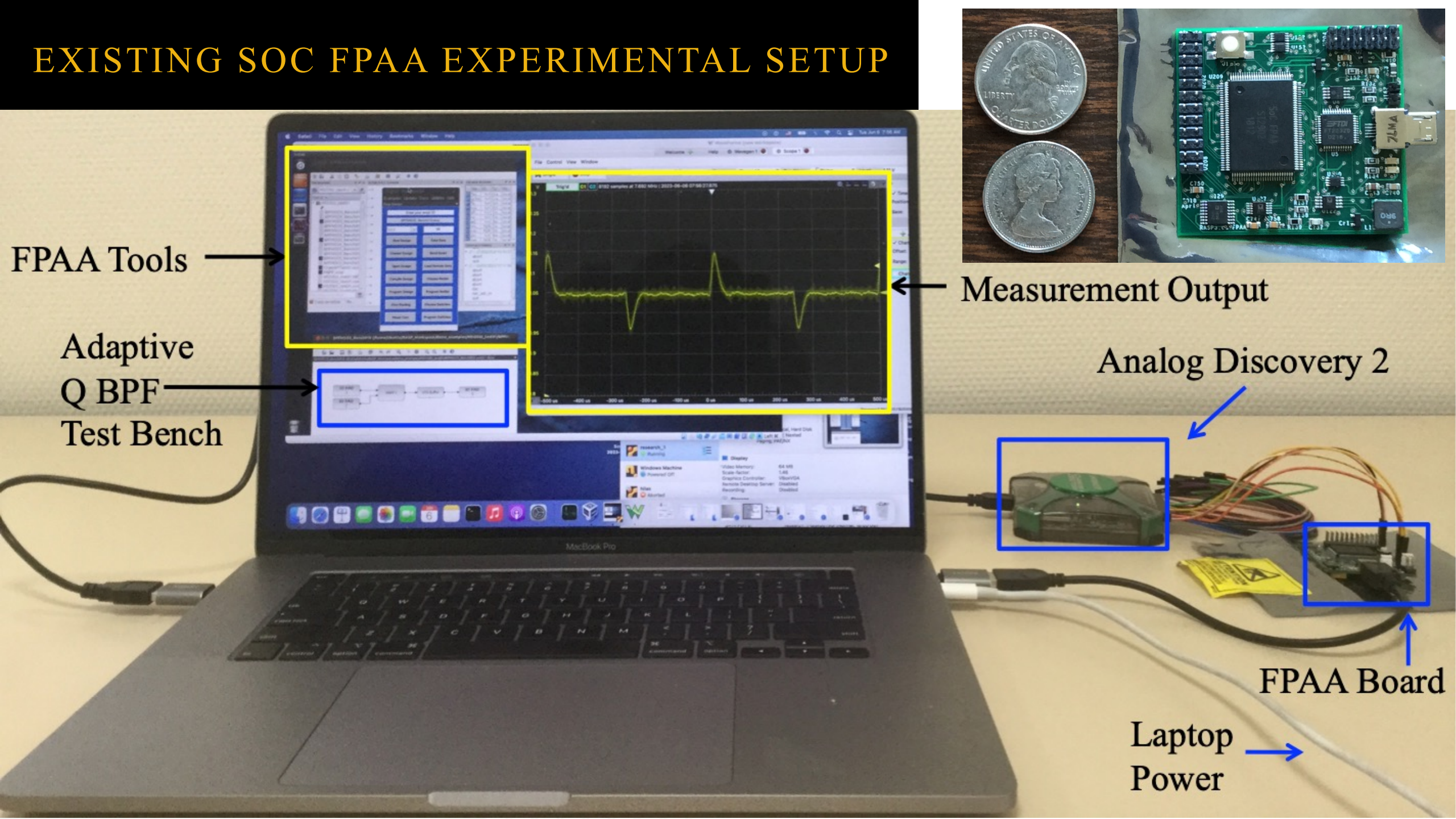
Analog Programmable Standard Cells and Synthesis

Georgia Institute of Technology (ATL & Metz)

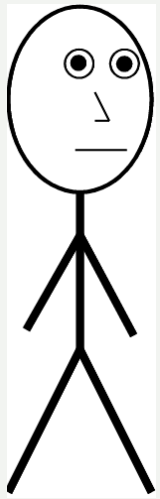
Jennifer Hasler

Georgia Tech College of Engineering
School of Electrical and Computer Engineering

EXISTING SOC FPAA EXPERIMENTAL SETUP



ANALOG COMPUTING IMPLEMENTATIONS SEEM EXTREMELY DIFFICULT



Analog Implementations Seem Hard
(Lived Experience)

- Device mismatch
- Programmability?
- Configurability? (Writing a program)
- Build on multiple levels?
- **Design Tools**
- Not enough computational examples
- Noise and noise accumulation
- Put elements together to compute?
- Computability theory?

Current Capabilities &
Developing Capabilities

Floating-Gate (FG) on CMOS

Field Programmable Analog Arrays (FPAAs)

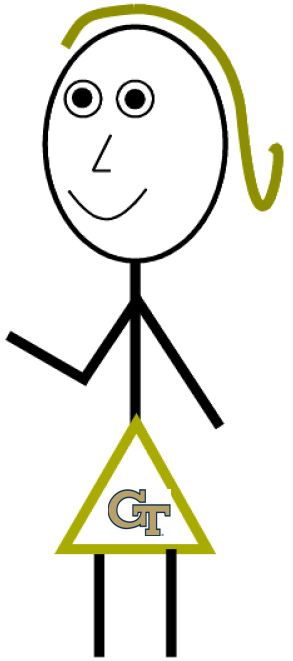
Analog Abstraction

FPAA targeting & IC synthesis

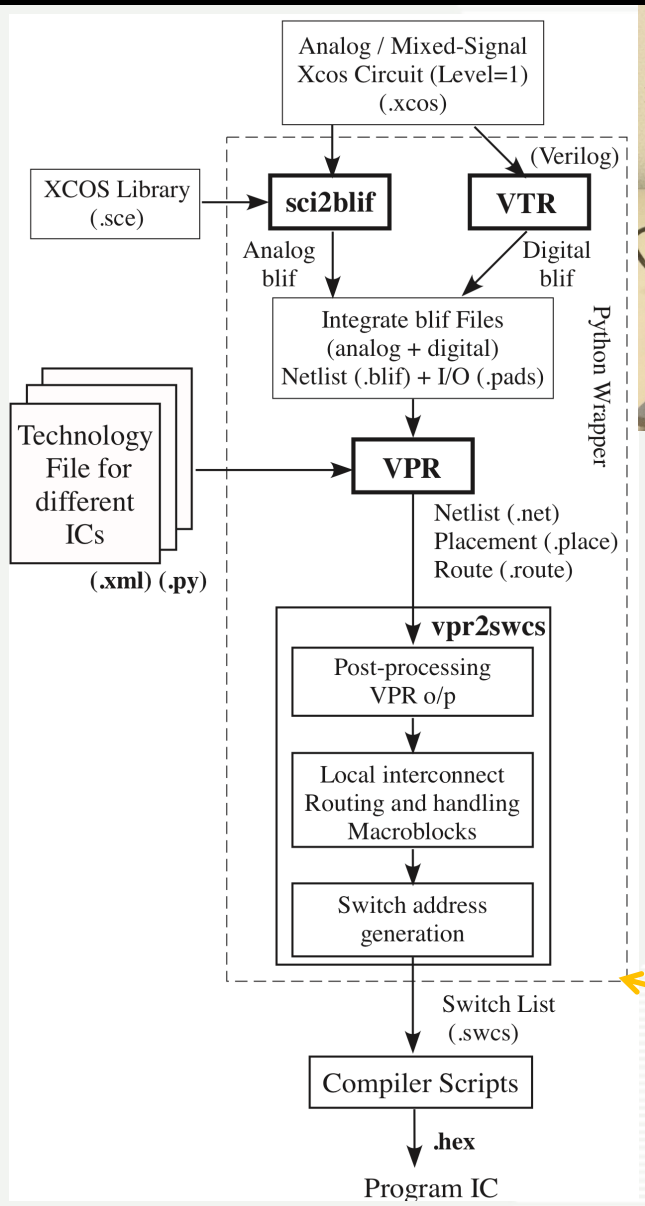
Analog Numerical Analysis

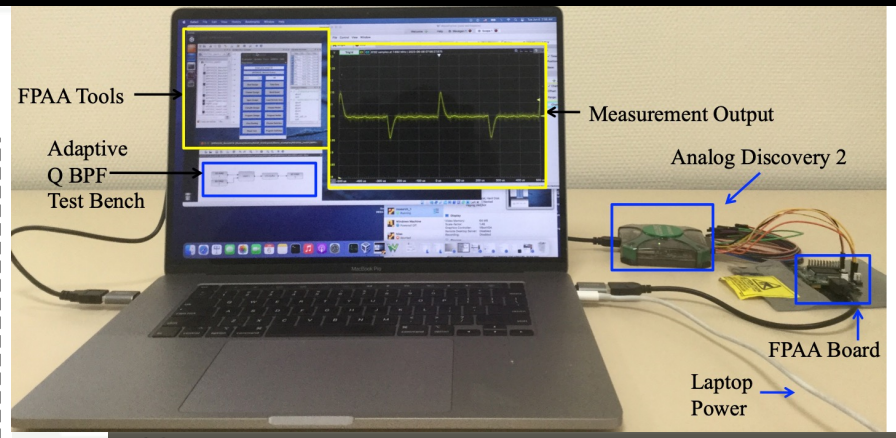
Analog Architectures

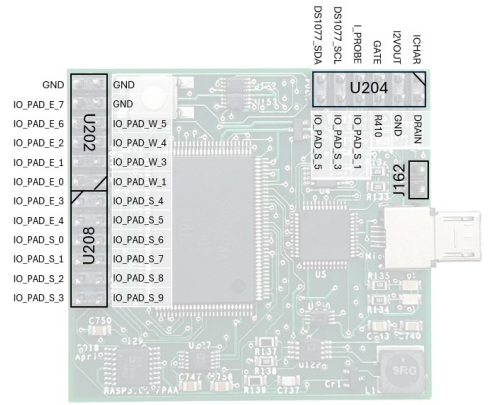
Compute over Reals, Physical Computing, $\aleph_1$ Turing



FPAA TOOL FRAMEWORK

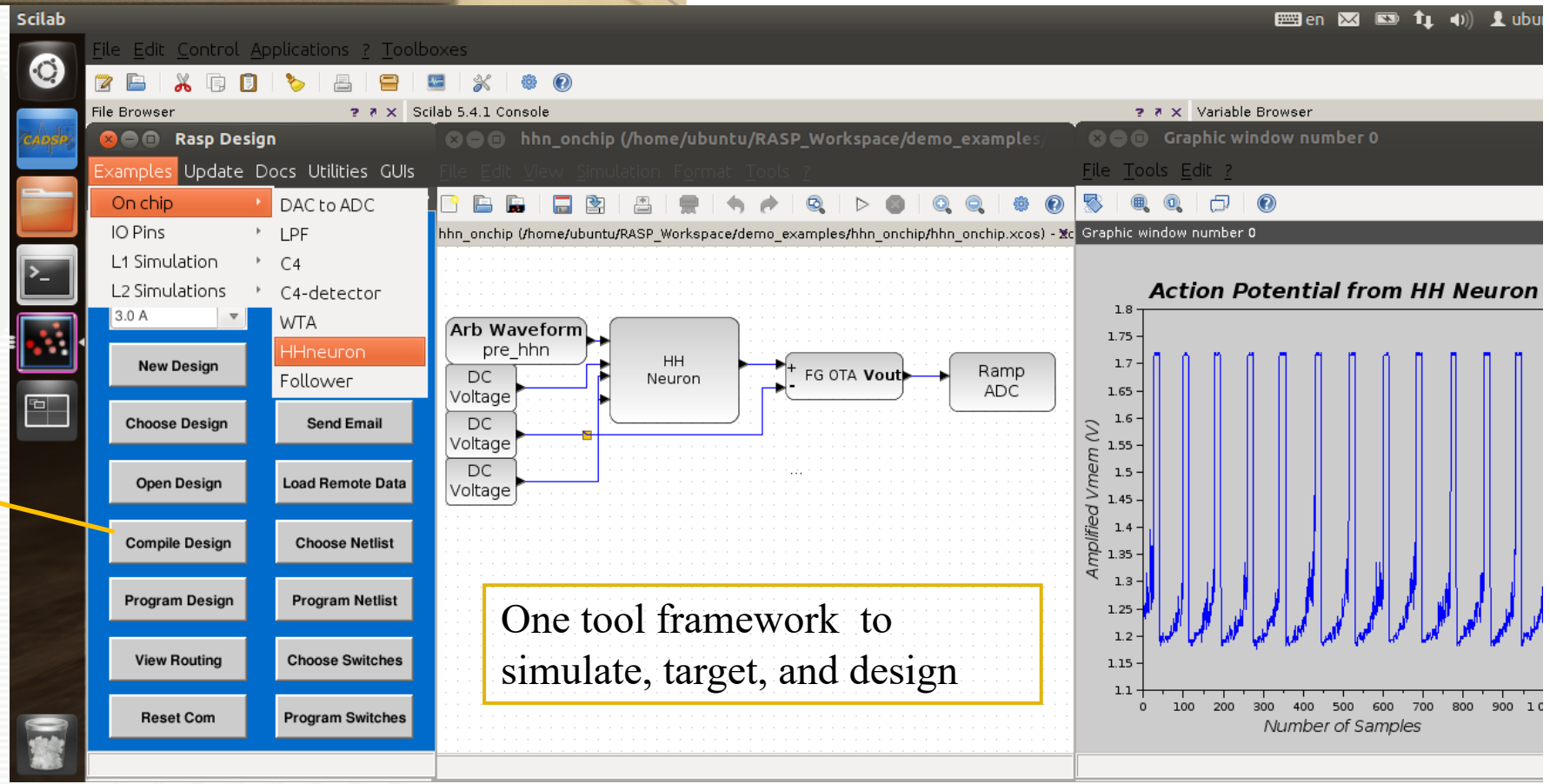


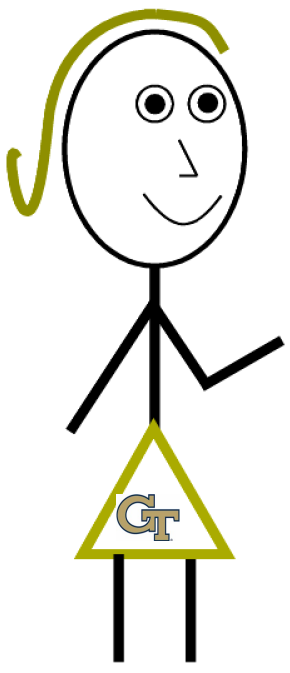




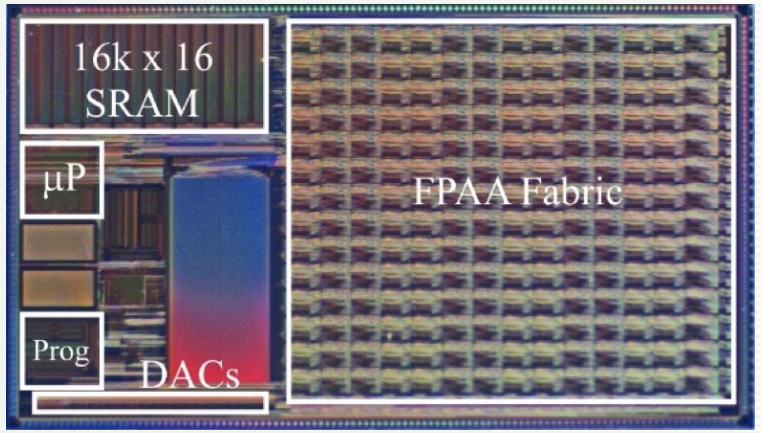
Design
→Hardware
→ Simulation

Used in classes
> 10 years





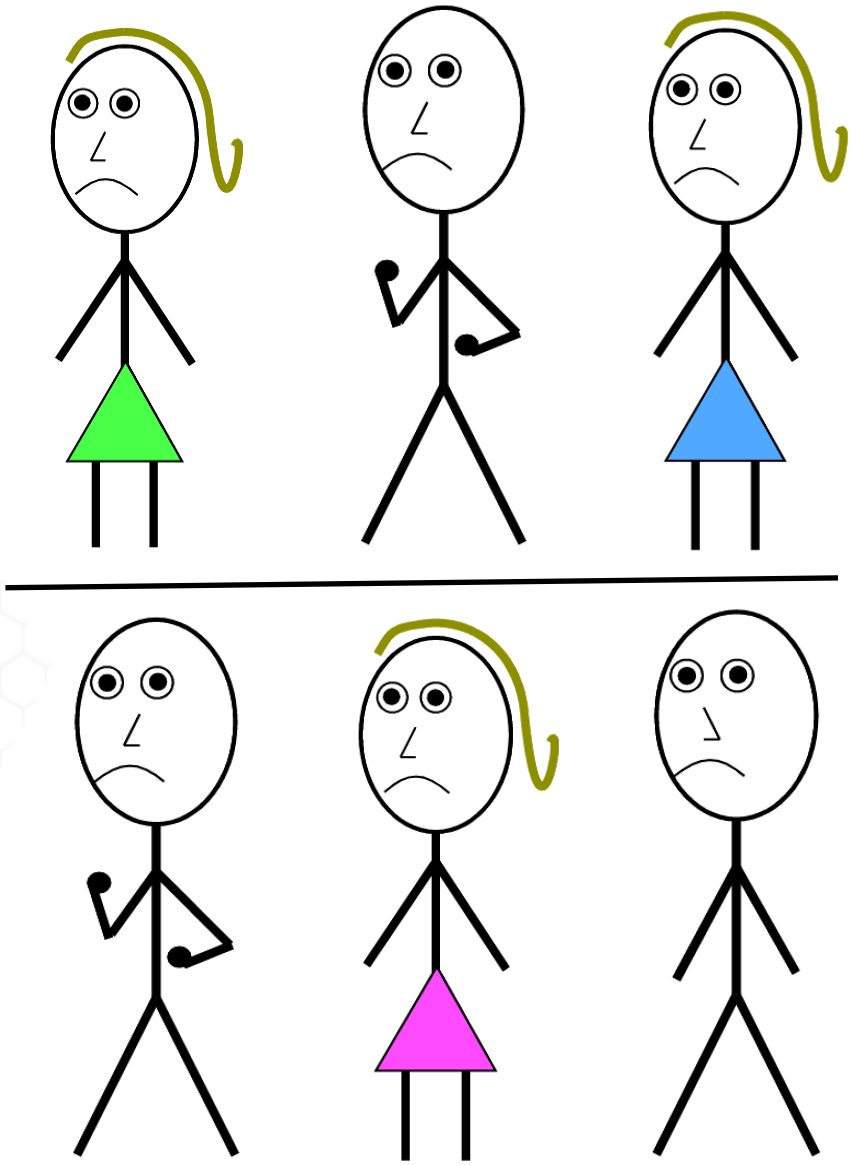
Analog IC Design



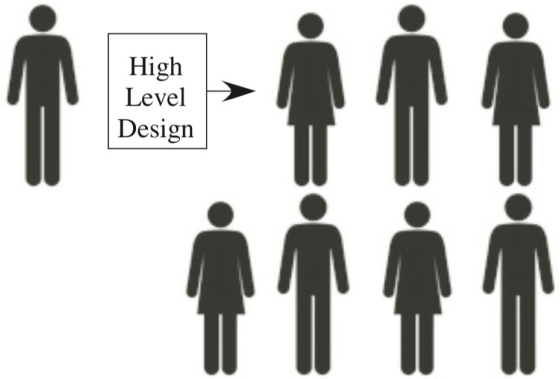
High-Level
Mixed-Signal
Architecture
(e.g. FPAA)

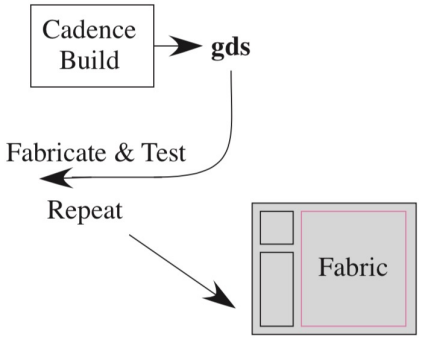
Month 6

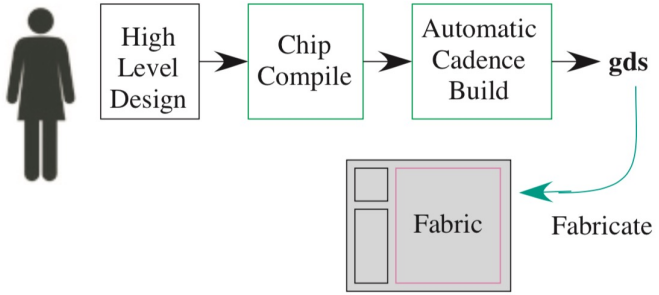
Tapeout (.gds)



