A 5mW 250MS/s 12-bit Synthesized Digital to Analog Converter

Elnaz Ansari and David D. Wentzloff
University of Michigan, Ann Arbor, MI, 48109, USA
{elnaz,wentzlof}@umich.edu

***Abstract* — Design automation of analog circuits is becoming inevitable as CMOS technology scales, mainly because the extensive amount of design rule checks cannot be easily handled by manual analog design approaches. This paper presents a low-power 12-bit, 250MS/s digital-to-analog converter (DAC) completely implemented using standard digital design flows and automatic place and route (APR). This is a current-steering DAC, and because the layout of current cells and standard digital cells are APRed together, the resulting custom design effort and time, power, and area are all minimized. Three different calibration algorithms are implemented in order to compensate for the systematic mismatch caused by APR, as well as the inter-die and intra-die variations. The DAC is fabricated in a 65nm CMOS technology, and achieves an SFDR >50dBc at up to a 100MHz input frequency while consuming only 5mW. With minimal (re-) design effort, this DAC achieves a performance that is comparable to that of conventional designs.**

***Index Terms* — Design automation, Synthesized analog circuits, DAC, Digital calibration.**

I. Introduction

Electronic design automation (EDA) plays an essential role in today's electronic systems, especially very large scale digital designs, e.g., processors with billions of transistors integrated on a chip. A key step of EDA is the abstraction and reuse of regular common blocks such as standard cells, and at a higher level, arithmetic blocks. Analog circuits, on the other hand, require complicated design techniques with many optimization variables. This process is not easily automated, and results in critical, high-performance analog blocks still being designed manually by highly-skilled engineers.

For decades, the separation of using EDA for digital circuits and doing full-custom design for analog circuits has worked. However, the number of manufacturing design rules in modern CMOS processes is growing exponentially [1]-[3], and with it the time required to produce full-custom layout of high-performance analog blocks (e.g., the analog front-end of a digital-to-analog converter) is increased. This is especially true for designing large analog circuits that use numerous building blocks, e.g., high-resolution data converters. A 1-bit resolution increase in a digital-to-analog converter (DAC) or analog-to-digital converter (ADC), roughly doubles the number of blocks used in it, leading to a rapid increase in design time and effort when done manually. This favors a cell-based EDA approach that utilizes both digital and analog cells, and is much faster at producing an optimized layout.

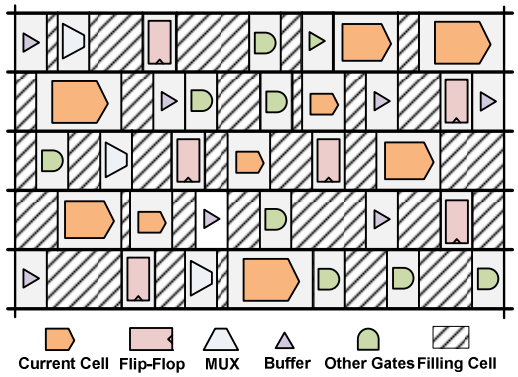

Figure 1. APRed digital and analog blocks.

Several recent designs employ digital automatic placement and routing (APR) techniques to lay out analog blocks. Designs of a synthesized all-digital phased-lock loop (ADPLL) using only standard digital cells are demonstrated in [4] and [5]. Using a similar approach, an all-digital ADC entirely from Verilog code and a standard cell library has been synthesized in [6]. In [7], a combination of analog and digital cells are automatically placed and routed via existing synthesis tools, in order to build an ADPLL.

This paper presents the first published results from a synthesized DAC that benefits from the APR of unit-sized analog cells, along with digital standard cells. In this approach, analog complexity is relaxed in favor of automation, and as a result, the (re-)design cycle (including architecture and schematic designs, layout, parasitic extraction, and simulation) is dramatically shortened. Furthermore, APR enables higher levels of integration and better scaling of analog designs, and thus benefits from Moore's law without being hindered by the

978-1-4799-3286-3/14/$31.00 ©2014 IEEE

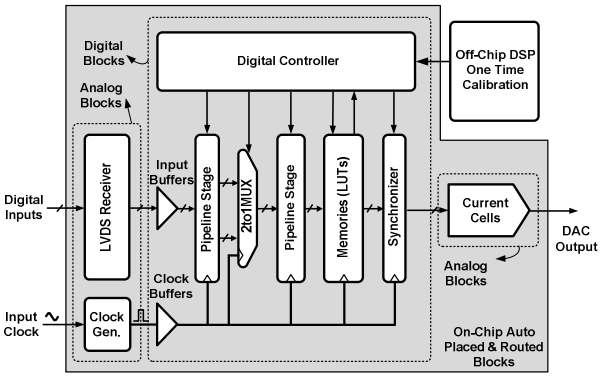

Figure 2. High level architecture of the synthesized DAC.

