Density Optimization for Analog Layout based on Transistor-array

A dissertation submitted for the degree of Doctor of Philosophy
in computer system and VLSI design

23 September 2020

Department of Information and Media Engineering

The University of Kitakyushu, Japan

Author: Chao Geng

(Student ID: 2017DCB401)

(Advisor: Shigetoshi Nakatake)

Author's personal biography and publications are shown on the website ORCID, for more information, please visit the web link https://orcid.org/0000-0002-9637-3922.

ABSTRACT

In integrated circuit design of advanced technology nodes, layout density uniformity significantly influences the manufacturability due to the CMP variability. In analog design, especially, designers are suffering from passing the density checking since there are few useful tools. To tackle this issue, we focus on a transistor-array(TA)-style analog layout, and propose a density optimization algorithm consistent with complicated design rules. Based on TA-style, we introduce a density-aware layout format to explicitly control the layout pattern density and provide the mathematical optimization approach. Hence, a design flow incorporating our density optimization can drastically reduce the design time with fewer iterations. In a design case of an OPAMP layout in a 65nm CMOS process, the result demonstrates that the proposed approach achieves more than 48× speed-up compared with conventional manual layout, meanwhile, it shows a good circuit performance in the post-layout simulation.

The main contributions of this paper are listed as follows: 1) To the best of our knowledge, this is the first work handling DRC and layout density simultaneously. We provide a density-aware format for predictability of analog layout density. Besides, the density optimization design flow has great potential for eliminating aggressive dummy-feature-filling-induced problems. 2) We formulate the process to approach the centering value of the density among constraints as a mathematical optimization problem. Furthermore, we provide a reasonable approach to solve the problem, which searches for an optimum by a Min-Dum scheme to avoid exhaustive search on all the feasible solutions, simplifying the problem as a quadratic programming problem. 3) We develop a TA-style analog layout design automation flow incorporating the density optimization, and we demonstrate a design case of an OPAMP layout in a 65nm CMOS process. Compared with a manual layout by the traditional method, the experimental results demonstrate the high efficiency and the effectiveness of our method.

Keywords: analog layout, design for manufacturability, layout density, transistor array, algorithm design

CONTENTS

ABSTRACT I
LIST OF FIGURES...V
LIST OF TABLES..VII
CHAPTER 1 INTRODUCTION..................................... 1
CHAPTER 2 FUNDAMENTAL THEORY 4
2.1 Physical Design and Verification 5
2.1.1 Dominant EDA Tools Used for Layout 10
2.1.2 The Challenges of Today's IC Layout Design...... 11
2.1.3 DFM Analysis and Verification 13
2.2 Density Issue 16
2.2.1 CMP (Chemical Mechanical Polishing) 18
2.2.2 Investigation for Density Issue Handling 22
2.2.3 Layout Density Uniformity.................................. 28
2.3 Transistor Array 30
2.4 Our Proposal to Address Density Issue 32
2.4.1 OPAMP Circuit for Design Example 35
2.4.2 Parasitic Extraction for Post-layout Simulation . 45
CHAPTER 3 DENSITY-AWARE LAYOUT FORMAT..
.... 52
3.1 Density Checking..52
3.2 Key Idea...53
3.3 Device/pattern parameters for TA-style Layout 54
3.4 Density and DRC Constraints 56
CHAPTER 4 DENSITY OPTIMIZATION STRATEGY
.... 64
4.1 Density Optimization Problem 65
4.2 Min-Dum Scheme 66

4.3 Nonlinear Programming 68
CHAPTER 5 DENSITY-AWARE TA-STYLE ANALOG LAYOUT 71
5.1 Feasible Device/Pattern Parameters 73
5.2 Optimum Device/Pattern Parameters 74
5.3 Design Example 76
CHAPTER 6 CONCLUSITON AND FUTURE WORKS 90
ACKNOWLEDGEMENTS... 92
REFERENCES... 94

LIST OF FIGURES

2.1 Flowchart of IC design..4

2.2 An example of IC layout 5

2.3 An example of design rules over an analog CMOS layout....6

2.4 An analog CMOS fabrication with respect to its layout pattern...8

2.5 Graphical view of density checking in an EDA tool............10

2.6 Increasing design rules and operations with each process...12

2.7 DRC Plus offers fast 2D pattern matching to find and fix problematic configurations..15

2.8 Lithographic process hotspot verification based on lithography simulation...15

2.9 Density checking evolvement with the process nodes shrinking...17

2.10 A diagram of the CMP process in VLSI fabrication...........18

2.11 A physical picture of the CMP process in VLSI fabrication..19

2.12 Layer stacking of chip in three different cases................20

2.13 Photo under electron microscope shows post-CMP thickness variation of each layer..21

2.14 Density variation to the CMP profile............................22

2.15 Density issue handling in digital and analog domains......26

2.16 Dummy fill insertion in a digital layout.........................27

2.17 Dummy fill insertion for diffusion layer in an analog layout...27

2.18 Density variation among neighboring subregions impacts wafer topography...29

2.19 Transistor array.. ..31

2.20 Irregular structures among neighboring transistors lead to post-CMP thickness irregularities.......................................32

2.21 Flowchart of our proposal...34

2.22 Symbol of OPAMP and its equivalent circuit....................35

2.23 Voltage output swing...38

2.24 An example of gain of OPAMP with respect to frequency...39

2.25 Frequency response curve of gain against frequency for an OPAMP..41

2.26 Settling time is the time required for an output to reach...44

2.27 Design flowchart of parasitic extraction based on Star-RCXT...48

2.28 CCI-based parasitic extraction flow..............................49

2.29 Post-layout simulation in Virtuoso for the circuit with parasitics..51

3.1 Checking window over the layout...................................53

3.2 Density distribution and checking result.........................54

3.3 A density-aware layout format of TA-style.......................55

3.4 DRC and density constraints in a 65nm CMOS process......61

3.5 Demos to show the layout pattern density controlling........62

4.1 Density optimization strategy..64

5.1 Design flow based on the TA layout synthesis...................72

5.2 A flowchart for device/pattern parameters extraction........76

5.3 OPAMP schematic for a design example.........................77

5.4 OPAMP layouts...79

5.5 Testbench circuits for DC simulation, load capacitance CL is 1.8pF...81

5.6 Testbench circuits for AC simulation, load capacitance CL is 1.8pF...83

5.7 Testbench circuits for transient simulation, load capacitance CL is 1.8pF...83

5.8 Post-layout simulation..86

5.9 Summary of our whole research....................................87

LIST OF TABLES

3.1 Parameters and annotations used in this work.................63

4.1 Feasible solution set and the corresponding feature density..68

5.1 Summary of design...77

5.2 Design specification and simulation results.....................84

CHAPTER 1

INTRODUCTION

An integrated circuit is fabricated by stacking layers of various materials in a pre-specified sequence, the electrical behavior of circuit depends greatly on the geometrical patterns of the layer [1], [2]. As the IC feature size continues to decrease, the design rules are increased exponentially. However, design period reduction and yield improvement have become more pressing [3], [4]. Chemical Mechanical Polishing (CMP), as a primary technique to control the planeness of silicon surface, is widely utilized in VLSI fabrication [5]. Despite being a predominant planarization technique, CMP is known to suffer from undesired pattern dependent problems [6], [7]. A non-uniform feature density distribution on each layer causes CMP to over/under polish, generating metal dishing and dielectric erosion, which results in the failure of interconnects and yield drop [8].

Previous studies show that post-CMP topography variation is strongly dependent on the underlying feature density, and has become a major cause of yield problem in modern chip manufacturing [9], [10]. For the topography variation reduction and yield improvement, major foundries pay more attention to the layout pattern density. Hence, a rigorous density controlling is imposed to layout design phase so as to achieve uniform topography [11], [12]. However, in the perspective of layout design, rule documentation just indicates layer type which is mandatory to check for density and density level that layout pattern must reach. It cares not for the root-cause or mechanisms behind the density issue. The layout is evaluated as a fine design as long as density meets the required constraints. Therefore, apart from the CMP-directly-relevant layers, such as metals, the layout designer still needs to consider other layers if specified, such as diffusion, poly, and contact.

For advanced processes, a minimum and maximum density of a particular layer within a specific area should be specified. Dummy feature filling is a recommended technique by foundries to increase the density of sparse regions, many papers related to filling analysis and synthesis are proposed in past decades [13]-[16]. Density-related research is always a focus in the field of computer-aided design, most of the works consider optimizing the amount of the fills and accelerating the filling process by algorithm [17]-[19]. With respect to dense regions, slotting/removal on the interconnects is applied [20]. However, turnaround time increases due to iterative verification.

Digital circuit often benefits from EDA tools, layer density levels are normally reached with automatic routing. As for analog and RF circuits, gate and metal layers have to be added manually after verification completed. Dummy feature filling is not only error-prone, but also introduces unexpected parasitics to sensitive signals or devices [21], [22]. As reported in the work [23], dummy feature filling may incur problems as technology nodes advance to 65nm and below. Besides, tuning the dimension and position of device layers such as poly, diffusion, metals, contact, to control layout density, is time-consuming and costly. Consequently, the present method to address analog layout density issue is of low efficiency and reliability.

On the other hand, a transistor-array(TA)-style is proposed for analog layout to suppress process-induced variability. The works related to TA-style demonstrate that the circuit performance is not deteriorated even when introducing unit transistor decomposition [24], [25], [26]. Based on the mechanism of density check, we propose a novel scheme where checking window is portioned into identical tiles. A verification-passed transistor-array is assigned into the tile, and then to cover a given layout area by tiles. Thus, any region can pass density check while moving window inspects layout density levels. A key step is to ensure that the transistor-array meets DRC and density constraints.

In this work, we propose an algorithm which aims at solving DRC and density control simultaneously. Combining two

processes results in a design speed-up. We develop a predictive CMP density model, and show that the feature density distribution on each layer can be predicted by calculating the total area of the layer within a tile. In addition, through an effective algorithm, our method can prune some inferior solutions so that optimum solution is obtained for yield improvement. Therefore, the overall efficiency of analog layout design is significantly improved.

The rest of work is organized as follows. **Chapter 2** gives the preliminaries regarding the density issue, the layout density uniformity, the density checking and the TA-style layout. In this chapter, we also mention the metrics of OPAMP, and parasitic extraction for post-layout simulation. **Chapter 3** formulates density and DRC constraints used in this work. **Chapter 4** is devoted to describing the density optimization problem and proposing a method to solve the problem. **Chapter 5** demonstrates the overall flow for generating the density-aware TA-style analog layout, and also gives a design example auto-generated by our design method. **Chapter 6** concludes this work.

CHAPTER 2

FUNDAMENTAL THEORY

As information explosively increases nowadays, electronic equipment such as smartphone, wearable device, and laptop are seen everywhere in people's daily life. ASICs (application specific integrated circuits) or SoCs (system on chips), as critical components inside those equipment, are massively used with the growth of smart terminals in global market. Predictably, in next decade, semiconductor industry will be booming as great needs for chips in emerging technologies, such as, AI (artificial intelligence), big data, cloud computing, automatic driving and smart electronics.

An integrated circuit (IC), sometimes called a chip or microchip, is a semiconductor wafer on which thousands or millions of tiny resistors, capacitors, and transistors are fabricated. From the formulation of specification of circuit to product shipment, the whole flow is extremely complicated and time-consuming. As seen in **Figure 2. 1**.

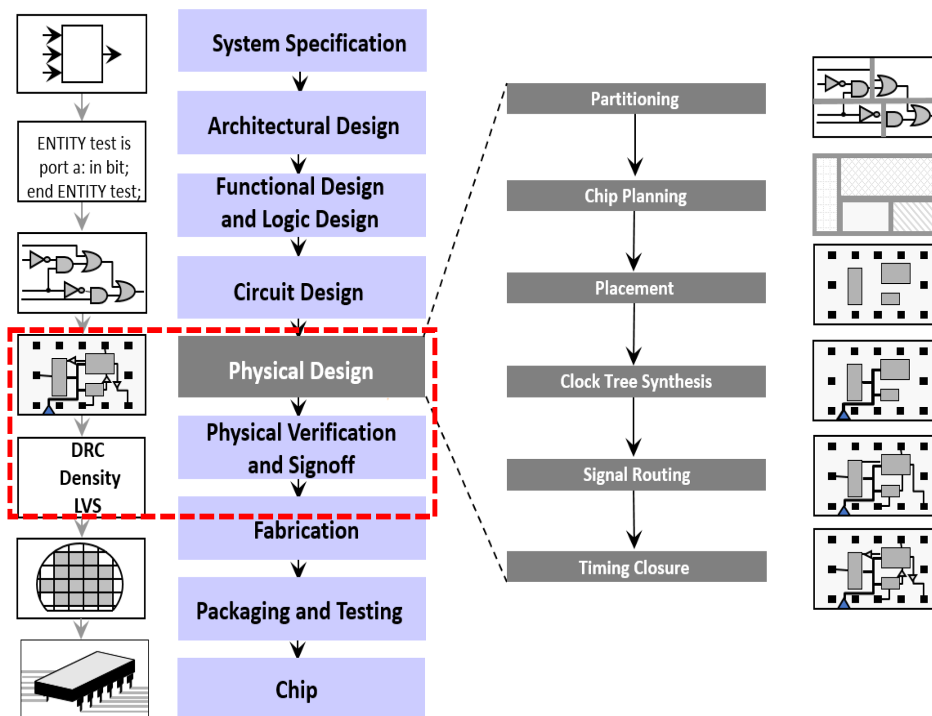


Figure 2. 1: Flowchart of IC design.

An IC can function as an amplifier, oscillator, timer, counter, computer memory, or microprocessor. A particular IC is categorized as either linear (analog) or digital, depending on its intended application.

Analog ICs are used as audio-frequency (AF) and radio frequency (RF) amplifiers. The operational amplifier is a common device in these applications.

Digital ICs are used in computers, computer networks, modems, and frequency counters. The fundamental building blocks of digital ICs are logic gates, which work with binary data, that is, signals that have only two different states, called low (logic 0) and high (logic 1).

2.1 Physical Design and Verification

Physical design of integrated circuit, also known as layout design (see **Figure 2.1.** *1*), is to create planar geometric shapes corresponding to patterns of metal, diffusion, via or other multiple semiconductor layers which make up the device of the integrated circuit. This geometric representation is called integrated circuit layout, after valid layout data is delivered to foundry, the foundry converts the data into another format and use it to generate the photomasks used in a photolithographic process of semiconductor device fabrication [27]-[29].

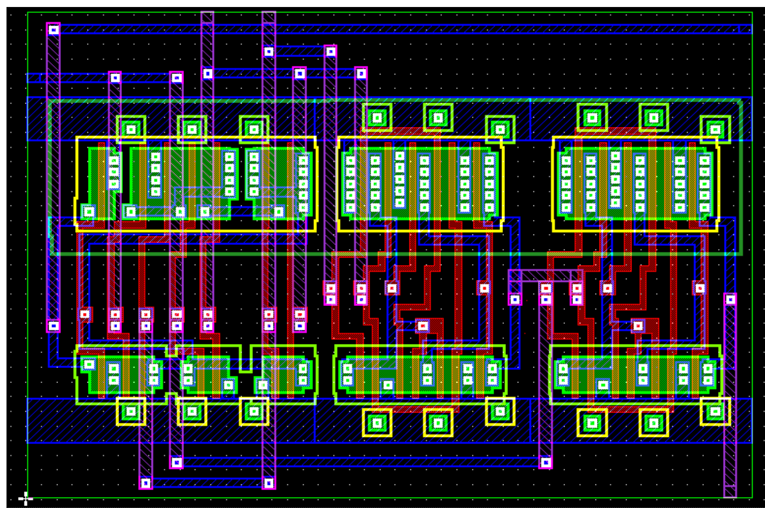


Figure 2.2: An example of IC layout

Since final behavior of chip depends largely on the positions and interconnections of the geometric shapes, high-quality layout guarantees that circuit can be turned into chip with excellent performance.

ICs consist of miniaturized electronic components built into an electrical network by photolithography on a monolithic semiconductor substrate. The physical layout of a certain circuit is typically critical to the success. In order to achieve the desired speed of operation, measures are taken such as: to segregate noisy portions of an IC from quiet portions; to balance the effects of heat generation across the IC; or to facilitate the placement of connections to circuitry outside the IC.

When using a standard process to create layout, interaction between interconnect wires and devices must be taken into account. In general, typical effects such as, interaction of the many chemical, thermal, and photographic variables, will be considered and controlled by foundry.

For the manufacturability, layout must be manipulated under design rules to meet certain criteria: performance, size, density, power dissipation. In the design of very-large scale integration (VLSI), massive design rules and constraints are imposed to layout creation, such as minimum space between geometries, interconnects. Minimum or maximum size of polygon shapes. As shown in **Figure 2.3**.

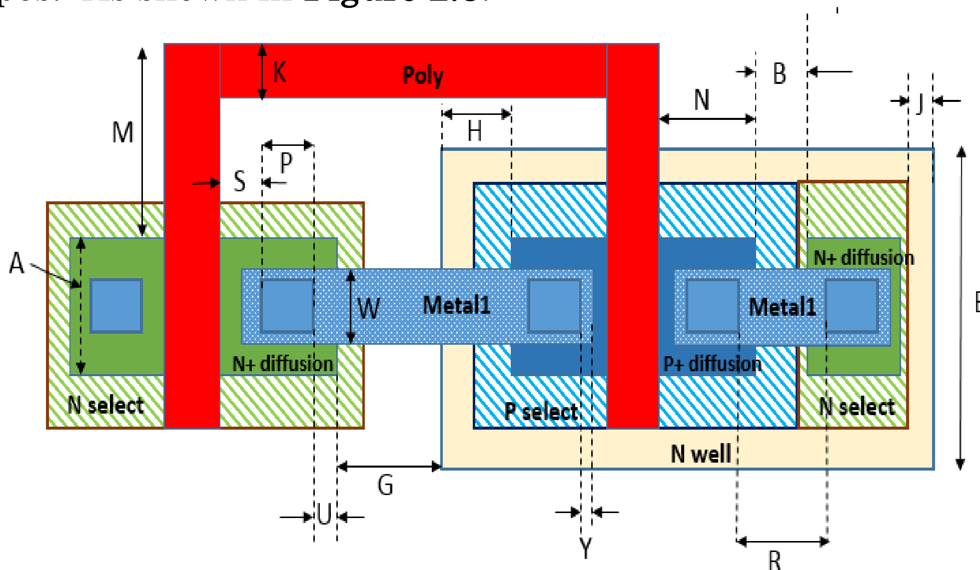


Figure 2.3: An example of design rules over an analog CMOS layout.

After layout is generated, which needs to be inspected by series of checks, in order to ensure manufacturability and functionality of circuit, this process is called layout verification. Physical verification checks the correctness of the generated layout design. Most common checks in verification process are listed as below. This consists of verifying that the layout:

- Complies with all technology requirements – Design Rule Checking (DRC).
- Is consistent with the original netlist – Layout vs. Schematic (LVS).
- Has no antenna effects – Antenna Rule Checking.
- **Passes the density verification at the full chip level. Density error-free is a very critical step in the lower technology nodes (Density Checking).**
- Complies with all electrical requirements – Electrical Rule Checking (ERC).
- Guarantees the extra extracted parasitics will still allow the designed circuit to function – Parasitic Extraction.

When all of verification is completed, the data is translated into an industry-standard format, typically GDSII, and sent to a semiconductor foundry for fabrication. The process of sending this data to the foundry is called tape-out because the data used to be shipped out on a magnetic tape. The foundry translates it into another format and use it to make photomasks.

As shown in **Figure 2.4**, an analog CMOS layout pattern is transformed into a circuit in silicon wafer.

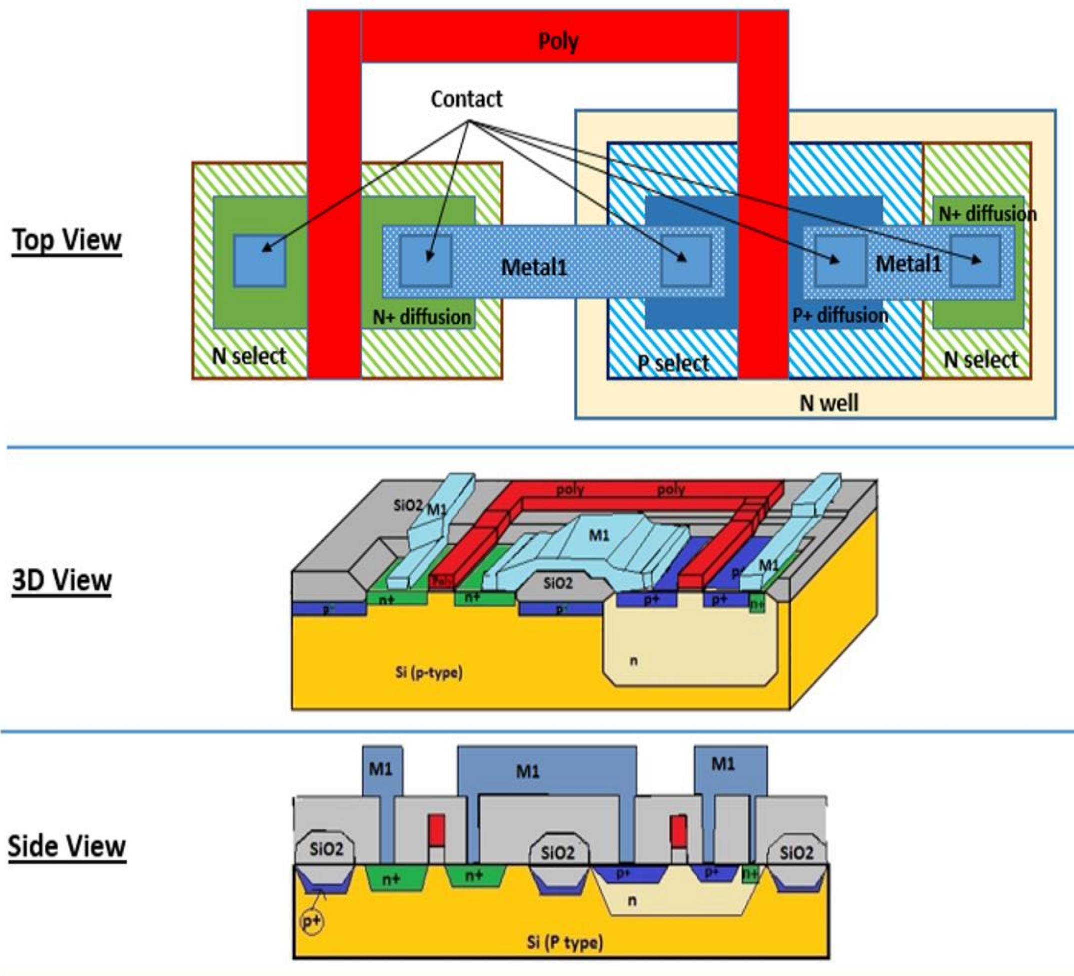


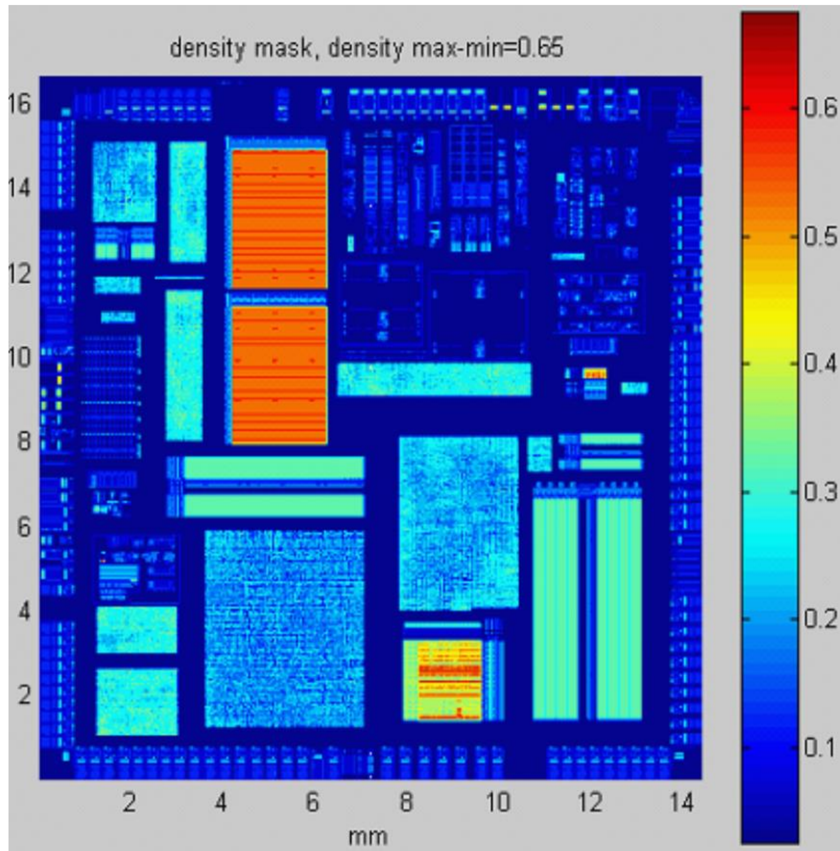
Figure 2.4: An analog CMOS fabrication with respect to its layout pattern [30].

Each layout pattern layer is formed in silicon wafer by photolithographic technique and all layers are stacked in an order prescribed by foundry. The silicon topography of fabricated circuit is shown by a side view of the figure above.

In the technology process nodes at 65nm and below, to improve the manufacturability and yield of IC chips in advanced process nodes, a rigorous density checking has to be imposed to layout design phase.

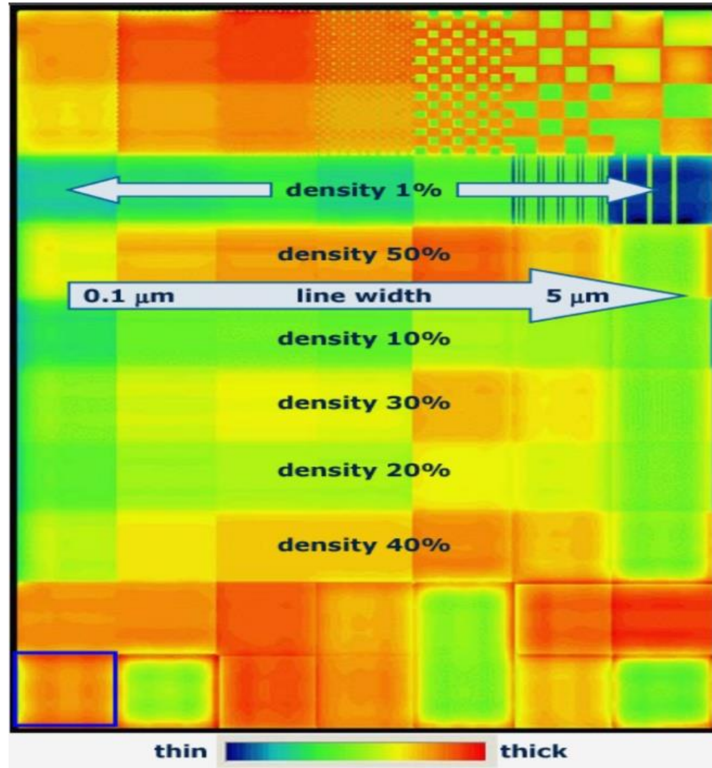
Figure 2.5 shows that layout designer inspects locally or globally the density level across a chip, but only few EDA tools provide such advanced graphical view for density checking. The

view in **Figure 2.5(a)** shows the difference in color which distinguishes the density level. Higher density tends to be red while lower density tends to be blue, by which layout designer can better be aware of the density uniformity of a chip. The view in **Figure 2.5(b)** shows the density distribution across tiles in a layout, by which we can learn the density gradient among neighboring regions.



(a)

Figure 2.5: Cont.



(b)

Figure 2.5: Graphical view of density checking in an EDA tool [31]. (a) Density checking in a chip layout. (b) Density distribution across tiles in a layout.

2.1.1 Dominant EDA Tools Used for Layout

In early layout design before 1970, there were no computing system and the term "software" was not yet invented. The circuits were simple, and the layout was drawn with pencils and rulers, and the physical geometries were checked by eyeballs. With the development of semiconductor technology, the scale of circuit and system become larger, drawing layout by hand was low efficiency and can no longer meet the requirement of the time to market.

As computer technology thrived at next decades, electronic design automation (EDA) tools was developed rapidly with the software technology. For now, mostly used software targeted at modern IC layout are **Cadence**, **Synopsys**, **Mentor Graphics**, as they are dominant among EDA tool vendors. With the aid of IC

software, including place and route tools or schematic-driven layout tools, the speed of design is accelerated significantly.

- Cadence Virtuoso Platform – Tools for designing full-custom integrated circuits, includes schematic entry, custom layout, physical verification, extraction and back-annotation. It is mainly used for analog, mixed-signal, RF, and standard-cell designs.
- Synopsys Design and Verification Platform – Deliver the best silicon chips faster with the world's No.1 electronic design automation tools and services, industry's broadest portfolio of high-quality, silicon-proven IP. Major products include Design Compiler, IC Compiler.
- Mentor Graphics Platform – Best-known for its IC verification tool, such as Calibre nmDRC, Calibre nmLVS, Calibre xRC, Calibre xACT 3D, Mentor Graphics company is now acquired by German company Siemens, becoming a part of the Simens PLM software business unit.