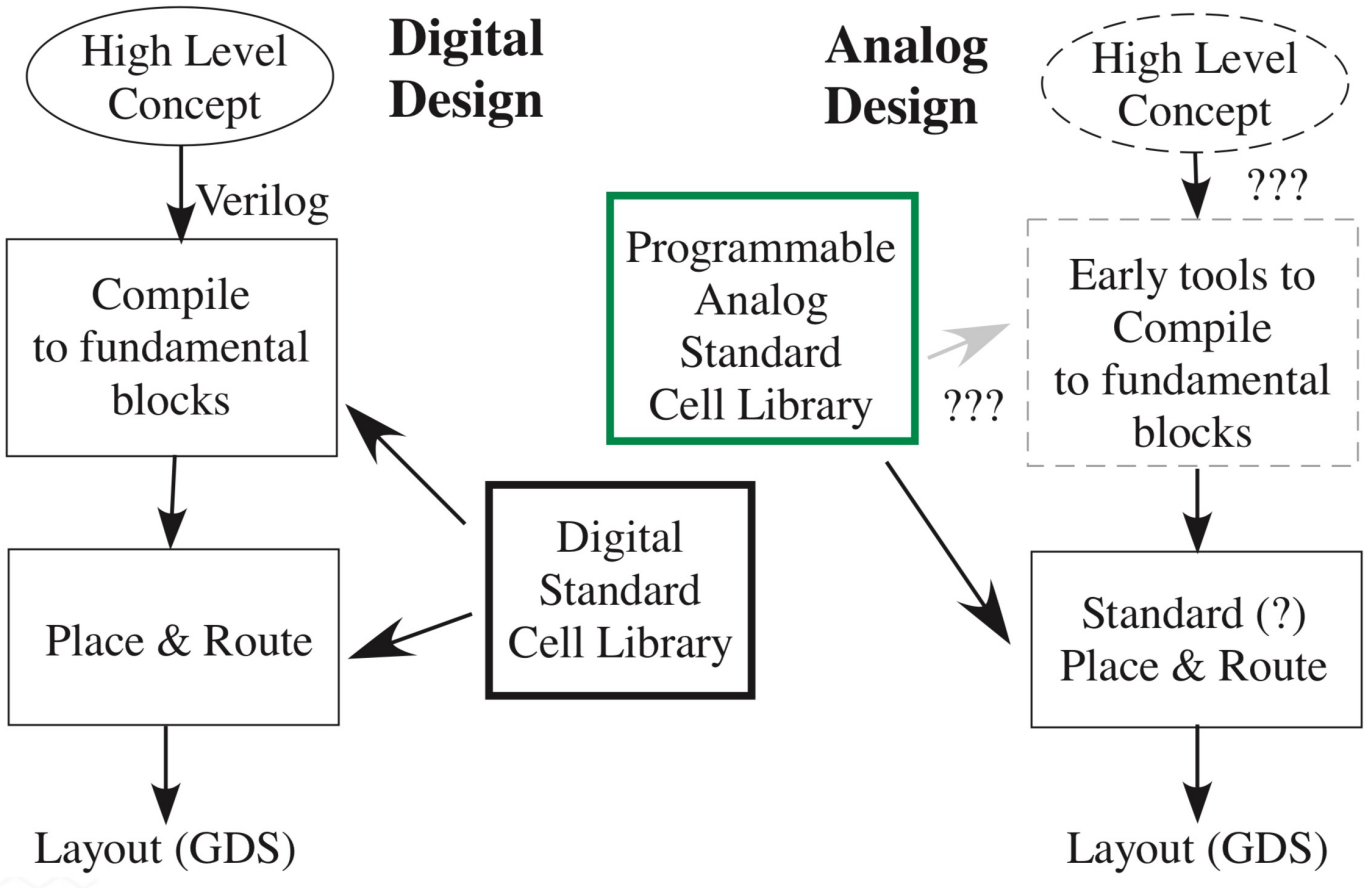
ANALOG & MIXED-SIGNAL SYNTHESIS



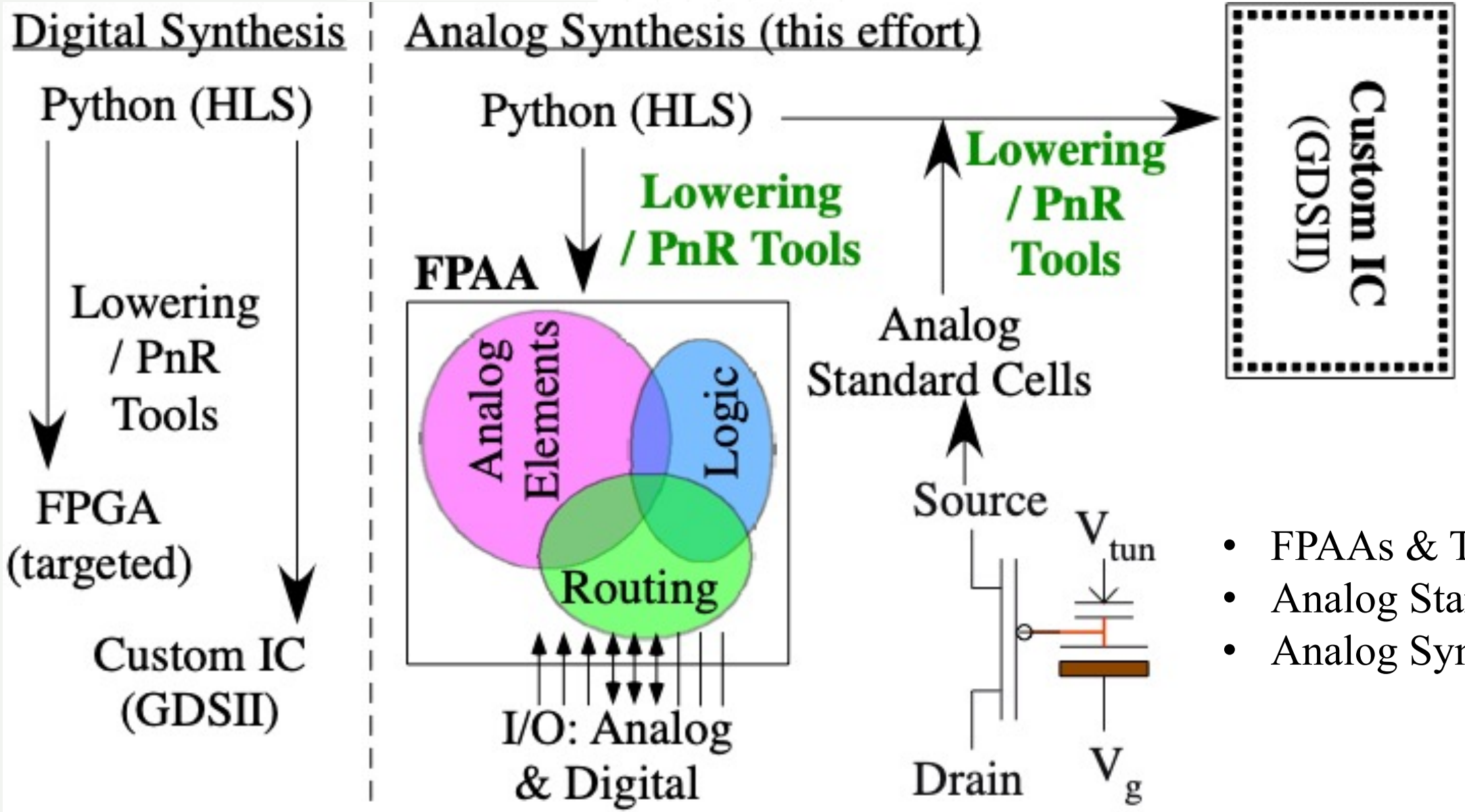




# of Designers	4-8
# of design / test months	9-12 months
# of iterations	1-3



DIGITAL SYNTHESIS →ANALOG SYNTHESIS



- FPAAs & Targeting
- Analog Standard Cells
- Analog Synthesis

NEXT GENERATION ANALOG TOOLS & TARGETING FPAAS & PROGRAMMABLE ANALOG STANDARD CELLS

Analog & Mixed-Signal Standard Cells

65nm CMOS

350nm CMOS

28nm CMOS

Skywater

130nm CMOS

180nm CMOS

16nm CMOS

[CAS I, Hasler, et. al, 2024]
[CICC, Mathews, et. al, 2024]
[ESSERC, Ayyappan 2025]

HLS Application specification

Lowering

Algorithms

Lowering & PnR

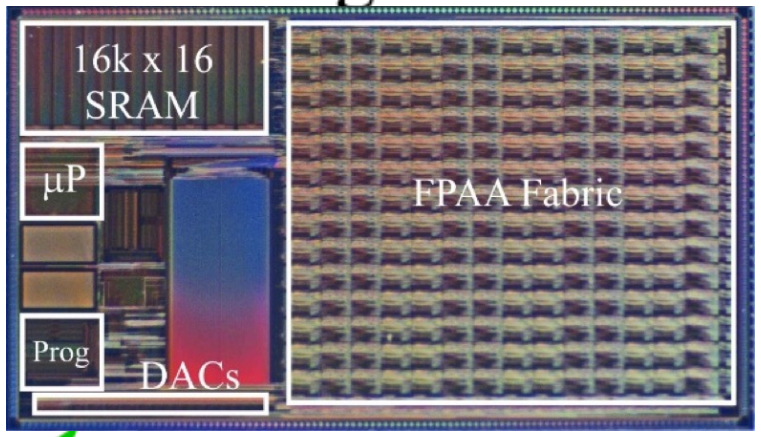
FPAA Tech File

GDSII (New IC)

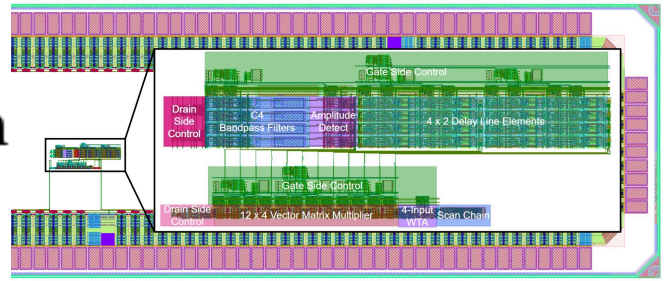
Targeted FPAA

[Ige, Yang, et. al 2023]
[Hasler & Cao, 2024]

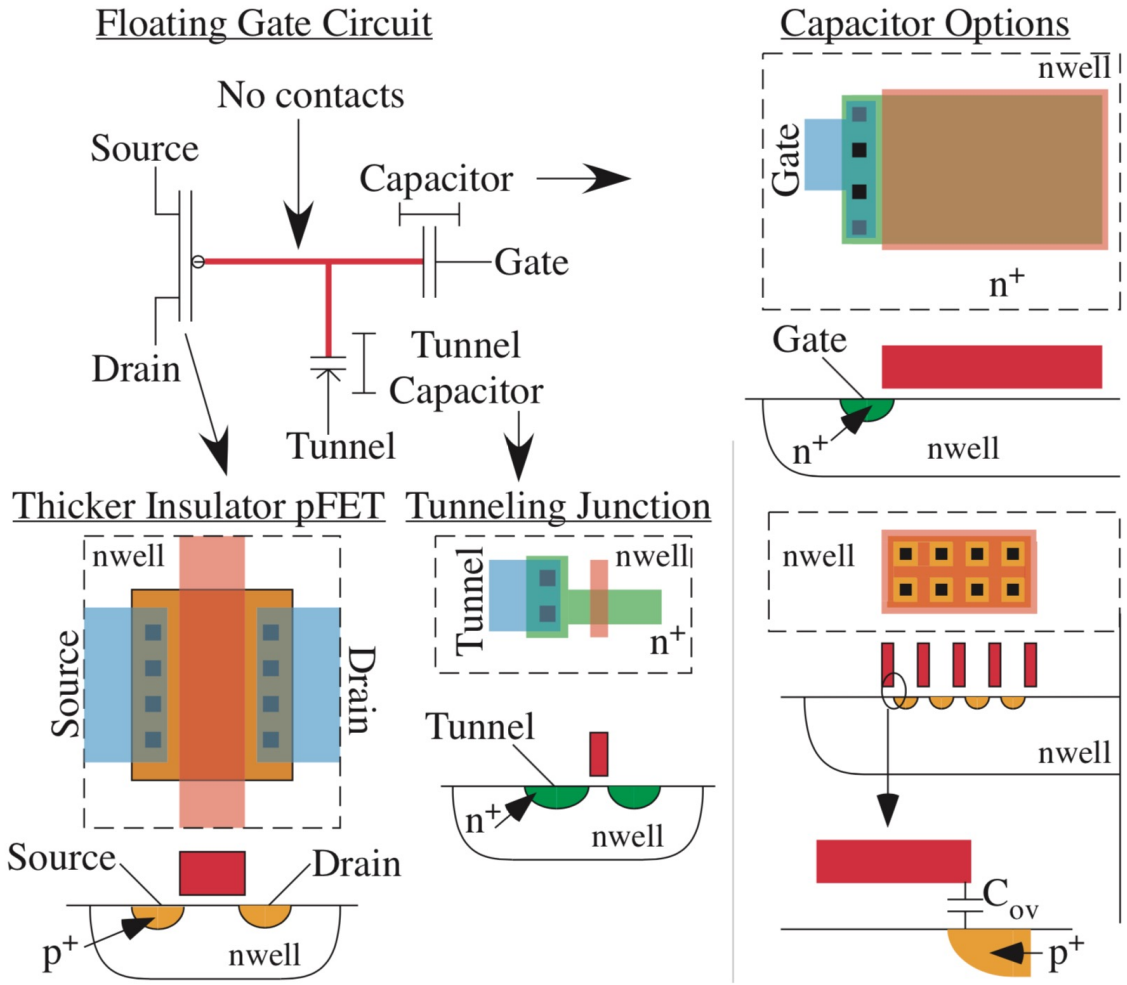
Existing FPAA

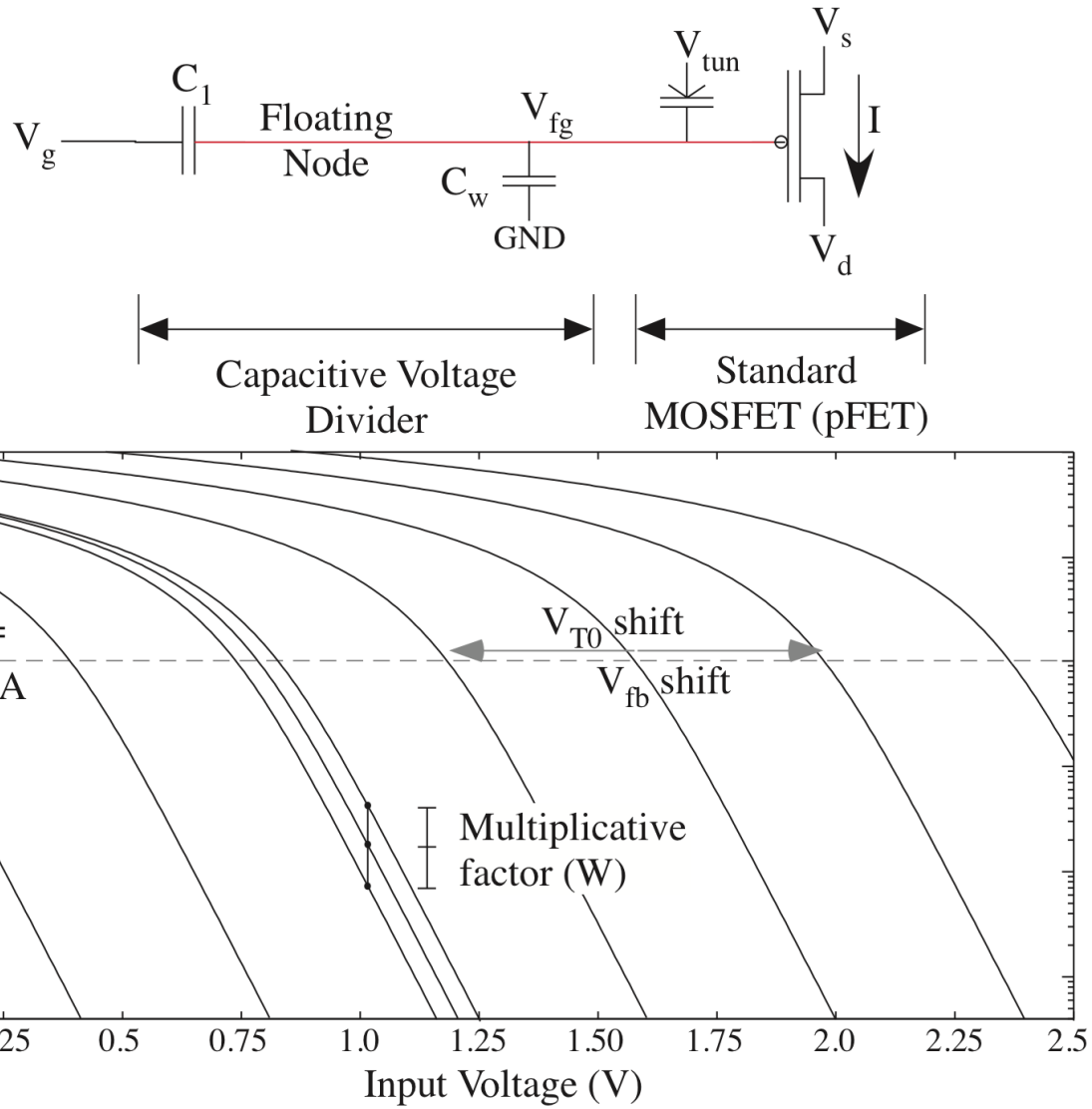


Custom device



ANALOG PROGRAMMABILITY →FLOATING-GATE (FG) CIRCUITS



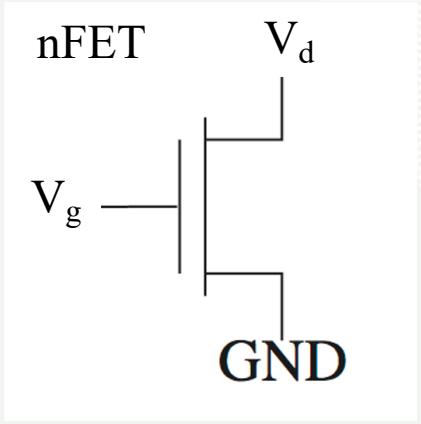


Program FG charge once and holds for part lifetime
(i.e. 10 year lifetime = 10-100μV)

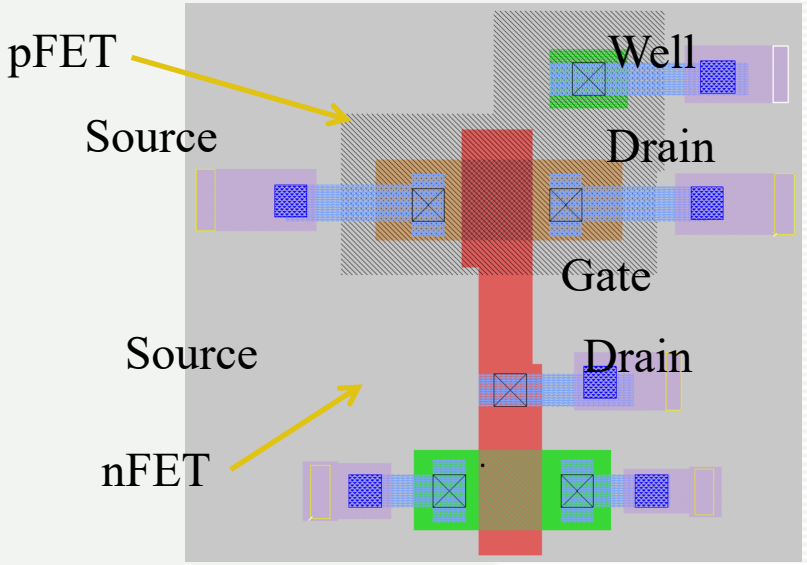
Programmed to 14bit accuracy: e.g. 60μV on 1V supply

Floating-Gate (FG) circuit techniques enables direct solution for mismatch (V_{T0}) (Standard CMOS)