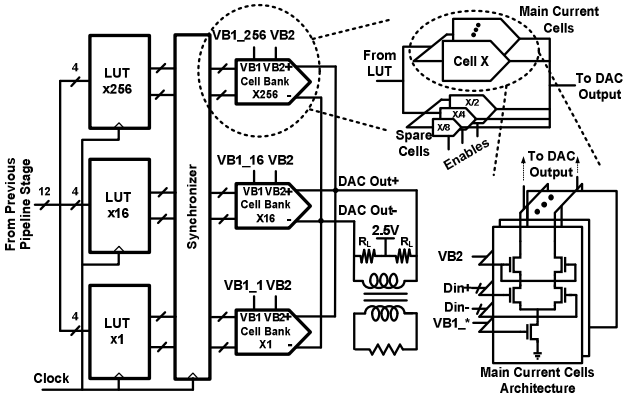

Figure 3. DAC current cells architecture.

increasingly complex design rules. The drawback of this approach is the non-idealities and the mismatches introduced by (almost) randomly laid out cells (see Figure 1), for which we rely on automated digital calibration techniques to digitally correct them.

The rest of the paper is organized as follows. Section II describes the DAC architecture. The proposed calibration techniques are explained in section III, and experimental results are discussed in section IV.

II. DAC ARCHITECTURE

Current-steering DAC architectures are the most widely used because of their speed [8]-[12]. We employ a current-steering architecture that uses many current cells that are designed on a standard cell grid so they may be automatically laid out with digital cells. Our design approach is to use digital circuitry as much as possible, and use analog blocks only when needed. This allows us to express the majority of the circuit at a behavioral level using Verilog. The analog parts consist of regular blocks that are expressed by structural Verilog to instantiate and wire many unit analog cells. As a result, our DAC design is completely synthesized and laid out via existing CAD tools.

Figure 2 shows the high-level architecture of our proposed 12-bit cell-based current-steering DAC. Double data rate (DDR), low voltage differential swing (LVDS) inputs are converted into full swing digital signals through pseudo differential LVDS receivers followed by synchronous multiplexers. Multiple pipeline stages are placed along the signal path in order to increase the operating speed of the DAC, as well as synchronize the digital signals. A digital controller manages the entire system, and performs important tasks such as reading/writing into/from on-chip look-up-tables (LUT), defining the required number of pipeline stages, and setting the control bits that select different paths for the input signals and the clock.

The main core of the DAC is comprised of many current steering cells. The cells are divided into three banks, (with three different weights), each of which contains 15 cells. The four most significant bits (MSBs) of the digital input control the bank with the largest cells, and the four least significant bits (LSBs) control the bank with the smallest cells. Assuming the current weight coefficient of the smallest bank is 1, then the weight of the middle bank and the largest bank is 16 and 256 respectively. As depicted in Figure 3, each current cell consists of only five transistors.

In addition to the main DAC cells, tri-state spare DAC cells are also placed for calibration purposes. They can be switched off, having no effect on the circuit, or they can be switched on, in which case they act as a regular current steering cell. LUTs control when these calibration cells are used on a per-sample basis. All the current cells, including the original and spare cells, are individually designed and laid out on the standard cell grid, and vary in size from that of a minimum-sized inverter to two D flip-flops. The cells are then abstracted and integrated into the EDA flow, via a structural Verilog description in which each cell is instantiated many times. The remaining parts of the circuit are expressed via behavioral Verilog and are synthesized along with the current cells.

The current cells are controlled by LUTs that can be programmed to activate different cells for each input pattern. The contents of the LUTs are changed during the calibration phase, in order to compensate for the non-linearities by activating better cell combinations and possibly utilizing spare cells. The one-time calibration process is described in the next section.

III. CALIBRATION

As mentioned earlier, automatic calibration is a key step of the DAC's design process. In addition to the non-idealities introduced by cell mismatches (due to intra-die process variations), the APRed layout also causes systematic mismatches in the interconnecting wires and further degrades the performance of the DAC. To compensate for these, we added flexibility and several degrees of freedom to the DAC, which enable us to exploit the following calibration techniques.

The programmable on-chip LUTs provide several degrees of freedom, because they allow any combination of the current cells to be activated for any given input pattern within their bank. By changing the contents of the LUTs in the calibration phase, we select a combination of the cells that provide the most linear output for a ramp input. Furthermore, the LUTs control the tri-state spare cells, which are used to fine-tune the previous calibration step. This is done by bringing in spare cells whenever the main cells cannot provide a linear output.

