A CMOS Programmable Analog Standard Cell Library in Skywater 130nm Open-Source Process

Jennifer Hasler
Electrical and Computer Engineering
Georgia Institute of Technology
jennifer.hasler@ece.gatech.edu

Barry Muldrey and Parker Hardy
Electrical and Computer Engineering
University of Mississippi
muldrey@olemiss.edu, pwhardy@go.olemiss.edu

***Abstract*—This work presents an implemented, and where allowed, openly-available programmable analog standard cell library in the open-source skywater 130nm bulk CMOS processes. Standard CMOS elements enable programmablility for many standard-cell components, eliminating the need for large number of device geometries required in classic analog design. This effort presents the methodology in developing these analog standard cell library. The standard cell library was designed to integrate with digital open-source synthesis tools.**

Digital standard-cell libraries are ubiquitous for custom digital IC design (Fig. 1), enabling tool abstraction for compiling from higher-level representations (e.g. [1]) as well as research into automated generation of digital standard cells [2]. Analog standard cell libraries are not used because of the far larger number of cells required for basic functionality due the large number of classical analog design parameters (Fig. 1). The previous history of attempts at analog standard cell approaches is sparse (e.g. history described in [3]) with no system design expectation. Although one can generate a list of generic standard cells, without programmable parameters ([4], [5], [6]), the library block size is nearly intractable, requiring many different transistor width (W) and length (L) to set bias currents and other device parameters, and dealing with restrictions due to transistor mismatch (e.g. V_{T0} mismatch).

This work presents a programmable analog standard cell library (Fig. 1), motivated by previous educational efforts [3], developed and taped out in the Skywater 130nm open-source CMOS process[1], where the entire design was developed with open-source tools (e.g. magic, Xschem), as well as the cells are openly available to be integrated into additional designs. Analog programmability is essential. Cell level programmability greatly reduces the complexity of an analog library, where parameters such as bias currents and voltage offsets are programmed instead of set by multiple transistor sizes.

I. Programmable FG Standard Cell Library

This library achieves programmability using Floating-Gate (FG) devices possible on standard IC CMOS processes, and the FG cell design sets the structure for this standard cell library. The FG programmability [7], [8], as well as extensive experimental measurements and modeling of these devices gives reasonable confidence for or first-pass results. Often FG parameters are uncertain during design as such structures are not characterized in a particular IC process, or not characterized in a way that directly translates to the design.

[1]Cells and design available at https://gitlab.com/um-ece/ftl-lab/hilas

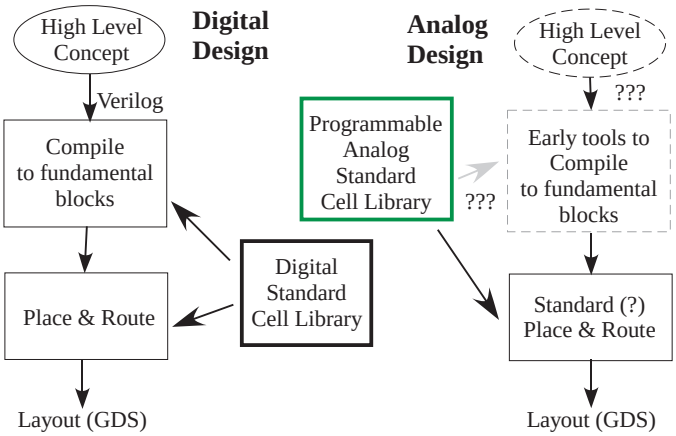

Fig. 1. Digital standard cells enable large-scale synthesis. This effort demonstrates an programmable analog standard-cell library designed in multiple IC processes (350nm, 130nm, and 14nm) that could utilize existing standard place-and-route tools.

Developing an FG device in a CMOS process requires understanding the availability of the input capacitors and the tunneling junction capacitance (Fig. 2) to the gate conductor layer. The FG node is entirely made of the gate conductor layer with no contacts to this layer (Fig. 2) to avoid any long-term charge leakage paths. Long-term (e.g. 10 year) FG retention requires SiO_2 insulators greater or equal to 5nm CMOS (e.g. thicker insulator in 130nm). Hot-electron injection in pFET devices, the method for precisely programming electrons on the FG node, robustly operates for subthreshold and near-threshold currents across process nodes and temperatures with fairly predictable operating ranges (e.g. [9]). V_{inj} is the high-voltage supply that gets raised for injection. the chip V_{dd} does not move under programming.

