Defining Analog Standard Cell Libraries for Mixed-Signal Computing enabled through Educational Directions

Jennifer Hasler, Georgia Institute of Technology, jennifer.hasler@ece.gatech.edu

***Abstract*—This work presents a mixed-signal cell library built through multiple generations of educational experiences. Digital standard cell libraries are ubiquitous for commercial and academic IC design. Analog standard cell libraries are rare within system-level analog design. Educational efforts in analog system design provide a path for developing these standard cell libraries. The effort included determining the essential blocks for this standard cell library based on previous explorations, as well as constraints to make the layout efficient. Analog and mixed-signal standard cells enable compilation of a high-level description of analog and mixed signal designs to full IC layout. A standard cell library definition allows components designed in any process to have broad utilization. A high-level to layout compilation empowers rapid movement between different IC processes.**

Digital standard cell design is ubiquitous for commercial and academic IC design (Fig. 1). Very few individuals do custom digital IC circuit or layout design unless they are building, or are in the process of building, digital standard cells. This intellectual structure even enables research into automated generation of digital standard cells [1]. Digital standard cells enable tool abstraction for compiling from higher-level representations (e.g. [2]).

Analog and resulting mixed-signal design abstractions are in a very different place (Fig. 1). Major companies that have a wide array of digital standard cells (e.g. ARM) do not have analog standard cells. Companies offering larger blocks have struggled in commercial spaces (e.g. [3]). Previous research efforts into analog standard cells are very few [4], [5], [6]. One can generate an initial list of generic standard cells, such as amplifiers, comparators, DACs and ADCs, references, bias currents, oscillators, and filters [4], [5], [7], topics one would find in a classic analog circuit design textbook (e.g. [8]). A few blocks (e.g. Op-amps, capacitors, and switches) could build a filter [4]. One could make straight-forward generalizations from digital standard cells [6]. The typical view is that analog standard cells require considerably larger number of design parameters for each cell than digital cells, resulting in far larger difficulties in developing analog standard cell libraries.

Interest in larger analog computing systems was initially fueled by neurally inspired computing and machine learning systems in the 1980s e.g. [9]), starting soon after the digital VLSI approach had become an accepted practice (e.g. [10]). These topics have regained some of that previous momentum. Research efforts in analog computing typically start with building a set of initial blocks to be used in later design. These efforts are excellent educational efforts by teams of

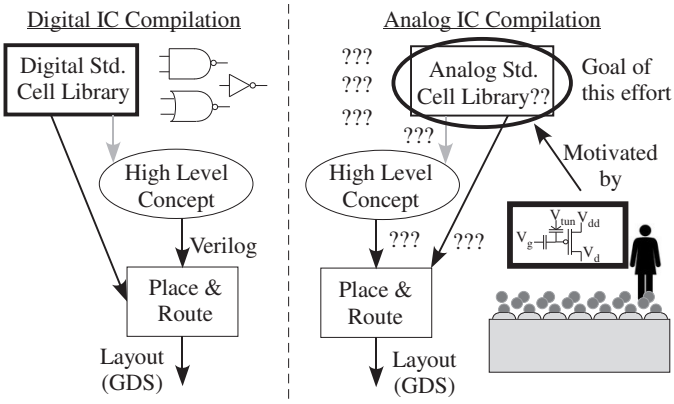

Fig. 1. Digital Standard Cells empower digital synthesis tools by building useful general abstractions (e.g. logic gates) of IC layout. Analog design does not have an equivalent Analog Standard Cell framework, and most can not imagine analog system synthesis. The goal of this effort is to work through a first analog standard cell library developed through teaching analog system design, particularly programmable and configurable analog system design.

graduate students, but rarely do such efforts build up to a reusable set of components that can span over a range of IC technologies. Within a few more years, the entire exercise must be repeated. Having analog standard cell blocks for computing, signal processing, and signal conditioning enables quick utilization of these approaches as well as encapsulates the wisdom of earlier generations.

