AUTOMATION OF ANALOG CIRCUIT DESIGNING AND SIMULATION

A Project Thesis submitted in partial fulfillment of
Requirements for the Degree of

Master of Technology

in

VLSI Design and Embedded Systems

BY

SARAGADAM SAILAJA

213EC2211

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela

2013 - 2015

AUTOMATION OF ANALOG CIRCUIT DESIGNING AND SIMULATION

A Project Thesis submitted in partial fulfillment of
Requirements for the Degree of

Master of Technology

in

VLSI Design and Embedded Systems
Electronics and Communication Engineering

BY

SARAGADAM SAILAJA

213EC2211

Under the guidance of

Prof. Debiprasad Priyabrata Acharya

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela

2013 - 2015

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Odisha, India – 769008

CERTIFICATE

This is to certify that the project thesis entitled

'**Automation of Analog Circuit Designing and Simulation**',

submitted by

Ms. Saragadam Sailaja

bearing Roll No. 213EC2211, in partial fulfillment of the requirements for the award of

Master of Technology

in

'Electronics and Communication Engineering'

with specialization in

'VLSI Design and Embedded Systems'

during session 2013-15 at National Institute of Technology, Rourkela., is an authentic work carried out by her under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any Degree or Diploma.

Place: Rourkela
Date: 22nd May, 2015

Prof. D. P. Acharya
Department of ECE,
National Institute of Technology,
Rourkela

Dedicated to

Mr. Saragadam Sudarsanam,

Mrs. Saragadam Janaki Sudarsanam

and to the beautiful part of my life at NIT Rourkela

ABSTRACT

With the advent of sub-micron CMOS process technologies, application of drain current model equations alone, which is good for long-channel devices, to design an analog circuit is becoming futile. So, the design of analog circuits is the time-limiting bottleneck, in comparison with the digital counterparts, of the mixed IC signal designs. Shortage of analog integrated circuit design automation tools results in relying on engineering experience and time-consuming trial-and-error simulation runs to design an analog IC.

The research presented in this thesis aims to improve the efficiency of analog IC design process with a new design methodology. This new methodology is mainly based on the small-signal analysis and designers' experience on how to achieve the desired specifications. A CAD tool is also developed to characterize the devices such as MOSFETs, Resistors, and Capacitors, which is useful to follow the newly developed design methodology. To demonstrate the usefulness and reliability of the new methodology and the tool, some design examples are also presented. This thesis has also given the study report on existing analog IC design automation methods.

Apart from the design automation, another major time-consuming constraint is to do the Simulation for existing designs, mainly if it requires multiple analyses to be performed to extract the required parameters. This issue is also addressed in this thesis, to some extent, by taking a test-case of VCO circuit.

ACKNOWLEDGEMENT

This research work is one of the significant achievements in my life and is made possible because of the unending encouragement given by so many in every part of my life. It is an immense pleasure to have this opportunity to express my gratitude and regards to them.

First and foremost, I would like to express my deepest gratitude to my project guide Prof. D.P. Acharya, for opportunity he spawned for me to work in an industry which has been very beneficial to attain real-time knowledge in the domain of analog circuit design. I am very much thankful to him for his invaluable advice on problems I have faced, for his encouragement and care, and for believing in my abilities throughout the period.

I am also extremely grateful to Mr. Prajit Nandi and Mr. Priyanko Mitra, from Sankalp Semiconductors Pvt. Ltd. for guiding me throughout my stay in the industry and for providing me with valuable suggestions regarding my work. I also extend my sincere thanks to Siddhartha Hazra, Kaustav Bose, Amiya Prasad Behera, Toushik Basu and Debraj Sengupta for giving comprehensive knowledge in this field and for helping me time to time in solving every single doubt, whether it was a small or big.

I express my heartfelt respect and gratitude to Prof. K.K.Mahapatra, Prof. P.K.Tiwari, Prof.A.K.Swain and Prof. N.Islam for their thoughtful teaching, for motivating us to do various hobby projects and for making available all the necessary facilities and infrastructure.

Being an M. Tech student here, I had a privilege of getting and working together with many of my peers nationally. Stay with such diversified classmates have made it more delighted and memorable. I express my whole-hearted thanks to Rakesh, Srikanya, Anil Kumar, Anusha, Ashutosh, Hanumantha Rao, Krishna Reddy and Sidharth, who stayed with me even at my hard times and made my life in NIT Rourkela the most joyful and memorable one.

Finally, I owe my heartiest gratitude and love to my father Mr. Saragadam Sudarsanam, mother Mrs. Saragadam Janaki Sudarsanam, sister Saragadam Srividhya and brother Saragadam Bharat Sri for their unconditional affection, support, inspiration and sacrifices.

Saragadam Sailaja

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

CMOS	Complimentary Metal Oxide Semiconductor
IC	Integrated Circuit
CAD	Computer Aided Design
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
SPICE	Simulation Program with Integrated Circuit Emphasis
VCO	Voltage Controlled Oscillator
EDA	Electronic Design Automation
ADE	Analog Design Environment
BJT	Bipolar Junction Transistor
ASIC	Application Specific Integrated Circuit
SoC	System on Chip
GPCAD	Geometric Progression CAD
SiO_2	Silicon Dioxide
PCB	Printed Circuit Board
DC	Direct Current
AC	Alternating Current
PSS	Periodic Steady State
PERL	Practical Extraction and Reporting Language
CGI	Common Gateway Interface
OCEAN	Open Command Environment for Analysis
g_m/I_D	transconductance generation efficiency
f_T	unity currrent gain frequency
PDK	Process Design Kit
L	Channel Length
W	Channel Width
PSRR	Power Supply Rejection Ratio
BSIM	Berkley Simulation IGFET Model
TSMC	Taiwan Semiconductor Manufacturing Company Ltd.

GUI	Graphical User Interface
UGB	Unity Gain Bandwidth
RC	Resistance and Capaitance
PM	Phase Margin
PLL	Phase Locked Loop
RF	Radio Frequency
PVT	Process, Voltage and Temperature
LDO	Low Voltage Dropout Circuit

1 THESIS OVERVIEW

1.1 Introduction

The invention of computers has affected every field of life and has made our day-to-day tasks easy. With the help of computers, the data processing and design tasks can be performed much faster and accurate than using traditional hand calculated analysis. As the Moore's law states that 'Number of transistors every square inch of an IC had multiplied consistently twice per year since the IC was invented', along with the quick development rate in CMOS technology, every peripheral domain has a comparative growth, mainly CAD tools. These tools and computer- based design methodologies have opened a new door of possibility to analog and digital IC design engineers. The most potent driving force for using them in Microelectronics is competition in the semiconductor and electronics industry, as it increases the productivity and reduces the design time and cost for the IC design. These CAD tools have risen as a help in assisting analog design engineers to choose and optimize various design models and technology.

In digital IC design, the design procedure is almost fully automated enabling semiconductor companies to produce advanced digital IC designs very fast. These design styles are categorized into full custom, semi-custom and programmable. The ideology to automate a circuit design has been migrated from digital to analog. However, there are inborn complexities of the analog design process. The non-linear nature of analog devices is the major cause of these. This forces the analog designers to use traditional methods to design a circuit. They are using

long-channel device equations to calculate the device parameters and also use trial and error method to take care of all the short channel effects. These make the analog circuit design automation for analog circuits difficult.

1.2 Motivation

Many board level analog/digital systems have begun to migrate down toward single chip implementation, and it has become clear that design of analog parts of these chips has been often the critical time-limiting bottleneck of the overall design process. Hence, CAD tools must be developed to cope with both the complexity of the large-scale analog circuit designs, and with the requirement of rapid design times.

With the advent of sub-micron CMOS technologies, application of long-channel device equations, is becoming inaccurate. The behavior estimated using these formulae appear to be different from that predicted by the sophisticated circuit analysis simulators such as SPICE. Hence, using circuit simulators in analog CAD systems could be an effective design approach.

A new design methodology based on simulation and knowledge-based approach that simplifies and speeds up the process of designing analog ICs is expected to help analog designers tackle the challenges as the transistor channel lengths and supply voltage scale down with each generation of CMOS technologies.

1.3 Objective

The primary objective of this thesis is to develop and examine a new computer-based

design methodology, to develop a CAD tool that automates the process of characterizing the MOSFET, independent of technology used, and to automate the simulation analysis of complex analog building blocks, in this case, a VCO. The specific objectives are to contribute to the design of analog IC as follows:

1. To provide analog circuit designers with a design methodology to design basic parts of their circuits automatically, so that they can concentrate on designing complex parts of it.
2. To reduce the design cycle time of analog ICs by providing accurate design solutions so that analog designers need not run time-consuming trial and error simulations.
3. To automate the simulation analysis process for complex analog circuits, for which multiple analyses have to be performed to get certain parameter values.

1.4 Thesis Organization

Following an overview, the rest of the dissertation is structured as follows.

- Chapter 2 introduces the concept of analog IC designing and gives a brief literature review of analog IC automation and developed CAD tools.
- Chapter 3 details the basic concepts related to the MOS devices, their modeling structures, the EDA tool Cadence Virtuoso ADE and the scripting languages.
- Chapter 4 proposes a design methodology for analog circuit designing and explains its advantages and limitations.
- Chapter 5 points out the designed tool for characterizing the analog devices and the flow used to develop it.

- Chapter 6 lists various design flows developed using the proposed design methodology and the experimentation results obtained by designing the analog circuits.
- Chapter 7 explores the automation of multiple simulations to be performed on analog circuit VCO, which is considered as a test-case in this dissertation.
- Chapter 8 gives the concluding remarks with the discussion on the future scope for this work.

2 LITERATURE REVIEW

2.1 Analog IC design

An Integrated Circuit is an electronic circuit where all the discrete components (both passive and active elements) are mounted on a single chip, regularly Silicon. The idea of integrating electronic circuits into a single device was introduced in 1949 and is conceived in 1960. In 55 years, the technology has progressed from producing small chips containing tens of components to fabricating processors and memories comprising more than a billion devices. Also, the technology node has dropped from about 25um in 1960 to 14nm in 2014, resulting in a tremendous improvement in the speed of an IC.

Though in earlier days BJT technology is used, MOS technologies became more practical from the early 1960s, initiating a revolution in the semiconductor industry. This is because of lower power consumption, higher packaging density, simple fabrication process and lesser design cost, for CMOS comparing with BJT. The digital market had been rapidly captured by this CMOS technology and the design process is a lot easier in this domain. This results in all the signal processing applications to move from the analog to the digital domain. Though digital signal processing become very powerful because of advances in IC technology and CAD tools, it can only process digital signals, in contrast to nature. So the demand for analog IC designs is still prevailed for interface circuits in the fields of wireless communication systems, biomedical instrumentation, and remote sensing.

Analog IC designing[9] is a challenging, iterative and knowledge-intensive task, and so is referred as 'state-of-art' work. There is no circuit independent design procedure for analog circuits till date. The main hurdle for this is that analog circuits have to deal with analog signals, which are continuous in amplitude and are continuous time-dependent. This makes the analog designing more experience-dependent than to be in a systematic way. The various phases and iterative nature of an analog IC design process are as shown in Figure 2.1 and is described in detail below.

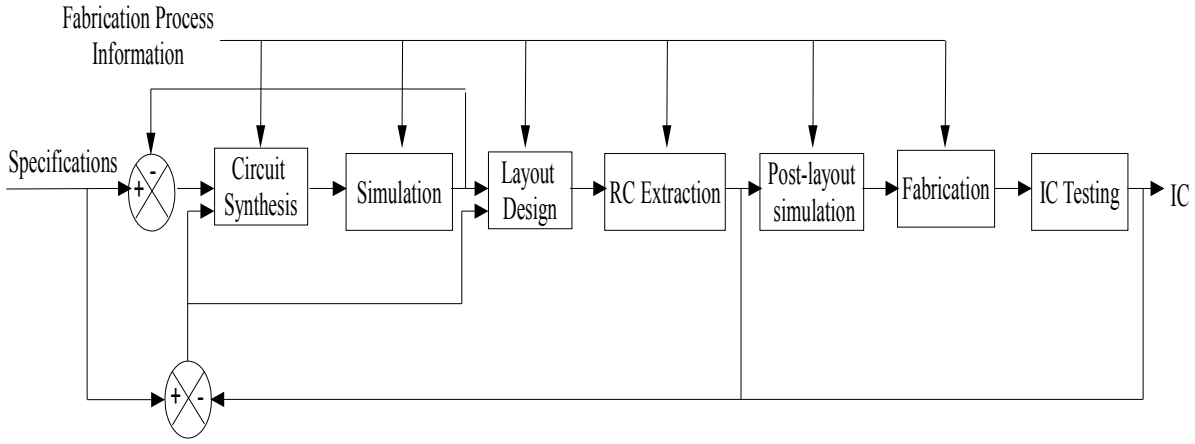


Figure 2.1: Analog IC Design Process

2.1.1 Circuit Synthesis

Analog circuits are generally different variants of the same architecture, adjusted to meet required specifications. So circuit synthesis is basically selecting the right architecture from existing alternatives to suit the particular needs of the application. Then they use simple analytic design equations, that are derived from basic laws of circuit theory and fundamental operation of associated devices, to predict circuit behavior. However, as the circuit complexity increases, the usage of these equations becomes complex and also inaccurate because of short channel effects.

Thus, small-signal modeling and analysis method is used, which overcome the aforementioned problems and also facilitates qualitative understanding of the circuit. Algorithmic knowledge in the form of proven, step-by-step design strategies, which develops from the experience, is beneficial for the designers. This is useful to minimize possible performance conflicts and also to relax performance trade-offs. These design algorithms are not unique for a circuit and are designer dependent generally. So a generalized proven algorithm has to be used.

Another important aid to this process is knowledge of circuit heuristics. This, along with design intuition gained by experience, is helpful for designers to deal with the circuits and predict their behavior. A typical example to circuit heuristics is the fact that the small-signal gain of a circuit decreases with the increase in bias current.

Hierarchical decomposition of a complex analog circuit is done into the smallest level of the cell, for which design algorithm has already been present. This helps in reuse of design algorithms so that the database is kept to be with minimal algorithms. An example of hierarchical design is an operational amplifier, which is decomposed into current-mirrors, current-sources etc.

Thus, the output of this phase is a circuit designed, which is referred as 'first cut design'. Its performance is assessed in the next phase.

2.1.2 Simulation

Circuit Simulation is used to predict the behavior of analog IC, prior to fabrication. SPICE and SABER are tools to do this. The performance of the circuit is checked after simulation and the manual tweaking of the circuit design variables is done to improve the performance if needed. The number of iterations of the simulation and tweaking design variables

depends on the designers expertise and is done until the required specifications are met in an acceptable tolerance.

2.1.3 Layout designing

The next phase is to design a layout (placement and routing), which refers to physical and geometric description of a circuit on a silicon die. Parasitic components, device matching, symmetry, and crosstalk are the typical layout constraints which affect the performance of a circuit.

2.1.4 RC Extraction and Post-Layout Simulation

The parasitic components of the circuit introduced by its layout are extracted in this phase. As these may deteriorate the circuit performance, post-layout simulation is done to check it. Necessary adjustments in the layout are made to resolve the errors. If not sufficient, then the designer has to go to adjust in the circuit itself and the process continues.

2.1.5 IC Fabrication and Testing

The circuit is fabricated, after getting an acceptable laid-out design. Then it is extensively tested for any errors. Then they are released for mass production or returned to the designer if any errors are there, for further adjustments of circuit or layout. In general, 70% of design time goes for IC verification than its designing.

2.2 Automation of Analog IC design

Economic and technical factors have been encouraging designers to place both analog and digital counterparts of a system onto the same die. Application Specific Integrated Circuits (ASIC), which are custom design chips are migrating towards the integration of the complete system on chip (SoC).[2] Recent studies state that almost 70% of all ASIC's designed are mixed analog and digital. Though, on this chip, only 30% of the area is occupied by the analog circuit, it generally takes 70% of total chip design time to design it. This makes analog circuit design in the whole design process a critical time-limiting bottleneck. Hence, it is clear that developing CAD tools to automate analog designing is important to handle complex analog circuit designs and also to reduce design time. The analog design automation approaches[6] are basically classified as shown in Figure 2.2

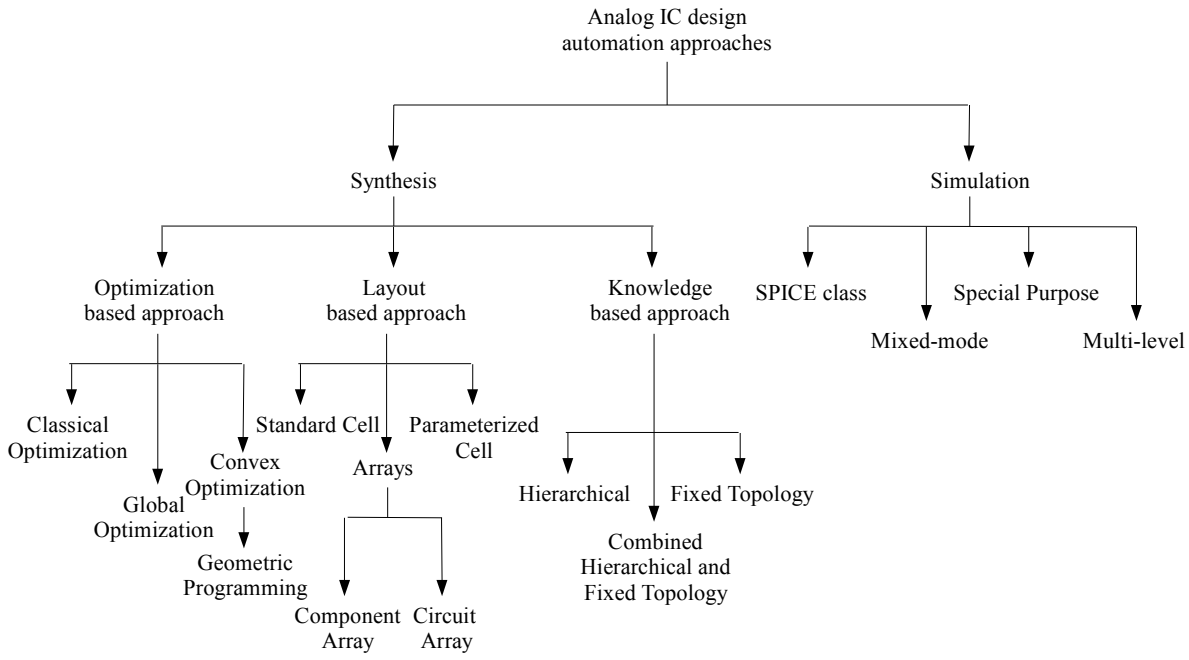


Figure 2.2: Analog IC design automation approaches

The automation can be done in two phases of the design process, i.e., at the synthesis level and the simulation level. The approaches in both levels are explained in detail below.

2.2.1 Synthesis level

At this level, the automation is possible in both circuit design synthesis and layout synthesis. The former one refers to selecting circuit topology, finding MOSFET sizes and biasing points of them so that required specifications are met, whereas the latter one refers to transforming the synthesized circuit onto silicon die. In this level, classification is made on the basis of how the problem has been addressed.

2.2.1.1 Optimization based approach

This approach is used in first attempts of automating the design. In this, the specifications and starting points are taken from the user and the optimization algorithms are iteratively used to meet the given specifications. A simulator is also used in the optimization loop to check the performance of the circuit in every loop. This approach is again sub-divided into classical, global and convex optimization.

Classical optimization approach uses the methods of steepest descent, sequential quadratic programming, Lagrange's multiplier approach. DELIGHT-SPICE, ECSTACY, and ADOPT tools fall in this category. Though this approach is less time-consuming, the major disadvantage of this approach is that they can only find local optimal designs. If the starting point given by the user is far away from the actually optimized design, they may not be able to find the design. Also, the obtained design may not be the feasible one.

Global optimization approach uses Simulation Annealing and multi-start techniques and so have overcome the above disadvantages. But it is much time-consuming and is expensive computational wise. ASTR/OBLX and OPTIMAN tools fall in this category.

Convex optimization approach minimizes the convex functions subject to convex sets by using equality and inequality constraints to express the boundaries. This method can solve large problems with thousands of variables and constraints, very fast and efficiently and gives a globally optimized design, irrespective of starting point given. GPCAD tool is an example of this approach.

2.2.1.2 Layout based approach

It is a semi-custom bottom-up approach, where analog components are previously designed and their layouts are also designed for different configurations, sizes, levels of complexity. Then from the circuit designed, the required functions are selected from the database and are interconnected.

SLIDE and CAPSIZE tools are of this type using switched capacitor arrays, which can fabricate complex filters having more than 50 filters, without any trimming and external components. But they can't provide design flexibility for high-performance analog circuits. They are restricted to components used and also their values. Also, the fabricated designs are cost-ineffective in terms of Si area, as the unused array components are left.

Standard cells address this problem of Si usage, by using pre-designed and laid-out standard cells. The required function is implemented by selecting the necessary cells and placing and routing. The main disadvantage of this approach is that it is difficult to configure and

maintain enough library of cells.

Usage of parameterized cells is similar to that of standard cells, with an additional feature of customizing the cells and their parts, according to the required specifications. A module generator, which is a program to generate the layout of the design, from the given set of inputs, is used and its complexity controls the degree of flexibility. AIDE2 and CONCORD are the tools developed based on this approach.

2.2.1.3 Knowledge-based approach

In this approach, the domain knowledge (design equations and circuit heuristics) on designing analog circuits is used. The design task is addressed in a full - custom way and so it allows maximum flexibility and can achieve better performance.

Hierarchical design approach involves breaking a complex circuit along with its specifications to smaller blocks and its corresponding specifications. The number of hierarchical levels depends on the complexity of the circuit and sophistication of the design system. But a great deal of domain knowledge is required to do hierarchical splitting. PROSAIC, BLADES, and OASYS[1] are developed using this approach.

Fixed topology approach uses fixed, unsized, device level circuit topologies are stored in the database and also the knowledge to size the devices is required. Thus, the topology required is first selected and then MOSFETs are sized according to required specifications. IDAC2, OPASYN and OAC systems follow this approach. Design flexibility is limited in this case and also large overhead is created if a topology has to be added to this system.

