %s w=w l=l nf=2 par=(1) par_nf=(1) * (1)
      m=1 plorient=0 acv_opt=-1 \\\ \n'...
'      ptwell=0 ngcon=1 nscon=1 ndcon=1 p_la=0 p_wa=0 ulp=0
      tie_orient=0 swrfmhc_local=0 analog=0 \n'...

```

```

        '//mn (vdn vgn 0 vbn) %s l=length*1e-6 nfin=%d nf=%d\n'
        ...
        '//mp (vdp vgp 0 vbp) %s l=length*1e-6 nfin=%d nf=%d\n'
        ...
'\n'...
'options1 options psfversion="1.4.0" gmin=1e-13 reltol=1e-4
    vabstol=1e-6 \\\ \n'...
'        iabstol=1e-12 temp=%d tnom=27 scalem=1.0 scale=1.0
    rforce=1 \\\ \n'...
'        vthmod=vthcc ivthn=300e-9 ivthp=70e-9 ivthw=0 ivthl=0
    maxnotes=5 \\\ \n'...
'        maxwarns=5 digits=5 cols=80 pivrel=1e-3 \\\ \n'...
'        vdsatmod=slow_gds subcktoppoint=yes subcktcap=1 rawfmt=
    psfbin rawfile="./techsweep.raw" \n'...
'sweepvds sweep param=ds start=0 stop=%d step=%d { \n'...
'    sweepvgs dc param=gs start=0 stop=%d step=%d \n'...
'}\n'...
'sweepvds_noise sweep param=ds start=0 stop=%d step=%d { \n'...
'    sweepvgs_noise noise freq=1 oprobe=vnoi param=gs start=0 stop
    =%d step=%d \n'...
'}\n'...
'save mn:ids mn:gds mn:gm mn:id mn:vth mn:vgs mp:ids mp:gds mp:gm
    mp:id mp:vth mp:vgs \n'...
'saveOptions options save=allpub \n'...

], c.modelfile,c.WIDTH,...
c.modeln,...
c.modelp,...
c.modeln,...
c.NFIN,...

```

```

c.NFIN,...
c.modelp,...
c.NFIN,...
c.NFIN,...
c.temp-273, ...
c.VDS_max, c.VDS_step, ...
c.VGS_max, c.VGS_step, ...
c.VDS_max, c.VDS_step, ...
c.VGS_max, c.VGS_step);

% Write netlist
fid = fopen('techsweep.scs', 'w');
fprintf(fid, netlist);
fclose(fid);

return

% Matlab script for technology characterization
% Adopted from Boris Murmann, Stanford University
% Tested with SPECTRE21.10.460
% September 12, 2017

clearvars;
close all;
clc
oldpath = path;
path('/CMC/tools/cadence/SPECTRE21.10.460_lnx86/tools.lnx86/
spectre/matlab/64bit',oldpath)

```

```

% Load configuration
c = techsweep_config_bsimimg_22_spectre_modified;

% Write sweep info
nch.INFO    = c.modelinfo;
nch.CORNER  = c.corner;
nch.TEMP    = c.temp;
nch.NFIN    = c.NFIN;
nch.L       = c.LENGTH';
nch.W       = c.WIDTH;
nch.VGS     = c.VGS';
nch.VDS     = c.VDS';
nch.VSB     = c.VSB';
%
pch.INFO    = c.modelinfo;
pch.CORNER  = c.corner;
pch.TEMP    = c.temp;
pch.NFIN    = c.NFIN;
pch.L       = c.LENGTH';
pch.W       = c.WIDTH;
pch.VGS     = c.VGS';
pch.VDS     = c.VDS';
pch.VSB     = c.VSB';

% Simulation loop
for i = 1:length(c.LENGTH)
    str=sprintf('L = %2.3f', c.LENGTH(i));
    disp(str);
    for j = 1:length(c.VSB)

```



```

% Write simulation parameters
fid=fopen('techsweep_params.scs', 'w');
fprintf(fid, 'parameters length = %d\n', c.LENGTH(i));
fprintf(fid, 'parameters sb = %d\n', c.VSB(j));
fclose(fid);

pause(5)

% Run simulator
[status, result] = system(c.simcmd);
if(status)
    disp('Simulation did not run properly. Check
         techsweep.out.')
    return;
end