This paper looks at one analog standard cell framework based on recent analog computing ICs enabled through educational efforts (Fig. 1). Often blocks developed as part of course activity, either for simulation or experimental results, in both explaining part of the field to a class, or as teaching exercises for the class. Arguably the most successful analog computing group in the 1980s and 1990s was at Caltech and was directly set by teaching these concepts, developing building blocks both used experimentally (CNS 182) over 30 weeks of class as well as openly shared with students for hands-on design projects (CNS 184). The mixture of experimental verification of core blocks that are commonly reused on a common platform concepts are exactly the framework for current standard cell methodologies. Educational experiences are essential in this area, much like educational experiences launched the original digital VLSI revolution [10]. As we see renewed interest in analog neural network and machine learning, one expects that having analog standard cells to integrate these functions will accelerate progress in this field

978-1-7281-3320-1/20/$31.00 ©2020 IEEE

	Cell Type	Variations
Analog Cells	Transistors (nFET + pFET)	Standard (W/L=1) Medium (W/L = 10), Large (W/L = 100)
	Transconductance Amplifiers	(Single, Diff) Input, (W/L, large W/L) input (non-FG, FG) input, (non-FG, FB bias)
	Capacitors	Five individual sizes, FG selectable cap
	Comparator (Latched)	
	Transmission Gate	

FG Cells	Cell Type	Variations
	Crossbar Cell (Same C_T)	(Indirect, Direct) Prog, (1, 2x2)
	FG Gate cell (Same C_T)	(0, 1, 2) gate inputs, (indirect, Direct)

Special Cells
Bootstrap Current Source
Signal by Signal Multiplier
7-Bit (expandable) DAC Module
Switch-capacitor resistor
Buffers driving off-chip loads

Fig. 2. Analog Standard Cell Library using Analog and FG cells. Mixed-signal compilation would include a digital standard cell library. *Analog Cells*: The analog cells utilize a number of fine-grain components (e.g. Transistors) and medium level components (e.g. Transconductance amplifiers, Comparators), for system compilation. These cells require a finite number of options for most system cases. *FG cells*: The FG cells utilize two values of total FG capacitance (C_T), with the Crossbar cell having a single smaller C_T, and the FG Gate cell has a single larger C_T. Digital on-chip memories use the direct FG cell.

while removing some barriers to entry in this field, particularly eliminating the need for each group to spend significant effort creating their own blocks as historically has been the case.

Our approach builds upon several generations of educational experiences across a number of labs to revisit building analog standard cells. Building and analyzing the results from the educational process provides the necessary knowledge to finishing an analog standard cell library. Likely an initial one standard cell library won't be universally efficient for all applications, and yet, this initial library enables the innovation towards multiple libraries with further innovations. Part of the intellectual work is figuring out what blocks are essential for a standard cell library (Sec. I), and how to make the layout and design efficient using these cells (Sec. II). The impact of this work would drastically decrease the design time of larger system analog and mixed-signal ICs, and laying a solid technology-independent foundation to abstract and automate analog design. Analog and mixed signal designs can be compiled from a high-level IC description, or an extension of it, to full IC layout (e.g. GDS) given a set of standard cells, paralleling a subroutine library for coding projects.

I. Creating an Analog Std. Cell Library

Educational experiences enable creating the definition and the resulting development of an analog standard cell library for IC design. Successful analog and mixed-signal courses develop and struggle to define important computational components constantly distilled by the finite number of weeks available in that particular course. The IC design library would utilize standard CMOS processes, utilizing IC design based on MOS transistors and capacitors where resistors and capacitors are far more expensive for typical designs. Concepts should be easily modifiable to SOI and FinFET CMOS process nodes. Some specializations may not be captured, such as RF transistors that require transistors of a particular characterized topology given the challenges of modeling RF parasitics [11].

Given the starting capabilities and assumptions from a number of educational experiences [12], we build a first attempt at fundamental analog standard-cell library blocks. For our standard cell library, we assume:

- programmability is essential, and
- subthreshold and near-threshold analog design.

System analog without programmability is extremely difficult, particularly given the extreme limitations of device mismatch that are compounded as IC processes scale down. Systems can be programmable as well as be insensitive to environmental conditions such as temperature or power supply variations. A standard cell library without programmability will grow very large to enable at least some programmability through the large number of cells at prefabrication design time. Near-threshold and subthreshold design appear in most high performance designs both given the high transconductance per unit current available, as well as the disappearance of an above-threshold transconductance significantly improved over the transconductance at the threshold current.