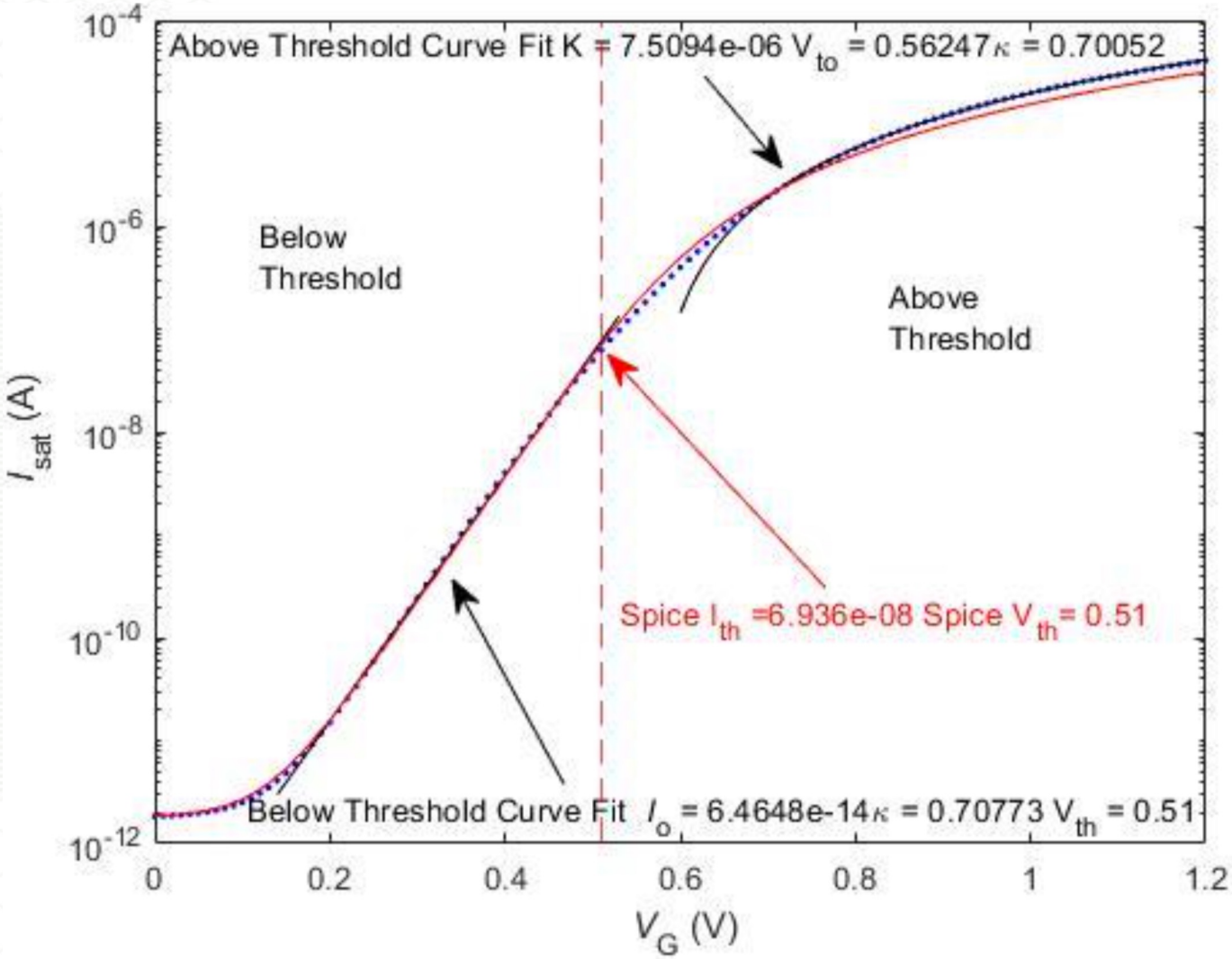
MOSFET TRANSISTORS ON AN IC



Magic 130nm Layout

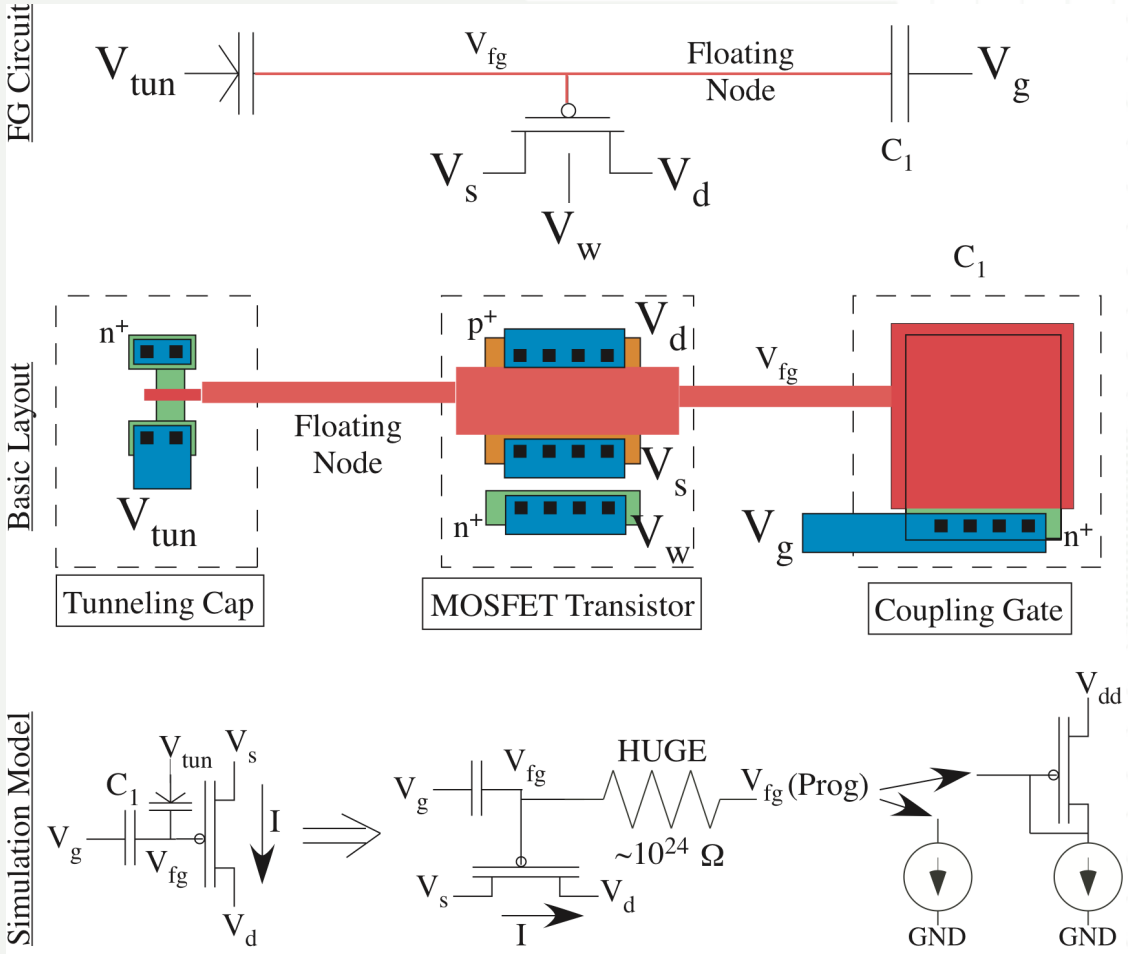


Skywater 130nm nFET Data

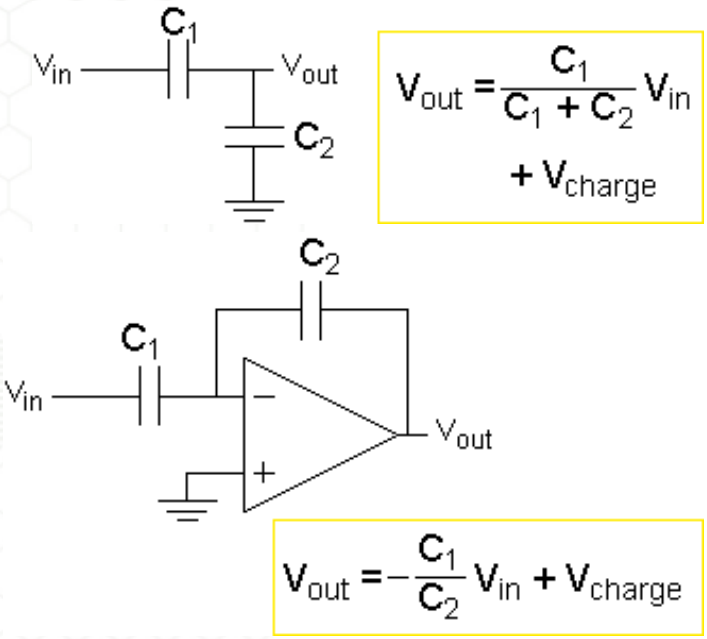


$$I = I_{th} e^{(\kappa(V_g - V_{T0}) - V_s + \sigma V_d)/U_T}$$

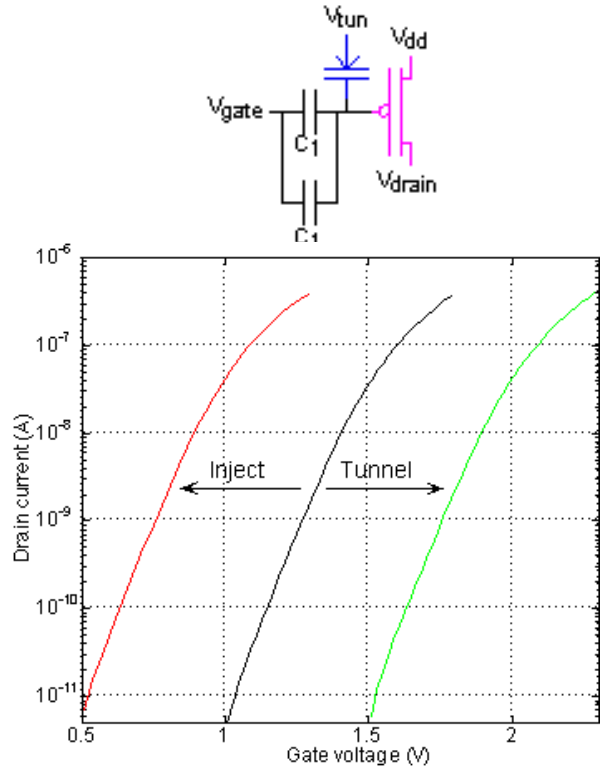
FLOATING-GATE (FG) CIRCUITS



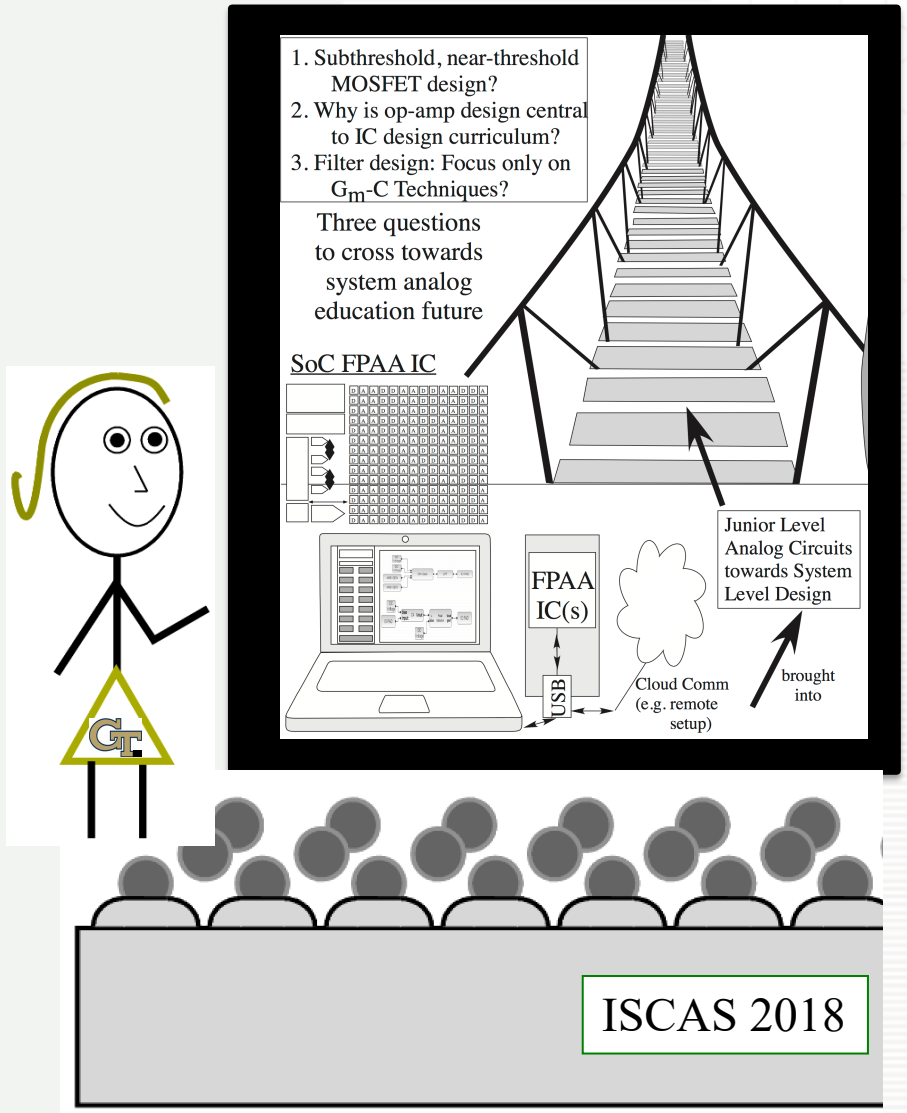
Capacitor-Based Circuits



Program Charge



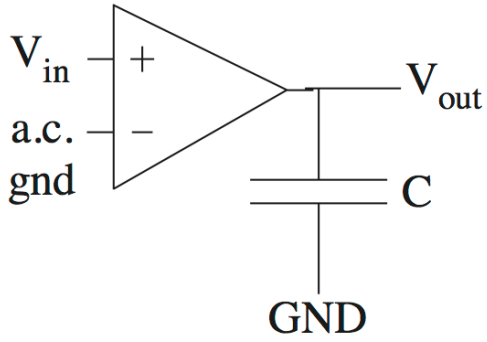
- Resistors and Inductors define the circuit dynamics
- Capacitors are the natural elements on silicon ICs
- Decrease FG Q → hot-electron injection



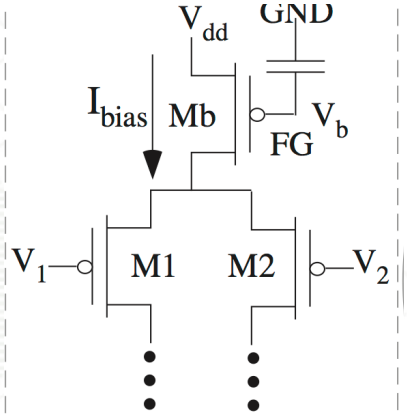
Transconductance-C design

Advantages:

1. Highest bandwidth for given energy
2. Lowest Noise for given I_{bias}

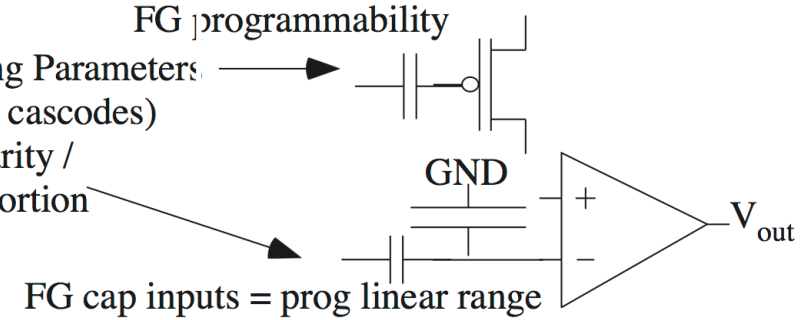


Needs Programmability & Linearity & Offset compensation



Issues:

1. Setting Parameters (bias, cascodes)
2. Linearity / Distortion



Abstract Analog Circuits

Prog accuracy not by feedback

IC design transformed by programmability →

- Prog. FG biasing
- FG TA inputs
- GmC, SubV_T Design (**no bank of passives**)
- Configurable Design

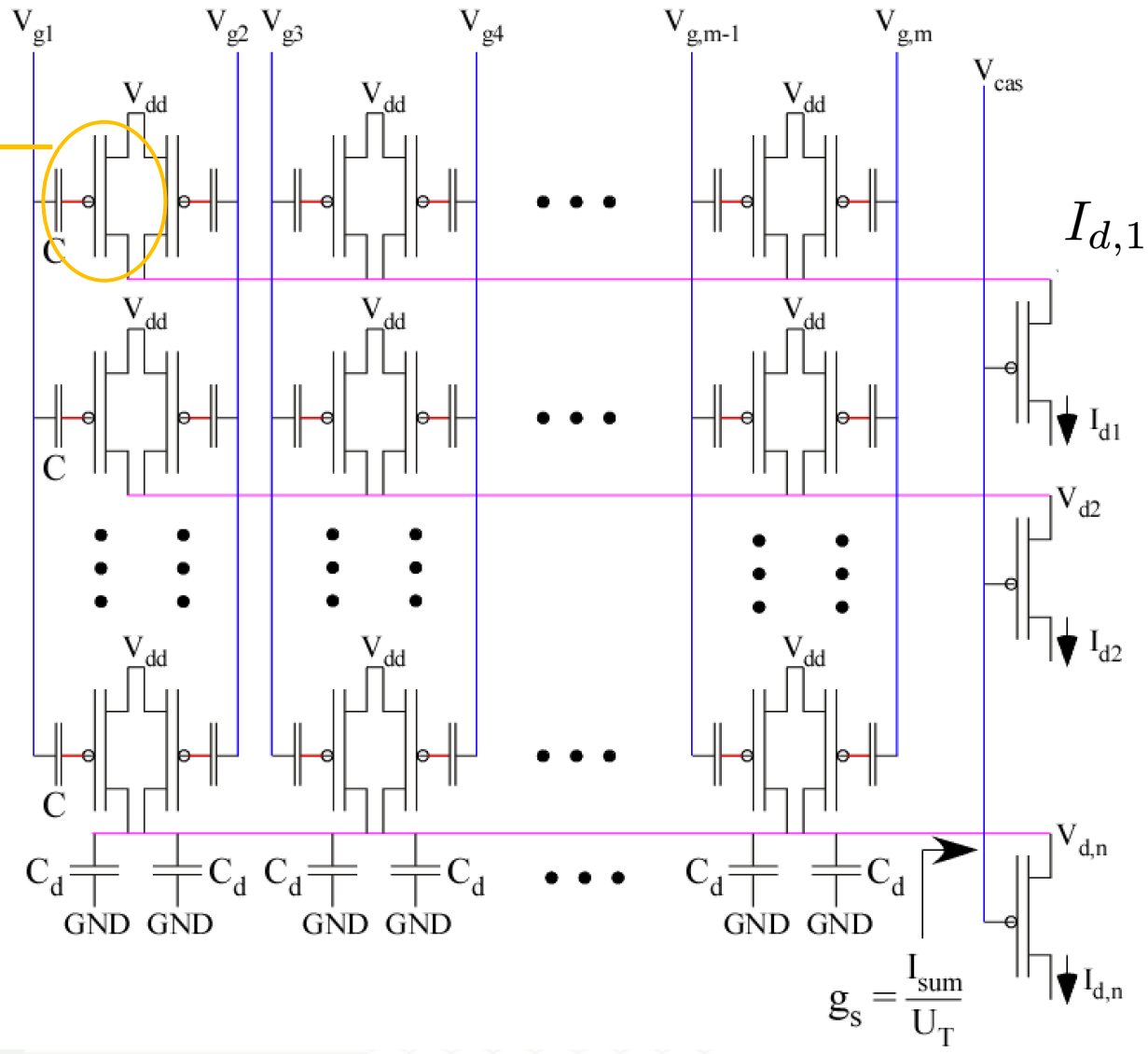
Transistor Centric Design → Few (to 0) passives other than capacitors

Subthreshold Centric Design → Wide tuning range, low-power / energy

PROGRAMMABILITY → SIGNAL PROCESSING

Banks of Programmable
(Transconductance-Capacitor) Filters

$$I \approx g_{m1} V_{g1} = g_m W_{1,1} V_{g1}$$



Single-Transistor
Learning Synapses
(1994)