The combined hierarchical and fixed topology approach combines features of both the

methods. It provides design flexibility and also topology modification. With smaller circuit library, a wider circuit performance can be covered. ASAIC and CAMP systems fit in this class.

2.2.2 Simulation Level

With the accessability of computing resources, simulation-based automation came into light and is most widely recognized strategy found in recent researches. In simulation-based designing, as on account of AIDA-C, a circuit simulator like SPICE, is utilized to access the circuit execution.

One major disadvantage observed from all the aforementioned tools is poor and inefficient usage of circuit simulators and the information derived from them. Depending completely on the domain knowledge or on the simulator to perform iterative checks to design a circuit is not an effective way. Using both of them in a balanced way, a new design methodology is introduced in Chapter 4.

3 BASIC CONCEPTS

3.1 MOSFET Device Physics

Metal Oxide Semiconductor Field Effect Transistor (MOSFET) is a four terminal device, which is mainly utilized for signal switching and amplification. Figure 3.1 shows the structure of n-channel MOSFET.

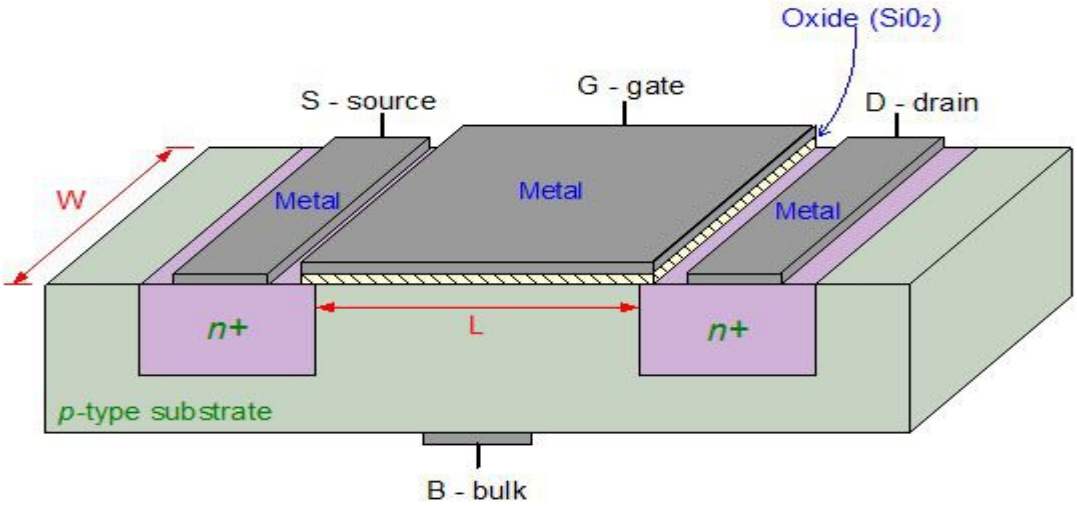


Figure 3.1: n-channel MOSFET structure

Two heavily doped n+ areas, framing Source (S) and Drain (D) terminals are diffused into a lightly doped p-type substrate area called Bulk. A heavily doped conductive poly-Silicon working as Gate (G) is insulated from the substrate by thin SiO_2 layer. The channel length (L) is defined as the horizontal distance between the source and drain regions. The perpendicular width of these S and D regions is the channel width (W). The device characteristics are controlled by substrate voltage.

In CMOS technology, both n-channel MOSFET and p-channel MOSFET devices are used. The PMOS device can be obtained by negating all the doping types in an NMOS structure. As it is difficult to take two different kinds of substrates for a circuit design, a single wafer is used for both NMOS and PMOS and either one of them is fabricated on a local substrate called 'well'. In general, the PMOS structure is fabricated on n-well as shown in Figure 3.2.

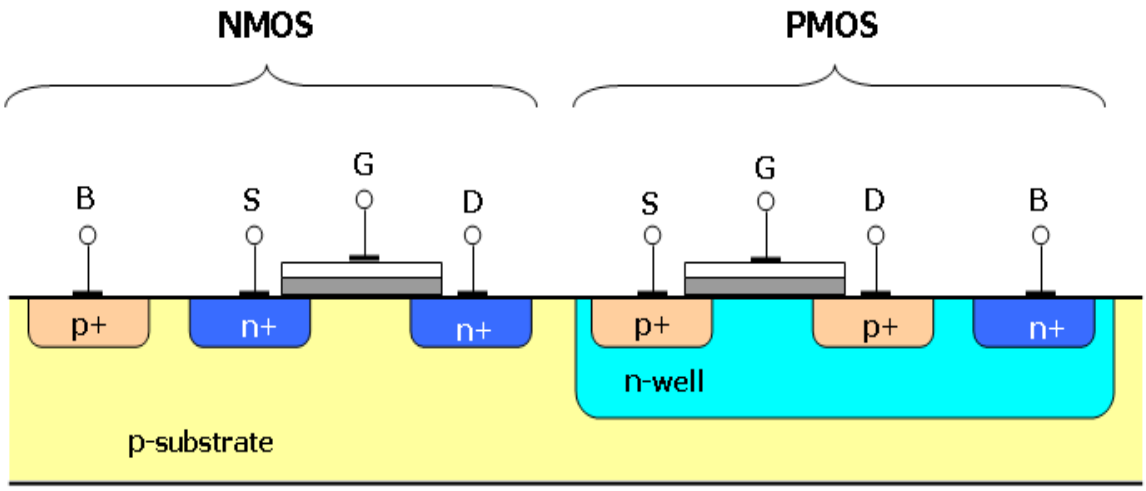


Figure 3.2: NMOS and PMOS structures on same substrate

Though MOSFET is a four terminal device, in general, the bulk is tied to V_{DD} and Ground terminals for PMOS and NMOS devices respectively. So, only three terminals are efficient and Figure 3.3 shows how these devices are represented.

Figure 3.3: NMOS and PMOS symbol representation

Coming to the current-voltage characteristics of MOSFET device, as it is a voltage-controlled device, the current flows from drain to source regions and is controlled by gate voltage V_{GS}. Depending on the channel structure formed due to terminal voltages, the current flow is controlled and is divided into three regions of operation (Saturation, Triode, Cutoff regions) as shown in Figure 3.4.

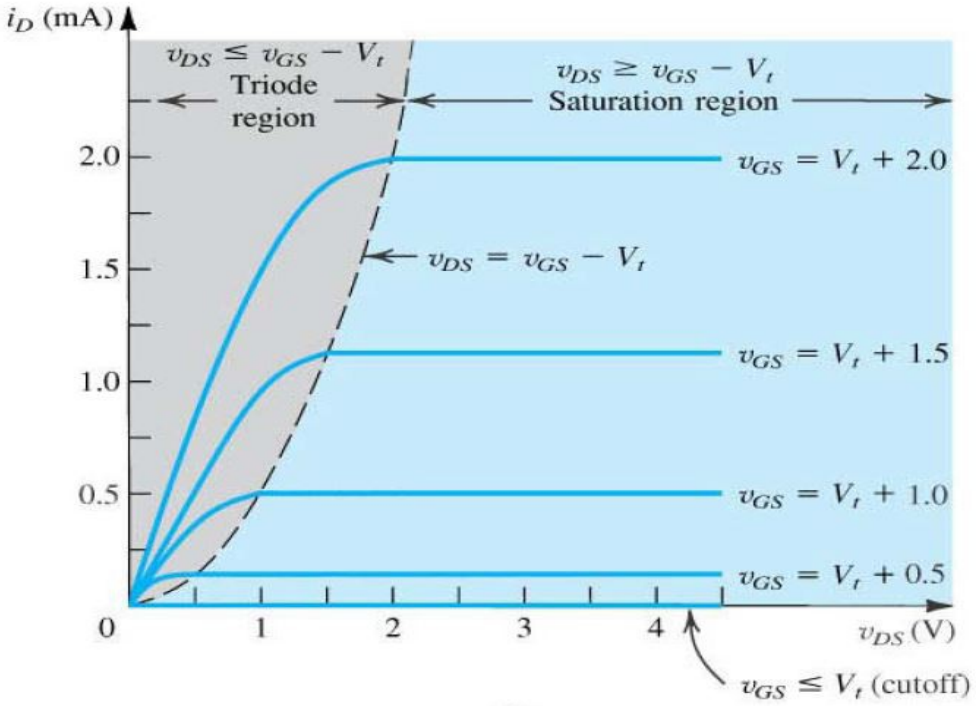


Figure 3.4: I-V characteristics of MOSFET

The drain current equation in these three regions of operation is as given below.

For Cutoff region ($V_{GS} < V_{th}$),

$$I_D \approx 0$$

For Triode region ($V_{GS} > V_{th}$ and $V_{DS} < V_{GS} - V_{th}$),

$$I_D = \mu_n C_{ox} \frac{W}{L} [(V_{GS} - V_{th}) V_{DS} - \frac{1}{2} V_{DS}^2]$$

For Saturation region ($V_{GS} > V_{th}$ and $V_{DS} > V_{GS} - V_{th}$),

$$I_D = \frac{1}{2}\mu_n C_{ox}\frac{W}{L}(V_{GS} - V_{th})^2(1 + \lambda V_{DS})$$

These equations are known as the square-law model for the MOSFET. The variables used in them are described below.

I_D - Current flow from drain region

μ_n - Electrons mobility

C_{ox} - Gate-oxide capacitance per unit area

λ - Channel length modulation

V_{th} - Threshold Voltage

V_{GS} - Gate-to-source Voltage

V_{DS} - Drain-to-source Voltage

3.2 Small-signal modeling of MOSFET

Applying this design equation to every possible device in a circuit is difficult, particularly when there is a large number of variables involved and more devices are present. So, in order to derive such analytic design equations without losing much accuracy and to develop qualitative understanding of the circuit behavior, circuit designers use a special technique called small-signal modeling.[3] This modeling also takes care of the short channel effects of MOSFET, like Channel length modulation, body effect, sub-threshold V_{DS} conduction. Figure 3.5 shows the small-signal model for an n-channel MOSFET.

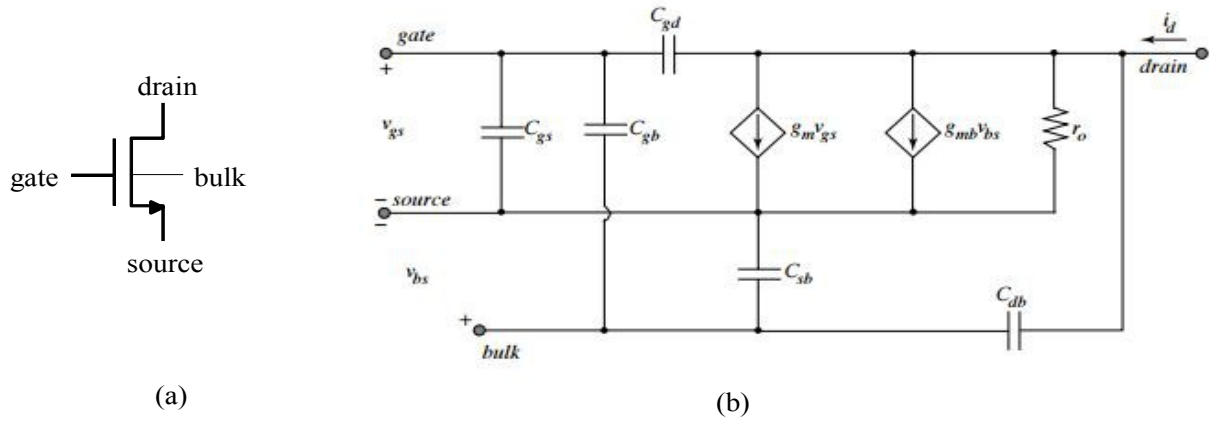


Figure 3.5: (a) n-channel MOSFET symbol; (b) Small-signal model for it

In this model, the complex MOSFET structure is represented as a set of transconductances (g_m, g_{mb}, and $1/r_o$) and the parasitic junction capacitances between each terminal. The dependent current source $g_m V_{gs}$ reflects how drain current is controlled by gate-source voltage. The channel length modulation effect, which corresponds to the dependence on V_{ds} voltage source, is represented as r_o resistance. The current-source $g_{mb} V_{bs}$ reflects the body effect on the MOSFET. The parasitic junction capacitances, shown in Fig 3.5 affects the behavior of the MOSFET, mainly on high-frequency operation. Similar to this model, the p-channel MOSFET is modeled as shown in Figure 3.6

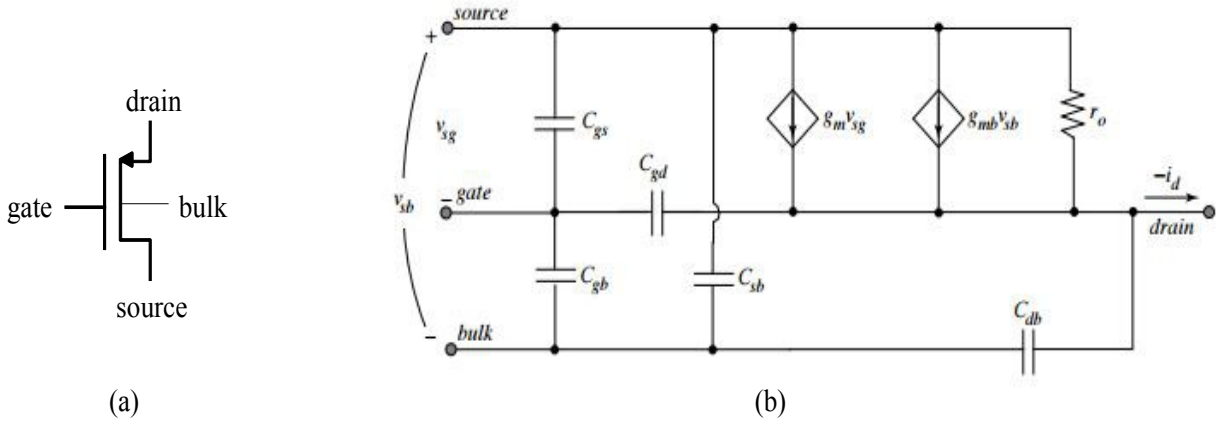


Figure 3.6: (a) p-channel MOSFET symbol (b) Small signal model for it

3.3 Cadence Virtuoso Analog Design Environment

Electronic Design Automation (EDA) is a class of software tools to design electronic systems such as ICs, PCBs. Cadence is one such environment, which permits distinctive applications and tools to incorporate into a single framework, hence permitting to support all phases of IC design and verification from a single domain. It is generalized and supports all the fabrication technologies. It gives all the analog designers a design flow from Schematic to GDSII stream format, which is optimized and has been proven all the way from circuit design, layout and measurements. An overview of this flow is shown in Figure 3.7 [8]

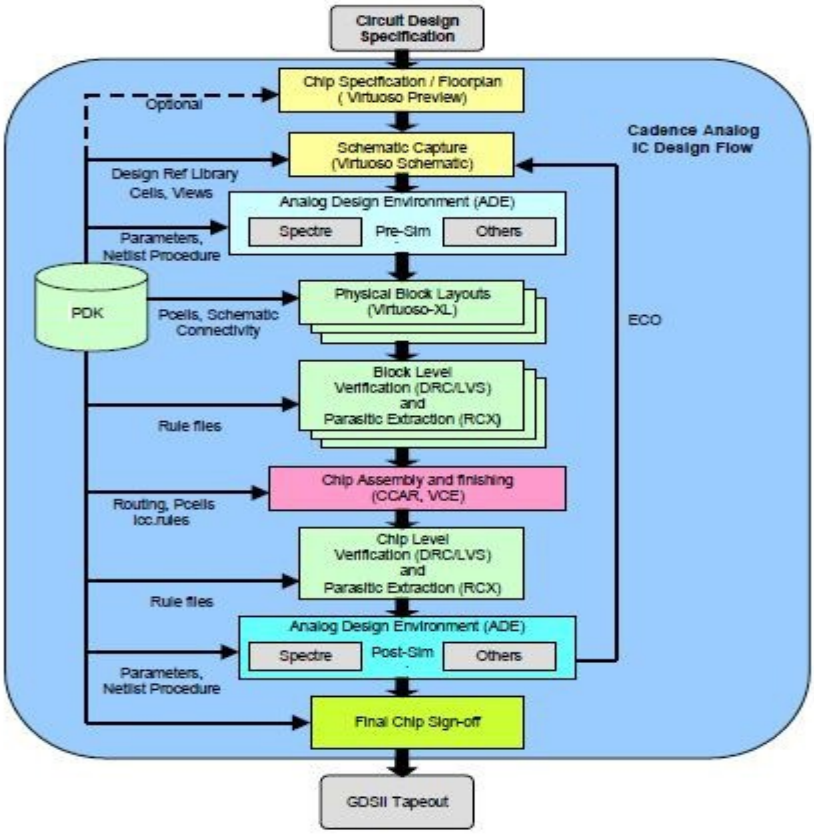


Figure 3.7: Cadence Analog Design Flow

Our area of interest in this flow is only till schematic designing and verifying it in a multi-mode simulation. This is because the automation of analog designing is done till getting a design schematic for the given specifications. The design flow till the schematic design is thus briefly explained in this section.

Cadence can be made to run only on Unix terminals or on Linux. The required setup files for running it are:

1. cds.lib – A text file where all the information regarding technology libraries, user-defined libraries are stored.
2. display.drf – A file which contains all the display related information like the color of MOSFET to be shown to the user etc.
3. .cdsenv – An environment setup file for Cadence
4. .cdsinit – A file which stores all information regarding the special tools used like their extraction paths
5. .cshrc – All the license related information is stored in this file.

With these files existing, the Cadence environment can be run. For the flow till schematic design, mainly 3 licenses are needed. They are: “111” license to invoke Virtuoso, “schematic L” license for Schematic editing and “Spectre” for running Simulation. Spectre is a default Simulator given with the Cadence. Any user-defined library created in this environment is attached to existing a technology library so that the information regarding that library is only used.

The conventional technique for depicting a transistor-level or gate-level design is by means of the composer schematic editor. It gives various capacities to encourage quick and

simple design entry, including configuration collaborators that speed basic assignments by as much as 5x. A cell view is created through this editor, where the circuit design is made by describing all the electrical properties of the required components and their interconnections. This data is crucial for generating the netlist file corresponding to the design, which is utilized in later stages of the design. To generate this file, the simulator to be used and the other simulation environment variables are related inputs.

Once the transistor-level description is done, its functionality and electrical performance have to be verified using a Simulation tool, like Spectre. Initially, the model library file path to be used, which has all the properties of the components selected, has to be selected. The design variables for the component parameters, which are used to be assigned values at the time of simulation, has to be allotted some default values. Different kinds of analysis can be performed on a design like DC, AC, transient, PSS analyses etc. The analyses, which are required to be performed on the design has to be selected and the input variables related to the analysis like start time, stop time etc., has to be given. The design variables from the schematic design can be assigned multiple values so that the output can be observed at various values of them. A text file will be created from all this information, which is used by the simulator for its operation. Then the simulation can be made to run and all the output data would be stored in the form of a RAW file or PSF file in general. The output can be seen using tools like WaveSkill. The required output expression can be calculated using Waveform Calculator.

From the output waveforms, it is checked whether the given specifications are met or not. If not, necessary changes have to be done in the schematic design and is verified again.

3.4 Scripting Languages used

Scripting Languages, which is categorized as dynamic high-level languages, are the interpreter language, which can glue all codes between different tools, environments, and languages. These scripts may have less speed of execution than the static high-level languages like C, C++, but the programming time reduces. There is a high level of flexibility to use the data-types. The scripting languages used in this automation tool are PERL, Cadence SKILL, Shell scripting, and OCEAN scripting, which are explained in detail in this section.

3.4.1 PERL scripting

PERL[10] is an acronym for 'Practical Extraction and Reporting Language', which has the motto of 'There is more than one way to do it', emphasizing its flexibility. It is developed by Larry Wall in 1987. It is open-source software, most portable and is intended to be simple for humans to write. It has rich data structures like hashes. It supports both procedural and object-oriented programming. It gets language structures and ideas from various languages like C, awk, sed and even English. It is mainly used for CGI programming, extracting information from one format and translating into another. In our tool, this language is used to generate the GUI (Graphical User Interface) and to generate plots.

3.4.2 Cadence SKILL scripting

It is developed by Cadence to use along with their tools to add features to them, along with the inbuilt functions. Its power is gotten from these substantial libraries of subroutine calls

to control data structures related to design like nets, cells, instances etc. It is mainly used to custom circuit designing and to handle Cadence database. SKILL shell can be run in both batch mode and interactive mode. All the procedures of circuit designing using Cadence tool can be made automated using this language. It is an interpreter language like PERL. The syntax and keywords used should be taken care so that they won't cross over with the inbuilt functions.

3.4.3 Shell scripting

In UNIX structure, shell surrounds the kernel and act as user-interface to the operating system. It is a command line interpreter, which usually intakes the command, execute it and shows the output on the screen. A shell script is basically a set of these commands written in a plain text file so that all can be executed in series. Shell commands can be created according to the user requirement. It is used to automate system administration work. There are three versions of this scripting: Bourne shell (sh), C shell (csh), and Korn shell (ksh).

3.4.4 OCEAN scripting

OCEAN stands for 'Open Command Environment for Analysis'. It is used to analyze the output simulation data of the design. It can be run on both Command Line Interpreter as well as UNIX shell. The commands can be given in both interactive session for it one after the other or as a text file and can be loaded directly. It is used to verify the circuit performance and output data can be plotted or can be written in the output file. The simulations can be made to run in a non-graphical mode using this. The statistical and parametric analysis, which take a longer time, can be taken care effectively.