2.1.2 The Challenges of Today's IC Layout Design

Nowadays there are many commercial EDA tools available for accelerating IC design, reducing the period of design. However, as the process node is shrinking, both the size and spacing of design features are decreasing. Besides, as semiconductor technology enters deep sub-micron era, many physical process effects that were relatively insignificant at earlier nodes begins to impact the yield and performance.

Especially, with the shift to nanometer geometries from 65nm to 45nm, design rule compliance no longer guarantee that layouts could be turned into chips as expect. Some unexpected effects may severely affect the electrical characteristics of circuits.

On the other hand, millions of transistors integrated across a die is quite normal at advanced nodes. Since the size of features and spaces between them decrease, interactions of features become more significant and sophisticated.

To ensure manufacturability and to control interactions of IC layout, numerous design rules for advanced nodes are imposed to layout creation, they often encompass multiple operations per rule, such as multi-variable equations that express complex spatial constraints and relationships between design features within a certain 2D proximity.

As the integrated circuit feature size continues to decrease, the design rules are increased drastically. As these rules became more numerous and more complex with the process node (see **Figure 2.6**), the computational complexity of design rule checks (DRCs) and the number of potential violations grew exponentially.

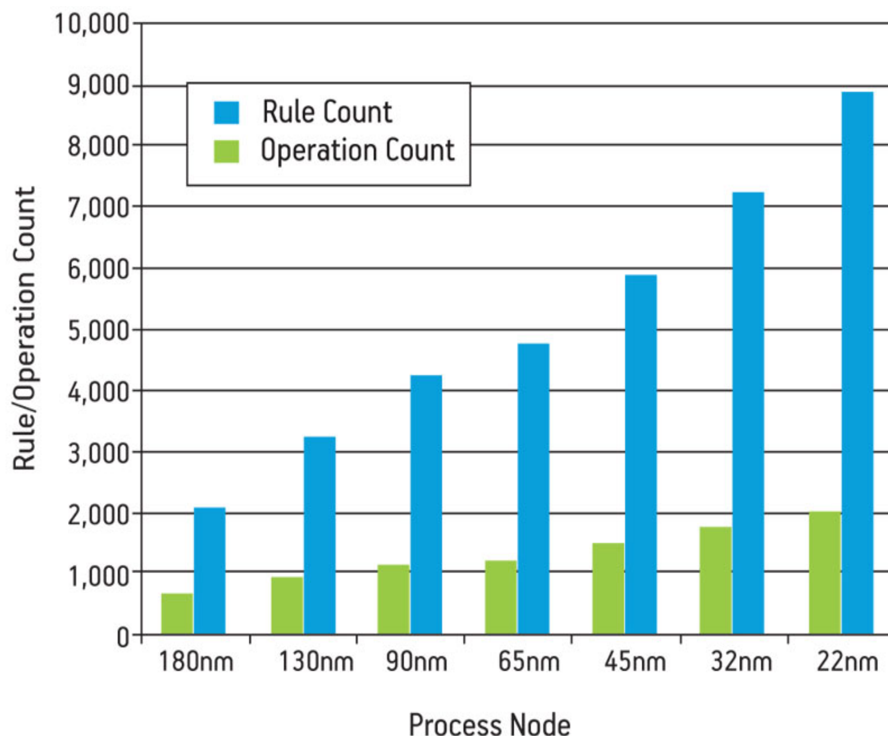


Figure 2.6: Increasing design rules and operations with each process [32].

Although commercial EDA tools yet provide powerful features to deal with those problems, it still cannot keep pace with the development of process nodes. Especially for analog circuits, layout creation by hand is major method and few tools provide automated solution for analog IC design, hence making it more time-consuming.

At the same time, analog application is increasing and reflecting strong growth in wireless and sensing technologies. According to a market research, analog circuitry takes up only 20% of the area of today's modern mixed signal devices, whereas it's likely to account for as much as 80% of yield loss.

Challenges that IC designers are facing today [33]-[35], are summarized as below.

- Complexity of IC design and verification is increasing.
- Physical effects induced by the process variation become more significant.
- The risk of catastrophic failure during fabrication is increasing and yield is decreasing.
- The conflict between time dissipation and design complexity become inevitable due to poor feature of EDA tools for analog circuit.
- The demands for reducing design time and improving yield in mass production become increasingly urgent.

2.1.3 DFM Analysis and Verification

As the process node moves to nanometer, interactions between features become significantly and physical effects due to process variation severely affect yield. Therefore, the demand for DFM (design for manufacturability) becomes stronger [36]-[39]. To deal with manufacturing issues, foundry impose more performance-driven constraints and yield driven constraints to design rules. On the other hand, in layout design process, EDA vendors develop more powerful tools embedded with DFM-driven features, making IC design more effectively converge in fewer iterations.

There are already some applications and approaches addressing yield issues caused by random effects or fabrication failures. The process-based DFM solutions identify and fix design areas that are easily introduced into design violations. Such as, shorts and opens. Wire spreading, via doubling and critical area analysis becomes mainstream.

Some effective technologies to deal with manufacturing issues are introduced below.

- Rule-based DFM.
- DRC Plus.
- Model-based DFM.
- Variability based DFM.

Rule-based DFM. The design rule is a set of geometrical constraints that define spacing between features and interconnect layers, layout creation is manipulated under the constraint of the design rules. Only configurations that comply with constraints can be fabricated. Design compliance ensure that circuits can work as expect. Even complex issues such as dielectric constant (k), numerical aperture (NA), source frequency (λ), source shape and off-axis illumination, are summarized in the form of geometric measurements such as minimum line width, minimum space, and forbidden pitches. Complex interactions can be defined as specifications on tip-to-tip spacing and tip-to-line spacing using DRC. Sufficient design rules can maximize circuit performance and minimize process variation. As the process nodes move to 65nm and below, the mount of the design rules increases dramatically.

In manufacturing process of integrated circuits, some cases always happen that some features can't be fabricated even if geometries comply with design rules. Designers quickly recognize the limitations of traditional design rules at advanced nodes. Considering this fact, designer augments design rule checks using DRC Plus, which adds fast 2D pattern matching to standard DRC to identify problematic configurations. By this way, designers can quickly fix undesirable geometries. As seen in **Figure 2.7**.

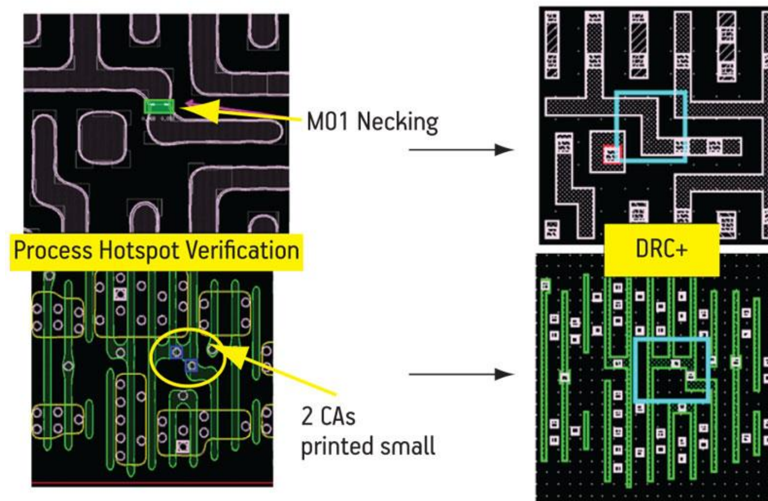


Figure 2.7: DRC Plus offers fast 2D pattern matching to find and fix problematic configurations [32].

Model-based DFM. This tool predicts manufacturing results by using lithographic simulation, allowing designers to refine and correct layouts before tape-out. It's effective to identify specific design areas most likely to suffer distortion in the actual manufacturing. EDA vendors provide designers with design kit, much like DRC kit, to run simulation and to get an accurate description of layout creation under given process. Then, hotspots, also called error-prone area, can be identified and revised for the DRC clean. See the **Figure 2.8**.

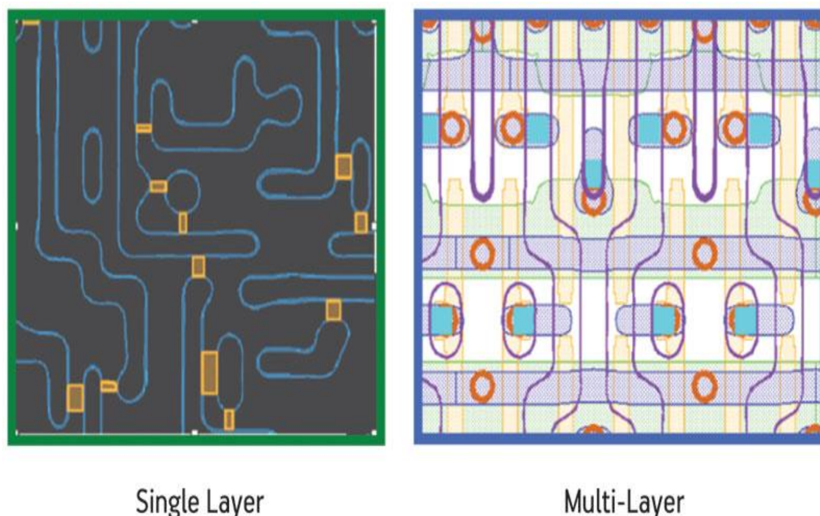


Figure 2.8: Lithographic process hotspot verification based on lithography simulation [32].

Variability based DFM. There is a variation between the silicon shape and drawn layout due to pattern fidelity issues at

advanced nodes. It will erode performance and cause catastrophic manufacturing failures. Designers must extract variation and bring it into the design flow for analysis, control and minimization. In order to avoid poorly matching silicon behavior of transistors and interconnect layers, variation-based DFM becomes essential to layout design. One example of variation-based DFM is lithography simulation, which predicts silicon shape by utilizing information extracted from layout. Then realistic process-based pattern can be used for chip fabrication. Contour-based extraction uses simulated critical dimensions for transistor gates to extract timing information. With accurate comparison to silicon measurements, an accurate model of current density can be obtained.

As the demand of design for manufacturability is growing, some manufacturing issues that was previously handled during fabrication, now has been pushed up to design and verification stage in the top-down IC design flow. Processed-based simulation provides accurate model for silicon shape, and other techniques enhance verification, which will help improve yield significantly.

2.2 Density Issue

In the process of chip fabrication, there is a step to ensure the planarity of the layer surface, called chemical mechanical polishing (CMP). Since only planar-shape silicon can be manufactured and uniform thickness of dielectric reduce the process variation. In the layout design stage, uniformity of layer is corresponding to planarity of silicon shapes. For the manufacturability, the density of each layer must be inspected under a set of density constraints. Density constraints at each process is provided by foundry. As the process nodes decrease, density checking is evolving progressively. See **Figure 2.9** and the following summaries.

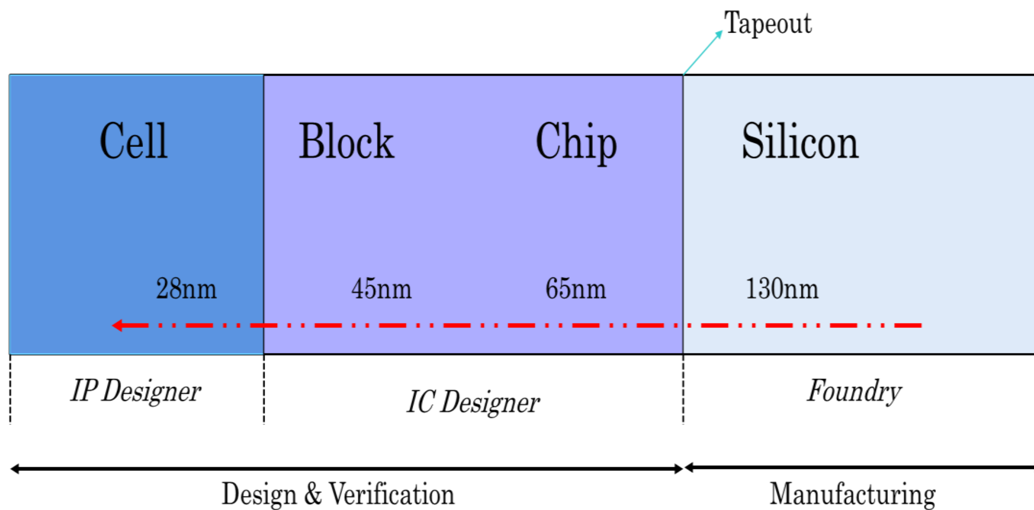


Figure 2.9: Density checking evolvement with the process nodes shrinking.

- Checking area moves from silicon to cell.
- Responsibility for density checking moves from foundry to IP designer (cell designer).
- Density concern moves from manufacturing to design and verification stage.

In technologies before 130nm, to reduce density variation, foundry adds extra metal shapes in the empty spaces without even telling customers. Since the process is thought of having no impact on electrical characteristics of the circuit, dummy fill is nonfunctional circuit, meaning that, it's not part of the circuit.

However, with each new technology, foundry has to solve massive challenges. In solving those challenges, they have to make comprises that add new design rules. As mentioned above, both the design rules and density rules increase with the process nodes.

Thus, layout is being constrained at very local level, and density checking is being constrained at macro level (density over a small area). A new technology and layer may be more sensitive to variation in the density, thus needing a new rule for allowable density gradient. Each new technology has made density constraints stringent, meanwhile, adding more restrictions on the layout manipulation. Today's layout design and verification has become a tough work in process-driven design.

After 130nm, as the process nodes decrease, fill placement was becoming more aggressive and closer to signal lines. It's difficult for foundry to convince customer there was no impact on the electrical behavior of their designs. By around 65nm, it was common for designers to control fill placement themselves, using design rule decks and density requirements provided by the foundry. Even down to 45nm, it was still a chip-level issue to be solved at chip assembly.

2.2.1 CMP (Chemical Mechanical Polishing)

Density checking is important to the manufacturability. Studies show that post-CMP topography variation is strongly dependent on the underlying feature density. A uniform layout pattern density contributes to a uniform topography of silicon wafer, so that the electrical performance of the fabricated circuit can be guaranteed. CMP is a primary technique to control the planeness of silicon surface, which is widely utilized in VLSI fabrication. **Figure 2.10** shows the whole process.

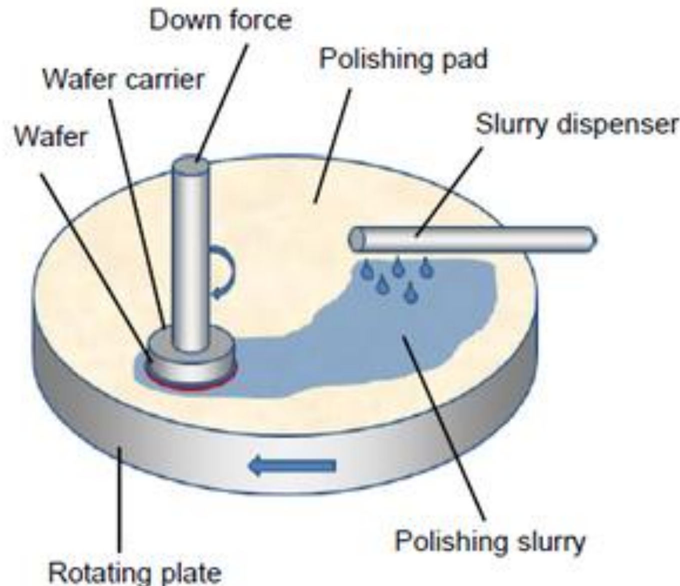


Figure 2.10: A diagram of the CMP process in VLSI fabrication [40].

As seen from the above figure, CMP is comprised of the wafer carrier that the silicon wafer is attached to the wafer carrier, the slurry feeder that provides chemical slurry, and the polishing pad that grinds the surface of silicon wafer.

The wafer carrier and the polishing pad are rotating simultaneously, the slurry feeder is dropping chemical slurry when the polishing pad grinds the silicon surface.

The CMP process is realized to remove the unwanted layer with the combination of chemical and mechanical forces. Physical picture of CMP is given in **Figure 2.11**. In the VLSI manufacturing process, layout pattern is transformed into the circuit in silicon wafer through the photolithography, various materials are then deposited onto the trench etched by the corrosive acid. Metal interconnects are generally softer than the dielectric material, which usually is silicon dioxide. Therefore, metal interconnects are easily removed by mechanical force while dielectric material is removed by chemical slurry.

Figure 2.11: A physical picture of the CMP process in VLSI fabrication [41].

CMP is a critical step to ensure the manufacturability and functionality correctness of IC chip. A plane silicon surface for each layer contributes to a good profile of stacking layers. As shown in **Figure 2.12**, layer stacking of chip in the absence of CMP is twisted and deformed, such profile would severely damage the functionality of chip, as a result, decreasing the yield of chip in mass production. Without considering any CMP related defects, layer stacking of chip after CMP process becomes better as the post-CMP profile is more planar. However, actual post-CMP profile still shows silicon topography variation in the

presence of dishing and erosion side effect. Therefore, it is insufficient for planarizing the silicon surface by CMP only. In some cases, density uniformity improvement in layout design phase is necessary for reducing the topography variation.

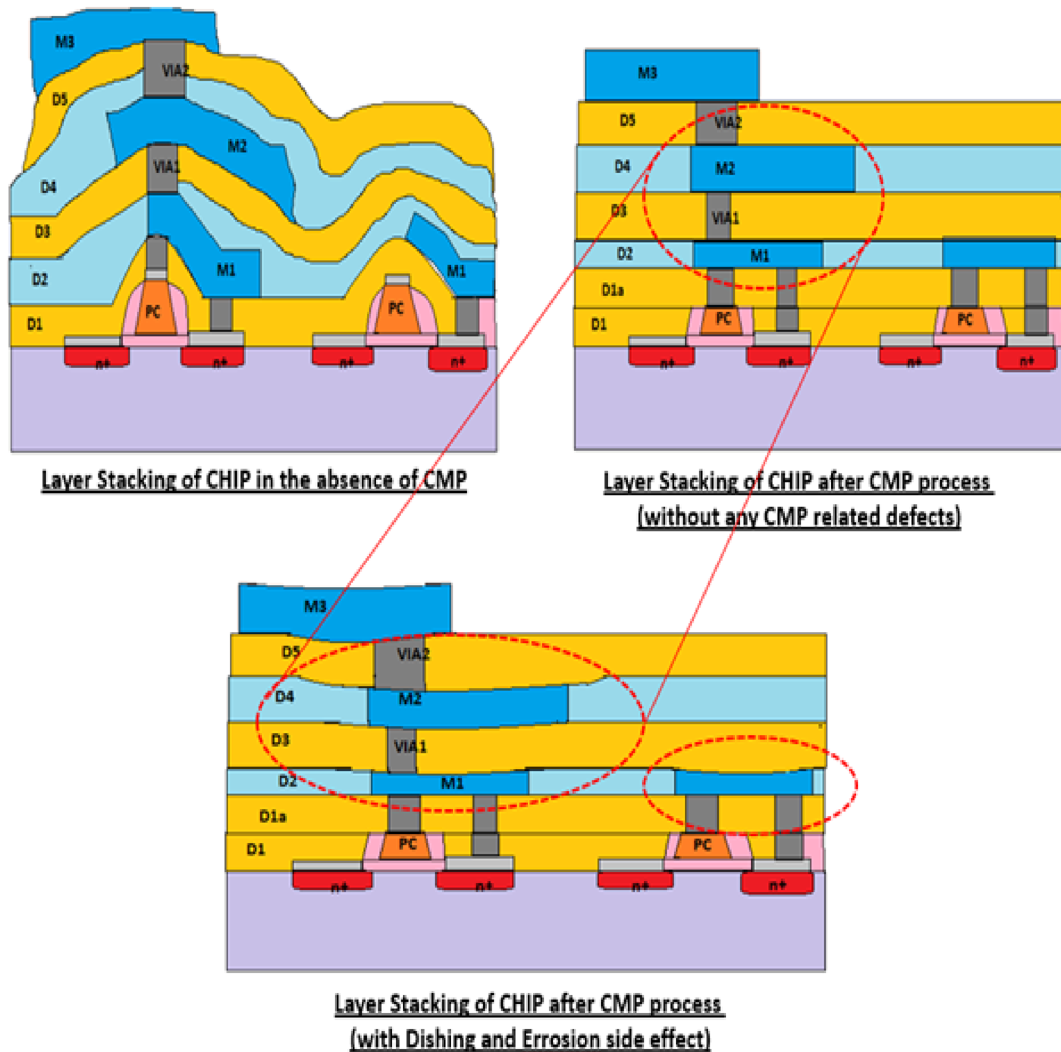
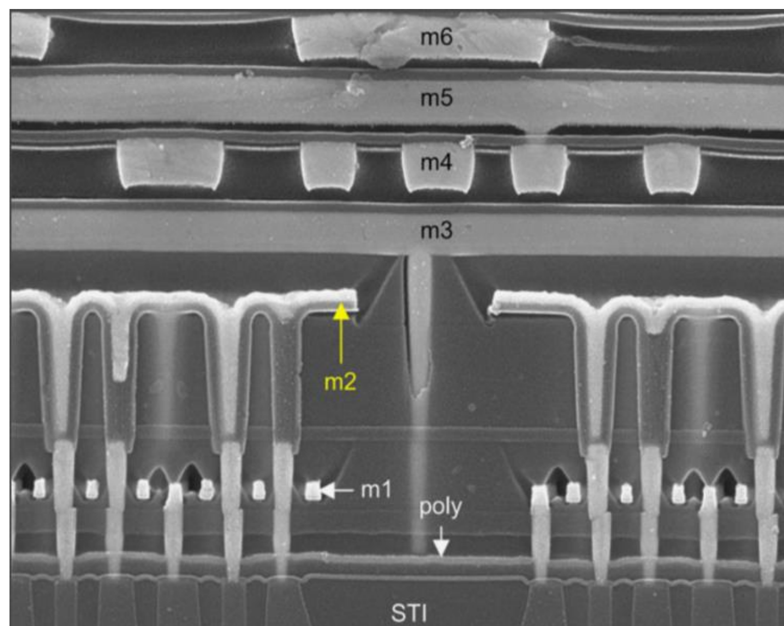


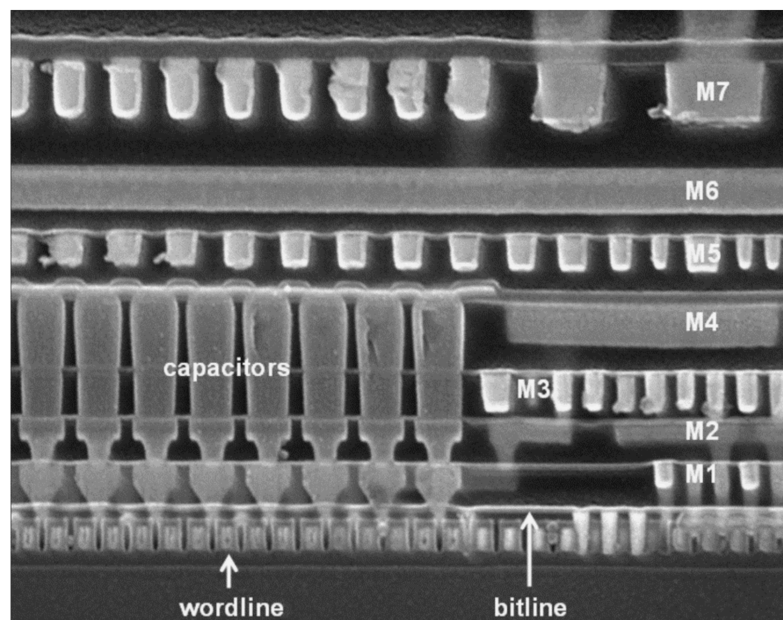
Figure 2.12: Layer stacking of chip in three different cases [42].

When enlarging the post-CMP profile in a photo under electron microscope, as shown in **Figure 2.13**, we can see the thickness variation of each layer. This demonstrates topography variation on silicon wafer still exists. Therefore, despite being a predominant planarization technique, CMP is known to suffer from undesired pattern dependent problems. A non-uniform feature density distribution on each layer causes CMP to over/under polishing, generating metal dishing and dielectric

erosion (CMP side effect), which results in the failure of interconnects and yield drop.



(a)



(b)

Figure 2.13: Photo under electron microscope shows post-CMP thickness variation of each layer [43]. (a) Thickness variation of layer in silicon topography for an analog circuit. (b) Thickness variation of layer in silicon topography for a digital circuit.

2.2.2 Investigation for Density Issue Handling

Chip manufacturing is largely dependent on the features of device and interconnect in deep-submicron technology. The quality of CMP is highly related to the uniformity of density contribution, and a predictable layout is desirable for good CMP performance [44]. The density distribution to affect profile of silicon shape is shown in **Figure 2.14**.

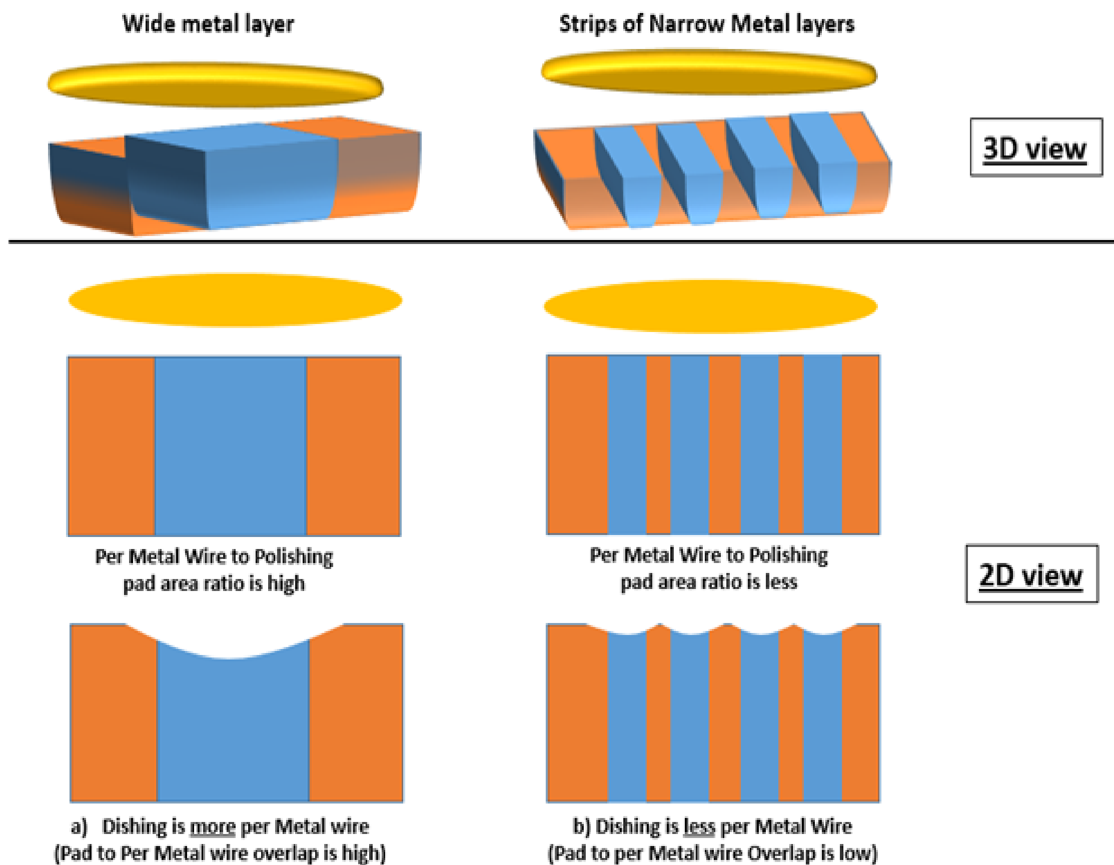


Figure 2.14: Density variation to the CMP profile [45].

In general, density requirements provided by foundry is a set of ranges that define maximum and minimum values for each layer. If the density of a layer is over maximum value, there will cause over polishing on the silicon. If the density of a layer is under minimum value, there will cause under polishing on the silicon. Only the density that falls in the range is considered to be safe. The layer types for density checking vary with the technology process and rule documentation provided by the semiconductor foundry.

In the technology process node 130nm and above, CMP affects only the back-end-of-line (metal layers). The density rule for the front-end-of-line (diffusion and poly) does not consider for CMP variation. Contact and via layers are not restricted by density rule. Most copper processes employ dual-damascene process and CMP is not done for contact/via layers. With the decreasing process node, those layers previously ignored by density checking, however, are becoming more important, as the effects induced by physical limitation become more significant. It is inevitable to consider the layers required for density checking in rule documentation, regardless of what mechanism they base on. Note that in our research, we consider 5 layers that combine different mechanism resulting in the density variation for layout pattern.

Our research focuses on manufacturability and yield issue arising from layout pattern density. Rule documentation just indicates layer type which is mandatory to check for density and density level that layout pattern must reach. It cares not for the root-cause or mechanisms behind the density issue. Layout is evaluated as a fine design as long as density meets required constraints.

