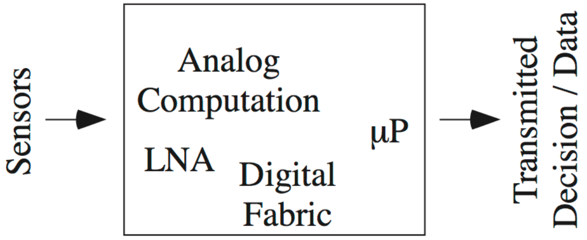
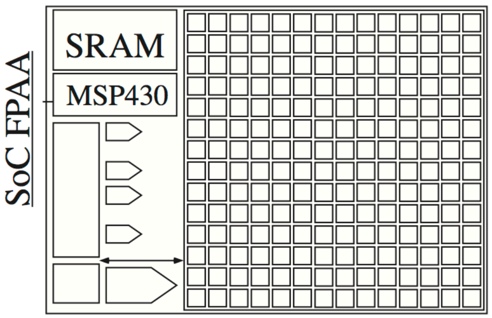
Embedded Classifiers for Energy Constrained IoT Network Security

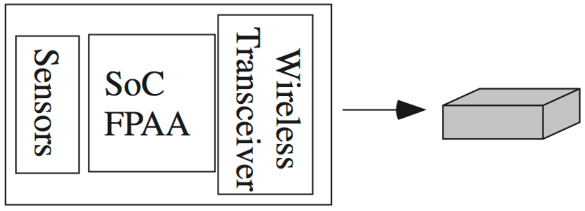
Jennifer Hasler

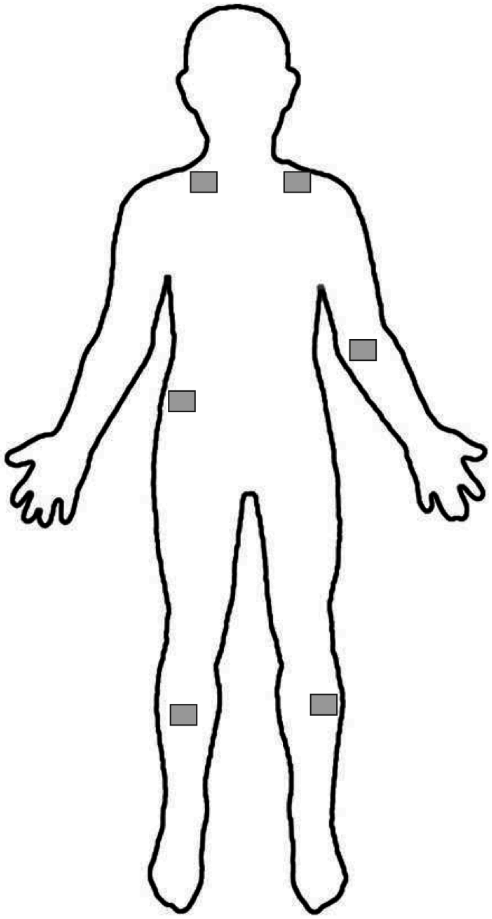
Georgia Institute of Technology

Physical Computing Enable Small Sensor Nodes

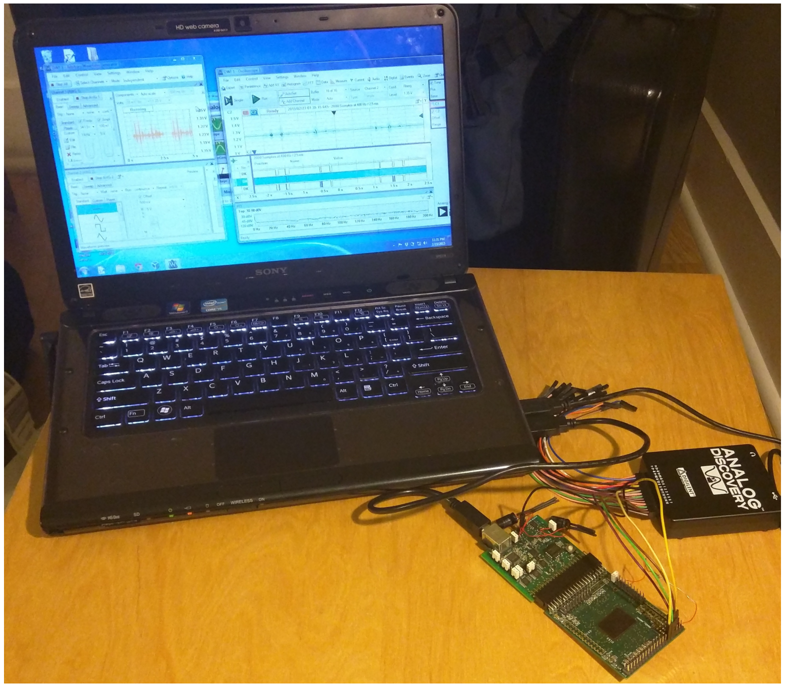








Physical / Analog / Mixed-Signal Computing is Here!



Command Word < 23μW power

Knee-Joint Rehab < 15μW

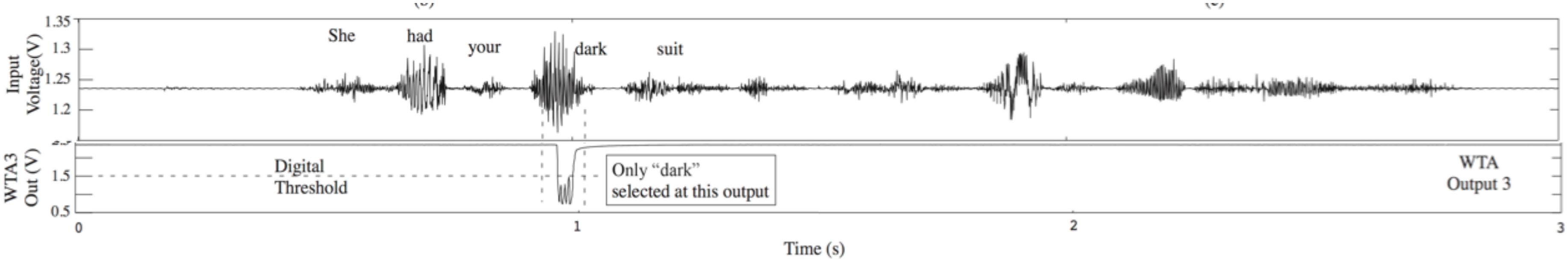
Analog + Digital FPAA

Applications in sensors, acoustics, imaging

On-chip Machine learning shown (VMM+WTA)

Capability over multiple IC processes

Measured Results for a phrase from the TIMITdatabase to recognize the word “Dark”

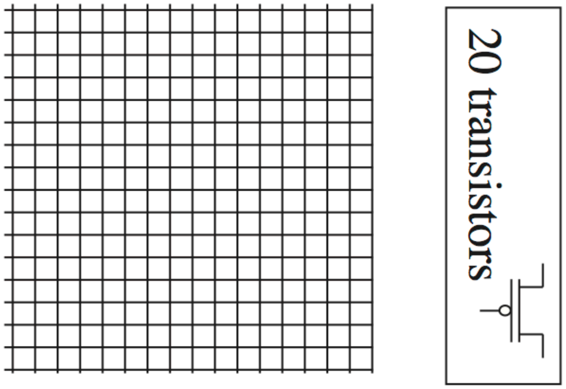
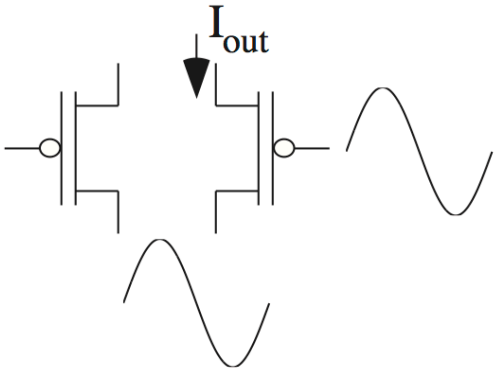


Parameter Density for highly accessible components ~ x 1M greater than next closest device (PSOC)

Analog SP ~ 1000x Custom Digital SP

Why Analog (Physical Based) Processing?

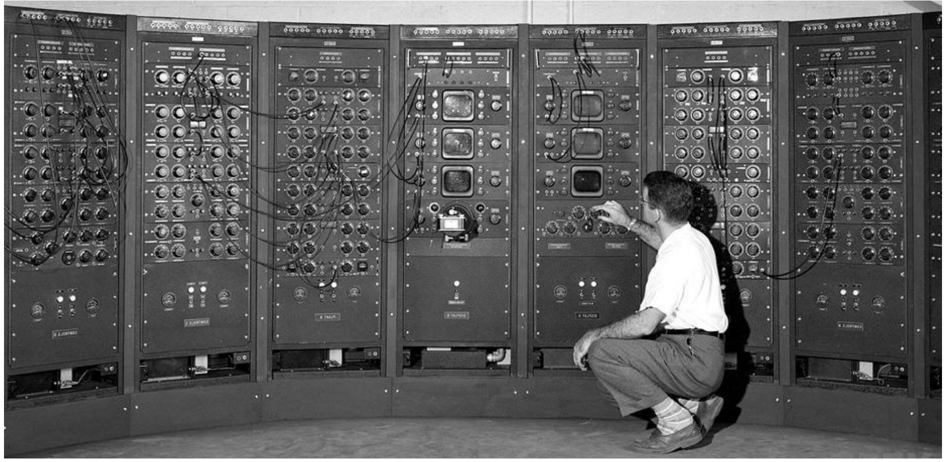
Mead Hypothesis (1990): Analog x1000 efficiency improvement

	Digital	Analog
Multiplication (digital: 16bit)		
Energy/ operation	x1000	x1
Size	x100	x1

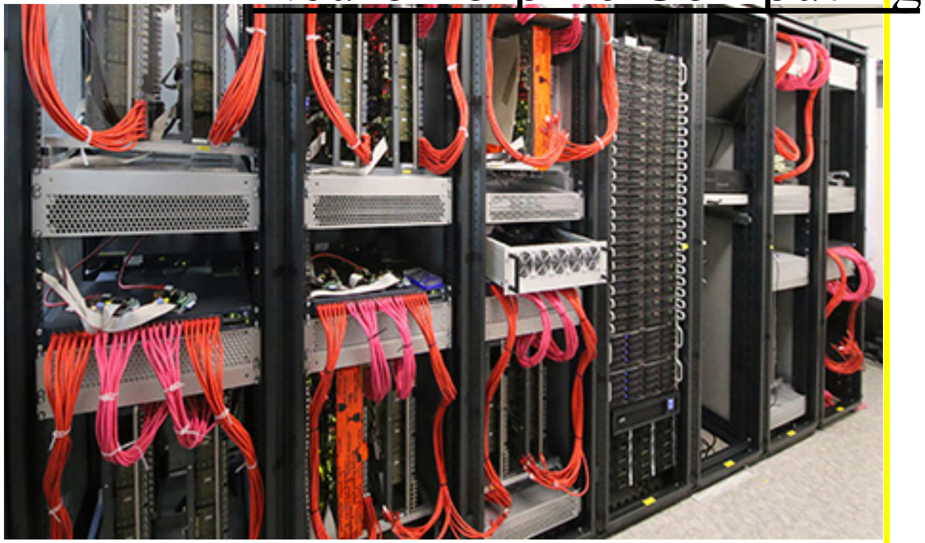
- Analog (VMM): ~100 fJ / MAC (10MMAC/μW) @ yield
- Other Analog SP similar: Freq Decomp / Analog FT
VMM, GMM
Classifiers
Adaptive Filters

