

# Other Rogues - Analog, Reconfigurable, and Neuromorphic Accelerators and Related Tools

FPAAC Content provided by Dr. Jennifer Hasler, ECE and CRNCH faculty

CREATING THE NEXT MOORE'S LAW



Center for Research into  
Novel Computing Hierarchies

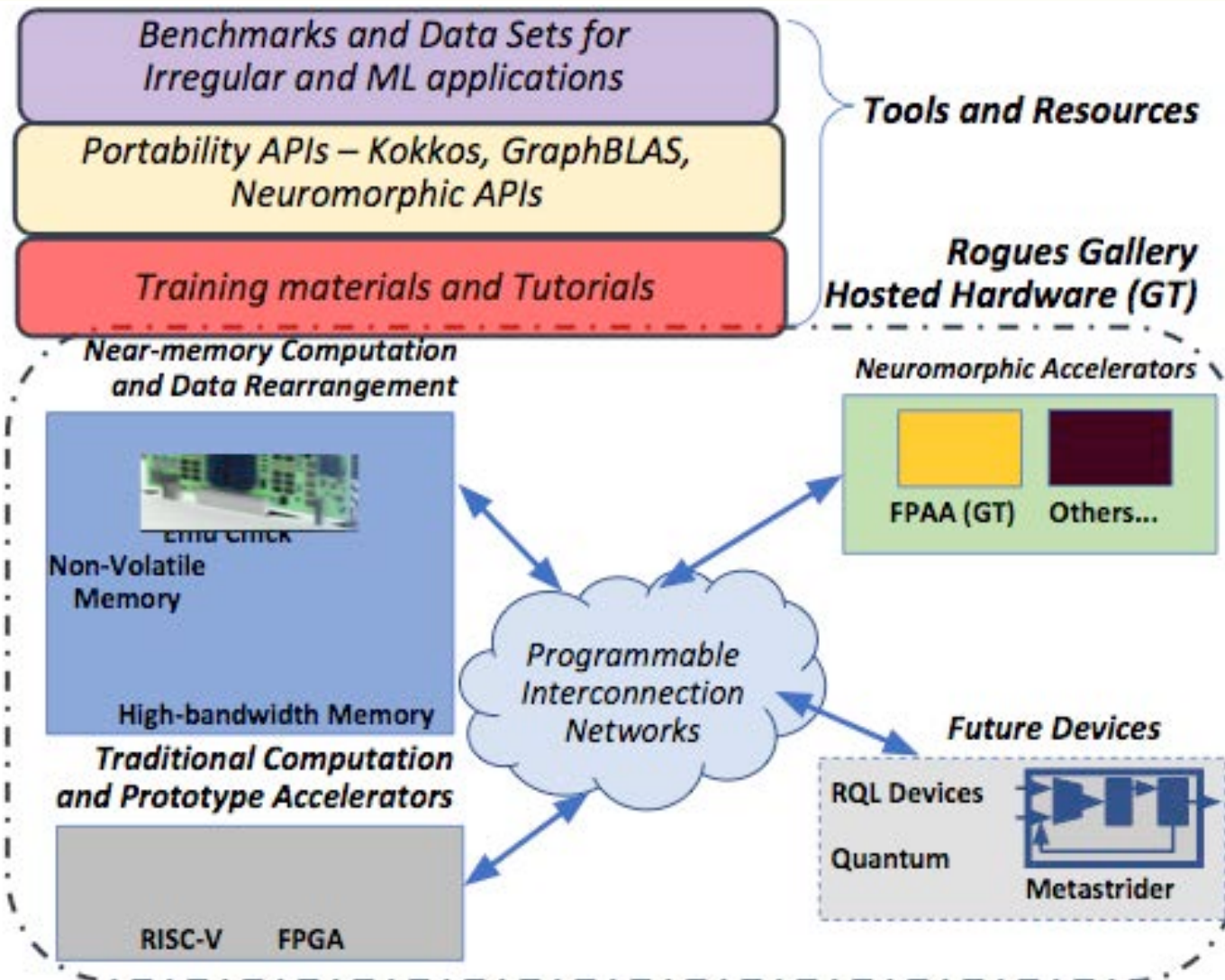
For more information, visit our website at  
[www.crnch.gatech.edu](http://www.crnch.gatech.edu) or send email to  
[crnch@gatech.edu](mailto:crnch@gatech.edu)