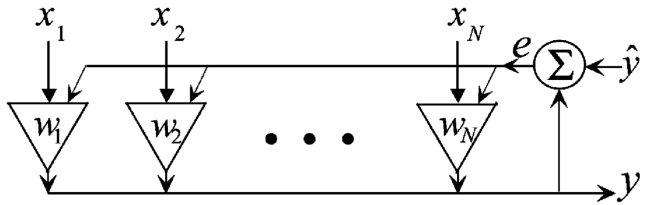
Vector-Matrix Multiplication

$$I_{d,1} = I_{sum,1} \approx g_m \sum_{k=1}^{m} g_m W_{1,k} V_{g,k}$$

$$\mathbf{i}_d = g_m \mathbf{W} \mathbf{v}_g$$

Vector Matrix Vector

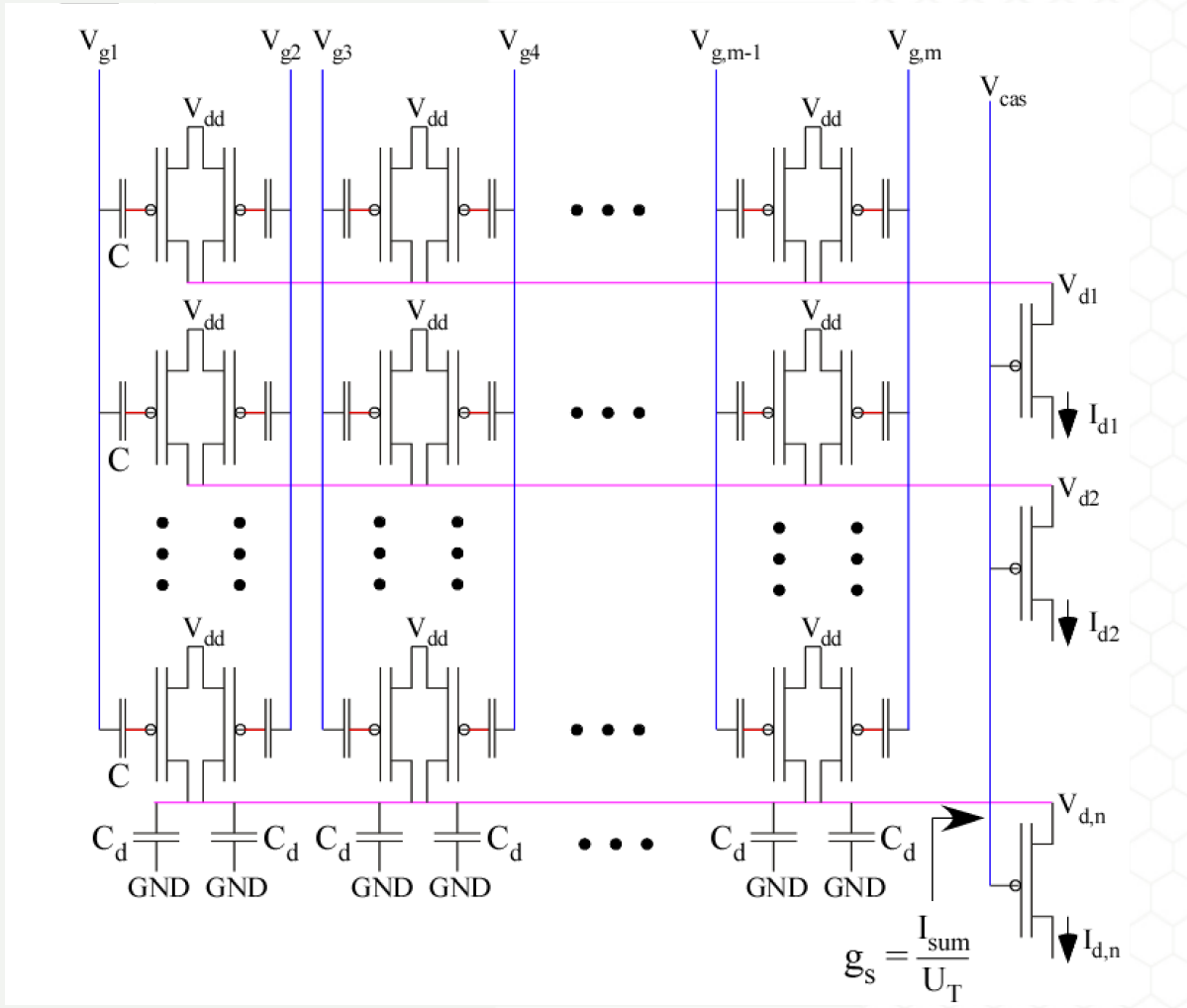
Adaptive Filters [Dugger & Hasler, 2004-2005]



Continuous Adaptation

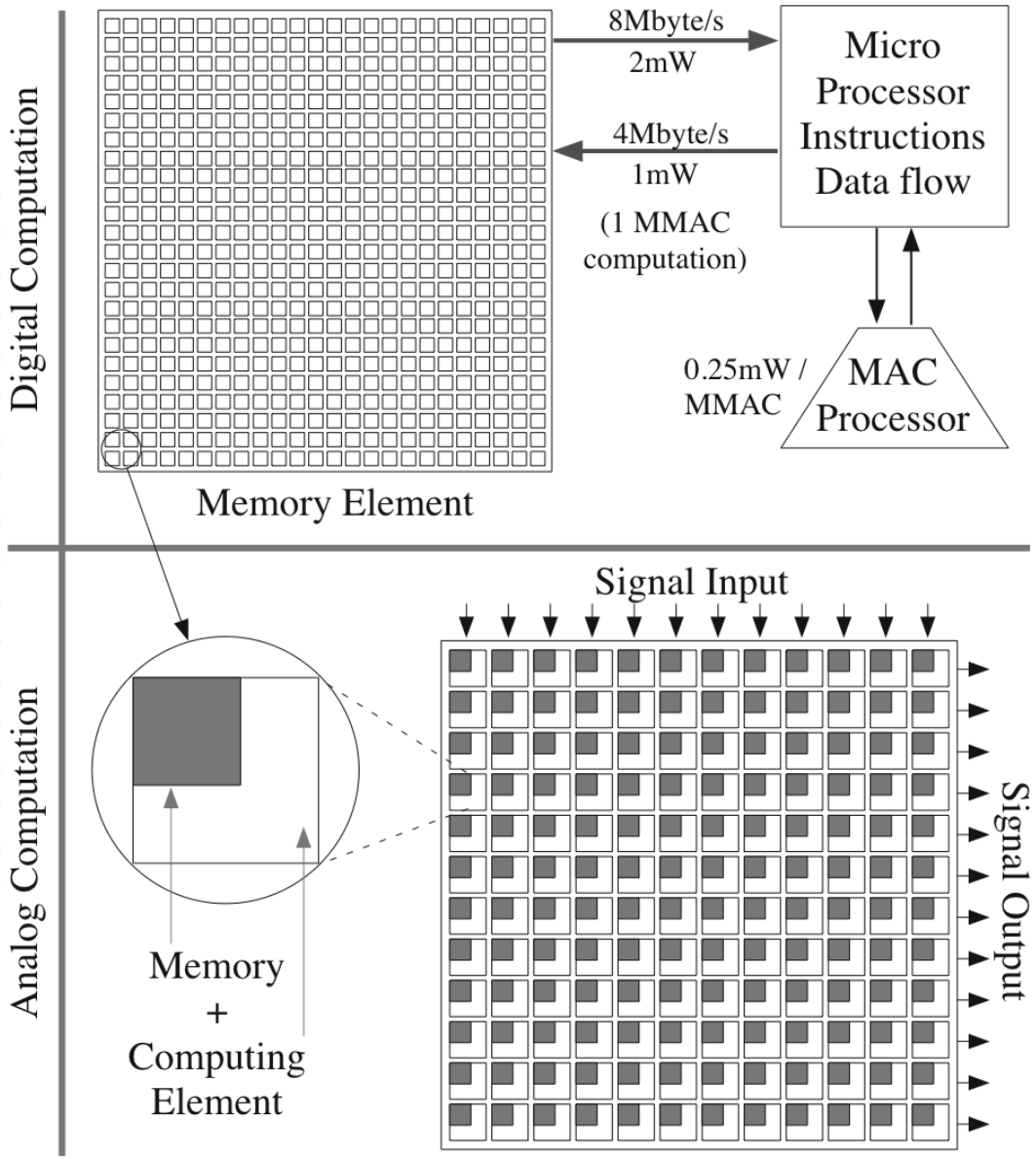
→ Hand-Developed Energy Efficient Analog ASIC ICs

FG ARRAY → COMPUTE IN MEMORY

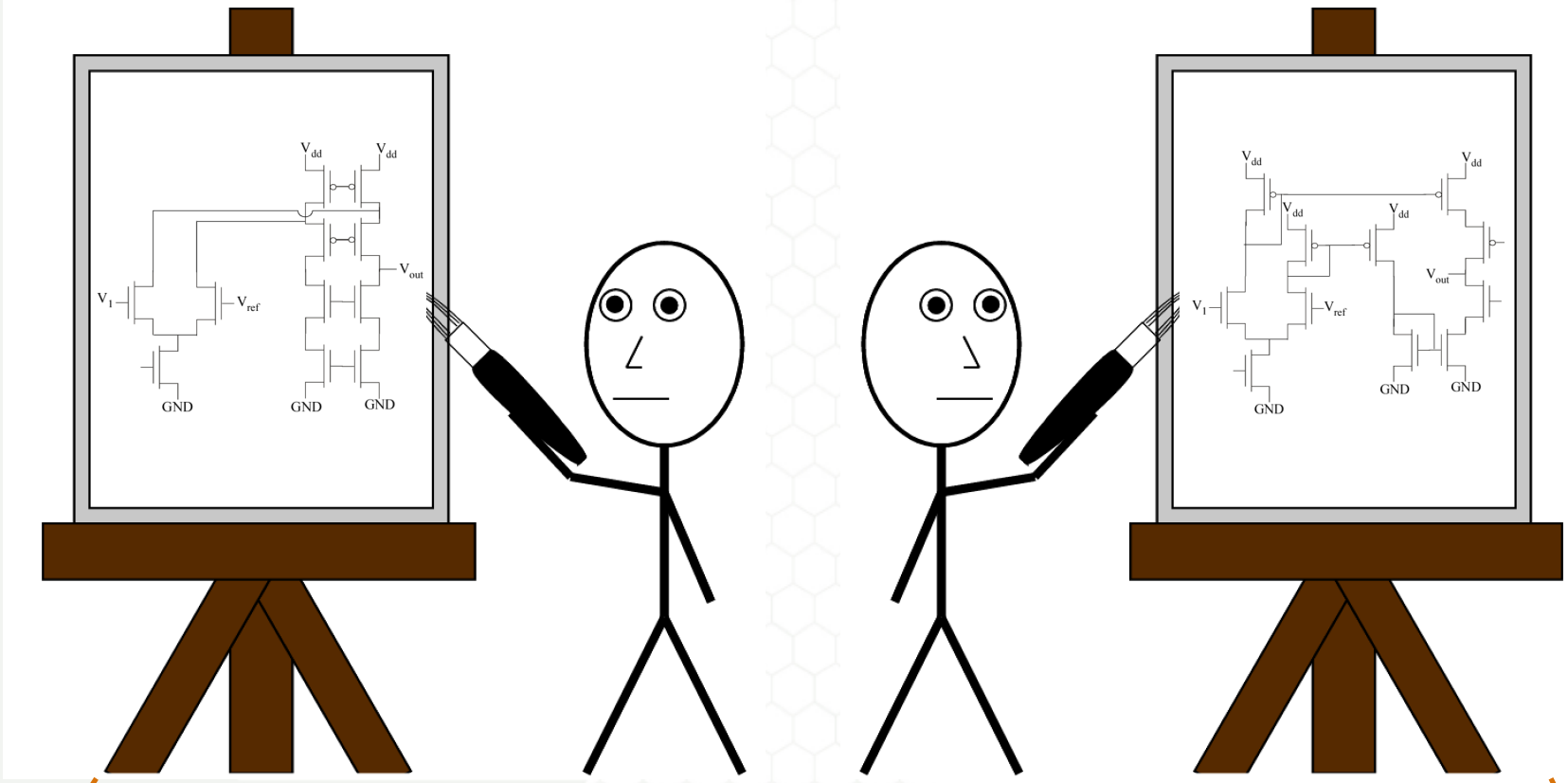


→ Hand-Developed Energy Efficient Analog ASIC ICs

Compute in Memory (2001)