Physical Computing Unifies Approaches

Analog Computing

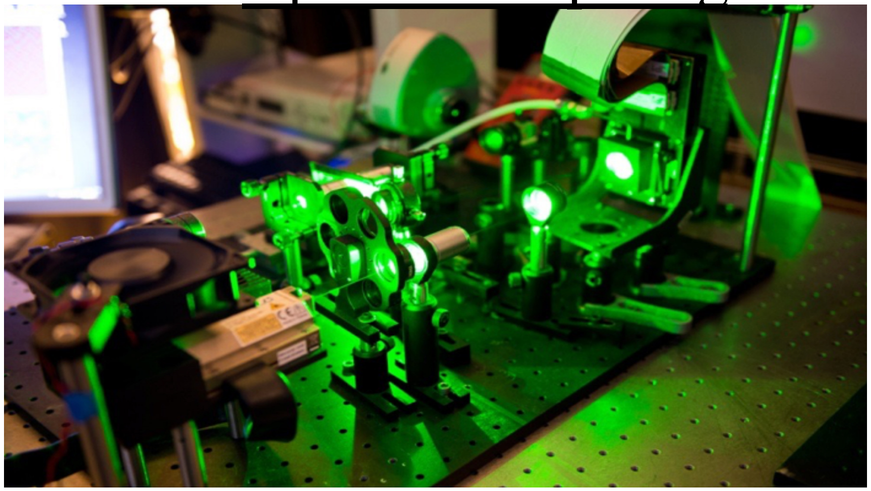


Neuromorphic Computing

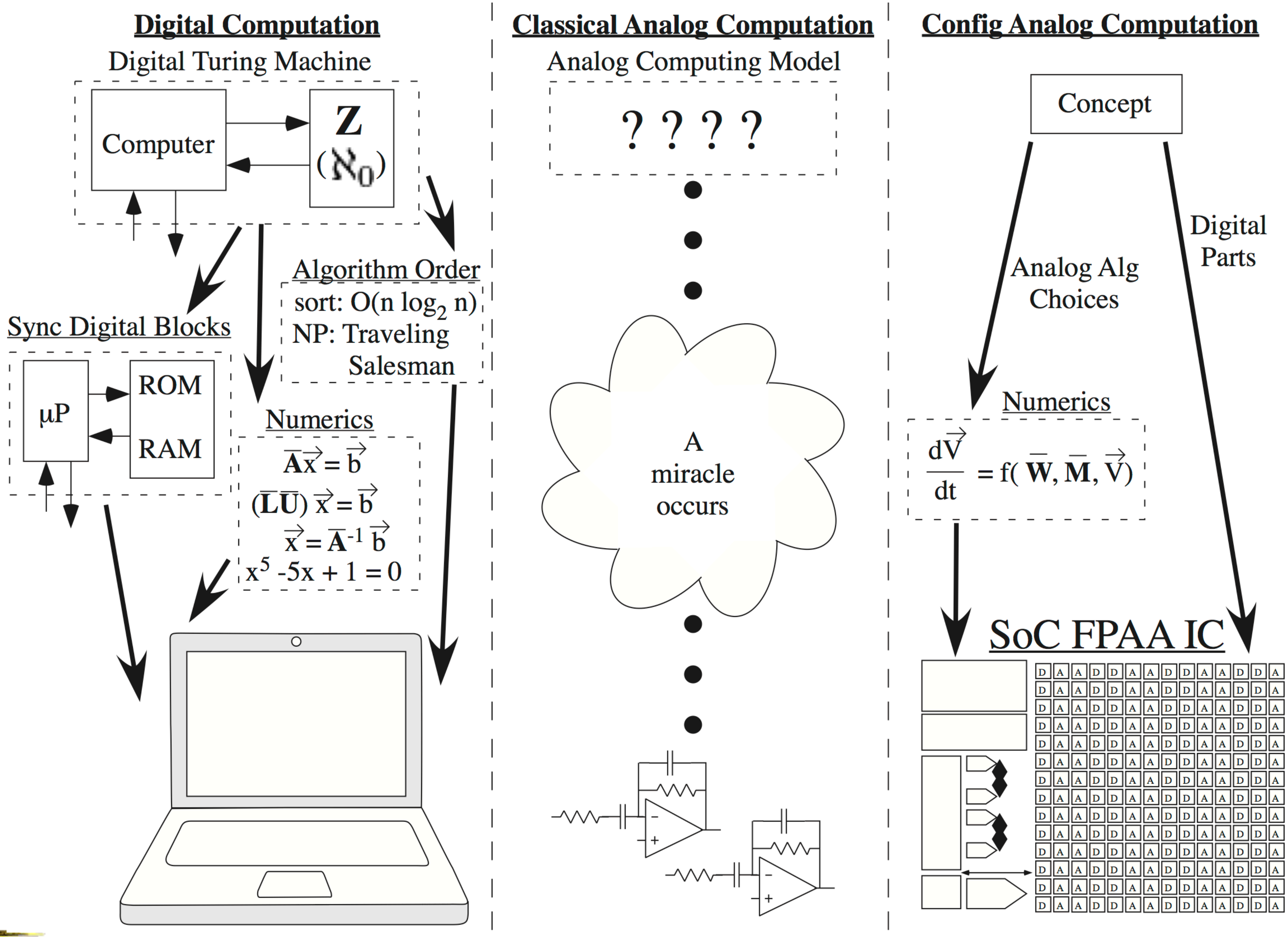


Quantum Computing

Optical Computing



Framework for Analog Computation is Necessary and Arriving



Digital Computation

Digital Turing Machine

Computer

Z

($\aleph_0$)

Sync Digital Blocks

µP

ROM

RAM

Algorithm Order

sort: $O(n \log_2 n)$

NP: Traveling Salesman

Numerics

$\bar{\mathbf{A}}\vec{x} = \vec{b}$

$(\bar{\mathbf{L}}\bar{\mathbf{U}})\,\vec{x} = \vec{b}$

$\vec{x} = \bar{\mathbf{A}}^{-1}\,\vec{b}$

$x^5 - 5x + 1 = 0$

Classical Analog Computation

Analog Computing Model

? ? ? ?

A miracle occurs

Config Analog Computation

Concept

Analog Alg Choices

Digital Parts

Numerics

$$\frac{d\vec{V}}{dt} = f(\,\bar{\mathbf{W}}, \bar{\mathbf{M}}, \vec{V})$$

SoC FPAA IC

D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A
D A A D D A A D D A A D D A

Physical Computing → Increased Computational Efficiency

10MMAC(/s)/W
(1979)

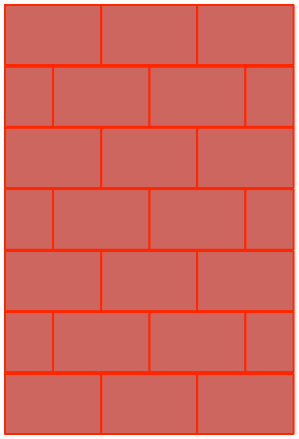
x1000
Improvement

CMOS
Scaling

10MMAC(/s)/mW
(2010)

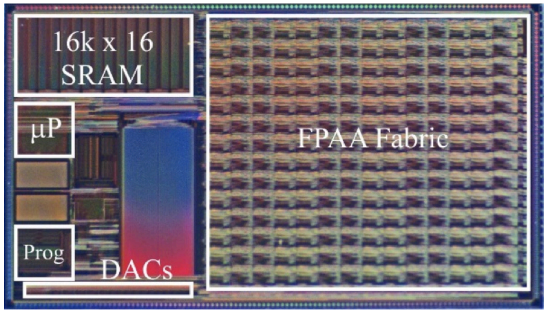
x1000
Improvement

Physical
Computation



Energy
Efficiency
Wall

Large-Scale Field Programmable
Analog Arrays (FPAA)

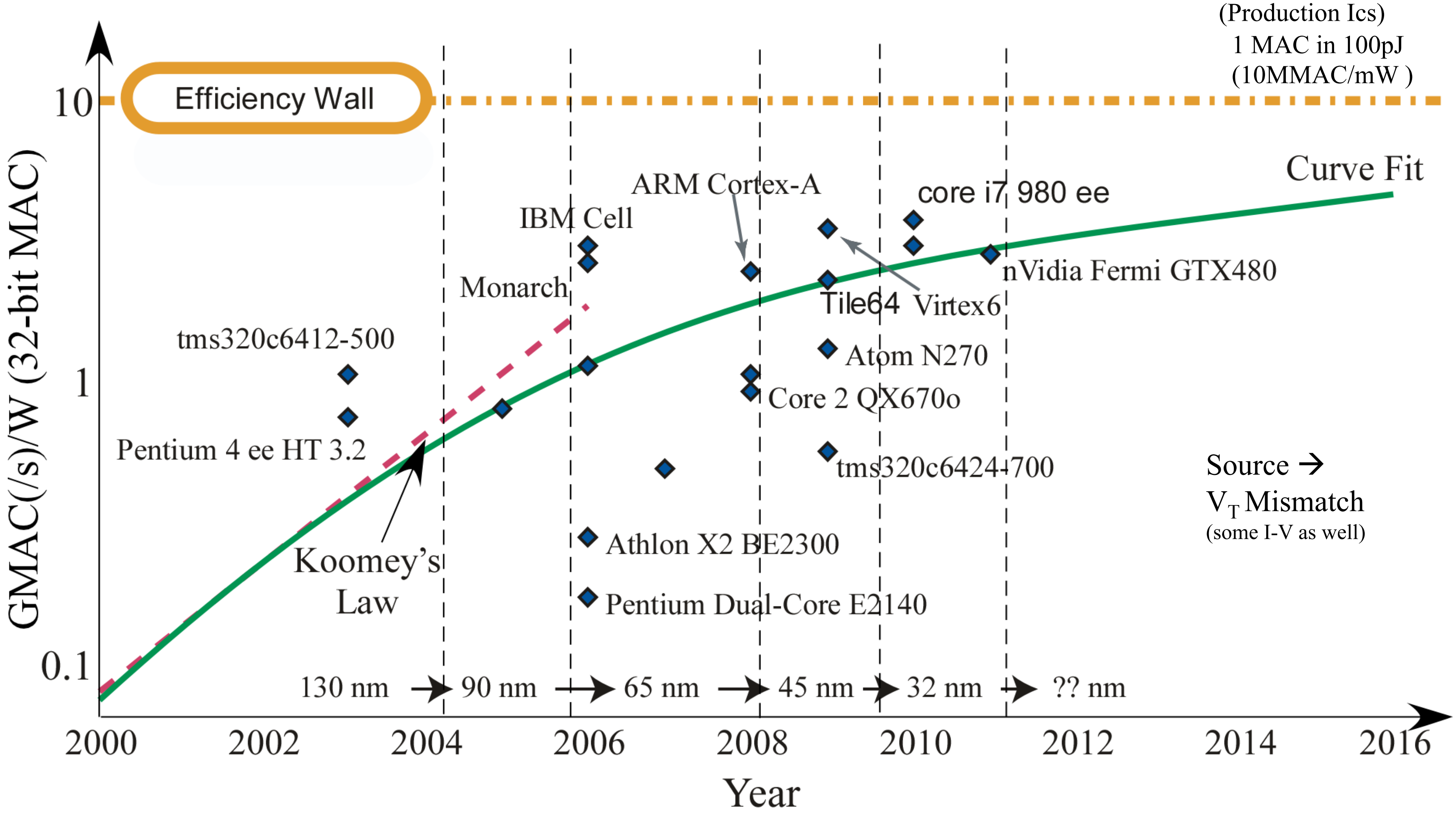


10MMAC(/s)/µW

Wearable Devices Require more Efficiency

Why Analog (Physical Based) Processing?