# Outline

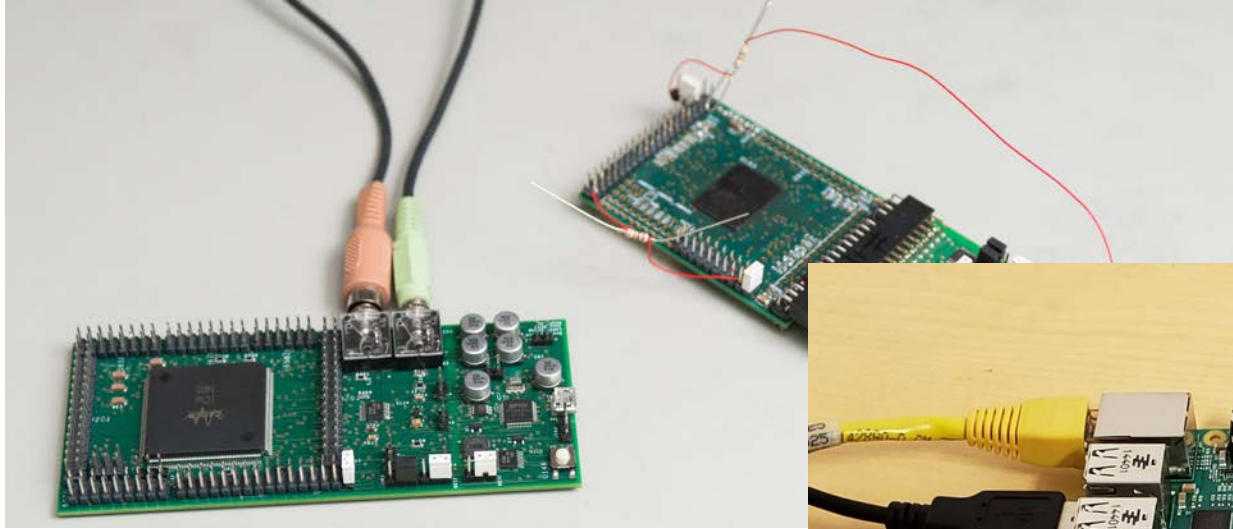
- Neuromorphic Computing
  - Field Programmable Analog Arrays
  - DANNA
  - Other future platforms
- Reconfigurable Computing
- Benchmarking and Tools

