MAGICAL 1.0: An Open-Source Fully-Automated AMS Layout Synthesis Framework Verified With a 40-nm 1GS/s ΔΣ ADC

Hao Chen*, Mingjie Liu*, Xiyuan Tang*, Keren Zhu*, Abhishek Mukherjee, Nan Sun, and David Z. Pan

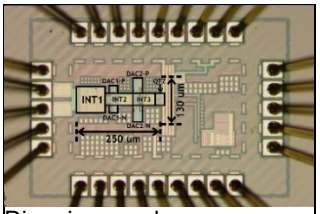
*Equal contribution; The University of Texas at Austin, Austin, TX

The layout process is incredibly tedious as it is being exacerbated by technology scaling, where design rules become increasingly complicated in advanced nodes. While digital blocks' layout can be easily synthesized, analog/mixed-signal (AMS) layout designs are still heavily manual, setting bottlenecks for time-to-market. Though endeavored in research for decades, the automation of AMS circuit layouts has not been as successful as its digital counterpart. The reason is rooted in the sensitiveness and complexity of AMS circuits layouts [1]. Researchers have proposed specialized digital-like AMS circuit architectures [2], [3], which are robust against layout mismatches. Those circuit architectures can be synthesized using commercial digital place-and-route (P&R) tools but fail to produce a general-purpose solution to other circuit architectures. On the other hand, procedure-based methodologies have been proposed to synthesize AMS circuits based on parameterized layout templates [4]. Procedure-based layout generators provide a standardized flow to migrate circuit layouts to different circuit sizings and manufacturing technologies. Still, they require a significant amount of manual effort to program the layout templates. While there are attempts to apply P&R algorithms on AMS layouts [1], to the best of the authors' knowledge, no prior work has demonstrated a silicon-proven fully automated layout of a real-world mixed-signal system.

This paper presents a generic AMS layout synthesis flow, as shown in Fig. 1, where the layout is automatically generated in a hierarchical approach similar to manual practice. Tackling AMS circuit sensitiveness while minimizing manual efforts, automatic symmetry constraint generation is performed at the building block level. In-loop simulation and Bayesian optimization are integrated into the flow to guide layout refinement automatically. The signal flow can be provided as an optional input to facilitate circuit placement at the top level. The proposed flow minimizes manual efforts by taking only circuit netlists as necessary inputs while providing the option to specify manual guidance for design preference.

This synthesis flow is process-portable and has been verified through a variety of circuit types, covering block-level components, such as comparators and OTAs, and complete AMS systems, including data converters and power management units. Due to the page limit, a 1GS/s 3rd-order ΔΣ ADC prototype with the fully-synthesized layout is demonstrated in the paper (top in Fig. 4). It includes hybrid passive-active loop filters, a comparator, and FIR DACs. Running at 1GS/s with 3rd-order noise-shaping capability, the design requires a careful layout with symmetric placement and routing, minimum signal coupling, and minimum IR drop. The measured performance is comparable to the manual layout design [5], while it only takes around 90 seconds for layout synthesis, presenting many orders of improvement in the design efficiency.

The proposed hierarchical flow consists of building block-level layout synthesis and top-level integration. The building blocks, such as Gm1, are synthesized first, and the top-level system is then integrated. The device layouts are generated based on foundry templates and then placed and routed by an optimization-based P&R engine. The symmetric constraints are automatically extracted by graph pattern matching and are used to guide the P&R engine for the block-level circuits. The block-level circuit performance is further ensured through in-loop simulation, as shown in Fig. 2. The flow performs parasitic extraction and post-layout simulations on synthesized block-level layouts. The Bayesian optimization process is utilized to guide the P&R engine to refine the layouts if the design specifications are not satisfied. For the top-level layout integration, the proposed flow automatically determines the building blocks' positions and completes the interconnections. It also can take extra inputs specifying system signal flows, serving as an interface for users to adjust the circuit placement. This proposed methodology combines both machine intelligence and human experience for high-quality layouts with low effort.



Die micrograph.

The placement process is shown in Fig. 2. The blocks are first randomly placed at the center. Then global placement generates the rough block positions where it formulates the cost function, including wire length, area, and overlapping as a non-linear programming (NLP) problem. The cost function is numerically reduced with gradient descent-based optimization techniques. The NLP process is iteratively updated until it meets the non-overlapping and symmetry target. It adopts a self-adaptive NLP problem updating scheme, which automatically adjusts the numerical optimization process towards different circuits' scales. The improved numerical behavior benefits the global placement and reduces the average routed wire length by 23% and area by 34% over various block-level circuits. The detailed placement step then refines the solution and enforces symmetry constraints using linear programming. The bottom of Fig. 2 shows an example of the placement process on an OTA.