% Initialize data blocks
for m = 1:length(c.outvars)
    nch.(c.outvars{m})(i,:,:,j) = zeros(length(c.VGS),
        length(c.VDS));
    pch.(c.outvars{m})(i,:,:,j) = zeros(length(c.VGS),
        length(c.VDS));
end

% Read and store results
for k = 1:length(c.n)
    params_n = c.n{k};
    struct_n = cds_srr(c.outfile, c.sweep, params_n{1});
    values_n = struct_n.(params_n{2});
    params_p = c.p{k};

```

```

struct_p = cds_srr(c.outfile, c.sweep, params_p{1});
values_p = struct_p.(params_p{2});
for m = 1:length(c.outvars)
    nch.(c.outvars{m})(i,:,:,j) = squeeze(nch.(c.
        outvars{m})(i,:,:,j)) + values_n*params_n{3}(m
    );
    pch.(c.outvars{m})(i,:,:,j) = squeeze(pch.(c.
        outvars{m})(i,:,:,j)) + values_p*params_p{3}(m
    );
end
end
% Noise results
for k = 1:length(c.n_noise)
    params_n = c.n_noise{k};
    % note: using cds_innersrr, since cds_srr is buggy
    for noise
        struct_n = cds_innersrr(c.outfile, c.sweep_noise,
            params_n{1},0);
        field_names = fieldnames(struct_n);
        values_n = struct_n.(field_names{4});
        params_p = c.p_noise{k};
        % note: using cds_innersrr, since cds_srr is buggy
        for noise
            struct_p = cds_innersrr(c.outfile, c.sweep_noise,
                params_p{1},0);
            field_names = fieldnames(struct_p);
            values_p = struct_p.(field_names{4});
            nch.(c.outvars_noise{k})(i,:,:,j) = squeeze(values_n)
                ;
            pch.(c.outvars_noise{k})(i,:,:,j) = squeeze(values_p)

```

```
        ;  
    end  
end  
end  
  
save(c.savefilen, 'nch');  
save(c.savefilep, 'pch');
```

Appendix B

Folded-cascode OTA Design and optimization MATLAB code

```
%To Design folded cascode stage

clearvars;
close all;

load 22nch_eguslvt_b.mat;
load 22pch_eguslvt_b.mat;

%Taking decision for Cascode side's Length of the amplifier
%Based on the Gain considerations

settling_error = 0.003; %from design spec
G = 2; %from design spec
beta_max = 1/(1+G);
beta = 0.75*beta_max;
k = 0.95;
Vdd = 0.8;
L = linspace(0.07,3,1200);
```

```

L_23 = L;
L_45 = L;
gm_Id_2 = 15;
gm_Id_3 = 15;
gm_Id_4 = 15;
gm_Id_5 = 15;
gm_gds2 = look_up(nch, 'GM_GDS', 'GM_ID', gm_Id_2, 'VDS', 0.2, 'VSB',
    0.8, 'L', L_23);
gm_gds3 = look_up(nch, 'GM_GDS', 'GM_ID', gm_Id_3, 'VDS', 0.2, 'VSB',
    0.8, 'L', L_23);
gm_gds4 = look_up(pch, 'GM_GDS', 'GM_ID', gm_Id_4, 'VDS', 0.2, 'VSB',
    0, 'L', L_45);
gm_gds5 = look_up(pch, 'GM_GDS', 'GM_ID', gm_Id_5, 'VDS', 0.2, 'VSB',
    0, 'L', L_45);
for i = 1:length(gm_gds2) %L_load
for j = 1:length(gm_gds4) %L_gain
    % L0(i,j) = beta*k/(1/((1+gm_gds5(i)*gm_gds4(j))) +...
    %           1/((1+1/3*gm_gds2(i))*gm_gds3(j)));
    L0(i,j) = beta*k./( 1./((1+gm_gds5(j))./gm_gds4(j) + 1./(1+
        gm_gds2(i)/3)./gm_gds3(i) ));
end
end
L0_minimum = (1-settling_error)/(settling_error); %L13-14 Slide
8
% figure(99);
% imagesc(L_23,L_45,L0)
% xlabel('L_{2,3} (um)');
% ylabel('L_{4,5} (um)');
% ylabel(colorbar,'Low Frequency Loop Gain');
% title('Length Sizing Given Fixed K (K=0.95)','fontweight','bold

```