Programmable analog design changes the assumptions in analog design enabling of required analog circuit blocks for system-level design. Classic analog design assumes no programmability and puts a premium on feedback around high-gain amplifiers to guarantee sufficient accuracy in a fixed design space. For example, a highly accurate unity gain buffer achieves its accuracy by generating very high (e.g. 100,000) signal gain. Programmability allows for system-level adjustments which reduces the need for high-gain amplifiers to compensate for precision and mismatch, a space developed through educational efforts (e.g. [12]). This analog standard-cell library utilizes Floating-Gate (FG) techniques for programmability, although other techniques could be utilized where available. Without programmability, the set of cells is much much larger. Effectively one is getting programmability through numbers of elements at design time, and really only programmability at design time.

Studies in configurable mixed-signal and analog techniques provide alternate methods for analog standard cells and abstraction. D'Mello originally made a connection between standard cells and Field Programmable Analog Arrays (FPAA) illustrating that these are two alternate viewpoints of a similar space[13]. Recent work in configurable tools (e.g. [14]), used in educational directions (e.g. [12]), guides the structure of an analog standard cell library as well as the use of higher level

design tools similar to digital approaches (e.g. VPR/VTR [15]) using these standard cells. Decades of FPAA experience and fundamental circuit blocks provide an analog library roadmap and potential layout compilation. Standard cell libraries requires macromodelled simulation for layout optimization, a process already developing for configurable devices [14].

Figure 2 shows the blocks in the analog standard cell library. These cells would be used with a digital standard cell library for full mixed-signal capability. The standard cells in Fig. 2 have small overlap with the traditional analog circuit standard cell definition allowing for finer granularity to enable larger compilation from these stages. Digital blocks for mixed-signal computation should be compiled from Verilog. Some digital components for mixed-signal computation might be compiled for density, such as decoders or custom shift-register / SPI blocks. One could imagine adding additional analog or mixed-signal cells because of their higher density, and yet, like digital standard cell libraries, these components are not essential. And similarly, a baseline number of cells would be necessary for compilation like digital structures.

The analog standard cell library includes a mixture of fine and moderate level granularity components (Fig. 2), cells likely to be fabricated, characterized, and macromodelled for effective high-level synthesis. Transistor standard cells include an nFET and pFET, balancing its pitch with other topologies, where either or both transistors can be accessed. When designing with near threshold and subthreshold transistors, only a small number of W/L ratios are required. The key factors are selecting W/L for a sufficiently high threshold current as well as selecting W and L large enough for low 1/f noise applications. Transconductance Amplifier cells are core cells designed for on-chip amplification driving mostly capacitive and not resistive loads, and therefore, these components encompass the space of on-chip operational amplifiers. Multiple ampllifier topologies have similar capabilities, allowing to select a single design style (transconductance amplifiers) that would not have added stability issues with unknown capacitive loads. The number of W/L cases are similar to the three transistor sizes, and are primarily for input differential pairs either for low thermal noise [16], high threshold currents, or low 1/f noise output voltage. Amplifier topologies also include choices between FG for bias currents and/or amplifier inputs, as well as single-ended or differential inputs. Cascode transistors are set through FG programming. The programmable (FG) cells include direct and indirect programming devices (e.g. [17]) in a crossbar pattern, as well as slightly larger cells that may utilize one or more input gates. The 2x2 crossbar devices take advantage of area advantages for mirrored cells. A standard set of test cells would enable characterizing these test cells.

II. Efficient Analog Standard Cell Design

After defining the cell library components, one is faced with developing a framework for placing and routing these cells. Targeting circuits on SoC FPAA designs [18] provide framework for analog synthesis of high-level abstracted designs to a netlist representation for place and route that then translate to