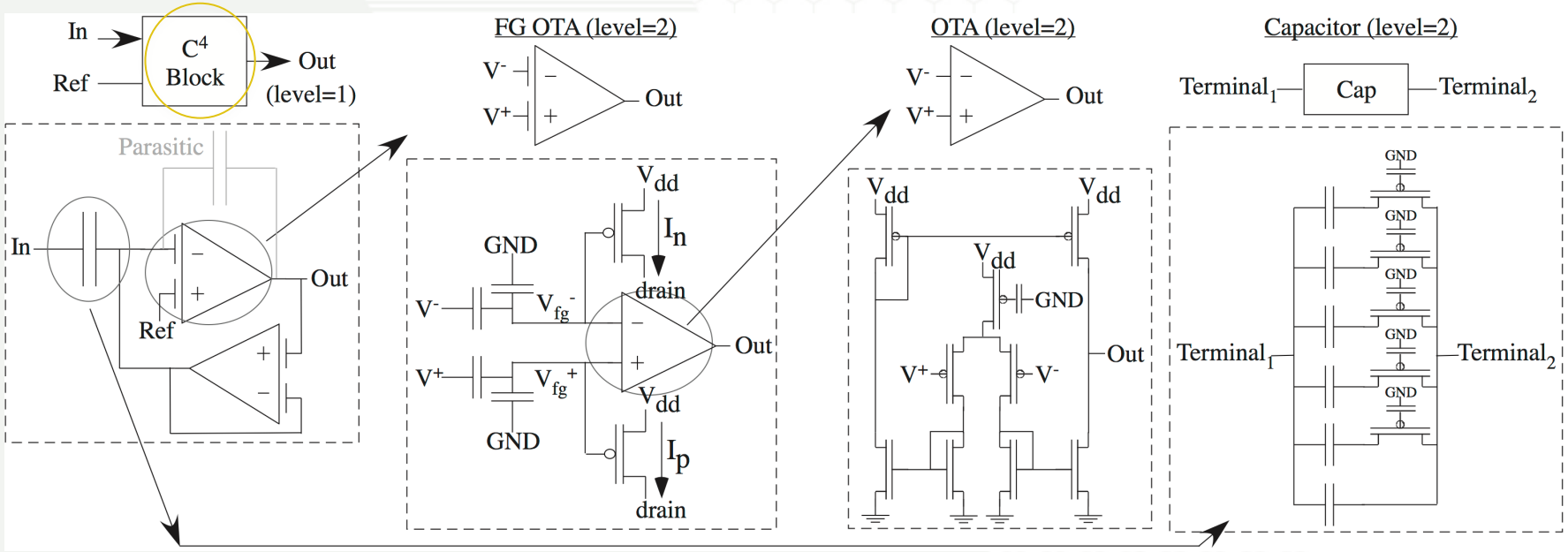


ANALOG ABSTRACTION

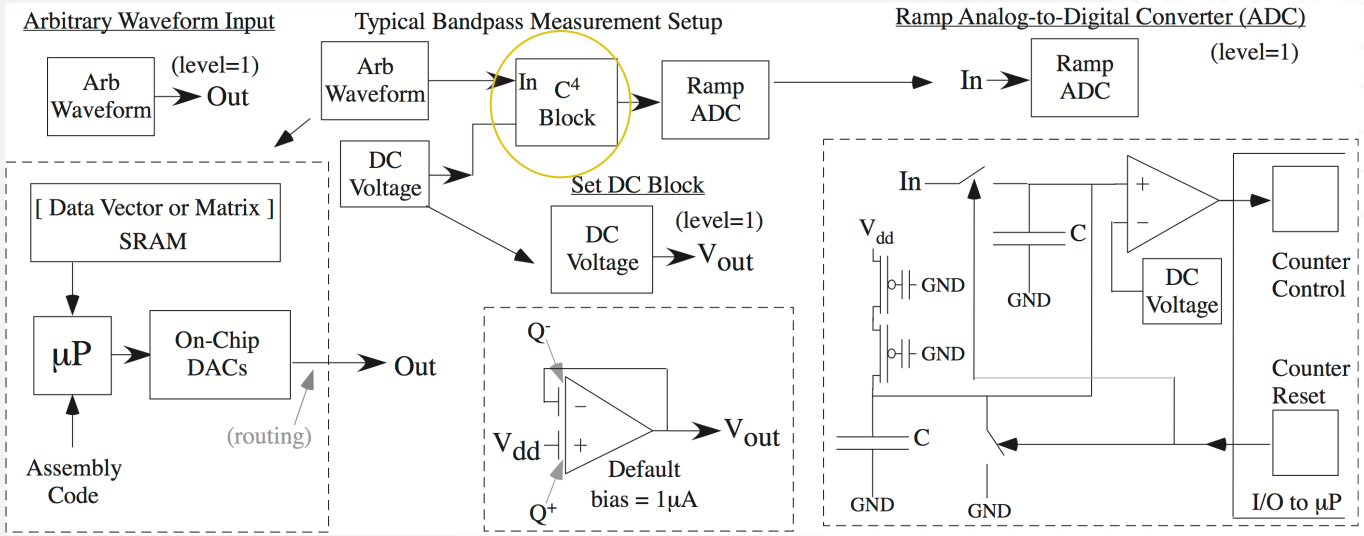


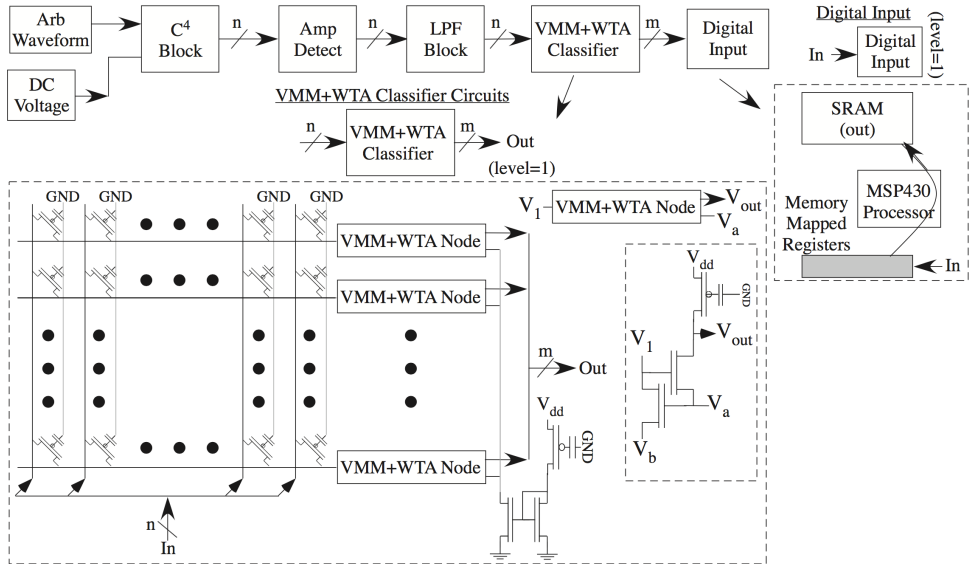
ABSTRACTION IS ESSENTIAL FOR ANALOG SYSTEM DESIGN

C⁴ BPF Abstraction

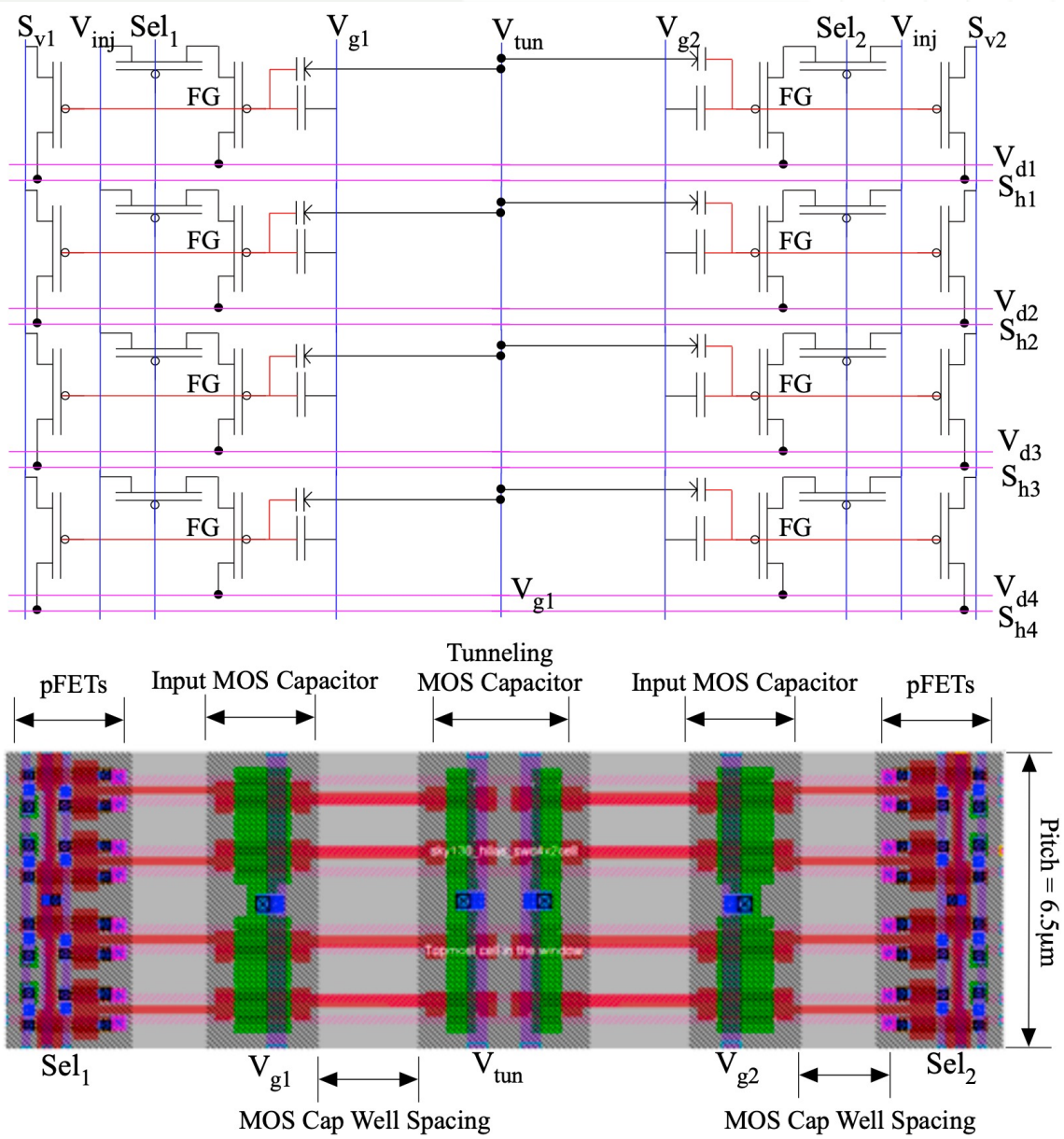


C⁴ Test Bench Example





ANALOG STANDARD CELLS & FG PROGRAMMABILITY

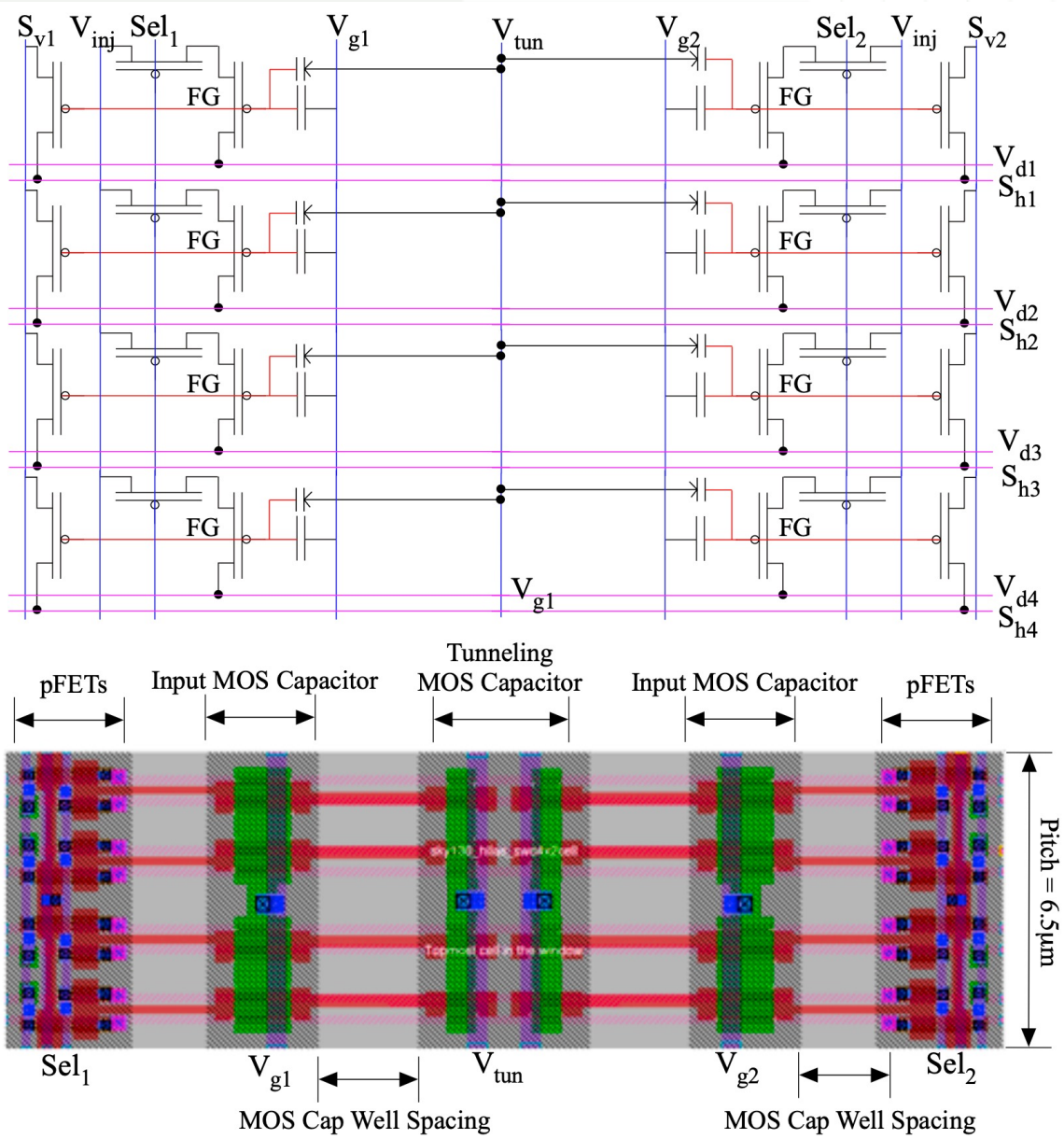


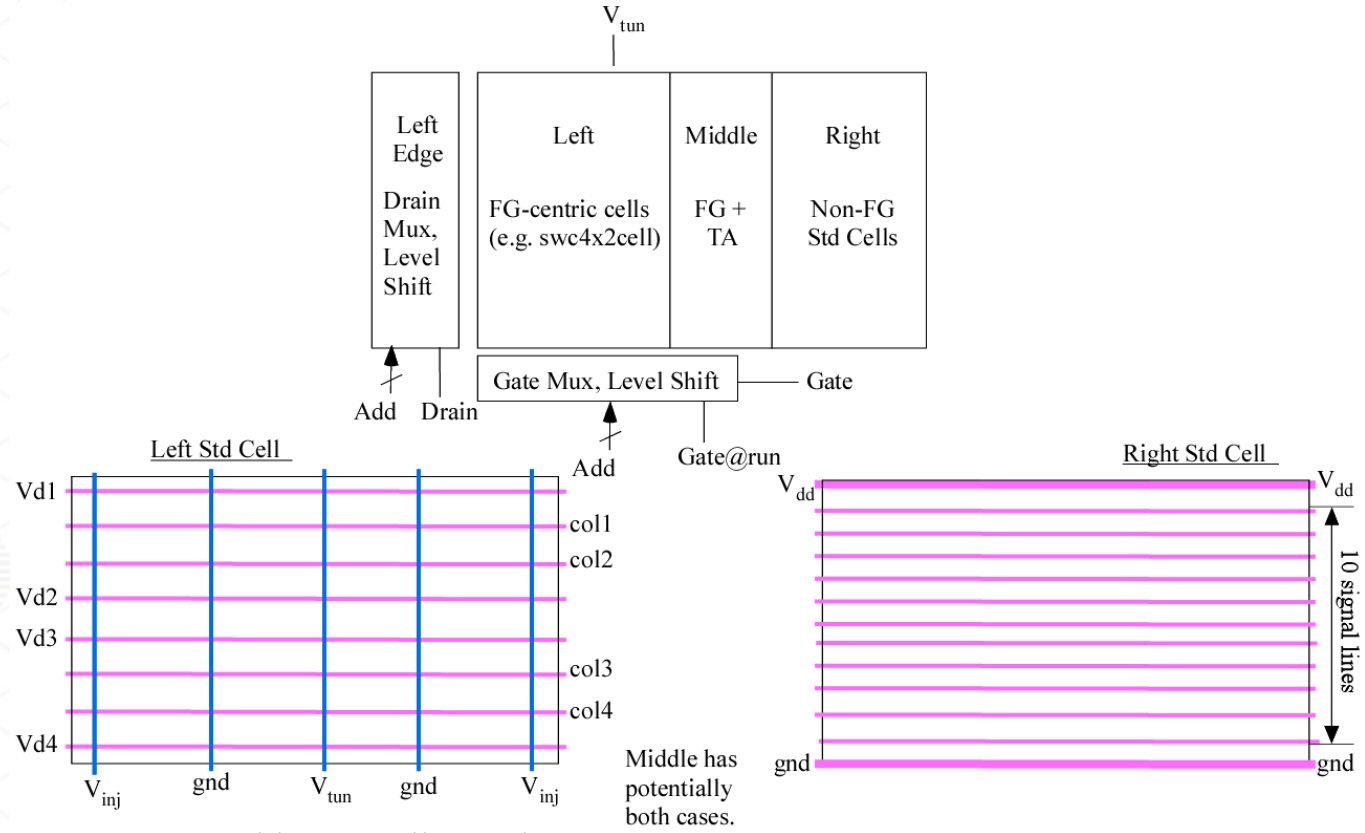
Tight FG arrays are essential everywhere

SWC 4x2 cell sets the pitch

Tight FG arrays don't fit a typical standard cell approach

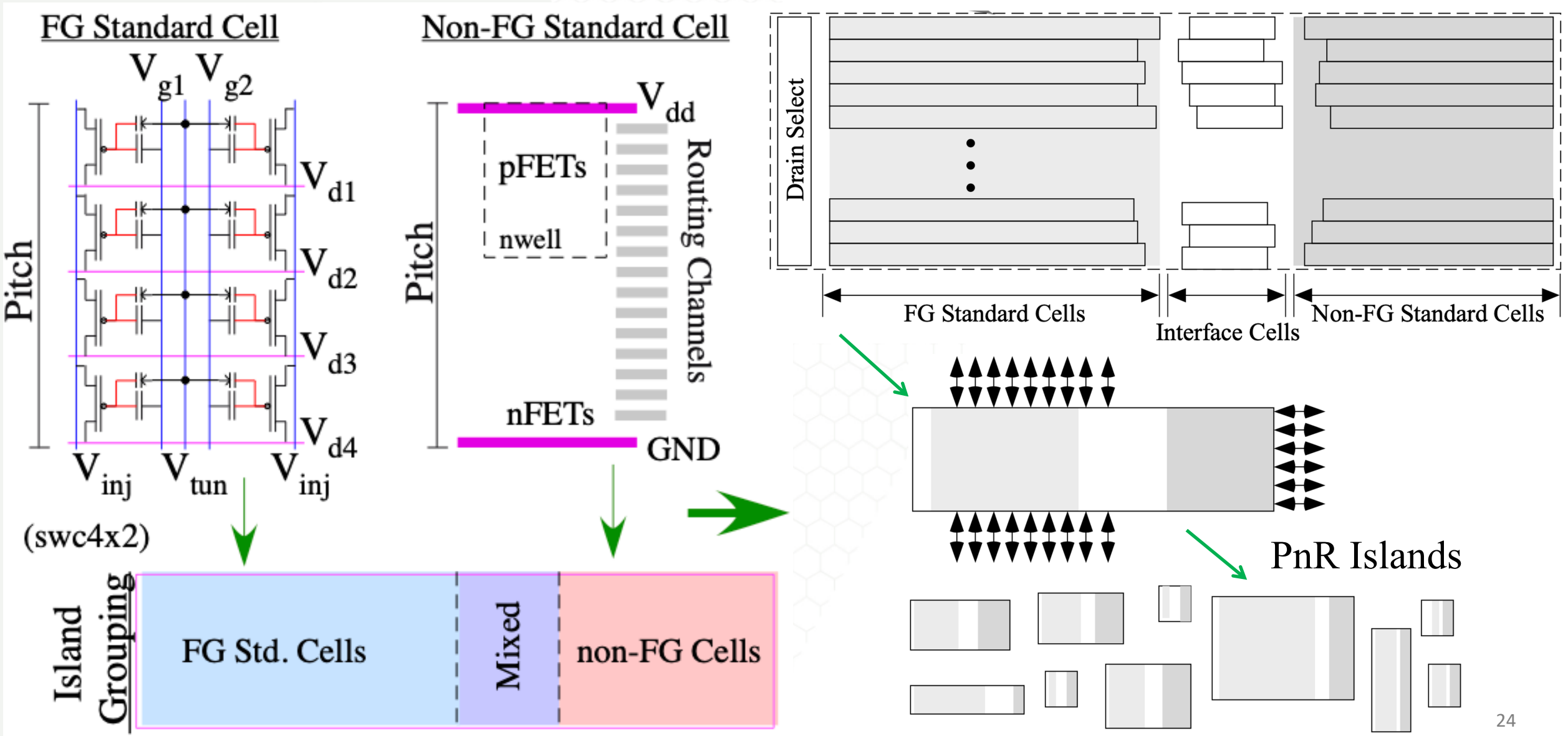
ANALOG STANDARD CELLS & FG PROGRAMMABILITY





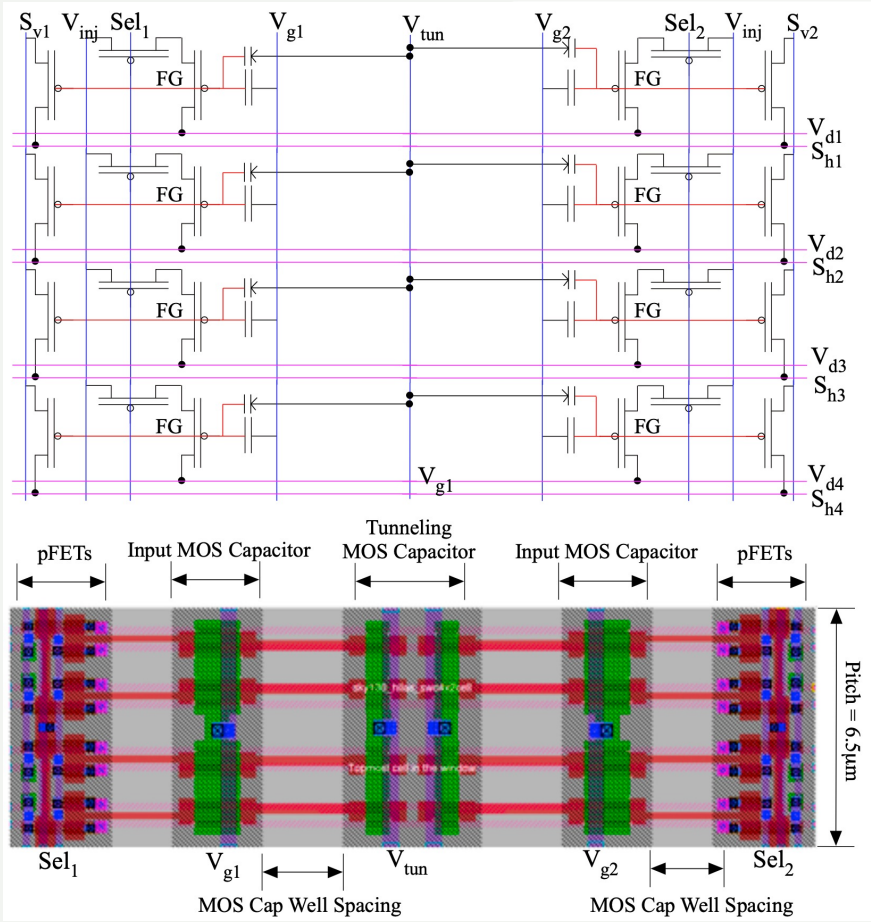
Programmable FG Cells requires different routing (vertical V_{tun}, V_{inj}, V_{dd})

PROGRAMMABLE ANALOG STANDARD CELLS



130NM STANDARD CELL LIBRARY

Std Cell libraries fabricated in 350nm, 180nm ,130nm, 65nm, 28nm, 16nm CMOS



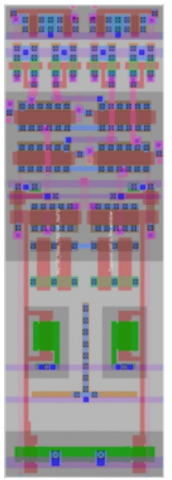
Analog Cells:

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

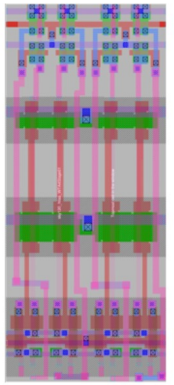
FG Cells:

Cell Type	Variations	Name	Width
Crossbar Cell (Same C_T)	4×2 Cell	swc4x2cell	20.12μm
	4×1 Bias	swc4x1BiasCell	10.11μm
	4×1 cell	swc4x1cell	10.08μm
FG Gate cell (Same C_T)	2 x 1 FG Biases	FGBias2x1cell	11.53μm
	2 x1 FG Transistors	FGtrans2x1cell	11.52μm
	2 x1 FG Trans, small input C	FGBiasWeakGate2x2cell	11.53μm
FG Characterization cell	Tun, pFET (inj), Capacitors	FGcharacterization01	29.95μm

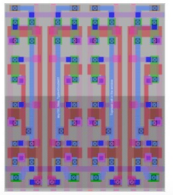
TA2Cell noFG



WTA4Stage01

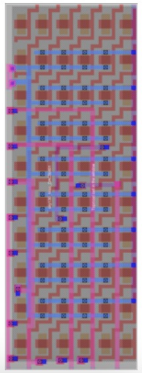


Tgate4Double01

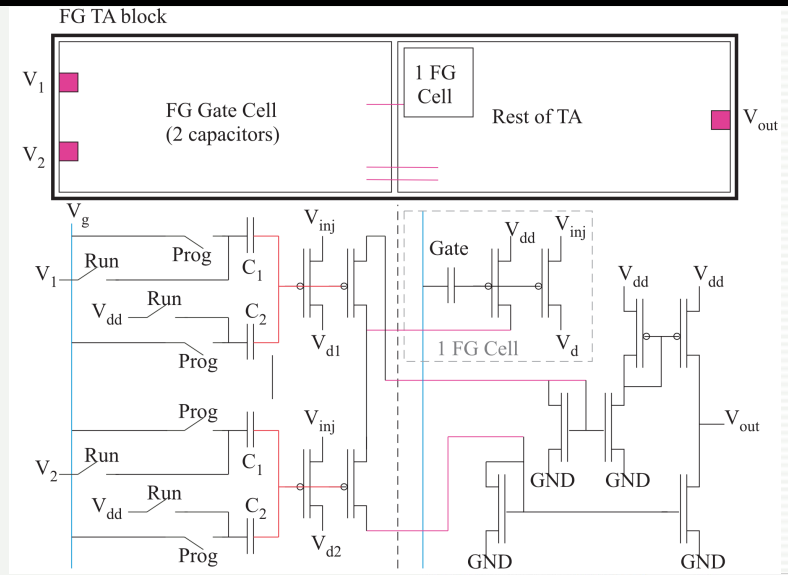


capacitorArray01

DAC5bit01

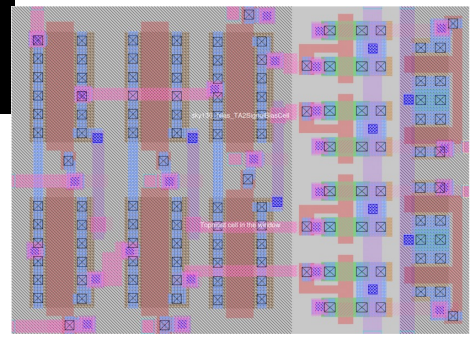


TRANSCONDUCTANCE AMPLIFIERS (TA)