4 THE DESIGN METHODOLOGY

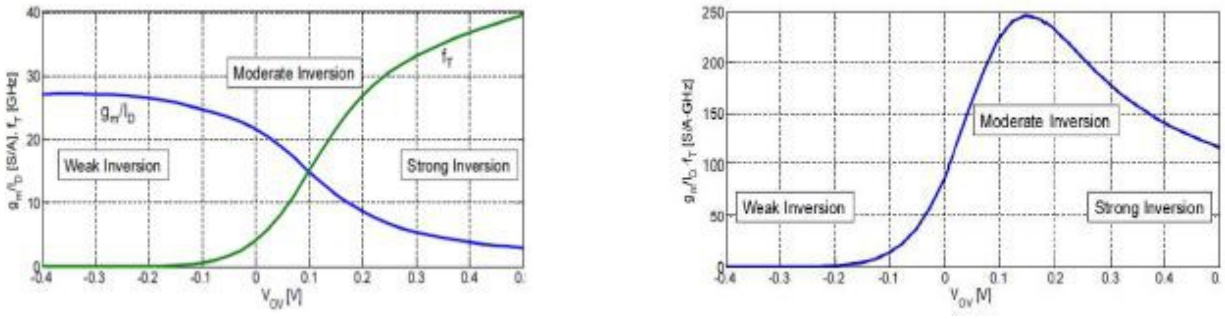
4.1 g_m/I_D methodology

One of the important trade-off required to design an analog circuit is between speed and power. In CMOS analog circuits, least power utilization is accomplished if the MOSFETs operate in weak inversion region, however at the expense of less speed. The maximum speed can be achieved if it is in strong inversion region, at the cost of more power. So, the best compromise between these two performance metrics is achieved in moderate inversion region. A new design methodology that permits an efficient design in all regions of operation of the MOS transistor is g_m/I_D methodology[5].

The choice of g_m/I_D ratio (transconductance generation efficiency) is in light of its importance for the accompanying reasons:

1. It is firmly related to the performances of analog circuits.
2. It provides an evidence of the device operating region.
3. It gives a tool for calculating the dimensions of transistors.

The figures of merit for device characterization are g_m/I_D (larger gm is required for as little current as possible) and f_T (larger transit frequency is required to get high speed). The product g_m/I_D * f_T peaks in moderate inversion, where the design will achieve optimal performance in terms of speed, area and power efficiency. The variation g_m/I_D , f_T and their product with respect to the over-drive voltage (V_{OV}) is as shown in Figure 4.1 [7]



(a) (b)

Figure 4.1: (a) Design Trade-offs: g_m/I_D and f_T (b) Product of g_m/I_D and f_T

4.2 Proposed Design Methodology

In this section, the proposed design methodology for the realization of basic building blocks of analog circuits is described. Figure 4.2 shows the flowchart for the proposed design methodology. Initially, specifications, like gain, bandwidth, phase margin, offset voltage etc. are taken from the user. Depending upon them, the proper building block topology is selected. For example, if the gain of an amplifier is given as 80dB, it is difficult to achieve this when we design a single-stage differential amplifier. Hence, we go for the two-stage differential amplifier for that specification. The next step is to perform small-signal analysis and derive the design equations in terms of transconductance, output conductance, and intrinsic junction capacitances. The design equations are used only to get the logical relation between the parameters, but not the actual values. This makes the design tool independent of PDK and technology library used. Then, using the g_m/I_D methodology and the designers' experience, the transconductance, output conductance, DC bias voltages' ranges are estimated. From this, a flow is developed to obtain

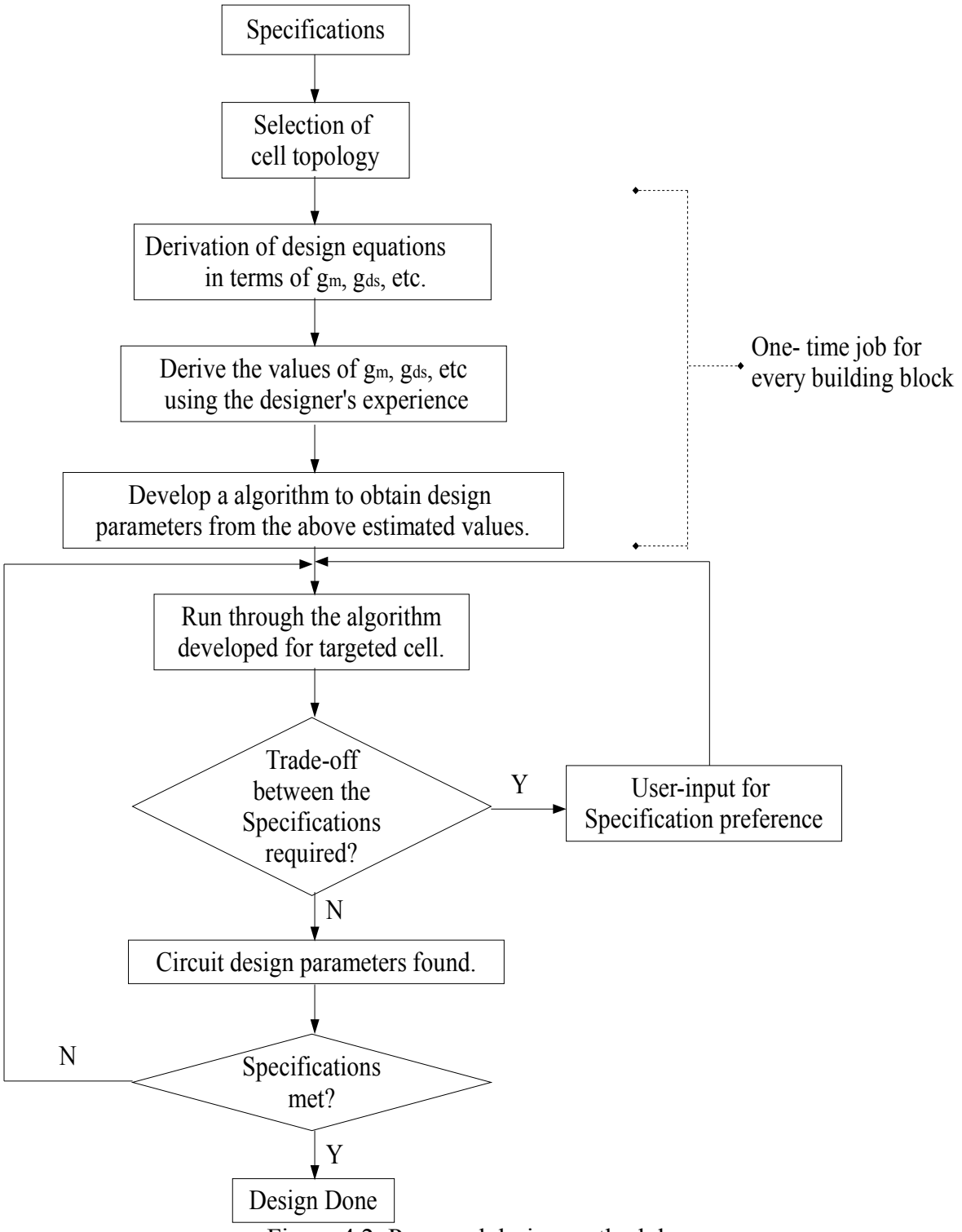


Figure 4.2: Proposed design methodology

the design parameter ranges i.e., bias voltages, W, L from the above-estimated values. This is the crucial step of design methodology. All the extreme conditions of input specifications and the output design parameters' range are to be considered and a decision has to be taken about what to do at each possible extreme. Look-up tables generated for design variables i.e., g_m, g_{ds}, intrinsic capacitance etc., with change in design parameters i.e., bias voltages and currents, W and L are used to go through the flow developed above. These steps have to be done once for every building block whose design is to be automated.

To automate the design of a building block for which design flow is already achieved, the MOSFET device models using which we are designing the building block are also taken as input from the user. Then, these devices are characterized to get the look-up tables required. Then, the algorithm developed above is made to run manually (which can be automated by developing a computer program accordingly). Intermediate simulations may be required in this process, in order to get the actual values obtained for Specifications like PSRR, for which the exact value can't be calculated from the expressions of it. The design parameters are to be found in this process. These design parameters are used in the design of building block and is checked whether the Specifications are met or not. If not, the design flow is continued to run. In some cases, it is difficult to achieve all the given specifications. In that cases, the trade-offs among the specifications are required. The user can select which specification is to be preferred and the flow will continue accordingly. The structure of the automation tool, after developing the algorithm is as shown in Figure 4.3

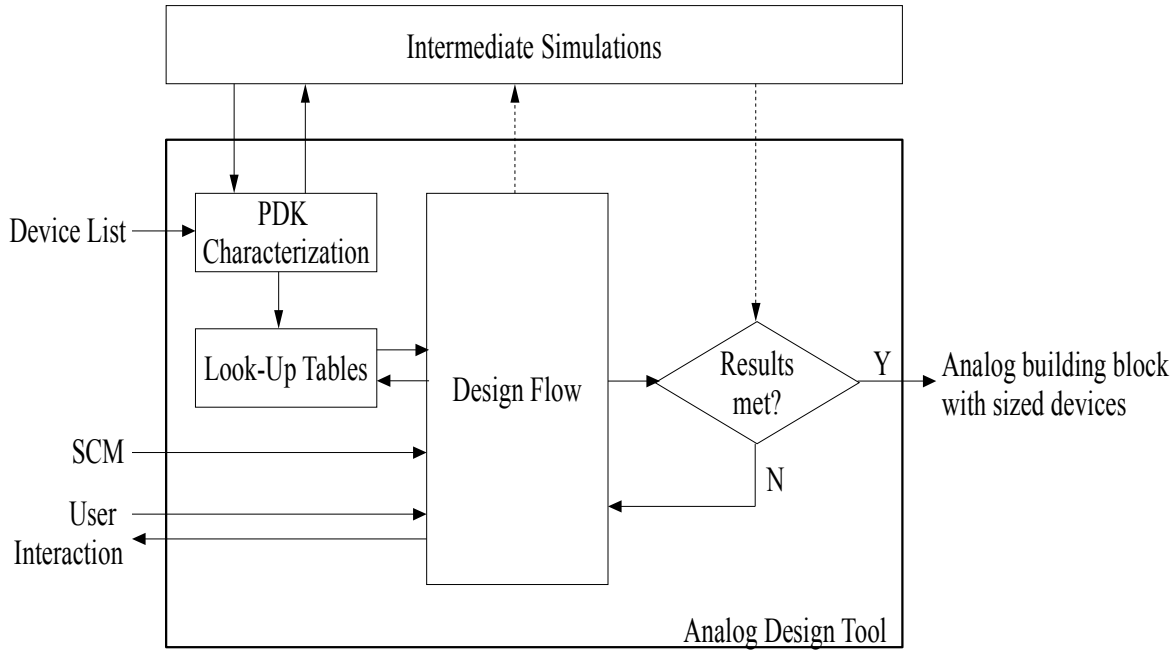


Figure 4.3: Structure of Automation Tool

4.3 Advantages of this Methodology

The design is basically right on target. Typical variances are on the order of 10-20%, for the most part due to V_{DS} condition, finite output resistance and so forth. These issues have been resolved by using pre-computed spice data (look-up tables) in the design process. Regardless of possibility that these issues are significant, there is always the possibility to find their causes.

The design process is independent of PDK or technology library used. This is because we are using spice data to get the relation between parameters.

4.4 Limitations of this Methodology

The look-up tables generation is to be generated for every device for every technology

used, which may be a bit cumbersome. But once the data is generated, it can be used for any circuit design.

4.5 Need for Look-Up Tables

Square Law model is inefficient for design in short-channel CMOS. But simulation models like PSP, BSIM are too complex for hand calculations. This issue has a tendency to drive many of designers towards a methodology where there are no hand calculations and the circuit is iterated in SPICE until it somehow meets the specifications. Along these lines, the main goal of this design methodology is to work without using a set of MOSFET design equations. So, the procedure used is to use look-up tables for designing.

5 PDK CHARACTERIZATION TOOL

5.1 Introduction

Characterizing a device is important to know the inter-relation between the parameters defining the device. There is a huge dependency on the device technology used for getting the device parameters like threshold voltage (V_{th}), transconductance (g_m), device capacitance (C_d), and so forth. These values vary because of short channel effects, fabrication issues etc. So predicting values for it would be rather difficult. Though a design can be made based on some predicted nearby values, the design may not be in acceptable tolerance for the given specification. These discrepancies are increasing as the device technology length decreases, because of increasing short-channel effects. These can be taken care to a great extent if a simulator tool is used, which will consider level 2 (or more) design equations. One such tool is Cadence Spectre simulator. So, for a given device, if a test-circuit is created, simulated and the required device parameters are extracted for varying device sizes, bias voltages and currents, a circuit design can be made easily and efficiently. In this thesis, an attempt is made to design a tool, which can characterize devices like MOSFET, Resistors, and Capacitors across a set of PDK technologies.

5.2 MOSFET Characterization

5.2.1 Device Parameters

For an MOSFET device, the main parameters which will vary with varying device sizes, bias currents and voltages are described in detail below.

5.2.1.1 Threshold Voltage (V_{th})

It is the minimum voltage required between gate and source terminals so that a conducting channel can be made between source and drain regions. As the channel length increases, this voltage required increases as it has to attract more minority charge carriers to the channel area. This variation of threshold voltage with respect to channel length for an NMOS device is as shown in Figure 5.1

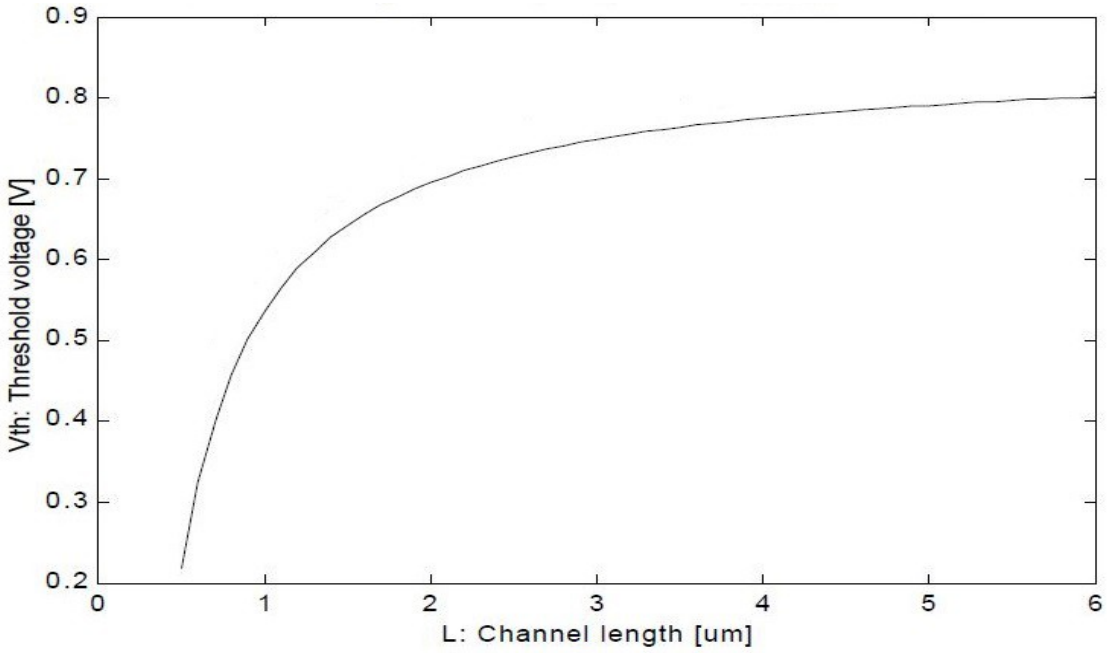


Figure 5.1: Threshold voltage vs Channel Length

As the source-to-bulk voltage difference (V_{BS}) changes, there is an effect on depletion layer width, which inturn reflects on gate oxide charge and results in the variation of threshold voltage (V_{th}). Thus, the difference observed in V_{th} equals the difference in depletion layer charge divided by the gate-oxide capacitance. Figure 5.2 depicts the variation of threshold voltage with a change in V_{SB}.

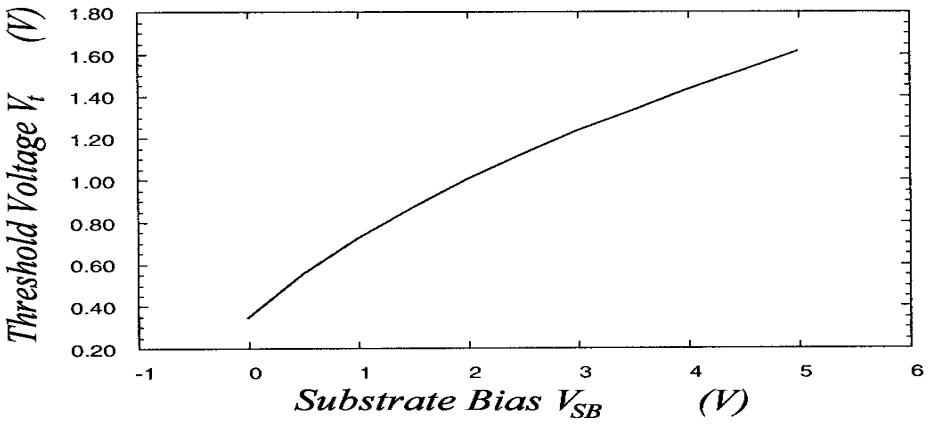


Figure 5.2: Threshold Voltage vs Substrate Bias

Though in long-channel devices there won't be much effect on threshold voltage by channel width, in narrow channel devices, narrow and short-channel effects results in a significant variation of it with channel width. This variation is depicted in Figure 5.3

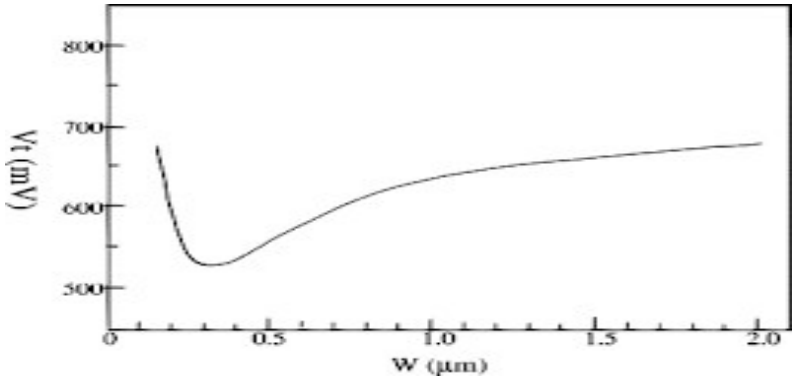


Figure 5.3: Threshold Voltage vs Channel Width

5.2.1.2 Over-drive Voltage (V_{OV})

It is defined as the gate-to-source voltage (V_{GS}) which is in excess to the threshold voltage (V_{th}), that controls the current conduction in the MOSFET. The drain current (I_D) is directly dependent on this voltage. For $V_{OV} < 0$, no current flows through it, as it is in cutoff region. For $V_{OV} >= 0$, the drain current increases linearly with the over-drive voltage until it goes to saturation region.

For constant bias currents and voltages, if the channel length is increased, according to the square-law model current equation as given in Equation 5.1, the over-drive voltage also increases in order to maintain constant drain current. For similar conditions, if the channel width is increased, V_{OV} decreases as it is inversely proportional to W. If for fixed MOSFET sizes, increase in drain current results in an increase in over-drive voltage. The term ($1+\lambda V_{DS}$) in the design equation represents channel length modulation effect. As the voltage across the drain and source V_{DS} is increased, for constant bias currents and device sizes, the over-drive voltage Vov decreases.

In saturation region, $I_D = \frac{\mu_n C_{ox}}{2} \frac{W}{L} V_{ov}^2 (1+\lambda V_{DS})$ where $V_{ov} = V_{GS} - V_{th}$ ..... [5.1]

5.2.1.3 Output Resistance (r_0)

Due to channel length modulation, the drain current is somewhat reliant on the drain-to-source voltage V_{DS}. Along with the current flow caused by gate-to-source voltage V_{GS}, there would be extra current flow because of V_{DS} as shown in Equation 5.1 This extra current is

considered to be flowing from an output resistor with resistance r_0, while considering small-signal model for the MOSFET. This resistance r_0 should be more in order to restrict this current flow and make MOSFET work fine. The mathematical expression for r_0 is as given in Equation 5.2

$$r_0 = \frac{1 + \lambda V_{DS}}{\lambda I_D} \quad \text{..... [5.2]}$$

where λ is the MOSFET device parameter and is an inverse of early voltage V_A.

For constant I_D and V_{DS}, if the channel length L is increased, channel length modulation factor λ decreases. This results in an increase in output resistance r_0. Similarly, an increase in drain-to-source voltage V_{DS} results in an increase in r_0, for fixed drain current and device sizes. If the bias current is increased by the external factors keeping all other factors constant, r_0 decreases.

5.2.1.4 Transconductance (g_m)

The transconductance of a device is defined as the ratio of current variation at the output to the voltage variation at the input. For an MOSFET device, it is the ratio of drain current to gate-to-source voltage. Mathematical expression for it is as shown in Equation 5.3

$$g_m = 2\sqrt{\frac{\mu_n C_{ox}}{2}\frac{W}{L} I_D} \quad \text{.... [5.3]}$$

From this equation, it is clear that transconductance varies with channel width, length and drain current. For fixed device sizes, g_m increases with the increase in the drain current I_D. Similarly, for fixed current, it increases with increase in aspect ratio W/L.

5.2.1.5 Unity Gain Bandwidth (f_T)

It is defined as the frequency of an MOSFET device where short-circuit current gain is equal to one. For the larger f_T value, the MOSFET gives better performance in the high-frequency domain. It is mathematically expressed as given in Equation 5.4

$$f_T = \frac{g_m}{2\pi(C_{gs}+C_{gd})} \quad \ldots [5.4]$$

where C_{gs} is the device capacitance between gate and source regions, C_{gd} is the gate-to-drain capacitance of MOSFET.

For constant bias current I_D, if the aspect ratio of the MOSFET is increased, both the transconductance g_m and device capacitances C_{gs} and C_{gd} are increased. Because of the square-root dependence of aspect ratio on g_m, the increase in g_m is less than the increase in device capacitances. Thus f_T decreases with increase in aspect ratio. In the same way, if the device size of MOSFET is fixed, the increase in bias current directly increases g_m and so is the increase in f_T.