# Rogues Gallery Recap

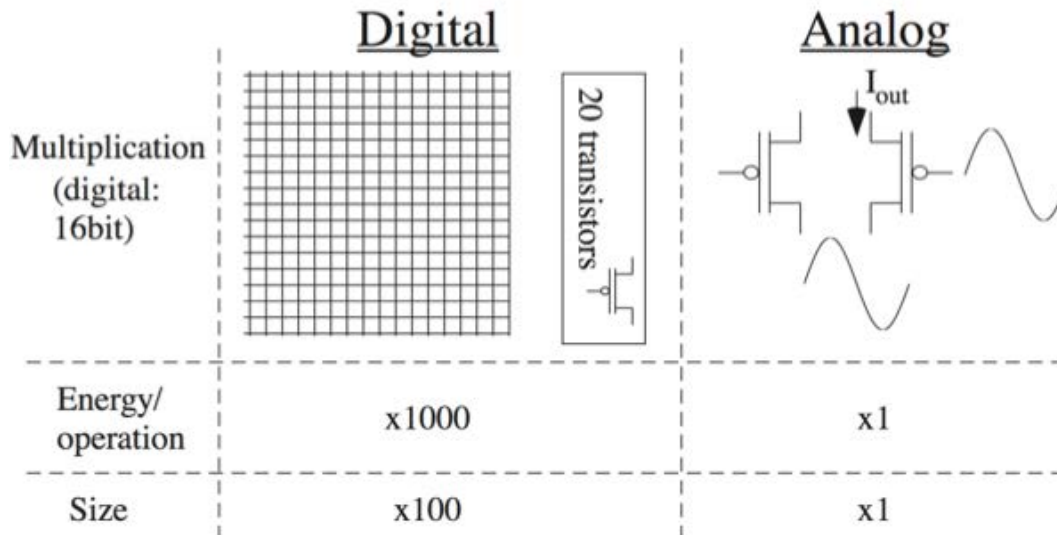




# Field Programmable Analog Array (FPAA)

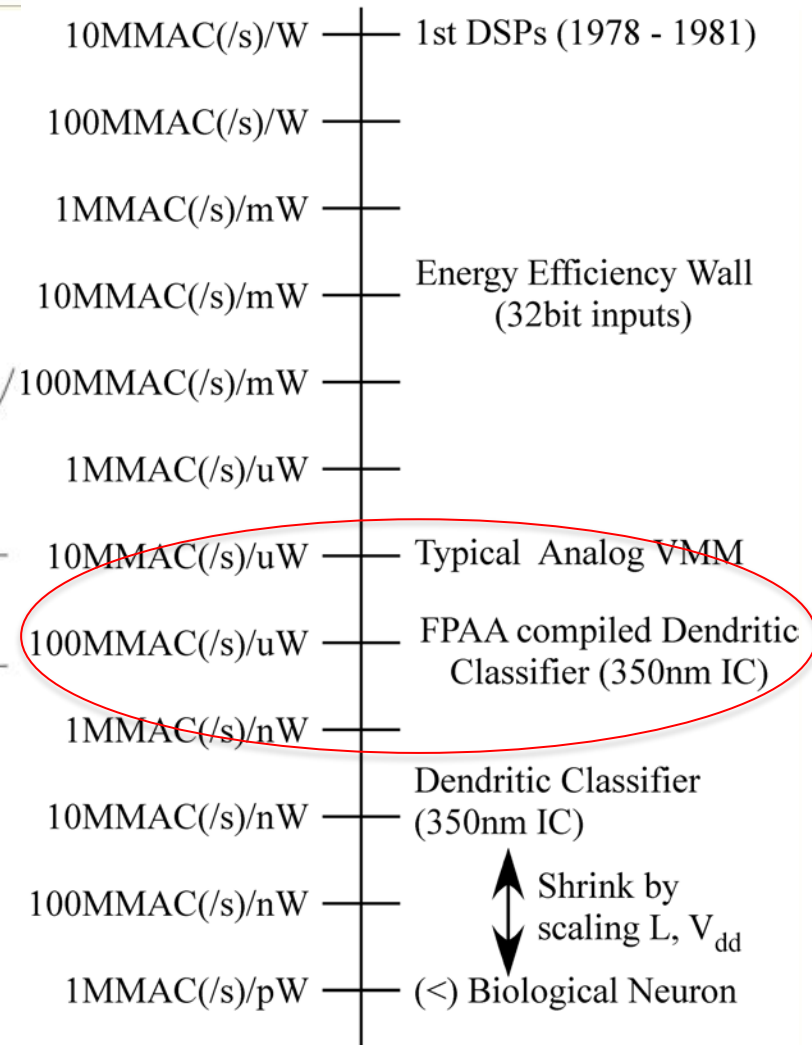


# Why Physical / Analog Computing?



Expansion of C. Mead's original hypothesis, 1990

## Power Efficiency Scaling

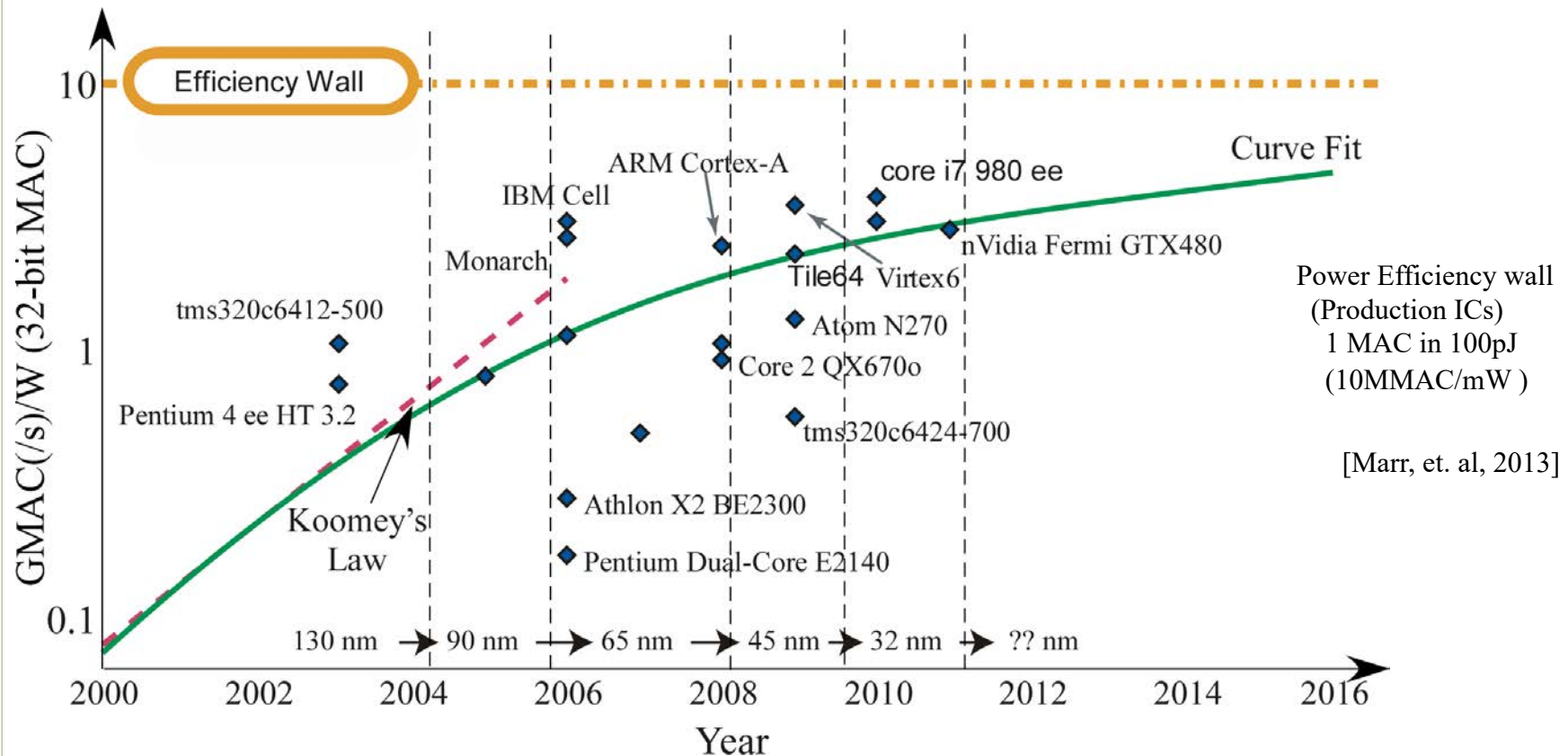


[Hasler and Marr 2013]