The bias voltage of each cell bank can also be adjusted to change their current output, and hence their relative weight. As shown in Figure 3, an adjustable bias voltage is applied to the tail transistors of the DAC cells. Finally, the last calibration resource is swapping the digital input codes at the source.

The calibration process is done once per chip, at low frequency, and is also automated using a digital signal processor (DSP). The first step in the calibration process is (fine-)tuning the current of each bank; hence their values match with each other with the appropriate weights. Next, the DSP measures the output voltages of the DAC, compares with the target values, and adjusts the LUTs accordingly to improve its performance. The calibration algorithm first measures the DNL of the DAC at its default configuration, and then re-orders the DAC cells until the DNL can no longer be improved. It then targets the input combinations for which the maximum distortion has occurred and enables spare calibration cell(s) of the appropriate size to eliminate the distortion. The algorithm continues until no progress can be achieved in distortion improvement, or until it runs out of spare calibration cells. The final calibration step is input-code re-ordering (code swapping) which is handled in the DSP that provides the digital inputs for the DAC. In this technique, the input codes will be re-ordered in such a way that the DAC generates the most linear ramp output.

Table I. Comparison of the proposed DAC with previous designs.

Specifications	This Work	[8]	[9]	[10]	[11]	[12]
Resolution (N)	**12**	12	12	10	12	14
F_s (GS/s)	**0.25**	2.9	1.6	0.3	0.5	0.2
Max F_{in} (MHz)	**150**	300	800	150	240	95
R_L (Ohms)	**50**	50	50	50	50	12.5
V_{p-p} (V)	**1.3**	2.5	0.8	6	1.5	0.5
SFDR (dBc)	**>43**	>66	>70	>44	>61	>78
Power (mW)	**5.5**	188	40	476	216	270
Area (mm^2)	**0.036**	0.32	0.02	2.25	1.13	2.4
Process (nm)	**65**	65	40	45	180	140

IV. MEASUREMENTS

The DAC is fabricated in a 65nm CMOS technology, and operates up to 250MS/s. Figure 4 shows the INL and DNL measurement results before and after the calibration. Before calibration, the DNL and the INL value ranges are [−86.1, 13.5] and [−28, 102] LSBs, respectively. After calibration, these ranges are improved to [−1, 5] and [−2, 4.4] LSBs; the LSB size is 317μV. Figure 5 shows the single tone measurement results of the DAC before and after calibration. The SFDR ranges from 65dBc to 43dBc

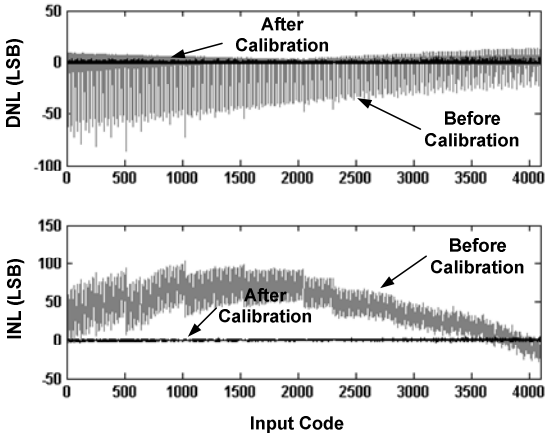

Figure 4. DNL and INL results before and after calibration.

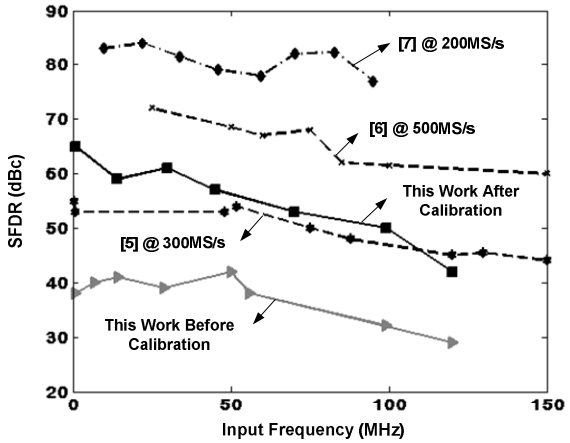

Figure 5. Measured SFDR results.

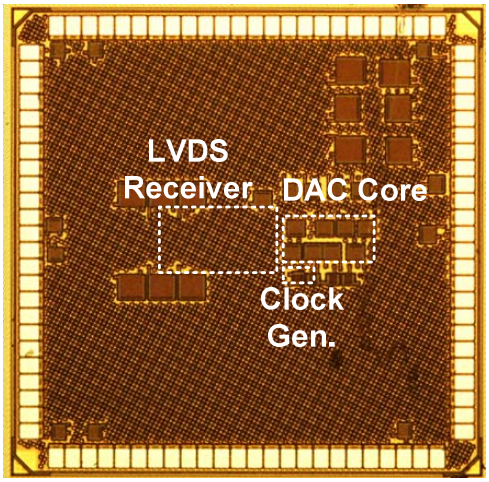

Figure 6. Die photo.

across the Nyquist band, which is improved by an average of 22 dB after calibration.