FG development requires choosing between various implementation capacitors to the FG node (Fig. 2). More capacitors available gives the IC designer more options. In some processes two-layers of polysilicon or a direct capacitor to the polysilicon layer is available, enabling linear capacitors including in a vertical stack, but this option was not available in Skywater's 130nm CMOS process. A typical CMOS process without two-layers of polysilicon (e.g. 130nm) has varactor

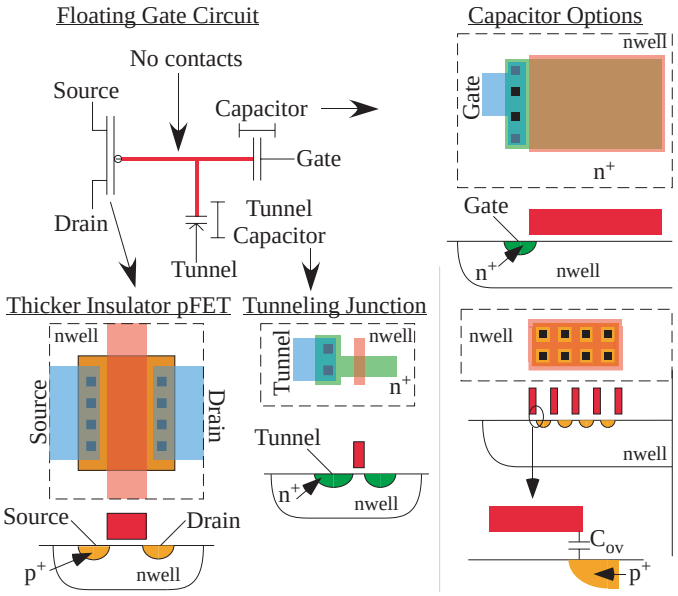

Fig. 2. A Floating-Gate (FG) circuit device requires a thick-insulator pFET for circuit operation and hot-electron injection programming, a single gate conductor with no contacts between elements, at least one input capacitor, and one tunneling capacitor. The capacitors for a typical bulk CMOS process (e.g. 130nm CMOS) include varactors (n^+ in nwell) and gate to diffusion overlap capacitors. Tunneling capacitor uses a minimal size capacitor, where the input capacitor(s) are typically a larger size. The total capacitance (C_T) sets FG programming timing.

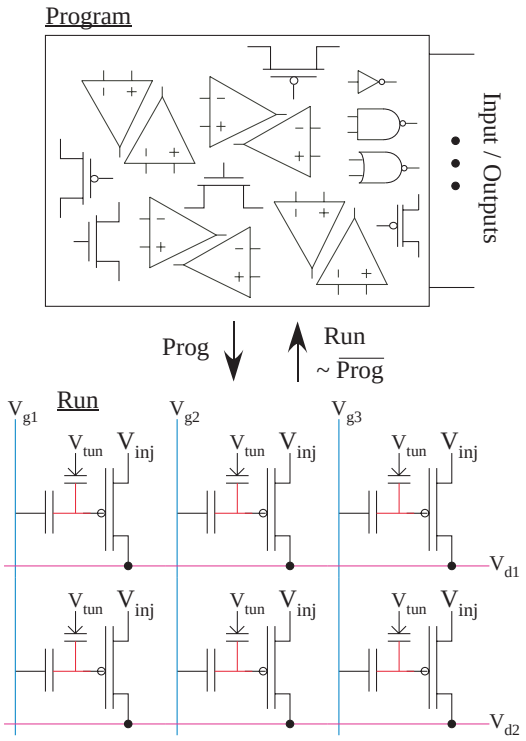

Fig. 3. Program (*Prog*) and run (*run*) mode. A system that uses analog devices and computation with inputs and outputs in *run* mode must be reconfigured to a standard crossbar array in *program* mode.