In the technology node 130nm and above, the density rule for layout design is simple and generally, for metal, density control is easily achieved. Advanced technology nodes are requiring even more complex density checking, a basic check for diffusion/poly/metal is mandatory (In our used CMOS process, density checking for contact is also mandatory). We believe that the density rule for diffusion and poly are relevant with CMP variation. Many literatures point out that the density of diffusion and poly affects the CMP quality [46], which in return affects the function of the circuit.

As for poly, just assuming that if a layout has small tiny poly structure far away from other poly structures, this small poly will get etched more than the other poly. Thus, the layout has problem with the uniformity of the surface for the next process step. This isolated poly also can be easily cracked under extreme temperature or voltage.

As for diffusion, the CMP process for the STI (shallow trench isolation) has been optimized as a trade-off between junction leakage and transistor leakage (hump effect). When the STI to ACTIVE step is too high, junction leakage happens. When the STI/ACTIVE step is too negative, transistor leakage increases. The STI step uniformity is depending upon the ACTIVE density uniformity. This is why insertion of dummy active areas is mandatory if density constraints as described in the density rule are not reached.

In the advanced 65-nm technology process, it is effective to combine variations arising from different layers. We think that if we only consider metal layers, the density optimization for the yield improvement will become pointless as poly and diffusion can also affect manufacturability. Besides, we introduce a weight parameter into the objective function, in order to distinguish the priority and importance of each layer.

Although in mechanism point of view, CMP is not done for contact/via layers. In the dual-damascene process, trench for metal deposit and hole for contact/via are formed to a combination. However, it still indirectly affects the topography of upper layer to be polished. As advanced manufacturing process requires multiple parallel vias/contacts to ensure reliable connectivity among layers, verifying the existence of sufficient vias/contacts in the layout becomes necessary. In the perspective of manufacturability, we have to take contact (in our work, the density rule requires only check for contact) into account. In designing layout pattern of advanced technology nodes, designers always try to extend the enclosure of the diffusion area when possible, since overlay may make that one contact falls on the border of the diffusion area, thus generating a junction leakage. Designers also follow DFM guideline to double contact and extend poly and metal 1, in order to reduce the electro-migration effect and risk of open circuits. In our objective function, the weight for contact is relatively low compared with other layers, because it has a small impact on the manufacturing of the given pattern. In addition to the consideration of layer type, the handling method for solving the density is also important.

To solve high-density problem, there usually takes measures like:

- Ripping up the layout and making spaces between features larger to decrease density.
- Splitting wide interconnect into multiple lines.
- Slotting on the area where the size of feature is large.
- Reducing dummy fills or features that have no impact on electrical behavior of the circuits.

To solve low-density problem, there are many place and route tools available for digital circuits, mainly inserting dummy fills in the empty spaces to increase density. In the past decades, there were many papers in EDA (electronic design automation) domain proposing effective methods to insert dummy fills. At the same time, they still consider parasitic effect like coupling capacitance. Some papers adopt effective algorithms to formulate model based on the CMP process, then to do density analysis by model.

Although they have made great contributions to solve density issue, their methods are just applicable to digital circuits. As for analog circuits, there are few automated tools to provide features for layout design.

As the process nodes decrease, circuits are becoming larger while spaces between features are becoming smaller, the limitations of fill insertion become significant. For instance, in the chip assembling before tape-out. There are density violations in blocks over the layout, however, all blocks are complete, and positions are yet fixed. In this situation, to insert dummy fills is difficult and miserable to designers. Hence, fill insertion seems to have reached its bottleneck at advanced nodes.

Previously, fill insertion was an effective way to solve density issue. But now, it’s reaching to its limitations at advanced nodes. Challenges for now to solve density issue are summarized as following.

- Many EDA tools provide powerful features to address density issue, while few options are available for analog layout.
- Fill insertion becomes more aggressive and dummy fill becomes closer to signal lines as the process nodes decrease.
- Dummy fills bring unexpected parasitics that significantly affect the electrical characteristics of the circuit.
- Tuning circuit and redrawing layout for fill insertion severely influence the yield and the time-to-market.

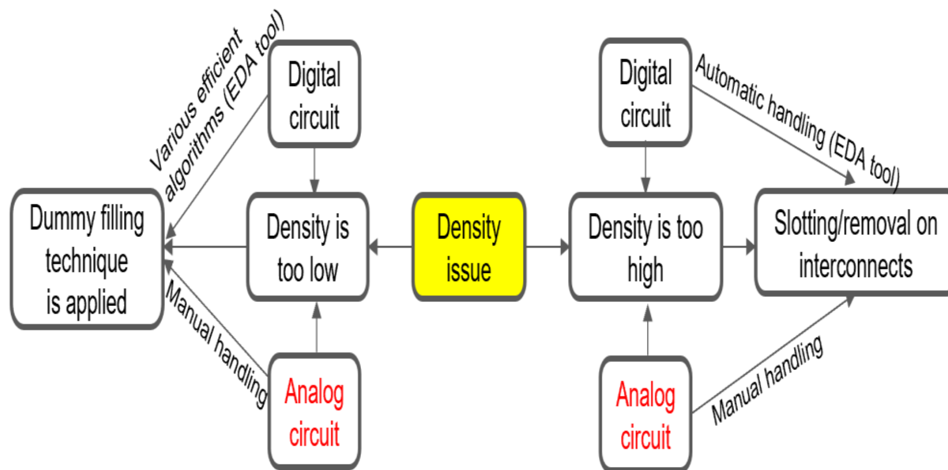


Figure 2.15: Density issue handling in digital and analog domains.

Figure 2.15 shown above summarizes the density handling in digital and analog domains. To compensate the CMP variability and topography variation, layout techniques such as dummy filling for sparse region and slotting/removal on interconnects for dense region are applied to control the layout pattern density.

For the density issue in digital circuits, EDA tool is very convenient to handle the issue, and most of the works consider optimizing the amount of the fills and accelerating the filling process by algorithm. For the density issue in analog circuits, however, layout designers have to handle the issue manually as there are few useful tools.

Herein, we emphasize the drawback of dummy feature filling in analog layouts by using cases of a digital circuit and an analog circuit, respectively.

Figure 2.16 shows dummy fill insertion in a digital layout which is a logical module for an ADC (analog-to-digital converter) in a 65nm CMOS process. As seen from the figure, some empty regions in layout are filled up with the metal dummy features, in order to reduce the inter-layer dielectric (ILD) thickness variation. As such, the layout pattern density can be uniform and the silicon topography in each layer can become smooth.

Since an EDA tool provided an efficient way, dummy feature is automatically filled up and density checking is easily passed in just few minutes.

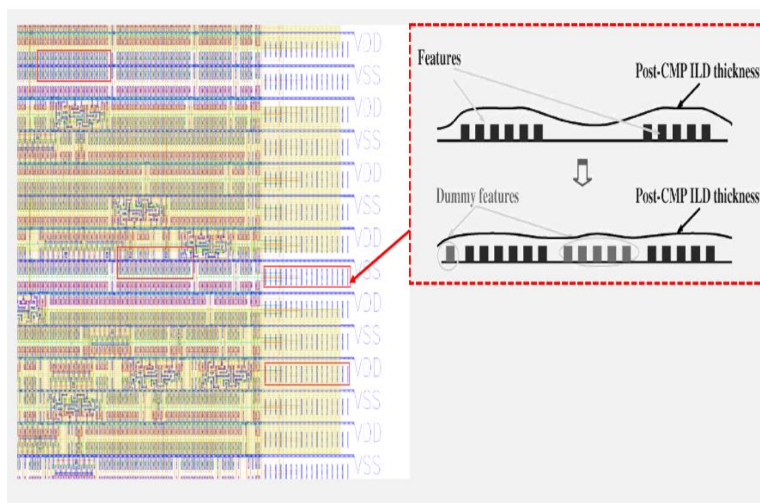


Figure 2.16: Dummy fill insertion in a digital layout.

However, in a dummy fill insertion for diffusion layer of an analog layout designed for a low-pass filter in a 65 CMOS process, as shown in **Figure 2.17**, it spends several days to pass the density checking.

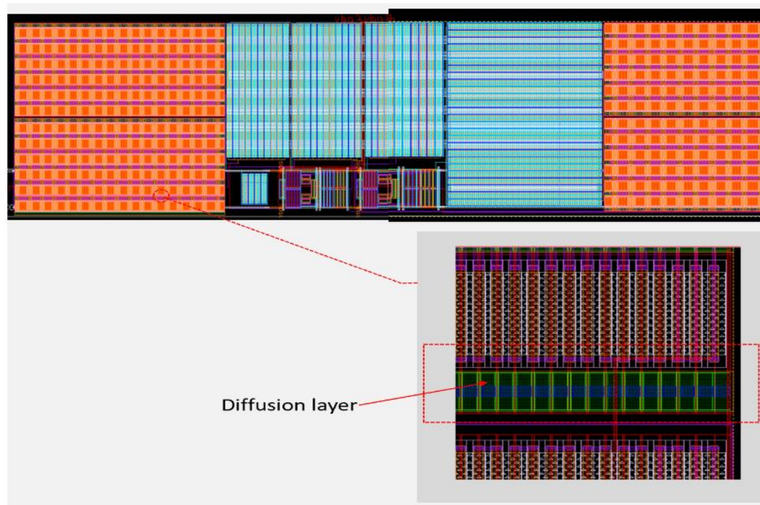


Figure 2.17: Dummy fill insertion for diffusion layer in an analog layout.

Because in a limited given layout size, there is no more empty space to fill diffusion layer. Meanwhile, analog layout designer needs to calculate the diffusion density that can pass the density checking. Most importantly, the designer has to ensure that manual insertion in resistor array for the diffusion layer does not introduce other DRC violations.

The density checking is finally passed until an appropriate density value is properly calculated and dummy fills are carefully added in empty regions. The experience of dummy fill insertion in an analog layout is really suffering.

In summary, comparisons between the digital and analog circuits are as follows:

- Density-related research is always a focus in the field of computer-aided design (digital circuit).
- Various algorithms provide efficient ways to control layout pattern density (digital circuit).
- Dummy filling is error-prone and introduces unexpected parasitics to sensitive signals or devices (analog circuit).
- Tuning the dimension and position of device layers is time-consuming and costly (analog circuit).

Therefore, the present method to address density issue for analog circuits is of low efficiency and low reliability, providing an efficient approach for analog layout to handle the layout pattern density has a great significance.

2.2.3 Layout Density Uniformity

To improve the CMP quality, layout design must comply with density rules and fill dummy features to restrict the variations on each layer.

Local pattern density within every predefined window must be within a specified range, these density bounds can help in minimizing the multi-layer accumulative effect. However, unbalanced wire distribution still exists even layout pattern density satisfies the constraints. Density variation among neighboring subregions impacts topography, thereby influencing

the CMP quality and the yield. Hence, it is not enough to satisfy the density constraints only. Seeking for the minimum wire-density gradient can further improve the yield, which is the objective of our algorithm after-mentioned.

As shown by the example in **Figure 2.18**, aerial view for layout pattern and lateral view for wafer topography are given respectively. If the density lower and upper bounds are 20% and 80%, respectively. Wafer topography variation is reduced after inserting dummy fills in empty regions, whereas the feature distribution in two subregions can be different even their densities are same (see **Figure 2.18(a)**). In **Figure 2.18(b)** and **Figure 2.18(c)**, the four adjacent tiles all satisfy density constraints. However, **Figure 2.18(c)** is better for CMP control because it has the minimum wire-density gradient. Thus, density uniformity is critical to optimize the yield.

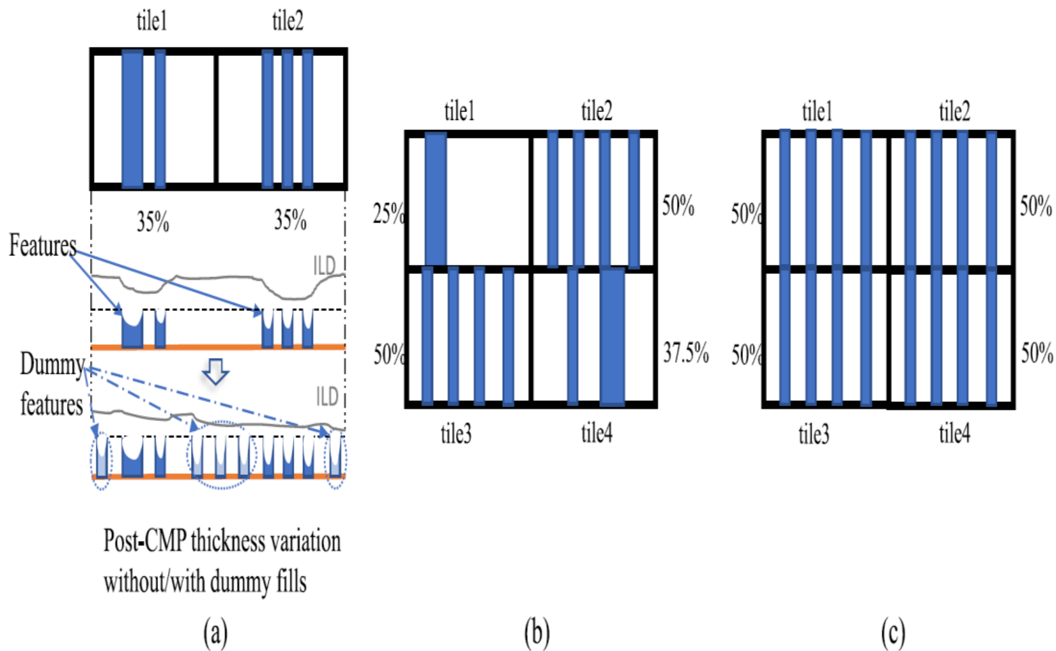


Figure 2.18: Density variation among neighboring subregions impacts wafer topography. (a) Different wire distribution in a subregion exists even under the same density. Large density variation among neighboring subregions leads to post-CMP thickness irregularities. (b) Four adjacent tiles all meet density constraints but result in an unbalanced wire distribution. (c) Reducing density gradient among tiles contributes to uniform topography.

2.3 Transistor Array

In MOS analog layout, [24] addresses the layout-dependent variability based on the measurement results of test chips on a 90nm CMOS process. As shown in **Figure 2.19(a)**, when increasing the channel size, i.e., $L \times W$, the variation decreases. This is consistent with the Pelgrom model [25].

However, for two transistors with the same channel length and width, if they have different layout structures, the difference of V_{th} might be bigger than that of the transistors with the same structure. This result reveals that the transistors with unified channel length and channel width can alleviate the layout-dependent variation as expected.

Yang et al. [26] proposes transistor-array(TA)-style for analog layouts. As an extended research of TA, Liu et al. [47] presents a twin-row layout style for transistors-pair with the matching feature and routability.

In TA-style, a large transistor is decomposed into a set of unified sub-transistors, which are connected in series or parallel. Since the transistor decomposition in the channel length direction does not introduce a significant error, all the sub-transistors are then able to be arranged on a uniform grid like an array, thereby obtaining a well-structured layout as illustrated in **Figure 2.19(b)**.

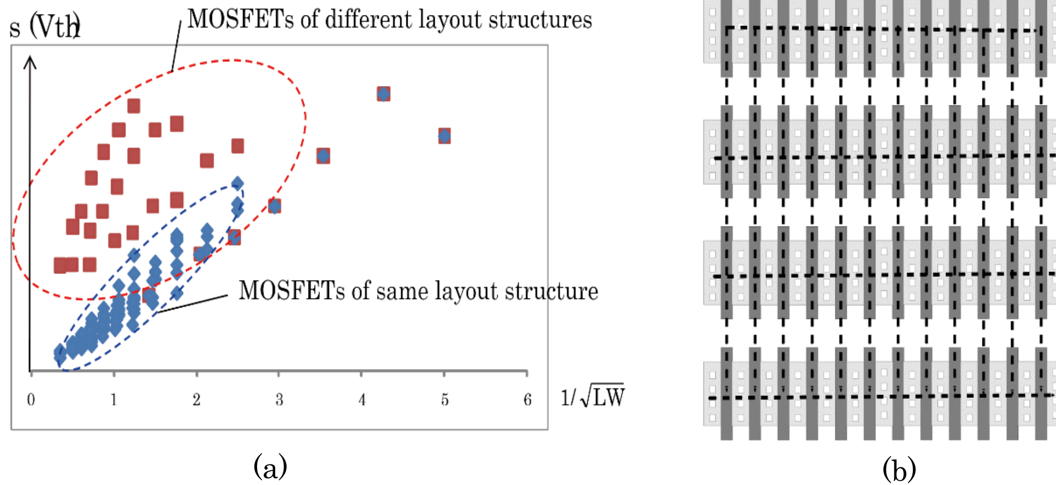


Figure 2.19: Transistor array. (a) The spatial-dependent variation in Vth for the transistors with the same/different layout structures. The boxes of red represent the spatial-dependent variation with different layout structures, and the boxes of blue are with the same structure. (b) Unified transistor

With such an array-based structure, a better post CMP profile is expected to be achieved as well [26], and the STI (Shallow Trench Isolation)-stress is evened up.

The works introduced in [24]-[26] clarify that, analyzing the DC/AC measurement results from the test chip, the channel decomposition of the MOS transistor, it does not show too much difference between the decomposed transistor and the original transistor. Therefore, if the design does not require very strict electrical characteristics, the channel decomposition of the MOS transistor, as well as TA-style layout are applicable to analog designs.

Actually, we use a diffusion-shared structure as shown in for layout generation. This structure has also reportedly shown good capability to suppress mismatch in Ids. On the contrary, Irregular structures among neighboring transistors lead to a bad post-CMP profile, and cause uneven STI-stress, as seen in **Figure 2.20**.

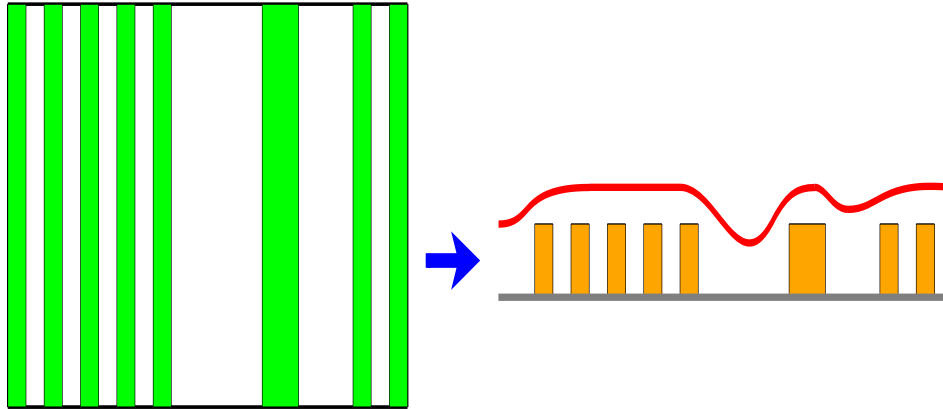


Figure 2.20: Irregular structures among neighboring transistors lead to post-CMP thickness irregularities, i.e., a bad post-CMP profile, and cause uneven STI-stress.

2.4 Our Proposal to Address Density Issue

Present challenges to solve density issue are summarized as below.

- Many EDA tools provide powerful features to address density issue, while few options are available for analog layout.
- Fill insertion becomes more aggressive and dummy fill becomes closer to signal lines as the process nodes decrease.
- Dummy fills bring unexpected parasitics that significantly affect the electrical characteristics of the circuit.
- Tuning circuit and redrawing layout for fill insertion severely influence the yield and the time-to-market.

Focus on these challenges, this work proposes a method to deal with them.

- Density-aware format enables designers to evaluate and adjust density level earlier, ensuring density predictable and controllable.
- Constraint (DRC and Density)-driven design makes layout highly conform to requirements, reducing iterations for verification.
- TA-style layout enhances the flexibility of design, where layout can be adjusted by changing the array pitch,

stretching the poly gates or widening the diffusion of unit-transistors.

In our proposal of this work, a framework dedicated to a fully automated solution for analog IC design, is used to obtain the optimal device/pattern parameters for density optimization and construct a TA-style design flow for analog layout generation.

Based on the transistor array and the density checking procedures in layout verification, we propose to partition a given layout area into identical tiles where a tile is filled up with a transistor array, so that any regions covered by a checking window can pass the density checking.

We then define the device/pattern parameters that describe a density-aware layout format of a transistor array. Then, based on a 65nm CMOS process, we propose a density optimization objective function for a transistor array that is subject to the formulated DRC and density constraints, the objective is for the density uniformity of layout pattern and uniform density gradient among tiles. An efficient mathematical optimization approach is used to simplify the problem and find the optimal device/pattern parameters.

Then, once the optimal parameters are obtained, a TA-style analog layout design flow is proposed, which consists of circuit partition, floorplanning, placement, and routing. The design flow fully conforms to the common layout design flow under consideration of the matching and symmetry constraints.

Finally, layout design examples of full-transistor OPAMP circuit, an automatic layout under the TA-style design flow incorporating with density optimization, and a manual layout by a traditional method, are used to demonstrate the effectiveness of our proposed method.

Post-layout simulation results of both layouts are performed on a Star-RCXT platform, which provides convincing comparison results as it is the industry standard for the silicon-accurate and high-performance parasitic extraction of advanced process technologies.

Both layouts are thoroughly compared with respect to the major metrics of the OPAMP circuit, consisting of DC, AC transient performance specifications. Merits and demerits of an automatic layout by our method are discussed sufficiently.

Figure 2.21 shows the flowchart of our proposal to address the density issue of analog layout. In particular, we emphasize the design example of the analog layout and post-layout simulation steps for demonstrating the effectiveness of our method. The design example used in this research is an operational amplifier (OPAMP).

As the most commonly used circuit in the analog domain, OPAMP is very convenient and suitable for prototypes to demonstrate the feasibility of whole research. Once the whole flow is demonstrated to be feasible, we can progress the research to a higher level where more complicated circuits would be used. The parasitic extraction is performed on Star-RCXT platform, then the netlist files with parasitics are delivered to Cadence, where post-layout simulation is done through Virtuoso. Conclusion and analysis can be drawn by comparing the metrics of both layouts.

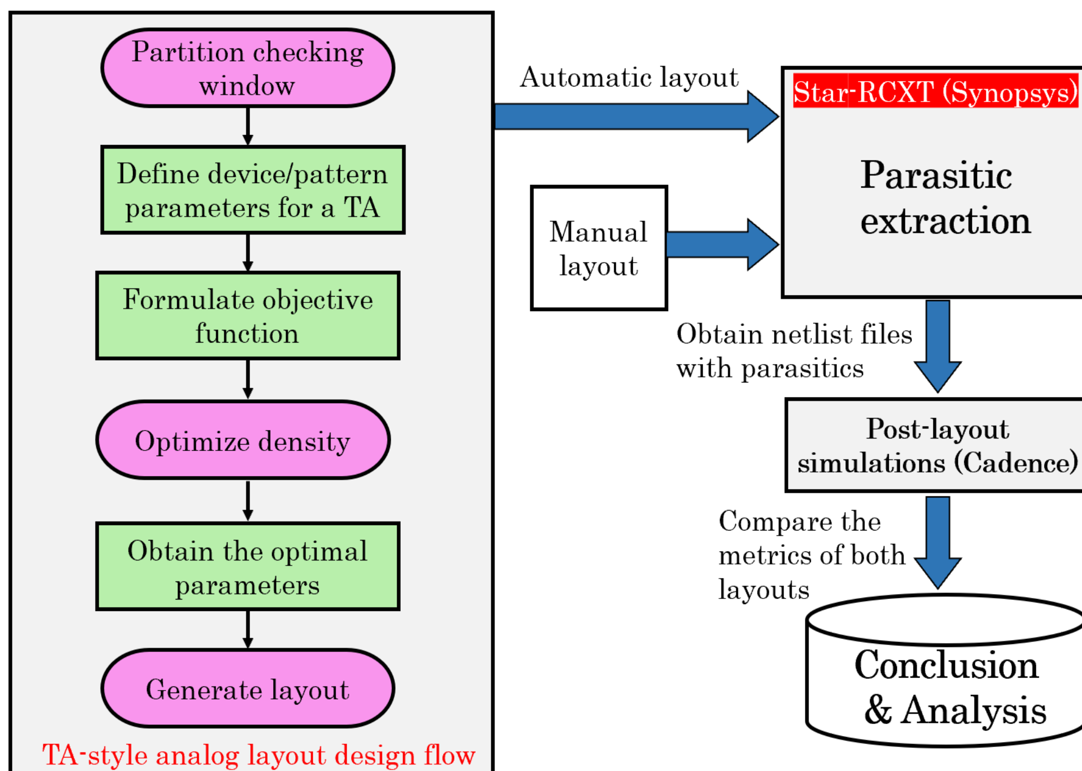


Figure 2.21: Flowchart of our proposal.

2.4.1 OPAMP Circuit for Design Example

Operational amplifier, or OPAMP for short, is a basic building block of analogue electric circuits. OPAMP is a DC-coupled high-gain electronic voltage amplifier with differential inputs and, usually, a single-ended output.

An OPAMP generally has three terminals (excluding power supply terminals). One of the inputs is called the inverting input, marked with a negative or “minus” sign (-). The other input is called the non-inverting input, marked with a positive or “plus” sign (+). In this configuration, an OPAMP produces an output potential (relative to circuit ground) that is typically 100,000 times larger than the potential difference between its input terminals. In a linear operational amplifier, as the input stage of an OPAMP is in fact a differential amplifier, the output signal is known as the amplifier’s gain (-A) multiplied by the value of the difference ($V_2 - V_1$) between the two input signals and is depending on the nature of these input and output signals.

Symbol of OPAMP and its equivalent circuit are shown in **Figure 2.22**. Where V_1 denotes the inverting input and V_2 denotes the non-inverting input. +V_{supply} and -V_{supply} are power supplies. Z_{in} and Z_{out} denotes the input impedance and the output impedance of OPAMP, respectively. The output is denoted by V_{out}.

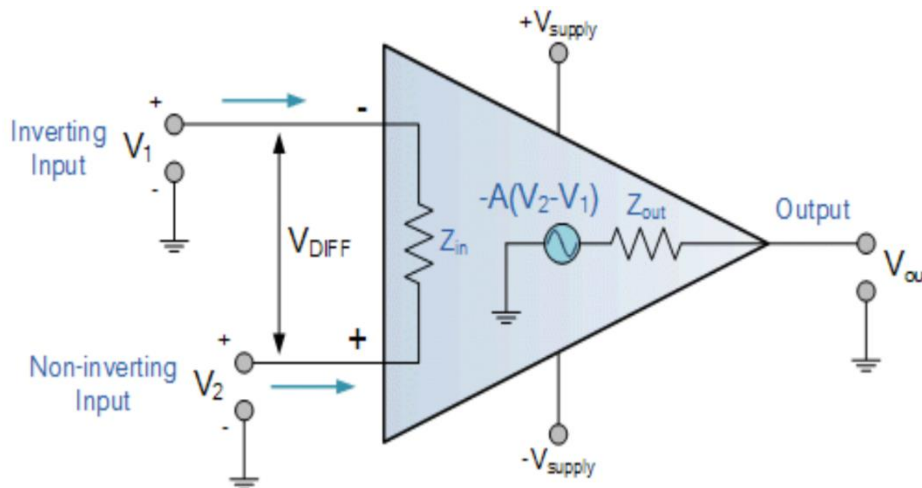


Figure 2.22: Symbol of OPAMP and its equivalent circuit.

Operational amplifiers have their origins in analog computers, where they are used to perform mathematical operations such as add, subtract, integration and differentiation, in many linear, non-linear, and frequency-dependent circuits.

Due to its versatility, OPAMP is popular as a building block in analog circuits. By using negative feedback, the characteristics of an OPAMP circuit, its gain, input and output impedance, bandwidth, etc. can be determined by external components. The characteristics are nearly independent of temperature coefficients or engineering tolerance in the OPAMP itself. In a vast array of consumer, industrial, and scientific devices, OPAMPs are the most widely used electronics today, circuit designers can configure an OPAMP circuit with negative feedback constituted by resistors, capacitors, or both. The OPAMP circuit is capable of handling signal amplification, filtering, or arithmetic circuit operations described above, it can also be used to form various functional circuits using different resistors and capacitors as well as configurations. Such as differential OPAMP, summing OPAMP, differentiator OPAMP, integrator OPAMP, non-inverting amplifier, inverting amplifier, and voltage follower circuit. In the active filters and analog-to-digital converters (ADCs), OPAMP is employed as well.

An OPAMP CMOS circuit is an essential element in the analog integrated circuits, and therefore is very suitable for a design example of our density optimization for analog layout based on transistor array. Note that in this research, OPAMP circuit is completely constituted by transistors, as we first demonstrate the feasibility of our proposed design flow in a prototyping algorithm and then attempt to improve its generality in the future works.

The complexity of design example would increase by considering the resistor arrays and capacitor arrays. For this research, because the effectiveness of our method is demonstrated by the comparison results of a manual layout and an automatic layout, it is fundamental to understand the various metrics with regard to electrical performance.