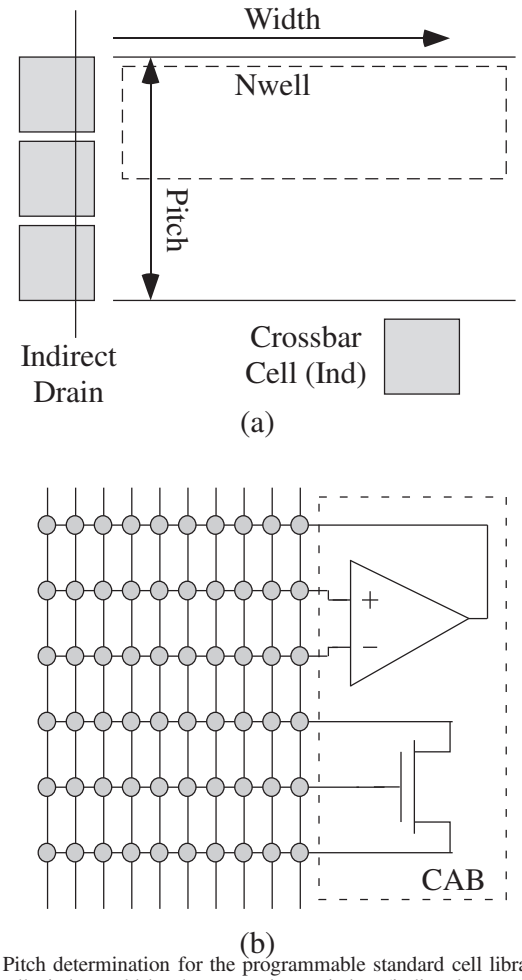

Fig. 3. Pitch determination for the programmable standard cell library. (a) A typical cell pitch would be three crossbar switches (indirectly programmed), given the experience of FPAA structures where most analog components fit vertically within three crossbar elements. The cell typically have power supplies running along the top of the cell, or have particular metal vias to higher level metals. (b) Typical FPAA routing infrastructure utilizes three lines for many elements, including transconductance amplifiers and transistors, where each routing circle is an indirect-FG switch (e.g. [18]).

connections to particular analog and digital components [14], [19]. These tools include a range of realistic macromodeling that enables simulation at multiple levels [14], [20]. A similar design flow would would target this design library (e.g. LEF, GDS, and timing info) towards a custom IC layout. One early attempt simply just paralleled early digital efforts in this area without specific analog system level experiences [6].

Standard cells tend to utilize a single standard cell pitch, or a selectable pitch, to allow cells to easily connect along the same row (Fig. 3). The pitch constrains the overall IC layout. Borrowing from FPAA designs, where three routing lines are required for an transconductance amplifier, T-gate, transistor, and many other elements (e.g. [21]), a typical pitch size would likely be the vertical size of three Indirect Programmed Crossbar Cells built in the target IC process (Fig. 3). The cell layout should constrain routing to the first two metal levels, with via contacts to higher level metals as connection points. Higher level metals can route over these cells as required, although some protection might be used (e.g. ground plane

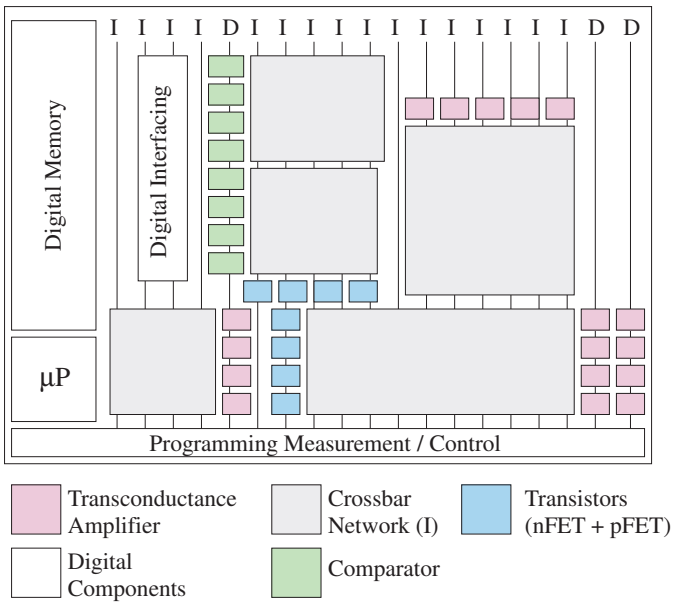

Fig. 4. Illustration of a mixed signal chip with using programmable analog standard cell elements, including infrastructure for programming these cells. Particular cells will have programmable parameters that might utilize direct (D) or indirect (I) programming, requiring that the device be placed along the corresponding vertical measurement line.

for metal 3) for more sensitive analog components.

Figure 4 shows the constraints on placement for routing analog standard cells. We expect additional constraints when integrating analog and digital standard cells on the same die, particularly as they likely have different pitch width, likely resulting in analog and digital components on different rows in a particular block. Figure 4 shows FG programming infrastructure requiring direct (D) and indirect (I) columns / rows. FG standard cells could utilize either indirect or direct programming, requiring different columns for effective programming. The library components can be utilized in a number of ways. A crossbar block would be utilized for a switch matrix, a memory array, or for computing several weighted-input sums.

