Reinforcing the Connection between Analog Design and EDA

(Invited Paper)

Kishor Kunal, Meghna Madhusudan, Jitesh Poojary, Ramprasath S, Arvind K. Sharma, Ramesh Harjani, and Sachin S. Sapatnekar
University of Minnesota, Minneapolis, MN 55455

***Abstract*—Building upon recent advances in analog electronic design automation (EDA), this paper discusses directions for reinforcing the connection between design and EDA, in order to develop solutions that are meaningful to designers. Two aspects, both related to bridging the gap between EDA and designers, are highlighted. The first discusses the use of test structures to generate meaningful characterized data to aid design automation, specifically understanding the impact of random, correlated, and systematic variations on the design of matched structures. Results on a recent test chip that analyzes these variations and their impact on EDA design choices will be presented. The second illustrates a design testcase that applies analog EDA techniques, using the ALIGN layout engine, to design an RF MIMO receiver, and describes how this experience has helped both in advancing the state of analog EDA and in building circuits with enhanced designer productivity.**

I. Introduction

There has been substantial progress in the area of analog design automation in recent years. The task of automated layout generation, in particular, has seen strong and protracted efforts through the development of tools and flows such as BAG2 [1], MAGICAL [2], [3], and ALIGN [4], [5]. However, beyond the development of algorithms and software, there is a significant need for analog EDA tools to engage with the overall semiconductor ecosystem, from design to fab, to generate solutions that create a compelling case for the widespread use of these tools and flows. Building this strong connection can better demonstrate the value proposition of emerging analog EDA solutions, encouraging their widespread adoption.

In this paper, we present some experiences in our work with ALIGN. ALIGN consists of a suite of tools that translates a circuit netlist to a GDSII layout. The ALIGN effort involves the development of a systematic approach, novel algorithms, and open-source software [6]. Our efforts in this direction have resulted in substantial advances in both algorithmic techniques [7], [8], [9], [10], [11], [12], [13], [14], [15], [16] and machine learning methods as applied to analog design [17], [18], [19], [20], [21], [22], [23], [24]. ALIGN has been used in multiple industry and academic settings on a variety of designs [25], [26], [27], [28].

Beyond the development of new alogrithms than underpin ALIGN, this paper focuses on two experiences: one related to connecting improved analysis with the design of variation-tolerant systems [29], and another to using ALIGN to build an RF MIMO receiver with performance comparable to human design, but with better area and significantly improved designer productivity [30].

II. Characterizing Mismatch

Motivation. Precision analog circuits such as amplifiers, comparators, and data converters are sensitive to device mismatch due to within-die process variations. Mismatch may be caused by (1) uncorrelated variations, whose mismatch decreases as device size increases [31], [32], [33]; (2) systematic distance-dependent variations, typically modeled as linear gradients [34]; (3) correlated random variations [32] where the standard deviation of the mismatch increases with distance. In analog circuits, common-centroid (CC) techniques are widely used to reduce linear distance-dependent mismatch [35], [36]. However, CC patterns, which require more routing resources than clustered (NonCC) patterns, incur cost overheads in both their area and parasitics [37], [8]. These costs are more significant in newer technology nodes, where resistive metal parasitics are higher and design rules force unidirectional wires and high via counts [38].

The existing literature on device variations has typically been applied to old technology nodes [39], [40] rather than newer technologies in FinFET nodes and beyond. Several models for distance-dependence within-die variations target digital applications [41], where the norm is to use minimum device length, L, and the smallest possible device width, W; in contrast, analog designers typically use large W and larger-than-minimal L to control uncorrelated variations, and are more interested in mismatch than absolute variation. Understanding and characterizing these variations can be of great use in building layouts that reduce transistor mismatch.

Test Chip Design and Measurement Setup. Our test circuit for mismatch characterization is shown in Fig. 1(a), and the corresponding floorplan and die photo are depicted in Fig. 1(b). Our device under test (DUT) in a commercial 12nm technology is a stacked diode-connected FinFET NMOS that has an effective W/L equal to 1.15μm/0.28μm. Since layout-dependent effects (LDEs) are acute in FinFET nodes, we add dummies to each DUT, and the neighborhood of each element is matched. An element, shown by the blue box in Fig. 1(a), consists of a DUT and NMOS/PMOS switches (S_1, S_2, S_3, and S_4). To measure a DUT, the switches connected to it are turned on by the output of a row/column decoder. A current source, I_D, is connected to the drain of the DUT, via transistors S_1/S_2, and the V_{GS} of the DUT is measured via transistors S_3/S_4. This three-point measurement technique avoids any voltage drop across the transistors S_1/S_2 that supply current, as voltage measurements are carried out via transistors S_3/S_4 that do not carry any current. The off-chip high-gain, low offset OTA in the return current path ensures the absence of on-chip IR drops. The virtual ground, plus the zero current flowing through the bottom path ensures that the source of DUT is at the same voltage as the OTA positive input (0V). For the diode-connected DUT, the threshold voltage, V_{TH}, and the current factor, β, can be calculated from the measured V_{GS} for two or more I_D values.

