1- and 80-MS/s SAR ADCs in 40-nm CMOS With End-to-End Compilation

Mingjie Liu , *Member, IEEE*, Xiyuan Tang , *Member, IEEE*, Keren Zhu , *Member, IEEE*, Hao Chen , *Student Member, IEEE*, Nan Sun, *Senior Member, IEEE*, and David Z. Pan , *Fellow, IEEE*

***Abstract*—This letter presents an end-to-end successive-approximation-register (SAR) analog-to-digital converter (ADC) compiler that generates design solutions from top-level specification to GDSII layout with a short turnaround time of 5 h. Two prototype SAR ADCs operating at 1 and 80 MS/s are compiled in 40-nm CMOS. Measurement results demonstrate a wide conversion range, presenting both analog and technology-limited performances. The compiler requires minimum manual involvement and can significantly boost design efficiency for performance retargeting, technology migrations, and agile development of IP blocks.**

***Index Terms*—Agile development, analog-to-digital converter (ADC), compilation, post-layout effect, successive-approximation-register (SAR).**

I. Introduction

Successive-approximation-register (SAR) analog-to-digital converter (ADC) has gained increasing attention in the quickly growing IoT market due to its excellent energy efficiency, scaling friendliness, and vast versatility covering low-power information sensing to high-speed data communications. However, the conventional SAR ADC design heavily relies on manual efforts due to the sensitivity and complexity of AMS circuits, such as sample and hold (S&H) circuits, comparator, and DAC, which sets bottlenecks for time-to-market. Although there were attempts to shorten the SAR development cycle, prior works with automated analog layout generation still require manual work in designing layout templates [1] or layout libraries [2], such as defining placement positions and net routing topologies. Another line of work attempts to entirely remove the need for custom layout generation by replacing all analog building blocks with digital standard cells [3], which in turn severely limits design flexibility. A SAR compiler performing both device sizing and layout synthesis is presented in [4]. However, its device sizing is independent of layout generation, resulting in performance degradation due to various post-layout effects. As a result, no silicon demonstration is implemented.

In order to address such challenges, this letter presents an end-to-end SAR compiler, as shown in Fig. 1. It confers several key merits 1) by encoding design knowledge in the SAR compiler, it provides automatic spec breakdown for each building block; 2) the CDAC array redundancy allocation scheme is automatically determined by a nonlinear optimizer for maximum conversion error coverage; 3) an analog place and route (P&R) engine is developed to replace man-ual efforts in layout template designs; and 4) post-layout simulations are performed within the compilation loop to accommodate layout-dependent effects. Chip measurement results confirm the capability of parasitic-aware sizing and layout co-optimization.

Manuscript received 4 October 2022; revised 25 November 2022; accepted 10 December 2022. Date of publication 13 December 2022; date of current version 27 December 2022. This article was approved by Associate Editor Wei-Zen Chen. *(Mingjie Liu and Xiyuan Tang contributed equally to this work.)*

The authors are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: jay_liu@utexas.edu; xitang@utexas.edu).

Digital Object Identifier 10.1109/LSSC.2022.3228911

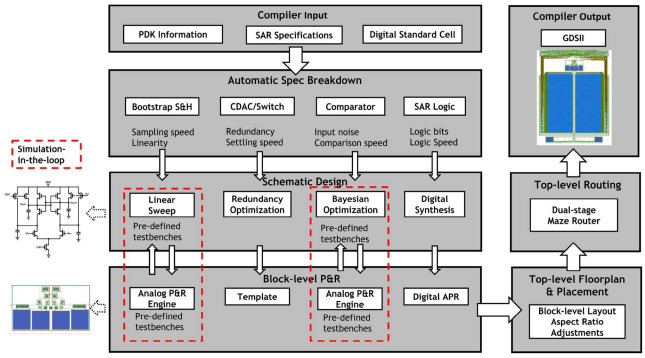


Fig. 1. Proposed end-to-end SAR compiler.