5.2.1.6 Device Capacitances

In order to know the AC behavior of MOSFET, the device capacitances of it have to be considered. Between every two terminals of the device, capacitance exists and its value might depend on the bias conditions of the device. These capacitances include gate-oxide capacitance, overlap capacitance, depletion capacitance and junction capacitance, out of which the latest one mostly depends on the bias conditions and so is the topic of interest in this thesis. The junction capacitances which affect the MOSFET operation mostly are:

1. Capacitance between drain and all other terminals ($C_d = C_{dg}+C_{ds}+C_{db}$)

2. Capacitance between gate and source/drain regions ($C_{g1} = C_{gs}+C_{gd}$)
3. Gate-to-source capacitance ($C_{g2} = C_{gs}$)

These capacitance values vary with the device sizes and bias conditions. As the channel width or length is increased, the area of the device is increased which results in higher junction capacitance values. These capacitances are also voltage dependent because of depletion region charge present at the junction of source/drain regions and substrate. As the terminal voltages increases, the depletion charge across the reverse biased source/drain and substrate junction increases and so the junction capacitances increases.

5.2.1.7 Saturation Voltage (V_{dsat})

It is the minimum drain-to-source voltage required to keep the MOSFET in the saturation region. This voltage is determined by gate-to-source voltage V_{GS} and the MOSFET physical parameter V_{th} and is expressed mathematically as given in Equation 5.7. This expression holds good for long channel devices.

$$V_{dsat} = V_{GS} - V_{th} \quad \ldots. [5.7]$$

For short-channel devices, the distance between source and drain regions are much narrower. So this inturn creates a high electric field, which causes the maximum velocity of carriers even at lower V_{DS} and thus results in pinch-off. Thus in short-channel devices, saturation is obtained at lesser V_{DS} itself, in comparison with long-channel devices.

5.2.2 Test-circuit used

In order to characterize an MOSFET device and calculate their device parameters

aforementioned, the test-circuit used is as shown in Figure 5.4

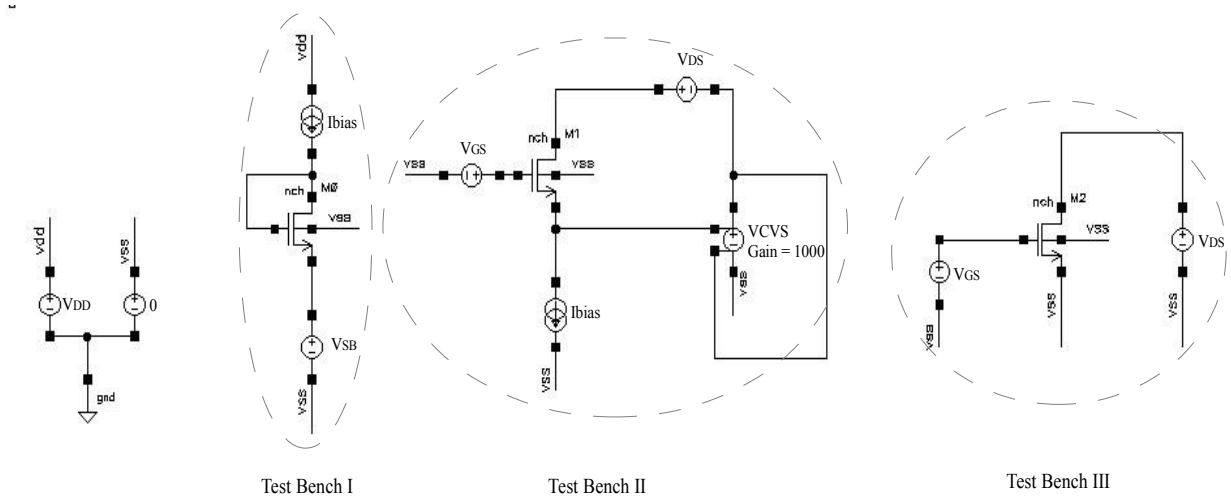


Figure 5.4: Test-circuit used to characterize n-channel MOSFET

In this template test-circuit, the MOSFET device model used is 'nch'. To characterize a different MOSFET device model, this 'nch' device is replaced with the corresponding device by the tool. This template can be used for any device model of any technology or PDK. This process of generating test-circuit automatically is made using SKILL scripting. In Test Bench I, the diode connected MOSFET is used and the variables varied are Ibias, V_{SB}, and W/L. These type of diode connected MOSFETs are generally used in Current Mirror circuits and so the parameters extracted from this test-circuit are used to automate such circuits. In Test Bench II, the gate-voltage of MOSFET device is fixed to $V_{DD}/2$ and the other variables i.e., I_{bias}, V_{DS}, and W/L are varied. This test-circuit is similar to a single-stage differential amplifier, where I_{bias} gives as the bias current, V_{DS} control circuit is used to fix the V_{DS} across the device in the same way as current-mirror pair in the amplifier. So the parameters extracted from this test-bench are used in automating the differential amplifier design. In Test Bench III, the current through the device is

not controlled and the terminal voltages V_{DS} and V_{GS} are varied along with the channel width W and length L. The parameters like junction capacitances depend only on terminal voltages and geometry and so this test-bench is useful to calculate such parameters of the device. Similar to the n-channel MOSFET test-circuits, the template test-circuit used for p-channel MOSFET devices is as shown in Figure 5.5. In this test-circuit, template device used is 'pch' which will be replaced with the required device by the tool.

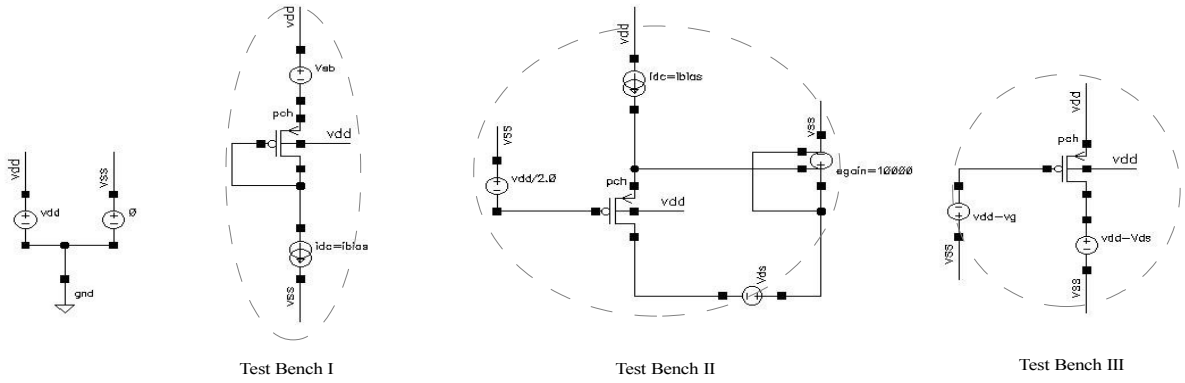


Figure 5.5: Test-circuit used for p-channel MOSFET characterization

5.2.3 Analysis Done

After the test-circuit is created by the tool, the next step to do is to perform analysis on that circuit. In order to do that, a netlist is generated for the generated test-circuit and a test-bench file tb.scs is written as per the requirements. To calculate the device parameters such as Vth, Vov, gm etc., DC analysis is required to be done on the given test-bench. AC analysis is to be done to extract parameters which are operation frequency related such as f_T. Thus, these two analysis are done at maximum and the required parameters are extracted. The parameters that are extracted from the given test-bench circuits are given in Table 5.1

Table 5.1: Device parameters for MOSFET device

Test Bench Used	Parameters Extracted	Sweep Variables
Test Bench I	Vth	VSB, W, L
	Vov	Ibias, W, L
	r0	Ibias, W, L
	gm	Ibias, W, L
Test Bench II	r0	Ibias, VDS, W, L
	gm	Ibias, VDS, W, L
	gm/ID	Ibias, VDS, W, L
	Vdsat	Ibias, VDS, W, L
	Vov	Ibias, VDS, W, L
Test Bench III	Cd = Cds + Cgd + Cdb	VGS, VDS, W, L
	Cg1 = Cgs + Cgd	VGS, VDS, W, L
	Cg2 = Cgs	VGS, VDS, W, L
	fT	VGS, VDS, W, L

The required parameters such as V_{th}, V_{OV}, g_m, V_{dsat}, r_0, C_{gs}, C_{sd}, etc. can be saved in the simulation output in order to avoid saving of all the DC parameters, including those which are not used and so some disk space can be saved. This kind of saving required data is useful when a large-simulations are made to run. The test-bench file written can be then made to execute and the required parameters can be extracted from the output .raw file. SHELL scripting is used to make the simulation execute automatically and OCEAN scripting is used for extracting data.

5.2.4 Result format

After the simulation is done, the parameters are extracted and written in the prescribed format. In the result files, all the variables, that are used to sweep in a test-bench, are mentioned even though the parameters won't sweep with them to make the format generalized. So, for all

the parameters that are extracted from Test Bench I, the sweep variables are I_{bias}, V_{SB}, W and L. Similarly, for Test Bench II parameters, I_{bias}, V_{DS}, W and L are sweep variables. For Test Bench III, V_{DS}, V_{GS}, W and L are sweep variables.

The look-up table for V_{th} generated from Test Bench I by the tool is as shown in Table 2. The simulation conditions taken for this analysis are:

- VDD = 1.8V
- VSS = 0V
- L varies from 0.5um to 5um
- W varies from 0.5um to 5um
- Ibias has values 1uA and 2uA
- VSB has values 0V ans 0.05V

The table will extend further horizontally and vertically for various V_{SB} and I_{bias} values respectively.

All the parameters extracted from Test Bench I will be in the same format as given in Table 5.2

Table 5.2: Look-Up table generated for Vth (All the values of Vth are in Volts)

Ibias = 1.000e-06 A
Vsb = 0.000e+00 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	7.56E-01	7.56E-01	...	7.57E-01	7.57E-01
1.00E-06	7.09E-01	7.09E-01	...	7.09E-01	7.09E-01
...	...	...	...	...	...
4.50E-06	6.70E-01	6.70E-01	...	6.70E-01	6.70E-01
5.00E-06	6.68E-01	6.68E-01	...	6.68E-01	6.68E-01

Ibias = 1.000e-06 A
Vsb = 5.000e-02 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	7.70E-01	7.70E-01	...	7.71E-01	7.71E-01
1.00E-06	7.23E-01	7.23E-01	...	7.24E-01	7.24E-01
...	...	...	...	...	...
4.50E-06	6.84E-01	6.84E-01	...	6.84E-01	6.84E-01
5.00E-06	6.83E-01	6.83E-01	...	6.83E-01	6.83E-01

Ibias = 2.000e-06 A
Vsb = 0.000e+00 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	7.56E-01	7.56E-01	...	7.57E-01	7.57E-01
1.00E-06	7.09E-01	7.09E-01	...	7.09E-01	7.09E-01
...	...	...	...	...	...
4.50E-06	6.69E-01	6.70E-01	...	6.70E-01	6.70E-01
5.00E-06	6.68E-01	6.68E-01	...	6.68E-01	6.68E-01

Ibias = 2.000e-06 A
Vsb = 5.000e-02 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	7.70E-01	7.70E-01	...	7.70E-01	7.70E-01
1.00E-06	7.23E-01	7.23E-01	...	7.24E-01	7.24E-01
...	...	...	...	...	...
4.50E-06	6.84E-01	6.84E-01	...	6.84E-01	6.84E-01
5.00E-06	6.83E-01	6.83E-01	...	6.83E-01	6.83E-01

Similarly, the look-up table for g_m generated from Test Bench II by the tool is as shown in Table 5.3 The simulation conditions taken for this analysis are:

- V_{DD} = 1.8V
- V_{SS} = 0V
- V_{GS} = 0.9V
- L varies from 0.5um to 5um
- W varies from 0.5um to 5um
- I_{bias} has values 1uA and 2uA
- V_{DS} has values 0.2V ans 0.25V

All the parameters extracted from Test Bench II will be in the same format as given in Table 5.3

Table 5.3: Look-Up table generated for gm (All the values of gm are in Siemens)

Ibias = 1.000e-06 A
VDS = 2.000e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	1.32E-05	1.63E-05	...	2.22E-05	2.25E-05
1.00E-06	9.79E-06	1.26E-05	...	1.90E-05	1.94E-05
...	...	...	...	...	...
4.50E-06	3.71E-06	6.37E-06	...	1.17E-05	1.21E-05
5.00E-06	3.37E-06	5.94E-06	...	1.14E-05	1.17E-05

Ibias = 1.000e-06 A
VDS = 2.500e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	1.33E-05	1.64E-05	...	2.22E-05	2.25E-05
1.00E-06	9.98E-06	1.26E-05	...	1.90E-05	1.94E-05
...	...	...	...	...	...
4.50E-06	4.45E-06	6.91E-06	...	1.18E-05	1.21E-05
5.00E-06	4.08E-06	6.55E-06	...	1.15E-05	1.18E-05

Ibias = 2.000e-06 A
VDS = 2.000e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	2.05E-05	2.64E-05	...	3.95E-05	4.03E-05
1.00E-06	1.39E-05	1.96E-05	...	3.20E-05	3.30E-05
...	...	...	...	...	...
4.50E-06	3.78E-06	7.42E-06	...	1.88E-05	1.96E-05
5.00E-06	3.39E-06	6.74E-06	...	1.81E-05	1.88E-05

Ibias = 2.000e-06 A
VDS = 2.500e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	2.09E-05	2.67E-05	...	3.96E-05	4.04E-05
1.00E-06	1.50E-05	2.00E-05	...	3.21E-05	3.31E-05
...	...	...	...	...	...
4.50E-06	4.73E-06	8.90E-06	...	1.92E-05	1.98E-05
5.00E-06	4.26E-06	8.17E-06	...	1.85E-05	1.91E-05

The look-up table for f_T generated from Test Bench III by the tool is as shown in Table 5.4 The simulation conditions taken for this analysis are:

- V_{DD} = 1.8V
- V_{SS} = 0V
- L varies from 0.5um to 5um
- W varies from 0.5um to 5um
- V_{GS} has values 0.2V and 0.25V
- V_{DS} has values 0.2V ans 0.25V

All the parameters extracted from Test Bench III will be in the same format as given in Table 5.4

Table 5.4: Look-Up table generated for fT (All the values of gm are in Hertz)

VGS = 2.000e-01 V
VDS = 2.000e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	2.15E+09	1.41E+09	...	4.84E+08	4.46E+08
1.00E-06	8.10E+08	5.54E+08	...	2.15E+08	2.00E+08
...	...	...	...	...	...
4.50E-06	6.01E+07	5.66E+07	...	2.76E+07	2.58E+07
5.00E-06	4.86E+07	4.69E+07	...	2.38E+07	2.24E+07

VGS = 2.000e-01 V
VDS = 2.500e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	2.19E+09	1.42E+09	...	4.86E+08	4.48E+08
1.00E-06	8.42E+08	5.64E+08	...	2.16E+08	2.01E+08
...	...	...	...	...	...
4.50E-06	7.64E+07	6.46E+07	...	2.81E+07	2.63E+07
5.00E-06	6.22E+07	5.45E+07	...	2.44E+07	2.28E+07

VGS = 2.500e-01 V
VDS = 2.000e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	3.15E+09	2.15E+09	...	8.12E+08	7.52E+08
1.00E-06	1.07E+09	8.10E+08	...	3.39E+08	3.17E+08
...	...	...	...	...	...
4.50E-06	5.72E+07	6.01E+07	...	4.10E+07	3.88E+07
5.00E-06	4.59E+07	4.86E+07	...	3.51E+07	3.33E+07

VGS = 2.500e-01 V
VDS = 2.500e-01 V

L (in m)/W (in m)	5.00E-07	1.00E-06	...	4.50E-06	5.00E-06
5.00E-07	3.28E+09	2.19E+09	...	8.17E+08	7.57E+08
1.00E-06	1.20E+09	8.42E+08	...	3.42E+08	3.20E+08
...	...	...	...	...	...
4.50E-06	7.41E+07	7.64E+07	...	4.28E+07	4.03E+07
5.00E-06	5.94E+07	6.22E+07	...	3.69E+07	3.48E+07

5.3 Resistance Characterization

A resistor is a two-terminal passive device which has the property of electrical resistance and so lowers the voltage levels within the circuit. A typical resistor has the structure of a block of resistive material from which two conductive metal terminals are connected on the either end to accommodate connection to external devices. So, the resistance value depends on the finger width and length. In order to avoid using large size resistors, a resistance can also be obtained by serially connecting a number of resistors, which will add up all the resistance values. The number of segments value in a resistor gives out the number of blocks to be used of given finger width and length to obtain a single resistance. Thus, the variables that affect the resistance value are Finger Length, Finger Width and Number of Segments. Characterizing a resistor device is necessary for the obtaining the geometry of it required to get a specific resistance value.

The test-circuit used for resistance characterization is as shown in Figure 5.6

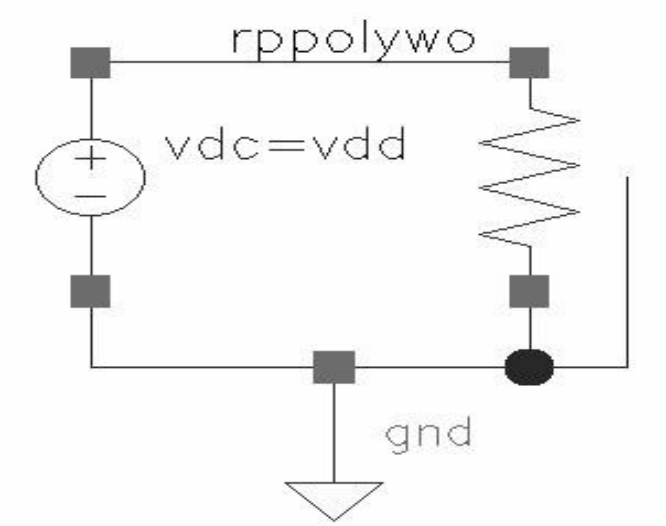


Figure 5.6: Test-circuit used for Resistance Characterization

A voltage source is applied across the resistor and its resistance value is extracted by calculating the ratio of the voltage applied and current flowing through it. DC analysis is enough to be done to extract these values. Thus, the tool initially generates the schematic design in Cadence Virtuoso Schematic Design Tool. Then it generates its netlist, writes the test-circuit file for performing DC analysis, performs a simulation and extracts the resistance value. The extracted data is written in the format of a look-up table as shown in Table 5.5 The devices characterized are from the 40nm technology with supply-voltage of 1.8V.

Table 5.5: Look-up table generated for Resistance Characterization

Resistor	No. of segments	Length	Width	Value (in Ohms)
rppolywo_m	1	5.00E-06	1.00E-06	3.18E+03
rppolywo_m	1	5.00E-06	2.00E-06	1.58E+03
rppolywo_m	1	1.00E-05	1.00E-06	6.36E+03
rppolywo_m	1	1.00E-05	2.00E-06	3.17E+03
rppolywo_m	2	5.00E-06	1.00E-06	6.37E+03
rppolywo_m	2	5.00E-06	2.00E-06	3.17E+03
rppolywo_m	2	1.00E-05	1.00E-06	1.27E+04
rppolywo_m	2	1.00E-05	2.00E-06	6.34E+03
rppoly	1	5.00E-06	1.00E-06	1.05E+02
rppoly	1	5.00E-06	2.00E-06	5.28E+01
rppoly	1	1.00E-05	1.00E-06	1.58E+02
rppoly	1	1.00E-05	2.00E-06	8.62E+01
rppoly	2	5.00E-06	1.00E-06	1.58E+02
rppoly	2	5.00E-06	2.00E-06	8.62E+01
rppoly	2	1.00E-05	1.00E-06	2.89E+02
rppoly	2	1.00E-05	2.00E-06	1.49E+02

The table extends further for more sweeps of Finger length, width and number of segments for more resistor types.

5.4 Capacitance characterization

A capacitor is a two-terminal passive device used to store the electrostatic energy in an electric field. It's typical structure has two conducting plates separated by a dielectric material. The capacitance value depends on the area of the used conducting plates and the area in between them. For some capacitor models, the capacitance value is split to a number of horizontal fingers and vertical fingers, in order to reduce the structure size to be used. Thus, the variables that affect the capacitance value are finger Width, finger Space, Number of horizontal fingers, Number of

vertical fingers, finger Length, multipliers, the material used etc.

The test-circuit used for capacitance characterization is as shown in Figure 5.7

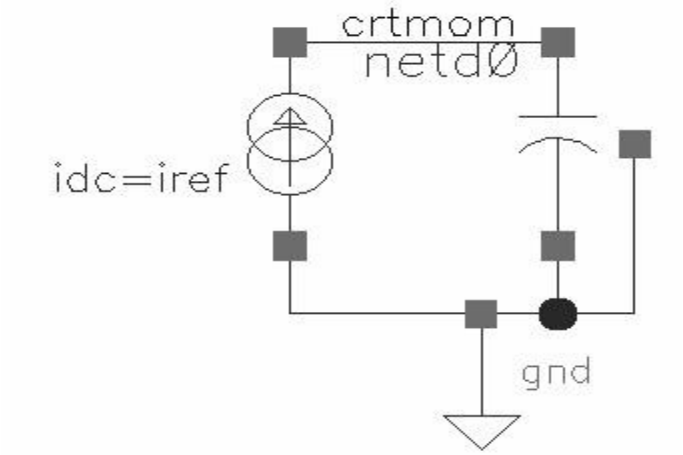


Figure 5.7: Test-circuit used for Capacitance characterization

A current source is connected in series with the capacitor to be characterized. DC analysis is done on this circuit in order to get the rate of change of voltage across the capacitor. The capacitance value is calculated as the ratio of current flowing through the capacitor and the rate of change of voltage across it, which is shown in Equation 5.8

$$C = \frac{I}{\partial V / \partial t} \quad \ldots. [5.8]$$

The netlisting, test-bench file generation, simulation run and output extraction are done. The number of sweep variables to be used for characterization is limited to a maximum of 4, whereas the user can select the sweep variables to be used and fix the other variables to default values. The extracted data is written in the format of a look-up table as shown in Table 5.6 The devices characterized are from the 40nm technology with bias current as 1uA. All the capacitance values in the table are in Farads.