Digital Hitting Limits of Power Efficiency



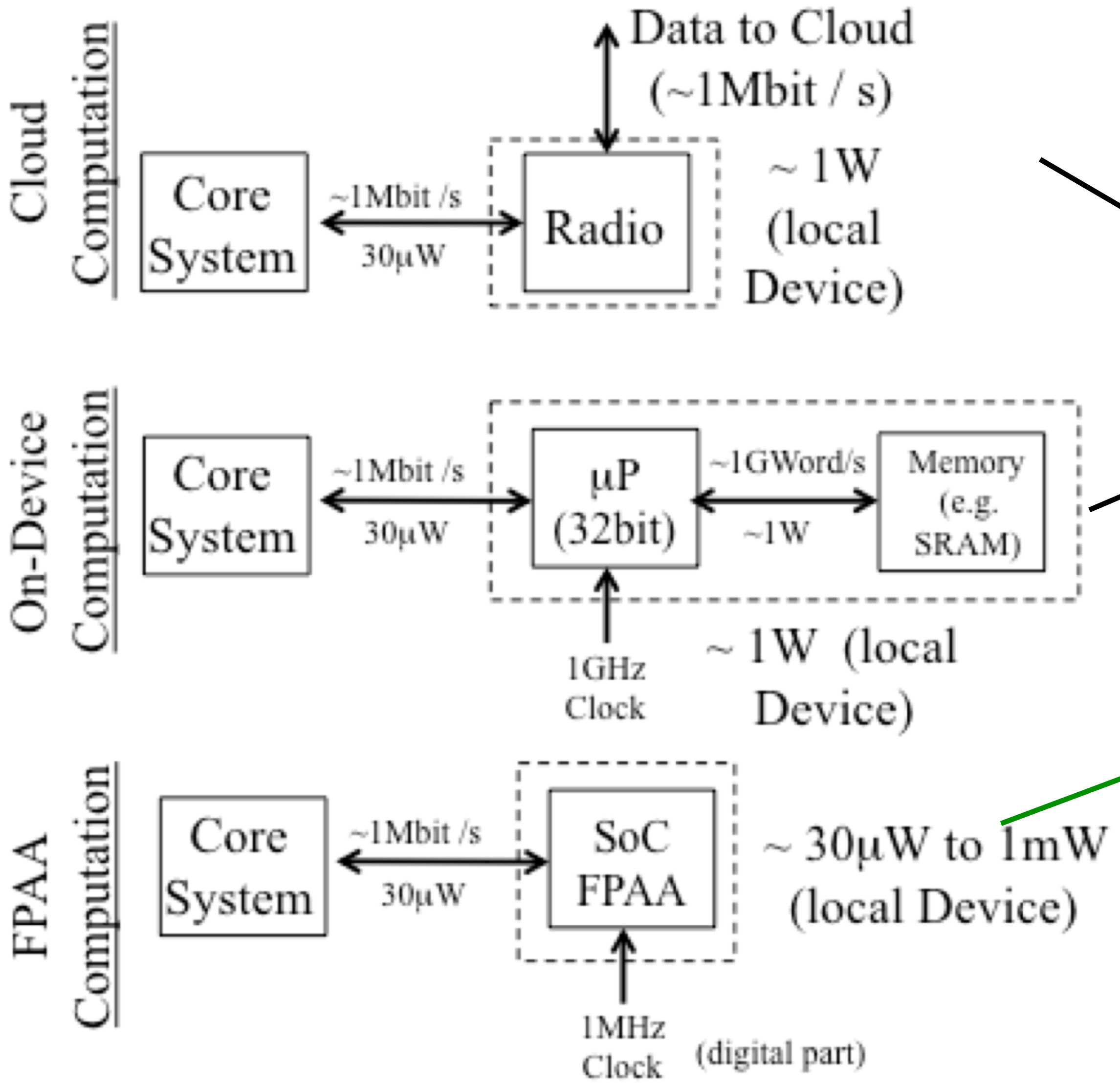
Power Efficiency wall
(Production Ics)
1 MAC in 100pJ
(10MMAC/mW)

Source →
V_T Mismatch
(some I-V as well)

Results created its own DARPA program

Battery Energy Density: x10 over 40 years

FPAA vs. Embedded / Cloud Computation



If cloud is "free"….

If cloud is "drops", then disaster

If both → more cost

SoC FPAA decreases energy and resulting complexity

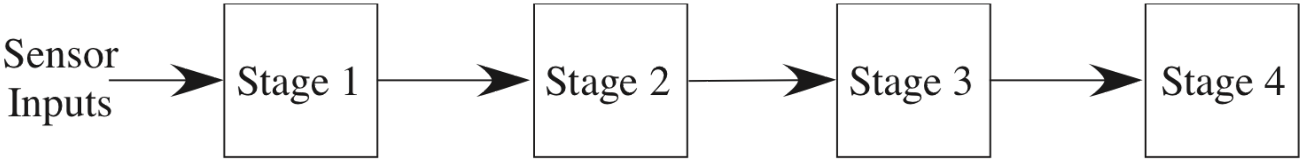
Physical Algorithms empowering wearable devices

Energy Harvesting: 10uW / cm^2

Where to use ultra-low energy?

Sensor node < 100µW

x1000 energy improvement utilizes context-aware physical computing to enable 100µW end-to-end sensor node.

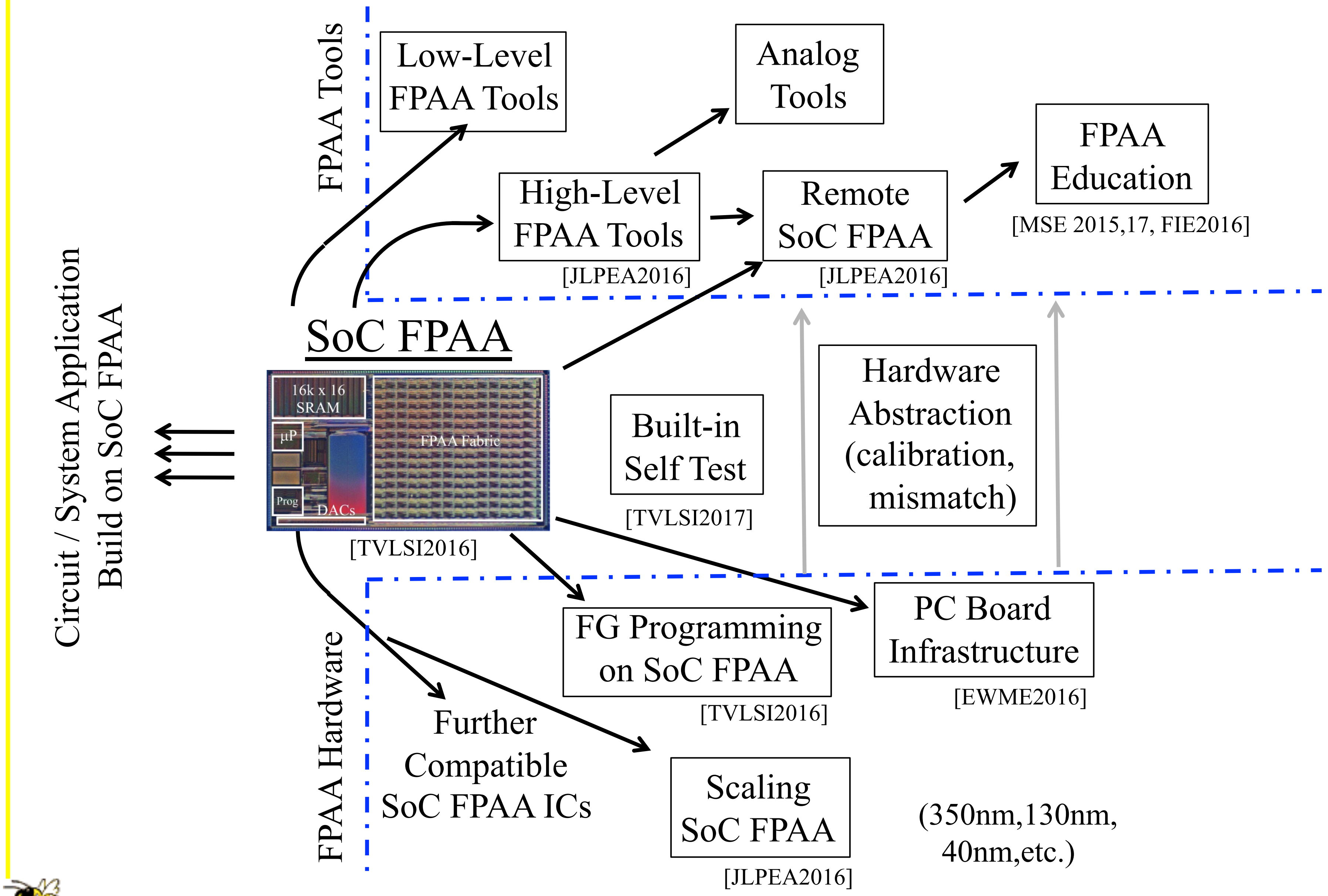


	Continuous Operation (analog)	Classification (analog + digital)	Second Wakeup (e.g. processor wakeup)	Full Processing (e.g. Transceiver)
Average on time	100%	1-3%	0.1-0.2%	0.01%
Operating power	1 to 10µW	~100µW	1-5mW	30-100mW
Total(max) Power	10µW	3µW	10µW	10µW
Digital		<1MMAC/s	~10-20MMAC/s or 20MHz clock	Transciever on
Analog	10-100MMAC/s	1GMAC/s	50GMAC/s	