- **Static power dissipation**
  We first discuss DC power dissipation and explain the calculations for this. The first part of power dissipation is the quiescent power that is dissipated due to quiescent current and supply voltage. By simply multiplying the total supply voltage $(+V_s-(-V_s))$ by the quiescent current I_q, we attain the quiescent power dissipation P_q by the following formula.

$$P_q=I_q \bullet (+V_s-(-V_s)) \tag{2.1}$$

- **Input common-mode range (ICMR)**
  ICMR is a key parameter important for all OPAMP applications in circuits, and it is one of the first terms of which an analog designer thinks. V_{ICM} describes a particular voltage level and is defined as the average voltage at the inverting and non-inverting input ports, V_{in-} (V_1) and V_{in+} (V_2). V_{ICM} is expressed as follows.

$$V_{ICM}=(V_{in+}+V_{in-})/2 \tag{2.2}$$

  In most applications, V_{in+} is very close to V_{in-} because closed-loop negative feedback causes one input port to closely track the other such that the difference between two inputs is close to zero. ICMR is defined as a range over which OPAMP circuit can work normally. An OPAMP whose ICMR ranges from V_{SS} to V_{DD} is called "rail-to-rail input operational amplifier", meaning an OPAMP with an excellent input signal voltage range.

- **Output swing**
  Under defined operating conditions where the OPAMP still can function correctly, output swing defines how close the OPAMP output can be driven to rail to rail (either power rail: VDD or VSS), as shown in **Figure 2.23**. To determine the amount of current that the amplifier is sinking or

sourcing, comparing voltage output swing specifications is the key. The smaller the output circuit current is, the closer the amplifier would swing to the rail. The voltage output swing capability of an OPAMP is dependent on the OPAMP output stage design and the load current.

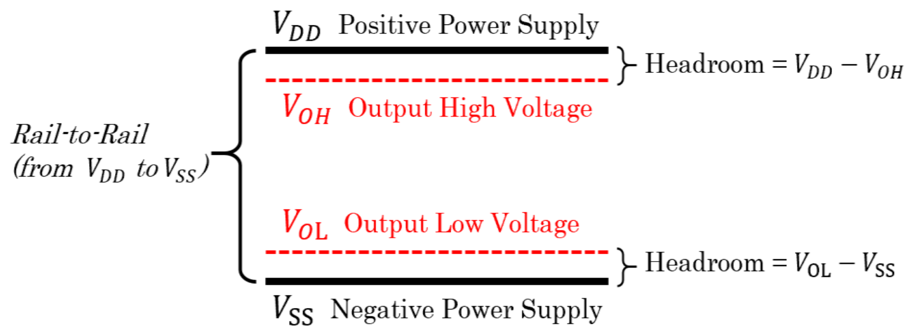


Figure 2.23: Voltage output swing.

- **Input offset voltage**
  In the case of the ideal OPAMP, the DC voltage of the V_{in+} and V_{in-} terminals match exactly when the input common-mode voltage V_{ICM} is 0 V. In reality, however, there are differences in input impedance and input bias current between the input terminals, causing a slight difference in their voltages. This difference called input offset voltage is multiplied by a gain, appearing as an output voltage deviation from the ideal value 0 V. When used in amplifiers of sensors, etc., the input offset voltage of an OPAMP results in an error of sensor detection sensitivity. To keep sensing errors below a specified tolerance level, it is necessary to select an OPAMP with low input offset voltage. The input offset voltage actually reflects the circuit symmetry inside the OPAMP. The better the symmetry is, the smaller the input offset voltage is. Input offset voltage is a very important performance parameter of operational amplifier, especially when it is used in high-precision OPAMP or DC amplifier. The input offset voltage has a certain relationship with the manufacturing process, and the input offset voltage of bipolar process (i.e. the standard silicon process) has a certain relationship with the manufacturing process. The input offset voltage would be larger if the FET is used as the input stage. For high-

precision operational amplifiers, the input offset voltage is generally less than 1mV. Additionally, input offset voltage is parameter associated with temperature.

- **DC/Open-loop gain**

The main function of an OPAMP is to amplify the input signal and the more open-loop gain it has, the better. When overall feedback is excluded from the circuit, the DC/open-loop gain of an operational amplifier can be obtained. Open-loop gain, in some amplifiers, can be exceedingly high. An ideal OPAMP has infinite open-loop gain. Typically, an OPAMP may have a maximal open-loop gain of around 10^5. To achieve the desired performance, the very high open-loop gain of the OPAMP allows a wide range of feedback levels to be applied. Normally, feedback is applied around an amplifier with high open-loop gain so that the effective gain is defined and kept to a desired figure. At a fixed frequency, the open-loop gain can be represented as follows.

$$A_{OL} = \frac{V_{out}}{V_{in+} - V_{in-}} \tag{2.3}$$

Where, $V_{in+} - V_{in-}$ is the voltage difference being applied to the input terminals. The following **Figure 2.24** shows the gain of OPAMP with respect to frequency, where R_f is a feedback resistance and R_{in} is an input resistance.

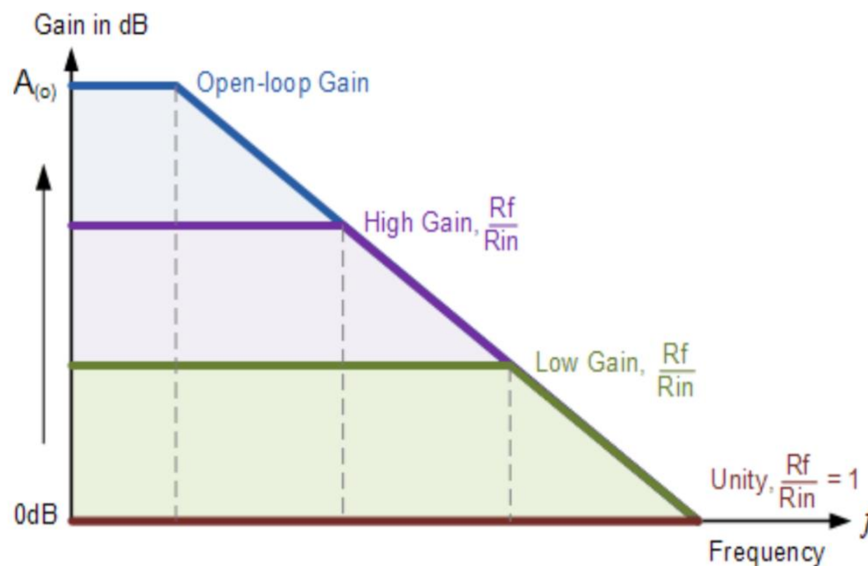


Figure 2.24: An example of gain of OPAMP with respect to frequency.

- **Phase margin**

  The phase margin PM is a measure for the stability of a system with feedback. The higher the phase margin, the more stable the system. Capacitive loading will reduce the phase margin. The phase margin (PM) is the difference between the phase lag φ (< 0) and -180°, for an amplifier's output signal (relative to its input) at zero dB gain or output is same as of input. The phase margin PM is expressed as follows.

$$PM = \varphi - 180^{\circ} \tag{2.4}$$

  For example, if the amplifier's open-loop gain crosses 0 dB at a frequency where the phase lag is -135°, then the phase margin of this feedback system is -135° - (-180°) = 45°. In practice, feedback amplifiers must be designed with phase margins substantially in excess of 0°, even though amplifiers with phase margins of, say, 1° are theoretically stable. However, many practical factors can reduce the phase margin below the theoretical minimum. A prime example is when the amplifier's output is connected to a capacitive load. Therefore, operational amplifiers are usually compensated to achieve a minimum phase margin of 45° or above.

- **Gain-bandwidth product (GBP)**

  The gain-bandwidth product (GBP) for an amplifier is the product of the amplifier's bandwidth and the gain at which the bandwidth is measured. For an amplifier in which negative feedback reduces the gain to below the open-loop gain, the gain–bandwidth product of the closed-loop amplifier will be approximately equal to that of the open-loop amplifier. This quantity is commonly specified for operational amplifier deign, and allows circuit designers to determine the maximum gain that can be extracted from the device for a given frequency (or bandwidth) and vice versa. **Figure 2.25** shows the frequency response curve of the product of the gain against frequency, we can see that GBP is constant at any point along the curve.

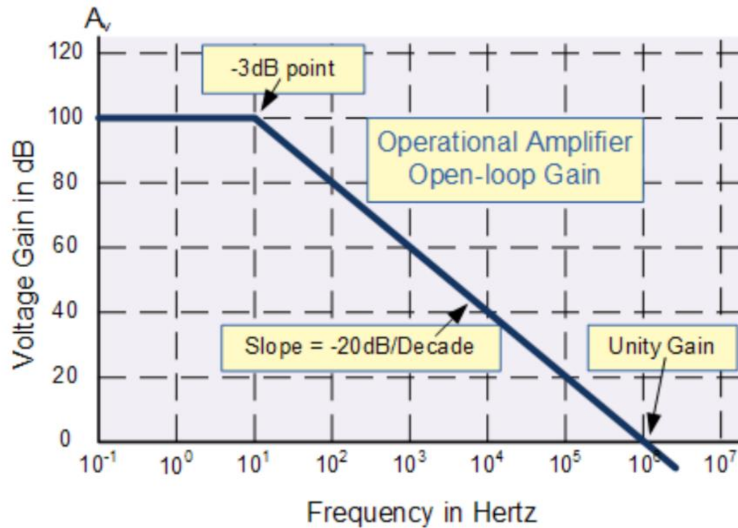


Figure 2.25: Frequency response curve of gain against frequency for an OPAMP.

We can also see that the unity gain (0dB) frequency also determines the gain of the amplifier at any point along the curve. Therefore, we have the formula as follows.

$$GBP = A \times BW \tag{2.5}$$

Where A is the gain of OPAMP, and BW denotes the bandwidth. For example, from the graph above the gain of the amplifier at 100kHz is given as 20dB or 10, then the gain bandwidth product is calculated as GBP = 10^6. Similarly, the operational amplifiers gain at 1kHz = 60dB or 1000, therefore the GBP is given as GBP = 10^6. We can see the results are same.

- **Common-mode rejection ratio (CMRR)**
  The common mode rejection ratio (CMRR) of a differential amplifier (or other device) is a metric used to quantify the ability of the device to reject common-mode signals, i.e. those that appear simultaneously and in-phase on both inputs. The CMRR is the most important specification and it indicates the how much of the common mode signals present to measure. The value of the CMMR frequently depends on the signal frequency and the function should be specified. The function of the CMMR is specifically used to reduce the noise on the transmission lines. An ideal differential amplifier would have infinite CMRR, however this is not achievable in practice. A high CMRR is required

when a differential signal must be amplified in the presence of a possibly large common-mode input, such as strong electromagnetic interference (EMI). An example is audio transmission over balanced line in sound reinforcement or recording. Ideally, a differential amplifier takes the voltages, V_{in+} and V_{in-} on its two inputs and produces an output voltage $V_{out} = A(V_{in+} - V_{in-})$, where A is the differential gain. However, the output of a real differential amplifier is better described as:

$$V_{out} = A(V_{in+} - V_{in-}) + \frac{1}{2} A_{CM}(V_{in+} + V_{in-}) \tag{2.6}$$

Where A_{CM} is the common-mode gain, which is typically much smaller than the differential gain. The CMRR is defined as the ratio of the differential gain over the common-mode gain, measured in positive decibels. It is expressed by the following formula:

$$CMRR = 20 \log_{10}\left(\frac{A}{|A_{CM}|}\right) dB \tag{2.7}$$

As differential gain should exceed common-mode gain, this will be a positive number, and the higher the better.

- **Supply voltage rejection ratio (PSRR)**

Supply voltage rejection ratio (PSRR) is defined as the ratio of input offset voltage to supply voltage when OPAMP operates in linear region, which is a term often expressed in decibels. The PSRR reflects the influence of power supply variation on the output of OPAMP, and it is widely used to describe the capability of an electronic circuit to suppress any power supply variations to its output signal. Therefore, the power supply of operational amplifier needs careful treatment when it is used in DC signal processing or small signal processing for analog amplification. Of course, the OPAMP with high CMRR can compensate a part of PSRR. In addition, when using dual power supply, the PSRR of positive and negative power supply may be different. An ideal OPAMP would have infinite PSRR. The output voltage will depend on the feedback circuit, as is the

case of regular input offset voltages. But testing is not confined to DC (zero frequency), often an operational amplifier will also have its PSRR given at various frequencies. Some manufacturers specify PSRR in terms of the offset voltage it causes at the amplifiers inputs; others specify it in terms of the output; there is no industry standard for this issue. The following formula assumes it is specified in terms of output:

$$PSRR=20log_{10}\left(\frac{\Delta V_{supply}A}{\Delta V_{out}}\right)dB \tag{2.8}$$

Where A is the voltage gain. For example: an amplifier with a PSRR of 100 dB in a circuit to give 40 dB closed-loop gain would allow about 1 millivolt of power supply ripple to be superimposed on the output for every 1 volt of ripple in the supply.

- **Slew rate (SR)**

Slew rate is defined as the maximum rate of change of an OPAMP's output voltage and is given by units of volts per microsecond (V/μs). SR is measured by applying a large step voltage, such as 1V, to the input of the OPAMP, and measuring the rate of change from 10% to 90% of the output signal's amplitude. Although SR is not always mentioned, it can be a critical factor in ensuring that an amplifier is able to provide an output that is a faithful representation of the input. If SR is violated, some error might occur, and correct operation is no longer guaranteed. For example, when the input to a digital circuit is driven too slowly, the digital input value registered by the circuit may oscillate between 0 and 1 during the signal transition. In other cases, a maximum slew rate is specified in order to limit the high frequency content present in the signal, thereby preventing such undesirable effects as ringing. Since the input stage of the OPAMP is in the on-off state during the transition, the feedback loop of the OPAMP does not work, that is, the conversion rate is independent of the closed-loop gain. The conversion rate is a very important parameter for large

signal processing, for typical OPAMPs, SR < = 10 V/μs and for high-speed OPAMPs, SR > 10 V/μs. At present, the highest SR of high-speed operational amplifier is 6000 V/μ s. Thus, SR is used for the selection of OPAMP in large signal processing. In amplifiers, limitations in slew rate capability can give rise to non-linear effects. For a sinusoidal waveform not to be subject to slew rate limitation, the slew rate capability (in volts per second) at all points in an amplifier must satisfy the following condition:

$$SR \geq 2\pi f V_{pk} \tag{2.9}$$

Where f is the operating frequency, V_{pk} is the peak amplitude of the waveform. As an example, we take the scenario where an OPAMP is required to amplify a signal with a peak amplitude of 5 volts at a frequency of 25kHz. An OPAMP with a slew rate of at least $2\pi \cdot 25000 \cdot 5 = 0.785$ V/μs would be required.

- **Setting time**

  Settling time (as illustrated in **Figure 2.26**) of a dynamical system such as an amplifier or other output device is the time elapsed from the application of an ideal instantaneous step input to the time at which the amplifier output has entered and remained within a specified error band. Settling time includes a propagation delay, plus the time required for the output to slew to the vicinity of the final value, recover from the overload condition associated with slew, and finally settle to within the specified error.

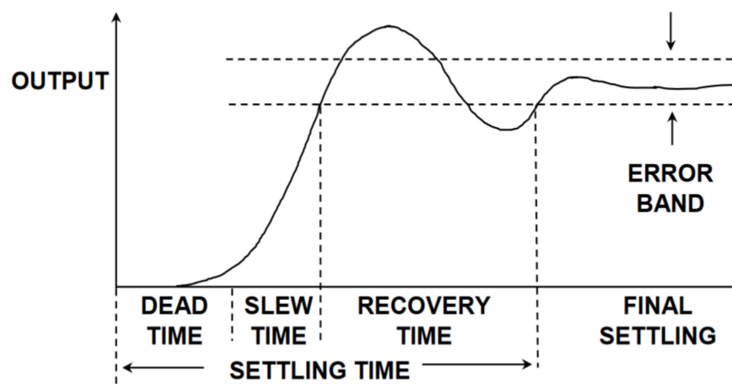


Figure 2.26: Settling time is the time required for an output to reach and remain within a given error band following some input stimulus.

2.4.2 Parasitic Extraction for Post-layout Simulation

In electronic design automation, parasitic extraction is calculation of the parasitic effects in both the designed devices and the required wiring interconnects of an electronic circuit: parasitic capacitances, parasitic resistances and parasitic inductances, commonly called parasitic devices, parasitic components, or simply parasitics.

The major purpose of parasitic extraction is to create an accurate analog model of the circuit, so that detailed simulations can emulate actual digital and analog circuit responses. Digital circuit responses are often used to populate databases for signal delay and loading calculation such as: timing analysis; power analysis; circuit simulation; and signal integrity analysis. Analog circuits are often run in detailed test benches to indicate if the extra extracted parasitics will still allow the designed circuit to function.

Interconnect resistance and capacitance are calculated by giving the extraction tool the following information: the top view layout of the design in the form of input polygons on a set of layers; a mapping to a set of devices and pins (from an LVS run), and a cross-sectional understanding of these layers including the resistivity of the layers. For the parasitic capacitance, this information is used to create a set of layout wires that have added capacitors where the input polygons and cross-sectional structure indicate. The output netlist contains the same set of input nets as the input design netlist and adds parasitic capacitor devices between these nets.

For the parasitic resistance, this information is used to create a set of layout sub-wires that have added resistance between various sub-parts of the wires. The above interconnect capacitance is divided and shared amongst the sub-nodes in a proportional way. Note that unlike interconnect capacitance, interconnect resistance needs to add sub-nodes between the circuit elements to place these parasitic resistors. This can greatly increase the size of the extracted output netlist and can cause additional simulation problems.

Basically, parasitic extraction provides the information about the parasitic devices which are not included as a part of original circuit design. But these parasitic devices affect the circuit performance in several ways. Due to these parasitic devices, the circuit may stop working or even not meeting design specifications. Therefore, a high-performance and accurate EDA tool for parasitic extraction will provide a reliable way to validate our research.

Examples for the effect of parasitic devices on circuit design are as follows:

- Extra power consumption.
- Affecting the delay of circuit, which can result in timing violation and impact IR drop.
- Reducing the noise margin which can cause logic failure.
- Increasing the signal noise.

With the extracted parasitics, we have the following benefits:

- During static timing analysis, parasitic extraction helps us to find out the R/C (delay) of the network, thereby helping us to do timing analysis.
- During noise analysis, crosstalk analysis, signal integrity check. For noise and cross talk analysis, it is important to the relationship between 2 wires and how these wires transfer the information between themselves. Coupling capacitance is the mode of interaction between them. Parasitic extraction helps us to find the coupling capacitance between 2 wires which helps us further to do SI (noise/crosstalk) analysis.
- In logic simulation, we need to know delay information and connectivity information. Parasitic extraction provides the netlist which has information of how different nets and devices are connected with each other. It helps us to do logic simulation.
- For IR analysis, resistance is one of the important considerations. Parasitic extraction outputs “resistance of the network” which help in IR analysis.

- In the analog design, a lot of noise through the substrate passes to other parts of the design. As we know, any channel through which any information can transfer has finite resistance. Parasitic extraction also helps to find the resistance of the substrate, which helps further for the substrate noise analysis.

In the technology nodes below 180nm, interconnect delay and coupling capacitance play a majority of role, therefore it is very important to extract this information correctly. However, the higher extraction accuracy for layout, the more time for extraction. There are three modes in parasitic extraction tool provided by different vendors so that user can extract only required information:

- Extract resistance only.
- Extract capacitance only.
- Extract resistance and capacitance both.

There are some very useful EDA tools available for parasitic extraction, in our research, we mainly focus on using Star-RCXT for our validation. On the one hand, it is a highly-efficient tool which provides accurate parasitic information and saves the time drastically. On the other hand, it is more convenient for us to use as it is already available in our research context.

Star-RCXT has become a popular parasitic parameter extraction tool in the industry due to its high, friendly user interface and good combination with other tools. It is specially designed for parasitic parameter extraction of processes of 0.18 μm and below. It uses 2.5-dimensional geometric extraction technology to achieve three-dimensional extraction, but it is much faster than three-dimensional extraction tools, and can quickly and accurately extract global parasitic parameters for millions of gate designs.

To apply Star-RCXT for fine parasitic parameter extraction, two files related to process parameters are also needed: mapping file and ITF (interconnect technology format) file. The ITF file is directly provided by the foundry. The information it contains

mainly includes: the various levels of the process (including dielectrics, vias and metal wires, etc.), and the physical dimensions of the thickness and width of each level in the process. Electrical parameters at various levels (such as dielectric constant, block resistance, etc.). The environment and files required for Star-RCXT can be explained by the design flowchart shown in **Figure 2.27**.

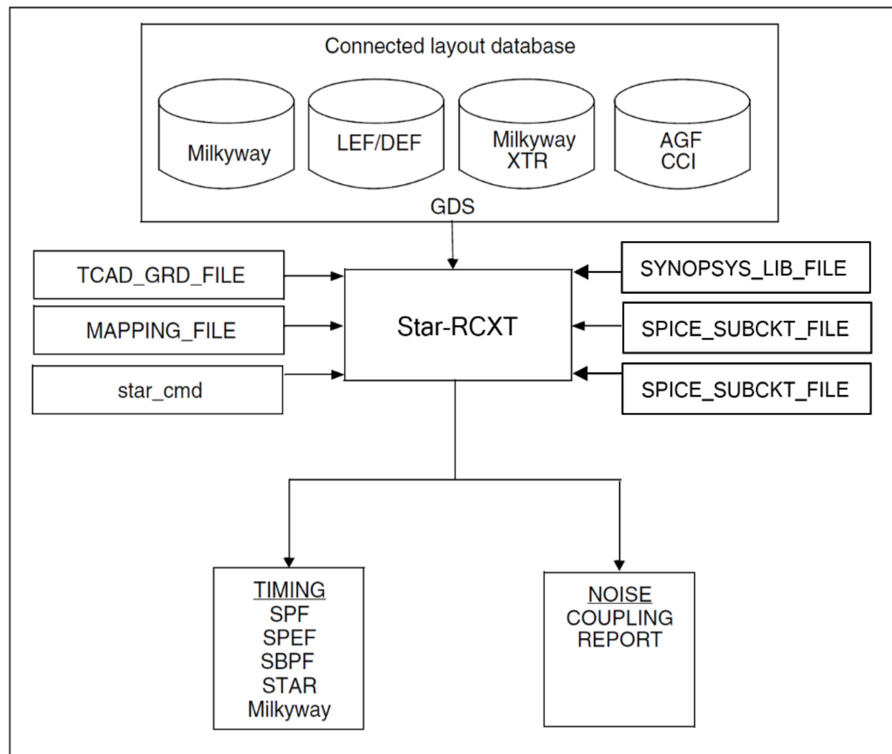


Figure 2.27: Design flowchart of parasitic extraction based on Star-RCXT.

As shown in Figure above, Star-RCXT can read directly the database generated by the process of Milkway, LEF/DEF, calibre connectivity interface (CCI) and Hercules.

TCAD_GRD_FILE is a file with an extension of nxtgrd, it includes processes such as square resistance. The content of the process file (ITF) for parameter definition, Star-RCXT is calculated based on these process parameters.

MAPPING_FILE is a file with the extension of map, which is a mapping between the layer name in TCAD_GRD_FILE and the layer name defined in the LVS runset file. Different LVS runset files need to define different mapping files.

Star_cmd is a file that contains the commands to be executed. It is usually used in the command line mode. User can add and modify the commands in the file to achieve the desired application.

Star-RCXT has output formats such as SPF, SPEF, SBPF, etc.

There are several operating processes: Milkyway database flow, LEF/DEF database flow, Hercules database flow, Calibre connectivity interface (CCI) flow. The main difference between each process is that the database containing the layout information generated by each is different. In our research, we mainly employ the CCI flow. The flow of CCI-based parasitic extraction is shown in **Figure 2.28**.

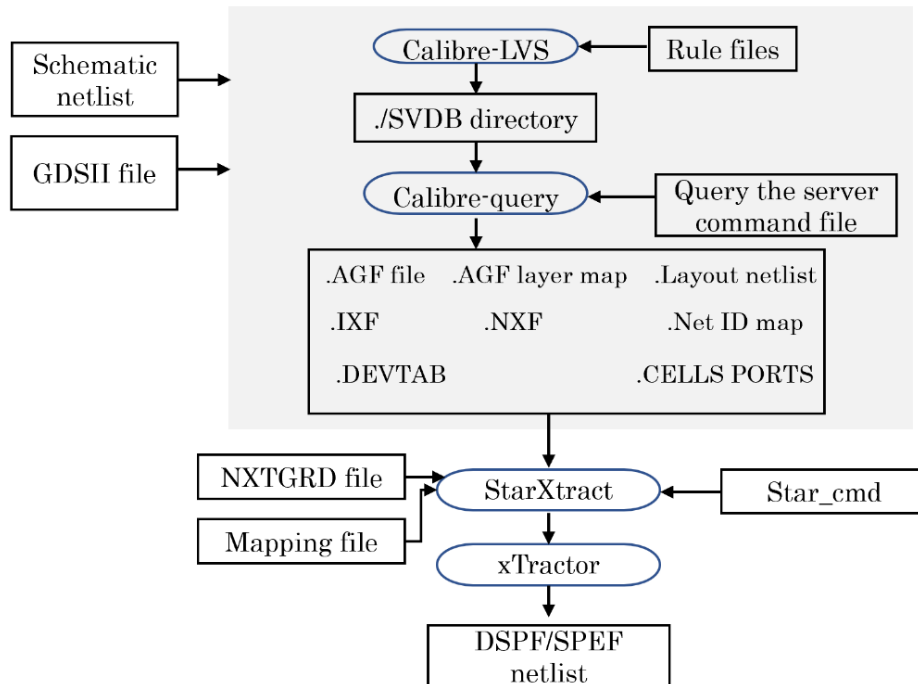


Figure 2.28: CCI-based parasitic extraction flow.

In Calibre, the flat method is used to perform LVS on the layout, and the SVDB directory is automatically generated after LVS is passed, setting the content for configuring LVS and saving it as a “lvs_set” file.

Then writing the query.cmd file, which specifies the location of the files needed for CCI-based Star-RCXT parasitic parameter extraction, such as the marked GDSII layout, layer

mapping information, top-level port information, ideal layout netlist, net-name information, interaction table and etc. After writing the query.cmd file, executing the command in the command line of the "calibredrv" tool: Calibre- query svdb < query.cmd.

The files needed for Star-RCXT parasitic parameter extraction will be generated, such as the file "xtalkdetailDRCLVS.agf" containing polygon and connectivity information, the component cross reference information file "xtalkdetailDRCLVS.ixf", the network cross reference information file "xtalkdetailDRCLVS.nxf", the ideal layout netlist file "xtalkdetailDRCLVS.nl", and the device table file "xtalkdetailDRCLVS.devtab" and etc.

In this flow, running the Star-RCXT command file (Star_cmd) to extract the parasitic parameters.

When writing the Star_cmd command file, input: module name "xtalkdetailDRCLVS", layer mapping file ".map", power/ground network name, VDD and GND, processing technology file ".nxtgrd", the output format of the extracted parameters, SPEF, the range of the extraction network, whether to convert the coupling capacitor into a capacitance grounding to earth, configuring the setting file "lvs_set" for LVS operation, and specifying the file location in "query.cmd" for extraction, based on CCI parasitic parameters extraction etc.

After the circuit netlist containing extracted parasitics is generated from Star-RCXT. We import the netlist into Virtuoso of Cadence and create a symbol corresponding to the netlist, and then replacing the circuit instance of the original testbench with a new symbol that represents the circuit with parasitics.

Thus, the process for preparing circuit simulation is complete, as shown in **Figure 2.29**. The remaining steps to do post-simulation are just as same as steps for pre-simulation.

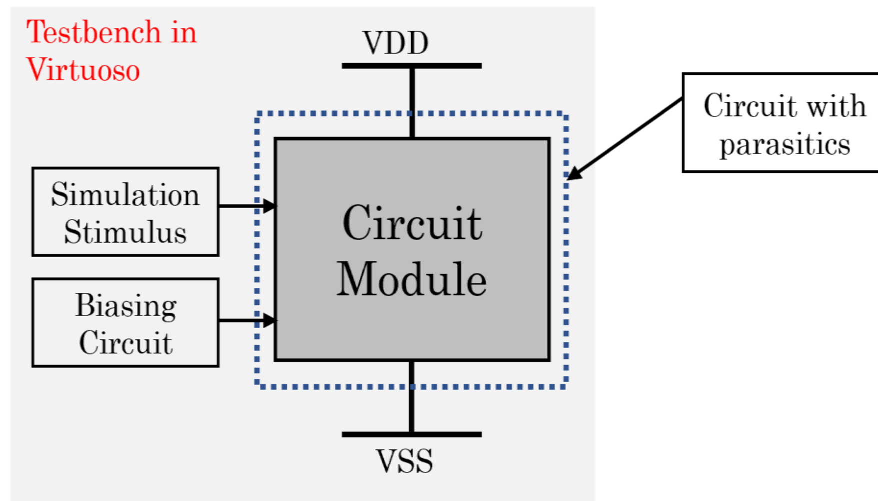


Figure 2.29: Post-layout simulation in Virtuoso for the circuit with parasitics.