Table 5.6: Look-up table generated for Capacitance Characterization

Capacitor	Finger Width (in m)	Finger Space (in m)	Horizantal_Fingers	Vertical_Fingers	Value
crtmom	7.00E-08	7.00E-08	32	32	4.12E-14
crtmom	7.00E-08	7.00E-08	16	32	2.06E-14
crtmom	7.00E-08	7.00E-08	16	16	1.02E-14
crtmom	7.00E-08	1.40E-07	32	32	5.27E-14
crtmom	7.00E-08	1.40E-07	16	32	2.64E-14
crtmom	7.00E-08	1.40E-07	16	16	1.32E-14
crtmom	1.40E-07	7.00E-08	32	32	5.79E-14
crtmom	1.40E-07	7.00E-08	16	32	2.90E-14
crtmom	1.40E-07	7.00E-08	16	16	1.45E-14
crtmom	1.40E-07	1.40E-07	32	32	4.56E-14
crtmom	1.40E-07	1.40E-07	16	32	2.28E-14
crtmom	1.40E-07	1.40E-07	16	16	1.14E-14
crtmom_rf	7.00E-08	7.00E-08	32	32	1.07E-08
crtmom_rf	7.00E-08	7.00E-08	16	32	5.36E-09
crtmom_rf	7.00E-08	7.00E-08	16	16	2.68E-09
crtmom_rf	7.00E-08	1.40E-07	32	32	4.13E-09
crtmom_rf	7.00E-08	1.40E-07	16	32	2.07E-09
crtmom_rf	7.00E-08	1.40E-07	16	16	1.02E-09
crtmom_rf	1.40E-07	7.00E-08	32	32	2.28E-09
crtmom_rf	1.40E-07	7.00E-08	16	32	1.14E-09
crtmom_rf	1.40E-07	7.00E-08	16	16	5.68E-10
crtmom_rf	1.40E-07	1.40E-07	32	32	3.14E-09
crtmom_rf	1.40E-07	1.40E-07	16	32	1.57E-09
crtmom_rf	1.40E-07	1.40E-07	16	16	7.84E-10

Along with the look-up tables generated, a feature to plot the graphs is also added to the tool. The GUI of the tool designed and the sample graphs are given in Appendix.

6 DESIGN EXAMPLES

6.1 Introduction

This chapter presents the design of analog IC building blocks using new design methodology. The schematics of the building blocks along with the design procedures developed are presented. The design parameters obtained from the tool are tested with Cadence Spectre simulation to prove the accuracy and reliability of the developed tool.

6.2 Designing Basic Current-Mirror circuit

Current-Mirrors are the important building blocks in the modern analog IC design, where the signal is processed primarily as a current rather than voltage. The basic current mirror circuit containing two MOSFET devices is as shown in Figure 6.1

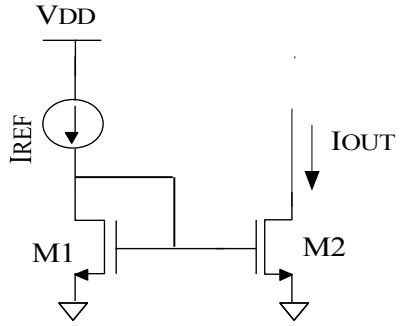


Figure 6.1: Basic Current Mirror Circuit

Transistor M1 is always in the saturation region, as its gate and drain terminals are connected together. Assuming drain-source voltage of M2 is large enough to keep it in saturation, we can write the relation between the drain currents as

$$\frac{I_{OUT}}{I_{REF}} = \frac{I_{d2}}{I_{d1}} = \frac{g_{m2}}{g_{m1}} \quad \text{.... [6.1]}$$

If the channel length modulation is neglected and both the transistors are assumed to be of same type, Equation 6.1 can be further resolved as

$$\frac{I_{OUT}}{I_{REF}} = \frac{(W/L)_2}{(W/L)_1} \quad \text{.... [6.2]}$$

This ratio is used to obtain relative amplification or attenuation. The over-drive voltage for both MOSFETs is assumed to be 120mV, as the square-law prediction is fairly close to practical values because of physical effects being negligible, for a minimum of 120mV over-drive voltage.

To design the basic current mirror circuit, the inputs taken from the user are IREF to be and IOUT required, along with the MOS device to be used. The design flow developed for this building block is shown as flowchart in Figure 6.2

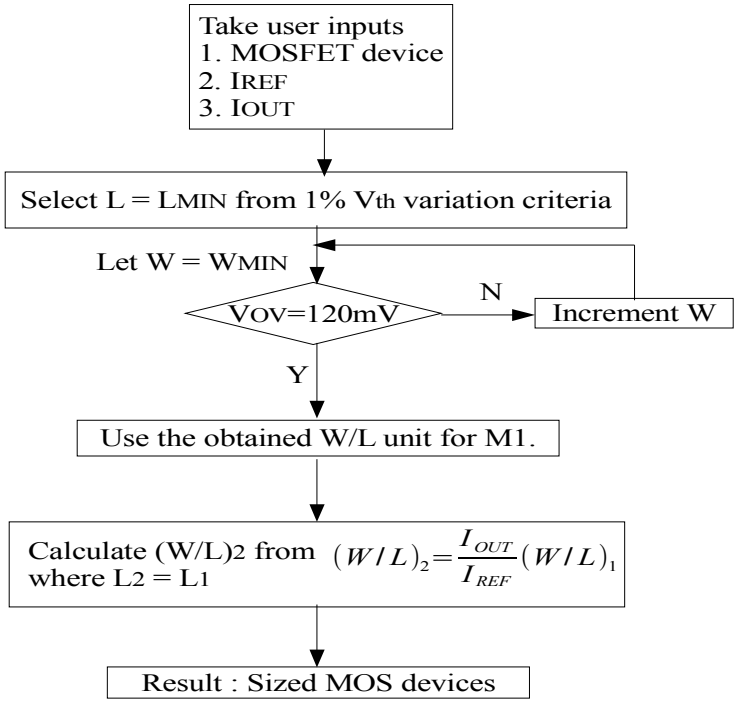


Figure 6.2: Design Flow for Basic Current Mirror Circuit

The look-up tables required for this flow are:

1. V_{th} vs L – This is used to select L_{MIN} from 1% V_{th} variation criteria. The 1% V_{th} variation criteria is considered because the Vth vs L plot varies drastically at very low L values and will get stabilized after a certain L. That L value is what we are considering as minimum L for the design. It would be generally near 1.2um to 1.5um.
2. Vov vs I, W, L – This is used to select W_1 in order to get Vov = 120mV.

As a design example, suppose the user provides the following input data:

MOS Device to be used – nch_25ud18,

I_{OUT} required – 6uA

I_{REF} to be given – 2uA

From V_{th} vs L look-up table, Lmin is observed to be 1.5um

Table 6.1: Basic current mirror circuit design flow step I

ibias = 2.000e-06			
Vsb = 0.000e+00			
l/w	5.00E-007	1.00E-006	1.50E-006
5.00E-007	7.56E-001	7.56E-001	7.56E-001
1.00E-006	7.09E-001	7.09E-001	7.09E-001
1.50E-006	6.92E-001	6.92E-001	6.93E-001
2.00E-006	6.85E-001	6.85E-001	6.85E-001
2.50E-006	6.80E-001	6.80E-001	6.80E-001
3.00E-006	6.76E-001	6.76E-001	6.76E-001
3.50E-006	6.73E-001	6.73E-001	6.73E-001
4.00E-006	6.71E-001	6.71E-001	6.71E-001
4.50E-006	6.69E-001	6.70E-001	6.70E-001

Table 6.2: Basic Current mirror design flow step II

ibias = 2.000e-06				
Vsb = 0.000e+00				
l/w	5.00E-007	1.00E-006	1.50E-006	2.00E-006
5.00E-007	8.75E-002	3.03E-002	2.36E-003	-1.56E-002
1.00E-006	1.92E-001	1.14E-001	7.65E-002	5.28E-002
1.50E-006	2.58E-001	1.66E-001	1.22E-001	9.38E-002
2.00E-006	3.11E-001	2.05E-001	1.56E-001	1.25E-001
2.50E-006	3.55E-001	2.39E-001	1.85E-001	1.51E-001
3.00E-006	3.95E-001	2.68E-001	2.10E-001	1.74E-001
3.50E-006	4.30E-001	2.94E-001	2.32E-001	1.94E-001
4.00E-006	4.63E-001	3.18E-001	2.53E-001	2.12E-001
4.50E-006	4.94E-001	3.41E-001	2.71E-001	2.29E-001

From Vov vs I, W, L look-up table, W is achieved to be 1.5um

Thus, W_1/L_1 = 1.5um/1.5um

As required I_{OUT} = 6uA i.e., 3 times the I_{REF}, W_2/L_2 = 4.5um/1.5um.

Therefore, the sized MOSFET devices are obtained. When checked using Virtuoso Spectre simulator, the obtained circuit is as shown in Figure 6.3.

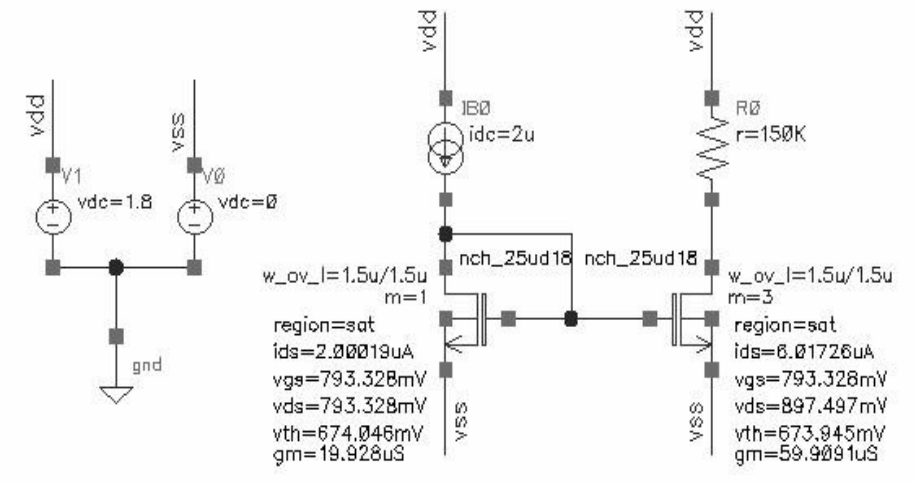


Figure 6.3: Design Example of Basic Current Mirror Circuit

6.3 Cascode Current-Mirror

The cascode current mirror circuit containing four MOSFET devices is as shown in Figure 6.4

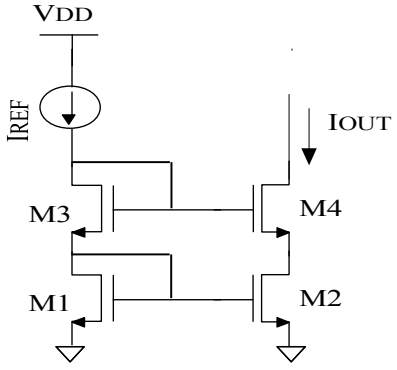


Figure 6.4: Cascode Current Mirror

The M1-M2 current-mirror pair is a simple current-mirror circuit, whose design flow is as explained before. As deduced in the previous section, Equation 6.2 (the ratio between the two currents) is applicable for this circuit also. Unless in the above case, where drain-source voltage V_{DS} of M2 may depend on load without affecting the current-mirror balance, here it should be a fixed voltage in order to balance the other current-mirror pair M3-M4. So, we consider $V_{DS2} = V_{DS1}$. Now, in order to design M3-M4 current-mirror pair, the V_{OV3} is adjusted in such a way that $V_{DS3} = V_{DS4}$. So, V_{OV3} is calculated from

$$V_{DS3} + V_{DS1} = V_{DD} - I_{OUT,REQ} * |Z_{load}| \quad \ldots. [6.3]$$

Thus, the overall design flow developed for this block is shown as flowchart in Figure 6.5

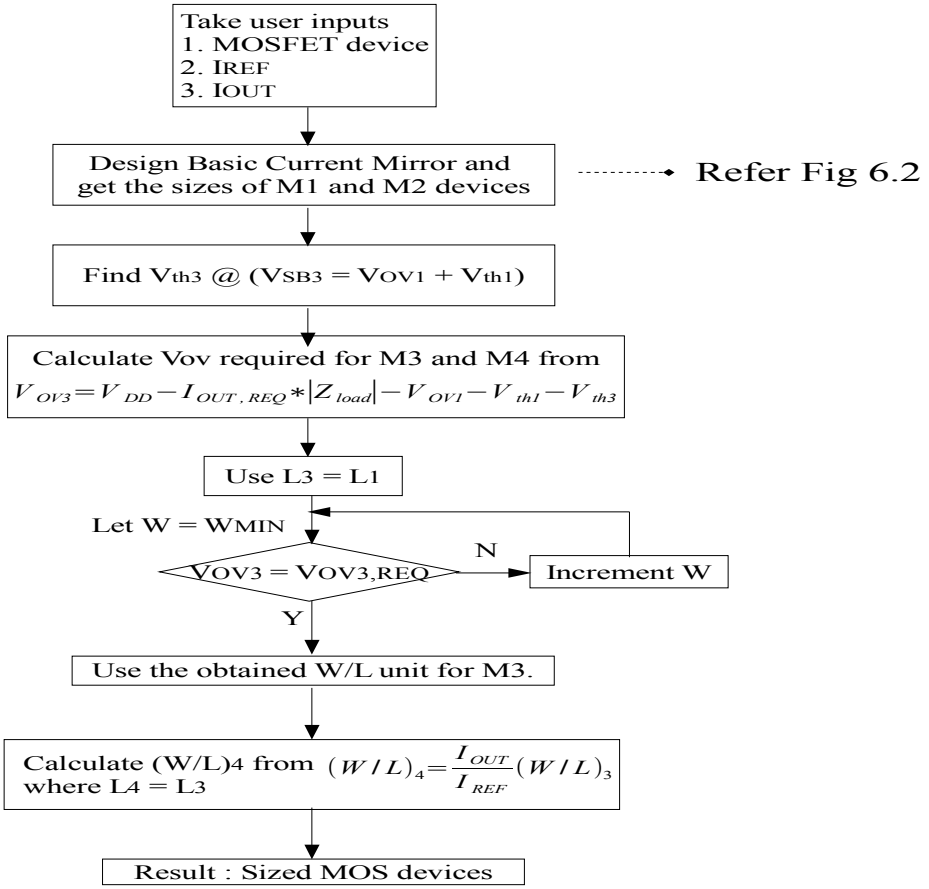


Figure 6.5: Design Flow for Cascode Current-Mirror circuit

The look-up tables required for this flow are:

1. V_{th} vs L, V_{SB}

2. Vov vs I, W, L

This design flow is tested with the given specifications and is tested with Cadence Virtuoso. The results are as shown in Table 6.3

Specifications:

Parameter	Value
I_{ref}	2uA
I_{OUT}	6uA
V_{DD}	1.8V
V_{SS}	0
Z_{load}	1K Ohm

Design Parameter Values:

Parameter	Value obtained from the tool	Value obtained from the Cadence
V_{OV1}	122mV	119.867mV
W1/L1	1.5um/1.5um	1.5um/1.5um
W2/L2	1.5um/1.5um	1.5um/1.5um
V_{th1}	693mV	674.046mV
V_{th3}	904mV	875.512mV
V_{OV3}	77mV	67.815mV
W3/L3	2.5um/1.5um	2.5um/1.5um
W4/L4	2.5um/1.5um	2.5um/1.5um
I_{ref}	2uA	2uA
I_{OUT}	6uA	6.00399uA

Table 6.3: Obtained results for Cascode Current Mirror

6.4 Low Voltage Cascode Current Mirror

The low voltage cascode current mirror circuit containing four MOSFET devices and a resistor is as shown in Figure 6.6

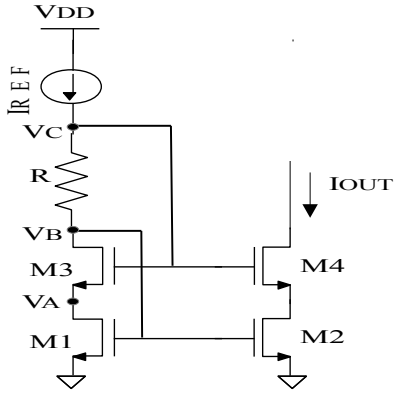


Figure 6.6: Low Voltage Cascode Current Mirror circuit

To keep M1 and M2 in saturation, the terminal voltage relations are as follows

$$V_B \geq V_{th1} \Rightarrow V_B = V_{th1} + 120\text{mV} \quad \text{…. [6.4]}$$

$$V_A \geq V_{OV1} \Rightarrow V_A = V_{OV1} + 120\text{mV} \Rightarrow V_A = 240\text{mV} \quad \text{…. [6.5]}$$

To keep M3 and M4 in saturation, the terminal voltage relations are as follows

$$V_C \geq V_A + V_{th3} \Rightarrow V_C \geq 240\text{mV} + V_{th3} \Rightarrow V_C = 240\text{mV} + V_{th3} + V_{OV3} \quad \text{…. [6.6]}$$

$$V_B \geq V_C - V_{th3} \Rightarrow V_B \geq V_{OV3} + 240\text{mV} \Rightarrow V_B = V_{OV3} + 240\text{mV} + 120\text{mV} \quad \text{…. [6.7]}$$

From Equation 6.4 and 6.7,

$$V_{OV3,\,max} = V_{th1} - 240\text{mV}$$

The resistor value is calculated as

$$R = \frac{V_C - V_B}{I_{Ref}} = \frac{V_{th3} - 120\text{mV}}{I_{Ref}}$$

The overall design flow developed for this block is as follows

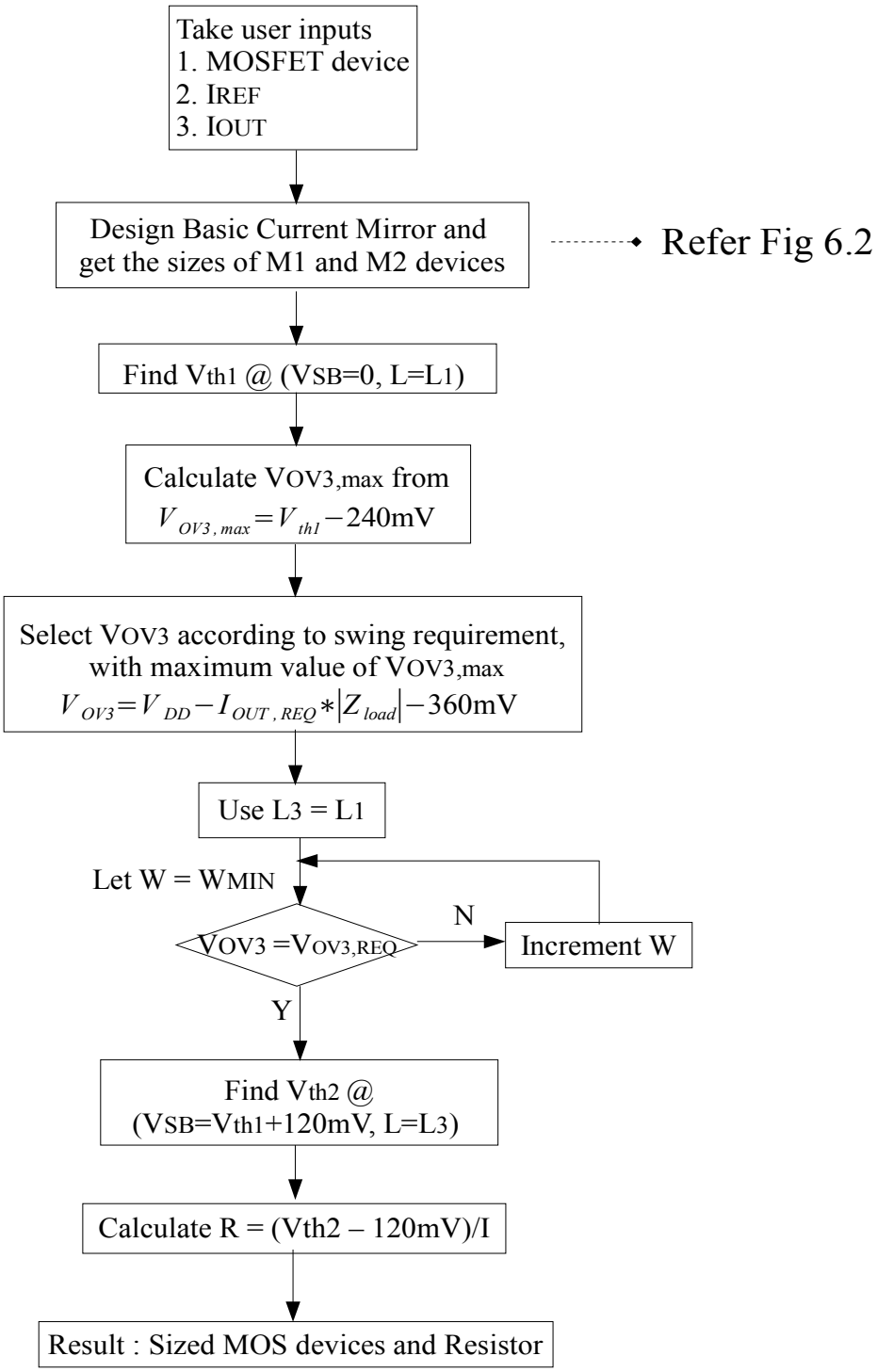


Figure 6.7: Design Flow for Low voltage Cascode Current Mirror circuit

The look-up tables required for this flow are:

1. Vth vs L, V_{SB}
2. Vov vs I, W, L

This design flow is tested with the given specifications and is tested with Cadence Virtuoso. The results are as shown in Table 6.4

Table 6.4: Obtained results for Low Voltage Cascode Current Mirror

Specifications:

Parameter	Value
I_{ref}	2uA
I_{OUT}	6uA
V_{DD}	1.8V
V_{SS}	0
Z_{load}	164.6K Ohm

Design Parameter Values:

Parameter	Value obtained from the tool	Value obtained from the Cadence
V_{OV1}	122mV	119.867mV
W1/L1	1.5um/1.5um	1.5um/1.5um
W2/L2	1.5um/1.5um	1.5um/1.5um
V_{th1}	693mV	674.046mV
V_{OV3}	452mV	459.238mV
W3/L3	2.5um/1.5um	2.5um/1.5um
W4/L4	2.5um/1.5um	2.5um/1.5um
V_{th3}	741mV	734.432mV
R	310.5K Ohm	310.5K Ohm
I_{ref}	2uA	2uA
I_{OUT}	6uA	5.99934uA