← More computation near sensor

Increasing Energy Decreasing Use →

SoC FPAA Components



Circuit / System Application
Build on SoC FPAA

FPAA Tools

FPAA Hardware

SoC FPAA

[TVLSI2016]

Low-Level FPAA Tools

Analog Tools

High-Level FPAA Tools

[JLPEA2016]

Remote SoC FPAA

[JLPEA2016]

FPAA Education

[MSE 2015,17, FIE2016]

Built-in Self Test

[TVLSI2017]

Hardware Abstraction (calibration, mismatch)

FG Programming on SoC FPAA

[TVLSI2016]

PC Board Infrastructure

[EWME2016]

Further Compatible SoC FPAA ICs

Scaling SoC FPAA

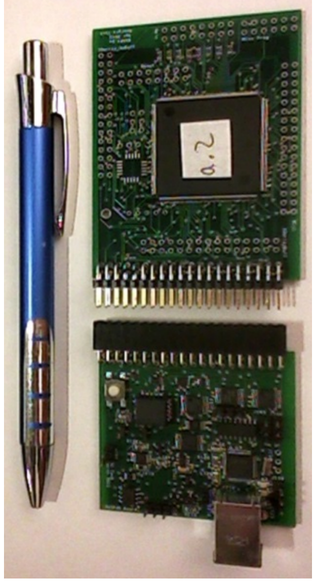
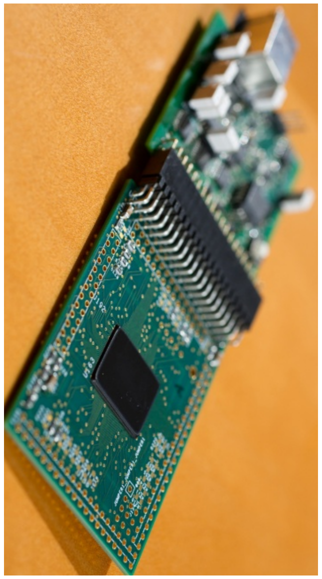
[JLPEA2016]

(350nm,130nm,
40nm,etc.)

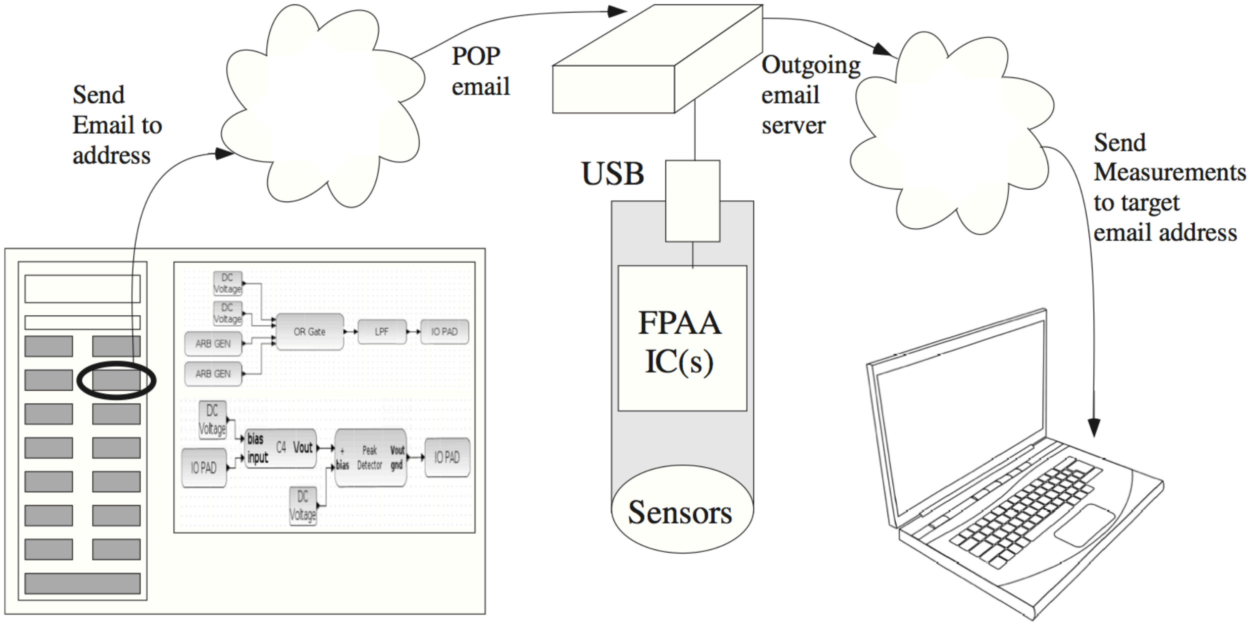
FPAA Infrastructure

- FG Programming looks like controlled download to µP device → Straightforward to program a device (code in Scilab, Python, Java, ….)

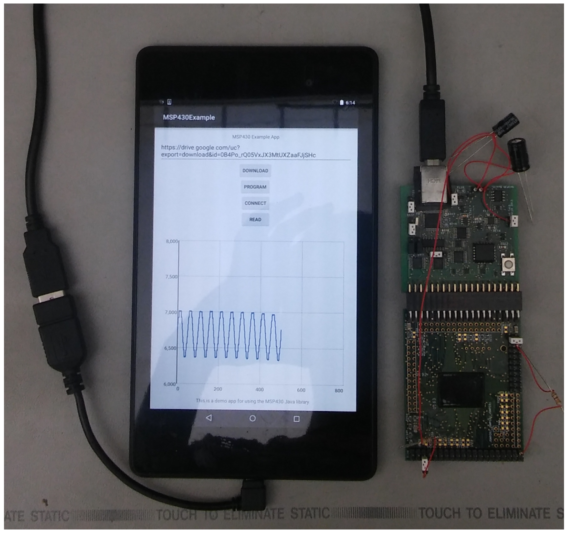
- USB powered and controlled → interfaces like a digital system

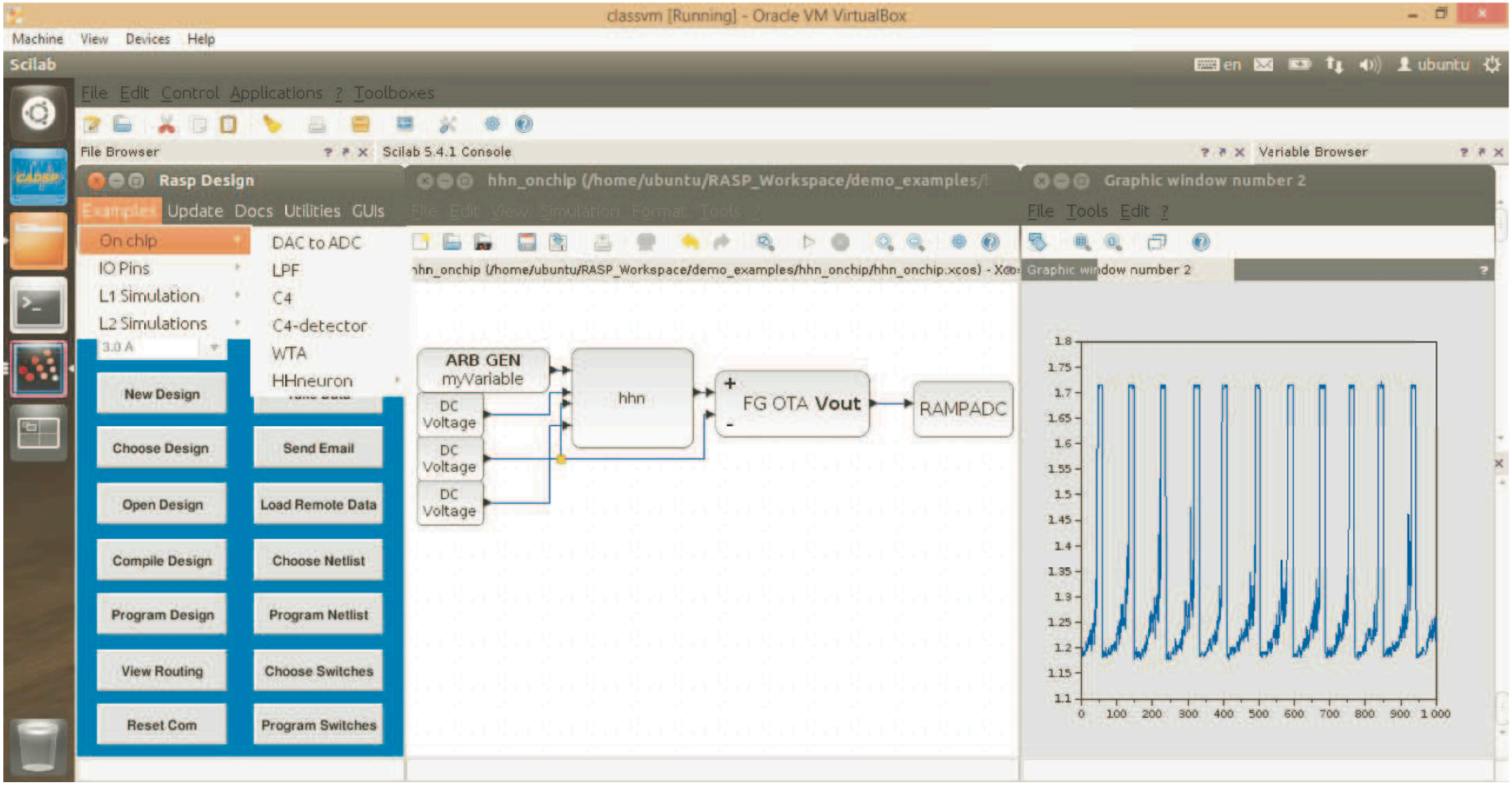
Remote FPAA System



Andriod Tablet FPAAs


6:14
MSP430Example
MSP430 Example App
https://drive.google.com/uc?export=download&id=0B4Po_rQ05VxJX3MtUXZaaFJjSHc
DOWNLOAD
PROGRAM
CONNECT
READ
8,000
7,500
7,000
6,500
6,000
0
200
400
600
800
This is a demo app for using the MSP430 Java library.
TOUCH TO ELIMINATE STATIC