III. Summary and Discussion

We presented an analog cell library for CMOS processes built through multiple generations of educational experiences, as well as multiple generations of FPAA ICs and tools [21]. The effort included determining the essential blocks for this standard cell library based on previous explorations (e.g. [12]), as well as constraints to make the layout efficient. Analog and mixed-signal standard cells can enable compilation of a high-level description of analog and mixed signal designs to full IC layout. Having a standard cell library definition allows components designed in any process to have broad utilization. A high-level to layout compilation empowers rapid movement between different IC processes. Scaling configurable designs admit both higher density (100× from 350nm to 40nm) and lower energy and power (100× lower from 350nm to 40nm), while lowering commercial cost per computation from existing designs [22]. Analog designers can develop a set of cells that immediately have impact in any IC process, rapidly improving portability of designs between different IC processes. We expect these techniques will have an impact on the design of new generations of custom mixed-signal ICs, particularly impacting the next generations of configurable ICs [21].

The opportunities from on-chip programmable analog circuit design enable this standard cell library definition [12]. Programmability transforms on-chip analog design from selecting many W/L values for setting bias currents to directly programming these currents values, programming that happens after fabrication and can directly be employed in a fielded part with precision (e.g. [21], [23], [24]). One can program the bias current of a FG transistor over the entire subthreshold and above threshold range (e.g. 1pA to 10μA), and completely different parameters could be available after fabrication (e.g. [25]). These programmable bias currents directly set the desired transconductance, output resistances, thermal noise level, and time constants, where the programmability allows for direct elimination of mismatch after IC fabrication. Using the three available W/L values, different W/L are available through parallel and series transistor combinations. This trajectory maps well to scaled-down technologies that restrict the number of W/L available. Programmability enables high-precision system design by giving confidence of signal gains, offsets, and time-constants without the need for high-precision passive components in feedback around amplifiers to guarantee a precise parameter. These techniques allow for a range of open-loop, or near open-loop, amplifier topologies with selectable gains (e.g. capacitor choices) for the desired signal devices. The larger and smaller W/L values allows one to make other key design choices such as having a higher or lower threshold current (I_{th}) for a particular transistor, or making design choices for improved 1/f noise performance (e.g. [16]). These new opportunities comes from new techniques in analog design, techniques that can be abstracted to enable the user through the use of analog high-level synthesis (e.g. for FPAAs [14]) and these standard-cell opportunities. These approaches allow constructing a complete set of computations over a wide range of features and capabilities, worked out over generations of FPAA designs (e.g. [21]).

One expects that a standard cell library would be expanded given the perspectives of the particular library designer as well as the opportunities given by a particular IC process. For example, although most CMOS processes do not have (vertical) BJT devices, where a process allows or where a designer's perspective requires a BJT device or related circuits, these structures could be added to a standard cell library. One expects there will be specialized cells for specialized interfacing of off-chip components. And yet, the specialized design of these components could find wider use than the single application for which they were designed. And although such a standard cell library is expandable for these functions, this initial analog library encompasses a very wide range of on-chip analog functions with a minimum number of cells.

REFERENCES

[1] E. Alon, "CRAFT Generator-Based Hardware Design - Digital," *GOMAC*, 2018, pg. 448-451.

[2] B. Khailany, M. Fojtik, A. Klinefelter, E. Krimer, M. Pellauer, N. Pinckney, H. Ren, Y. S. Shao, R. Venkatesan, Y. Zhang, and B. Zimmer, "A Modular Digital VLSI Flow for High-Productivity SoC Design," *GOMAC* 2018, pp. 452-455

[3] Barcelona Design, http://www.barcelonadesign.com. Company is not in business any longer (started in 1999); link provided as a reference.

[4] T. Pletersek, J. Trontelj, L. Trontelj, I. Jones, and G. Shenton, "High-Performance Designs with CMOS Analog Standard Cells," *IEEE Journal of Solid State Circuits*, vol. 21, no. 2, April 1986. pp. 215-222.

[5] R. J. Bowman, "Introduction to CMOS analog standard cell ASIC design," *IEEE ASIC Seminar*, Sept. 1989. pp. T2-1.1-1.4.

[6] S. Shinbara, *Analog Standard Cell*, US. Patent, 5,302,864, April 12, 1994.