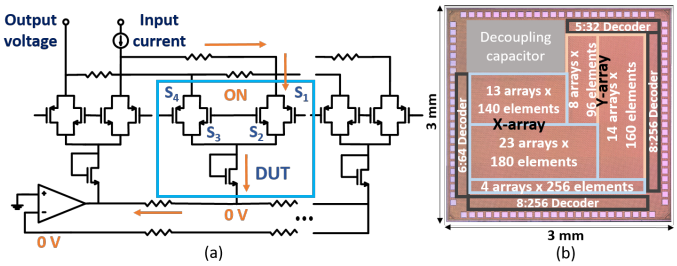


Fig. 1. (a) Schematic of our DUT. (b) Die photo and floorplan.

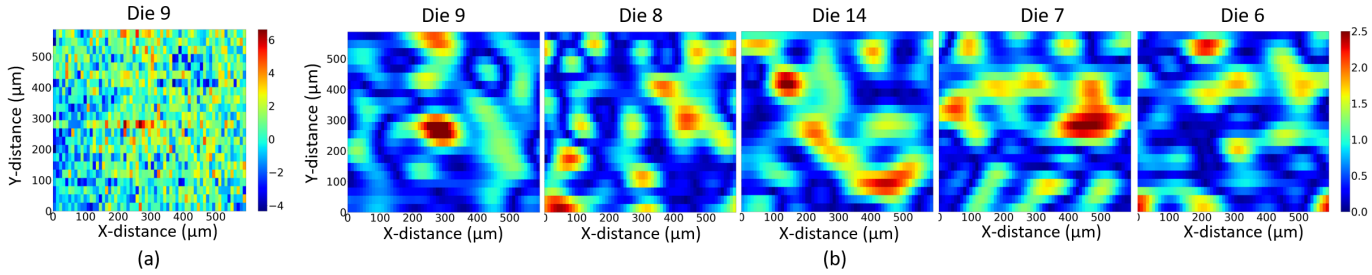


Fig. 2. (a) ΔV_{TH} surface on a die before filtering. (b) ΔV_{TH} surface on multiple dice after low-pass filtering. Note that the colorbar range for (b) is 4× smaller than for (a) for visual clarity.

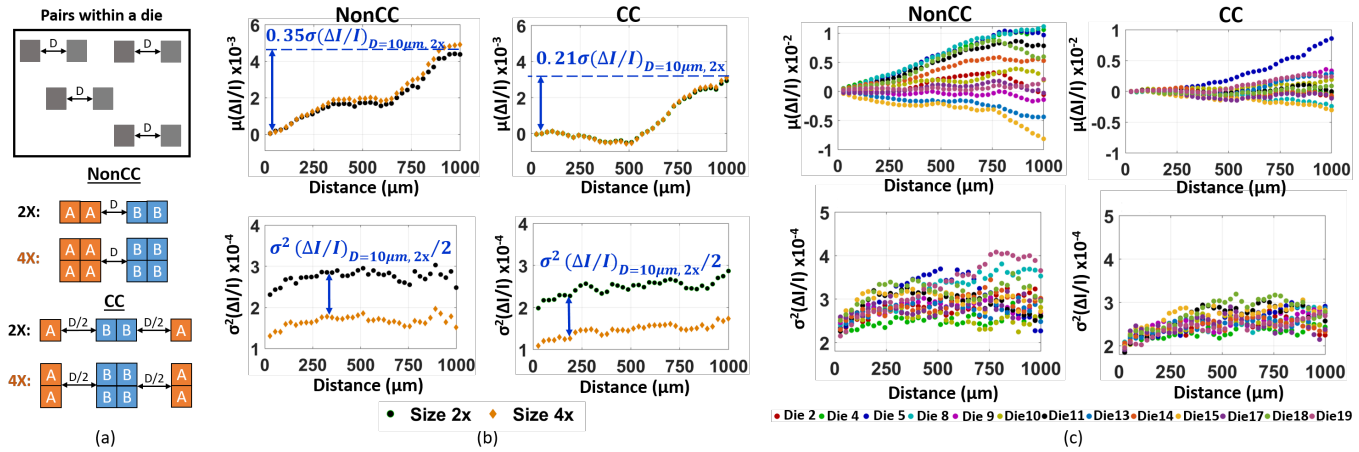


Fig. 3. (a) NonCC and CC patterns used to find $\Delta I/I$. (b) μ and σ of $\Delta I/I$ distribution of devices arranged in NonCC and CC and separated by distance, D. (c) μ and σ of $\Delta I/I$ distribution of devices of Size 2x on multiple dice.

We perform multiple measurements on the same device at 292K and report the median value; we confirm that the measurements have a low spread, validating their consistency.

Visualizing the ΔV_{TH} Surface on a Die. Fig. 2 shows ΔV_{TH}, which is the difference between the V_{TH} at a location and the mean V_{TH} over the 600μm×600μm die area. Fig. 2(a) shows the extracted ΔV_{TH} containing both the random and distance-dependent components. To see the distance-dependent component, we low-pass-filter the extracted ΔV_{TH} surface to remove the random component. Fig. 2(b) shows the ΔV_{TH} surface on multiple dice, where the zero in the color bar corresponds to the lowest value on the die as an artifact of filtering. The ΔV_{TH} surfaces in Fig. 2(b) have spatially correlated regions and are different on each die.

Devices Arranged in NonCC and CC. Based on the extracted V_{TH} and β values from measurement, we infer the currents of each DUT using simulation; details are provided in [29]. We then find the current mismatch, $\Delta I/I$, between device pairs arranged in NonCC and CC as shown in Fig. 3(a). We consider two different device sizes, Size 2x, and Size 4x having 2, and 4 unit cells, respectively, each of size W/L = 1.15μ/0.28μ.