6.5 Single-stage Differential Amplifier circuit

The differential amplifiers are one of the most important cells in today's high-performance analog circuits. The circuit diagram for the single-stage differential amplifier is as shown in Figure 6.8

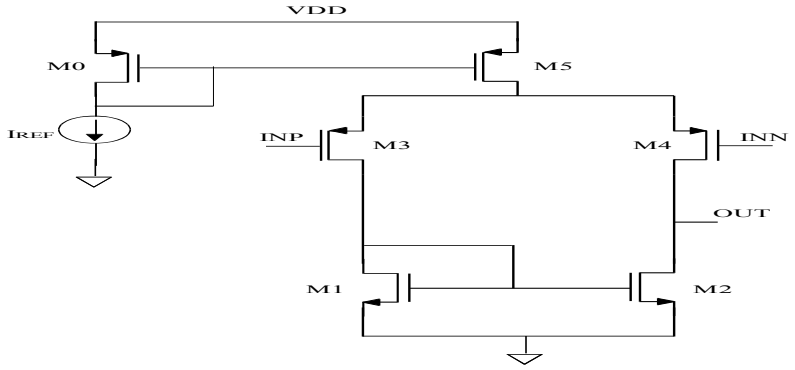


Figure 6.8: Single stage differential amplifier

In this circuit, M0 and M5 devices serve the bias current needed. M1 and M2 forms the current-mirror circuit used to balance the current in both the paths. M3 and M4 are the actual devices which do amplification for the input signals given. The output observed would be in in-phase with the input given at the terminal INP and in out-phase with the input at INN. The design methodology designed for automating the design of this circuit is as shown in Figure 6.9

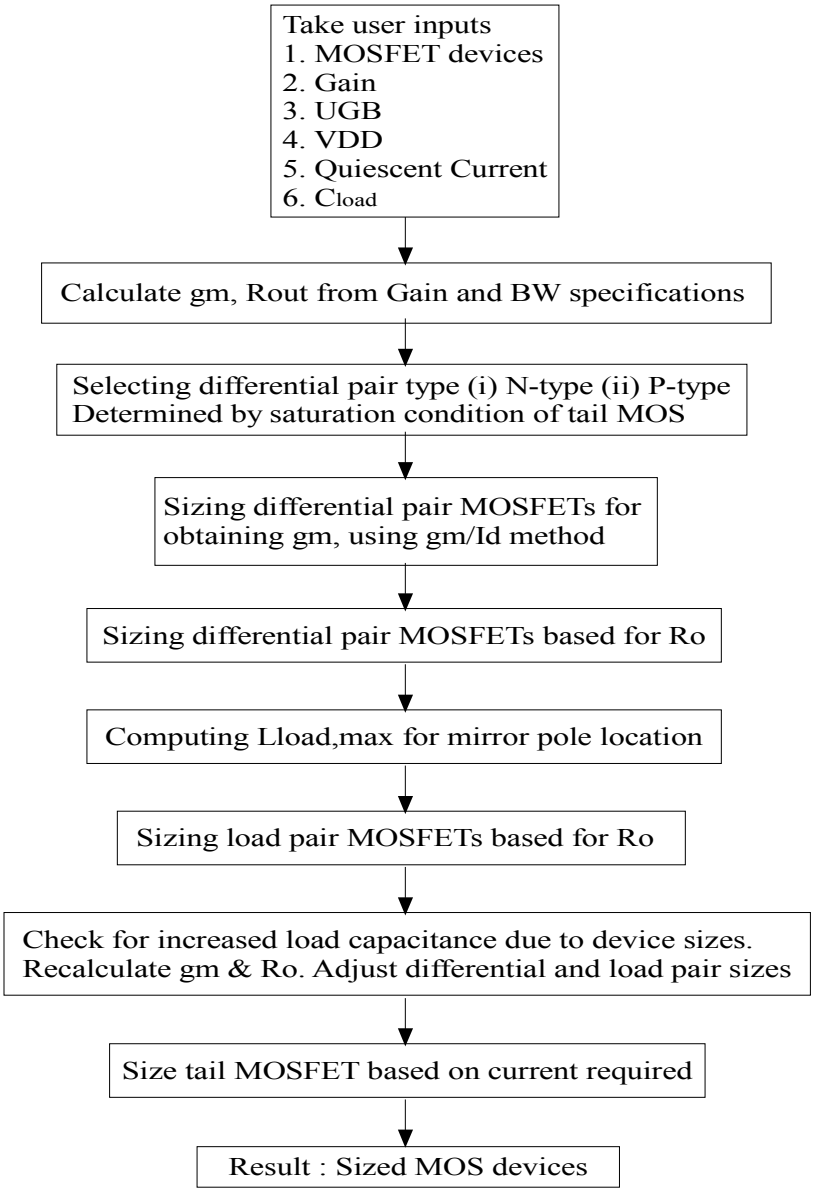


Figure 6.9: Design flow for Single-stage differential amplifier

In order to design a single-stage differential amplifier circuit, the analytic design equations, which can foresee adequately precise circuit behavior, so as to relate specifications to proper circuit design variables, are used. These equations for single-stage differential amplifier are given as:

$$Gain = g_m * (R_o, diff \| R_o, load) \quad \text{..... [6.1]}$$

$$UGB = \frac{g_m}{2\pi C_{load}} \quad \text{..... [6.2]}$$

The required values for gm and Ro are deduced from these equations. Depending on the saturation conditions of tail MOSFET, the differential pair to be used is selected. If both the types are valid to use, the one which has high $g_m/I_D * f_T$ value for maximum over-drive voltage is used. Then, the g_m required for differential pair MOSFETs is achieved by varying bias current (It should be within the limit of quiescent current given by the user.) or by sizing the MOSFETs. The channel length is varied in order to get the required $R_{o,diff}$, if is not sufficient then the gm is adjusted accordingly. The maximum channel length to be used for load pair MOSFETs is calculated, in order to keep mirror pole outside the UGB range. Using this, the load pair MOSFET is sized in order to get the required $R_{o,load}$. Till now, capacitance at the output terminal is considered to be as Cload itself. But the device capacitances also affect the output terminal and so affect the design parameters gm, Ro etc. Thus for the sized MOSFETs, these device capacitances are calculated and their effect on the output terminal is checked. The design parameters g_m and Ro are adjusted with accordingly to obtain required gain and bandwidth specifications. Then the tail MOSFET is sized for the required current, using the design flow for basic current-mirror circuit.

This design flow is checked for different specifications, and on different PDKs, as shown in Table 6.5

Table 6.5: Specification Compliance Matrix for Single-stage Differential Amplifier circuit

Parameter	Unit	Specification		
		Min	Typ	Max
VDD	V	1.68	1.8	1.92
VSS	V		0	
Temperature	°C		27	
ICMR	V	1	1.2	1.4
Ibias	uA	0.8	1	1.2
CL	fF		500	
Gain	dB	40		
		52		
UGB	Mhz	10		
		100		
IQ	uA			20 (for 10MHz)
				80 (for 100MHz)
PD Current	nA			10

The obtained results are tabulated as shown in Table 6.6

Table 6.6: Design examples for single-stage differential amplifier

Technology	SCM			RESULT				Differential Pair				Load Pair				Tail MOS			
	GAIN(dB)	UGB(MHz)	Iref (uA)	Min. GAIN(dB)	Max. GAIN(dB)	UGB(MHz)	IQ (uA)	W(um)	L(um)	Vov (mV)	Vdst (mV) = Vds - Vdsat	W(um)	L(um)	Vov (mV)	Vdst (in mV) = Vds - Vdsat	W(um)	L(um)	Vov (mV)	Vdst (in mV) = Vds - Vdsat
40nm	40	10	5	43	50	11.5	5	5	2	-43	186	7.5	1.4	53	609	2.8	1.2	119	132
40nm	45	Relaxed	5	48	57	23	40	16.8	5	-96	176	6	5	36	600	1.4	1.2	120	236
40nm	Relaxed	100	5	43	49	104	55	6	1	-15	176	3	0.8	91	628	5.6	1.2	135	180
28nm	40	10	3.5	40.2	44.4	11.8	7	3.75	5	87	128	5	2.2	-14	486	2.2	1	110	248
28nm	45	Relaxed	5	45	50.5	17	80	20	5	45	165	48	5	-42	451	1.6	1	108	270
28nm	Relaxed	100	5	28.6	34	101	105	15	0.8	162	130	11	0.9	40	535	2.5	1	141	175

This developed and tested design flow mainly focuses on specifications Gain and Bandwidth. If more specifications like PSRR, offset voltage etc. have to be taken care, then the design flow is further extended to get these specifications in such a way that already obtained

specifications won't be affected much. The flow developed is further extended for the specifications PSRR as described in this section.

Initially the small signal analysis is made for differential amplifier, to derive an expression for PSRR. The small signal model circuit for it is as shown in Figure 6.10

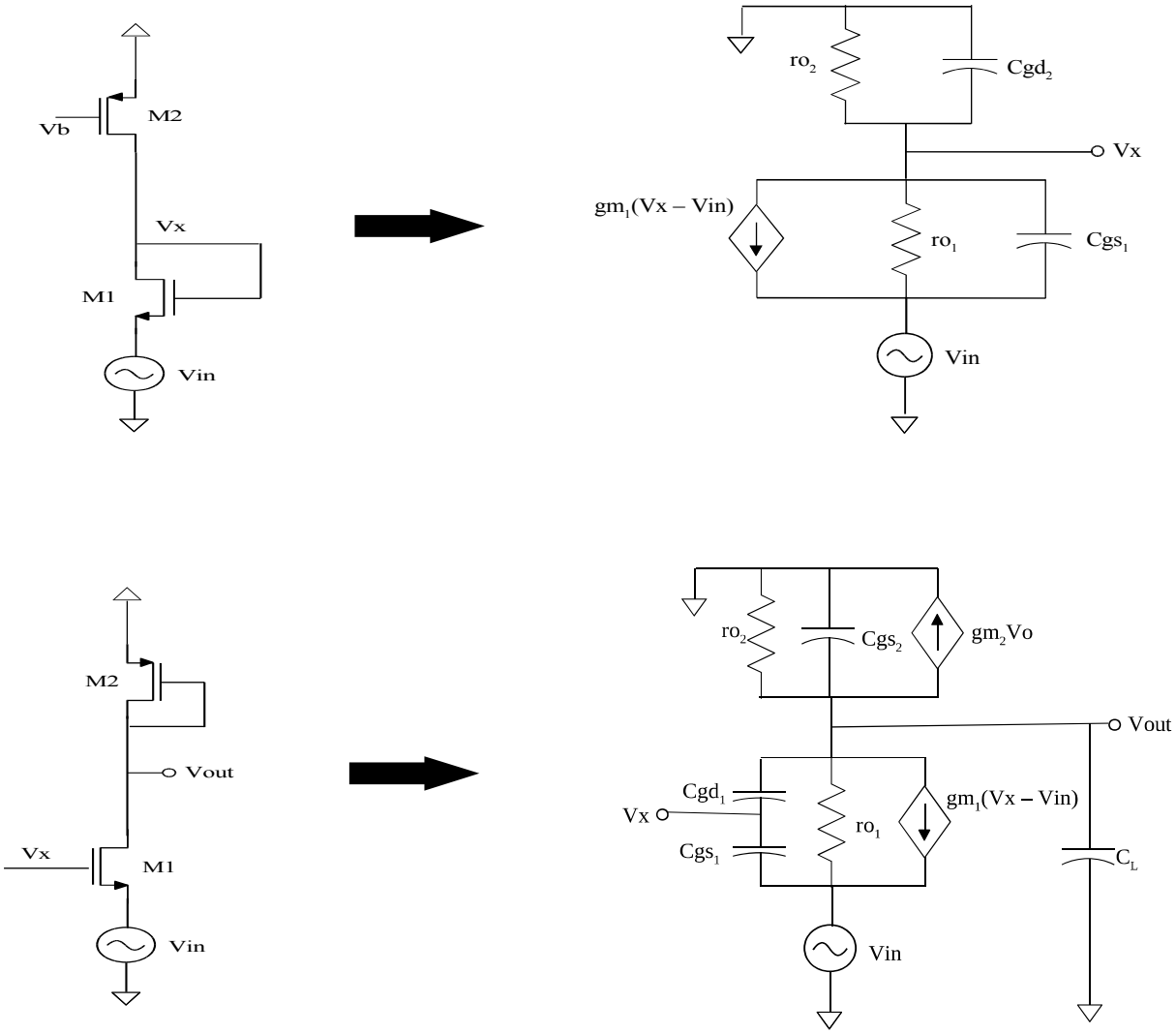


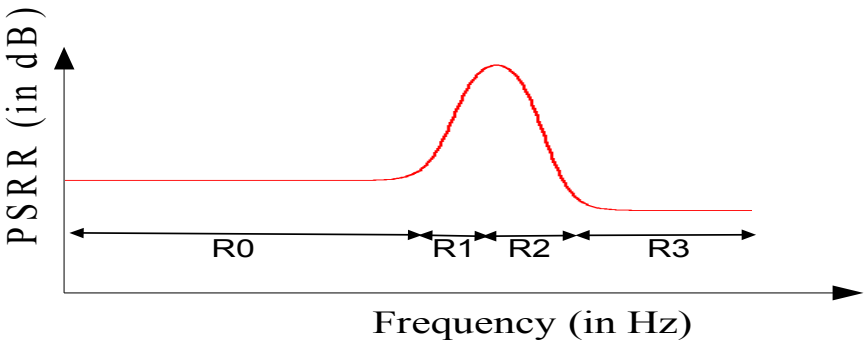
Figure 6.10: Small signal modelling for differential amplifier to calculate PSRR

From this analysis, the expression for output voltage Vo to input voltage Vi is derived as shown in equation 6.3

$$A_{dd}=\frac{V_o}{V_i}=\left[\frac{1}{1+g_{m2}.r_{o1}\|r_{o2}}\right]\frac{(1+s\,4r_{o1}\|r_{o2}C_{gd1}).(1+s\frac{C_{gd1}}{C_{gd2}}\frac{C_{gs1}}{g_{m1}})}{(1+s\frac{(C_L+C_{gd1}+C_{gs2})}{g_{m2}}).(1+s\frac{(C_{gd2}+C_{gs1})}{g_{m1}})} \quad \text{..... [6.3]}$$

This expression is shown in pole-zero format for A_{dd}. The PSRR is the inverse of this A_{dd} and so the postions of poles and zeroes for PSRR and the waveform for it can be estimated from these values. The methodology developed for PSRR is as described below.

1. Initially, calculate the Pole-Zero frequencies and magnitude of PSRR at the given frequency from the given equation.
2. Check on which frequency range the PSRR should be achieved from the specification given.



3. For Range R0, DC PSRR should be increased.
   - For that, increase both W and L of diff pair so that Vov remains constant and hence ro2 increases.
     - Make sure that L is less than Lmax because of mirror pole consideration.
     - Make sure that UGB is more than minimum UGB, as this variation causes UGB to decrease slightly.
   - If still the required PSRR is not achieved, increase W of diff pair so that gm2 increases.
     - Make sure that Vov does not go below Vov,min.
4. For Range R1,
   - Decrease W of load pair so that gm1 decreases.

- Make sure that UGB is more than minimum UGB, as this variation causes UGB to decrease slightly.

- If still it is not achieved, increase I by Current Multiplier of Bias Pair.
  - Make sure that I does not exceed IQ of given specification.
  - UGB increases slightly in this case.

5. For Peak at Poles and Range R2,

- Decrease W of load pair so that gm1 decreases.
  - Make sure that UGB is more than minimum UGB, as this variation causes UGB to decrease slightly.

6. For Range R3,

- Increase both W and L of diff pair so that Vov remains constant and hence ro2 increases.
  - Make sure that L is less than Lmax because of mirror pole consideration.
  - Make sure that UGB is more than minimum UGB, as this variation causes UGB to decrease slightly.
- If still the required PSRR is not achieved, increase W of diff pair.
  - Make sure that Vov does not go below Vov,min.

Thus, this design flow can be extended for more specifiations, without affecting much the already achieved ones. But it is not possible everytime to take care of all the specifications. Trade offs has to be made in that cases. Preference has to be made depending on the user requirement.

6.6 Two-stage differential amplifier circuit

In single-stage differential amplifier, the typical gain obtained is about 40dB. For amplifiers which require gain around 80dB, it would be difficult to design a single-stage circuit.

So, a two-stage differential amplifier is used in that cases, where a higher gain is required, but at the cost of decreased bandwidth. The circuit diagram for two-stage differential amplifier[4] is as shown in the Figure 6.11

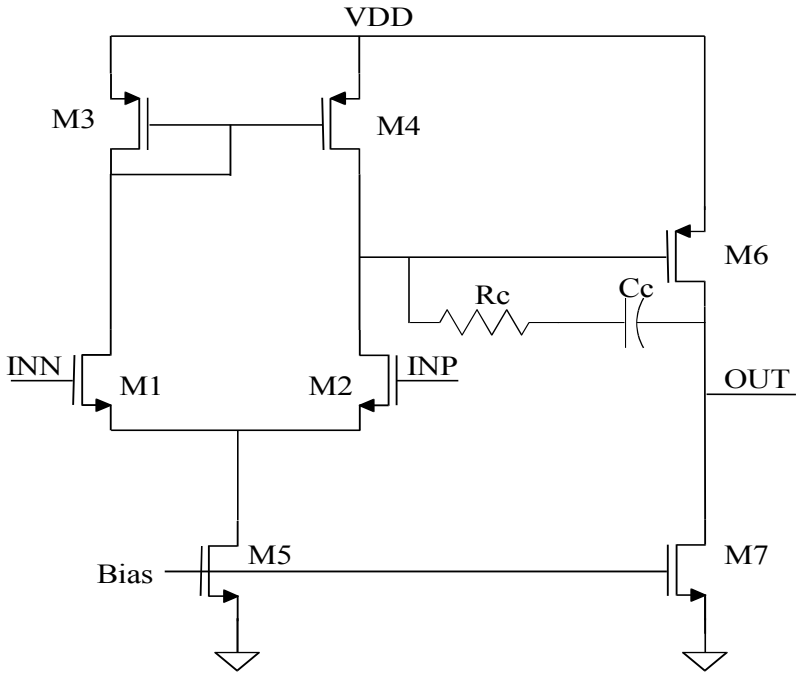


Figure 6.11: Two-stage differential amplifier circuit

From this figure, it is clear that the first stage of a two-stage differential amplifier is a simple single-stage amplifier. M6 device is added at the output of the first stage, which increases the gain of the whole circuit further. M7 gives the bias current required. RC compensation is used in this design in order to obtain maximum bandwidth, as the speed of two-stage differential amplifier is comparatively low. It is used also in order to increase the stability of the design. The design flow developed for this circuit is as shown in Figure 6.12

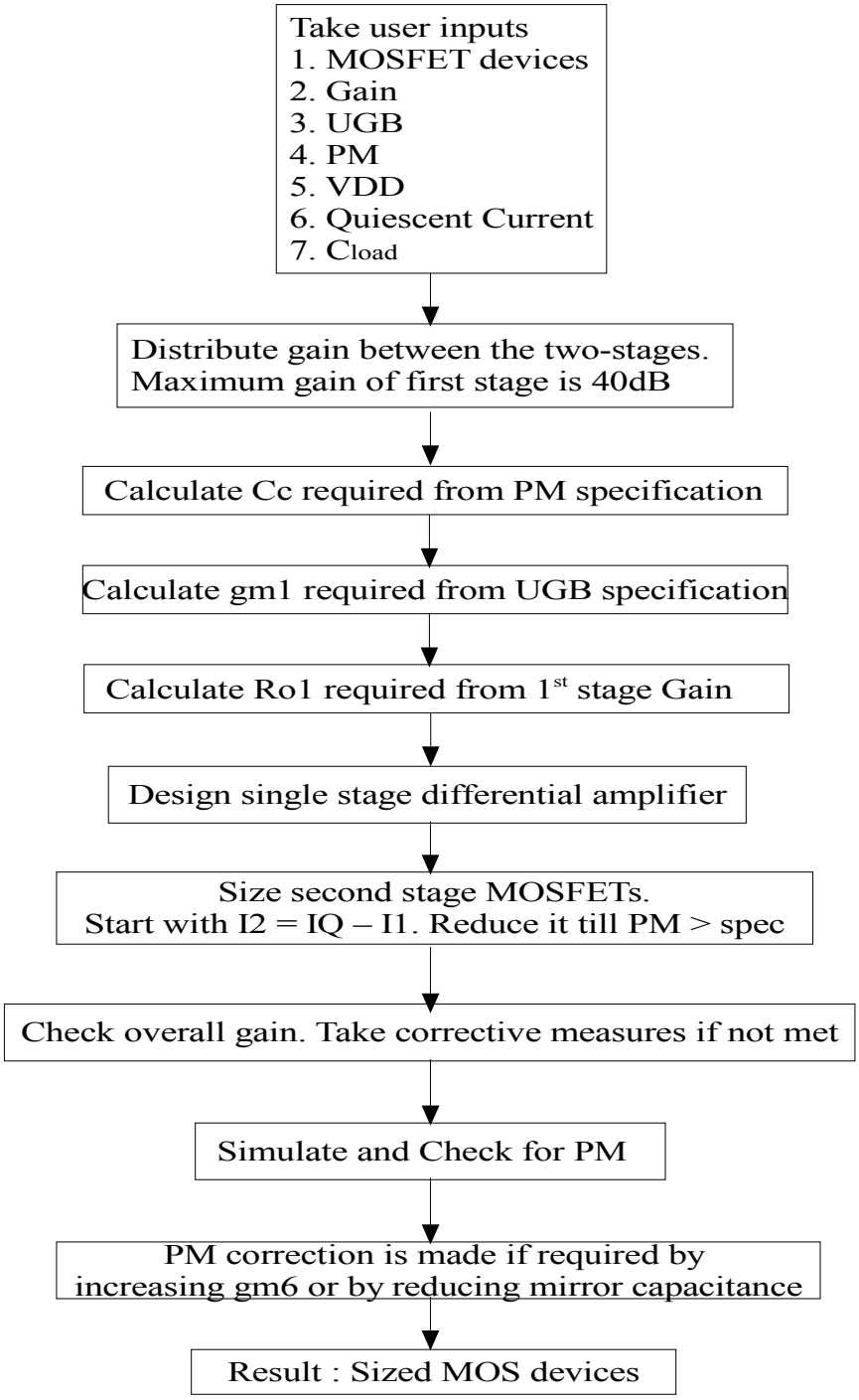


Figure 6.12: Design Flow developed for Two stage differential amplifier

The trade-offs have to be made between Gain, UGB and PM if required. As the load pair MOS size is increased in order to increase the gain of the circuit, the phase margin may drop to the extent of given specification. In that case, a trade-off has to be made between gain and PM by asking the user for the preference. Similarly, if the compensation capacitance value is increased in order to increase PM, the bandwidth reduces. So trade-off may be required at some point in this context also. The developed design flow is tested with different specifications and on different PDKs. The obtained results are tabulated as shown in the Table 6.7

Table 6.7: Design examples for Two stage differential amplifiers

Technology	SCM			RESULT			R	C	I_1	I_2	Diff Pair		Load Pair		Tail MOS	
	GAIN(dB)	UGB(MHz)	PM(Deg)	GAIN(dB)	UGB(MHz)	PM(Deg)	(Kohm)	(fF)	(uA)	(uA)	W(um)	L(um)	W(um)	L(um)	W(um)	L(um)
40nm	80	10	60	86	11.87	61.59	12.87	650	3	12	32	2.2	24	1	1	1
40nm	80	15	70	86	15.3	70.53	12.7	552	3	36	33	3.2	25	2	1	1
28nm	80	10	60	83	11	61	15.5	920	2	20	24	2.5	24	2	1	1
28nm	87	5	60	87	6	60.3	21.96	770	3	15	60	2.5	192	1	8	5

7 AUTOMATING VCO SIMULATION

7.1 Necessity of automation

Till this point, we have seen how a design can be automated. The automation can be updated for more analog building blocks, by using the proposed design methodology. However, for circuits having more than 40 MOS devices, the automation of a design is bit difficult yet possible, when developed by considering lower level hierarchical blocks. For such designs, design can be made by the expert designers. As mentioned earlier, along with the designers' expertise, a number of simulations have to be run in order to get the optimized circuit. These simulations also take a lot of time and there is a possibility to perform multiple simulations to get the required outputs. Usually, this may take hours or days of designers' time. In order to avoid this wastage of time, the multiple simulations which have to be run on a particular design can be automated. In this project work, one such example circuit design is considered i.e., VCO. For manually running the total simulation for the VCO in order to obtain all the result parameters, it is taking almost two to three days. However, after this process got automated, though it takes the similar amount of time, the designers' presence for that much long time is not needed and so is reducing the manual work.