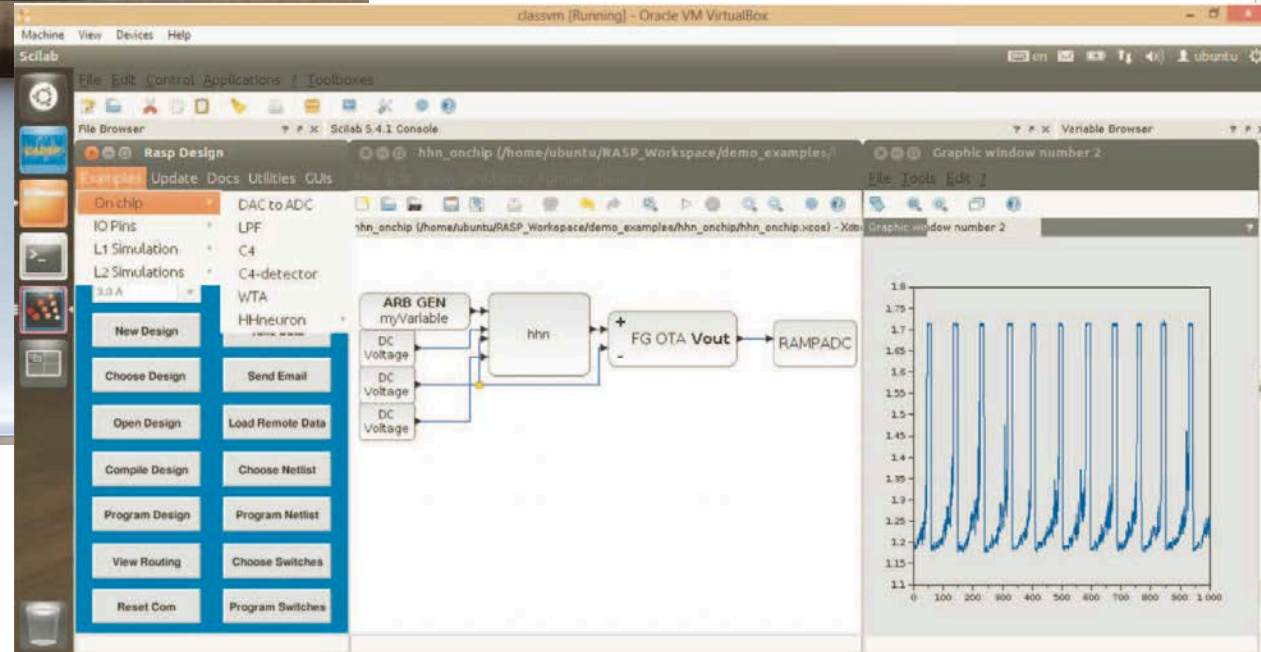
# Why Physical / Analog Computing?



## Digital Hitting Limits of Power Efficiency – Koomey's Law



# FPAAs enable Ultra-low Power Computing



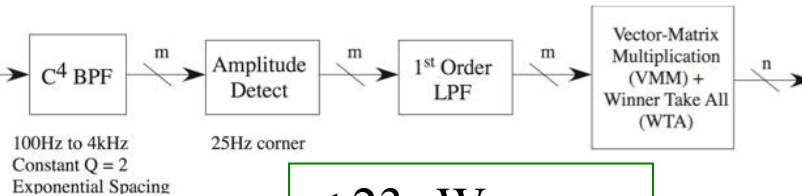
The FPAAs provide a mixed analog/digital platform with a general and configurable design that can be used to create secure and legacy resistant mixed-signal devices



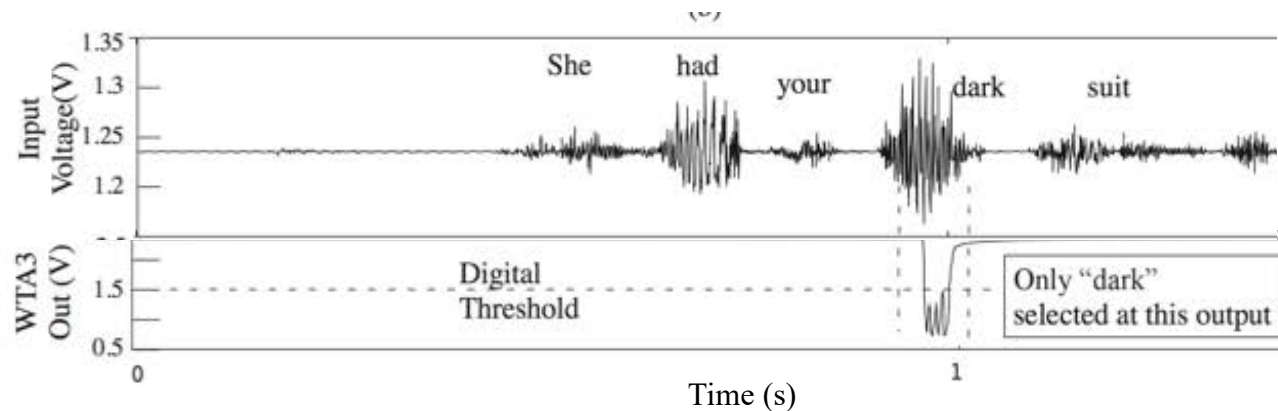
# FPAA Classification Task

Acoustic  
Classification

## Ultra-low energy computation



$< 23\mu\text{W}$  power



Embedded learning & classification:  $20\text{-}30\mu\text{W}$   
on full, 1s Nzero database (GOMAC 2016)



# FPAA Classification to Date



Built on Analog Vector-Matrix Multiplication (VMM), including in routing, Winner Take All (WTA), etc.

Initial Command-Word Classification (Hand-tuned Weights) (2015)

Biomedical classifier (Hand-tuned weights) and computational measures (Knee sounds, Heart monitoring)

Speech – No-Speech Detector (Hand tuned weights)

Acoustic Classification and Learning:

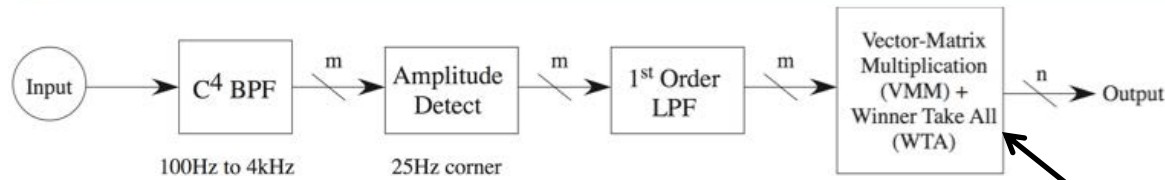
- Developed Theory and Training Algorithm for VMM+WTA Classification
- Trained using modified Lincoln Lab's acoustic data set for Nzero program.
- Yielded correct detection on all sets

Current work on Image classification (2-D sensor inputs)  
2-layer NN or DNN are possible...