We plot the mean, $\mu(\Delta I/I)$, and variance, $\sigma^2(\Delta I/I)$, of the $\Delta I/I$ distribution in Fig. 3(b) for both device sizes in NonCC (left) and CC (right) on one of the measured die. From the figure, we see that the mean is virtually independent of the device size and its magnitude is small when compared to $\sigma(\Delta I/I)$ of the Size 2x device even though it has a larger area. Moreover, since the devices arranged in the CC pattern cancel linear variations, a non-zero mean, as seen at the top right of Fig. 3(b), indicates nonlinear variations.

As expected from [31], [32], we see that the $\sigma^2(\Delta I/I) \propto 1/(WL)$ at minimum D since the random component dominates at this spacing. The distance-dependent component of $\sigma^2(\Delta I/I)$, however, remains the same for the different sizes as the curves show the same trend as D increases (also observed in [32]). This change in $\sigma^2(\Delta I/I)$ with distance in both CC and NonCC is a result of correlated spatial variations.

Next, we show how NonCC and CC formats are affected by variations on multiple dice for Size 2x devices, in Fig. 3(c). Here, we see that CC has a lower $\mu(\Delta I/I)$ and $\sigma^2(\Delta I/I)$ than NonCC since it cancels linear distance-dependent components. However, at short distances, the distance-dependent component itself is a small fraction of the random variations, that affect both NonCC and CC, hence its impact is lower. In summary, (a) within-die distance-dependent variations have nonlinear and spatially correlated components that affect both NonCC and CC, (b) the distance-dependent component is a small fraction of the random component even for large device sizes. Therefore, at short distances, CC may provide little advantage over NonCC as random variations dominate.

Impact on Analog Design. We find the combined distribution of $\Delta I/I$ of a minimum of 11,006 device pairs from 19 dice, at each distance D. We extract the distance-dependent mean, $\mu(\Delta I/I)_D$, and variance, $\sigma^2(\Delta I/I)_D$ by subtracting out the random component (variations at $D = 10\mu$m) and plot them for both NonCC and CC patterns in Fig. 6. Clearly, both patterns are affected by these variations. However, as noted earlier, the reduction in $\mu(\Delta I/I)_D$, and variance, $\sigma^2(\Delta I/I)_D$ for CC patterns is due to the cancellation of the linear components. Before applying this understanding to unary current-steering DACs we discuss the area difference between NonCC and CC layouts.

NonCC vs. CC Layout Area. In Fig. 7, we see example layouts of a current mirror circuit in 12nm FinFET showing a NonCC and a CC pattern. Here, devices A and B have the same size (WL) with 2 unit

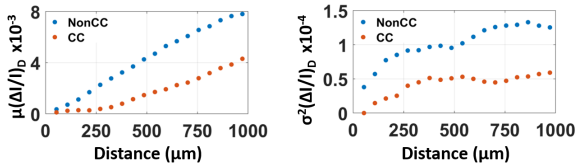


Fig. 6. Average distance-dependent variations from multiple dice.

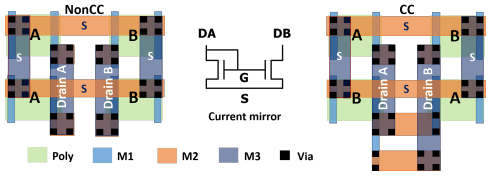


Fig. 7. Example NonCC and CC layouts of a 2-device (4 unit) current mirror (NonCC area < CC area)

cells and their source, and gate connections (not shown) are the same in both layouts. The drain connections, however, require more area in CC (one additional horizontal track), as shown in Fig. 7, because of the cross-coupled connection. Furthermore, the number of extra routing tracks in CC increases linearly with the number of devices in a row, resulting in larger area and parasitics.

Designing Current-Steering DACs. We apply our findings to unary DAC designs, designed to meet a mismatch specification on $\sigma^2(\Delta I/I)_{spec}$ for a 99.7% yield [36].

In the **6-bit DAC** with 63 devices, the distance-dependent mismatch is a small fraction ($\leq$ 5%) of the mismatch specification for NonCC and CC, and the required device size (WL) is similar for both as shown in Table I. However, the estimated layout area is higher in CC owing to the extra routing area required for drain connections as explained in Fig. 7. In the **8-bit DAC**, with 255 devices, the distance-dependent mismatch is 30% and 10% of the mismatch specification for NonCC and CC, respectively, and NonCC requires a larger device size than CC to meet the specifications. Even with this increase in device size, NonCC has a lower layout area when compared to CC. Here, we neglect the $\mu(\Delta I/I)$ since it is small compared to $\sigma(\Delta I/I)$. Therefore, in both these cases, NonCC is more advantageous than CC while meeting specifications and these finding can be extended to small devices in OTAs and lower-resolution DACs.

For the **10-bit DAC** with 1023 devices, it is not possible to meet the mismatch specification with NonCC as shown in Table I, and hence CC is required. Also, for the double CC pattern used here the distance between devices closer to the center and the edge is high, the distance-dependent mismatch is $\geq$ 50% of the total mismatch and the device size is very large, hence, this is not an optimal CC pattern, and other CC patterns with less distance between devices can be used. Since distance-dependent variations are often not included in foundry-provided models our measurement-based findings can be used by a designer to improve their layout quality.