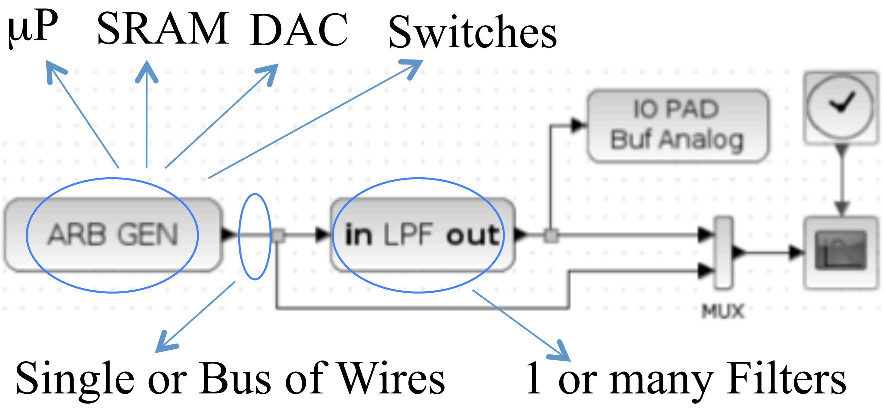
Scilab FPAA Synthesis & Modeling Tool

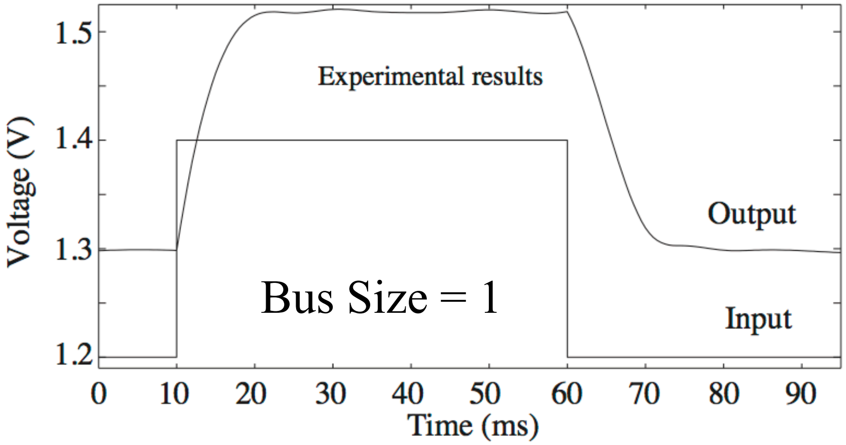


- Encapsulated in Ubuntu 12.04 VM
- Library of Components (low to high level)

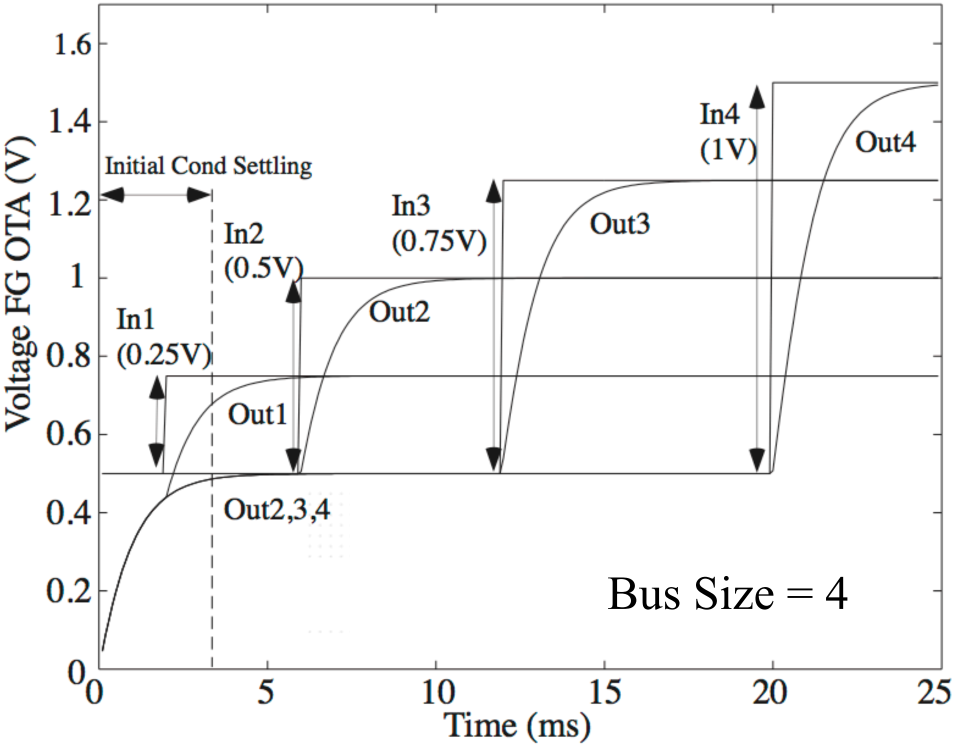
- Measurement transistor channel model of HH neuron

Tool: Measurement and Simulation (LPF)



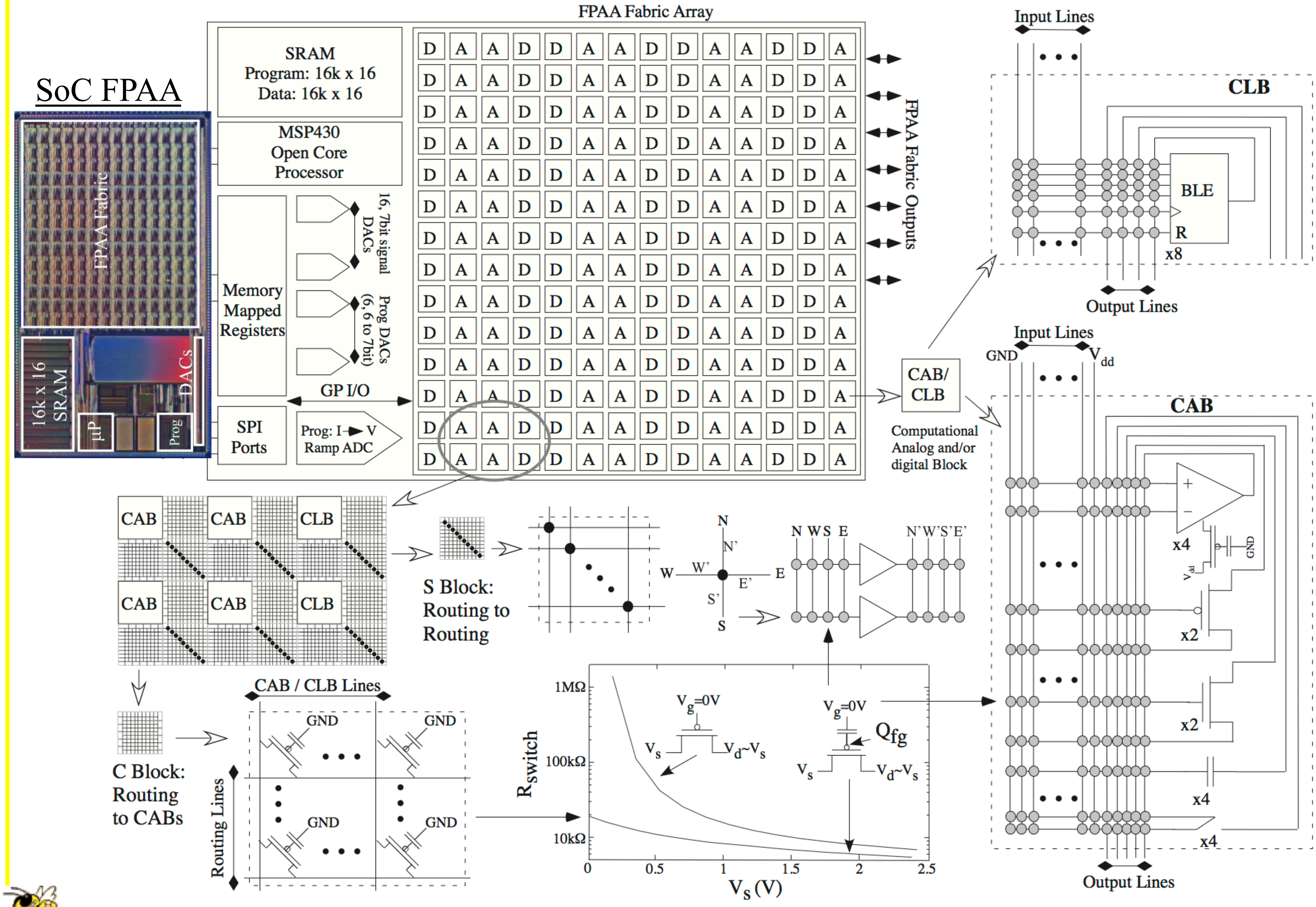


MacroModel Simulation (level = 1)



One toolset to design, to enable high level simulation, and to compile to hardware

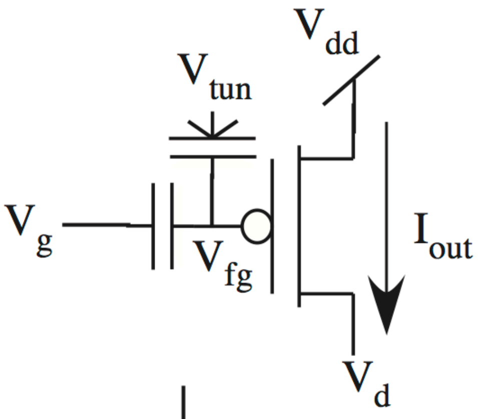
SoC FPAA IC



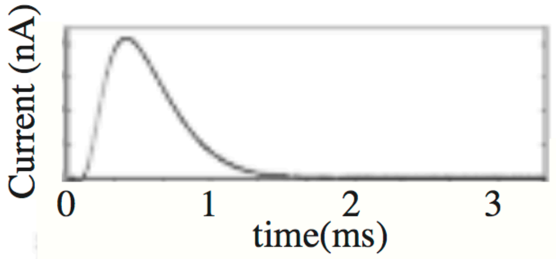
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A
D	A	A	D	D	A	A	D	D	A	A	D	D	A

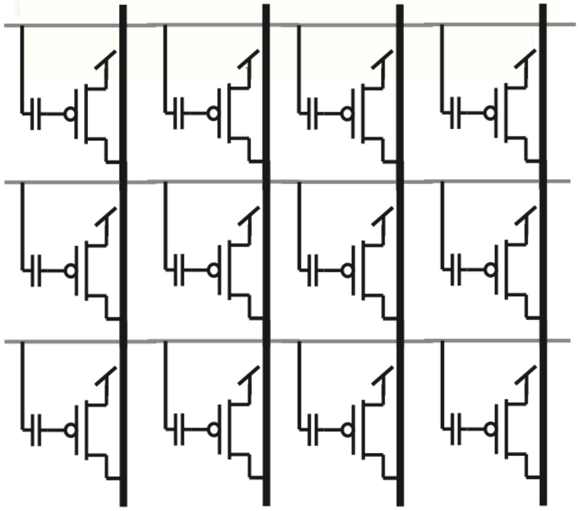
Single-Transistor Learning Synapse

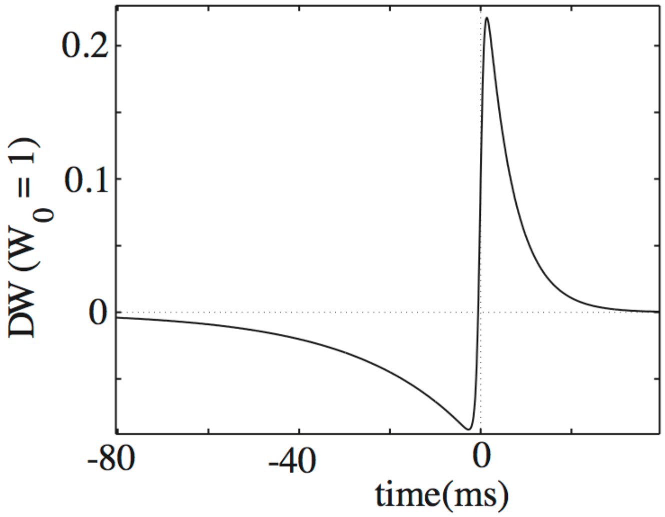
Floating-Gate Circuits: Nonvolilative storage, computation, programmable, adaptable