- VMM+WTA has worked well so far
- Can develop modular spiking (HH neuron) networks

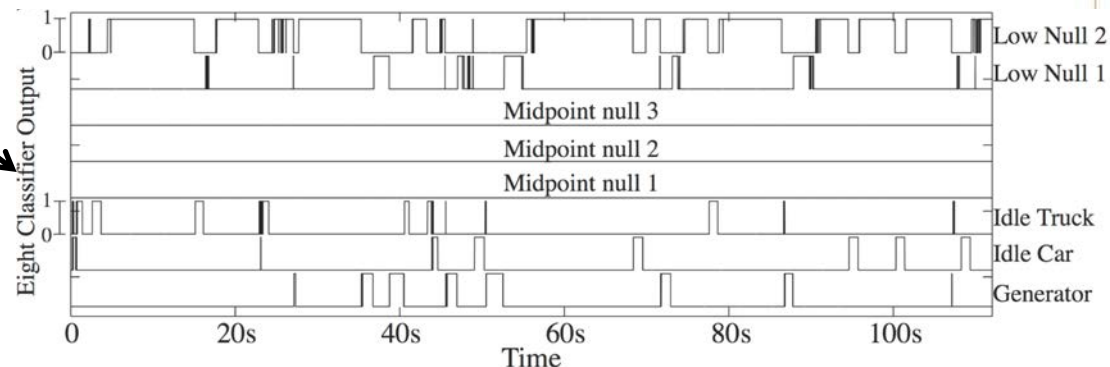
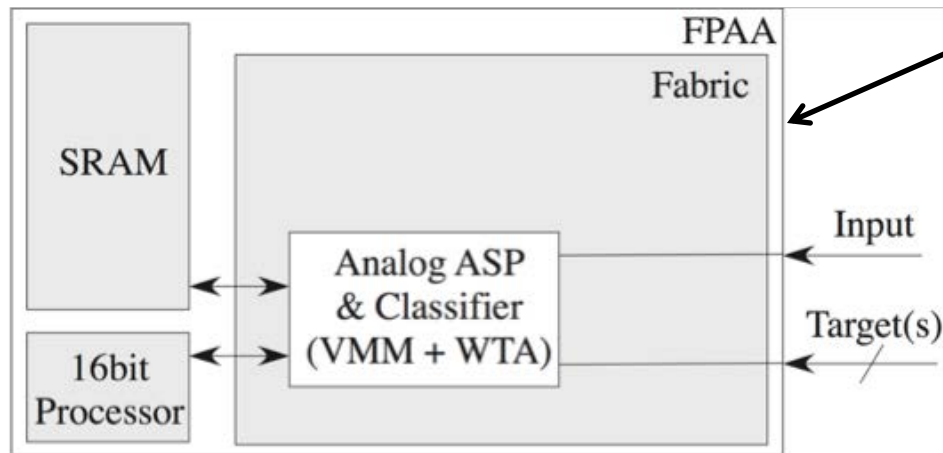


# More Analog Classifier (VMM+WTA)



Analog, n-WTA single layer block = universal approximator (2 layer NN)