Summary. From our measurements/simulations, we conclude that:

- For small devices (6-bit, 8-bit DACs, OTAs), random variations dominate, and NonCC and CC layouts have similar mismatch while NonCC reduces layout area.
- For larger devices (10-bit DACs and higher), the distance-dependent component cannot be ignored even for CC patterns and must be considered during design.

These criteria can be used to feed methods for automated layout that determine the optimal use of CC or non-CC (e.g., interdigitated or clustered) layouts [8], [9].

TABLE I
COMPARING NONCC AND CC UNARY CURRENT-STEERING DACS

	6-bit DAC		8-bit DAC		10-bit DAC	
	NonCC	CC	NonCC	CC	NonCC	CC
$\sigma^2(\Delta I/I)_{spec}(10^{-4})$	8.87		2.22		0.55	
$\sigma^2(\Delta I/I)_D(10^{-4})$	0.4	0	0.6	0.2	0.9	0.3
$\sigma^2(\Delta I/I)_R(10^{-4})$	8.47	8.87	1.62	2.02	-	0.25
Device $W \times L$ (μm^2)	0.162	0.155	0.849	0.680	-	5.50
Total layout area (μm^2)	230	306	2958	3418	-	80800

III. DESIGN OF A MIMO RECEIVER

Next, we discuss a design testcase that illustrates the application of design automation methodologies to build a complex circuit: success on such testcases is a core prerequisite to the widespread adoption of analog EDA solutions. Automation is particularly useful for RF/analog/mixed-signal (RF-AMS) circuits, which require careful design to avoid expensive respins and failures in the field. To build robust, high-performance RF-AMS blocks, it is desirable to evaluate a number of design options.

A typical RF-AMS design flow involves: (a) architecture design, (b) sub-block design, (c) device sizing, (d) layout, typically performed manually, (e) final verification with post-layout parasitics. In particular, for RF circuits the layout step is a critical determinant of circuit performance. During design iterations, devices may be resized in step (c) based on post-layout parasitics, but such sizing operations perturb the layout, leading to further changes in the parasitics, leading to long iterations between steps (c)–(e) till the design specifications are met. The layout step (d) is a tedious manual process, requiring expert human layout/mask designers. Advanced process technologies involve complex design rule checks (DRCs), which further slow the layout process. Typical design/layout iterations run into multiple weeks, limiting the number of designs that can be evaluated before tape-out. To reduce the design/layout iterations, designers resort to conservative parasitic estimates, resulting in designs with sub-optimal power and/or performance [25].

We discuss a use case where we compare the efficacy of automatic layout generation using ALIGN versus a manual layout for a state-of-the-art MIMO receiver [27]. Multiple automated layouts are generated simultaneously, all of which satisfy the required layout constraints such as symmetry, ordering, common-centroid and matching. With

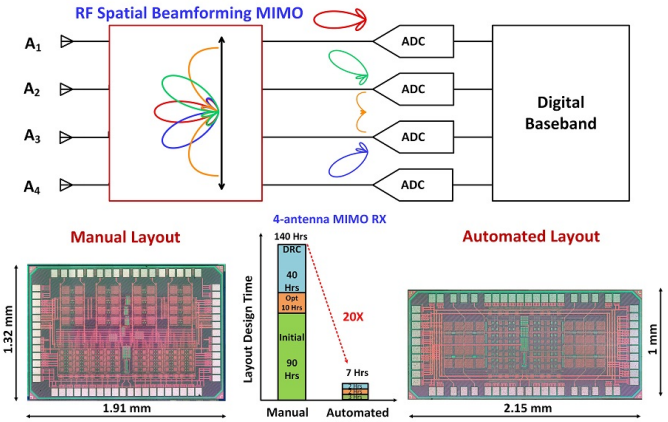


Fig. 8. MIMO design used to illustrate the process; chip micrographs for manual and automatic layouts; productivity improvement from automation.

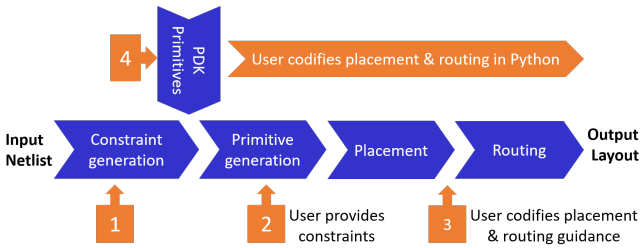


Fig. 9. Overview of the ALIGN flow.

rapid layout synthesis, ALIGN rapidly estimates parasitics during the design phase, reducing the number of design/layout iterations. The designer's intent for a floorplan can be specified in ALIGN in the form of user-defined constraints provided to the layout generator [6]. The layout is generated hierarchically and the designer can pick the best-performing layout for each hierarchy using post-layout extracted simulations. Fig. 8 shows the manual and automated layouts (microphotos) of the MIMO RX and shows a measure of how design productivity is enhanced by the latter. The manual approach took weeks for a single layout, as against a few hours for the automated process to generate multiple complete chip-level MIMO layouts.