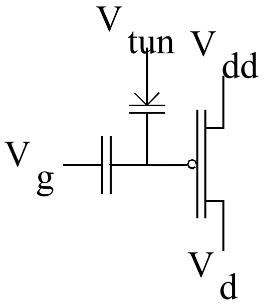
Non-volatile Storage

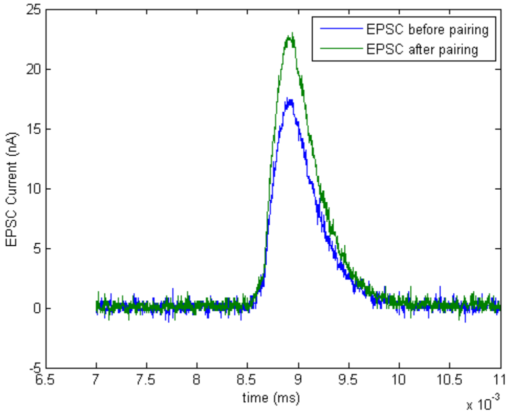






130nm STDP synapse data



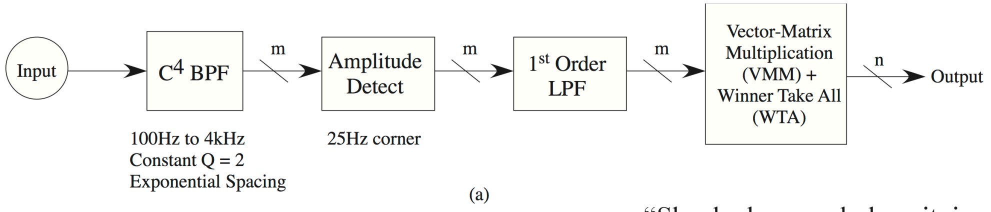


Si CMOS approach can achieve densities while avoiding issues with device integration with Si

[Hasler, et. al, NIPS 1994, BMES 1994, and later papers]

SoC FPAA Classifiers: VMM + WTA for Speech

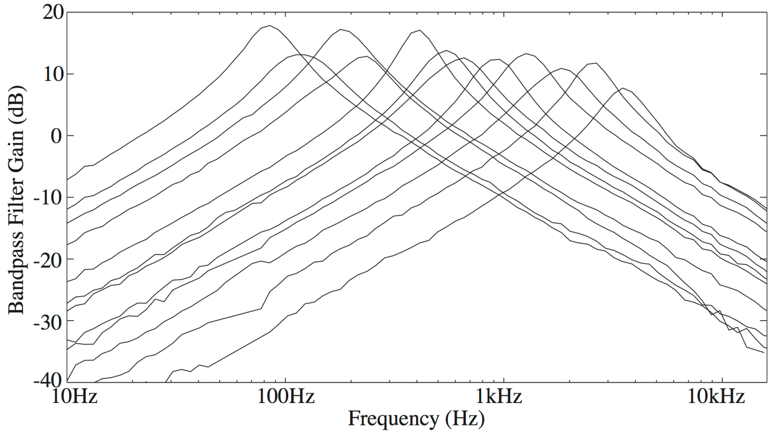
< 23μW

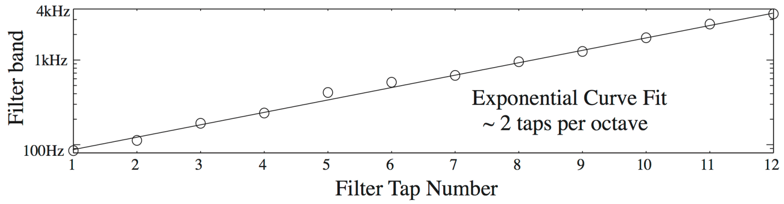

Input → C^4 BPF → Amplitude Detect → 1^{st} Order LPF → Vector-Matrix Multiplication (VMM) + Winner Take All (WTA) → Output

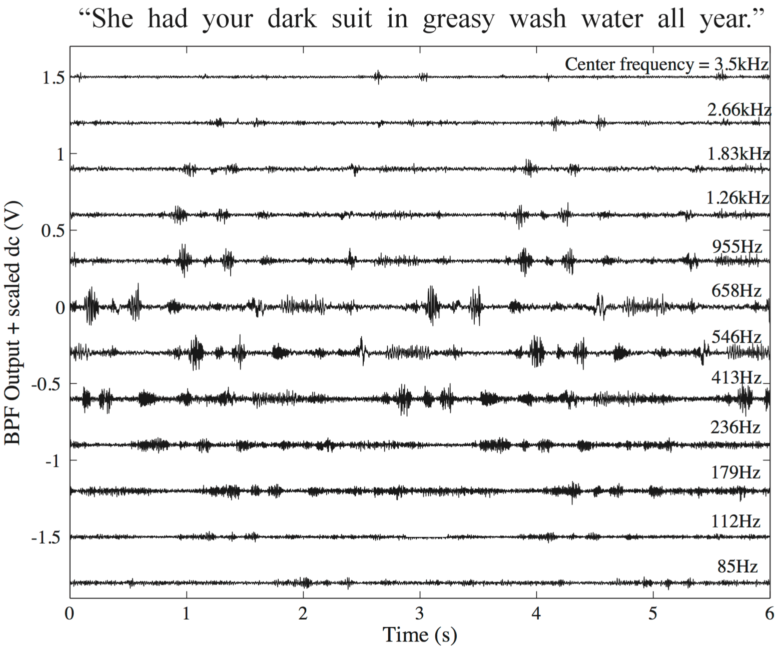
100Hz to 4kHz
Constant Q = 2
Exponential Spacing

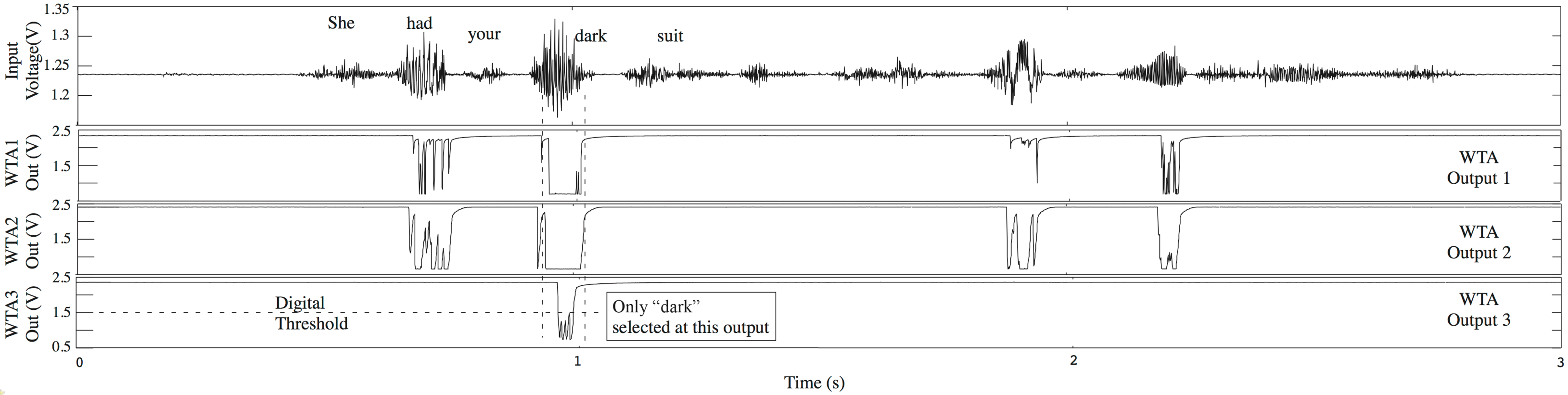
25Hz corner

(a)



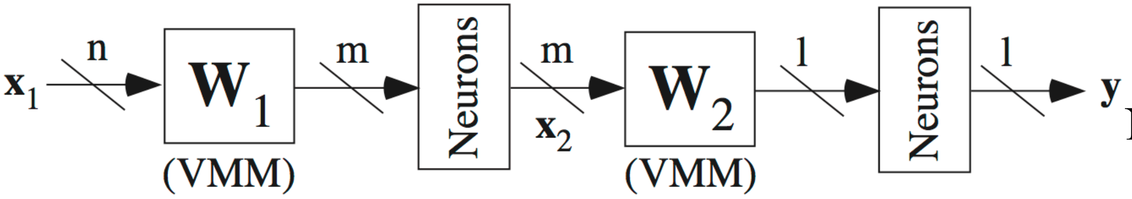






Compiled VMM+WTA Classifier

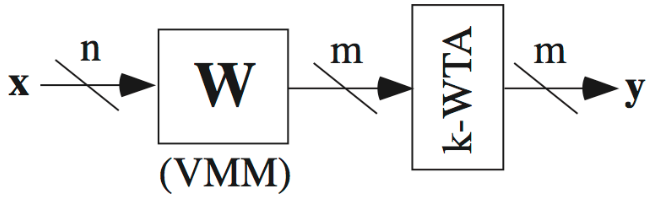
Two-Layer Neural Network (NN) Classifier