The AMS router optimizes the interconnections while handling intricate design rules. The proposed AMS routing framework (top-left in Fig. 3) consists of three phases: 1) pre-processing, which detects the unspecified symmetry constraints and models the geometric metal shapes into sets of access points (top-middle in Fig. 3) to reduce computational complexity; 2) iterative routing, which connects all nets through a rip-up and reroute process (bottom in Fig. 3) while satisfying the specified constraints, and maximize the overall symmetry degree of the routed wires; 3) post-refinement, which corrects the leftover design rule violations.

Various design aspects such as IR drop, symmetric routing, and signal coupling, should also be considered during routing. These design requirements are formulated into constraints and implicitly optimized by the router. For instance, to handle the IR drop issue, the routing width adapts automatically to routing distance. This paper proposes to consider cross symmetry and partial symmetry in routing. When perfect mirror symmetry is infeasible, instead of waiving the constraint, the router uses cross symmetry and partial symmetry (middle in Fig. 3) to maximize the degree of symmetry. This router also supports bi-directional routing, where both horizontal and vertical wires are allowed in each metal layer. With the expanded solution space, the router can realize more optimized wire length and significantly reduce routing iterations. Overall, the proposed router achieves 13% wire length reduction, 2.5× symmetry improvement, and 24× runtime speedup compared to conventional AMS routers.

The prototype ADC is fabricated in 40nm CMOS, occupying an area of 0.033mm^2, which is very close to the manual design (0.034mm^2 in [5]). Under 1.2V supply, it consumes 0.77mW when sampling at 1GHz, in which analog circuits consume 0.63mW, and 0.14mW is consumed by digital logic. The measured SNDR and SFDR are 67.4dB and 80.8dB, respectively (bottom in Fig. 4). In order to attest the robustness of the synthesized layout, 8 chips are measured, showing SFDR and SNDR variations are within 4dB and 2dB, respectively. The measured SNR/SNDR vs. input amplitude shows a DR of 68.5dB. Fig.6 summarizes this work's performance and compares it with prior publications, including the same design with a manual optimized layout [5]. The measured SNDR difference is within 2dB compared to the manual design, which is also subject to the process variation since they are two separate tape-outs. The measurement results demonstrate the proposed layout synthesis framework's capability, which produces an AMS layout with state-of-the-art performance automatically within 90 seconds. Moreover, in contrast to prior synthesis approaches, this framework and design methodology are applicable to various circuit architectures.

References:

[1] J. Scheible *et al.*, ISPD, 2015.
[2] S. Li *et al.*, CICC, 2019.
[3] T. Ajayi *et al.*, Hot Chips, 2020.
[4] E. Chang *et al.*, CICC, 2018.
[5] A. Mukherjee *et al.*, SSCL, 2020.
[6] C.-H. Weng *et al.*, JSSC, 2016.

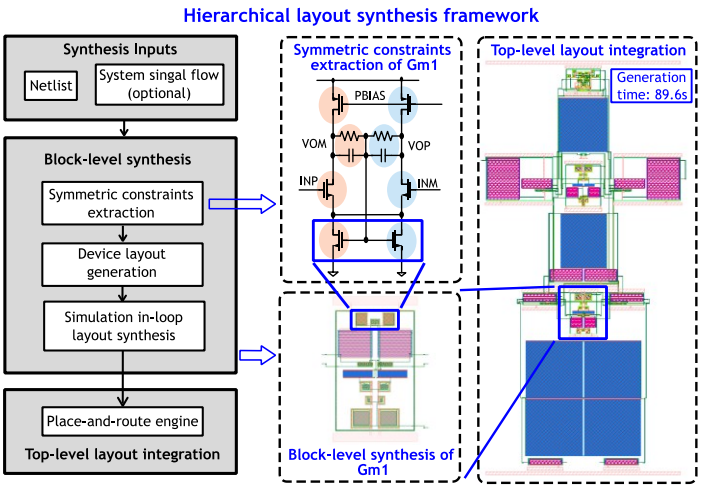


Fig. 1. The proposed hierarchical layout synthesis framework.

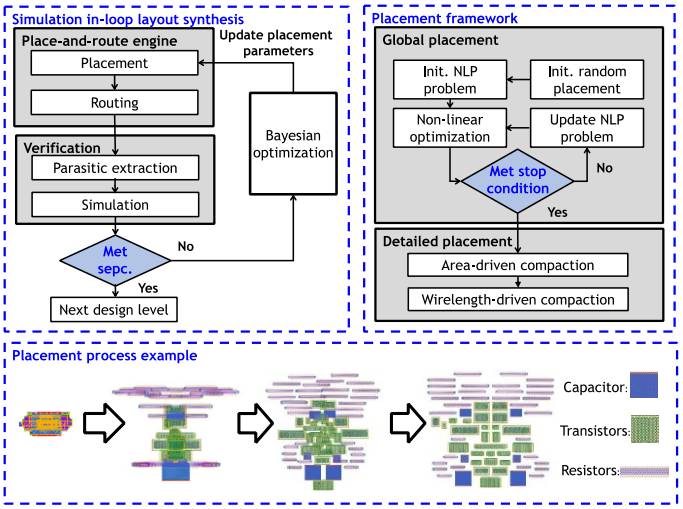


Fig. 2. The proposed simulation in-loop synthesis flow (top-left) and the proposed placement framework.