Design Flow An overview of the steps involved in the ALIGN flow is shown in Fig. 9. At the very basic level, the input can be a netlist and the output is a hierarchical layout in GDSII format. There are four major steps which are briefly described: (a) The first step of constraint generation identifies known sub-circuits in the netlist and layout constraints such as symmetry, common-centroid, ordering, and matching. ALIGN uses graph convolutional network (GCN) to identify hierarchies like OTA, LNA, etc. The designer may examine these identified hierarchies and constraints and augment them to reflect designer intent. Primitives are one or more devices that are typically laid out as a single layout entity such as resistors, capacitors, current mirrors, and differential pairs. (b) The second step generates layouts for each of the primitives in the first step. (c) The third step assembles these primitive layouts into a legal layout that meets layout constraints. (d) The last step is routing which connects various nets with wires of appropriate widths. This step also generates power grids for the supply/ground nets and connects them to the devices. Apart from the auto-generated constraints in the first step, users can input placement constraints such as maximum width/height, aspect ratio, and spacing between any pair of blocks, and routing constraints such as shielding for critical nets, clock nets, and matched routing for symmetric nets. As shown in Fig. 9, the user can intervene in the ALIGN flow at multiple points and add/delete constraints within the flow. There is also support to code the entire placement and routing using relative positions of blocks. To ensure that the layouts generated are compatible with foundry-specified PDKs, an abstract set of rules are honoured by all the layout generators. These rules are chosen to be broad enough to work for all tested foundries with minimal changes to the flow. The arithmetic values for the layout rules change for different foundries and technologies.

Black-box methodology. To enable designers to reuse layouts of sub-circuits with verified performance, ALIGN supports the inclusion of such layouts through a black box methodology. In this methodology, the user-input layouts are abstracted into the library exchange format (LEF) with defined pins, ports and obstacles. The abstraction step is automated for the input layouts in GDSII format. These layouts are instantiated in the placement step and appropriate connections are made during routing.

Engineering change order (ECO). Design/layout iterations are performed to subsume the impact of layout parasitics. In each iteration, the layout is perturbed due to one of the following: alteration of device sizes, the spacing between devices, or inclusion of new placement/routing constraints. Depending on the hierarchy at which such a change is made, the impact on the layout could be localized or span the entire design. ALIGN handles such a change using an ECO methodology. As an example, we may add space between blocks to reduce coupling, which could perturb the corresponding hierarchy, its parents and neighbours. ALIGN automatically identifies such a perturbation and rapidly performs incremental placement and routing on those blocks.

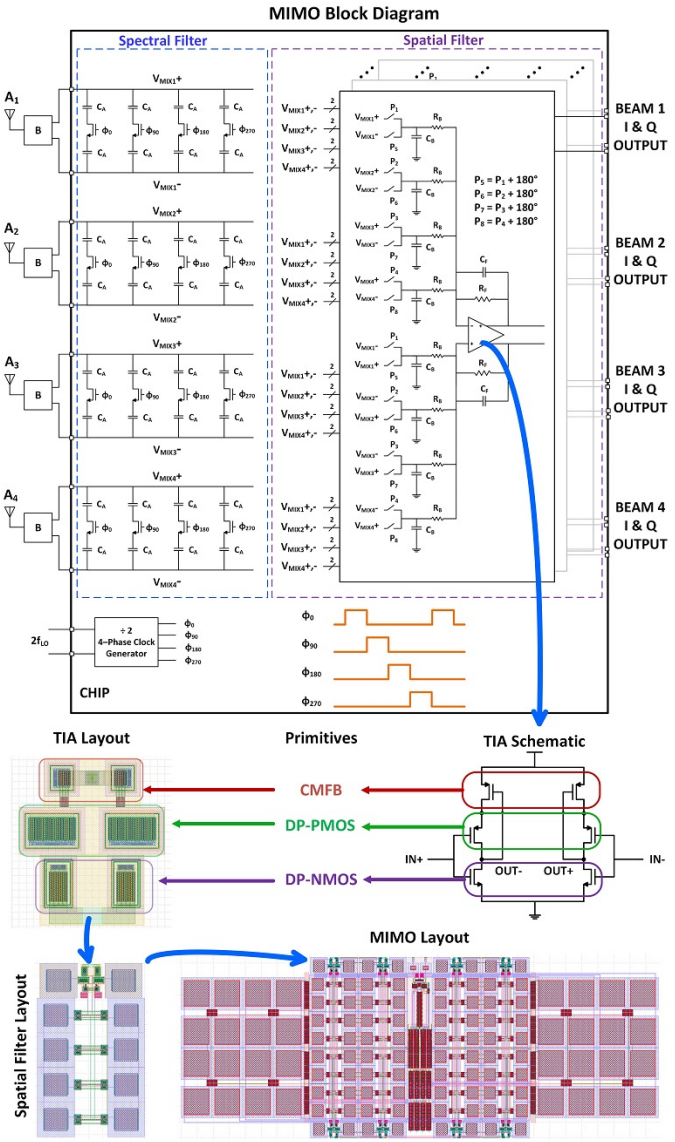
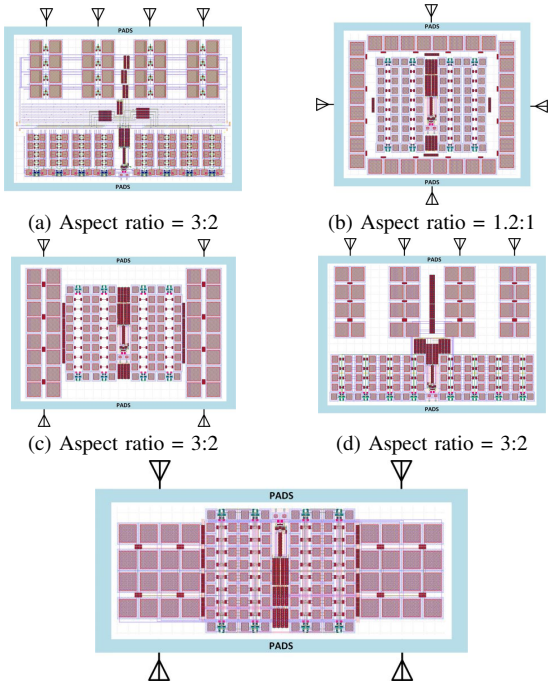