By integrating design automation techniques with circuit knowledge, two SAR ADCs operating at 1 and 80 MS/s in 40-nm CMOS are compiled, fabricated, and measured. Chip measurement results demonstrate the wide capability and flexibility of the SAR ADC compiler, which can generate designs from top-level specification to layout in an end-to-end fashion with a short turnaround time of 5 h.

To the authors' best knowledge, this work is the first to demonstrate end-to-end compilation of both schematic and layout synthesis while considering post-layout effects. The compiler takes in system-level target specifications, including the ADC resolution, conversion speed, and supply voltage. Specifications for sub-blocks are then automatically generated with margins to meet the system requirements. The performance targets on the sub-block level can be overrid-den by designers if needed. The sub-block design parameters, such as transistor sizing, are optimized automatically to ensure that the target specifications are met. Significantly different from [4], the updated optimization process in this work includes layout genera-tion and parasitic extraction within the loop, which considers layout effect and guarantees post-layout performance. Among all automated SAR ADCs, the two compiled designs realize the best input-referred noise and distortion and the highest conversion frequency, respec-tively, demonstrating superior optimization capability for both analog and technology-limited performances. The compiled 1-MS/s ADC achieves a state-of-the-art figure-of-merit (FoM) of 173.9 dB that is on par with human expert designs but with an orders-of-magnitude efficiency boost.

This letter is organized as follows. Section II describes the detailed SAR implementation, including compilation for each building block. Section III shows the measurement results. The conclusion is drawn in Section IV.

II. Proposed SAR ADC Compiler

A. Sampling Switching and Comparator

Analog circuit components, including the S&H circuit and comparator, share similar design automation strategies. The designer only

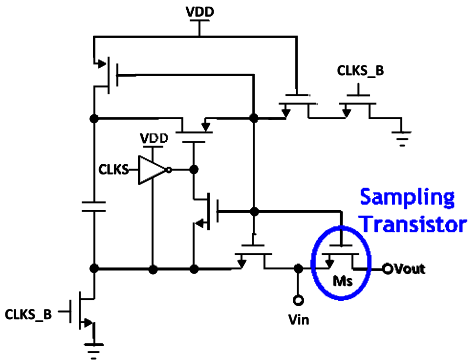


Fig. 2. Bootstrap Switch.

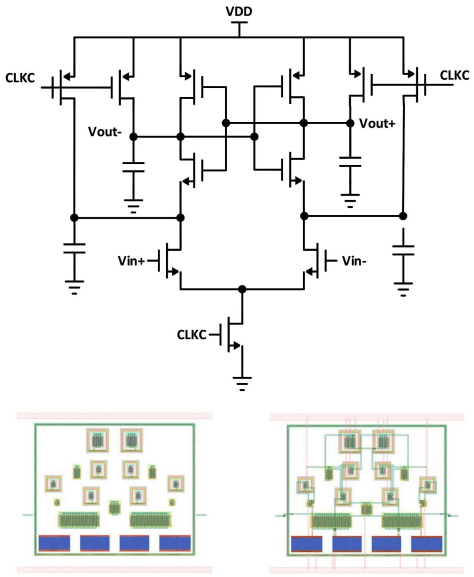


Fig. 3. Comparator schematic and with analog auto-placement and routing results.

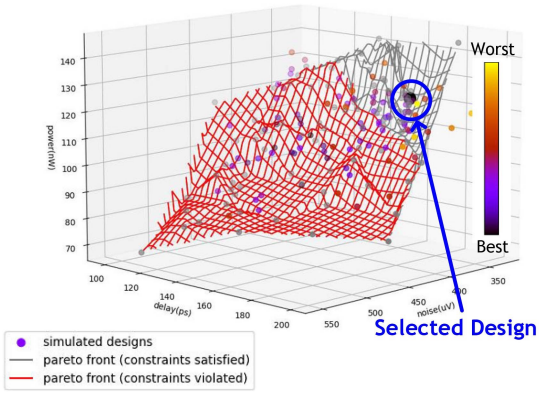


Fig. 4. Comparator optimization performance visualization.