The total synthesized area is 0.11mm^2, with only 32% occupied by active standard cells (0.035mm^2) and 1.1% occupied by current cells (0.0012mm^2); the rest is allocated to filling cells. Figure 6 shows the die photo of the fabricated chip. By leveraging a cell-based design and APR, this is the smallest reported high-resolution DAC in this technology.

The power consumption of the DAC is only 5mW from a 1V power supply, at 250MHz, excluding output current; 67% of this power is consumed in the memory blocks. Table I summarizes the specifications of this DAC, along with several recent DAC designs. Despite being the only DAC using a fully automated physical design flow, the linearity and SFDR place it on the lower end, but competitive with other state of the art DACs with full-custom design. The primary advantage being the design cycle time is significantly reduced by EDA tools. Secondly, the power and area of the DAC, even when scaled by sample rate and process node, is the best reported among these high-performance DACs. We attribute this entirely to the power digital EDA tools used to optimize the layout, compacting its size. Smaller area results in lower total switching capacitance, which significantly reduces the power.

V. Conclusions

This paper presented the design of a low-power, low-area cell-based and first fully synthesized DAC, in which analog building blocks (custom cells) are automatically placed and routed (APRed). With minimal design time and effort, this DAC achieves a performance comparable to conventional DACs. We predict that with the rapid growth in the number of design rules, a cell-based design approach becomes a necessity, as it is much faster than full-custom design approaches. Semiconductor scaling only makes the problem worse, and it is only a matter of time before complex analog designs are forced in this direction; a path adopted decades ago by digital designers. Our cell-based approach also allows porting the design into other processes with negligible effort, as all of the code and scripts are reused with only minor adjustments.

Acknowledgement

This work is partially supported by Massachusetts Institute of Technology - Lincoln Laboratory (MIT-LL).

References

[1] R. Todd, L. Grodd, K. Fetty, "Design Rule Checking," *in EDA for IC Implementation, Circuit Design, and Process Technology,* CRC Press, 2006.

[2] D. Macmillen, R. Camposano, D. Hill, T.W Williams, "An industrial view of electronic design automation," *IEEE TCAD*, 2000.

[3] M. White. (2010, January, 28). *Pattern Matching Might Solve World Hunger* [online]. Available: http://blogs.mentor.com.

[4] Y. Park, D. D. Wentzloff, "An All-Digital PLL Synthesized from a Digital Standard Cell Library in 65nm CMOS," *CICC*, 2011.

[5] W. Deng, Y. Dongsheng, T. Ueno, T. Siriburanon, S. Kondo, K. Okada, A. Matsuzawa, "15.1 A 0.0066mm2 780μW fully synthesizable PLL with a current-output DAC and an interpolative phase-coupled oscillator using edge-injection technique," *ISSCC*, 2014.

[6] S. Weaver, B. Hershberg, U-K. Moon, "Digitally synthesized stochastic flash ADC using only standard digital cells," *VLSI Circuits,* 2011.

[7] M. Faisal, D. D. Wentzloff, "An Automatically Placed-and-Routed ADPLL for the MedRadio Band using PWM to Enhance DCO Resolution," *RFIC,* 2013.

[8] C-H. Lin, F. van der Goes, J. Westra, J. Mulder, Y. Lin, E. Arslan, E. Ayranci, X. Liu, K. Bult, "A 12b 2.9GS/s DAC with IM3 $\ll -60$dBc beyond 1GHz in 65nm CMOS," *ISSCC,* 2009.

[9] W-T. Lin, T-H. Kuo, "A 12b 1.6GS/s 40mW DAC in 40nm CMOS with >70dB SFDR over entire Nyquist bandwidth," *ISSCC,* 2013.

[10] M. S. Mehrjoo, J. F. Buckwalter, "A 10-b, 300-MS/s power DAC with 6-Vpp differential swing," *RFIC*, 2013.

[11] K. Doris, J. Briaire, D. Leenaerts, M. Vertreg, A. van Roermund, "A 12b 500MS/s DAC with >70dB SFDR up to 120MHz in 0.18μm CMOS," *ISSCC*, 2005.

[12] Y. Tang, J. Briaire, K. Doris, R. van Veldhoven, P. van Beek, H. Hegt, A. van Roermund, "A 14b 200MS/s DAC with SFDR>78dBc, IM3<−83dBc and NSD<−163dBm/Hz across the whole Nyquist band enabled by dynamic-mismatch mapping," *VLSI Circuits*, 2010.