Fig. 10. Schematic of the MIMO receiver, with auto-annotation of subblocks, and hierarchical layout generation of the spatial filter.

MIMO floorplans. Fig. 10 shows the MIMO architecture with four spectral filters, eight spatial filters and the clock generation block. The spectral filter consists of a differential bottom-plate mixer architecture for improved IIP_3. The spatial filter consists of a differential summing amplifier with capacitor C_B acting as the voltage source. Spatial beamforming is performed by combining different antenna inputs with phase shifts. Fig. 10 shows some blocks recognized by



(a) Aspect ratio = 3:2 (b) Aspect ratio = 1.2:1

(c) Aspect ratio = 3:2 (d) Aspect ratio = 3:2

(e) Aspect ratio = 2.6:1 (final floorplan selected)

Fig. 11. Various MIMO layouts (a) Manual layout, (b) and (c) automatically generated layouts, (d) ALIGN mimicking manual layout through constraints, and (e) ALIGN layout with user-specified maximum height constraint.

ALIGN: the transimpedance amplifier (TIA) with common primitives such as common mode feedback (CMFB) transistor pairs, differential NMOS and PMOS pairs. After identifying these primitives associated with the amplifier, ALIGN automatically creates a symmetrical layout based on the device sizes. Internal routing widths can be user-defined, based on performance needs.

Fig. 11 compares various MIMO layouts generated using ALIGN with the aforementioned placement and routing constraints against a manual layout. For a fair comparison between manual and ALIGN-generated layouts, the layouts of primitive cells such as MIM capacitors and special RF transistors used in the manual layout were reused in ALIGN layouts using the black-box methodology. Fig. 11(a) shows the manual layout and Fig. 11(b) and (c) show two ALIGN-generated layouts with just the clock net constraint. The layout in Fig. 11(b) is the most compact of all variants, and its square aspect ratio of the layout makes it easy to match routing parasitics using an H-tree.

In each iteration, simulations with post-layout extracted parasitics were used to identify the performance-critical nets and blocks. The following changes were made in successive iterations based on the simulations: (a) improving the resistance of critical nets by widening wires using the net-specific routing width constraint, (b) reducing coupling by (i) increasing the spacing between blocks, and (ii) adding shielding between adjacent signal nets. These changes involved perturbation to both placement and routing and were implemented automatically using the ECO mode described in Section III. The entire placement and routing in ECO mode took only tens of minutes in each iteration. Fig. 11(d) shows the layout generated by ALIGN mimicking the manual layout. This ALIGN layout was achieved by manually specifying constraints for all the hierarchies. An external limitation on the die size constrained the maximum height of the MIMO layout to be 600μm which when input to ALIGN generated the layout in Fig. 11(e). This layout was selected for the tapeout.

Measurement A prototype of a four antenna MIMO system was implemented in the TSMC 65nm CMOS process. The die photo is shown in Fig. 12. The dies were wire bonded to a 60-pin QFN and then mounted on a two-layer PCB. Four BALUNs were placed on the PCB to create differential RF signals.

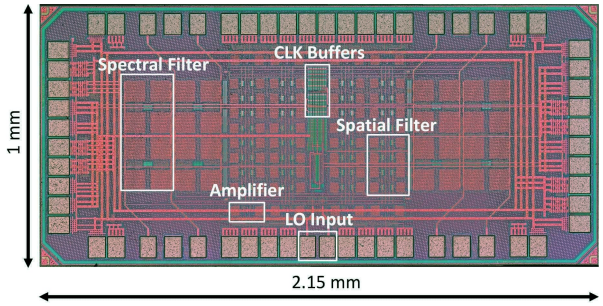


Fig. 12. Die photo of the ALIGN-generated MIMO receiver layout.

TABLE II
MEASURED PERFORMANCE: MANUAL VS. ALIGN LAYOUT.