[7] C. Toumazou, George S. Moschytz, Barrie Gilbert, *Trade-offs analog circuit design: the designer's companion* Kluwer Publishers, Boston, 2002.

[8] P.A. Allen and D. R. Holberg, *CMS Analog Circuit Design*, 2^{nd} edition, Oxford University Press, 2002.

[9] C. Mead, "Analog VLSI and Neural Systems," *Addison Wesley*, 1989.

[10] Carver Mead and Lynn Conway "Introduction to VLSI System Design,", Addison-Wesley, 1980. ISBN 0-201-04358-0. Early drafts found at http://ai.eecs.umich.edu/people/conway/VLSI/VLSIText/VLSIText.html.

[11] J. Laskar, B. Matinpour, S. Chakraborty, *Modern Receiver Front-Ends*, John Wiley and Sons, Wiley, 2004.

[12] J. Hasler, "Circuit Implementations Teaching a Junior Level Circuits Course Utilizing the SoC FPAA," *ISCAS 2018*, Florence, Italy, May 2018.

[13] D.R. D'Mello and P.G. Gulak, "Design Approaches to Field-Programmable Analog Integrated Circuits," *Analog Integrated Circuits and Signal Processing*, vol. 17,No. 1/2, 1998. pp. 7-34.

[14] M. Collins, J. Hasler, and S. George, "An Open-Source Toolset Enabling Analog–Digital–Software Codesign," invited paper *Journal of Low Power Electronics Applications*, Vol. 6, no. 1, February 2016, pp. 1-15.

[15] J. Luu, J. Goeders, M. Wainberg, A. Somerville, T. Yu, K. Nasartschuk, M. Nasr, S. Wang, T. Liu, N. Ahmed, K. B. Kent, J. Anderson, J. Rose, and V. Betz, "VTR 7.0: Next Generation Architecture and CAD System for FPGAs," vol. 7, 2014. pp 6:1–6:30.

[16] R. Harrison, "A low-power low-niose CMOS amplifier for neural recording applications," *IEEE Journal of Solid State Circuits*, vol. 30, no. 6, 2003. pp. 958-965.

[17] D. W. Graham, E. Farquhar, B. Degnan, C. Gordon, and Hasler, "Indirect programming of floating-gate transistors," *IEEE Transactions on Circuits and Systems I*, vol. 54, no. 5, 2007. pp. 951 - 963.

[18] S. George, S. Kim, S. Shah, J. Hasler, M. Collins, F. Adil, R, Wunderlich, S. Nease, and S. Ramakrishnan, "A Programmable and Configurable Mixed-Mode FPAA SoC," *IEEE Transactions on VLSI*, vol. 24, no. 6, 2016, pp. 2253-2261.

[19] J. Hasler, S. Kim, and A. Natarajan, "Enabling Energy-Efficient Physical Computing through Analog Abstraction and IP Reuse," *Journal of Low Power Electronics Applications*, December 2018. pp. 1-23.

[20] A. Natarajan and J. Hasler, "Modeling, simulation and implementation of circuit elements in an open-source tool set on the FPAA," *Analog Integrated Circuits and Signal Processing*, vol. 91, no. 1, April 2017. pp. 119-130.

[21] J. Hasler, "Large-Scale Field Programmable Analog Arrays," *IEEE Proceedings*, February 2020.

[22] J. Hasler, S. Kim, and F. Adil, "Scaling Floating-Gate Devices Predicting Behavior for Programmable and Configurable Circuits and Systems," *Journal of Low Power Electronics Applications*, vol. 6, no. 13, 2016, pp. 1-19.

[23] V. Srinivasan, G. J. Serrano, J. Gray, and P. Hasler, A precision CMOS amplifier using floating-gate transistors for offset cancellation, IEEE Journal of Solid-State Circuits, vol. 42, no. 2, pp. 280-291, Feb. 2007.

[24] V. Srinivasan, G. Serrano, C. Twigg, and P. Hasler, A Floating-Gate-Based Programmable CMOS Reference, IEEE Transactions on Circuits and Systems I, Vol. 55, No. 11, pp. 3448 - 3456, Dec. 2008.

[25] S. Kim, J. Hasler, and S. George, "Integrated Floating-Gate Programming Environment for System-Level Ics," *IEEE Transactions on VLSI*, vol. 24, no. 6, 2016. pp. 2244-2252.