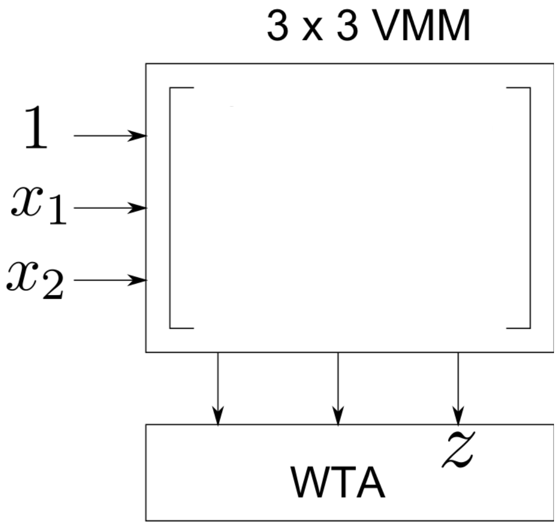


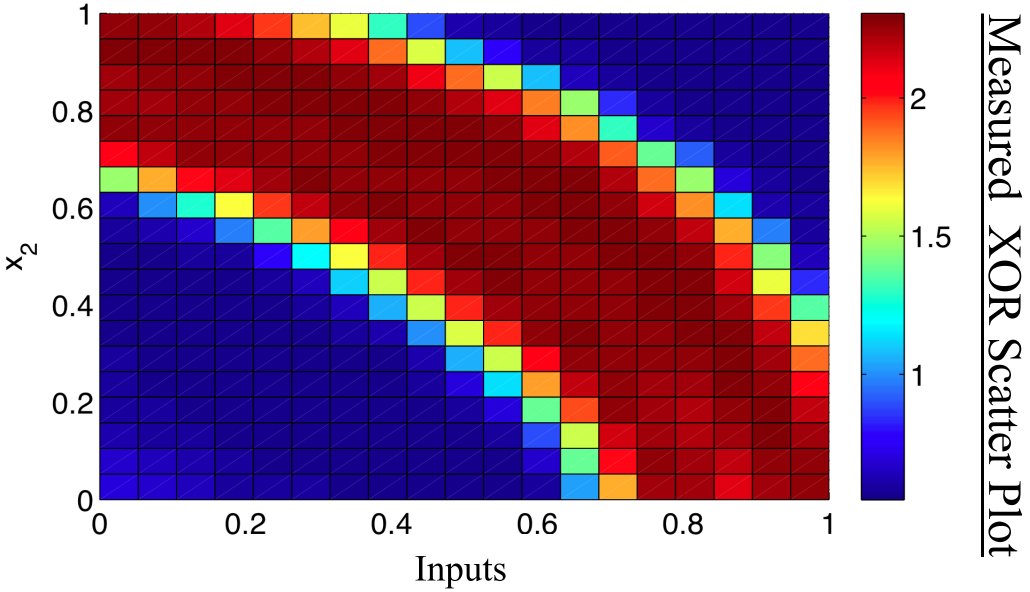
Minsksy 1967: XOR classification requires more than one layer

NN was silenced for 15 years
Could solve in **two** layers

VMM+WTA Classifier





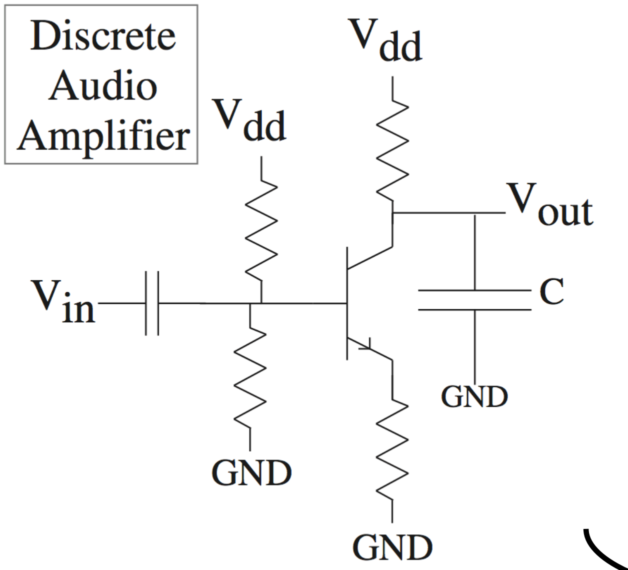


Analog, n-WTA single layer block can be a universal approximator [Maass, et. al, 2000, Ramakrishnan, et. al, 2013]

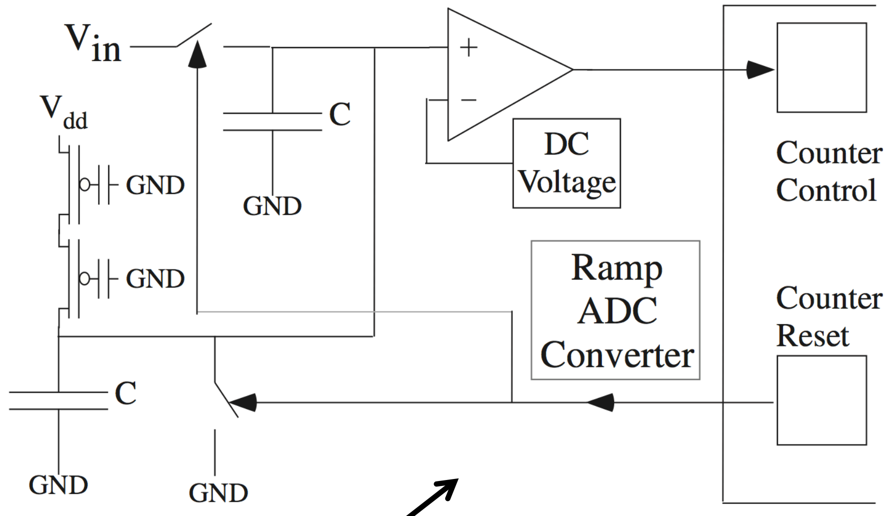
FG devices used to eliminate mismatch

SoC FPAA devices → Rethinking Circuits Education

Classical First Semester

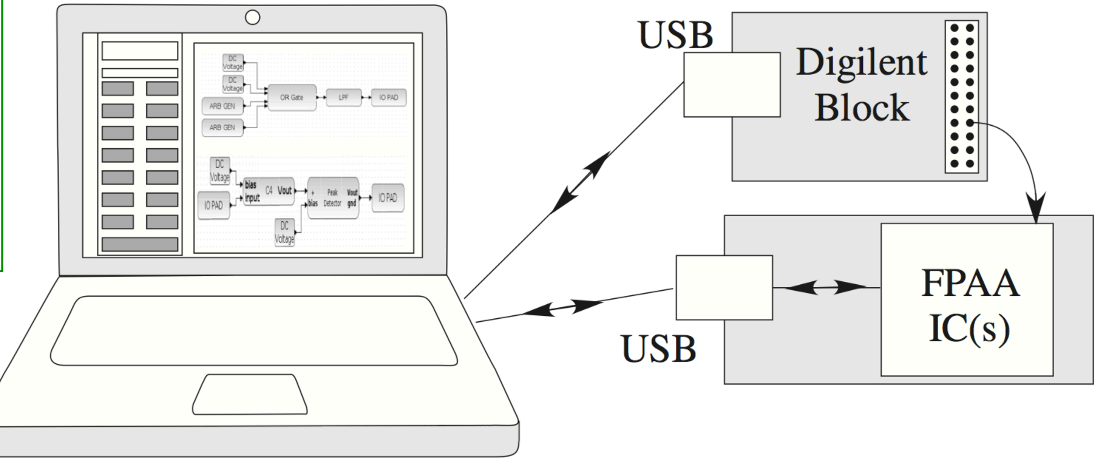


System-Focused First Semester



Transforming Low-Level Circuits to Systems Employing FPAAs

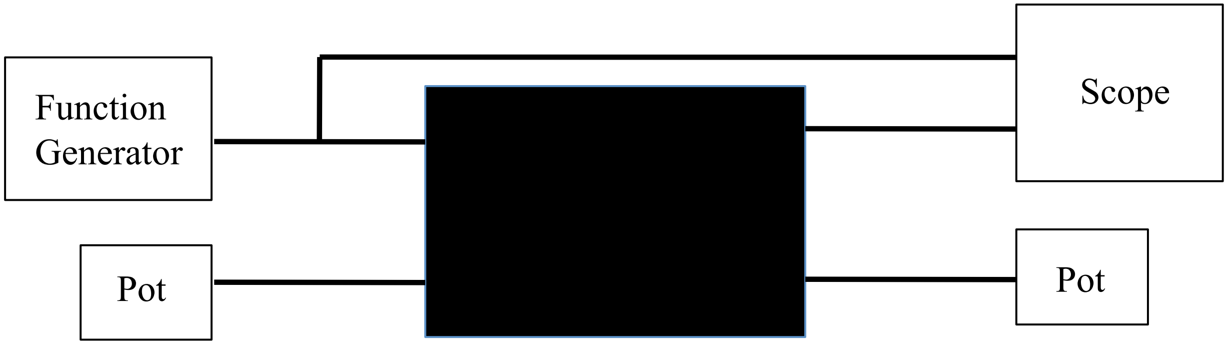
Other Classes:
AVLSI, IC Design
IC Dynamics



Enables moving techniques outside of circuits, possible for student projects.

CNS 182 (Caltech) Black Box Exam

(1989 – 1996)



Custom IC in a breadboard

No hints of circuit given

"Something related to what was done in class"

Typically 35 to 60% figured out the circuit

- Functional Circuit when entering lab doing "something"
- Two hours to take as much data as possible
- Four days to write up results
- Identify the resulting circuit

Certain similarities for extracting circuit knowledge. Some important differences:

- we have the circuit netlist (sometimes)
- we have the expected inputs and outputs (sometimes)
- one might have some indication of the proper function of the IC

Black Box Exams to Explore IC Verification

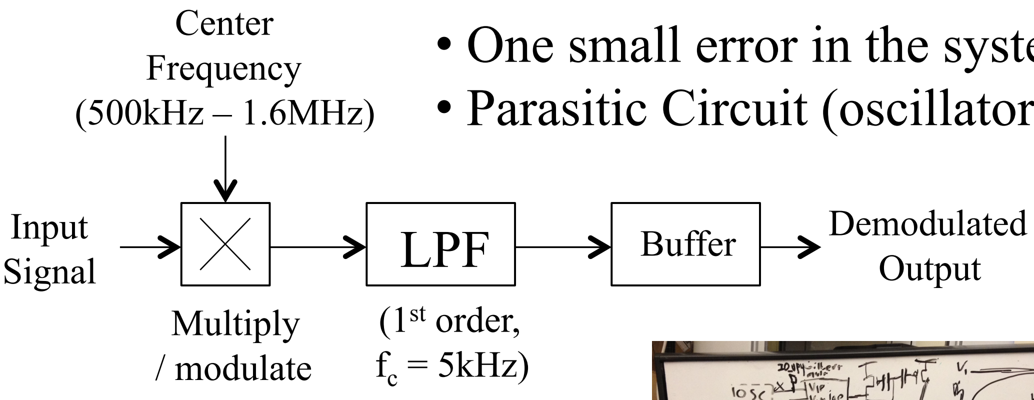
BB1	BB2	BB3	BB4	BB5	BB6
Basic analog Amplifers / mux	AM Demod (hidden circuit)	DAC: 5bit R-2R 3bit V-mode 8bit total	Low-Freq Receiver (Trasceiver block)	DAC (7bit) Controlled VCO	Mux DAC 2 in, 2 out, 1 DAC
Only IC	Switch list Hiearchy	Spice nelist (100 parts) Transistors + T-gates, caps 3 OTAs)	Spice nelist Transistors +T-gates, switches	Spice nelist Transistors +T-gates, switches	Spice nelist Transistors +T-gates, switches
DC I/V	Switch List Analysis (DC I/V #2)	Low-level Netlist analysis	Basic netlist tools, clustering	Basic netlist tools, clustering	Basic netlist tools, clustering
3 teams(6) 8+ hours	2 teams(6) 4 hours (each)	2 teams(6) 7-8 hours	2 teams (8) 6,8 hours	2 teams(6) 5-6 hours	2 teams (6) 4-5 hours

- knowledge the device was on an (circa 2010) FPAA, by those who designed the IC
  - no need to explain infrastructure
  - understood what primitives where possible (all identify IC)

Result from Black Box Exercise?