	ISSCC'21[42]	This work
Technology	65nm CMOS	65nm CMOS
Layout type	Manual	Automated
Operating frequency range (GHz)	1 – 3	1 – 2.3
Single element conver. gain (dB)	15	13
Max spatial suppression (dB)	27	28.4
Noise figure (dB)	10 – 12	10.8 – 12.3
In-band/In-beam OIP_3 (dBm)	18.1	14.9
Out-of-band/In-beam IIP_3 (dBm)	10.8 $\Delta f/BW$=4.6	19.3 $\Delta f/BW$=4.6
In-band/In-notch IIP_3 (dBm)	19.3	20.3
Out-of-band/In-notch IIP_3 (dBm)	21.35 $\Delta f/BW$=2	23.87 $\Delta f/BW$=2
In-band/In-beam B_{1dB} (dBm)	–10.67	–11.8
In-band/In-notch B_{1dB} (dBm)	0.26	–0.04
Area (sq.mm)	2.52	2.15
Power (mW)	130–242	130–175

Performance Comparison: A comparison of the measured performance for the manual and automated layouts is shown in Table II. The performance parameters of the manual layout [27] has been included with a loss calibration of 5dB. As can be seen, spatial suppression, IIP_3 and B_{1dB} of the automated layout are close to/exceeds the performance of the manual layout, except for the RF frequency range. We believe this is because all the clock buffers were placed in the center for the automated placement, resulting in an operating frequency of 1-2.3GHz as opposed to manual layout's 1-3GHz range. This parameter can be improved with few additional iterations in ALIGN.

Productivity Improvement. Fig. 13 compares the time required to generate the layout of a MIMO receiver using manual and automated approaches. FP 1 and FP 2 correspond to the floorplans shown in Fig. 11(b) and (e) respectively. FP 2.1 refers to the default layout generated by ALIGN with the maximum height constraint of 600μm. Post-layout extracted simulations on this layout identified critical nets whose resistance needed to be improved. Resistance parasitics were improved using net-specific routing width constraints and ECO mode described in Section III was used to realize these constraints. Simulations based on this layout identified nets whose coupling capacitance needed improvement. Using this feedback, shielding and

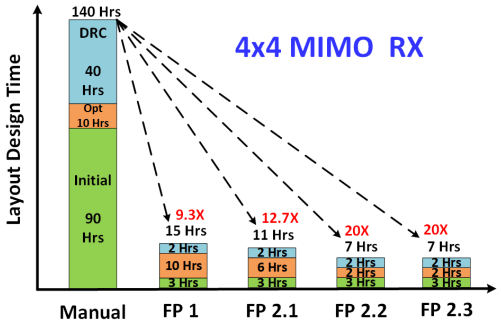


Fig. 13. Productivity: manual vs. automated layout (4 iterations).

increased spacing constraints were added and a second iteration of ECO was used to arrive at the final layout. As seen in Fig. 13, each of the iterations took hours to generate the layout and cleanup DRCs against the manual approach that took days for a single layout. The resultant automated layout has a similar performance to the manual layout. The productivity gain chart shows that within the same amount of time spent in generating a single manual layout, multiple automated layouts can be explored. As demonstrated, each layout can also be iteratively improved in a short time using performance evaluated with post-layout simulations.

IV. Conclusion

Sustaining recent advances in analog layout requires improved engagement with designers, incorporating designer concerns. This paper has presented progress through (a) mismatch characterization, which can be used to build higher-performance layouts with lower variation, and (b) productivity improvements that are demonstrated by the application of design tools to a MIMO receiver layout of significant complexity.

Acknowledgments. This work was supported in part by DARPA IDEA under SPAWAR Contract N660011824048; by the SRC, and by the National Science Foundation under Award 2212345.