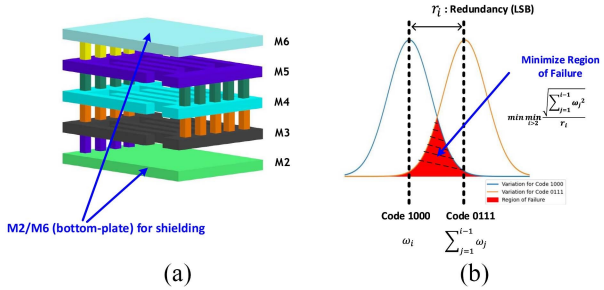


Fig. 5. (a) Unit-cap structure. (b) CDAC redundancy optimization.

needs to provide the circuit topology and simulation testbench as input, and the design parameters are searched automatically with black-box optimization techniques. Layouts of analog components are generated with an automatic analog P&R engine, similar to [5]. The placement of devices is numerically solved to minimize layout area and net half-perimeter wirelength. Matched devices are placed in mirrored symmetry determined automatically from the circuit topology. Nets are routed by a maze router while considering the symmetry of critical nets. Without additional human efforts, the obtained layouts are DRC and LVS clean. For the S&H circuit shown in Fig. 2, the sampling transistor design parameters are determined by linear sweep to satisfy ADC speed and linearity.

The design for the strong-arm latch comparator is optimized to minimize power consumption while satisfying the ADC speed and resolution requirements with Bayesian optimization. Its input-referred noise and delay requirements are calculated based on the ADC system specifications [6]: $V_{\text{noise}} < 0.25V_{\text{LSB}}$ and $t_d < 1/(2F_sN_{\text{cycle}})$, with V_{DD} the power supply voltage, N the targeted resolution, F_S the sampling rate, and N_{cycle} the number of conversions cycles required. Note that the comparator noise budget or delay requirement is only presented here as an example. The user can define their own spec as needed.

During each sizing optimization iteration, the circuit layout would be automatically generated, extracted for parasitic, and simulated without a human in the loop, as shown in Fig. 3. This significantly differs from [4], where device sizing does not consider layout effects, and manual intervention is sometimes required to alleviate layout-dependent performance degradation. Layout parasitics significantly impact the performance of analog integrated circuits, leading to discrepancies between schematic and post-layout performance and requiring several iterations to achieve design convergence. The inclusion of parasitic prediction in the optimization loop could guarantee the satisfaction of all design constraints, while schematic-only optimization fails numerous specification targets if verified with extracted layout parasitics.

The optimized comparator is shown in Fig. 4, including the Pareto front provided by Bayesian optimization. Our compilation process offers greater design flexibility and reduces the manual efforts of prior works in designing layout templates. By including automatically generated layouts in the sizing optimization, sizing and layout co-optimization reduces the manual intervention needed to achieve design closure.

B. CDAC Array

The CDAC array layout is automatically implemented with template-based layout generation for layout compactness and quality. The MOM unit capacitor uses M3–M5 for interdigitated fingers with M2 and M6 as shield plates to prevent unwanted coupling from substrate and power nets, as shown in Fig. 5. A single column of row-interleaved capacitors are placed on the side of the array for LSBs, and columns of capacitors are interleaved for the MSBs. Nonetheless, the template layout generator does impose additional restrictions on capacitor weights: the weights are even for interleave, MSB weights are multiples of the row of LSBs, etc. In this work, nonbinary weighting is implemented for error compensation [7]. If assuming a normal distribution of capacitance variation, the region of failure needs to be minimized for each bit weight, such that the redundancy is at least

	cycle index (i)	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Optimized	bit weight (w_i)	1	2	4	6	10	18	32	58	102	182	332	576	992	1792
Allocation	redundant range (r_i)	-	-	-	2	4	6	10	16	30	52	94	162	322	512
w/ Layout	bit weight (w_i)	1	2	4	4	8	16	32	64	68	136	272	544	1088	1904
Feasibility	redundant range (r_i)	-	-	-	4	4	4	4	4	64	64	64	64	64	336

Fig. 6. Bit weight results for 12-b SAR ADC.