CHAPTER 3

DENSITY-AWARE LAYOUT FORMAT

3.1 Density Checking

Layout pattern density in design rules is defined as a ratio of the sum of area for a layout layer divided by area of a pre-defined box, called checking window. As shown in **Figure 3.1**, a checking window moves along x-axis with a step denoted by x_{step}, when it reaches the right-side of the layout area, it returns to the initial position, and moves upward with the same step size (y_{step}), the window moving goes until sweeping over the whole layout. The density is calculated at each step of the checking window, and the checking at every step must be satisfied with the density constraint which is given as a pre-specified range of the density.

In general, the calculation of the density becomes more complicated when including the density effects from multiple layers. In the technology node 130nm and above, the density rule for layout design is simple and generally, for metal, density control is easily achieved. Advanced technology nodes are requiring even more complex density checking, a basic check for diffusion/poly/metal is mandatory. As the number and complexity of density checking increase, both the window size and the step size get smaller. In a 65nm CMOS process used in this work, density checking for contact is also mandatory. The width and the height of checking window are both 50 μm and denoted by w_{win} and h_{win}, respectively. The step sizes, x_{step} and y_{step}, are 1/2 of w_{win} and 1/2 of h_{win}, respectively.

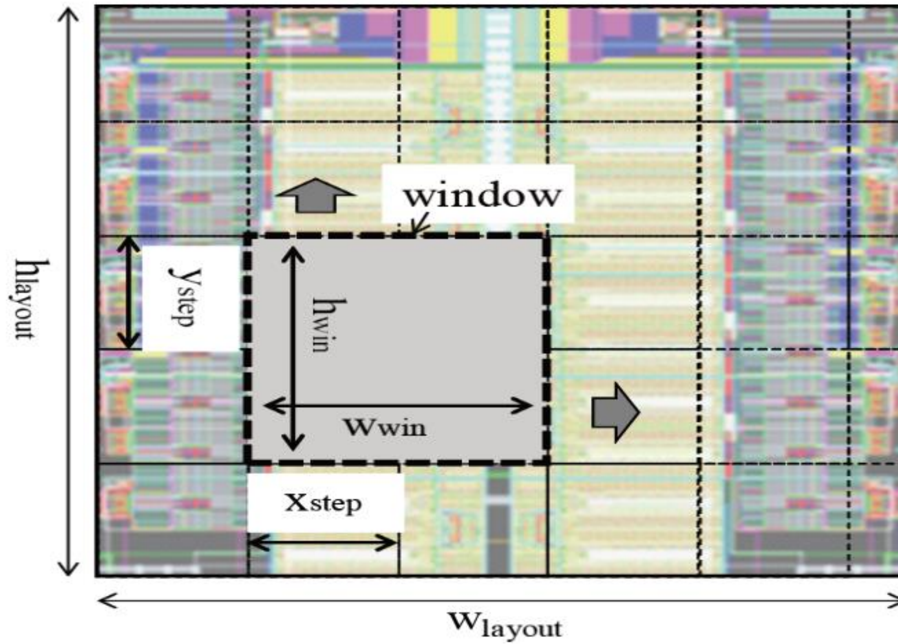


Figure 3.1: Checking window over the layout.

3.2 Key Idea

The density of a block might be changed depending on the position of the layout of the block. As shown in **Figure 3.2(a)**, the densities by the window in case-1, case-2 and case-3 might be different even if the window size is same. In fact, the checking result depends on the distribution of densities over the layout. In **Figure 3.2(b)**, the layout area is divided into eight square tiles, and the window covers 4 tiles and the width and height of the tile are the same as the step size. Plus, 'H' and 'L' represent high and low densities, respectively. Assume that 'H' and 'L' are 1.5 and 0.5, and the constraint is the density inside the window must be greater than 3.0 and less than 5.0. In the density distribution shown in the **Figure 3.2(b)**, all densities of the window are 4.0, and there is no error for checking. On the other hand, in the distribution shown in the **Figure 3.2(c)**, the density of the window at x1 is 2.0, resulting in a density error. Thus, nevertheless the sum of densities is same for both distributions, we have different checking results.

In other words, if the density over a block layout is even, the checking result of the block is independent of the window position. In this work, aggressively taking an advantage of TA-

style layout, we propose a layout generation such that all patterns of each layer are evenly distributed, limited to the array structure.

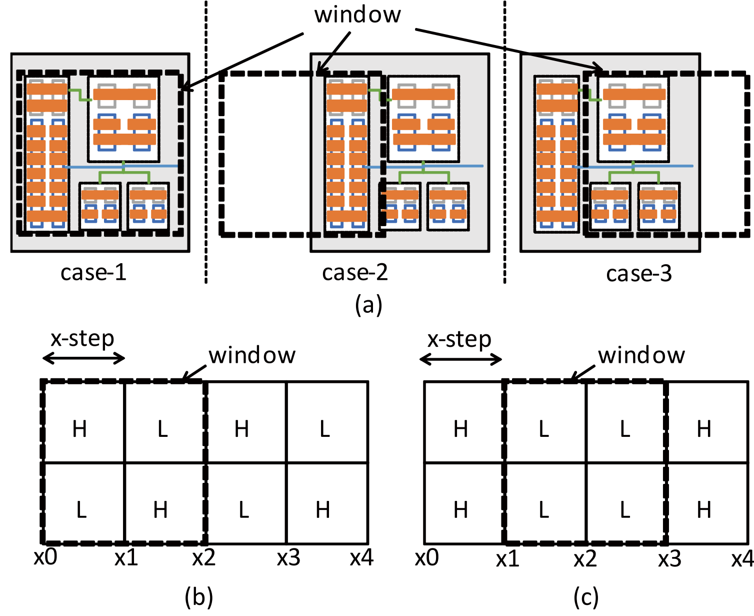


Figure 3.2: Density distribution and checking result.

3.3 Device/pattern parameters for TA-style Layout

In this work, we provide a layout format based on TA-style which is composed of parameters of density checking, transistor-array, and design rule. **Figure 3.3** illustrates a layout format example. As seen in the **Figure (a)**, a unit-transistor of TA-style has a unified channel length and width denoted by l_u and w_u, respectively, and they are user-defined values. h_{poly} and w_{diff} are height of poly gate and width of diffusion of a unit-transistor, respectively. These are tunable parameters to satisfy a given density constraint.

In this work, a checking window is divided into four square tiles according to its moving step, and the width and height of a tile are denoted by w_{tile} and h_{tile}, respectively (See the **Figure (b)**). Note that x_{step} and y_{step} maybe 1/3 or 1/4 of window size in other processes, therefore the window can be divided into 9 or 16 squares so as to apply to various processes. We define a format of TA-style layout corresponding to one tile as the one shown in the **Figure (c)**. Since we limit the layout structure is TA-style, we

can arrange tiles of the same structure to cover the given layout area ($w_{layout} \times h_{layout}$). The size of TA to a tile is $n_{row} \times n_{col}$ which is a parameter to pass the density checking while the size of window is prescribed by foundry. Note that unit-transistors unused in circuit are regarded as dummies, which has no impact on the functionality of the circuit.

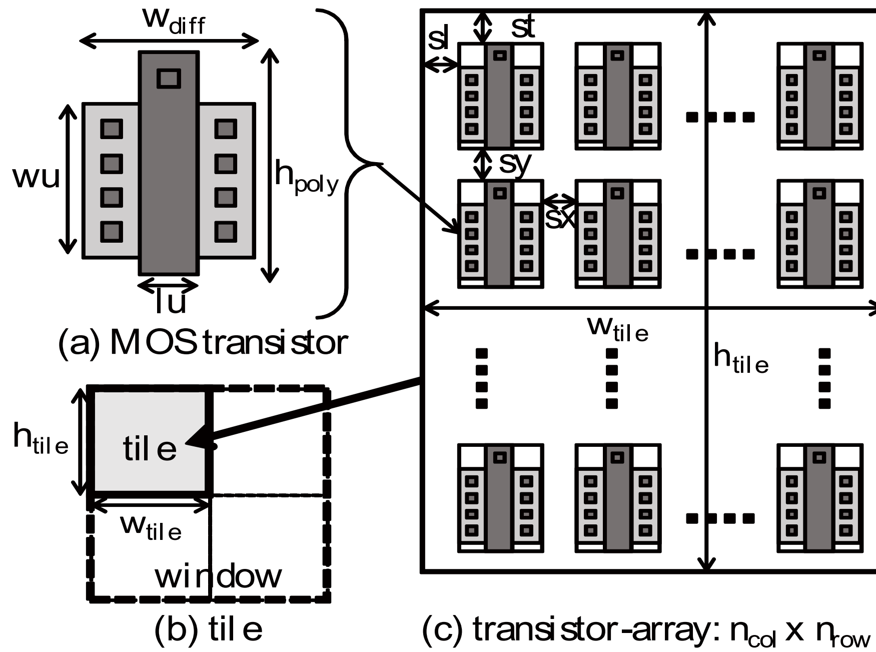


Figure 3.3: A density-aware layout format of TA-style: (a) Device parameters of a unit MOS transistor. (b) Checking window partition. (c) Pattern parameters of the transistor array.

Unit-transistors are placed by spaces s_x and s_y along x- and y-direction, respectively. The spaces are tunable parameters to satisfy a given density constraint. A tile has a boundary and a space from the left boundary to the left edge of the diffusion layer of the unit-transistor is s_l. Similarly, a space from the top boundary of a tile to poly layer is s_t.

Furthermore, a typical set of design rules prescribed by foundry must be considered. Such as the minimum values of channel length, channel width, diffusion, and poly gate of a transistor. Plus, the minimum area of diffusion and poly, the minimum spaces corresponding to s_x, s_y, s_t and s_l are also given. As for the density constraint, the minimum and maximum values for each layer are prescribed, respectively. $d_{min}(k)$ and $d_{max}(k)$ denote the minimum and maximum density of layer k.

3.4 Density and DRC Constraints

Given a tile of $n_{row} \times n_{col}$ TA, the density is calculated as a ratio of the total polygon area for a specified layer divided by the area of the tile. See **Figure 3.3**.

Density constraints for diffusion layer k are follows:

$$D_{min}^{k} \leq d_k \leq D_{max}^{k}, \tag{3.1}$$

Where

$$d_0 = \frac{w_{diff} \cdot w_u \cdot n_{row} \cdot n_{col}}{w_{tile} \cdot h_{tile}}, \tag{3.2}$$

$$d_1 = \frac{h_{poly} \cdot l_u \cdot n_{row} \cdot n_{col}}{w_{tile} \cdot h_{tile}}, \tag{3.3}$$

$$d_2 = \frac{a_{cont} \cdot n_{cont} \cdot n_{row} \cdot n_{col}}{w_{tile} \cdot h_{tile}}, \tag{3.4}$$

$$d_3 = \frac{a_{m1} \cdot 2 \cdot n_{row} \cdot n_{col}}{w_{tile} \cdot h_{tile}}, \tag{3.5}$$

$$d_4 = \frac{a_{m2} \cdot 2 \cdot n_{row} \cdot n_{col}}{w_{tile} \cdot h_{tile}}. \tag{3.6}$$

Their geometric relations in tile are as follows:

$$h_{tile} = 2 \cdot s_t + h_{poly} \cdot n_{row} + s_y \cdot (n_{row} - 1), \tag{3.7}$$

$$w_{tile} = 2 \cdot s_l + w_{diff} \cdot n_{col} + s_x \cdot (n_{col} - 1). \tag{3.8}$$

Foundries provide DRC rule for layout designers to create geometries that are manufacturable and provide density rule for each layer to refrain from post-CMP topography variation. These rules restrict the spacing and size of the geometric shapes and improve density uniformity among neighboring subregions. In our work, DRC and density constraints (the default unit is μm) in a 65 nm CMOS process are as follows,

$$10 \cdot e_{dp} \geq \frac{w_{diff} - l_u}{2} \geq 0.16, \tag{3.9}$$

$$10 \cdot e_{pd} \geq \frac{h_{poly} - w_u}{2} \geq 0.16, \tag{3.10}$$

$$s_x \geq 0.14, s_y \geq 0.14, \tag{3.11}$$

$$s_l \geq 0.14, s_t \geq 0.32, \tag{3.12}$$

$$a_{cont} = 0.0064, \tag{3.13}$$

$$a_{m1} \geq 0.03, a_{m2} \geq 0.03. \tag{3.14}$$

$$w_{win} = 50, h_{win} = 50, \tag{3.15}$$

$$D^k_{max} \geq d_k \geq D^k_{min}. \tag{3.16}$$

Here, the equalities and inequalities are translated from rule documentation provided by semiconductor manufacturer. Inequality (3.9) represents the allowable range of extension that diffusion exceeds poly, and (3.10) represents the allowable range of extension that poly exceeds diffusion. e_{dp} denotes the minimum extension from diffusion to poly in the horizontal direction, e_{pd} denotes the minimum extension from poly to diffusion in the vertical direction.

In our CMO process, e_{dp} and e_{pd} are both 0.16. Rule documentation just specifies lower bound for those two inequalities, upper bound is specified by us to avoid unreasonable long extension in TA-style layout. Inequalities (3.11) restrict the minimum spacing of diffusion shapes in x-axes and the minimum spacing of poly shapes in y-axes, respectively. Inequalities (3.12) restrict the minimum spacing between the transistor array and the boundary of tile.

Equality (3.13) specifies that the only allowed size of contact must be 0.0064 μm^2, inequalities (3.14) restrict the minimum area of metal 1 and that of metal 2, respectively. Equalities (3.15) specify the size of the checking window, for which the width is 50 and the height is also 50.

Inequality (3.16) represents the density bounds for the k-th layer following the sequence of diffusion, poly, contact, metal 1, metal 2.

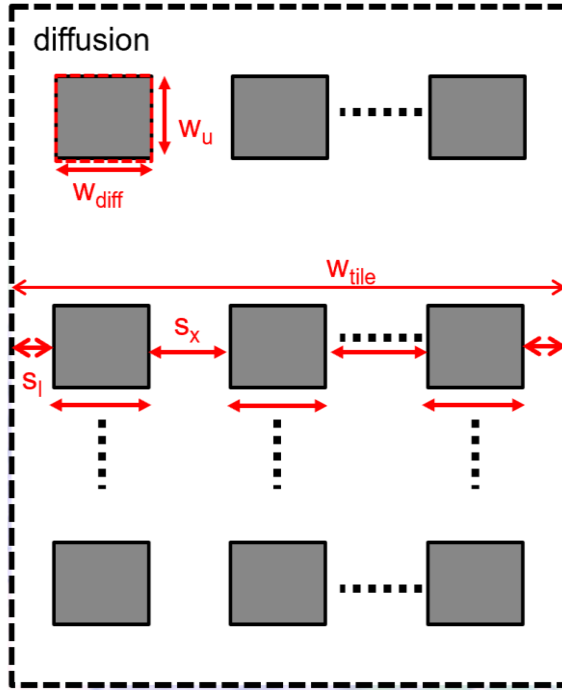
The design rules above are most basic rules for various manufacturing processes, the value of each parameter depends on the process and varies with the technique applied by the foundry.

Note that D^k_{max} and D^k_{min} are density bounds specified in the rule file for the manufacturing process, and are regarded as known constants for the input in TA-based approach. Due to the non-disclosure agreement with the foundry, we cannot describe real values related to the density bounds.

The digital circuit layout is synthesized from the digital library which is constituted by gate-level layouts. All the gate-level layouts are verified to be correct for DRC-clean, DRC errors usually occur on routing nets during assembling a single chip.

Dealing with design rules such as, metal width, enclosure and spacing are most common for digital circuit layouts. Additionally, DRC and density errors are automatically fixed by EDA software. As for analog layout, however, it not only needs to handle more rules such as diffusion extension, poly extension, contact enclosure, but also spends more time on layout completion due to the manual correction.

DRC and density constraints in the TA-style analog layout of this section is elaborated in **Figure 3.4**. We also give some demos in **Figure 3.5** to show how we control the layout pattern density. By changing of device/pattern parameters of a transistor array, the geometric shape and the area of a layer can be changed accordingly, thus the layout pattern density can be well controlled, such as stretching or shortening device parameters (for poly and diffusion) of a transistor array, and increasing or decreasing pattern parameters (for all layers) of a transistor array, are very convenient ways to achieve the density level desired. As for metal and contact, stretching or shortening the width of the metal layer and increasing or decreasing the number of the contact layer can explicitly control their density, respectively. The details of the two figures are shown in the following.



DRC constraints (unit is μm).

$$h_{tile} = 2 \bullet s_t + h_{poly} \bullet n_{row} + s_y \bullet (n_{row} - 1)$$

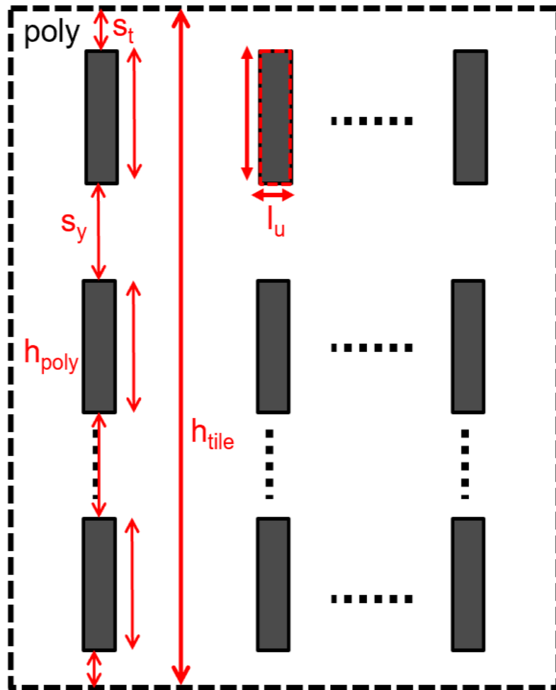
$$w_{tile} = 2 \bullet s_l + w_{diff} \bullet n_{col} + s_x \bullet (n_{col} - 1)$$

Density constraints (diffusion).

$$d\,(diff) = w_{diff} \bullet w_u \bullet n_{row} \bullet n_{col} / (w_{htile} \bullet h_{tile})$$

$$D^0_{min} \leq d(0) \leq D^0_{max}$$

(a)

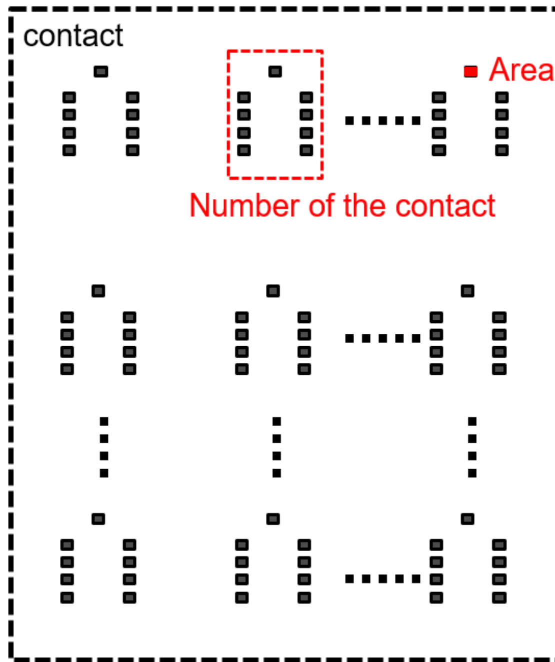


Density constraints (poly).

$$d\,(poly) = l_u \bullet h_{poly} \bullet n_{row} \bullet n_{col} / (w_{htile} \bullet h_{tile})$$

$$D^1_{min} \leq d(1) \leq D^1_{max}$$

(b)

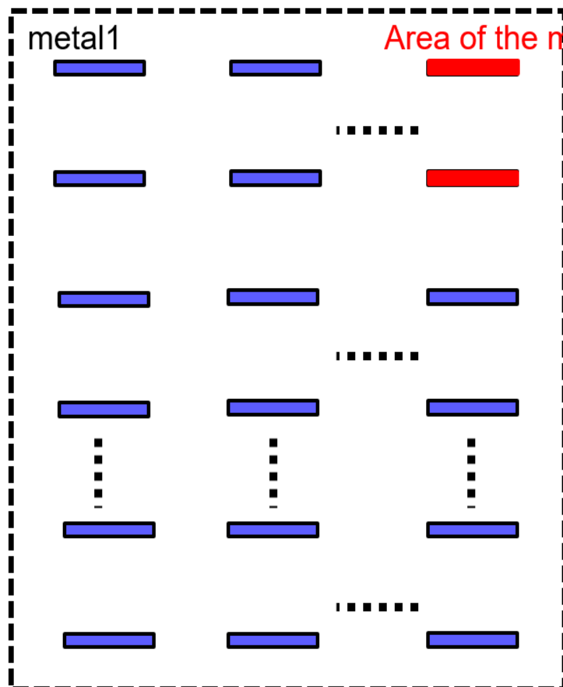


Density constraints (contact).

$$d\,(cont) = a_{cont} \cdot n_{cont} \cdot n_{row} \cdot n_{col} / (w_{htile} \cdot h_{tile})$$

$$D^{2}_{min} \leq d(2) \leq D^{2}_{max}$$

(c)

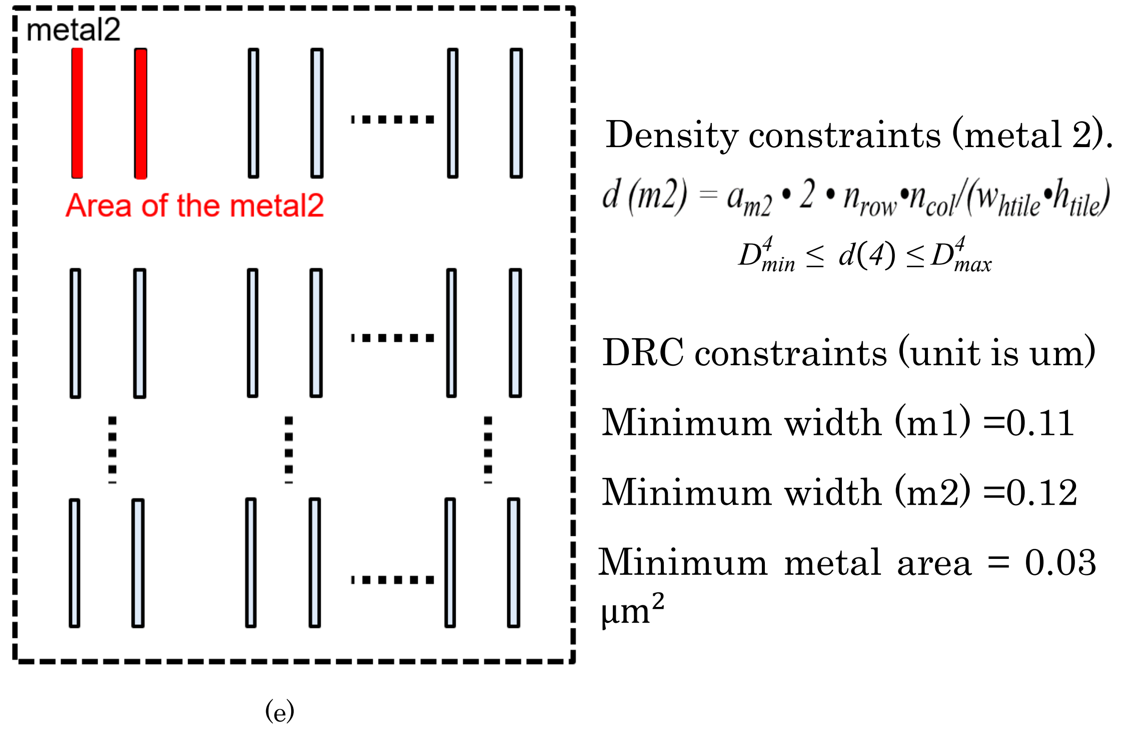


Density constraints (metal 1).

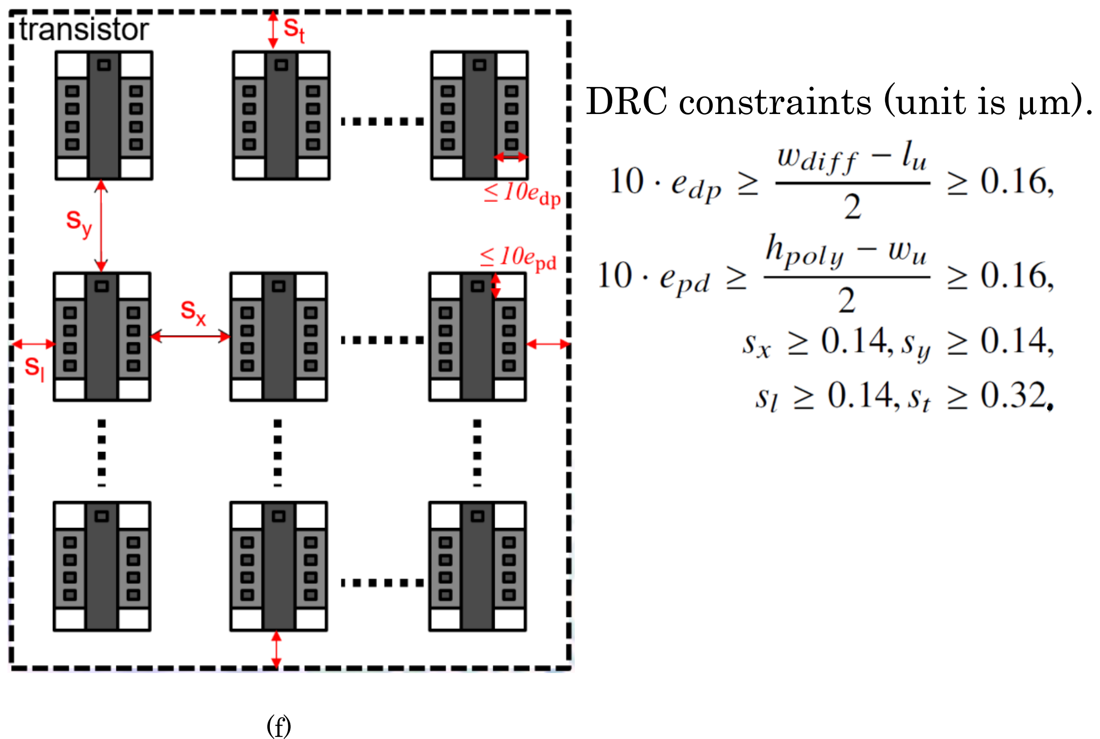
$$d\,(m1) = a_{m1} \cdot 2 \cdot n_{row} \cdot n_{col} / (w_{htile} \cdot h_{tile})$$

$$D^{3}_{min} \leq d(3) \leq D^{3}_{max}$$

(d)



(e)



(f)

Figure 3.4: DRC and density constraints in a 65nm CMOS process (Renesas technology). (a) Diffusion layer. (b) Poly layer. (c) Contact layer. (d) Metal 1 layer. (e) Metal 2 layer. (f) Layout pattern of a transistor array.

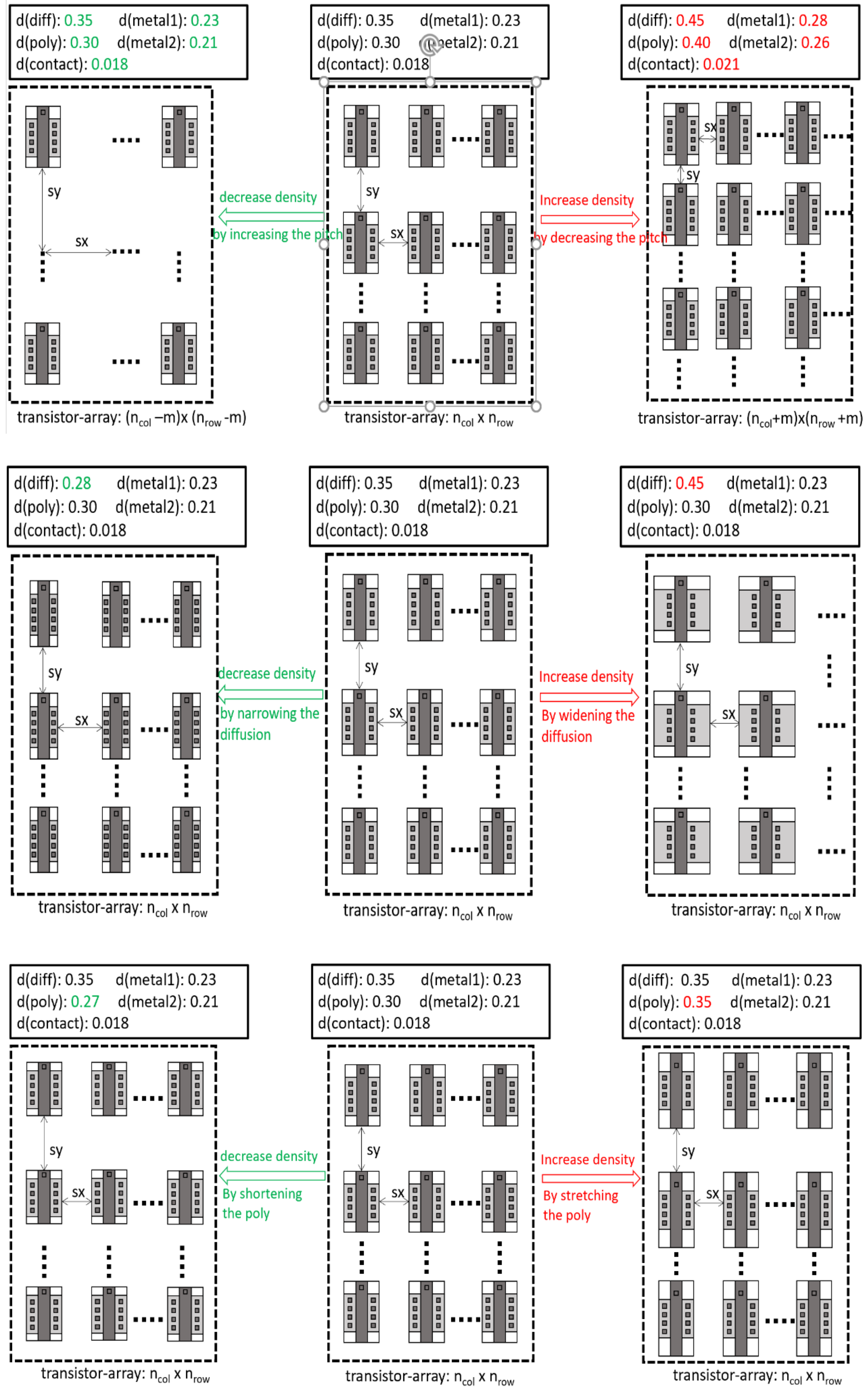


Figure 3.5: Demos to show the layout pattern density controlling.