Classification + On-Line Training

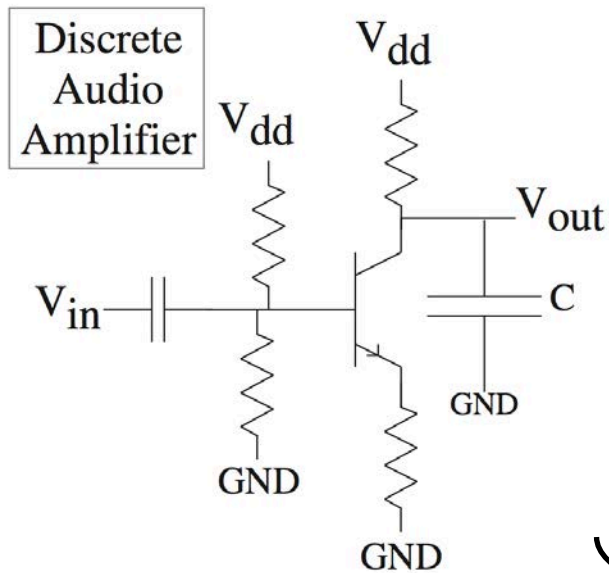


100% classification

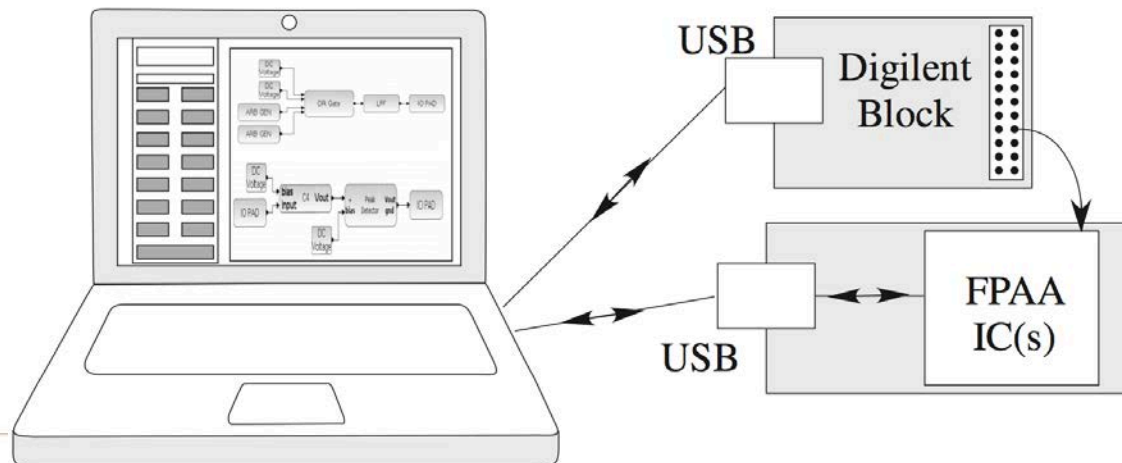
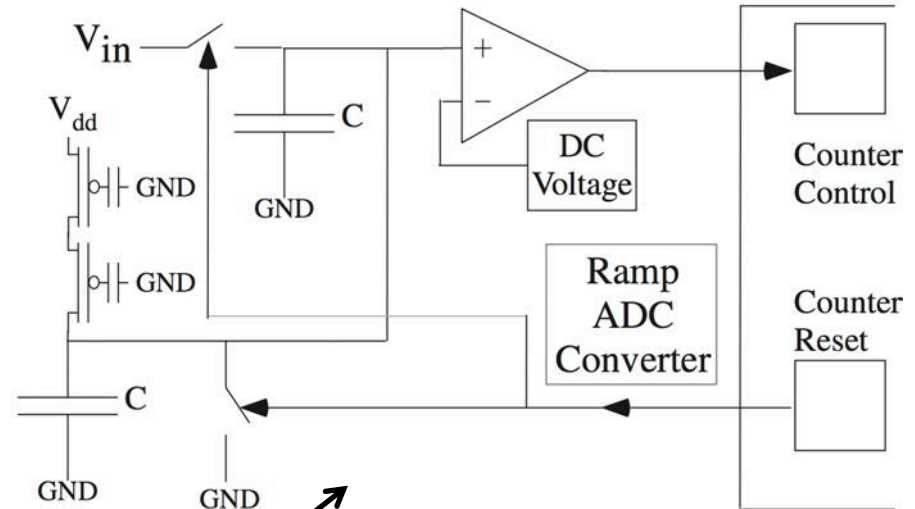
# SoC FPAAs in the Classroom



## Classical First Semester



## System-Focused First Semester



search into  
ing Hierarchies

# FPAA Toolflow and Examples (1)

The screenshot displays the Scilab 5.4.1 environment with the Rasp Design toolflow active. The Rasp Design dialog box is centered, showing options for creating a new design, choosing a design, opening a design, compiling, programming, and viewing routing. The Variable Browser on the right lists various variables and their types. The Command History window at the bottom right shows the execution of the 'quit' command.

**Rasp Design Dialog Box:**

- Examples Update Docs Utilities GUIs
- Rasp Design
- Enter your email ID
- No File Selected
- Choose Board (dropdown) Enter Chip Number
- New Design Take Data
- Choose Design Send Email
- Open Design Load Remote Data
- Compile Design Choose Netlist
- Program Design Program Netlist
- View Routing Choose Switches
- Reset Com Program Switches