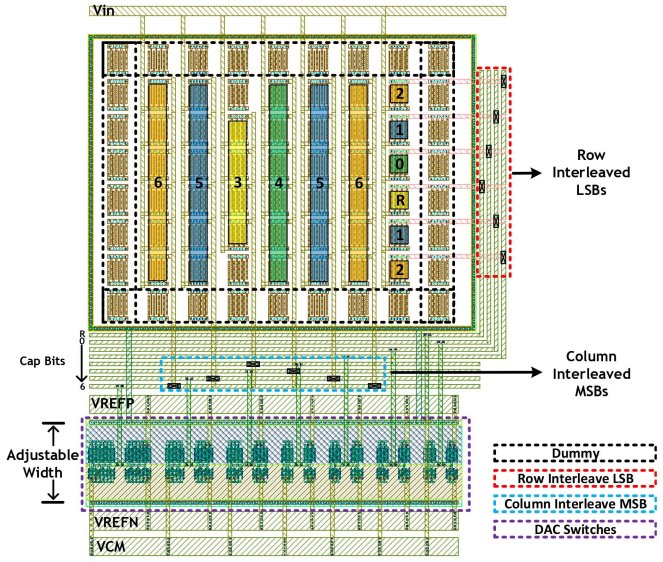


Fig. 7. Template-based CDAC layout.

positive for accurate quantization. The CDAC weight decision is formulated into a nonlinear optimization problem to maximize error coverage with constraints considering ADC resolution requirements and the feasibility of layout generation. The bit weight for the 12-bit ADC design with 14 conversion cycles is shown in Fig. 6.

A constraint example is elaborated with the 6-bit CDAC layout in Fig. 7: the 4th bit is a single column of unit capacitors, equal to the row interleaved LSBs from bits 0–2. The DAC switches are sized automatically for each bit to ensure the switches can sufficiently drive the load under ADC speed requirements. The finger widths of the transistor layout are adjusted such that the layout floorplan matches with the CDAC array. The DAC switches are placed at the bottom and routed to the CDAC array by channel routing.

C. Control Logic

Synchronous SAR logic is adopted in this compiler for its simplicity and wide usage. It is synthesized with the layout implemented by digital place and route (APR) tools. Additional buffers are automatically inserted and sized differently to ensure drive strength for DAC switches (MSBs have larger transistor gate width). The pins connecting to DAC switches are placed on the bottom boundary to ensure the design is routable at the top level.

D. Top-Level Integration

The top-level layout is integrated after all components are designed. For components layout, they are optimized for minimal area. During top-level layout integration, their aspect ratio can be further automatically adjusted to accommodate the floorplan. Since the comparator's extreme aspect ratio leads to increased parasitic, with a simulated decrease in ADC SNDR up to 3 dB, only the SAR logic and S&H layouts are adjusted. Top-level routing is performed automatically with a dual-stage maze router to ensure signal integrity. Sensitive analog nets, including clock and comparator signals, are routed first for net symmetry. In the second stage, the digital signals connecting SAR logic to CDAC switches are routed. The automatic router creates routing blockage to mitigate coupling from digital aggressors, as shown in Fig. 8. The spacing of DAC switches and component placements are

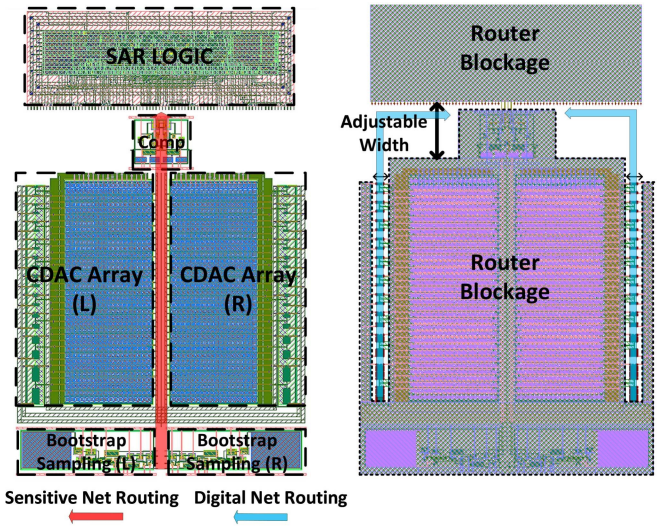


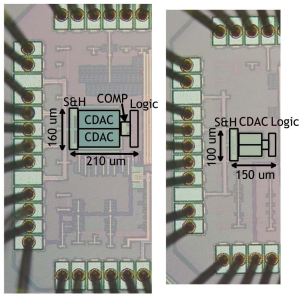
Fig. 8. Dual-stage top-level routing.