For the readability, parameters and annotations used in this work are listed in **Table 3.1**.

Table 3.1: Parameters and annotations used in this work.

density checking	
w_{layout}:	the width of layout area
h_{layout}:	the height of layout area
w_{win}:	the width of window
h_{win}:	the height of window
x_{step}:	the distance of step moving window to x-direction
y_{spet}:	the distance of step moving window to y-direction
Device/Pattern parameters for TA-style layout	
l_u:	channel length of unit transistor
w_u:	channel width of unit transistor
h_{poly}:	the height of poly
w_{diff}:	the width of diffusion
w_{tile}:	the width of tile
h_{tile}:	the height of tile
a_{m1}:	the area of the metal1 in unit transistor
a_{m2}:	the area of the metal2 in unit transistor
a_{cont}:	the area of the contact in unit transistor
n_{cont}:	the number of the contact in unit transistor
n_{row}:	row number of array in tile
n_{col}:	column number of array in tile
s_x:	space among transistors along x-direction
s_y:	space among transistors along y-direction
s_l:	space from left boundary of tile to array region
s_t:	space from top boundary of tile to array region
DRC and density constraints	
l_{min}:	minimum of channel length of a transistor
w_{min}:	minimum of channel width of a transistor
$w_{diff,min}$:	minimum of w_{diff}
$h_{poly,min}$:	minimum of h_{poly}
$a_{diff,min}$:	minimum area of $w_{diff} \cdot w_u$:
$s_{x,min}$:	minimum space of s_x
$s_{y,min}$:	minimum space of s_y
$s_{t,min}$:	minimum space of s_t
$s_{l,min}$:	minimum space of s_l
D^k_{min}:	minimum density of layer k
D^k_{max}:	maximum density of layer k

CHAPTER 4

DENSITY OPTIMIZATION STRATEGY

We provide a mathematical approach for optimizing the density of the tile consistent with the design rule.

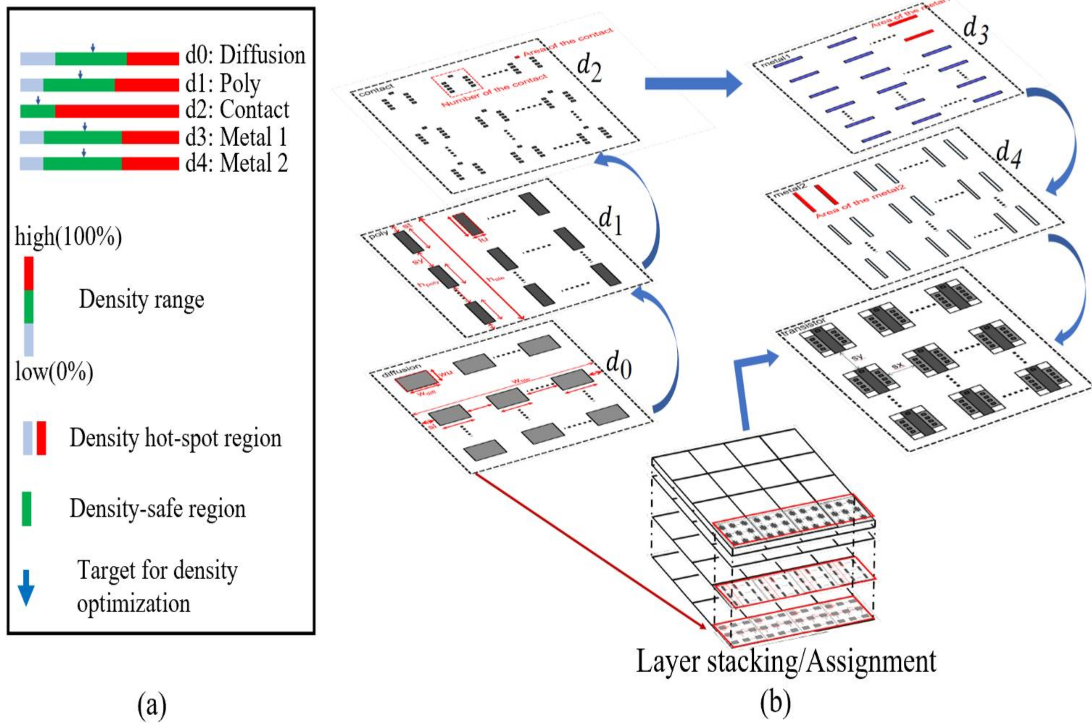


Figure 4.1: Density optimization strategy. (a) Density optimization aiming at the centering value of the density constraint. (b) Density controlling for the layout pattern.

Our objective can be illustrated by **Figure 4.1(a)** that is to make each layer required for density check to meet density bounds. Meanwhile, we expect the density of the layer to be in the middle of bounds. In general, layout can pass density check if density levels of layers are within specified ranges (density-safe region), otherwise (in the density hot-spot region) it reports errors in the verification step. Since too high/low feature density on each layer causes CMP to under/over polishing, thus we target at centering value of density-safe range for further density optimization. Feature density of layer is controlled by density parameters ($d_k, k \in 0,1,2,3,4$), and layer's corresponding geometries. Layers stack/assign in the order (see **Figure 4.1(b)**) to form a layout pattern of a transistor array.

4.1 Density Optimization Problem

Our objective is to make each layer required for density check meet density bounds. Meanwhile, we expect the density of the layer to be in the middle of bounds. As such, the generated layout has much density margin so as not to suffer from over/under polishing. In this work, without loss of generality, we regard a device/pattern parameter set X as a solution of density optimization problem of TA-style, where $X = (n_{row}, n_{col}, w_{diff}, h_{poly}, s_x, s_y, s_l, s_t, n_{cont}, a_{m1}, a_{m2})$. For a device/parameter set X, we introduce the following objective as aiming to place the density at the center of the density constraint. This work provides a TA-style layout generation according to the density-aware layout format. The procedures of our approach are organized as follows.

$$\delta(X) = \sqrt{\frac{1}{N} \cdot \sum_{k=0}^{N} w_k \cdot \left(\frac{d_k - \frac{D_{min}^k + D_{max}^k}{2}}{\frac{D_{min}^k + D_{max}^k}{2}} \right)^2}, \tag{4.1}$$

where $\delta(X)$ is an RMSE (Root Mean Square Error)-like function for density variation to evaluate the distance between the practical value and the central value, the only difference is that we introduce the weight into the function.

The definition of $\delta(X)$ combines variations arised from by different layers. It deems that if we only consider metal layers, the density optimization for the yield improvement will become meaningless as poly, diffusion, and contact can also affect manufacturability. Besides, we introduce a weight parameter into the objective function, in order to distinguish the priority and importance of each layer.

Here, w_k is the weight of d_k which is proportional to layer's contribution to layout pattern, $w_k \in (0, 1)$. N is the number of layers, d_k is the density of the k-th layer, D_{min}^k and D_{max}^k are density bounds of the k-th layer. Note that d_k is calculated depending on X (See **Eqs. (3.2)** to **(3.8)**).

We define the objective function as the Min-Var model, the solution to the optimal layout pattern is the one with minimal density variation. The weight of a specific layer depends on its occupied area in the whole pattern and the impact on CMP processing, the sum of all weights is 1.

For instance, in our design case, diffusion and poly are essential to constitute a transistor on which other layers are processed by CMP, we set both weights as 0.25. Metal 1 and metal 2 are critical for interconnects and CMP quality, we set both weights as 0.24. However, contact only occupies a small fraction of layout pattern as the rule file specifies the density of less than 10%, we set it as 0.02.

Note that the above weights are final results, our program iteratively tunes the weights from initial values given by our experience. We calculate the density variations of numerous TA-style layout patterns, compare their density uniformity under different weights, and find a set of weights that best fit this relationship. Thus, for all feasible sets of X of a tile, a mathematical optimization is formulated as:

$$\begin{aligned} \min_{\forall X} \quad & \delta(X) \\ \text{s.t.} \quad & \text{Eq. from (3.1) to (3.16)} \\ & n_{row} \in N^{+}, n_{col} \in N^{+}. \end{aligned} \tag{4.2}$$

In our optimization formulation, w_u, l_u, w_{tile}, h_{tile}, a_{cont}, D_{max}^{k}, D_{min}^{k}, can be regarded as constants, but the problem is still strongly non-convex with discrete constraints. Thus, it is hard to be solved directly.

4.2 Min-Dum Scheme

Since objective function is constituted by 11 variables, and there are nonlinear constraints in density equations (see **Eqs. from (3.2) to (3.6)**). It is nearly impossible to solve it directly because computation overhead is vastly expensive. Through effective Min-Dum scheme, however, we can prune some inferior solutions at the early stage and avoid exhaustive density

optimization on all feasible solutions. The overall efficiency can be significantly improved because of the shrinking of the solution space. In this section, we propose a flow based on Min-Dum scheme to simply the original density optimization problem described in **Eq. (4.2)**. The Min-Dum scheme is defined as follows.

Definition 1 (Min-Dum). *Min-Dum is defined as the minimum number of dummy transistors in a tile. Necessary amount for used unit transistors is equal to that total number of unit transistor in TA-style layout divided by the number of partitioned tiles, redundant transistors are dummies.*

Taking inequalities **(3.9)** and **(3.10)** into account, h_{tile} limits n_{row} and w_{tile} limits n_{col}. The upper bounds for row number and column number are given as follows.

$$n_{row} \in (0, \frac{h_{tile}}{w_u + 0.32}), n_{row} \in N^+, \tag{4.3}$$

$$n_{col} \in (0, \frac{w_{tile}}{l_u + 0.32}), n_{col} \in N^+. \tag{4.4}$$

In our work, the bound for row number is $n_{row} \in (0, 18.9)$, for column is $n_{col} \in (0, 10.7)$. both n_{row} and n_{col} are positive integer. Because it is extremely difficult to directly solve the original density optimization problem described in **Eq. (4.2)**, we first obtain feasible solutions from design constraints by enumeration method. The resulting feasible solutions by program is shown in **Table 4.1**.

Here, feasible set denoted by F is sets of values that satisfy all the constraints, non-feasible set denoted by NF indicates that there does not exist a solution for the problem. In the feasible region, density values with respect to device/pattern parameters are computed by a program.

In our implementation of an OPAMP circuit, the domain of optimization is a subset of the feasible region. In TA-style layout, less dummy is more desirable because redundant transistors affect parasitics and cause congestion in channel-based routing. In our implementation, the necessary unit transistors are 350 according to device decomposition. The necessary amount in a tile is 350 divided by 4, i.e., 87.5. Since the number of the

transistor must be an integer, the necessary amount in a tile actually is 88, the dummies for the whole layout are 2. Min-Dum is a scheme to constraint the value $n_{row} \times n_{col}$.

Table 4.1: Feasible solution set and the corresponding feature density.

Domain		device/pattern parameters	Feature density
	(n_{row}, n_{col})	$(w_{diff}, h_{poly}, s_x, s_y, s_l, s_t, n_{cont}, a_{m1}, a_{m2})$	$(d_0, d_1, d_2, d_3, d_4)$
NF	(1, 1)	n/a	n/a
	…	…	…
F	(4, 4)	(5.914, 2.204, 0.251, 3.484, 0.294, 2.866, 28, 0.841, 0.441)	(0.151, 0.113, 0.005, 0.322, 0.271)
	…	…	…
	(11, 8)	(2.488, 1.402, 0.533, 0.471, 0.68, 2.437, 30, 0.68, 0.28)	(0.35, 0.395, 0.027, 0.396 , 0.299)
	(11, 8)	(2.824, 1.472, 0.249, 0.581, 0.333, 1.499, 21, 0.694, 0.294)	(0.398, 0.414, 0.019, 0.398, 0.301)
	…	…	…
	(16, 10)	(2.328, 1.323, 0.151, 0.184, 0.179, 0.534, 22, 0.665, 0.265)	(0.596, 0.678, 0.036, 0.47, 0.328)
NF	…	…	…
	(18, 10)	n/a	n/a

4.3 Nonlinear Programming

Once we determine n_{row} and n_{col} , density equations become linear with only one variable in each equation. The square of objective function $\delta(X)$ becomes a quadratic function with respect to device/pattern parameters, parameters other than device/pattern parameters are constants in the afore-mentioned formulation. The quadratic programming problem with 9 variables, density inequalities, geometric equalities, can be formulated as follows.

$$\min_{x} \quad \frac{1}{2}x^T H x + f^T x$$

$$s.t. \quad A \cdot x \leq b, \tag{4.5}$$

$$Aeq \cdot x = beq, \tag{4.6}$$

$$lb \leq x \leq ub. \tag{4.7}$$

Here, H, A and Aeq are matrices, and f, b, beq, lb, ub, and x are vectors. x^T denotes the vector transpose of x. whole

formulation is a standard form of quadratic problem. In our case, substituting density equations (**Eqs.** from **(3.2)** to **(3.6)**) into objective function **(4.1)**, then the square of objective function can be transformed to the form above.

Matrice A and vector b can be obtained from the transformed objective function. Substituting each density equation into inequality **(3.1)** respectively, thus a linear constraint on each variable can be transformed to the form of **(4.7)**. Matrice Aeq and vector beq can be obtained from equalities of geometric constraints **(3.7)** and **(3.8)**.

Hence, quadratic programming (QP) techniques are used to solve our problem. As a special case of non-linear programming problem (NLPP), algorithms such as interior-point-convex, trust-region-reflective and active-set are commonly used. Besides, a variety of optimization solver with high performance and friendly interface are available. Such as CPLEX (IBM), ALGLIB (across-platform open source library) and MATLAB toolbox. Bound, equality and inequality constraints exist in our formulation, therefore interior-point-convex algorithm is applied to our work, because it is effective in solving convex problems with any combination of constraints. Algorithmic details for layout generation are shown in the **Algorithm 1**.

Algorithm 1 $O(n^2)$ Layout generation.

Input: w_{layout}, h_{layout}, l_u, w_u, w_{tile}, h_{tile}, DRC constraints, D^k_{min} and D^k_{max} for each layer k;

Output: (n_{row}, n_{col}), h_{poly}, w_{diff}, s_x, s_y, s_l, s_t, n_{cont}, a_{m1}, a_{m2}, density of each layer k;

1: Set user-defined parameters such as l_u, w_u, w_{tile}, and h_{tile};
2: Let feasible region of vector set X be $X^*= \{\{N_{row}, N_{col}\}^*, H^*_{poly}, W^*_{diff}, S^*_x, S^*_y, S^*_l$, and $S^*_t, N^*_{cont}, A^*_{m1}, A^*_{m2}\}$ respectively;
3: **for** i $\leftarrow 1, \cdots, R_{max}$ (upper bound for rows)do
4:     **for** j $\leftarrow 1, \cdots, C_{max}$(upper bound for columns) do
5:         Select a initial value randomly for W_{diff} within its feasible
6:         range, compute each parameter of vector X recursively
7:         according to formulation;
8:         **If** parameters all meet constraints **then**
9:             Update X^*, compute density d_k;
10:         **else** repeat step 5 to 10, **until** termination threshold reaches ξ;
11:     **end for**
12: **end for**
13: Perform *Min-Dum* pruning, X^* changes to $X^*= \{\{N_{row}, N_{col}\}, H^*_{poly}, W^*_{diff}, S^*_x, S^*_y, S^*_l$, and $S^*_t, N^*_{cont}, A^*_{m1}, A^*_{m2}\}$;
14: Perform QP algorithm to search the optimum solution, compute δ and d_k;
15: Place a tile at $(0,0)$. Let the upper-right corner of the tile be (x, y), that is, $(x, y) \leftarrow (w_{tile}, h_{tile})$;
16: **While**$(y \leq h_{layout})$ **do**
17:     **While**$(x \leq w_{layout})$ **do**
18:         Perform layout generation layer by layer according to the optimal pattern parameters; x += x_{step};
19:     **end while**
20:     x = w_{tile}; y += y_{step};
21: **end while**
22: **return** X^* and d_k

CHAPTER 5

DENSITY-AWARE TA-STYLE ANALOG LAYO-UT

We implement our density optimization approach based on the density-aware layout format and incorporate it into TA style design flow. The TA-style analog layout design flow is shown in **Figure 5.1**. The procedures of our approach are organized as follows.

1. The original circuit is decomposed into sets of unit transistors with unified channel length l_u and unified channel width w_u, called TA-style circuit. We decompose an analog circuit into a TA-style circuit, from which the dimension and the number of unit transistors can be obtained, as shown in the "partition for circuit elements" part.

2. We configure a solution space of feasible device/pattern parameters satisfying design rules and density rules. A feasible region is derived from design constraints, and also searching space for the optimum is decreased by Min-Dum scheme. Meanwhile, the feature density of each layer is computed in the density aware format.

3. The formulation is simplified to a quadratic programming problem according to explicitly known conditions. The program yields the best parameters for layout generation aiming at optimization objective. Note that, until this step, the program just obtains a set of parameters that characterize a TA-layout pattern with the least density variation, but without considering placement and routing.

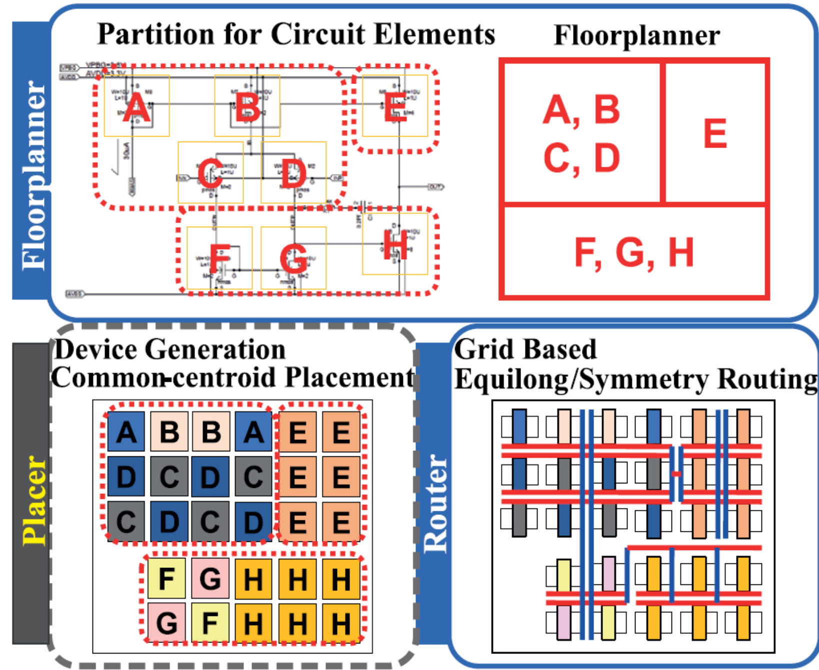


Figure 5. 1: Design flow based on the TA layout synthesis.

4. Once we obtain the device/pattern parameters and a transistor-array configuration, we can assign the unit transistors within a tile in TA manner, duplicate the tile filled with transistor-array and place tiles to cover a given layout area. Note that the site of each original transistor in a TA-style layout is unknown yet, based on a floorplanner, the unit-transistors are grouped into rows and the rows' locations are determined as well.

Each group of unit-transistors corresponds to an original transistor in the circuit, thus all the transistors in the original circuit are mapped into ones in TA-style layout constituted by unit-transistors. The floorplanning complies with the following rules: The unit-transistors belonging to the fundamental circuits such as differential pair, current mirror, will be arranged into same rows and assigned in close proximity to meet matching constraints. Besides, the floorplanner provides the feature for automatically recognizing the fundamental circuits from netlist. The procedure is shown in the "floorplanner" part.

5. With such a row-based structure, importing the layout constraints such as symmetrical constraints or common-centroid constraints, and employing the placement algorithms proposed

in [48] to generate a symmetric/common-centroid placement row by row, which can solve symmetrical problem effectively. As shown in the "placer" part.

6. We standardize the elementary components in the analog circuits as we are doing in digital circuit design. With such a standard-cells- or gate-array-like placement, it is possible to apply the channel routing algorithm [49] of digital design to analog design. Such as using the horizontal channels to connect the unit transistors in each row and using the vertical channels to connect the transistors in different rows. Dealing with the routing constraints is shown in the "router" part.

7. A TA-style layout is generated from a given circuit. Note that all the procedures are automatically implemented except for the input constraints set by hand. In addition, as done in the layout synthesis of digital design, it is necessary to fix minor DRC errors if non-convergence occurs in the routing phase.

5.1 Feasible Device/Pattern Parameters

In the following, we describe the procedures to extract feasible device/pattern parameters based on the density-aware layout format. In the **Algorithm 1**, the procedures from **step (2)** to **step (12)** configure feasible range of device/pattern parameters satisfying design rules and density constraints for specified layers. At **step (14)**, based on quadratic programming algorithm, we obtain the minimum density variation on vector set X and the corresponding device/pattern parameters. Once a TA-style layout for a tile is obtained by our generation algorithm, we copy tile and place the copied tile to cover the given layout area. Since TA in tile can meet the DRC and density constraints, any region of layout pattern surrounded by checking window can meet them as well.

Then, the given circuit, which is decomposed into a set of unit-transistors in parallel or serial connection, is assigned to the tiles. Note that if the number of rows and columns is insufficient, the program will automatically select other feasible

device/pattern parameters. In addition, if we have many redundant unit-transistors after the assignment, they are regarded as dummies.

5.2 Optimum Device/Pattern Parameters

In this work, targeting at a 65 nm CMOS process, we demonstrate an example of how to extract ranges of feasible device/pattern parameters. It is implemented under a common design flow applying to various processes. Our approach considers the most basic design rules for the processes, such as spacing, extension, and the area between geometries, as well as considering the density constraints of specific layers for the advanced processes.

It is necessary for program designers to translate human-readable rule documentation into machine-readable inequalities. Some inequalities for design rules also need to be merged into one as they have overlap. It is complicated to produce a set of inequalities covering all cases because the design rules vary from the process.

Nevertheless, our approach can deal with general rules, analog layout designer only needs to focus on layout generation without considering translation. With respect to the special cases, additional rules are needed to be carefully incorporated to design constraints. Based on our experience to realize an analog macro-cell layout by TA-style, we set a unit-transistor dimension as, l_u = 2 µm, w_u = 1 µm. The size of a tile is specified as, w_{tile} = 25 µm and h_{tile} = 25 µm. Hereinafter, we omit the unit µm from the numerical values.

The given design rules are translated as the following constraints: $20 \cdot 0.16 \geq w_{diff} - l_u \geq 0.32$, $20 \cdot 0.16 \geq h_{poly} - w_u \geq 0.32$, $s_x \geq 0.14$, $s_y \geq 0.14$, $s_l \geq 0.14$, $s_t \geq 0.32$, $a_{cont} = 0.0064$, $a_{m1} \geq 0.03$, $a_{m2} \geq 0.03$. In addition, taking density constraints into account, we calculate the range of feasible device/pattern parameters by program.

The table 4.1 for feasible solution set shows an example of the calculation of the range. We cannot describe the real parameters related to density constraints due to the non-disclosure agreement with the foundry. In this example, we make the table by extracting the parameter ranges for diffusion, poly, metal and contact layers. Their density values are automatically computed and displayed, so as to guide designer to control layout pattern density. An effective algorithm serves the efficient way to search the optimum solution.

From the feasible device/pattern parameters generated from the program, we obtain an optimum solution of (n_{row}, n_{col}) = (11, 8), w_{diff} = 2.93, h_{poly} = 1.40, s_x = 0.16, s_y = 0.55, s_l = 0.26, s_t = 1.29, n_{cont} =30, a_{m1}=1.45, a_{m1} = 1.45. A layout of the unit transistor used in the example has the minimum of w_{diff}. When l_u = 2.0, the minimum of w_{diff} is 2.32, but the parameter tuned is 2.93. Analogously, the minimum of h_{poly} is 1.32, but it is set to 1.40 automatically. This is because the parameters have been automatically tuned to satisfy the density constraints by the algorithm.

The procedures to extract feasible device/pattern parameters are illustrated in **Figure 5. 2**.

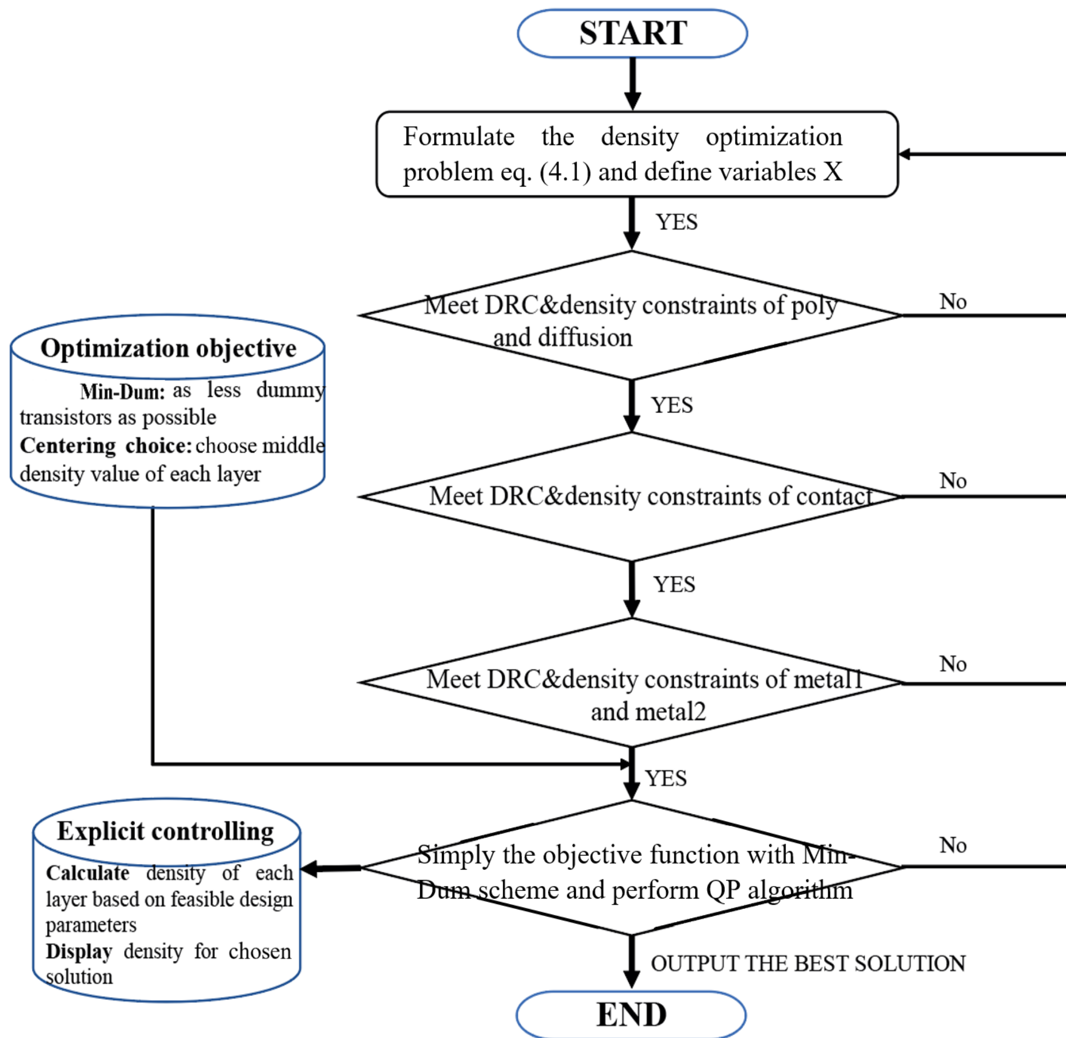


Figure 5. 2: A flowchart for device/pattern parameters extraction.