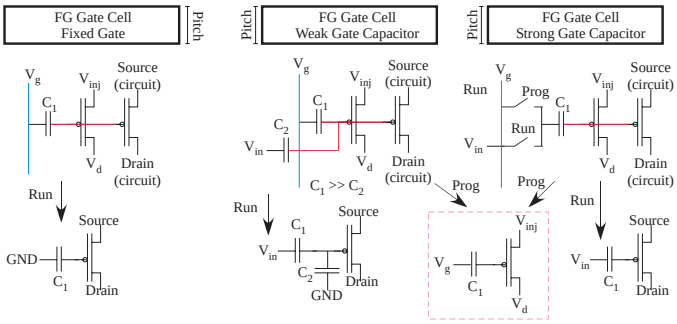

Fig. 4. llustration of program (*Prog*) and run (*run*) mode using the FG Gate Capacitor cells. The FG Gate cell conforms to a single crossbar element and a capacitor circuit element by reconfiguring T-gates set by the *run* or *prog* signals. The indirect FG elements use an additional pFET switch to select only a particular column, enabling high-isolation and wide-range (10pA to 100's μA) current measurement and programming.

capacitors as well as gate-to-source–drain overlap capacitors to the polysilicon gate layer. Varactor capacitors, n^+ regions in nwell, are commonly used structures for single-poly FG structures [9]. Gate-to-source–drain overlap capacitors use the overlap capacitance to create a linear capacitance; the resulting layout tends to be drawn in a *waffle* configuration to maximize this overlap capacitance per unit area. These overlap capacitances also provide a guard-ring to the substrate, reducing the breakdown in the device, and in-turn, increasing the operating voltage operation before reaching breakdown effects. These capacitances typically match well between devices. The standard cell designs includes both approaches.

Tunneling junctions are varactors (Fig. 2) with minimal gate capacitance to minimize tunneling voltage coupling into the FG allowing for lower tunneling operating voltages (e.g. 10-11V for 130nm CMOS). V_{tun} is the high-voltage signal / supply that is raised for electron tunneling. The higher tunneling capacitor voltage requires additional well-spacing because of the larger depletion regions. Tunneling through high-quality gate insulator results in large number of equivalent write cycles (e.g. $> 10^9$) [10]. The tunneling through high-quality insulator requires significant spacing between the tunneling well and other wells. Overlap capacitors require less area spacing between tunneling and active devices.

II. Programmable Standard Cell Library Design and Implementation

The methodology for this programmable standard-cell library developed from compiling, targeting and use of analog and mixed-signal systems (e.g. [11]) derived from educational experience [3]. Programmable analog design changes analog design assumptions, eliminating the need for feedback around high-gain amplifiers to guarantee sufficient accuracy typical in a fixed design space. FG programming after fabrication can directly be employed in a fielded part with precision (e.g. [11], [12], [13]) over the entire subthreshold and above threshold range (e.g. 1pA to 10μA) [8], enabling setting a desired transconductance, output resistance, thermal noise level, and time constant, as well as these programmable systems can be insensitive to environmental conditions such as temperature or power supply variations (e.g. [12], [13], [14]).

Analog Cells:

Cell Type	Variations	Name	Width
Transconductance Amplifiers (TA)	2 TA, FG bias, no FG inputs	TA2Cell_NoFG	17.92μm
	2 TA, FG bias, FG inputs (V_L large)	TA2Cell_1FG	28.09μm
	2 TA, FG bias, FG inputs (V_L small)	TA2Cell_1FG_Strong	28.10μm
	2 TA, signal bias, no FG inputs	TA2SignalBiasCell	8.45μm
Capacitors	Selectable 16 unit cap	capacitorArray01	36.70μm
	Two separate unit caps	capacitorSize04	5.78μm
	2 unit cap	capacitorSize03	5.79μm
	4 unit cap	capacitorSize01	10.42μm
	8 unit cap	capacitorSize02	7.97μm
Winner-Take-All (WTA)	4 WTA stages	WTA4Stage01	14.07μm
Ratioed Transistor DAC Modules	5 bit transistor module	DAC5bit01	16.58μm
Transmission Gates	4 Single Throw T-gates	Tgate4Single01	4.76μm
	4 Double Throw T-gates	Tgate4Double01	7.08μm
	4 T-gate for prog select (V_{inj})	drainSelect01	5.42μm
Transistors (nFET + pFET)	3 nFETs + 3 pFETs (W/L≈1)	Trans4small	2.80 μm
	2 nFETs + 2 pFETs (W/L≈10)	Trans2med	3.53μm
	1 nFETs (W/L≈100)	nFETLarge	4.37μm
	1 pFETs (W/L≈100)	pFETLarge	4.64μm