```

    ');

contour_levels = floor(min(L0(:))):100:floor(max(L0(:)));

g = figure;
[C, h] = contour(L_23, L_45, L0, contour_levels);
title('Low Frequency Gain')
set(h, 'LineColor', 'k');
clabel(C, h, 'FontSize', 9);
xlabel('{\itL_2_,_3} (\mum)');
ylabel('{\itL_4_,_5} (\mum)');
hold on;

% Define the area to highlight
%x_highlight = [0, 0.5, 0.5, 0];
%y_highlight = [1, 1, max(L_45(:)), max(L_45(:))];

% Plot the highlighted area
%plot(x_highlight, y_highlight, 'r--', 'LineWidth', 2);

% Plot the point
plot(0.150, 0.235, 'k+', 'MarkerSize', 9, 'LineWidth', 2);

%Chosen length
L_23 = 0.50; %0.5;
L_45 = 0.50;%1;

% Resulting device parameters

gmb_gm3 = look_up(nch, 'GMB_GM', 'GM_ID', gm_Id_3, 'VDS', 0.2,

```

```

...
    'VSB', 0.8, 'L', L_23);
gm_css3 = look_up(nch, 'GM_CSS', 'GM_ID', gm_Id_3, 'VDS', 0.2,
...
    'VSB', 0.8, 'L', L_23);
cdd_css3 = look_up(nch, 'CDD_CSS', 'GM_ID', gm_Id_3, 'VDS', 0.2,
...
    'VSB', 0.8, 'L', L_23);
cdd_w3 = look_up(nch, 'CDD_W', 'GM_ID', gm_Id_3, 'VDS', 0.2, ...
    'VSB', 0.8, 'L', L_23);
cdd_w2 = look_up(nch, 'CDD_W', 'GM_ID', gm_Id_2, 'VDS', 0.2, ...
    'VSB', 0.8, 'L', L_23);

% Nondominant pole frequency

fp2 = 1/2/pi*gm_css3*(1+gmb_gm3)/(1+2*cdd_css3*2*(cdd_w2/cdd_w3))
;

gm_gds2 = look_up(nch, 'GM_GDS', 'GM_ID', gm_Id_2, 'VDS', 0.2, 'VSB',
    0.8, 'L', L_23);
gm_gds3 = look_up(nch, 'GM_GDS', 'GM_ID', gm_Id_3, 'VDS', 0.2, 'VSB',
    0.8, 'L', L_23);
gm_gds4 = look_up(pch, 'GM_GDS', 'GM_ID', gm_Id_4, 'VDS', 0.2, 'VSB',
    0, 'L', L_45);
gm_gds5 = look_up(pch, 'GM_GDS', 'GM_ID', gm_Id_5, 'VDS', 0.2, 'VSB',
    0, 'L', L_45);

L0_DC = beta*k./( 1./(1+gm_gds5)./gm_gds4 + 1./(1+gm_gds2/3)./
    gm_gds3 );
AV0_DC = 1./( 1./(1+gm_gds5)./gm_gds4 + 1./(1+gm_gds2/3)./gm_gds3

```

```

    );

%% Compute required unity gain frequency
s.ts = 5e-7;    % Static error
s.ed = 0.4e-2;  % Dynamic error
s.vod_final = 0.5; % For slewing

s.fu1 = 1/2/pi * log(1/s.ed)/s.ts; % fu
s.vod_noise = 400e-6; % Output noise
s.FO = 1;      % Fan-out
s.G = 2;       % Closed-loop Gain
% Parameter setup

L1 = [0.1 0.2 0.3 0.4 0.5 0.7 1];
d.rself = 0;
d.gm_ID1 = (3:0.01:27)';
d.gm_IDcas2 = 15;
d.gm_IDcas3 = 15;
d.gm_IDcas4 = 15;
d.gm_IDcas5 = 15;
d.Lcas23 = L_23;
d.Lcas45 = L_45;
d.gamma = 0.4;
d.beta = beta_max*(0.2:0.001:1)';
d.cself = 0;
%
-----

% Channel length sweep

```



```

for i = 1: length(L1)
    d.L1 = L1(i);
    [m1(i) p(i)] = folded_cascode(pch, nch, s, d);
end

% Plot
y = figure;

% Create a tiled layout for better subplot control
tiledlayout(2, 1, 'Padding', 'compact', 'TileSpacing', 'compact')
    ;