5.3 Design Example

We employ a typical OPAMP circuit as a motif to design an analog circuit according to this format. The original circuit of OPAMP is shown in **Figure 5.3**. For the comparison, a manual layout of OPAMP with dummy fills designed by an expert is shown in **Figure 5.4(a)**. On the other hand, our TA-style layout of OPAMP is shown in **Figure 5.4(b)** and its floorplan is shown in **Figure 5.4(c)**. We can observe that layout (b) consists of four tiles in which the transistor-array is generated.

Table 5.1 shows the summary of OPAMP designs by hand and by our TA-style generation, both layouts can pass physical verification. Here, variation by Min-Var model is a percentage of

the result of the objective function, which is used to evaluate the density variation of a layout.

Table 5.1: Summary of design.

layer	layout (a) density	layout (b) density	Min-Var target density
metal1	0.175	0.414	0.425
metal2	0.168	0.419	0.425
contact	0.023	0.027	0.075
poly	0.546	0.394	0.4
diffusion	0.426	0.412	0.475
area	2,125 μm^2	2,500 μm^2	-
sign-off iteration	≥ 60	≤ 5	-
overall design time	≥ 24 h	≤ 0.5 h	-
variation by Min-Var model(%)	20.85	5.07	0

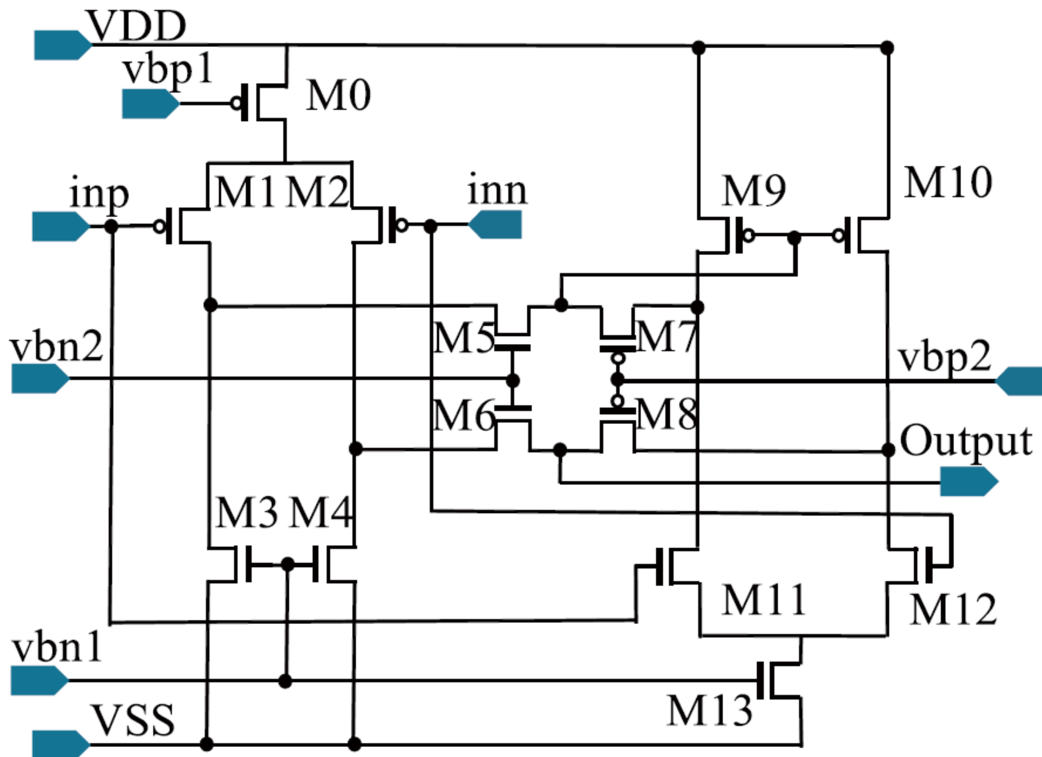


Figure 5.3: OPAMP schematic for a design example.

The measured density of TA-layout is nearly consistent with the predicted density. However, density for manual design is unknown before verification. Compared with the layout (a)

with inserting dummy fills in empty spaces, the density of each layer of the layout (b) is better because density variation is significantly reduced. Besides, layout(b) can achieve better CMP quality due to the better density uniformity and less density gradient by such mask pattern.

Obviously, the number of iterations of verification for TA layout is far smaller than that of the layout (a), our design time is reduced, and overall efficiency is improved considerably. Note that most of the design time for IC is spent on verification and incremental fixes, therefore fewer iterations accelerate the convergence of designs.

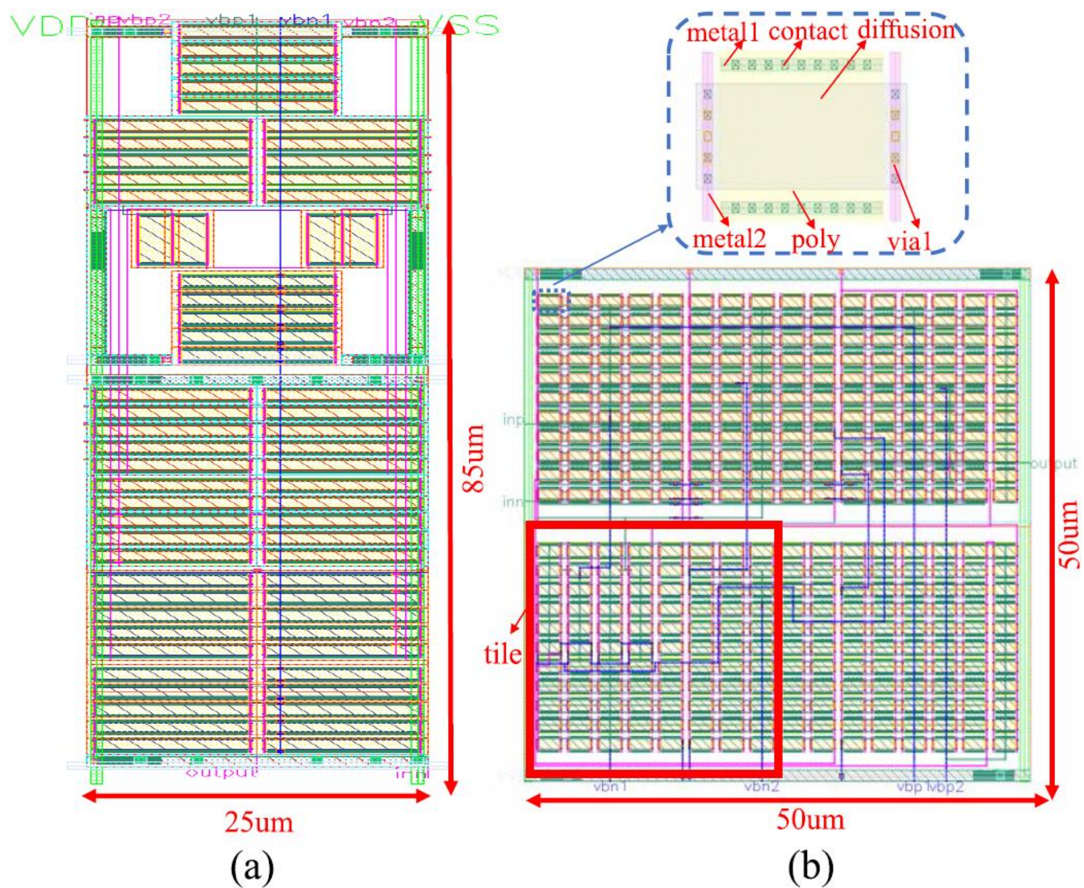


Figure 5.4: Cont.

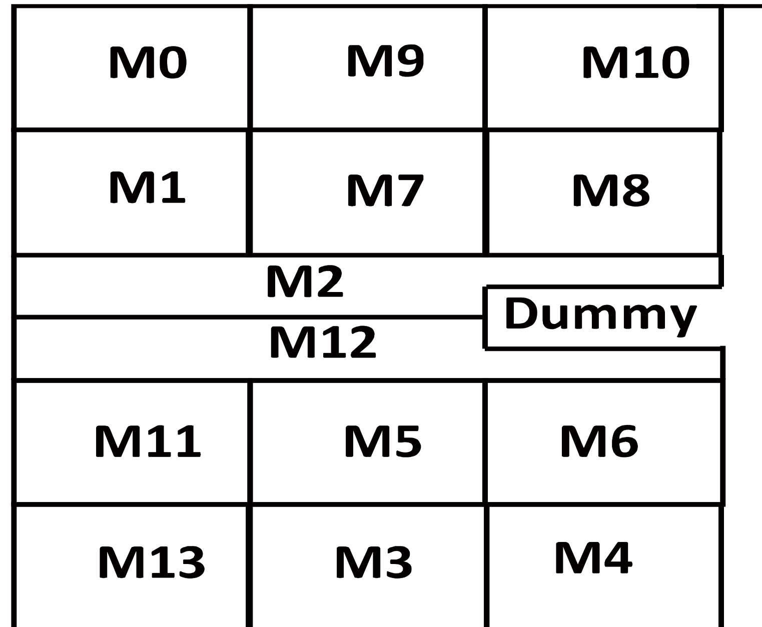
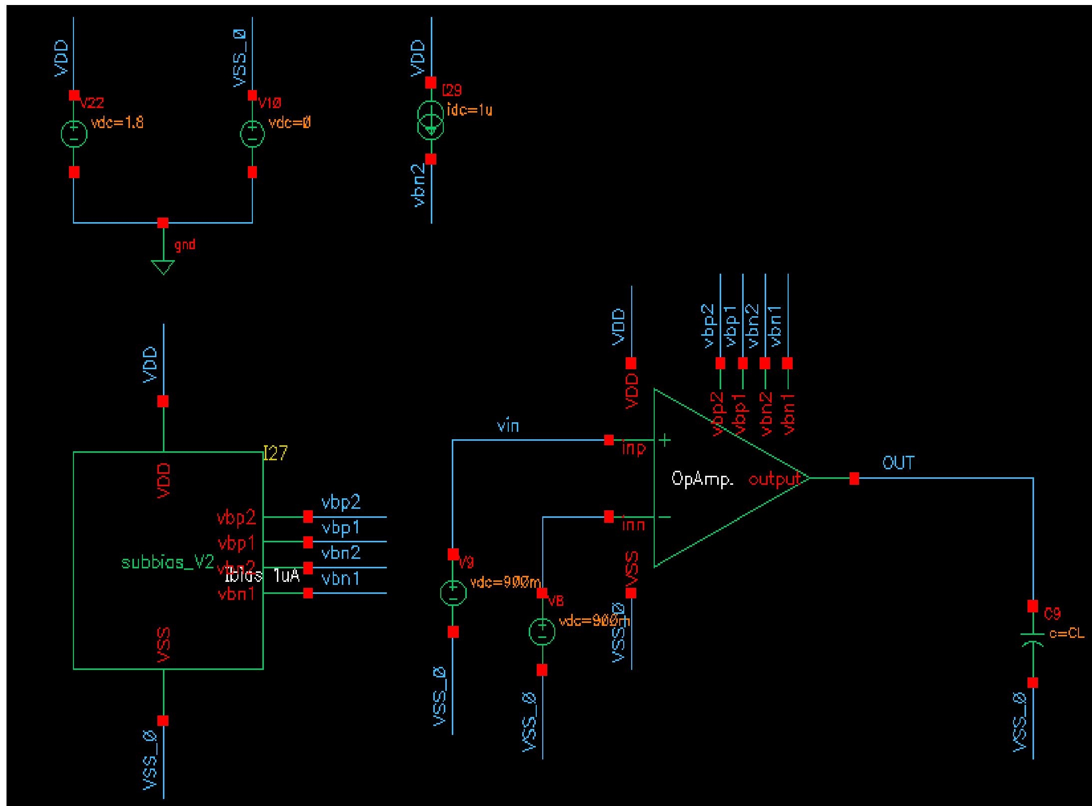


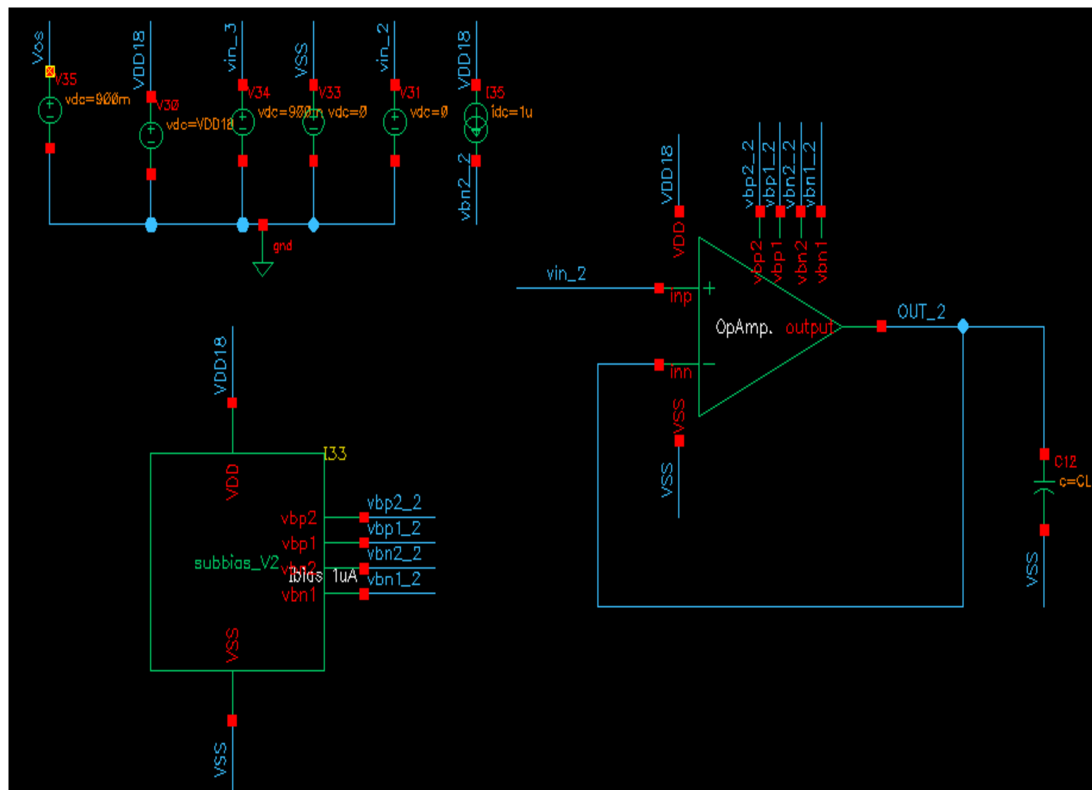
Figure 5.4: OPAMP layouts. (a) Manual layout by an expert. (b) Automatic layout by density-optimized method. (c) Floorplan of automatic layout.

In our implementation, iteration of the layout (b) is not 1 but less than 5. It is because the generated transistor-arrays always satisfy DRC and density constraints though, we still spend time fixing DRC violations on routing nets due to non-convergency of routing. Note that such violations are different from those existing in device layers which are hard to deal with, it is quite easy to fix a violation on a routing net once the DRC violation point is located by a EDA tool, because designers do not have to split layout to adjust device dimension or position. Improving the routing phase for our design flow and making a TA-layout pass all the sign-off checks at a time, is one of our future works. As for the incremental size of TA-layout, it is negligible if the macro-cell layout is placed to a chip with much area margin.

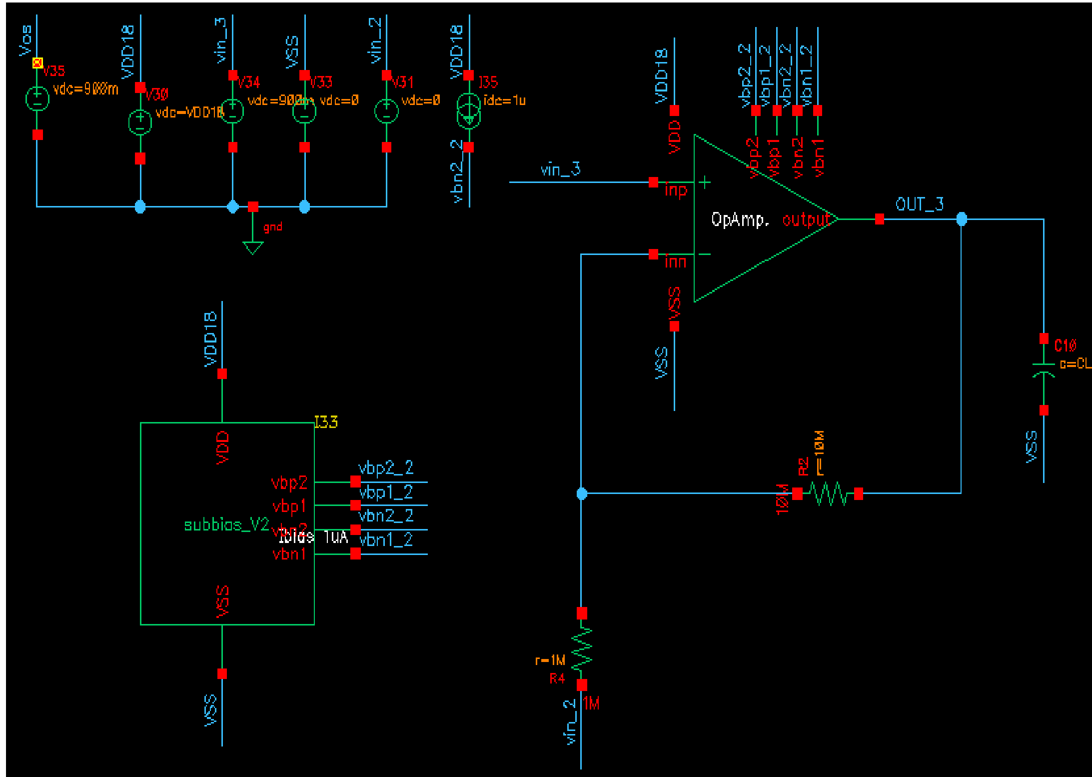
In order to further demonstrate the effectiveness of our method through metrics of interest (as stated in **Chapter 2.4.1**), we set up testbench circuits for pre-simulation of OPAMP circuit and for post-layout simulation of the two layouts, respectively, and measure their metrics. The testbench circuit topology for a metric is the same, as shown in **Figure 5.5, 5.6, 5.7**.



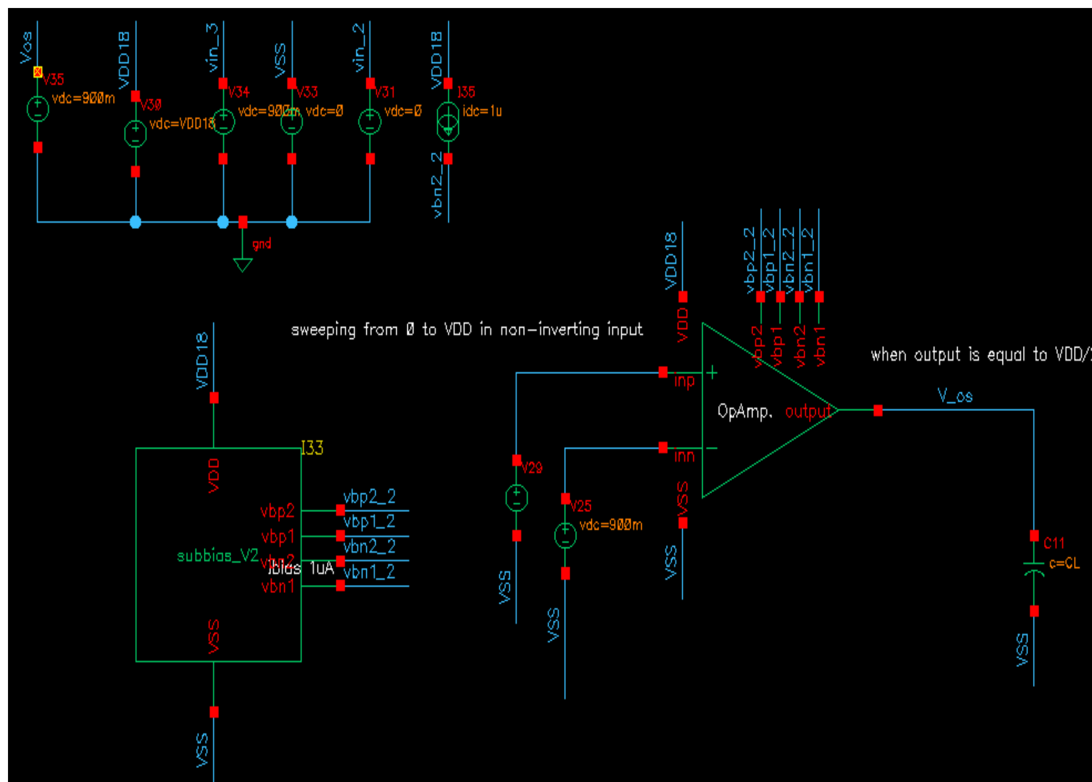
(a)



(b)

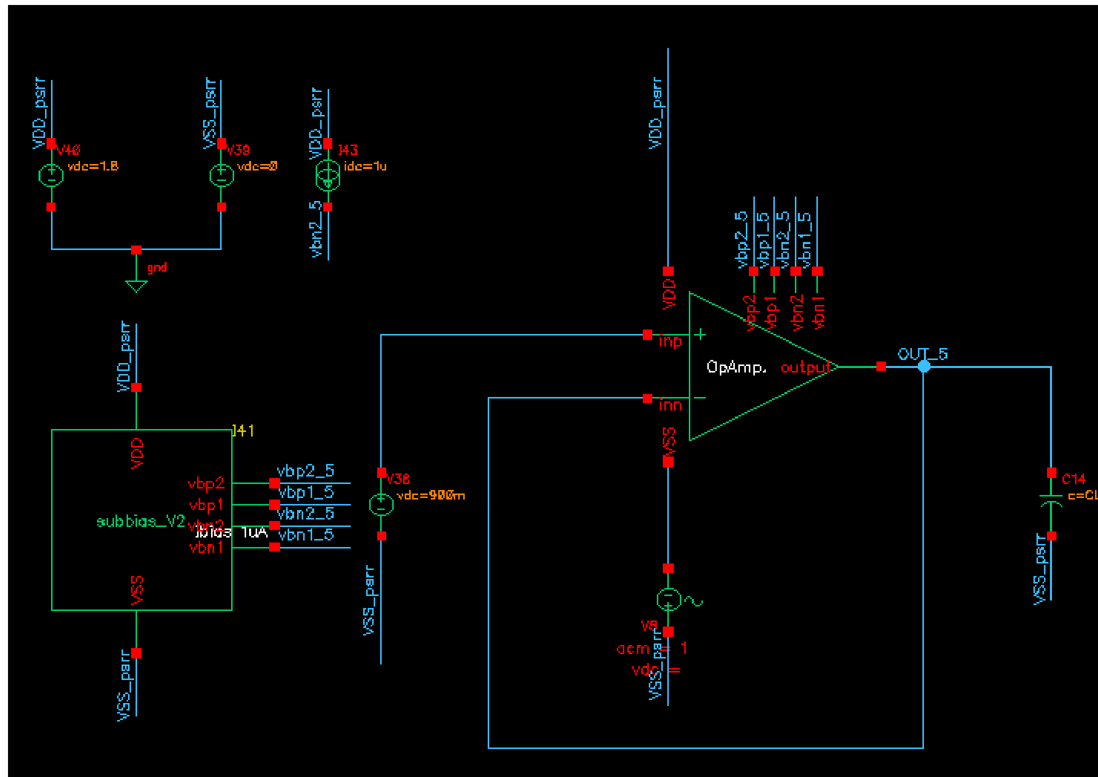


(c)

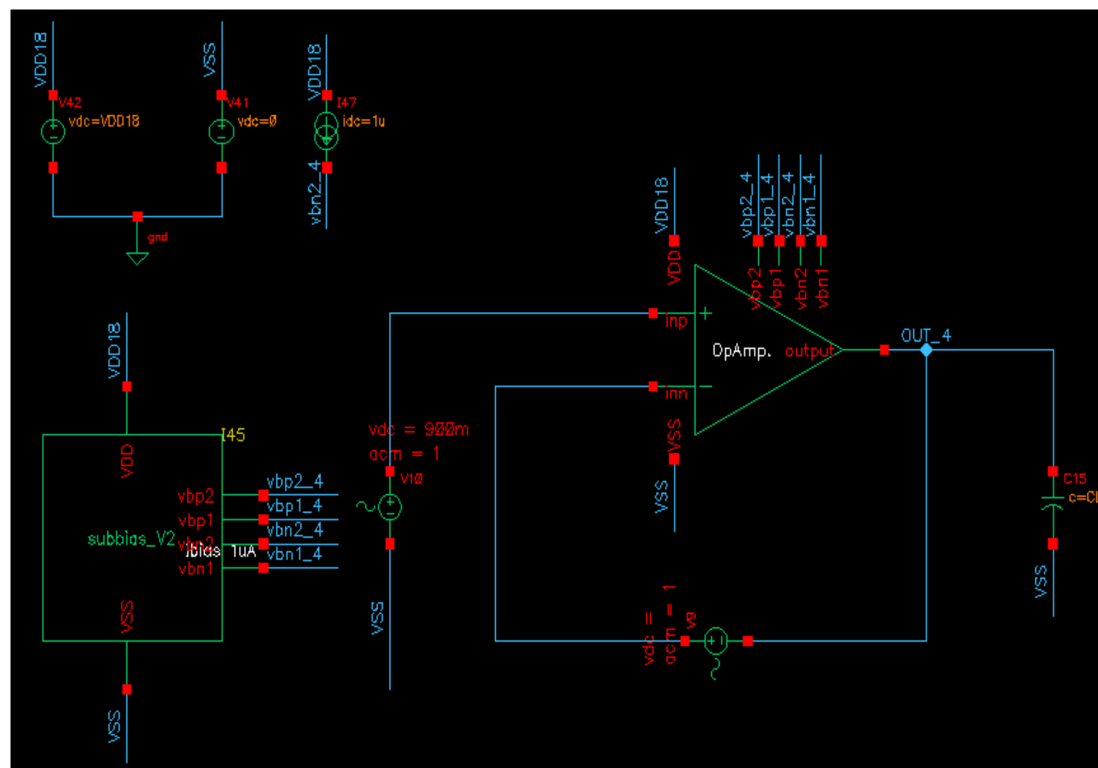


(d)

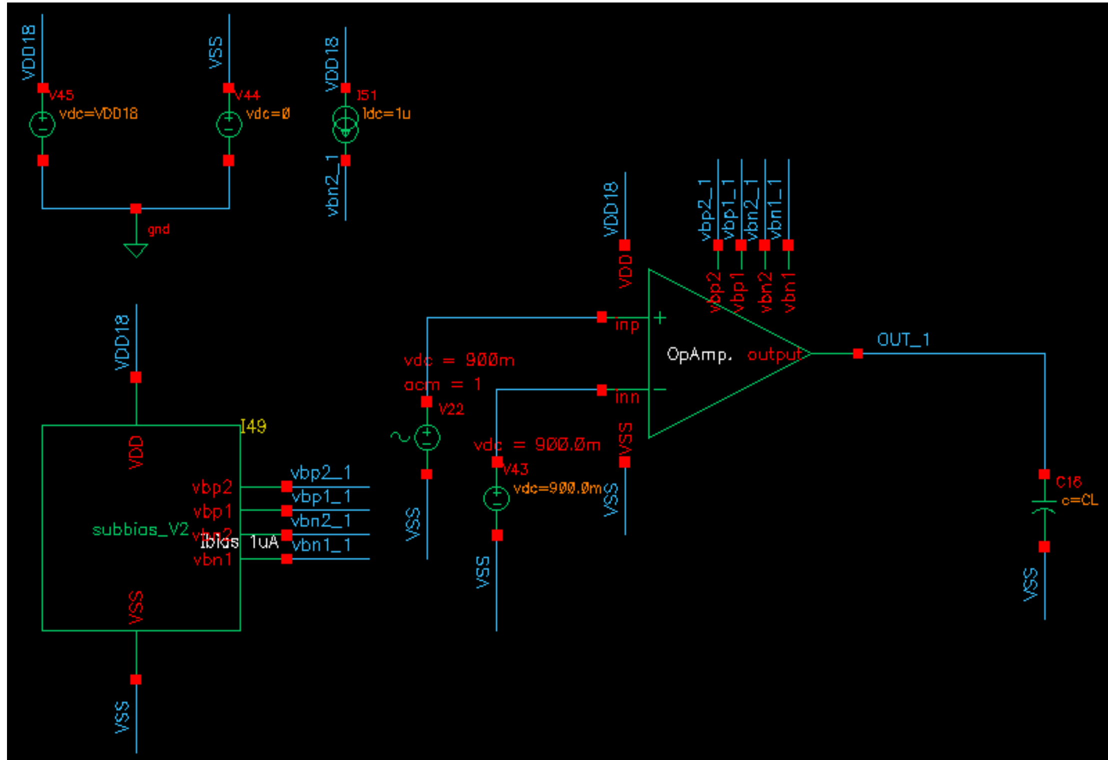
Figure 5.5: Testbench circuits for DC simulation, load capacitance CL is 1.8pF. (a) Measurement for DC operating point. (b) Measurement for ICMR. (c) Measurement for output swing. (d) Measurement for input offset voltage.