FG Cells:

Cell Type	Variations	Name	Width	V / O
Crossbar Cell (Same C_T)	4×2 Cell	swc4x2cell	20.12μm	V
		swc4x2cellOverlap	17.97μ	O
	4×1 Bias	swc4x1BiasCell	10.11μm	V
	4×1 cell	cellAttempt01	10.08μm	V
FG Gate cell (Same C_T)	2 x 1 FG Biases	FGBias2x1cell	11.53μm	V
	2 x1 FG Transistors	FGtrans2x1cell	11.52μm	V
	2 x1 FG Trans, small input C	FGBiasWeakGate2x2cell	11.53μm	V
FG Characterization cell	Tun, pFET (inj), Capacitors	FGcharacterization01	29.95μm	V+O

Fig. 5. The 130nm CMOS proposed analog standard cell library composed of Analog and FG cells. *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. The TA have different linear range (V_L) depending on their coupling capacitors. These cells only require a finite number of options to cover 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. The library includes Varactor (V) and Overlap (O) Capacitors.

FG devices have near ideal selectivity in a two-dimensional crossbar array (Fig. 3) [7], [9]. FG circuits have two modes of operation, program mode (*prog*) and run mode (*run*), and the circuitry must reconfigure to enable both modes (Fig. 3). When in program mode (*prog* = 1), multiple circuit parts remain at the operating V_{dd}, although they could be set at 0V. These techniques have been standard for programming heterogeneous groups of FG devices [15], [11]. The reconfigure circuitry between *prog* and *run* is highly dependent on the particular element. For a set of switch cells (Fig. 4), no reconfigurability is required inside the core cell element, but can be handled at the edges of the array. Other circuits require explicit switches (e.g. T-gate switches) to reconfigure between *prog* and *run* (Fig. 4). When using FG cells that have the one or two gate lines as a cell input (Fig. 4), the gate line must have switches between the cell input(s) and the particular column gate line control for programming. Using indirect programming techniques removes requiring a T-gate on the transistor drain line by having separate paths for computation and programming. The infrastructure required directly impacts each standard cell design, as well as core line routing and alignment.

The 130nm programmable standard library is a typical standard bulk CMOS flow (Fig. 5). This finite set of programmable cells provide a wide range of computational functions (e.g. [11]), including signal conditioning and filtering using TA and capacitor elements, mixed-signal DACs as well as derived ADC components, transmission gates for modulation and signal flow, and Vector-Matrix Multiplication (VMM) through the crossbar cells. The Winner-Take-All (WTA) block enables a number of mixed-signal computations, including a VMM+WTA classifer, a structure that densely implements classification / inference / on-chip embedded machine learning [11]. Additional cells going forward could include latched comparitor explicit Bandpass Filter, amplitude, and zero-