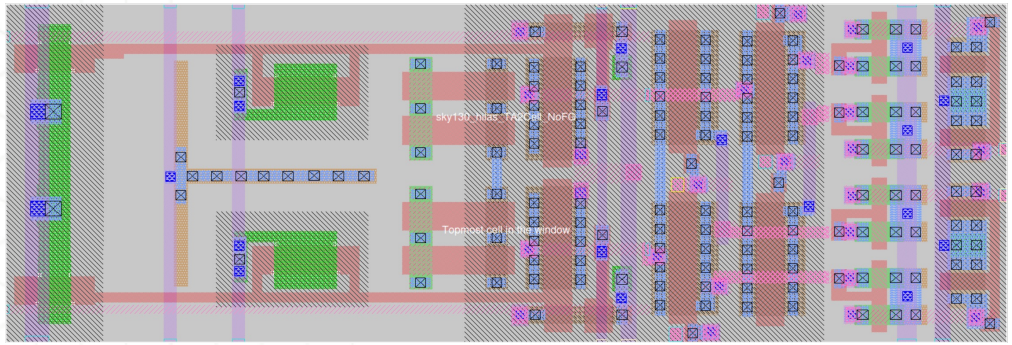


Pitch and Width Matched to swc4x2cell

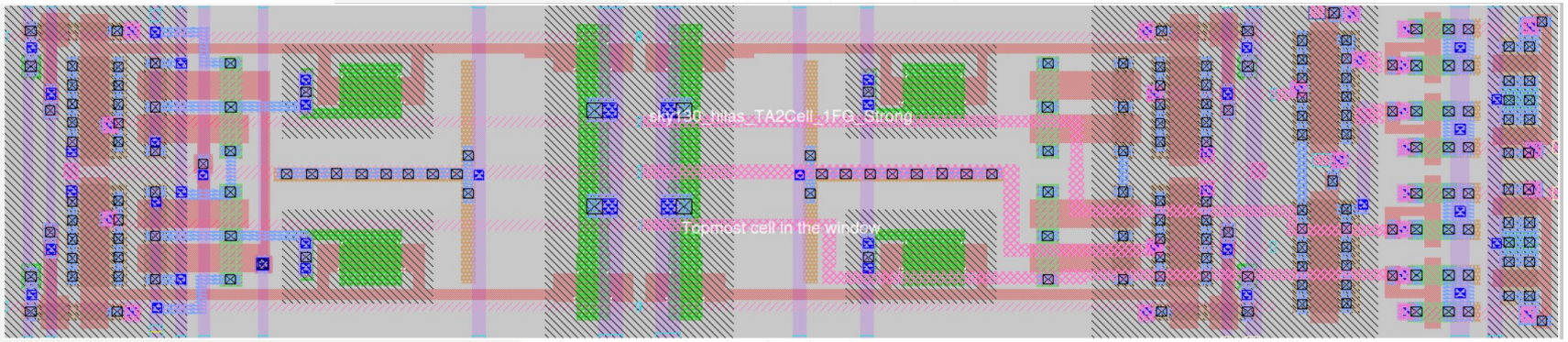
TA2SignalBiasCell



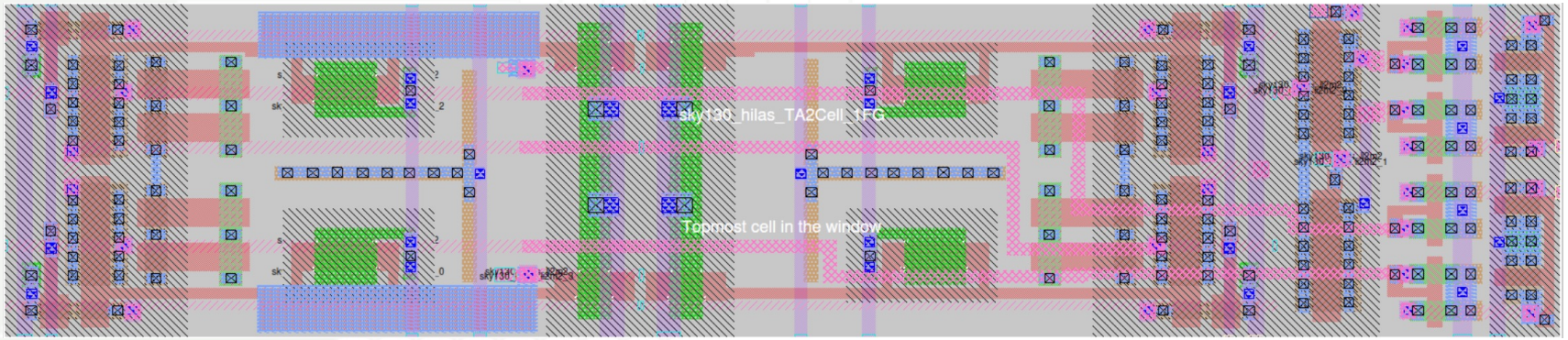
TA2Cell_NoFG



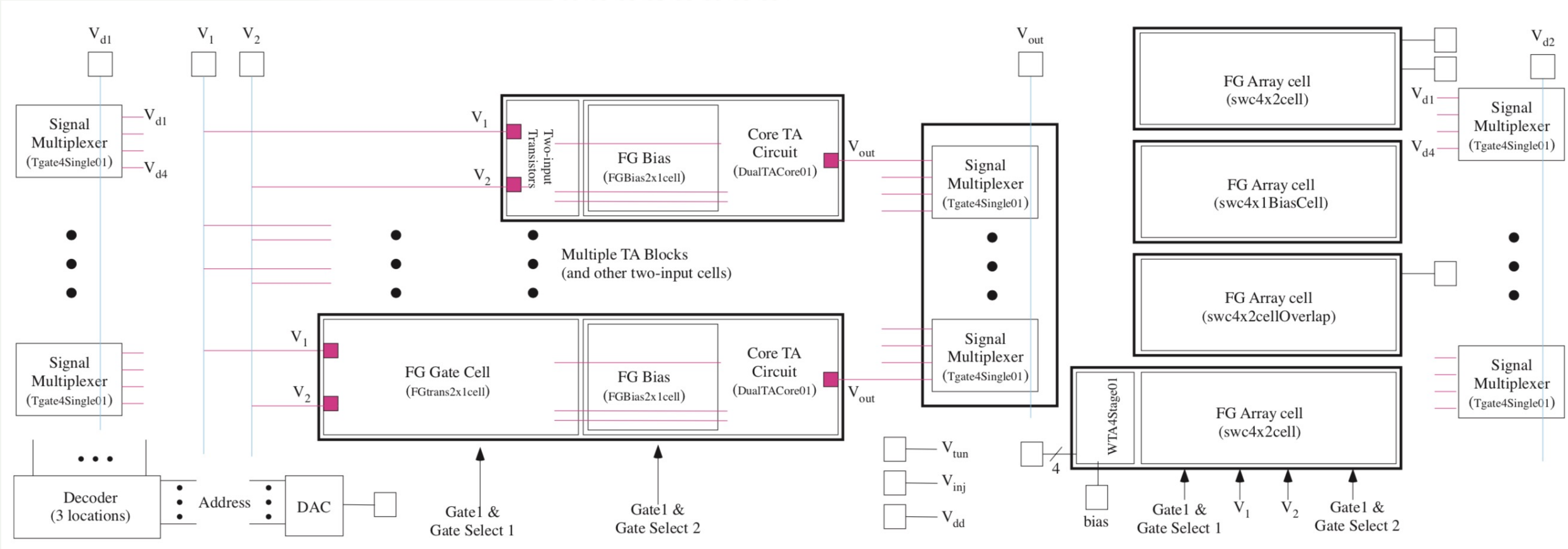
TA2Cell_1FG_Strong



TA2Cell_1FG

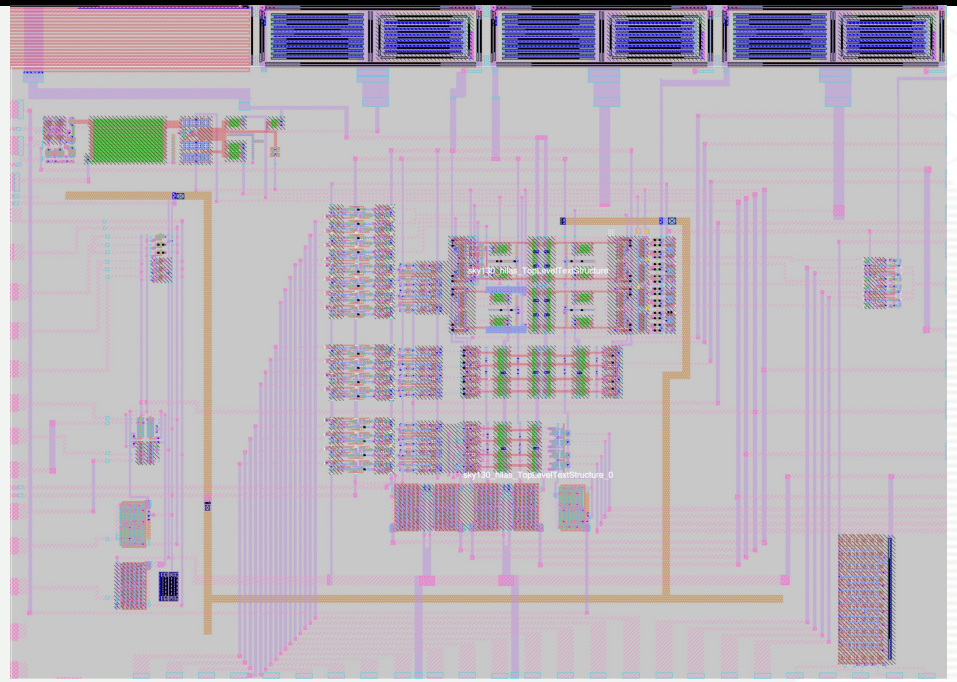


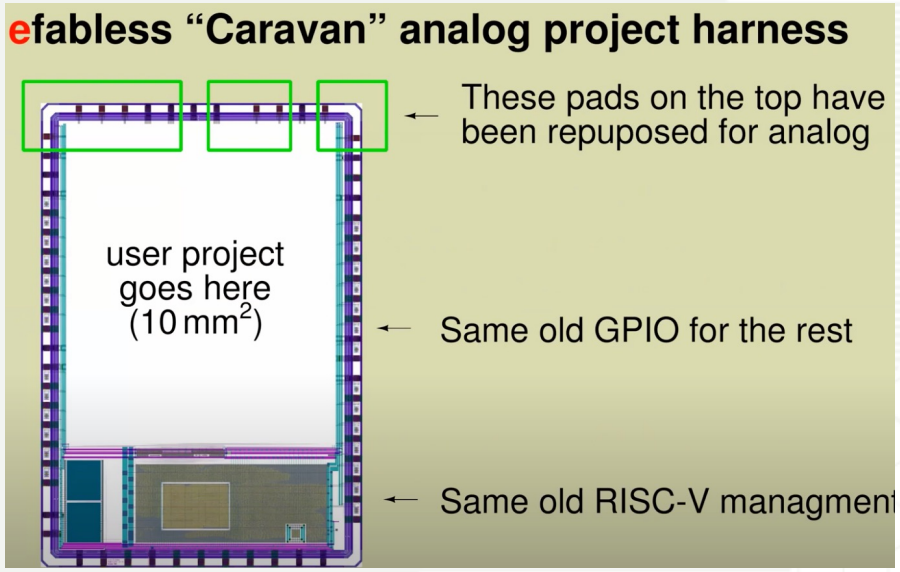
130NM STANDARD CELL TEST STRUCTURE

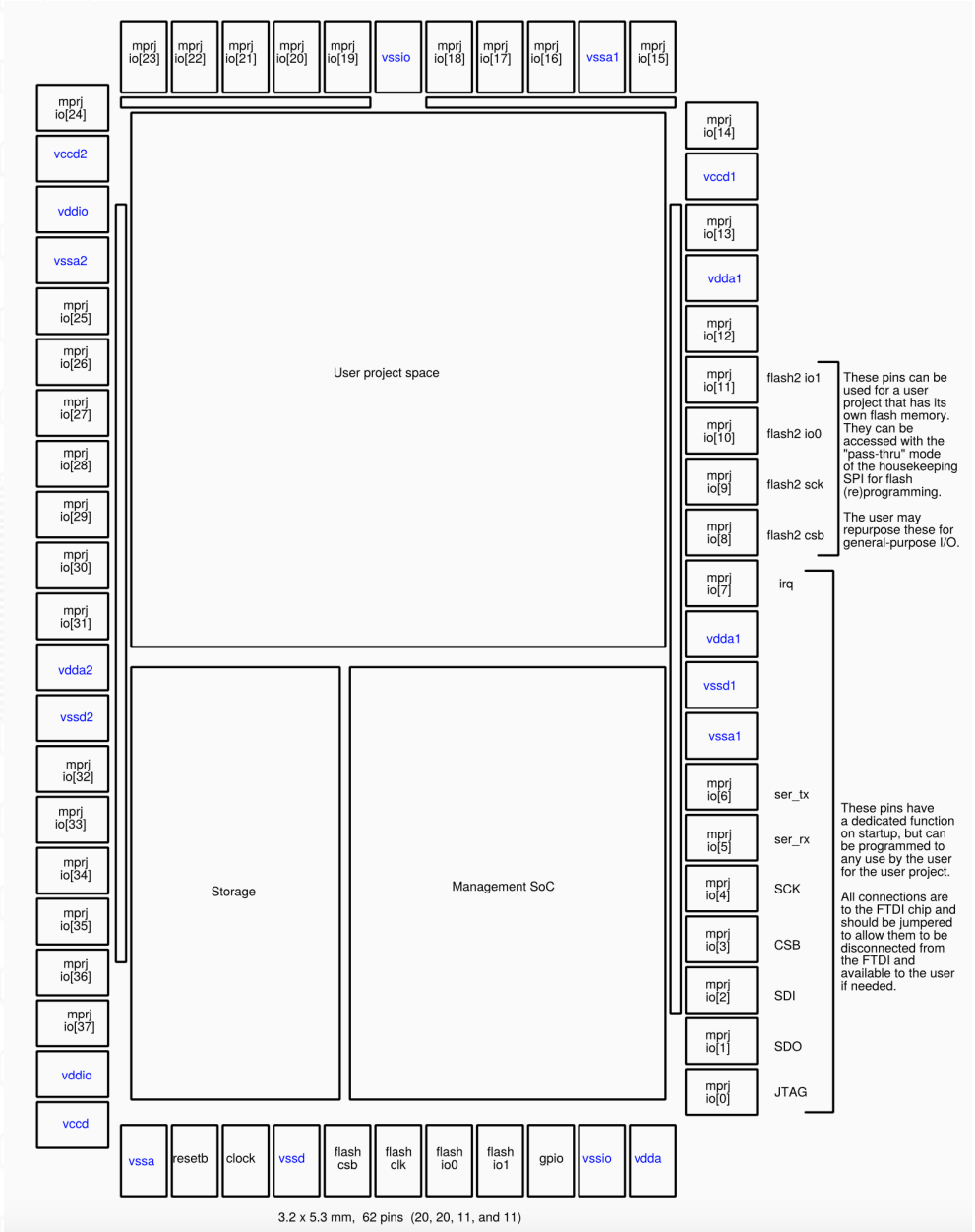


Measure cells with minimum number of pins with minimal risk / measurement complexity

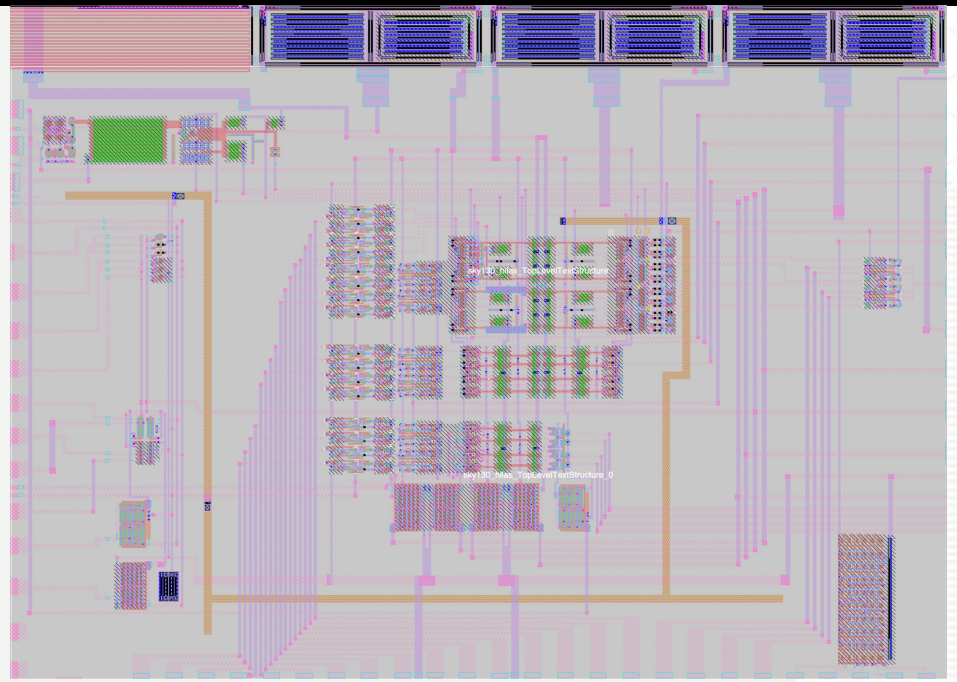
SKYWATER 130NM CMOS PROCESS

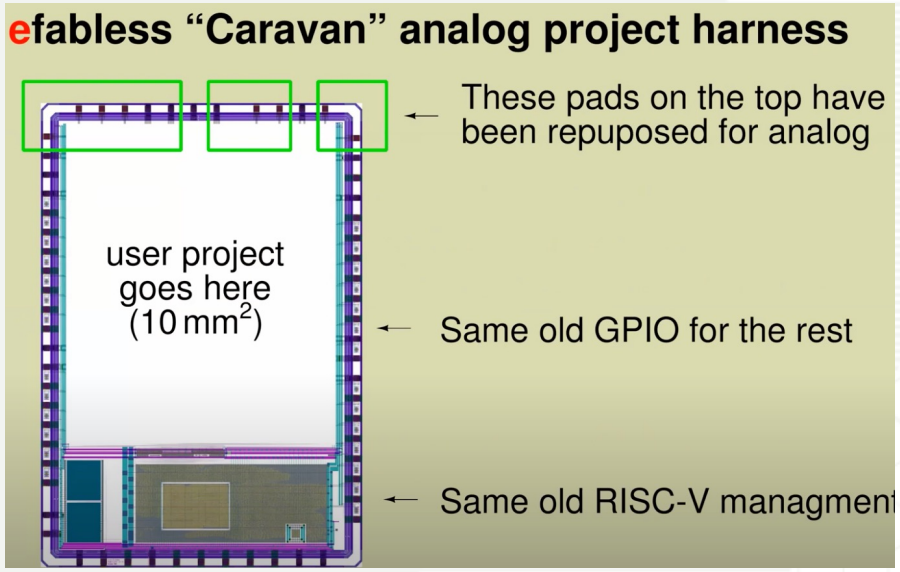


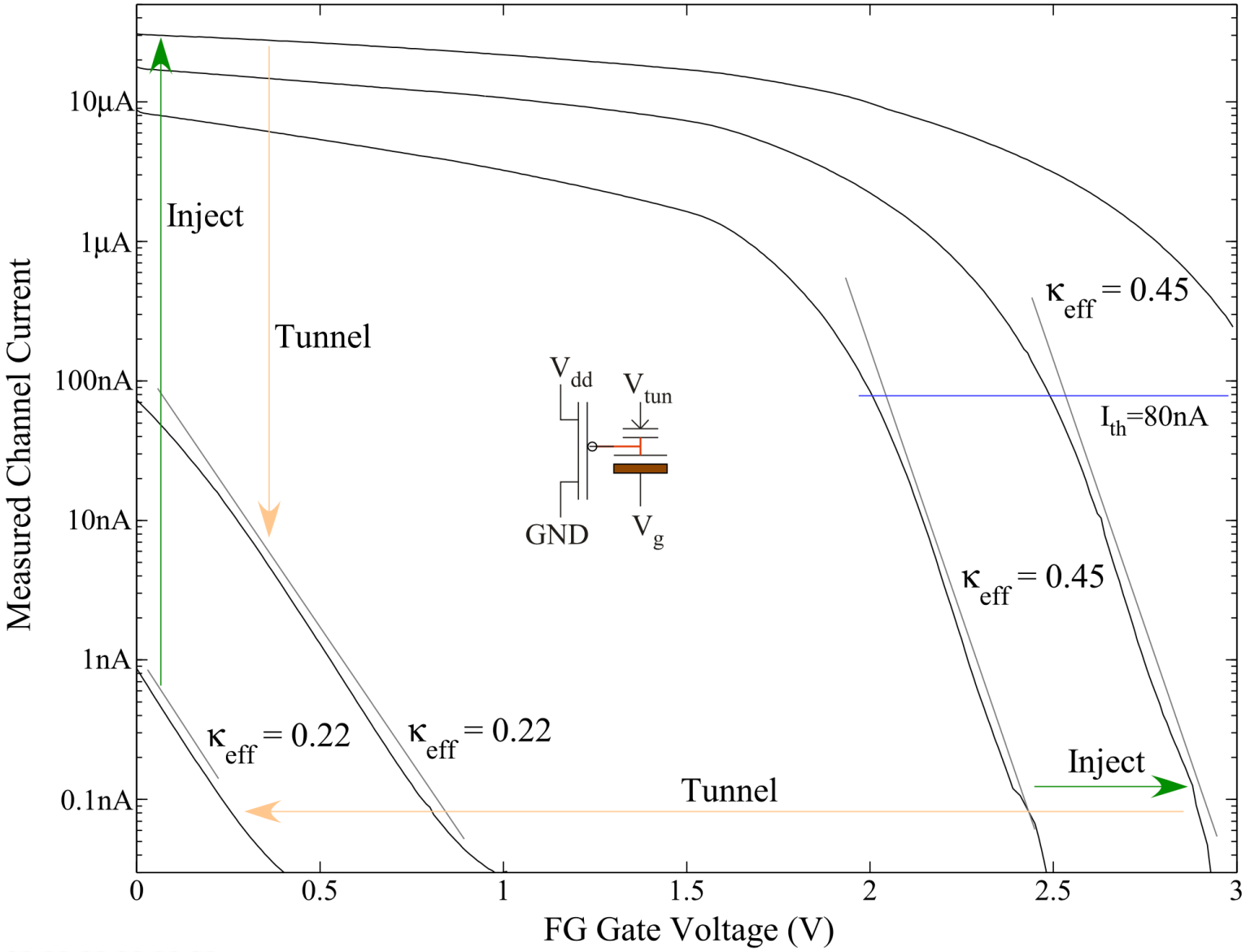




SKYWATER 130NM CMOS PROCESS







HIGH-LEVEL SYNTHESIS (ACOUSTIC BENCHMARK)