(a)



(b)



(c)

Figure 5.6: Testbench circuits for AC simulation, load capacitance CL is 1.8pF. (a) Measurement for PSRR. (b) Measurement for CMRR. (c) Measurement for DC gain, phase margin and GBW.

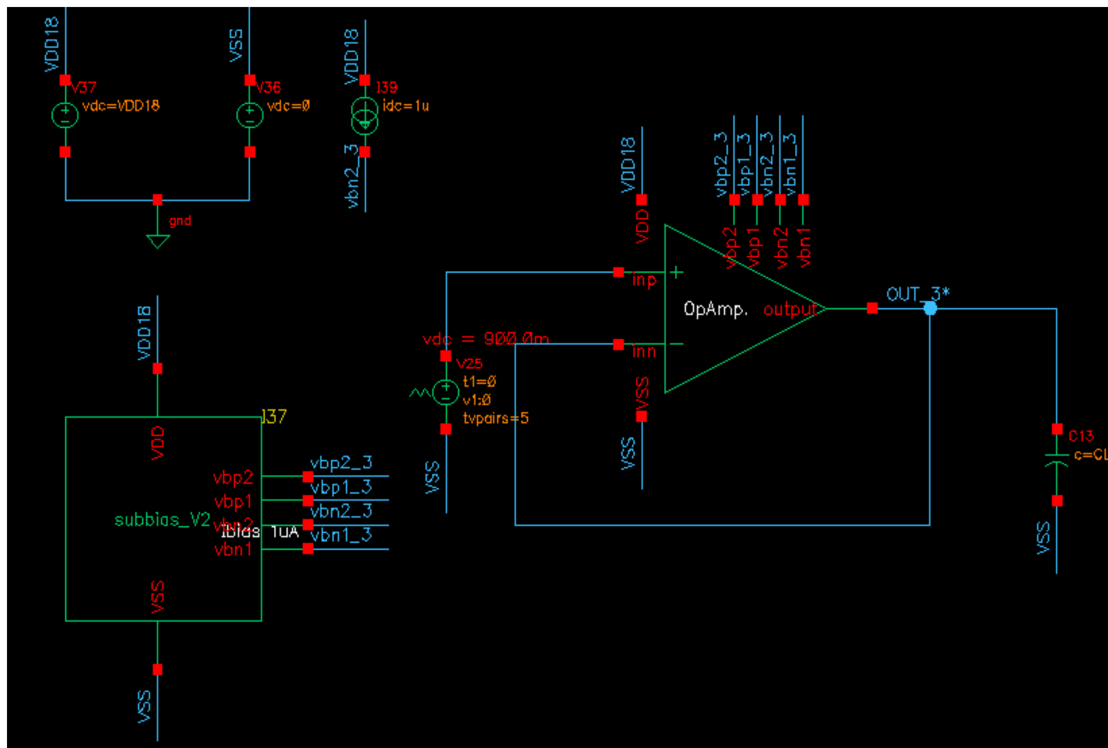


Figure 5.7: Testbench circuits for transient simulation, load capacitance CL is 1.8pF. Measurement for slew rate and setting time.

Table 5.2: Design specification and simulation results.

Performance Parameter (Supply Voltage 0-1.8V)	Design Specification	Pre-simulation	Post-layout Sim. Layout(a)	Post-layout Sim. Layout(b)
Static Power Dissipation(μW)	<6	5.46	5.32	4.98
ICMR(V)	0.2-1.4	0.2-1.48	0.2-1.44	0.18-1.45
Output Swing(V)	0.02-1.7	0.015-1.78	0.017-1.77	0.015-1.78
Input Offset Voltage(mV)	<10	9.9	8.4	6.2
DC/Open-Loop Gain(dB)	>70	82.65	84.19	85.79
Phase Margin(deg)	45-75	60.4	60.2	56.4
GBW(MHz)	>1	1.49	1.47	1.44
CMRR(dB)	-	104.5	105.8	111.4
$PSRR^{+}(dB)/PSRR^{-}(dB)$ @ 0Hz	-/-	68.4/63.9	68.4/62.0	70.9/64.5
Slew Rate($V/\mu s$)	>0.5	0.84	0.78	0.68
Setting Time(μs)	<2	1.22	1.37	1.64
Load Capacitance(pF)	1.8	1.8	1.8	1.8

Furthermore, we extract parasitics for manual layout and automatic layout respectively. Parasitic extraction is performed on Star-RCXT platform, which is the industry standard for silicon-accurate and high-performance extraction of advanced process technologies. We then perform post-layout simulation for the circuit with the extracted parasitics, simulation results shown in **Table 5.2** are from DC and AC as well as transient analysis. For the comparison, design specification and pre-simulation results with respect to performance parameters are listed as well. The only difference with pre-simulation is that, we replace the circuit instance of the original testbench with a new symbol that represents the circuit with parasitics (as stated in **Chapter 2.4.2**).

Both layouts can meet all of the specifications but differ in some performance parameters. As for DC characteristics, there is not much difference between layout (a) and layout (b) in terms of ICMR and output swing. However, layout (b) consumes less power as compared to layout (a), it is because some transistors in layout (b) operate under marginal saturation region according to DC analysis. Compared with layout (a) with all transistors operating under a saturation region, the drain current of the layout (b) is smaller since it has less drain-source voltage margin. This result is also consistent with the conclusion of research by

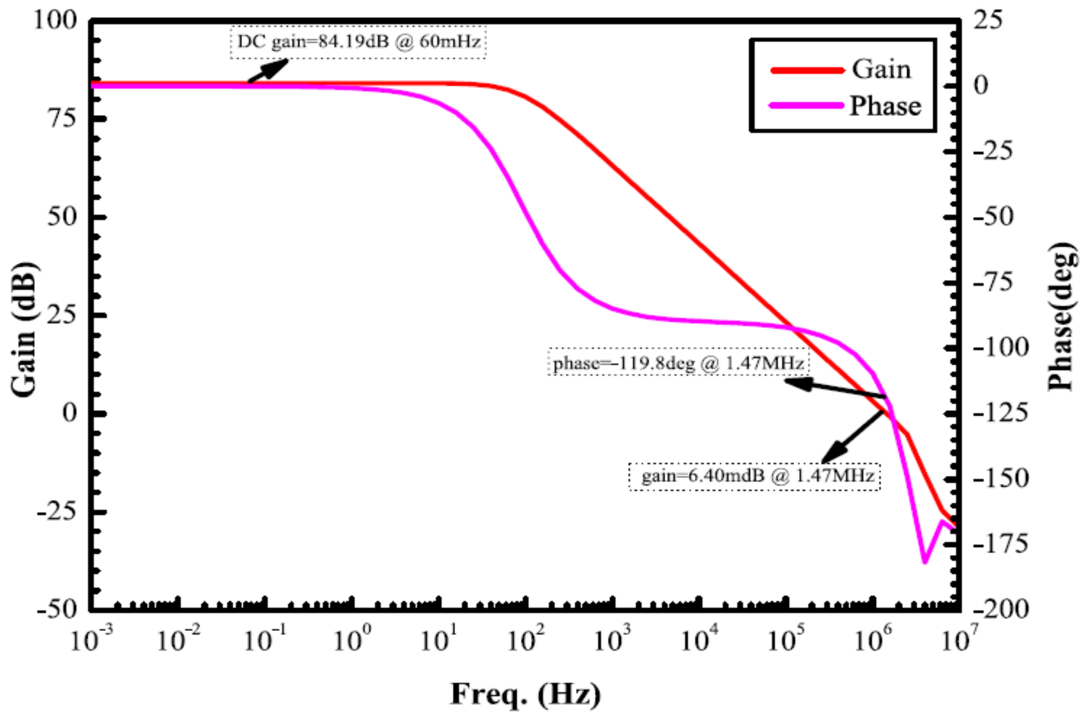
Yang et al. [50] that transistor decomposition in channel length direction contributes to low-power design. Layout (b) has smaller input offset than layout (a), it is because layout (b) is better in terms of transistor matching and circuit symmetry.

As for AC characteristics, due to parasitic effect, it is found that phase margin and unit-gain bandwidth of both layouts degrade compared with pre-simulation results, whereas their DC/open-loop gain increase, as shown in **Figure 5.8**. Due to parasitics existing on interconnects, drain current is actually smaller than that of the circuit without parasitics, and transconductance of the transistor is also reduced. Therefore, phase margin and unit-gain frequency reduce as transconductance becomes smaller. In our design, the open loop gain decreases as transconductance decreases, while it increases as the drain current decreases. However, the open loop gain of both layouts is still increased because the drain current has a greater impact than the transconductance.

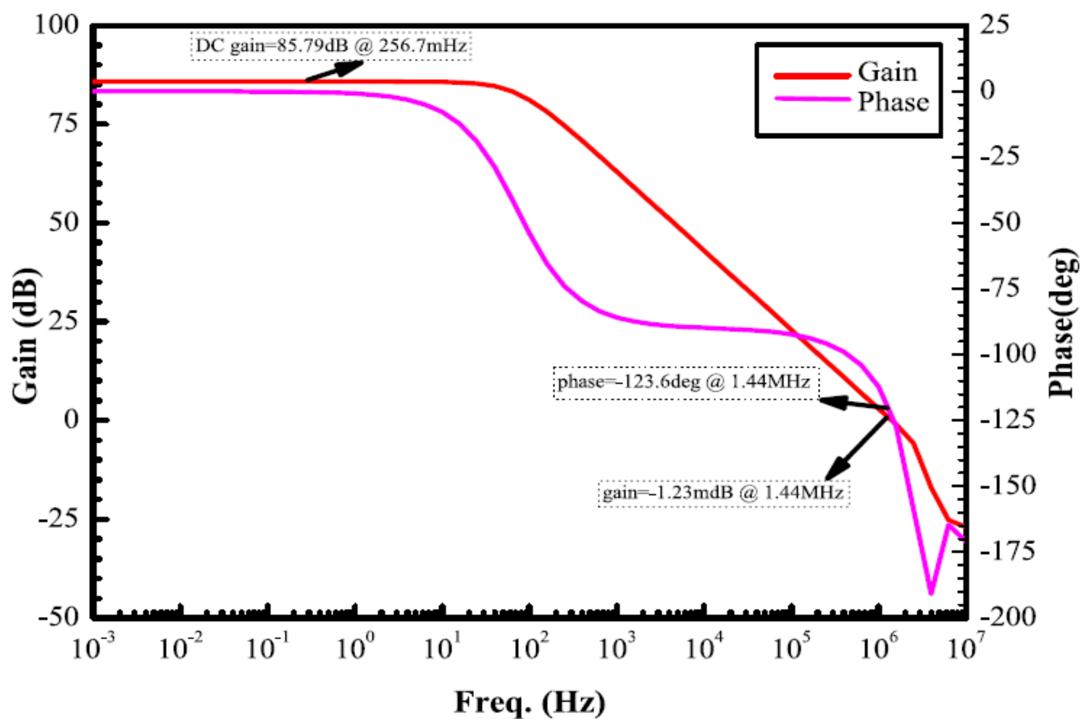
As for DC gain difference of 1.6 dB by layout, it is because the drain current of the layout (b) is smaller than that of the layout (a), therefore it has higher DC gain. It is worth mentioning that CMRR and PSRR of the layout (b) have improvements when compared to the pre-simulation results, whereas layout (a) has no significant difference. As for transient analysis, compared with pre-simulation results, it is found that slew rate and setting time of both layouts degrade due to parasitic capacitance. Nevertheless, it doesn’t show much effect on performance as such degradation is within a tolerance range. Besides, it is not essential for improving circuit performance as our research focuses on manufacturability.

Table 5.2 indicates that both layouts have their merits and demerits, it is hard to determine who is preferable, since the interaction between OPAMP performance parameters is sophisticated. Our OPAMP is employed in a low-pass filter to deal with a weak signal with thousands of Hertz. In terms of specification requirements, automatic layout generally preserves the performance of the circuit as good as manual layout. In summary, our design method in the implementation of the

OPAMP layout is effective, and it can be applied to some fields considering the trade-off of the design time, sign-off checks and application.



(a)



(b)

Figure 5.8: Post-layout simulation. (a) AC simulation results of manual layout. (b) AC simulation results of automatic layout.

Figure 5.9: Summary of our whole research.

The flow of our whole research is summarized in **Figure 5.9,** which is mainly comprised of four parts: part 1 consists of the circuit and algorithm; part 2 consists of the program and solution to density optimization; part 3 consists of the design flow and layout generation; part 4 consists of the verification and post-layout simulation.

In the end, we also give some discussion and justify the effectiveness of our method despite that we use a "low-speed" OPAMP as a design example. As process technologies continue to decrease, the parasitic effect has become more and more significant, especially for advanced technology nodes. Not like the early processes, for a "low-speed" OPAMP, there is likely no significant difference between the pre-simulation and post-layout simulation results. Therefore, it is difficult to check with a low-speed OPAMP whether there are degradations of OPAMP performance due to dummy features. In our case study, we have observed a significant difference between the pre-simulation and post-layout simulation results, despite OPAMP circuit in this work is not a high-speed circuit.

As we always emphasize that, our research focuses on the advanced technology nodes. Therefore, even for a low-speed OPAMP in our 65nm CMOS process, post-layout simulation results have also verified the existence of the parasitic effect. Since layout design is critical to affect the actual performance of a design, we perform the pre-simulation and post-layout simulation for both layouts, a manual layout and a TA-layout. We summarize the simulation results regarding performance parameters into a table (**Table 5.2**), and discuss the influence on performance as a result of degradation.

The discussion demonstrates the effectiveness and applicability of the proposed method under consideration of specification requirements.

As for the manual layout, the post-layout simulation conforms that it has deviation with the pre-simulation, that is to say, parasitic capacitances and resistances have affected circuit performance. After increasing the wire width/number of vias to

reduce R and reducing metal area/overlapping metals to reduce C, a circuit designer finalizes the layout as the performance deviation drops to a tolerance range.

As for TA-layout, the post-layout simulation conforms that it meets all the design specifications, though, it has slight degradation as well. However, it is more advantageous than manual layout as to some performance parameters, such as power dissipation, offset, CMRR and PSRR. This is because our method can control the number of contact layer and the area of metals to an appropriate level.

In addition, the dummy metal fill is minimized as we employ "Min-Dum" scheme to reduce dummy transistors. "high-speed" OPAMP needs a large slew rate or gain-bandwidth product, it employs topology of “cascode” and “gain boost”, this structure contains resistors and capacitors.

As we mentioned in the work, design example is mainly used to demonstrate the feasibility of TA-style analog layout design flow. However, we still need to improve our method for designing circuits mixed with resistors and capacitors. To enhance the reliability of our method is one of our future works. On the other hand, a "high-speed" OPAMP is typically used in video/audio applications, our OPAMP mainly aims at dealing with a weak signal with thousands of hertz, offset and temperature drift are main factors that we consider. Therefore, user needs to consider the trade-off of the design time, sign-off checks and application when implementing an analog circuit by employing our method.

CHAPTER 6

CONCLUSION AND FUTURE WORKS

Aiming at density issue that is previously ignored but critical to successful tape-out of advanced process, we propose a design flow to automate analog layout design and accelerate sign-off checks. We incorporate DRC and density constraints into the design process by utilizing a density-aware layout format of TA-style. In our density-optimized approach, the problem is formulated as an optimization problem in order to achieve density uniformity. We present a Min-Dum scheme to avoid exhaustive search on the all feasible solutions, and also simplify the problem as a quadratic programming problem. Our design cases for an OPAMP layout in a 65nm CMOS process, show that our method can reduce the iterative process of verification significantly. Compared with a manual layout by the traditional method, the experimental results demonstrate the high efficiency and the effectiveness of our method.

Our future works are to develop provably-good and practically-efficient algorithms to handle more complex circuits, which consist of:

- Improving the routing phase for our design flow to pass verification in one time.
- Designing more analog circuits with different topologies and complexity by our approach. The present research focuses only on an OPAMP layout design, we can attempt to design more analog circuits, such as, active filters with resistors and capacitors, operational transconductance amplifiers (OTAs), comparators and etc.
- Implementing an analog layout in different process nodes, such as the process nodes below 65nm, so as to generalize our method in more advanced technology process, and to verify the effectiveness of our method under more rigorous design rules.

- Demonstrating the reliability of our method by the silicon measurement results from a fabricated chip. The present experimental results are obtained from the layout implementation. In our future works, we can fabricate a chip and measure the silicon result for the two layouts generated by different methods, which is more reliable to verify the effectiveness of our method.

ACKNOWLEDGEMENTS

Through three more years of studying in Japan, I extend my expertise in the field of mixed-signal integrated circuits. By joining in Panasonic medical sensor project and a research project associated with machine learning, I deepen my knowledge of IC design and obtain an in-depth understanding of my future career. It’s an enjoyable process to submit a paper and attend an international conference. Thanks to the help of my advisor, I’ve made some progress by putting my expertise into practice. Benefitting from the research environment of our school, up to now, I’ve mastered the whole flow of IC design, comprising of circuit design and simulation, physical implementation and verification, chip testing. I become familiar with using EDA tools and reading technical documents provided by foundry, which are the essential ability to a researcher.

Hereby, I especially thank my advisor Prof. Dr. Nakatake for his tremendous help, support and encouragement. He is not only our mentor guiding us in research direction but also our friend bringing us with lots of joys. I’m also deeply grateful to his education and kindness as well as generosity. With his education, I gained not only the expertise in IC design but also the ability to collaborate with partners. His profound knowledge and rigorous attitude toward research inspire me to grow and will benefit me in my future professional and personal life.

Special thanks go to professors from the Dissertation Defense Committee of Dept. of Information and Media Engineering, for the efforts they put in this work.

I am also deeply grateful to The University of Kitakyushu and Japanese Society for their support on my study and livelihood. They help patiently for me to solve the problem whenever I encounter a headache. Having been living in Japan for 5 years, I feel the politeness and friendship of Japanese people. I hope to be a bridge of bilateral ties for promoting the mutual understanding of two peoples, I also hope to be a window

for Chinese people to study merits of the rest part of the world, and to be always humble and connected with other countries.

I would like to thank Dr. Liu for his guidance in my preliminary research, his suggestion is essential to this research from the idea to a paper composition. His patience for children and professional dedication for work inspire me a lot. His continuous encouragement and discussion with me are one of reasons to make me come to where I am.

I also would like to thank the members from the Naka lab for the help and happy memories with them, though, some of them have left for work. Throughout this long journey of my life, it's a short but enjoyable journey for studying in Japan, I would never forget their names.

Last but also the most important, I offer my deepest gratitude and love to my parents, sister and my girlfriend, for their unconditional love, support and confidence in me. Without their support, I could not have persisted so far.

REFERENCES

[1] W. Maly, "Computer-aided design for VLSI circuit manufacturability," in Proc. of IEEE, vol.78, no.2, pp.356–392, Feb. 1990.

[2] W.C. Tam and S. Blanton, "Design-for-manufacturability assessment for integrated circuits using RADAR," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.33, no.10, pp.1559–1572, Sept. 2014.

[3] G. Debnath and P.J. Thadikaran, "Design for manufacturability and reliability in nano era," in Proc. of International Conference on VLSI Design, pp.33–34, Jan. 2009.

[4] R. Puri and D.S. Kung, "The dawn of 22 nm era: Design and CAD challenges," in Proc. of International Conference on VLSI Design, pp.429–433, Jan. 2010.

[5] D.O. Oumaand D.S. Boning, J.E. Chung, W.G. Easter, V. Saxena, S. Misra, and A. Crevasse, "Characterization and modeling of oxide chemical-mechanical polishing using planarization length and pattern density concepts," IEEE Trans. Semicond. Manuf., vol.15, no.2, pp.232–244, Aug. 2002.

[6] R. Tian, D.F. Wong, and R. Boone, "Model-based dummy feature placement for oxide chemical mechanical polishing manufacturability," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.20, no.7, pp.902–910, July 2001.

[7] D. Boning and B. Lee, "Nanotopography issues in shallow trench isolation CMP," MRS Bull., vol.27, no.10, pp.761–765, Oct. 2002.

[8] A.B. Kahng and K. Samadi, "CMP fill synthesis: A survey of recent studies," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.27, no.1, pp.3–19, Jan. 2008.

[9] L. Wu, "Modeling of wafer topography's effect on chemical–mechanical polishing process," IEEE Trans. Semicond. Manuf., vol.20, no.4, pp.439–450, Nov. 2007.

[10] H. Melzner, N. Cui, and S. Postnikov, "Navigating in the density plane – achieving flat wafer surface and uniform metal thickness in CMP processes," in Proc. of Advanced Semiconductor Manufacturing Conference, pp.189–195, July 2010.

[11] H. Lachkar, O. Rizzo, J.-M. Portal, and O. Ginez, "Layout uniformity: A metric for yield enhancement," in Proc. of International Midwest Symposium on Circuits and Systems, pp.1–4, Sept. 2011.

[12] A.B. Kahng, G. Robins, A. Singh, and A. Zelikovsky, "Filling algorithms and analyses for layout density control," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.18, no.4, pp.445–462, April 1999.

[13] G. Shomalnasab and L. Zhang, "Analog layout density uniformity improvement using interconnect widening and dummy fill insertion," in Proc. of IEEE Int. Symposium on Circuits and Systems, pp.1–4, Sept. 2017.

[14] W. Yu, M. Zhang, and Z. Wang, "Efficient 3-D extraction of interconnect capacitance considering floating metal fills with boundary element method," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.25, no.1, pp.12–18, Jan. 2006.

[15] Y. Chen, A.B. Kahng, G. Robins, and A. Zelikovsky, "Hierarchical dummy fill for process uniformity," in of Proc. 2001 Asia and South Pacific Design Automation Conference, pp.139–144, Aug. 2001.

[16] B.E. Stine, D.S. Boning, J.E. Chung, L. Camilletti, F. Kruppa, E.R. Equi,W. Loh, S. Prasad, M. Muthukrishnan, D. Towery, M. Berman, and A. Kapoor, "The physical and electrical effect s of metal fill patterning practices for oxide chemical

mechanical polishing processes," IEEE Trans. Electron Devices, vol.45, no.3, pp.665–679, March 1998.

[17] Y. Tao, C. Yan, Y. Lin, S.-G. Wang, D.Z. Pan, and X. Zeng, "A novel unified dummy fill insertion framework with SQP-based optimization method," in Proc. of IEEE/ACM Int. Conf. on Computer-Aided Design, pp.1–8, Nov. 2016.

[18] Y. Lin, B. Yu, and D.Z. Pan, "High performance dummy fill insertion with coupling and uniformity constraints," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.36, no.9, pp.1532–1544, Sept. 2017.

[19] M. Mukherjee and K. Chakraborty, "A randomized greedy method for rectangular-pattern fill problems," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.27, no.8, pp.1376–1384, Aug. 2008.

[20] A.B. Kahng, G. Robins, A. Singh, H. Wang, and A. Zelikovsky, "Filling and slotting: Analysis and algorithms," in Proc. of Int. Symposium on Physical Design, pp.95–102, April 1998.

[21] H.Y. Foo, K.W.C. Leong, and R. Mohd-Mokhtar, "Density aware interconnect parasitic estimation for mixed signal design," in Proc. of IEEE International Conference on Circuits and Systems, pp.258–262, Oct. 2012.

[22] H. Xiang, L. Deng, R. Puri, K.-Y. Chao, and M.D.F. Wong, "Fast dummy-fill density analysis with coupling constraints," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.27, no.4, pp.633–642, April 2008.

[23] H.-Y. Chen, S.-J. Chou, S.-L. Wang, and Y.-W. Chang, "Novel wire density driven full-chip routing for CMP variation control," in Proc. of IEEE/ACM Int. Conf. Comput. Aided Des., pp.831–838, Nov. 2007.

[24] B. Yang, Q. Dong, J. Li, and S. Nakatake, "Structured analog circuit design and MOS transistor decomposition for high

accuracy applications," in Proc. of IEEE/ACM ICCAD, pp.721–728, Dec. 2010.

[25] B. Yang, Q. Dong, J. Li, and S. Nakatake, "Structured analog circuit and layout design with transistor array," IEICE Trans. Fundamental, vol.E96-A, no.12, pp.2475–2486, Dec. 2013.

[26] G. Chen, T. Fujimura, Q. Dong, S. Nakatake, and B. Yang. "DC characteristics and variability on 90 nm CMOS transistor arraystyle analog layout," ACM Trans. Des. Autom. Electron. Syst. (TODAES), vol.21, no.3, pp.45:1–45:21, May 2016.

[27] A. J. Strojwas and S. W. Director, "VLSI: linking design and manufacturing," IEEE Spectrum, vol. 25, no. 10, pp. 24-28, Oct. 1988.

[28] K. Hui, L. Ke and S. Y. Sheen, "Advanced Process Controls of Underdetermined Systems of Feature Profiles in Semiconductor Manufacturing," in Proc. of 2019 30th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), pp. 1-6, Aug. 2019.

[29] D. A. Hodges, L. A. Rowe and C. J. Spanos, "Computer integrated manufacturing (semiconductor processing)," in Proc. of Seventh IEEE/CHMT International Electronic Manufacturing Technology Symposium, pp. 1-3, Sept. 1989.

[30] "CMOS Layout Design: Introduction," VLSI EXPERT, Nov. 2014, http://www.vlsi-expert.com/2014/11/cmos-layout-design.html.

[31] Xie, Xiaolin. "Physical understanding and modeling of chemical mechanical planarization in dielectric materials." PhD diss., MIT, 2007.

[32] "Manufacturing a profit," TECH DESIGN FORUM, May 2010,https://www.techdesignforums.com/practice/technique/manufacturing-a-profit/.

[33] R. Puri and D. S. Kung, "The Dawn of 22nm Era: Design and CAD Challenges," in Proc. of 2010 23rd International Conference on VLSI Design, pp. 429-433, Jan. 2010.

[34] H. S. Lee, "IC design challenges and opportunities for advanced process technology," in Proc. of VLSI Design, Automation and Test(VLSI-DAT), pp. 1-2, Jun. 2015.

[35] S. S. Sapatnekar, "Physical Design Automation Challenges for 3D ICs," in Proc. of 2006 IEEE International Conference on IC Design and Technology, pp. 1-1, Aug. 2006.

[36] B. Yu et al., "Dealing with IC manufacturability in extreme scaling (Embedded tutorial paper)," in Proc. of 2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 240-242, Dec. 2012.

[37] A. J. Strojwas, "Design for manufacturability: a path from system level to high yielding chips," in Proceedings of 2000 Design Automation Conference. (IEEE Cat. No.00CH37106), pp. 375-376, Jan. 2000.

[38] A. B. Kahng, "IC layout and manufacturability: critical links and design flow implications," in Proceedings of Twelfth International Conference on VLSI Design. (Cat. No.PR00013), pp. 100-105, Jan. 1999.

[39] W. C. Tam and S. Blanton, "Design-for-Manufacturability Assessment for Integrated Circuits Using RADAR," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 33, no. 10, pp. 1559-1572, Oct. 2014.

[40] "Chemical Mechanical Polishing / Planarization: Slurry Measurements," Anton Paar, Accessed Sept. 2020, https://www.antonpaar.com/corpen/products/applications/chemical-mechanical-polishing-planarization-slurry-measurements/.

[41] Mark Osborne, "CMP system tackles tungsten films at lowest cost," FABTECH, Jun. 2011, http://www.fabtech.org/product_briefings/_a/applied_materials_reflexion_gttm_cmp_system_tackles_tungsten_films_at_lowes/.

[42] “Importance of CMP Process,” VLSI EXPERT, Jul. 2015,http://www.vlsi-expert.com/2015/07/importance-of-cmp-process.html.

[43] Dick James, “Intel’s e-DRAM Shows up in the Wild,” Chipworks,Feb.2014,http://chipworksrealchips.blogspot.com/2014/02/intels-e-dram-shows-up-in-wild.html.

[44] C. Dong, Q. Zhou, Y. Cal and D. Liu, "Partition-based global placement considering wire-density uniformity for CMP variations," in Tsinghua Science and Technology, vol. 16, no. 1, pp. 41-50, Feb. 2011.

[45] “Dishing and Erosion (CMP),” VLSI Expert, Aug. 2015, http://www.vlsi-expert.com/2015/08/.

[46] Boning D., Lee B., “Nanotopography Issues in Shallow Trench Isolation CMP,” in MRS Bulletin, vol. 27, no.10, pp. 761-765, Oct. 2002.

[47] B. Liu, G. Chen, B. Yang, and S. Nakatake, “Routable and matched layout styles for analog module generation,” ACM Trans. Des. Autom. Electron. Syst. (TODAES), vol.23, no.4, pp.47:1–47:17, July 2018.

[48] Q. Ma, E.F.Y. Young, and K.P. Pun, “Analog placement with common centroid constraints,” in Proc. of 2007 IEEE/ACM international conference on Computer-aided design, pp.579–585, Dec. 2007.

[49] T. Yoshimura and E.S. Kuh, “Efficient algorithms for channel routing,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol.1, no.1, pp.25–35, Jan. 1982.

[50] B. Yang, Q. Dong, J. Li, and S. Nakatak, “Analysis of channel decomposition for structured analog layout and low-power applications,” IEICE Technical Report, vol.110, no.316, pp.161–166, Nov. 2010.