Compilation time	1MS/s	80MS/s
DAC Switch	7m52s	6m34s
CDAC Array	1m21s	18s
S&H[1]	24m36s	17m30s
Comparator[2]	3h52m	3h38m
SAR Logic	3m10s	3m12s
Top Floorplan	1m4s	26s
Top Routing	3m16s	1m40s
Total	4h26m	4h8m

[1]In-loop simulation is performed. Design time of ~14 iterations: 8% Layout, 43% PEX, 49% simulation.

[2]In-loop simulation is performed. Design time of 200 iterations: 9% Layout, 52% PEX, 39% simulation.

(a)



(b)

Fig. 9. (a) Runtime table and (b) chip micrographs of the compiled 1-MS/s and 80-MS/s SAR ADCs.

TABLE I
PERFORMANCE SUMMARY AND COMPARISON WITH STATE-OF-THE-ART AUTOMATED SAR DESIGNS

	JSSC-17 Wulff	TCAS-II-18 Seo		TVLSI-18 Ding		This work	
Auto-Design	No	Yes (Digital)		Yes		**Yes**	
Auto-Layout	Yes	Yes		Yes		**Yes**	
Process [nm]	28-SOI	180	28	40	40	**40-LP**	**40-LP**
Area [mm^2]	0.00312	0.25	0.0086	0.031	0.056	**0.015**	**0.034**
Power Supply [V]	0.69	1.8	1	1	1	**1.2**	**0.7**
Fs [MS/s]	20	0.1	50	32	1	**80**	**1**
Resolution [bit]	9	12	11	8	12	**10**	**12**
Power [uW]	15.9	31.6	399	187	16.7	**512**	**9.5**
SNDR [dB]	48.8	63.3	56.8	47.4	61.1	**54.5**	**66.7**
SFDR [dB]	63.1	70.2	69.2	57.8	68.3	**67.4**	**86.7**
ENOB	7.8	10.2	9.1	7.6	9.9	**8.76**	**10.8**
IRND [uVrms][1]	1771.5	870.5	1022.1	3016.4	623	**1598.3**	**228.9**
FoM$_S$[2] [dB]	166.8	155.3	164.8	156.7	165.8	**163.4**	**173.9**
FoM$_W$[3] [fJ/c.-step]	3.5	265.5	14.1	30.7	18.1	**14.8**	**5.4**

[1]Input-referred noise and distortion (IRND) is calculated based on input range and SNDR

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

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

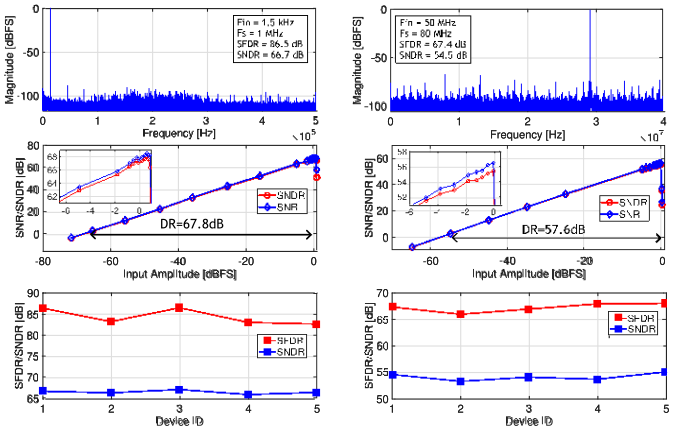


Fig. 10. Measured spectra, dynamic ranges, and performance variations across five devices of (a) 1 MS/s and (b) 80-MS/s ADCs.

also automatically adjusted to improve pin accessibility and reduce routing congestion.