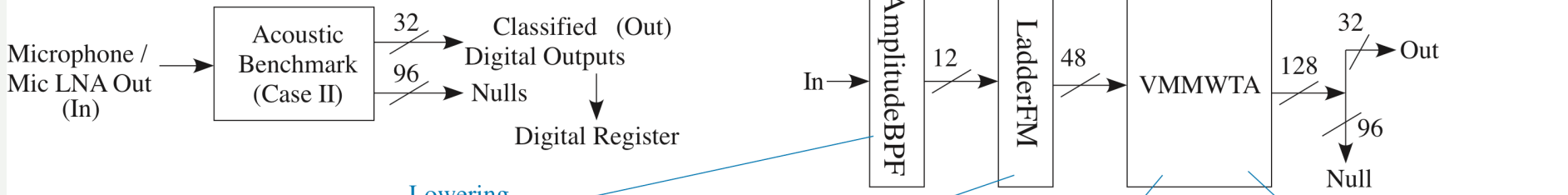
Case II Acoustic Benchmark: 32 out.
BPF filterbank, 3 delay stages, classifier, 3x nulls

Benchmark / Application Level
(High level algorithm)

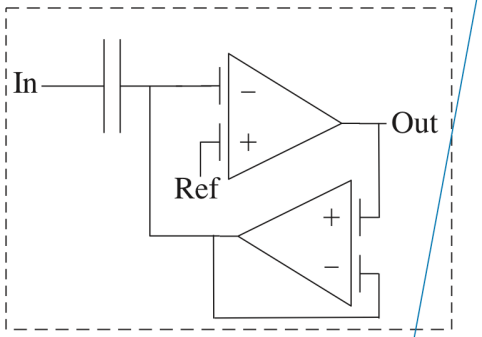
Lowering

Algoritthm Level

y1 = AmplitudeBPF(In)
y2 = M2V(LadderFM(input=y1,Ibias1=Ib1,Ibias2=Ib2))
Out = VMMWTA(input=y2,W=W1,Thresh=theta1,freq=400)

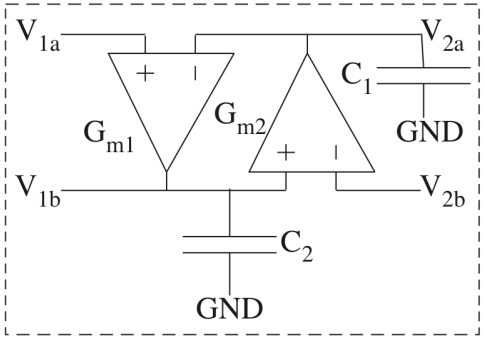


Lowering

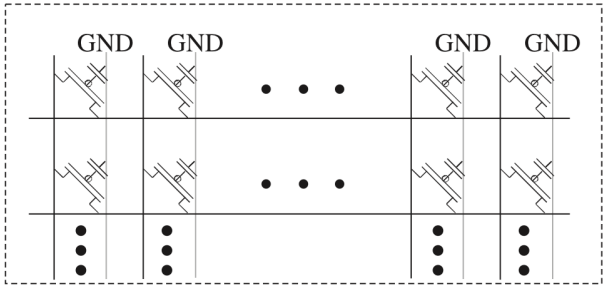


+

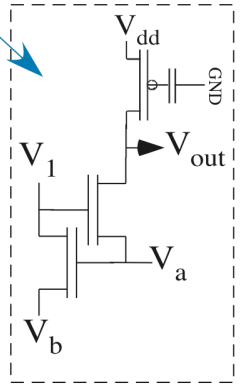
Lowering

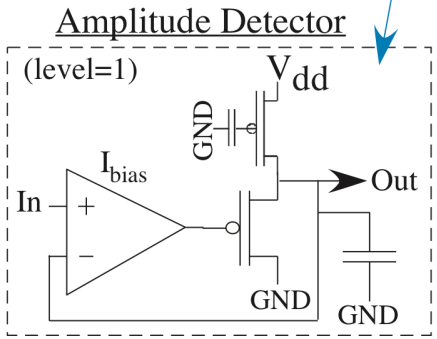


Lowering



Lowering





Lowering

(blif)

FPAA component map (tech file)

VPR (PnR) Gen2SWC

Switch List (FPAA targeted)

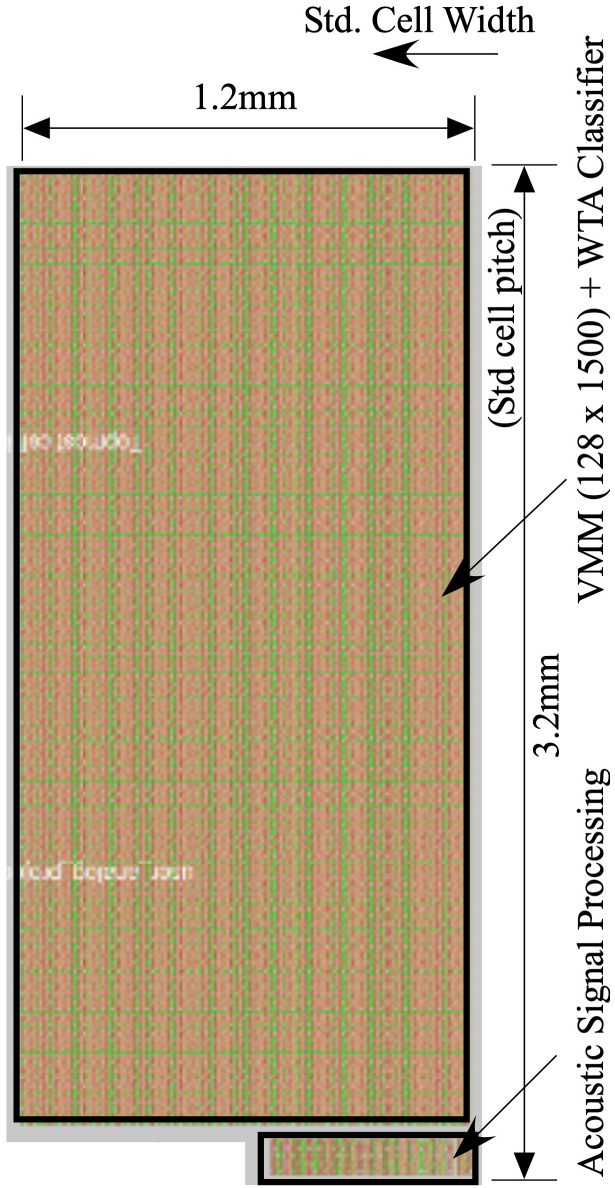
(gathered Islands)

FG-enabled Programmable Analog Std. Cells

PnR

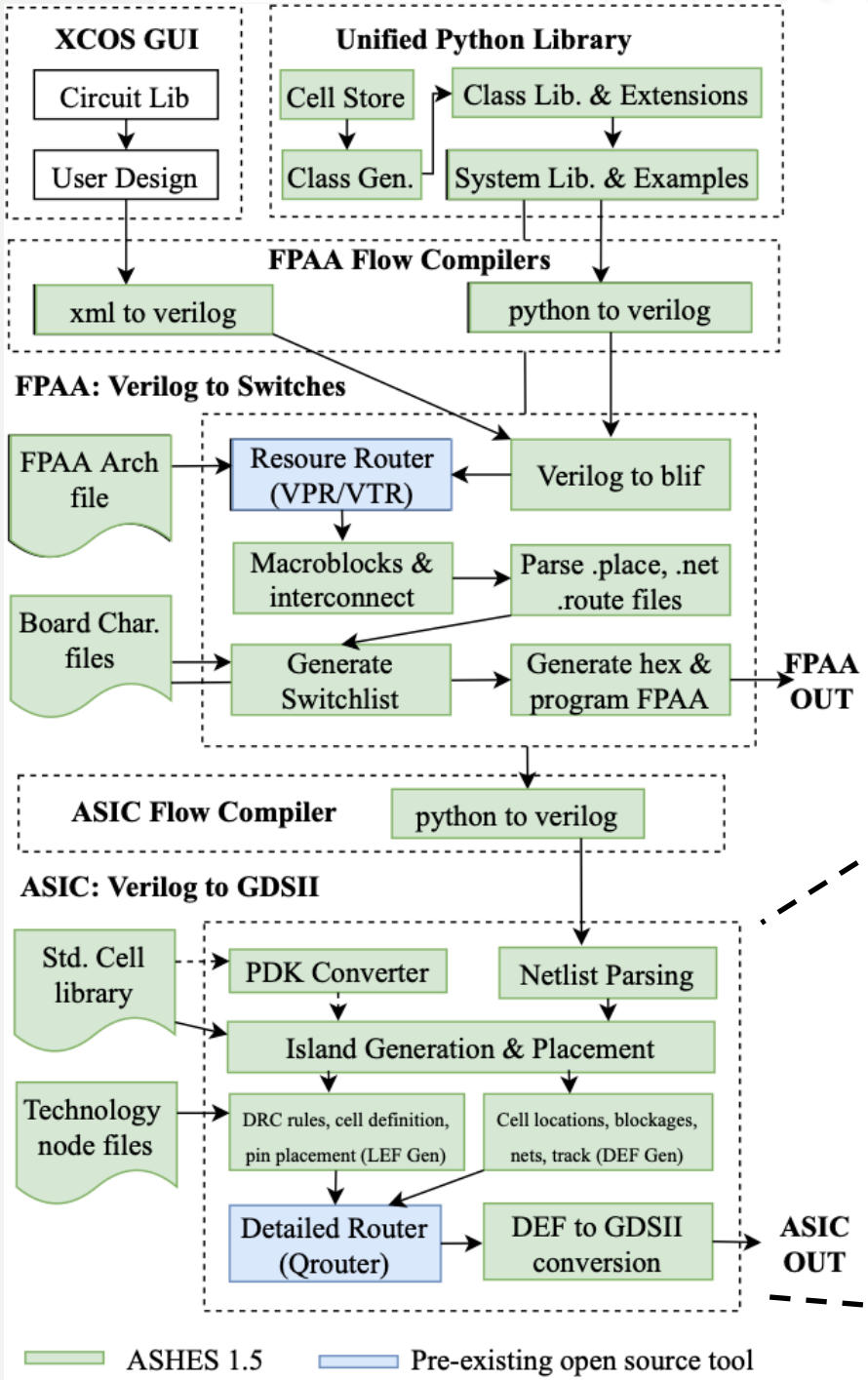
.gds (Fabricate IC)

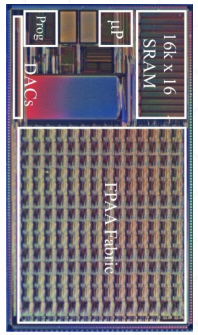
130nm CMOS

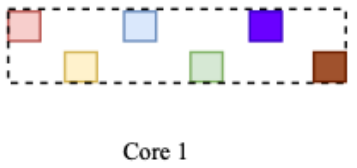


130nm Cells

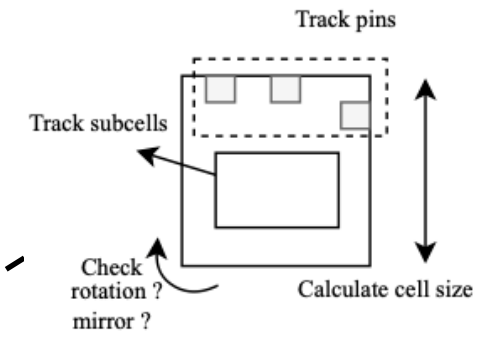
JOINT TARGETING & SYNTHESIS : ASHES 1.5



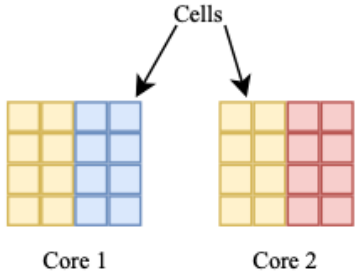




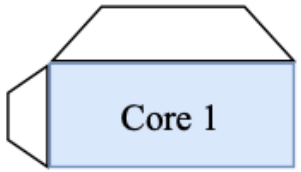
For non-abutting islands, perform core coagulation by placing cells in a checkered pattern.



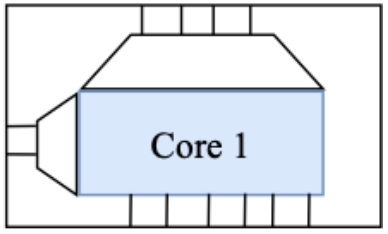
From the netlist, find the cell name in the library and parse all polygons from gds



For abutting islands, arrange the cells according to their connectivity



For a given core with floating gate cells, generate decoders and switches around the edges



Finally, route between islands or to a frame

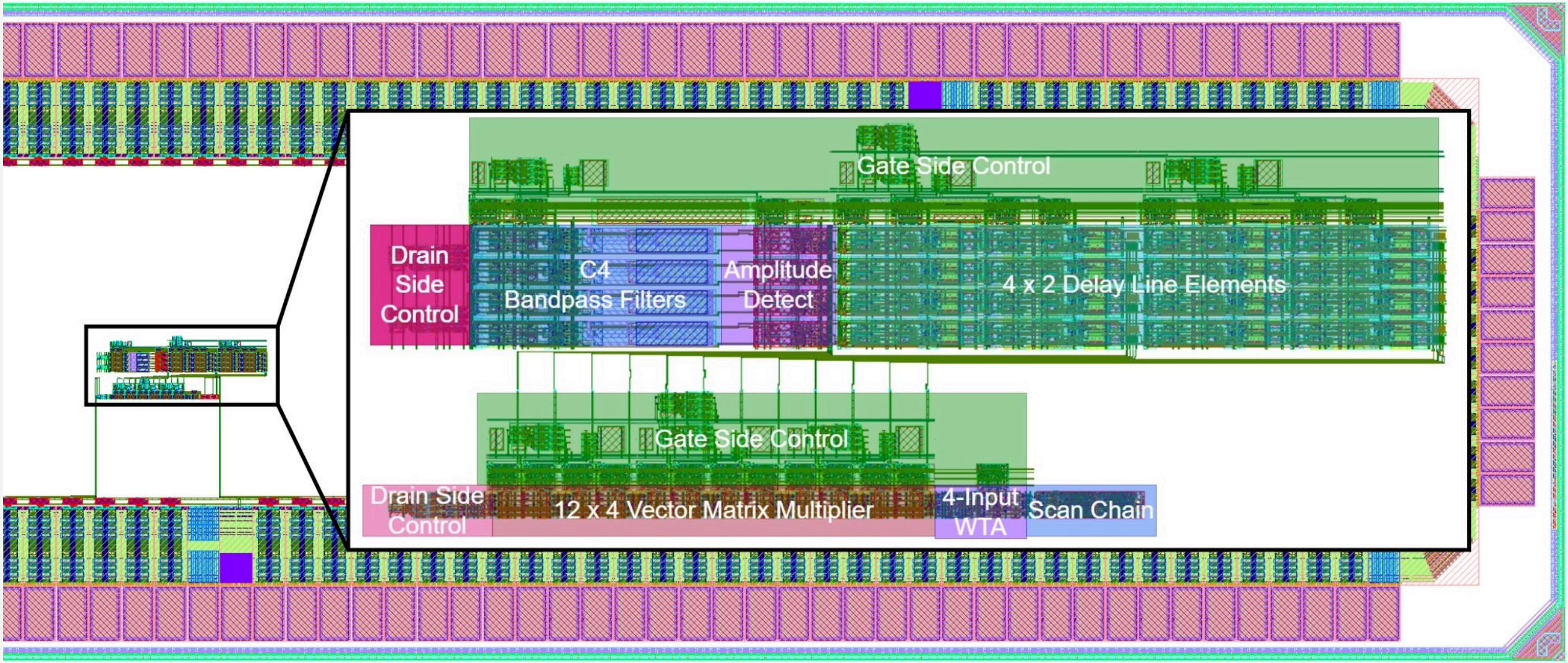
SMALL ACOUSTIC CLASSIFIER

Python Definition to GDSII

65nm CMOS

Acoustic Benchmark
(Part Case I):

4 BPF, AmpDetect,
2 Delay, VMM+WTA

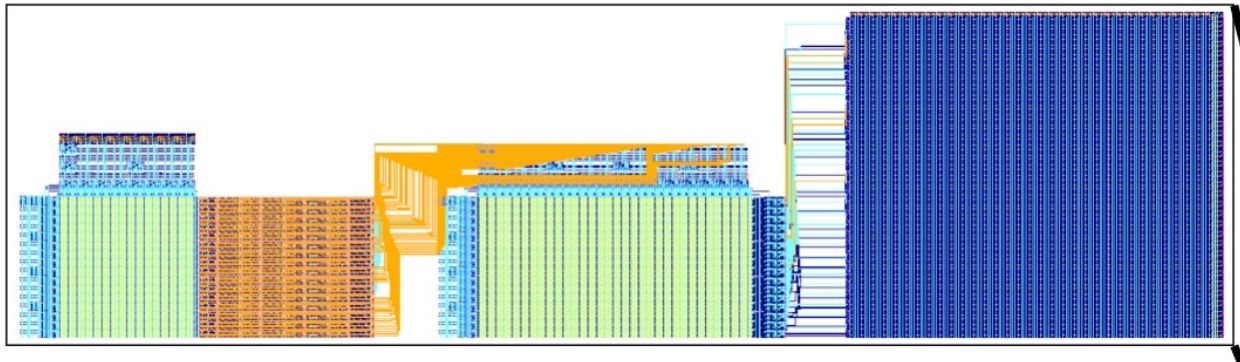


ADDITIONAL SYNTHESIS & COMPARISON

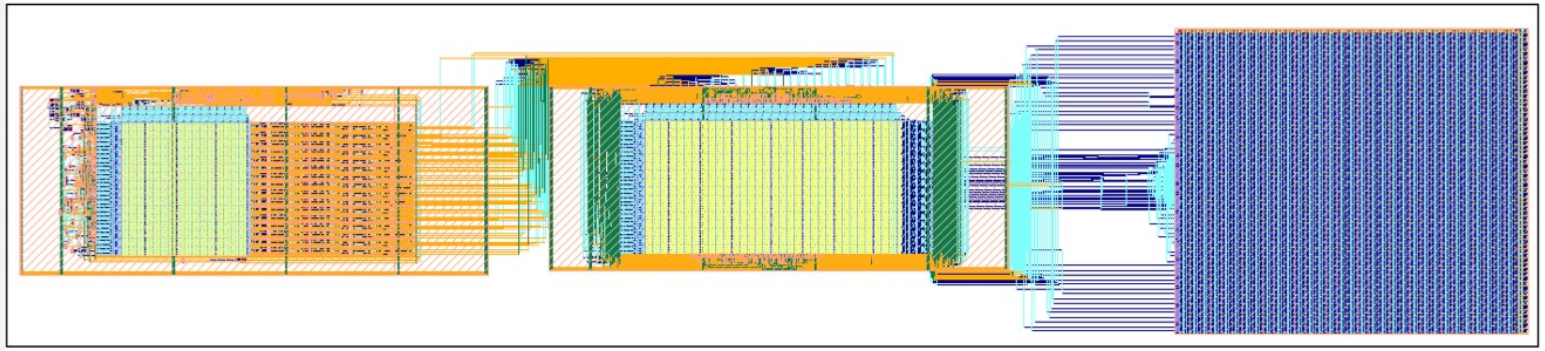
Arb Gen (32x64) → VMM (64 x 64) + WTA → D-FF array (64 x 64)