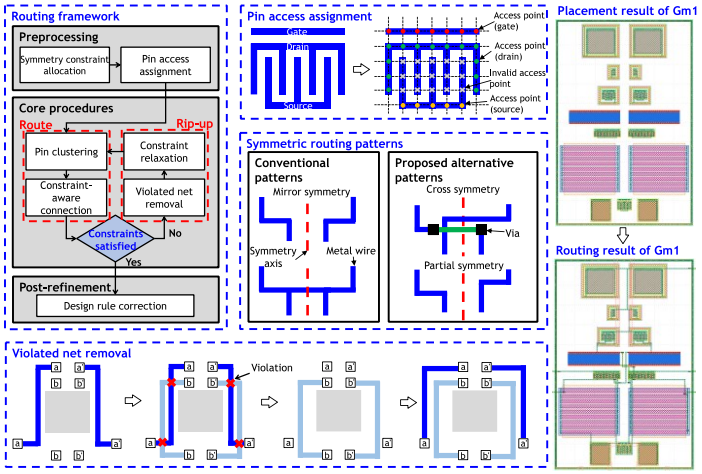


Fig. 3. The proposed routing framework.

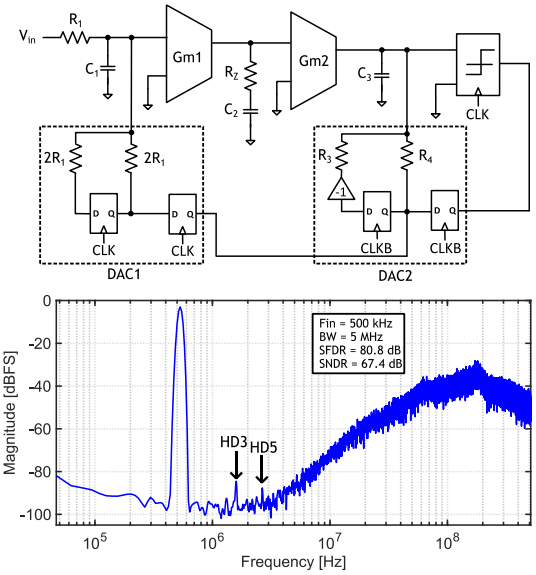


Fig. 4. The simplified schematic of the ΔΣ ADC (top) and measured output spectrum (bottom).

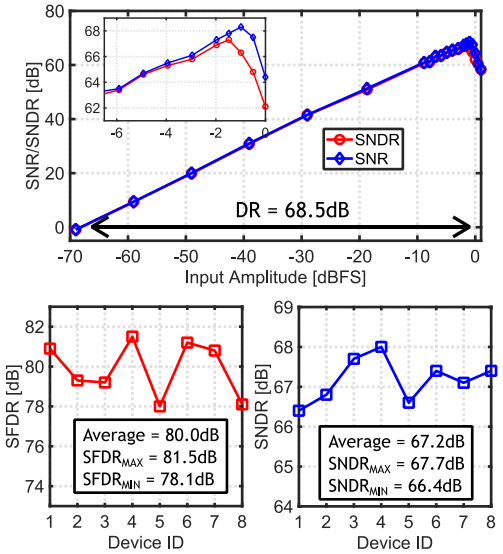


Fig. 5. Measured dynamic range (top) and SFDR/SNDR of 8 devices (bottom).

	JSSC-16 Weng [6]	SSCL-20 Mukherjee[1] [5]	CICC-19 Li [2]	**This work[1]**
Architecture	CTΔΣM	CTΔΣM	VCO-CTΔΣM	**CTΔΣM**
Layout synthesized	✗	✗	✓	✓
Universal synthesis framework	N/A	N/A	✗	✓
Hierarchical flow	N/A	N/A	✗	✓
Constraint generation	N/A	N/A	✗	✓
Order	4th	3rd	1st	**3rd**
Process [nm]	28	40	40	**40**
Area [mm²]	0.1	0.034	0.01	**0.033**
Fs [MHz]	320	1024	600	**1024**
Supply [V]	1.1/1.2	1.2	1.1	**1.2**
Power [mW]	4.2	0.79	1.08	**0.77**
BW [MHz]	10	5	4	**5**
SFDR [dB]	94.2	82.6	75	**80.8**
SNDR [dB]	74.4	65.6	68.8	**67.4**
FoMw[2] [fJ/conv-step]	49.3	51	60	**40.2**
FoMs[3] [dB]	174.5	163.6	164.3	**165.5**

[1]Same schematic

[2]$FoM_W = Power/(2^{ENOB} \cdot 2 \cdot BW)$

[3]$FoM_S = SNDR + 10 \cdot \log_{10}(BW/Power)$

Fig. 6. Performance summary and comparison with state-of-the-art ΔΣ ADCs.