7.2 Basic concepts of VCO

A **voltage-controlled oscillator** or **VCO** is an electronic oscillator, whose output

oscillation frequency depends on the input control voltage. An ideal VCO is a circuit whose output frequency w_{out} is a linear function of its control voltage V_{cont} .

$$w_{out} = w_0 + K_{VCO} V_{cont}$$

Here, w_0 represents the free running oscillator frequency and K_{VCO} denotes the 'sensitivity' or 'gain' of the circuit (expressed in rad/s/V). The achievable range from the VCO is called the "tuning range". The required tuning range is mostly dictated as the frequency range necessary for the application.

Before going for the flow of Automation, the important performance parameters of VCOs are summarized in this section.

K_{VCO} : The sensitivity of the circuit is defined as the variation of output frequency observed with respect to the variation in the given input control voltage at a particular input control voltage.

$$K_{VCO} = \frac{\Delta w_0}{\Delta V_{cntrl}}$$

K_{V2I} : It is the gain of voltage-to-current conversion in the circuit. It is defined as the variation in current sourced to the ring oscillator with respect to the variation in the given input control voltage at a particular input control voltage.

$$K_{V2I} = \frac{\Delta I_{tail}}{\Delta V_{cntrl}}$$

K_{I2f} : It is the gain of current-to-frequency conversion in the circuit. It is defined as the variation of output frequency observed with respect to the variation in current sourced to the ring

oscillator at a particular input control voltage.

$$K_{I2f} = \frac{\Delta w_0}{\Delta I_{tail}}$$

Output Swing: It is defined as the peak-to-peak amplitude of the output wave at a particular input control voltage.

Rise Time: It is defined as the time taken by the output signal to rise from 10% to 90% of the maximum or final value.

Fall Time: It is defined as the time taken by the output signal to fall from 90% to 10% of the maximum or final value.

Duty Cycle: It is defined as the ratio between the pulse duration and the period of the output signal.

PSRR: Power Supply Rejection Ratio is used to describe the amount of noise from the power supply that a particular device can reject. It is defined as the ratio of the change in supply voltage to the equivalent output voltage it produces, generally expressed in decibels.

Phase Noise: It is defined as the frequency domain representation of the rapid, short-term, random fluctuations in the phase of an output waveform caused by the time-domain instabilities.

7.3 VCO circuit designing

The VCO can be designed using a ring oscillator, whose supply voltage is controlled by the biasing circuit. The VCO circuit design that is considered for the automation is as shown in Fig 7.1

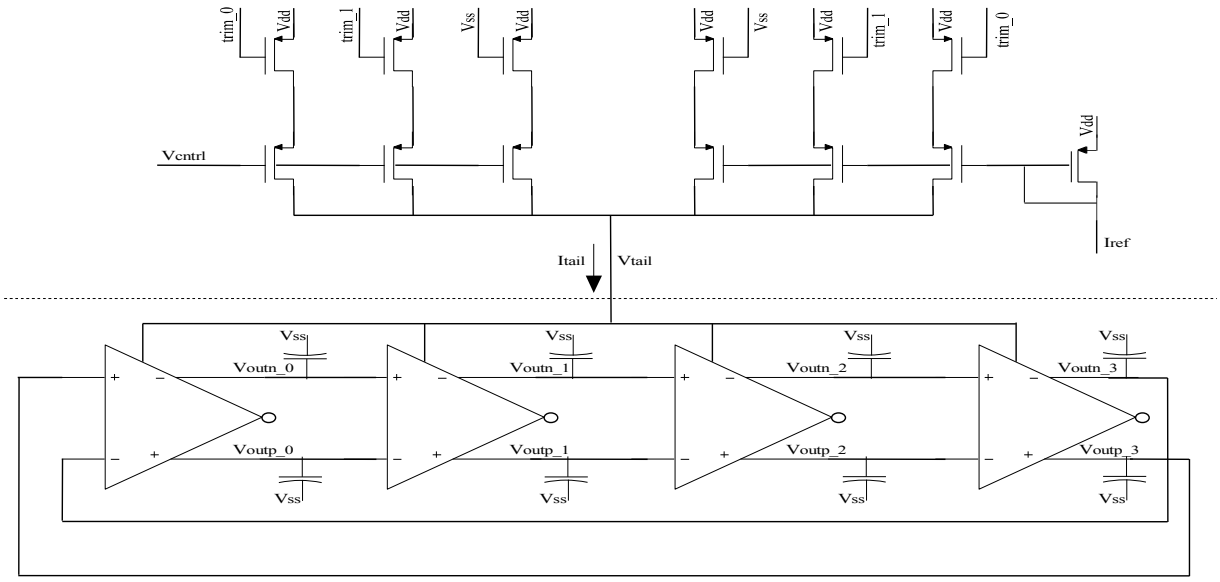


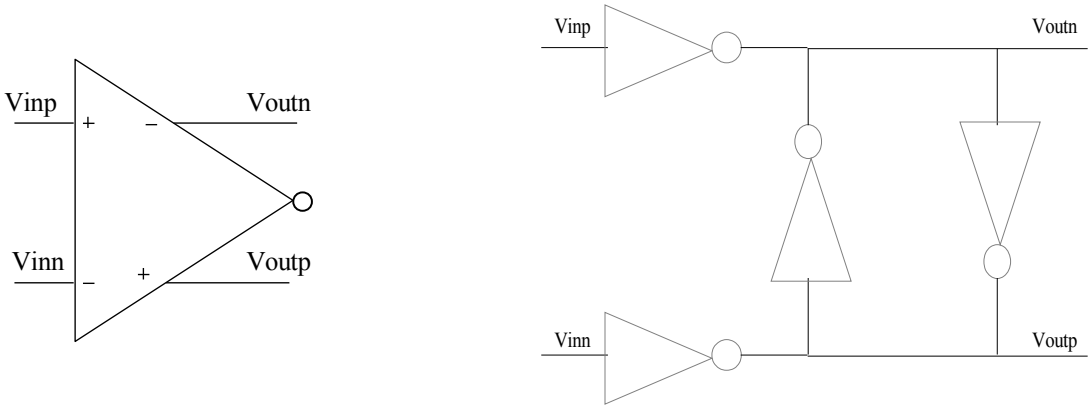
Figure 7.1: Circuit diagram for VCO

The dotted line separates the biasing circuit and ring oscillator. The biasing circuit is used to produce required input current Itail to the ring oscillator from the input control voltage. K_{V2I} is the gain of this biasing circuit. Iref is used to produce minimum current, Itail even though the control voltage is at zero. This is in order to produce the free-running oscillations in the ring oscillator, in the absence of control voltage. Trim pins are used to produce a similar amount of current across all the process variations. These are required because, as the process varies, the threshold voltage of MOSFETs are varied. This results in variation in currents through them, which inturn is the current Itail.

So, in order to reduce the impact of process variations on the output current of the biasing circuit, the trim pins are used. The number of trim pins can be varied from circuit-to-circuit and hence are taken care in the tool designed.

The ring oscillator is used to produce output oscillations at the required frequency,

controlled by the current input Itail. K_{I2f} is the gain of this ring oscillator. The differential inverter circuit uses the chain inverters and feed-forward inverters as shown in Figure 7.2



(a) (b)

Figure 7.2: (a) Circuit symbol of a dual-input/output inverter (b) Internal block diagram of it

The capacitors are placed at the output of each stage in order to hold the value obtained for a while.

The symbol created for this cell view is as shown in Figure 7.3

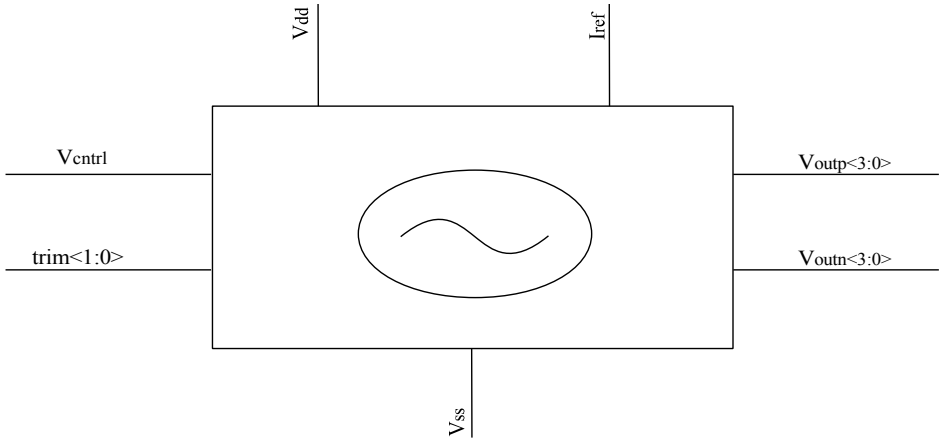


Figure 7.3: Circuit symbol of VCO

The cell design is taken as the input to the tool in the form of circuit symbol. The test-circuit is created for that cell automatically by the tool by adding all the required voltage and current sources for the input terminals V_{dd}, I_{ref}, Vcntrl, trim<1:0> and V_{ss}.

7.4 Simulation Flow

Now that the test-circuit is present, we can proceed for the simulation of analyses required to get the required output parameter values.

7.4.1 Transient Analysis

The transient analysis is performed by sweeping the control voltage as per the given control voltage limit. The oscillation frequency range is a user-input, as the required output frequency range depends on the usage of VCO in other circuits like PLL etc. So, from the analysis done, the control voltage required for generating the required oscillation frequency range is extracted. Along with this, the K_{VCO} , K_{V2I} , K_{I2f} , voltage-swing values for minimum and maximum control voltage are extracted. Also, the voltage, current and R_{th} values at the Tail terminal for minimum and maximum control voltage values are extracted in order to take Thevenin Equivalent circuit for the ring oscillator to do PSRR analysis. This analysis is to be done in default to do other analyses.

7.4.2 PSSP Analysis

In RF systems, oscillator phase noise can limit the final system performance. Hence, it is helpful to do noise analysis for the designed oscillator. PSS(Periodic Steady State) analysis is the

pre-requisite for all the periodic small signal analysis such as Pnoise (Periodic Noise), PAC (Periodic AC), PXF (Periodic Transfer Function), PSP(Periodic S Parameter) analyses. The PSS analysis is used to directly compute the periodic steady-sate response of a circuit in time-domain. It determines the fundamental frequency of the oscillator based on integral multiples of all the source frequencies. The circuit is evaluated for one frequency, and the period is adjusted until all node voltages and all branch currents fall in the specified tolerance. The magnitude of the output signal at first harmonic(maximum frequency required) is extracted by the tool and is used to calculate phase noise.

Pnoise analysis computes frequency convention effects, noise folding, aliasing. The effect of a periodically time-varying bias point on the noise generated by the various components in the circuit is included. It also includes the effects of thermal noise, shot noise, and flicker noise. It displays noise components for each component in the circuit, output noise and input referred noise. From this tool, the output phase noise is extracted for frequency range 10Hz to 10GHz in steps of decades.

7.4.3 PSRR Analysis

To proceed for the PSRR analysis, initially the test circuit has to be modified first. AC magnitude is added for the power supply signal V_{dd}. The ring oscillator is replaced by the Thevenin's equivalent circuit, where R_{th} and V_{th} are extracted from the transient analysis results.

Then the ac analysis is done, and the PSRR is calculated at input frequency range of 10Hz to 10GHz in steps of decades. Any DC Info required (i.e., V_{gs}, V_{th}, V_{ds}, V_{dsat}) for given set of MOSFETs in the biasing circuit are extracted at this step itself.

This tool is designed to perform all the aforementioned analyses using SKILL and PERL scripting languages. The GUI for this is as shown in Figure 7.4

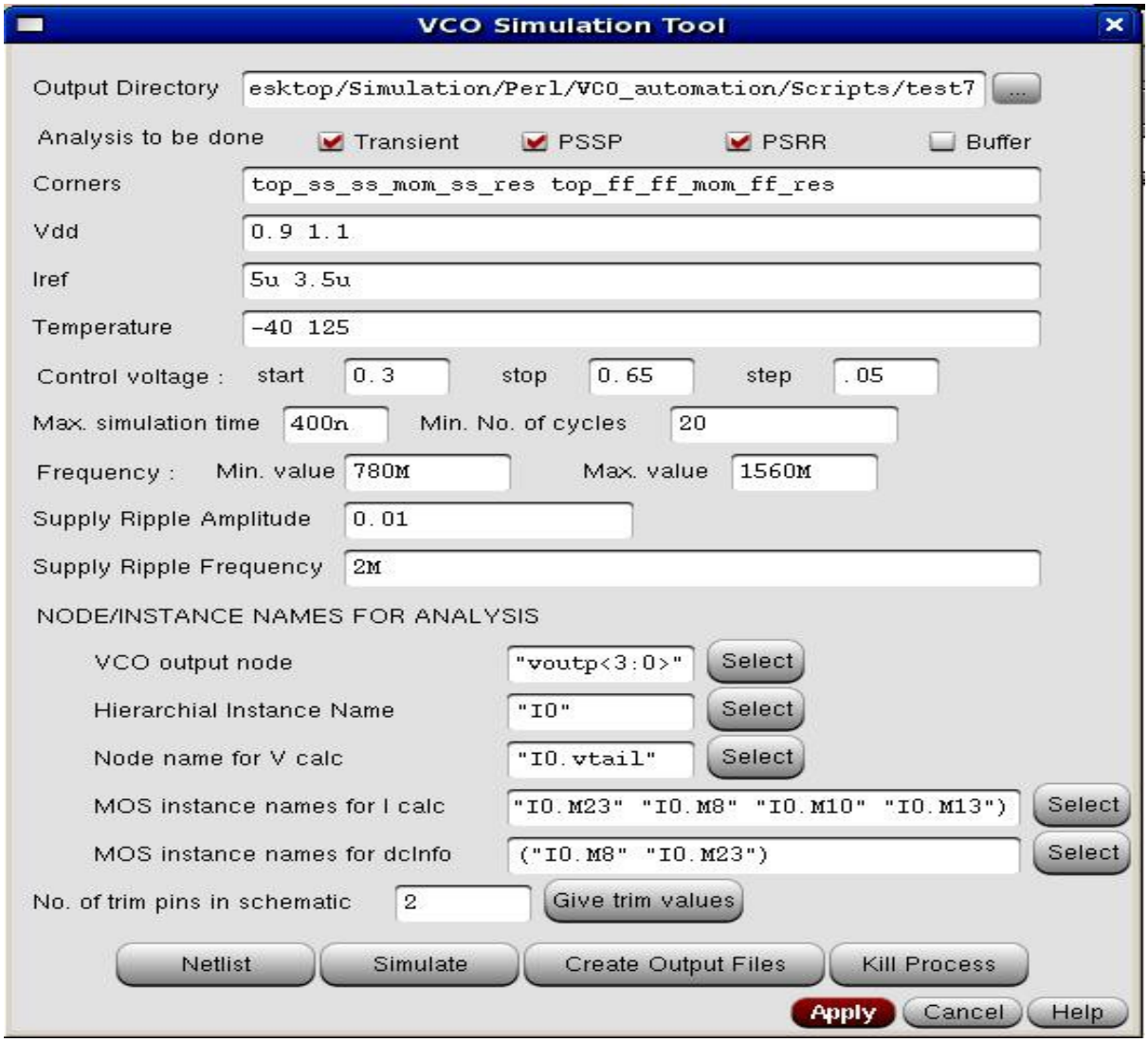


Figure 7.4: GUI developed for VCO simulation automation

8 CONCLUSION

8.1 Outcome of the Work

In this thesis, a tool is developed to characterize devices like MOSFET, resistor, and capacitor, across the PVT variations and on different PDKs. As there are a lot of dependencies like the CDF parameter names, device model names etc., on PDK technologies in order to create a schematic design in Cadence Virtuoso, taking care of all the technologies is a bit difficult task. The technologies 65nm, 40nm, and 28nm are taken care in the designed tool.

A design methodology is proposed in order to automate the analog circuit design. This methodology is made to be independent of the device technology used and the design developed using it is made to be optimized for area and power by using gm/Id methodology. The design flow for analog circuits like basic current mirror, cascode current mirror, low voltage cascode current mirror, single stage differential amplifier, and two stage differential amplifier are developed by using the proposed methodology and are tested for different specifications across PVT variations and on different technologies.

A tool is designed in order to automate performing multiple simulation analyses for a given analog circuit, like VCO which is used as a test case in this work. This tool is made to perform four different analyses i.e., DC, AC, PSS and Transient analyses and is made to extract various design parameters in the form of look-up tables.

8.2 Scope for the future work

There is a lot of scope to extend this project work in the directions listed underneath.

1. The PDK characterization tool developed may be made to be compatible with more technology libraries.
2. The design flow may be made to more complex analog circuits like operational amplifiers, LDO etc., which can use the already developed design flows.
3. The design methodology can be automated by designing a tool using scripting languages like SKILL, PERL, OCEAN. and SHELL.

APPENDIX A

In this section, the GUI for the developed PDK characterization tool is shown. Also, the output plots obtained from the tool is given.