References

[1] E. Chang, *et al.*, "BAG2: A process-portable framework for generator-based AMS circuit design," in *Proc. CICC*, pp. 1–8, 2018.
[2] B. Xu, *et al.*, "Magical: Toward fully automated analog ic layout leveraging human and machine intelligence," in *Proc. ICCAD*, pp. 1–8, 2019.
[3] H. Chen, *et al.*, "MAGICAL: An open-source fully automated analog IC layout system from netlist to GDSII," *IEEE Des. Test*, vol. 38, no. 2, pp. 19–26, 2021.
[4] K. Kunal, *et al.*, "ALIGN: Open-source analog layout automation from the ground up," in *Proc. DAC*, pp. 1–4, 2019.
[5] T. Dhar, *et al.*, "ALIGN: A system for automating analog layout," *IEEE Des. Test*, vol. 38, pp. 8–18, Apr. 2021.
[6] "ALIGN: Analog layout, intelligently generated from netlists." https://github.com/ALIGN-analoglayout/ALIGN-public.
[7] M. Madhusudan, *et al.*, "Analog layout generation using optimized primitives," in *Proc. DATE*, pp. 1234–1239, 2021.
[8] A. K. Sharma, *et al.*, "Common-centroid layouts for analog circuits: Advantages and limitations," in *Proc. DATE*, pp. 1224–1229, 2021.
[9] A. K. Sharma, *et al.*, "Performance-aware common-centroid placement and routing of transistor arrays in analog circuits," in *Proc. ICCAD*, pp. 1–9, 2021.
[10] N. Karmokar, *et al.*, "Common-centroid layout for active and passive devices: A review and the road ahead," in *Proc. ASP-DAC*, 2022.
[11] N. Karmokar, *et al.*, "Constructive common-centroid placement and routing for binary-weighted capacitor arrays," in *Proc. DATE*, pp. 166–171, 2022.
[12] N. Karmokar, *et al.*, "Constructive placement and routing for common-centroid capacitor arrays in binary-weighted and split DACs," *IEEE T. Comput. Aid. D.*, vol. 42, pp. 2782–2795, Sept. 2023.
[13] N. Karmokar, *et al.*, "Minimum unit capacitance calculation for binary-weighted capacitor arrays," in *Proc. DATE*, 2023.
[14] Y. Lin, *et al.*, "Are analytical techniques worthwhile for analog IC placement?," in *Proc. DATE*, 2022.
[15] Ramprasath S., *et al.*, "Analog/mixed-signal layout optimization using optimal well taps," in *Proc. ISPD*, pp. 159–166, 2022.
[16] Ramprasath S., *et al.*, "A generalized methodology for well island generation and well-tap insertion in analog/mixed-signal layouts," *ACM T. Des. Automat. El.*, vol. 28, pp. 69:1–69:25, Sept. 2023.
[17] K. Kunal, *et al.*, "GANA: Graph convolutional network based automated netlist annotation for analog circuits," in *Proc. DATE*, 2020.
[18] K. Kunal, *et al.*, "A general approach for identifying hierarchical symmetry constraints for analog circuit layout," in *Proc. ICCAD*, 2020.
[19] K. Kunal, *et al.*, "GNN-based hierarchical annotation for analog circuits," *IEEE T. Comput. Aid. D.*, vol. 42, pp. 2801–2814, Sept. 2023.
[20] Y. Li, *et al.*, "Exploring a machine learning approach to performance driven analog IC placement," in *Proc. ISVLSI*, 2020.
[21] Y. Li, *et al.*, "A customized graph neural network model for guiding analog IC placement," in *Proc. ICCAD*, 2020.
[22] T. Dhar, *et al.*, "A charge flow formulation for guiding analog/mixed-signal placement," in *Proc. DATE*, 2022.
[23] Y. Li, *et al.*, "A circuit attention network-based actor-critic learning approach to robust analog transistor sizing," in *Proceedings of Workshop on Machine Learning for CAD*, pp. 1–6, 2021.
[24] Y. Li, *et al.*, "Performance-driven wire sizing for analog integrated circuits," *ACM T. Des. Automat. El.*, vol. 28, Dec. 2022.
[25] J. Liu, *et al.*, "From specification to silicon: Towards analog/mixed-signal design automation using surrogate NN models with transfer learning," in *Proc. ICCAD*, 2021.
[26] X. Liu, *et al.*, "A digital LDO in 22nm CMOS with a 4b self-triggered binary search windowed flash ADC featuring automatic analog layout generator framework," in *Proc. A-SSCC*, pp. 2–4, 2022.
[27] J. Poojary and R. Harjani, "A 1-to-3GHz co-channel blocker resistant, spatially and spectrally passive MIMO receiver in 65nm CMOS with +6dBm in-band/in-notch B1dB," in *Proc. ISSCC*, vol. 64, pp. 96–98, 2021.
[28] S. Kamineni, *et al.*, "AuxcellGen: A framework for autonomous generation of analog and memory unit cells," in *Proc. DATE*, 2023.
[29] M. Madhusudan, *et al.*, "Understanding distance-dependent variations for analog circuits in a FinFET technology," in *Proc. ESSDERC*, pp. 69–72, 2023.
[30] J. Poojary, *et al.*, "Exploration of design / layout tradeoffs for RF circuits using ALIGN," in *Proc. RFIC*, pp. 57–60, 2023.
[31] K. Lakshmikumar, *et al.*, "Characterisation and modeling of mismatch in MOS transistors for precision analog design," *IEEE J. Solid-St. Circ.*, vol. 21, no. 6, pp. 1057–1066, 1986.
[32] M. Pelgrom, *et al.*, "Matching properties of MOS transistors," *IEEE J. Solid-St. Circ.*, vol. 24, no. 5, pp. 1433–1439, 1989.
[33] P. R. Kinget, "Device mismatch and tradeoffs in the design of analog circuits," *IEEE J. Solid-St. Circ.*, vol. 40, no. 6, pp. 1212–1224, 2005.
[34] U. Schaper, *et al.*, "Precise characterization of long-distance mismatch of CMOS devices," *IEEE T. Semicond. M.*, vol. 14, pp. 311–317, 2001.
[35] H. Elzinga, "On the impact of spatial parametric variations on MOS transistor mismatch," in *Proc. ICMTS*, pp. 173–177, 1996.
[36] A. van den Bosch, *et al.*, "A 10-bit 1-GSample/s Nyquist current-steering CMOS D/A converter," *IEEE J. Solid-St. Circ.*, vol. 36, no. 3, pp. 315–324, 2001.
[37] P. M. Ferreira, *et al.*, "Surface versus performance trade-offs: A review of layout techniques," *J. Int. Circuits Syst.*, vol. 1, pp. 1–16, 2022.
[38] A. L. S. Loke, *et al.*, "Analog/mixed-signal design challenges in 7-nm CMOS and beyond," in *Proc. CICC*, pp. 1–8, 2018.
[39] M. J. M. Pelgrom, *Analog to Digital Conversion*. Cham, Switzerland: Springer, 4th ed., 2022.
[40] P. Friedberg, *et al.*, "Modeling within-die spatial correlation effects for process-design co-optimization," in *Proc. ISQED*, pp. 516–521, 2005.
[41] J. Xiong, *et al.*, "Robust extraction of spatial correlation," *IEEE T. Comput. Aid. D.*, vol. 26, no. 4, pp. 619–631, 2007.
[42] J. Poojary and R. Harjani, "A 1-to-3GHz Co-Channel Blocker Resistant, Spatially and Spectrally Passive MIMO Receiver in 65nm CMOS with +6dBm In-Band/In-Notch B1dB," in *Proc. ISSCC*, pp. 96–98, 2021.