**Variable Browser:**

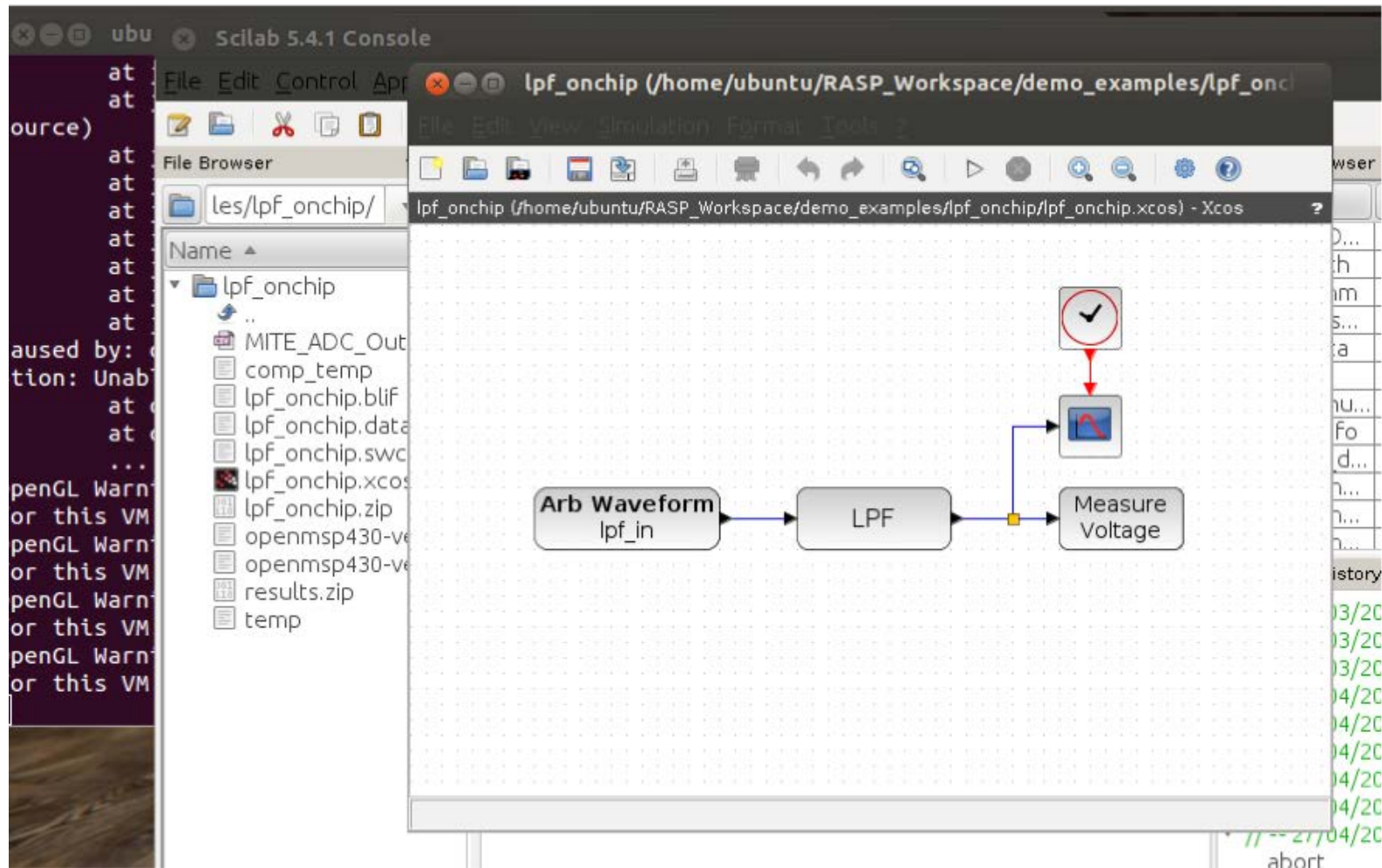
Name	Dimen...	Type	Visibility
%MOD...	1x1	String	local
csvpath	1x1	String	local
addvmm	N/A	Double	local
rm_res...	N/A	Double	local
csvdata	N/A	Double	local
plcvpr	N/A	Double	local
pass_nu...	N/A	Double	local
cap_info	N/A	Double	local
show_d...	1x1	Double	local
gui5_m...	1x1	Graphic ...	local
gui2_m...	1x1	Graphic ...	local
gui8_m...	1x1	Graphic ...	local

**Command History:**

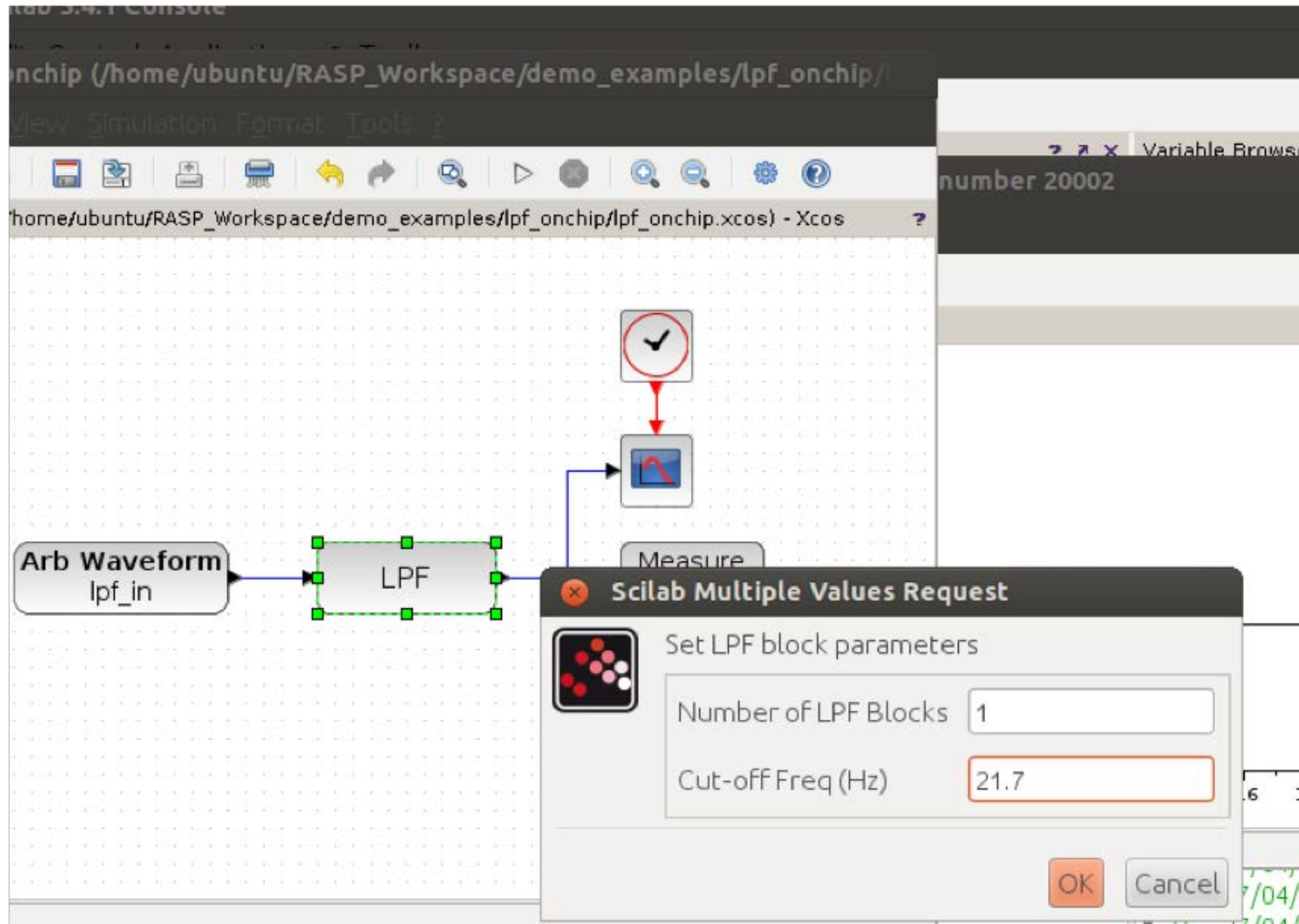
```
quit
// -- 29/03/2018 15:21:37 -- //
// -- 29/03/2018 15:31:26 -- //
// -- 29/03/2018 15:32:36 -- //
// -- 24/04/2018 10:39:02 -- //
// -- 24/04/2018 10:50:37 -- //
// -- 27/04/2018 11:06:11 -- //
// -- 27/04/2018 11:06:40 -- //
// -- 27/04/2018 11:38:26 -- //
// -- 27/04/2018 12:17:22 -- //
abort
// -- 06/02/2019 12:08:31 -- //
// -- 08/02/2019 15:14:13 -- //
```