III. Measurement Results

Two ADCs are compiled in 40-nm LP CMOS, as shown in Fig. 9. One-time foreground calibration is used to address capacitor mismatches [8]. The 12-b 1-MS/s chip occupies an area of 0.034 mm^2, consuming 9.5 uW under 0.7-V supply, with measured SNDR/SFDR of 66.7/86.7 dB, respectively. As shown in Fig. 10, the measured dynamic range is 67.8 dB. The 10-b 80-MS/s chip occupies an area of 0.015 mm^2, and consumes 512 uW under 1.2-V supply, with measured SNDR/SFDR of 54.5/67.4 dB, respectively. The dynamic range is 57.6 dB. Across five different devices, the SNDR variations are within 2 dB for both designs, demonstrating the robustness of the end-to-end compiled SARs.

Table I summarizes this work's performance and compares it with prior automated SAR designs. This work is the *first* to demonstrate end-to-end compilation of both SAR schematic and layout while considering post-layout effects. This compiler covers a wide conversion range, presenting both analog and technology-limited performances. The 1-MHz ADC achieves the best input-referred noise and distortion, demonstrating the compiler's outstanding capability to optimize analog performance. The 80-MHz one realizes the highest conversion frequency, limited by the speed of synthesized logic with 40-nm LP technology.

The design methodology and SAR ADC compilation framework is technology independent, where technology-dependent information is taken as additional input, such as the PDK information and digital standard cell library. Although the compiled two ADC presented in this letter are both in 40-nm technology, the framework could be further adapted to accommodate new processes quickly. Future work includes adapting the framework to other design processes, especially for advanced FinFET technology nodes. The demonstrated design methodology enables fast turn-around time by integrating recent advancements of design automation and is useful for performance retargeting, technology migrations, and future agile development of IP blocks (not limited to ADCs).

IV. Conclusion

This letter presented two SAR ADCs with an end-to-end compiler. This compiler covers a wide conversion range, presenting both analog and technology-limited performances. The demonstrated design methodology enables fast turn-around time and reduces manual labor by integrating design knowledge with recent automation advancements. It is useful for performance retargeting, technology migrations, and future agile development of IP blocks.

References

[1] C. Wulff and T. Ytterdal, "A compiled 9-bit 20-MS/s 3.5-fJ/conv.step SAR ADC in 28-nm FDSOI for Bluetooth low energy receivers," *IEEE J. Solid-State Circuits*, vol. 52, no. 7, pp. 1915–1926, Jul. 2017.

[2] M. Ding et al., "A hybrid design automation tool for SAR ADCs in IoT," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 12, pp. 2853–2862, Dec. 2018.

[3] M.-J. Seo, Y.-J. Roh, D.-J. Chang, W. Kim, Y.-D. Kim, and S.-T. Ryu, "A reusable code-based SAR ADC design with CDAC compiler and synthesizable analog building blocks," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 12, pp. 1904–1908, Dec. 2018.

[4] M. Liu, X. Tang, K. Zhu, H. Chen, N. Sun, and D. Z. Pan, "OpenSAR: An open source automated end-to-end SAR ADC compiler," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD)*, 2021, pp. 1–9.

[5] H. Chen et al., "MAGICAL 1.0: An open-source fully-automated AMS layout synthesis framework verified with a 40-nm 1GS/s 16 ADC," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, 2021, pp. 1–2.

[6] C.-P. Huang, J.-M. Lin, Y.-T. Shyu, and S.-J. Chang, "A systematic design methodology of asynchronous sar ADCS," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 5, pp. 1835–1848, May 2016.

[7] C.-C. Liu, C.-H. Kuo, and Y.-Z. Lin, "A 10 bit 320 MS/s low-cost SAR ADC for IEEE 802.11 ac applications in 20 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 50, no. 11, pp. 2645–2654, Nov. 2015.

[8] S.-W. M. Chen and R. W. Brodersen, "A 6-bit 600-MS/s 5.3-mW asynchronous ADC in 0.13-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2669–2680, Dec. 2006.