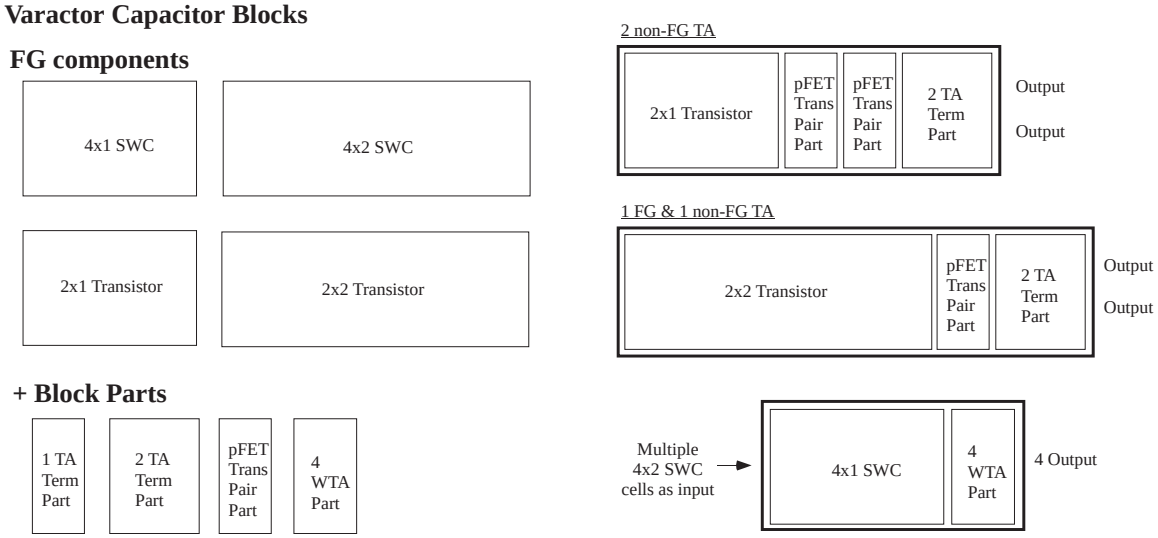

Fig. 6. Larger library blocks are often composed in part or nearly in whole from earlier blocks, often using on programmable block with a nonprogrammable block, as in the Transconductance Amplifier (TA) blocks or Winner-Take-All (WTA) block.

crossing blocks, explicit high-voltage digital blocks (e.g. shift-registers), as well as charge-pump blocks.

The FG switch cells set the structure and pitch of the FG-based Standard Cell library (Fig. 5). Cell names all have the prefix `sky130_hilas_` in the 130nm library. Pitch, or vertical size, of all cells is 6.5μm for this Skywater 130nm CMOS process, set by the 4 FG crossbar switches (*swc4x2cell*). Core FG cell is an indirect programmed switch with an additional cutoff transistor[16], [11]. For these cells, V_{dd} would be from 1.5V to 1.8V, V_{inj} would be 1.5-6V, and V_{tun} would operate at 1.8V and occasionally at 0V and 11-12V. The well definition is the same for either 1.8V and 5V devices and sets the well-to-well spacing for all devices except for a larger well-to-well spacing for tunneling capacitors.

This structure provides core functionality for more complex blocks, such as the FG TA block. The number of different total FG capacitance (C_T) needs to be minimized as FG device programming timing is proportional to C_T; this standard cell library uses only two C_T values. These core analog cells often have smaller subcomponents (Fig. 6) that could be used or modified by the user, although they are not in the core pitch and not explicitly discussed. The FG characterization structure (e.g. *FGcharacterization01*) provides a physics FG characterization, where the TA fixes the FG node while measuring low currents through a current integrator structure.

III. Summary

This discussion shows the design of a programmable analog standard cell library in Skywaters open-source 130nm CMOS prccess. Programmability eliminates the need for large number of device geometries required in classic analog design library. Programmable standard cells were defined, reduced to practice in layout, are in process for fabrication using a core test structure, and have the framework defined to be used with standard synthesis tools. Developing standard cells in different bulk CMOS processes demonstrated these techniques are applicable to a range of CMOS processes.

References

[1] 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.

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

[3] J. Hasler, "Defining Analog Std. Cell Libraries for Mixed-Signal Computing enabled through Educational Directions," *IEEE ISCAS*, 2020.

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

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

[6] 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.

[7] Hasler, B. Minch, and C. Diorio, Adaptive circuits using pFET floating-gate devices, *Advanced Research in VLSI*, March 1999, pp. 215-229.

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

[9] 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*, July 2016.

[10] Hasler, B. Minch, and C. Diorio, "An autozeroing floating-gate amplifier, *IEEE CAS II*, vol. 48, no. 1, pp. 7482, Jan. 2001.

[11] J. Hasler, "Large-Scale Field Programmable Analog Arrays," *IEEE Proceedings*, vol. 108. no. 8. August 2020. pp. 1283-1302.

[12] V. Srinivasan, et. al, "A precision CMOS amplifier using floating-gate transistors for offset cancellation," *IEEE JSSC*, vol. 42, no. 2, pp. 280-291, Feb. 2007.

[13] 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.

[14] S. Shah, H. Toreyin, J. Hasler, and A. Natarajan, "Models and Techniques For Temperature Robust Systems On A Reconfigurable Platform," *Journal of Low Power Electronics Applications*, vol. 7, no. 21, August 2017. pp. 1-14.

[15] M. Kucic, Hasler, J. Dugger, and D. Anderson, "Programmable and adaptive analog filters using arrays of floating-gate circuits," *Advanced Research in VLSI*, March 2001, pp. 148 162.

[16] C. M. Twigg and Hasler, "A large-scale reconfigurable analog signal processor (RASP) IC," *IEEE Custom Integrated Circuits Conference*, Sept 2006, pp. 5 8.