The GUI for the tool is initially displayed as given in Figure A.1

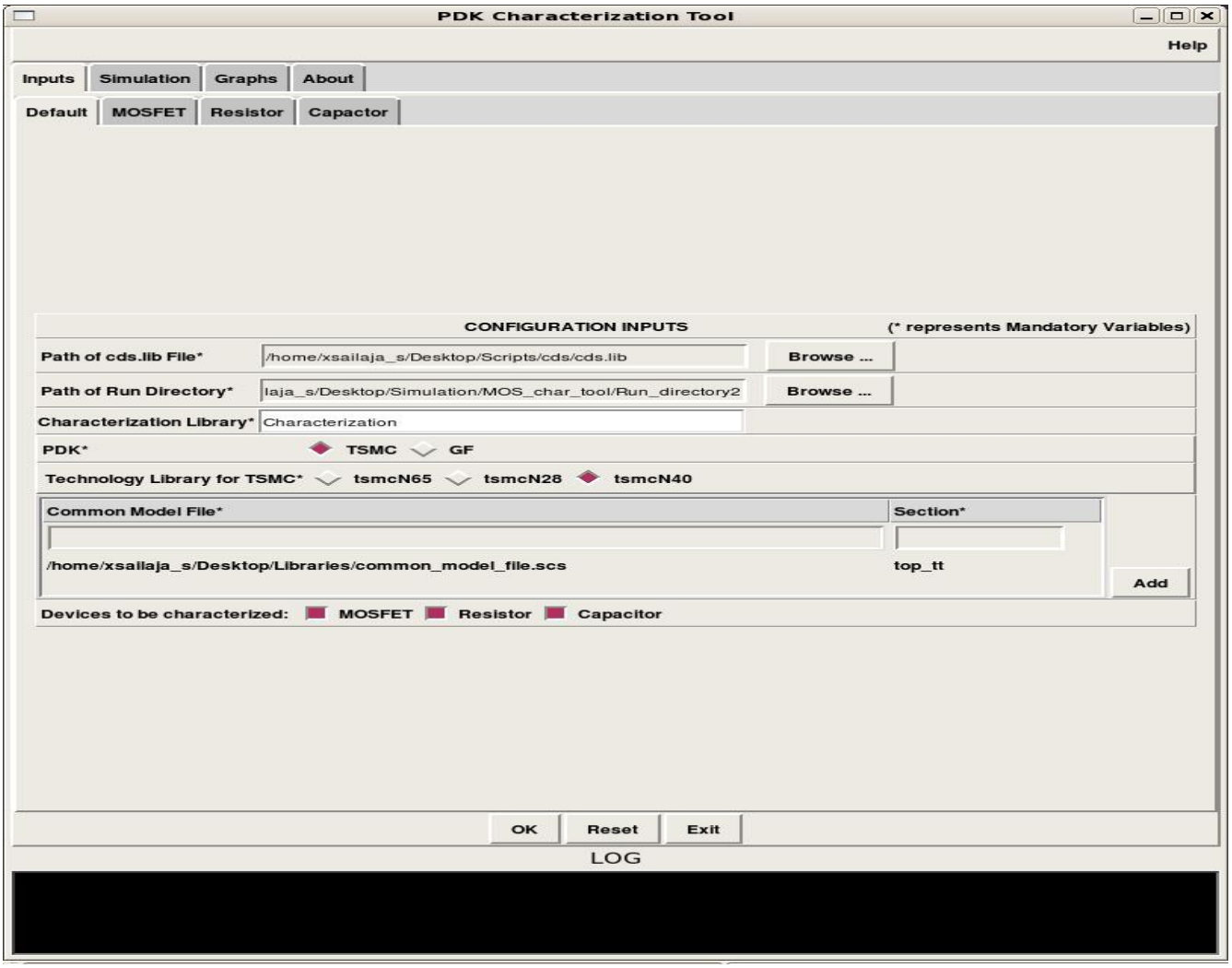
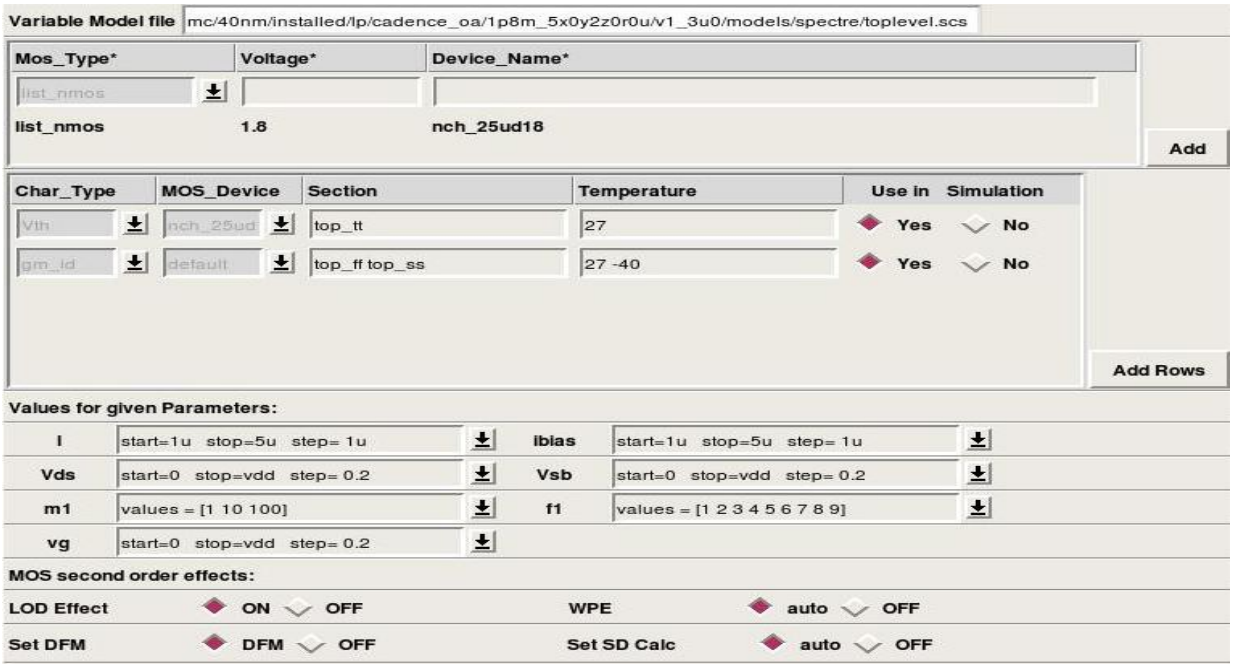


Figure A.1 GUI of PDK Characterization Tool

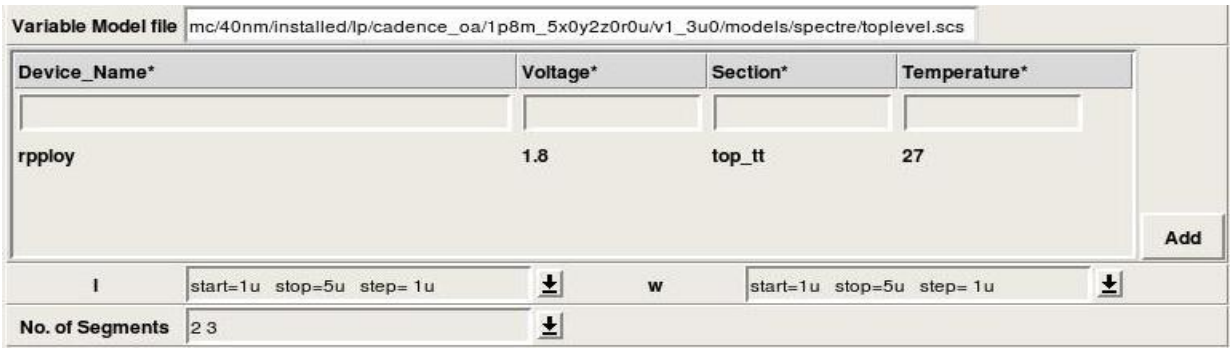
It has 4 tabs to take inputs required, to perform simulation, to generate graphs and to show the copyrights of the tool respectively. Inputs tab is further divided into 4 sub-tabs, one to take default inputs required for any device characterization, the others are for specific inputs required for that

device. All the user given inputs and also the tool internal functions performed are displayed in the LOG shown at the bottom of the GUI. Figure A.1 shows all the default inputs considered. Figure A.2 shows the inputs considered for the corresponding devices MOSFET, Resistor, and Capacitor.

(a)



(b)



(c)

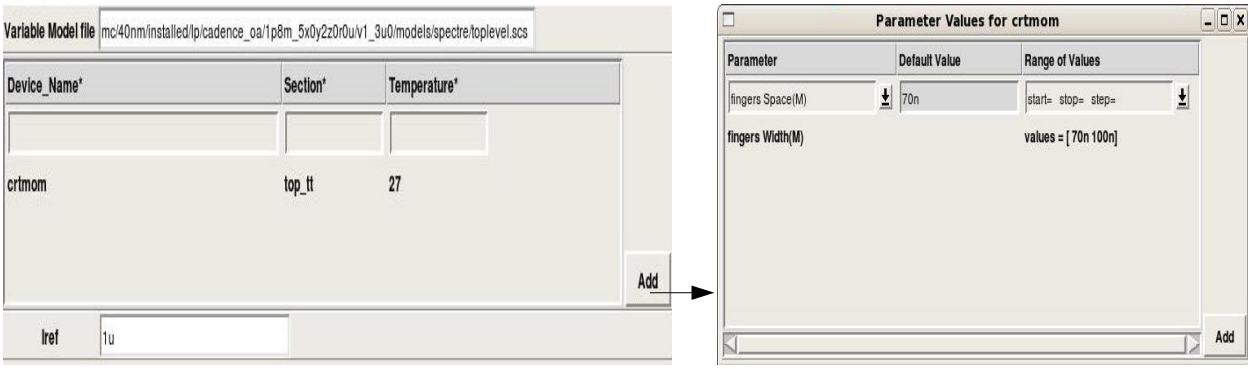
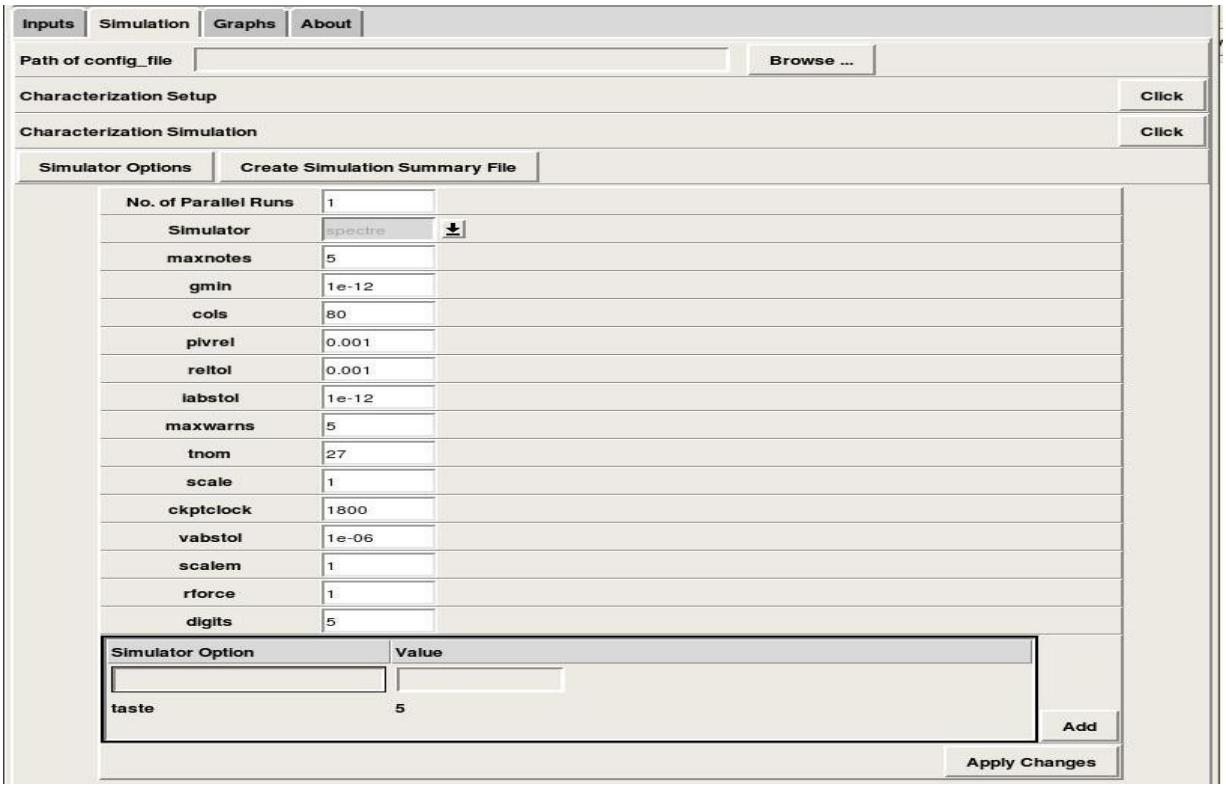


Figure A.2 Input tab for (a) MOSFET (b) Resistor (c) Capacitor and its sub-tab

The simulation tab structure and graphs tab structure are as shown in Figure A.3

(a)



(b)

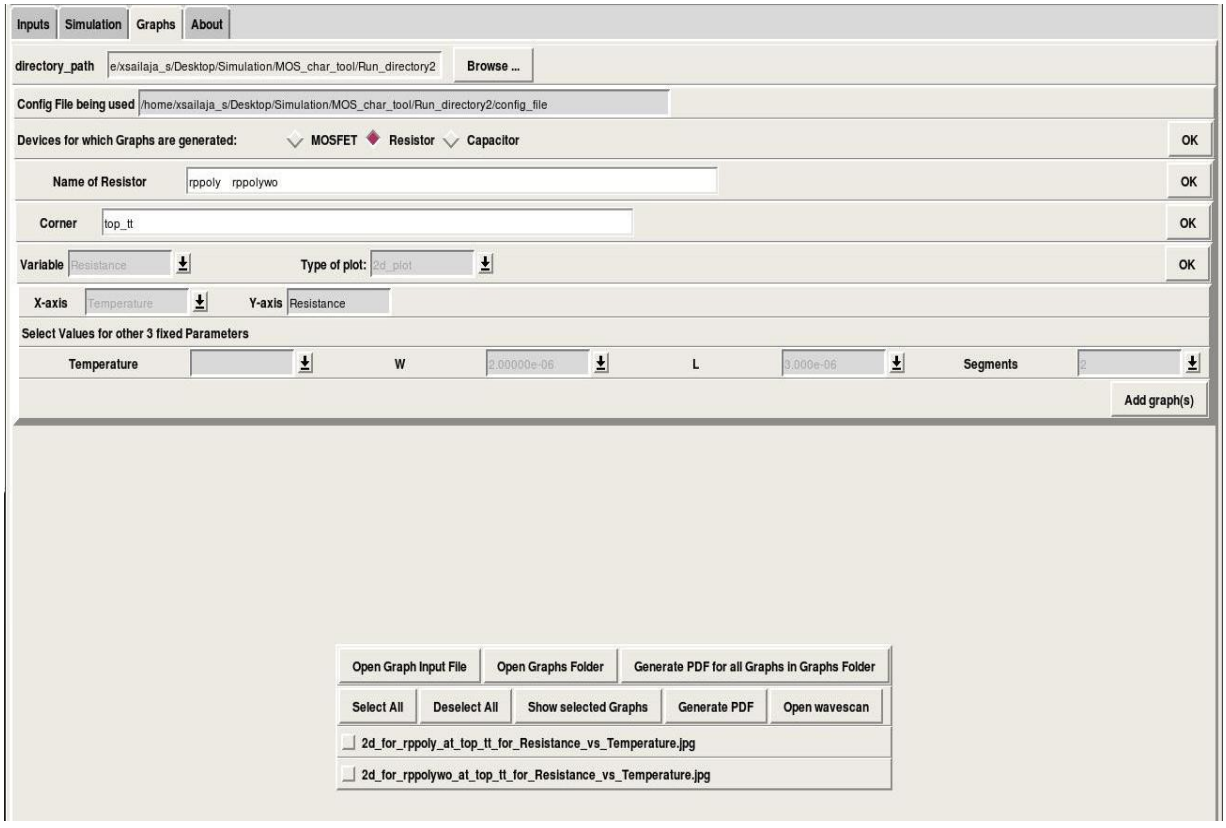
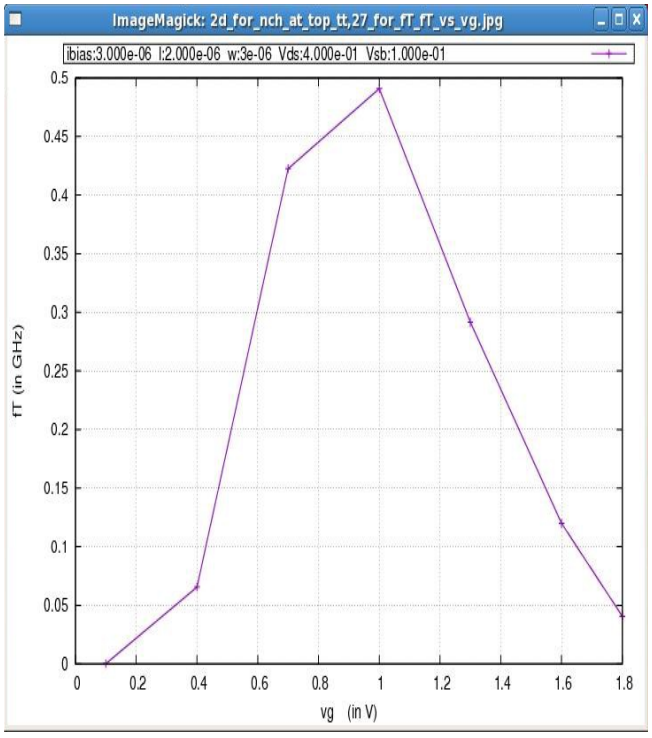
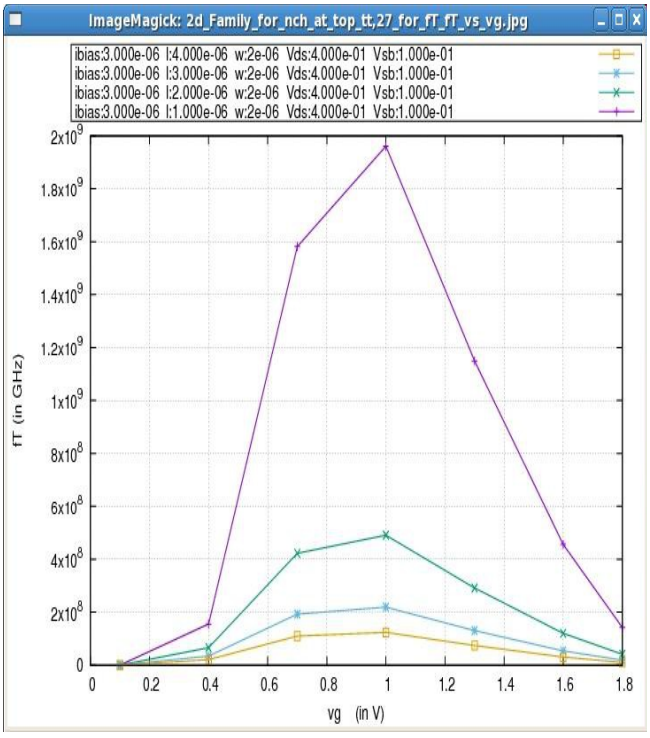


Figure A.3 (a) Simulation Tab (b) Graphs Tab

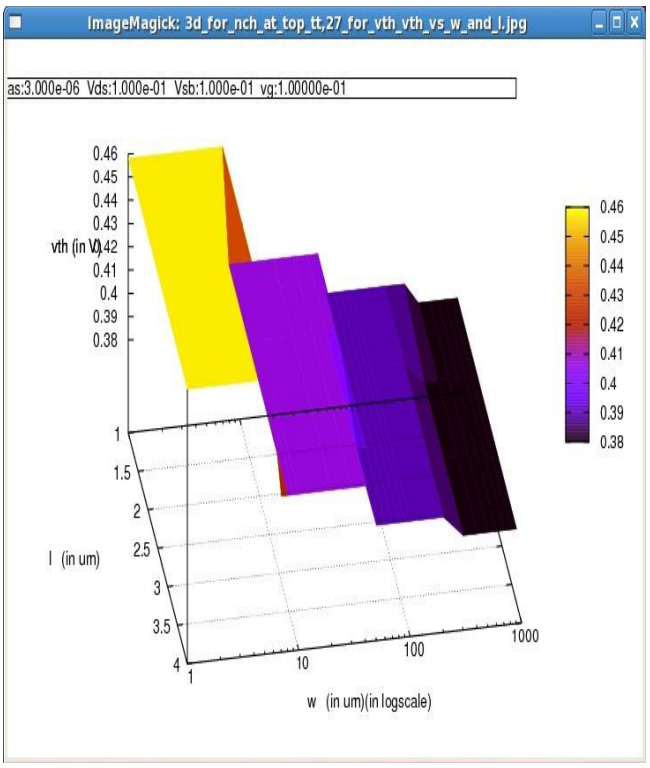
The generated graphs from the tool are as shown in Figure A.4 for reference.



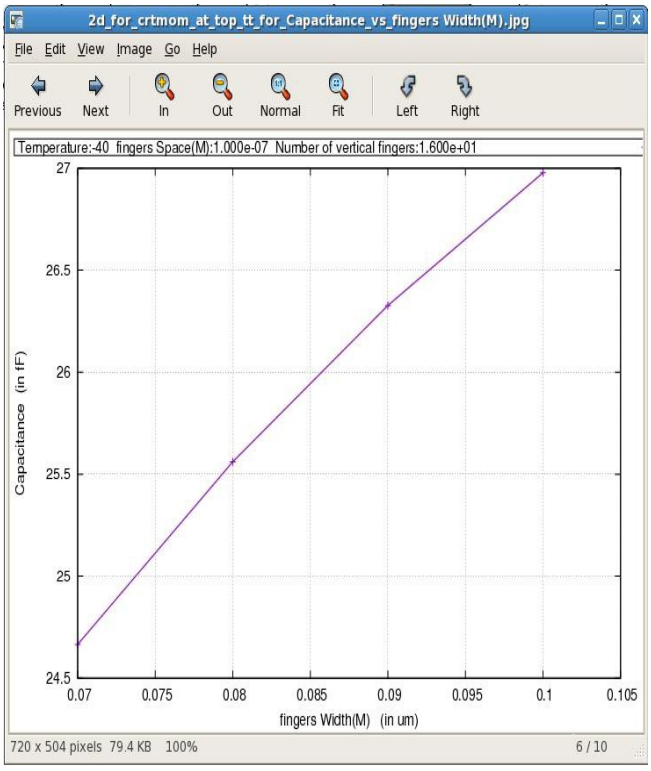
(a) Sample 2-dimensional plot



(b) Sample 2-dimensional family plot



(c) Sample 3-dimensional plot



(c) Sample 2-dimensional plot for Capacitance

Figure A.4 Sample Graphs plotted by the tool

BIBLIOGRAPHY

[1] R. Harjani, R.A. Rutenbar, L.R. Carley “OASYS: A framework for analog circuit synthesis” in IEEE Transactions on Computer Aided Design 8 (12) (1989) 1247–1266

[2] Jihyun Lee, Yong-Bin Kim “ASLIC: A low power CMOS analog circuit design automation” in Integration, the VLSI journal 39 (2006) 157–181

[3] B.Razavi “Design of Analog CMOS Integrated Circuits” McGrawHill International Edition, 2001

[4] Phillip E. Allen, Douglas R. Holberg “CMOS Analog Circuit Design” Oxford University Press, Third Edition, 2012

[5] B.Murmann Lectures on “gm/Id design”, Stanford University, Winter 2012-13

[6] Christofer Toumazou, Costas A. Makris “Analog IC Design Automation Part I – Automated Circuit GenerationL New Concepts and Methods” in IEEE Transactionson Computer Aided Design of Integrated Circuits and Systems Vol 14 No. 2 February 1995

[7] F. Silveira, D. Flandre, and P. G. A. Jespers “A gm/Id based Methodology for the Design of CMOS Analog Circuits and Its Application to the Synthesis of a Silicon-on-Insulator Micropower OTA” in IEEE Journal of Solid state devices, Vol 31 No. - 9, September 1996

[8] “Cadence Analog/Mixed- signal Design Methodology” by Cadence Design Systems Inc.in 2009

[9] L. Richard Carley and Rob A. Rutenbar “How to automate analog IC designs” in IEEE spectrum August 1998

[10] Tom Christiansen, Nathan Torkington “PERL Cookbook” by O'Reiliy Media