% First subplot
nexttile;
plot(m1(1).gm_ID, m1(1).ID*1e3, 'k', 'LineWidth', 2);
hold on;
plot(m1(2).gm_ID, m1(2).ID*1e3, 'k');
plot(m1(3).gm_ID, m1(3).ID*1e3, 'k--');
plot(m1(4).gm_ID, m1(4).ID*1e3, 'k-.');
xlabel({'(\itg_m)/(\itI_D)}_1 (S/A); '(a)'});
ylabel('{\itI_D_1} (mA)');
axis([5 27 0 1.2]);

leg = legend('{\itL} = 100nm', '\itL} = 200nm', '\itL} = 300nm'
    , '\itL} = 400nm', 'Location', 'north');
pos = get(leg, 'Position');
set(leg, 'Position', [pos(1)-0.13, pos(2), pos(3), pos(4)]);

% Second subplot
nexttile;

```

```

plot(m1(1).gm_ID, p(1).cself ./ p(1).cltot, 'k', 'LineWidth', 2);
hold on;
plot(m1(2).gm_ID, p(2).cself ./ p(2).cltot, 'k');
plot(m1(3).gm_ID, p(3).cself ./ p(3).cltot, 'k--');
plot(m1(4).gm_ID, p(4).cself ./ p(4).cltot, 'k-.');
xlabel({'(\itg_m)/(\itI_D)}_1 (S/A)'; '(b)'});
ylabel('{\itr_s_e_l_f}');
axis([3 27 0 0.7]);

```

```

% Search parameter setup

```

```

d.L1 = 0.2;
rself = zeros(1,6);
d.gm_ID1 = (5:0.01:27)';
d.beta = beta_max*(0.2:0.001:1)';

```

```

% Self-loading sweep

```

```

for i = 1:length(rself)
    d.cself = rself(i);
    [m1 p] = folded_cascode(pch, nch, s, d);
    % Find minimum current point and record parameters
    [ID1(i) m] = min(m1.ID);
    gm_ID1(i) = m1.gm_ID(m);
    cltot(i) = p.cltot(m);
    beta(i) = d.beta(m);

    % Use actual self-loading at optimum as guess for next
    iteration

```

```

        rself(i+1) = p.cself(m);
end

% Plot
z = figure;

% Create a tiled layout for better subplot control
tiledlayout(2, 1, 'Padding', 'compact', 'TileSpacing', 'compact')
    ;

% First subplot
nexttile;
plot(1:length(cltot), rself(1:end-1)./cltot, 'k-o', 'LineWidth',
    1);
xlabel({'Iteration'; '(a)'});
ylabel({'\itr_s_e_l_f'});
axis([1 length(ID1) 0 0.3]);
grid on;

% Second subplot
nexttile;
plot(1:length(ID1), ID1*1e3, 'k-o', 'LineWidth', 1);
xlabel({'Iteration'; '(b)'});
ylabel({'\itI_D_1' (mA)});
axis([1 length(ID1) 0 0.2]);
grid on;

% Final sizing point
ID1_opt = ID1(end);
gm_ID1_opt = gm_ID1(end);

```

```

beta_opt = beta(end);
CLtot = cltot(end);
Cself = rself(end);
beta_opt/beta_max
Freq_ratio = fp2/s.fu1;

% Compute all capacitances
CF = (CLtot-Cself)./(s.F0*s.G + 1-beta_opt);
CS = s.G*CF;
CL = s.F0*CS;

% Find current densities and widths
ID_W1 = look_up(pch, 'ID_W', 'GM_ID', gm_ID1_opt, 'L', d.L1);
ID_W2 = look_up(nch, 'ID_W', 'GM_ID', d.gm_IDcas2, 'L', d.Lcas23,
    'VDS', 0.2);
ID_W3 = look_up(nch, 'ID_W', 'GM_ID', d.gm_IDcas3, 'L', d.Lcas23,
    'VDS', 0.4, 'VSB', 0.2);
ID_W4 = look_up(pch, 'ID_W', 'GM_ID', d.gm_IDcas4, 'L', d.Lcas45,
    'VDS', 0.4, 'VSB', 0.2);
ID_W5 = look_up(pch, 'ID_W', 'GM_ID', d.gm_IDcas5, 'L', d.Lcas45,
    'VDS', 0.2);
W1 = ID1_opt/ID_W1;
W2 = 2*ID1_opt/ID_W2;
W3 = ID1_opt/ID_W3;
W5 = ID1_opt/ID_W5;
W4 = ID1_opt/ID_W4;