BB2: AM Demodulator

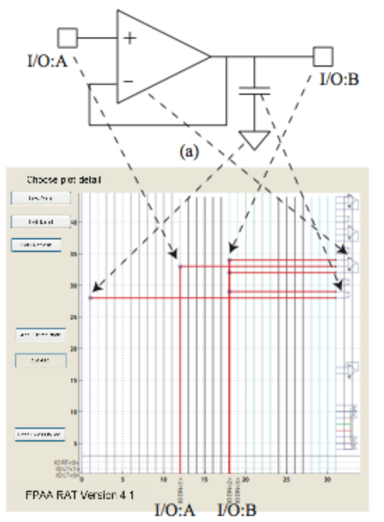
- One small error in the system
- Parasitic Circuit (oscillator)

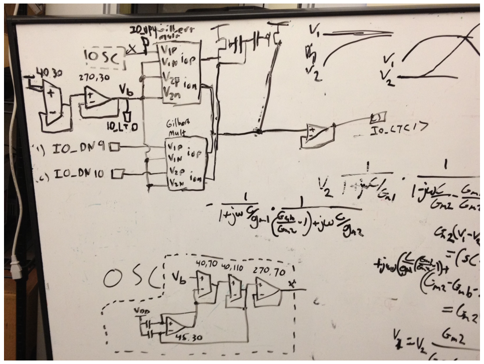


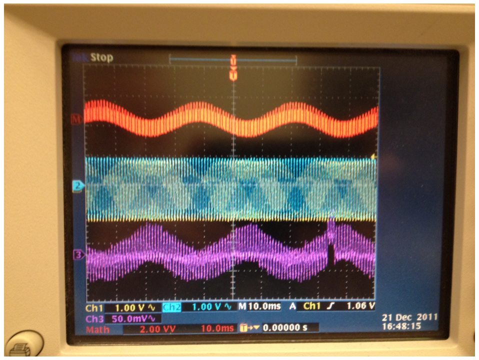
Switch List

275	34	1.8
275	35	1.8
0	7	1.8
276	24	1.8
277	25	1.8
0	52	1.8
1	51	1.8
3	45	1.8
279	62	1.8
3	88	1.8
279	81	1.8
36	8	1.8

RAT tool used heavily







Extract an unknown system

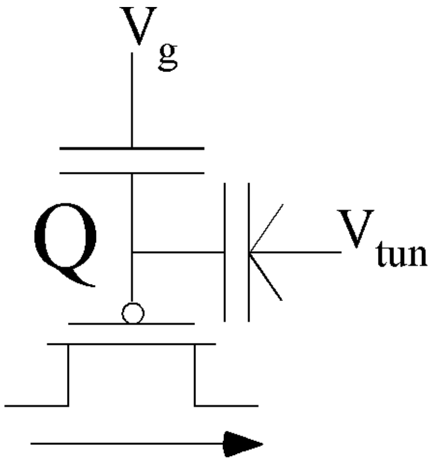
After (4) students trained (3 days, 2 days to write report):

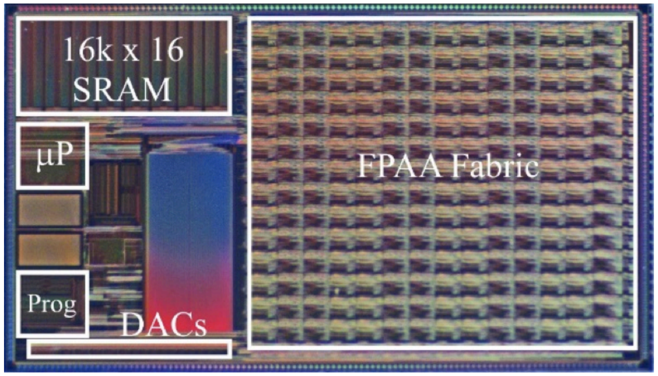
- Custom IC built (not us)
- Found: 4 interleaved DACs, 10GSPS DAC, PLL, Registers, digital control, on-chip oscillator (only pins)
- Only 1 DAC populated, multiple digital no connected (and other errors)
- Were to have electrical info, none obtained (raw delayering, no n or p)
- VCO error: GND on core transistor circuit (not working)

Good Security FPAA Aspects

Program entirely stored in FG

- Floating-Gate (FG): nonvolatile memory, 10 year lifetime
- No SRAM loading vulnerability
- Analog values hard to measure without disturbing significantly
- Digital computation can be encoded with analog
- Low-power circuits are hard to externally measure
  (side channels, transistors don't light up)
- Can enable some self-destruct mechanisms on tampering

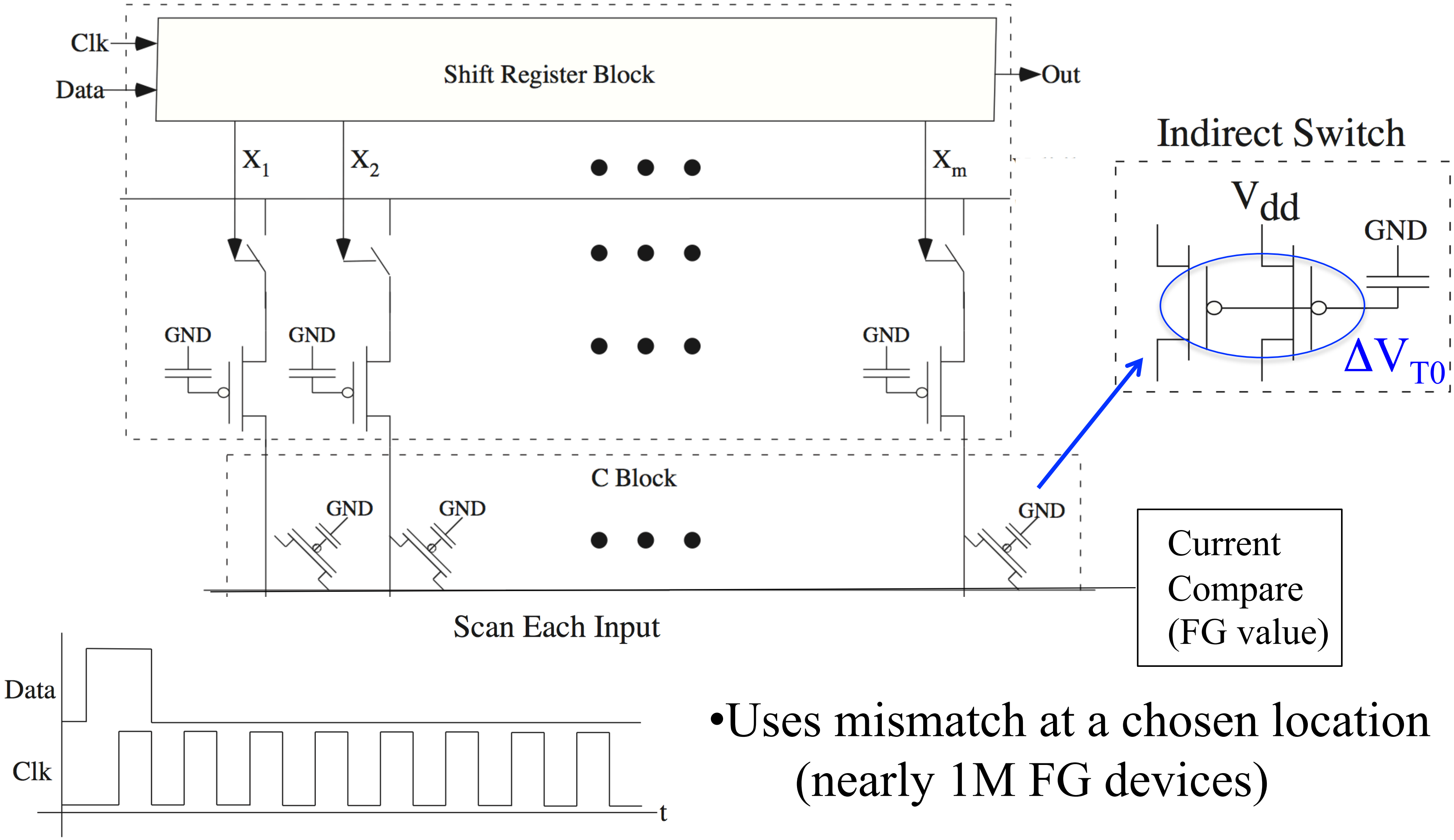




FPAA structure for Secure Design

- FPAA structure is generic, general, and generally known
- Every node can be measured, therefore find if trusted
- Secure code can be programmed in a secure space
  (Analog or Digital)
- Programming code is not the IC (µP)
- Layout says almost nothing about function

Unique Functions in FPAA IC Device



- Uses mismatch at a chosen location (nearly 1M FG devices)

Security of Initial Network Nodes

10mW, 1mW, < 1mW average energy, Radio on infrequently

Potentially Class 0, 1, or 2 *size* systems for communication

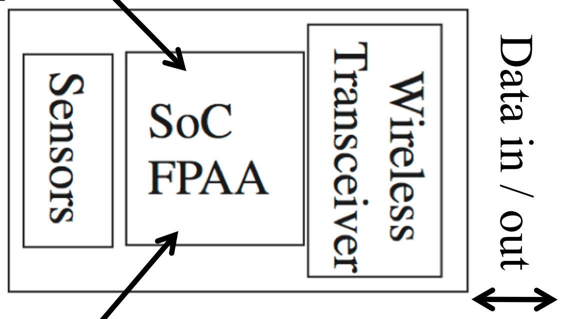
Physical FPAA attack:

- obtain device
- avoid self-destruct
- avoid charge loss in read
- reconstruct IC function
- find mismatch

Biggest Risk: Transceiver Port

might allow reprogramming

Secure key (e.g. PUFs), some encryption essential for security



Typical ICs or on FPAA

Always Requesting attack:

- Drain battery life
- Receiver modulation codes, (known keys, waveforms)
- Ultra-low power RF sensing

Pretend to be host

FPAA attack, I/O pins:

- take I/O port to get control
- question of programmed ports (USB, SPI)
- Attempts to stall computation

Further analog classification to identify particular attack strategies that get through

Secure FPAA Network Nodes

- FPAA enables Physical Computing → computing opportunities
  (ultra-low energy, small)

- Creates potential security issues & opportunities:
  Can we have an ultra-low power secure system?

- Opportunities for 10mW, 1mW, < 1mW nodes,
  low digital memory. Security?

- Approaches are accessible for educational spaces,
  already utilized for teaching

Open questions / opportunities moving forward