350nm Cells

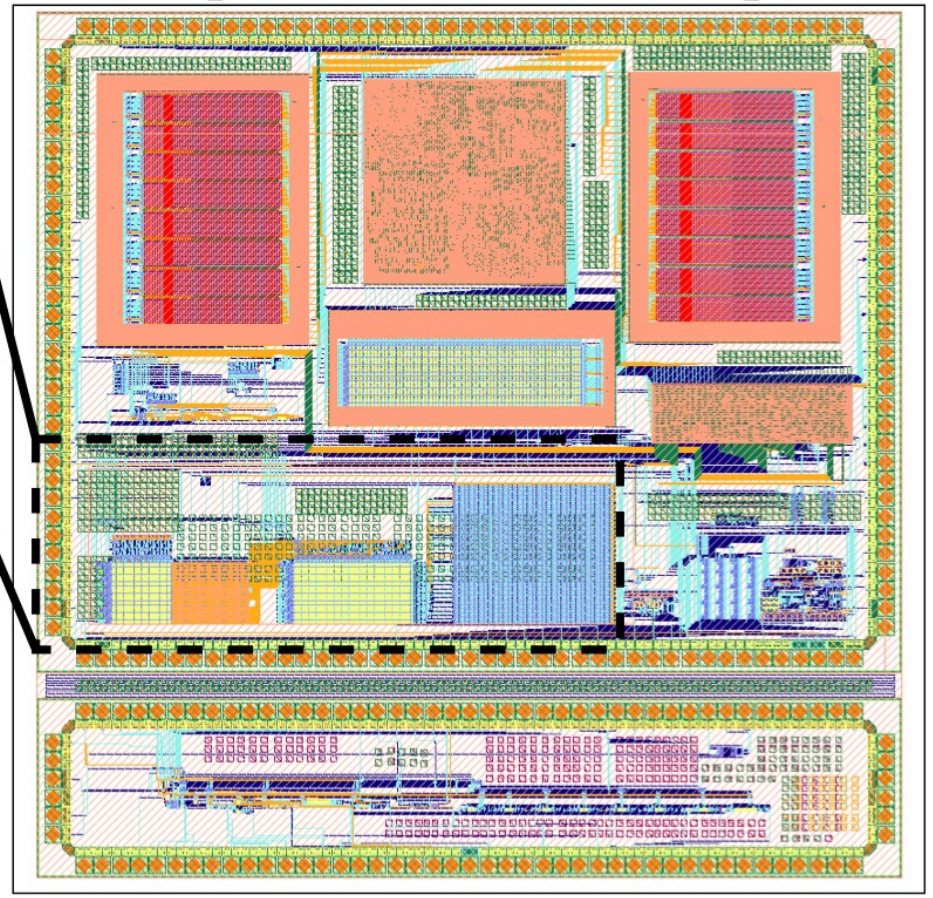
ASHES 1.5



Cadence: Genus + Innovus

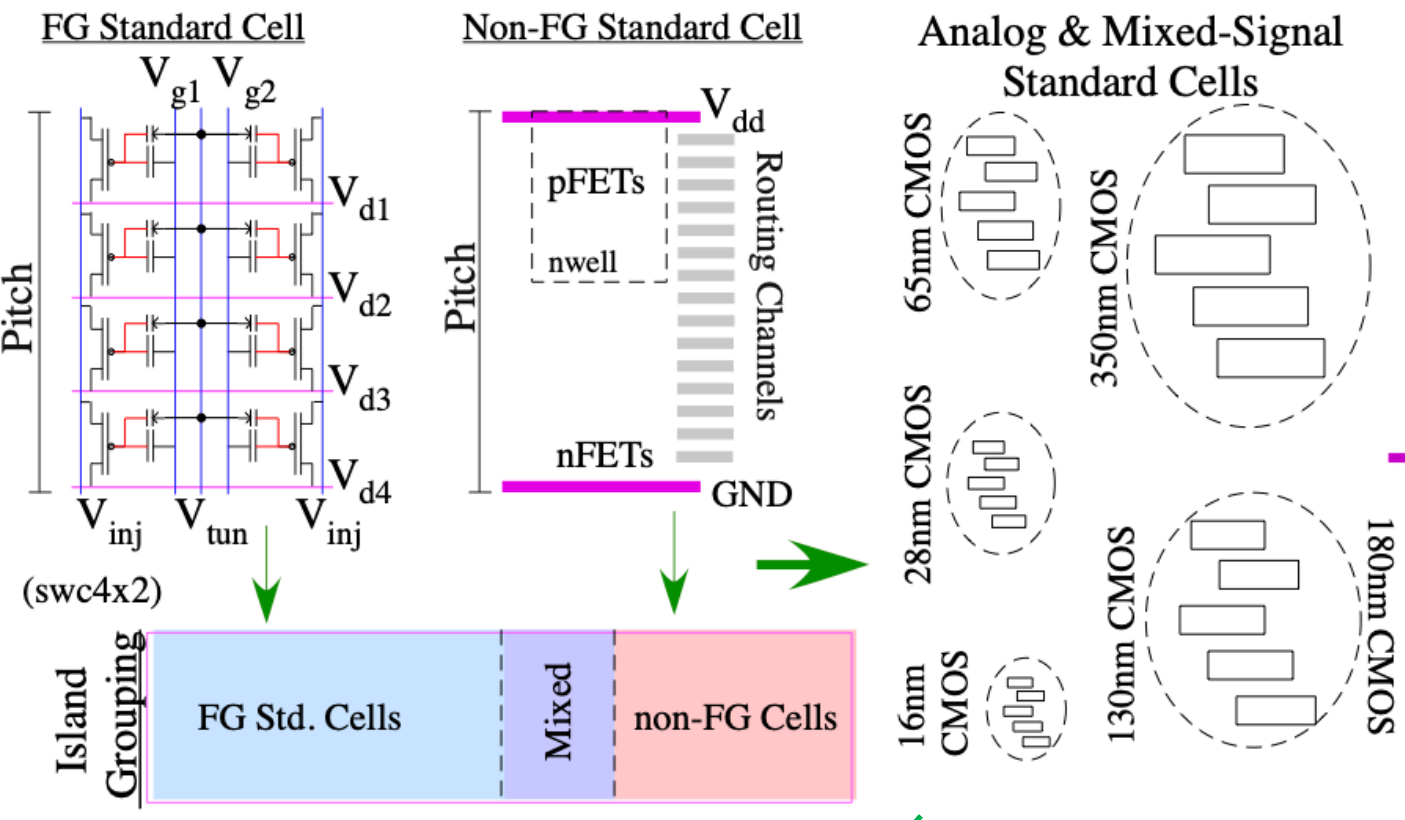


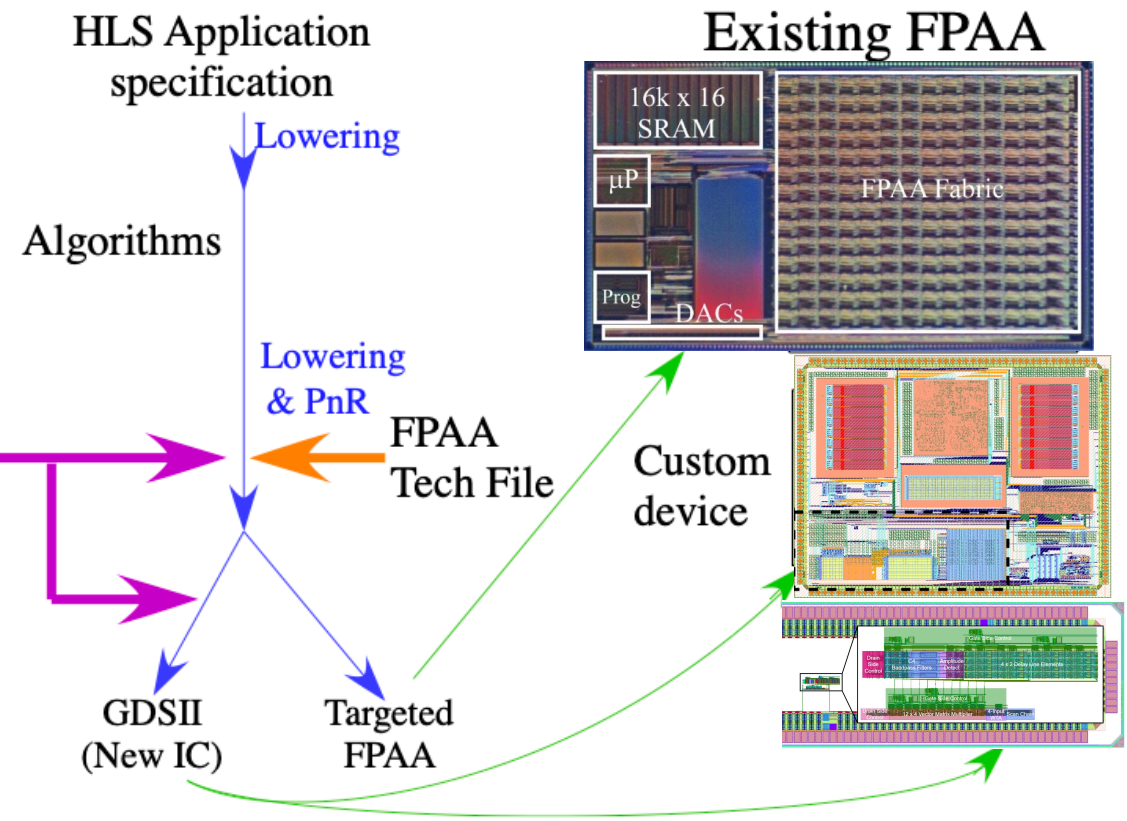
Taped out 350nm test chip



	Area μm^2			Wirelength μm			Placement (Routing) time s		
	Cadence	ASHES	ratio	Cadence	ASHES	ratio	Cadence	ASHES	ratio
Arb. waveform gen	469,560	450,065	1.04	40,149	20,490	1.96	-	-	-
Universal Classifier	506,212	418,576	1.21	173,172	31,764	5.45	-	-	-
Full Speed System	$3.2*10^6$	$2.4*10^6$	1.33	375,321	134,472	2.79	52 (1050)	0.53 (416)	98(2.52)

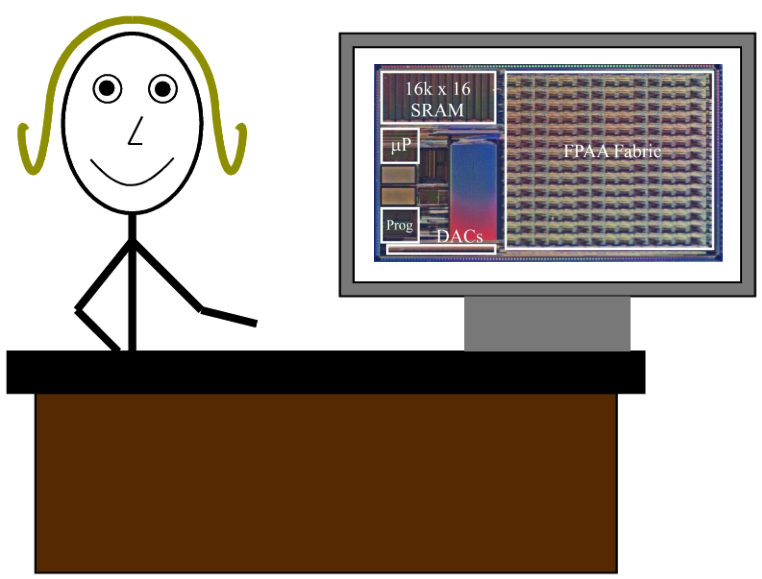
ANALOG & MIXED-SIGNAL HLS WITH FG STANDARD CELLS





- ✓ Floating-Gate (FG)
- ✓ Analog
- ✓ Analog Std. Cells

Want to read more on FPAAs?



INVITED PAPER

J. Hasler, "Large-Scale Field-Programmable Analog Arrays," *Proceedings of IEEE*, 2020.

Large-Scale Field-Programmable Analog Arrays

By JENNIFER HASLER, *Senior Member IEEE*

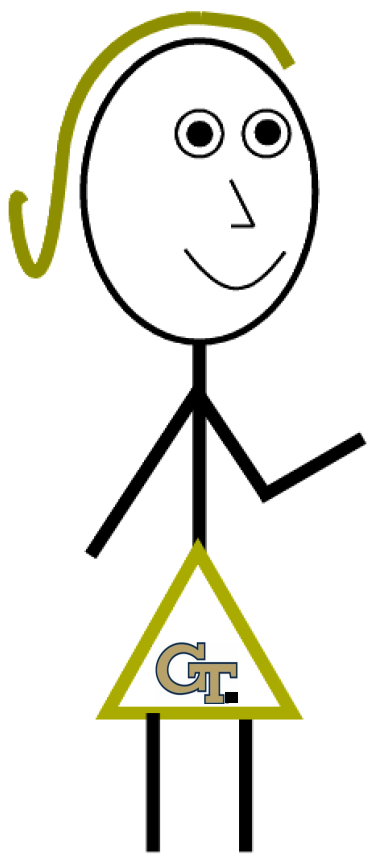
ABSTRACT | Large-scale field-programmable analog array (FPAA) devices could enable ubiquitous analog or mixed-signal low-power sensor to processing devices similar to the ubiquitous implementation of the existing field-programmable gate array (FPGA) devices. Design tools enable high-level synthesis to gate/transistor design targeting today's FPGA devices and the opportunity for analog or mixed-signal applications with FPAA devices. This discussion will illustrate the FPAA concepts and FPAA history. The development of FPAAs enables the development of multiple potential metrics, and these metrics illustrate future FPAA device directions. The system-on-chip (SoC) FPAA devices illustrate the IC capabilities, computation, tools, and resulting hardware infrastructure. SoC FPAA device generation has enabled analog computing with levels of abstraction for application design.

KEYWORDS | Analog-digital integrated circuits, analog integrated circuits, CMOS integrated circuits, field-programmable analog arrays (FPAAs), field-programmable gate arrays (FPGAs)

lution [1] enabled further separation of roles to address the increasing complexity resulting from Moore's law scaling [2]–[4]. Digital microprocessors (μP) are ubiquitous from embedded applications to general-purpose (GP) computing. Programmability enables changing parameters or coefficients in a particular algorithm. Changing the stored matrix of weights for a vector–matrix multiplication (VMM) is an example of programmability. Configurability enables changing the data flow, topology, as well as the order or operations. Changing the program for an μP is an example of configurability. Field-programmable gate array (FPGA) devices, programmable and configurable gate-level digital devices, enabled digital designers' design capabilities from gate- to system-level designs. FPGAs are ubiquitous digital computing devices found everywhere over the last two decades, arising from their initial conception (1980) and commercialization (mid-1980s) [5].

Modifying the parameters or control flow requires significant changes, such as soldering new components

ADDITIONAL RESOURCES

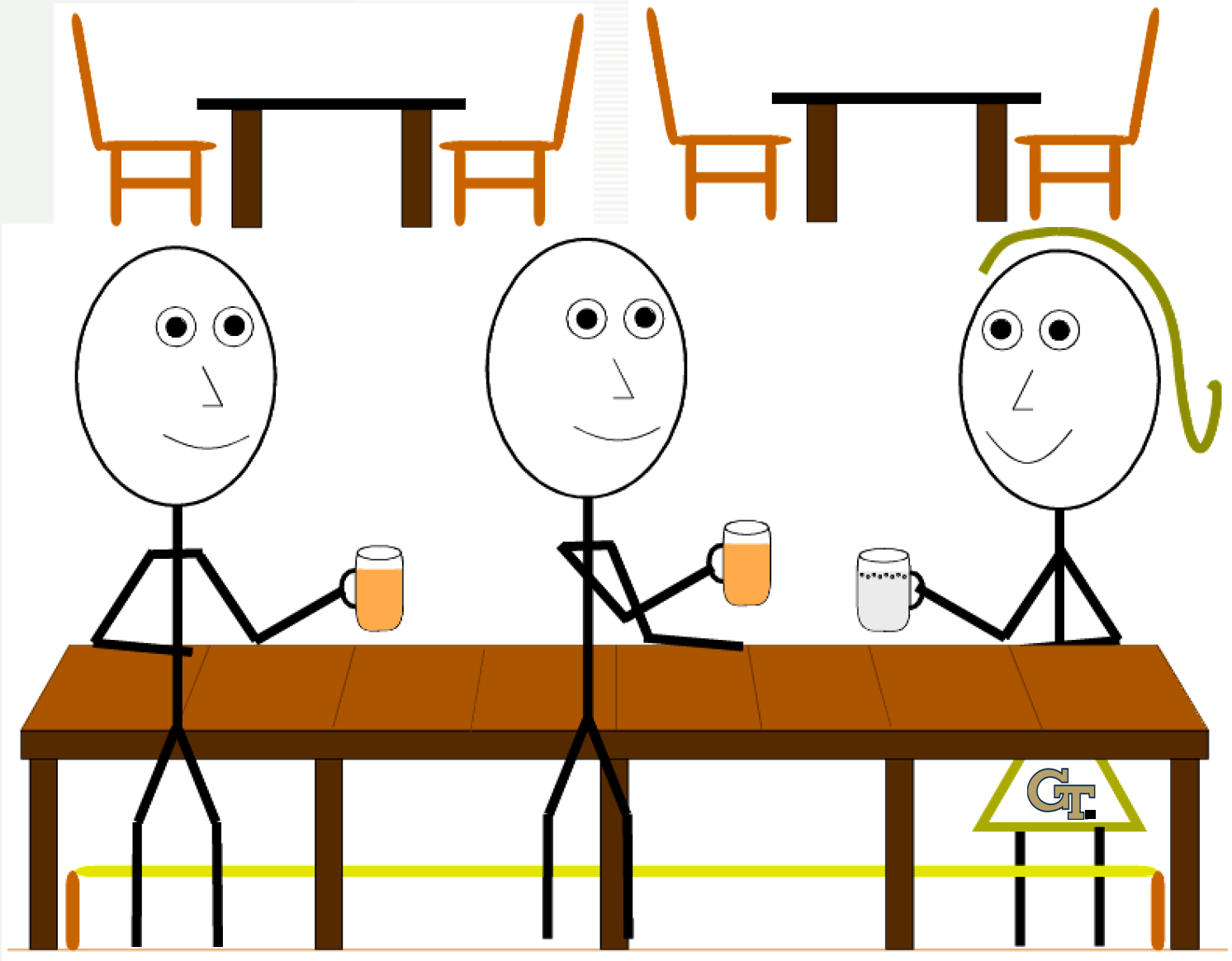


Videos:

- J. Hasler, FPAA: History, Development, Classification: https://youtu.be/2lsz9gi8Oz8
- J. Hasler, Future of FPAA opportunities: https://youtu.be/rpSdb88ubfk
- J. Hasler & A. Natarajan, Intro Open-Source FPAA Toolset: https://www.youtube.com/8SVdhztVroc
- J. Hasler, Historical FG Perspective, https://youtu.be/R8iV01KZch4
- J. Hasler, FPAA Enabling Physical Computing, https://youtu.be/IGzinnykZIw

FPAA on-line Workshop: http://hasler.ece.gatech.edu/FPAAWorkshop/index.html

PHYSICAL COMPUTING



Further questions are definitely welcome and appreciated.