% Biasing voltages

```

```

VGS.Vgs_1 = look_upVGS(pch, 'GM_ID', gm_ID1_opt, 'VDS', 0.2, 'VSB', 0,
    'L', d.L1);
VGS.Vgs_2 = look_upVGS(nch, 'GM_ID', d.gm_IDcas2, 'VDS', 0.2, 'VSB',
    0, 'L', d.Lcas23);
VGS.Vgs_3 = look_upVGS(nch, 'GM_ID', d.gm_IDcas3, 'VDS', 0.4, 'VSB',
    0.2, 'L', d.Lcas23);
VGS.Vgs_4 = look_upVGS(pch, 'GM_ID', d.gm_IDcas4, 'VDS', 0.4, 'VSB',
    0.2, 'L', d.Lcas45);
VGS.Vgs_5 = look_upVGS(pch, 'GM_ID', d.gm_IDcas5, 'VDS', 0.2, 'VSB',
    0, 'L', d.Lcas45);

% [vsc, nm2g, *nm3g, *mn4g, mn5g] *=optimizable

Vg_selected = [Vdd-0.2-VGS.Vgs_1 VGS.Vgs_2 0.2+VGS.Vgs_3 Vdd-0.2-
    VGS.Vgs_4 Vdd-VGS.Vgs_5];

function [m1, p] = folded_cascode(dev1, dev2, s, d)

kB = 1.3806488e-23;
gm_gds1 = look_up(dev1, 'GM_GDS', 'GM_ID', d.gm_ID1, 'L', d.L1);
wt1 = look_up(dev1, 'GM_CGG', 'GM_ID', d.gm_ID1, 'L', d.L1);
cgd_cgg1 = look_up(dev1, 'CGD_CGG', 'GM_ID', d.gm_ID1, 'L', d.L1)
    ;
gm_gds2 = look_up(dev2, 'GM_GDS', 'GM_ID', d.gm_IDcas2, 'L', d.
    Lcas23);
cdd_gm3 = look_up(dev2, 'CDD_GM', 'GM_ID', d.gm_IDcas3, 'L', d.
    Lcas23);
cdd_gm4 = look_up(dev1, 'CDD_GM', 'GM_ID', d.gm_IDcas4, 'L', d.
    Lcas45);

m1.ID = NaN(1, length(d.beta));

```

```

m1.gm_ID = NaN(1,length(d.beta));
p.cltot = NaN(1,length(d.beta));

for j = 1:length(d.beta)

    % Compute excess noise factor and total load to meet the
    noise spec
    alpha = 2*d.gamma*(1+ d.gm_IDcas5./d.gm_ID1 + 2*d.gm_IDcas2./
        d.gm_ID1);
    cltot = alpha./d.beta(j)*kB*dev1.TEMP/s.vod_noise^2;
    CF = (cltot-d.cself)./(s.F0*s.G + 1-d.beta(j));
    CS = s.G*CF;

    % Estimate kappa and compute required gm for given fu1
    kappa = 1./(1+ 1./gm_gds1.*d.gm_ID1/d.gm_IDcas2 + 2/gm_gds2);
    % X = s.vod_final/2.*d.beta(j).*d.gm_ID1; % For slewing
    optimization
    % X(X<1) = 1;
    % s.fu1 = 1/2/pi *(X-1+log(1*X/s.ed))/s.ts; % For slewing
    optimization
    % gm1 = 2*pi*s.fu1.*cltot./d.beta(j)./kappa; % For slewing
    optimization
    gm1 = 2*pi*s.fu1*cltot./d.beta(j)./kappa; % W/O slewing
    optimization
    ID1 = gm1./d.gm_ID1;

    % Compute actual feedback factor and find match
    cgs1 = gm1./wt1;
    cin = cgs1.*(1 + cgd_cggs1.*d.gm_ID1/d.gm_IDcas2);
    beta_actual = CF./(CF+CS+cin);

```

```

m = interp1(beta_actual, 1:length(beta_actual), d.beta(j), '
    nearest', 0);
if(m)
    m1.ID(j) = ID1(m);
    m1.gm_ID(j) = d.gm_ID1(m);
    p.cltot(j) = cltot(m);
    % m1.X_valid = X(m); % For Slewing
end
end

% self loading
gm3 = m1.ID.*d.gm_IDcas3;
gm4 = m1.ID.*d.gm_IDcas4;
p.cself = gm3*cdd_gm3 + gm4*cdd_gm4;

end

```