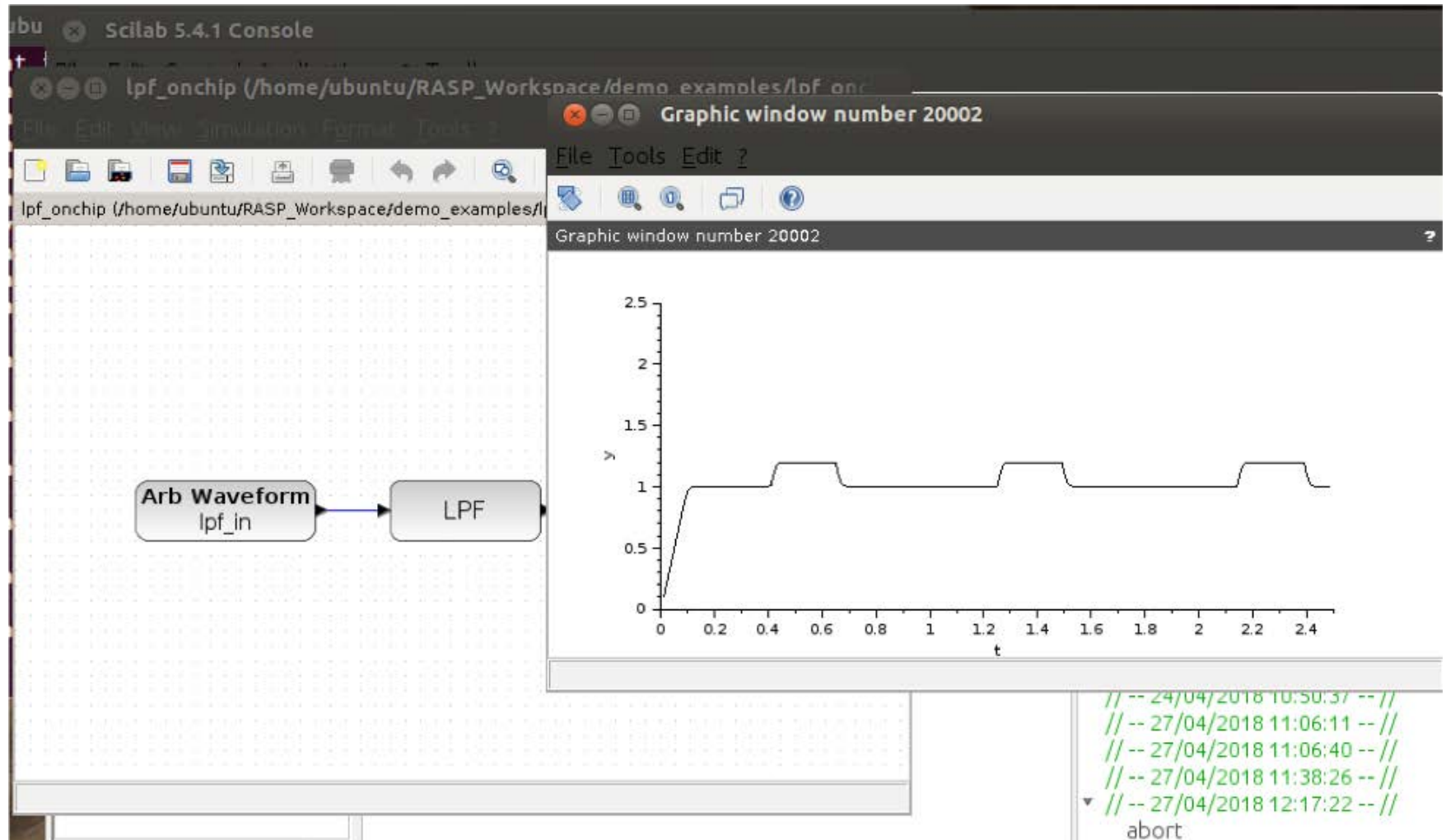
# FPAA Toolflow and Examples (2)



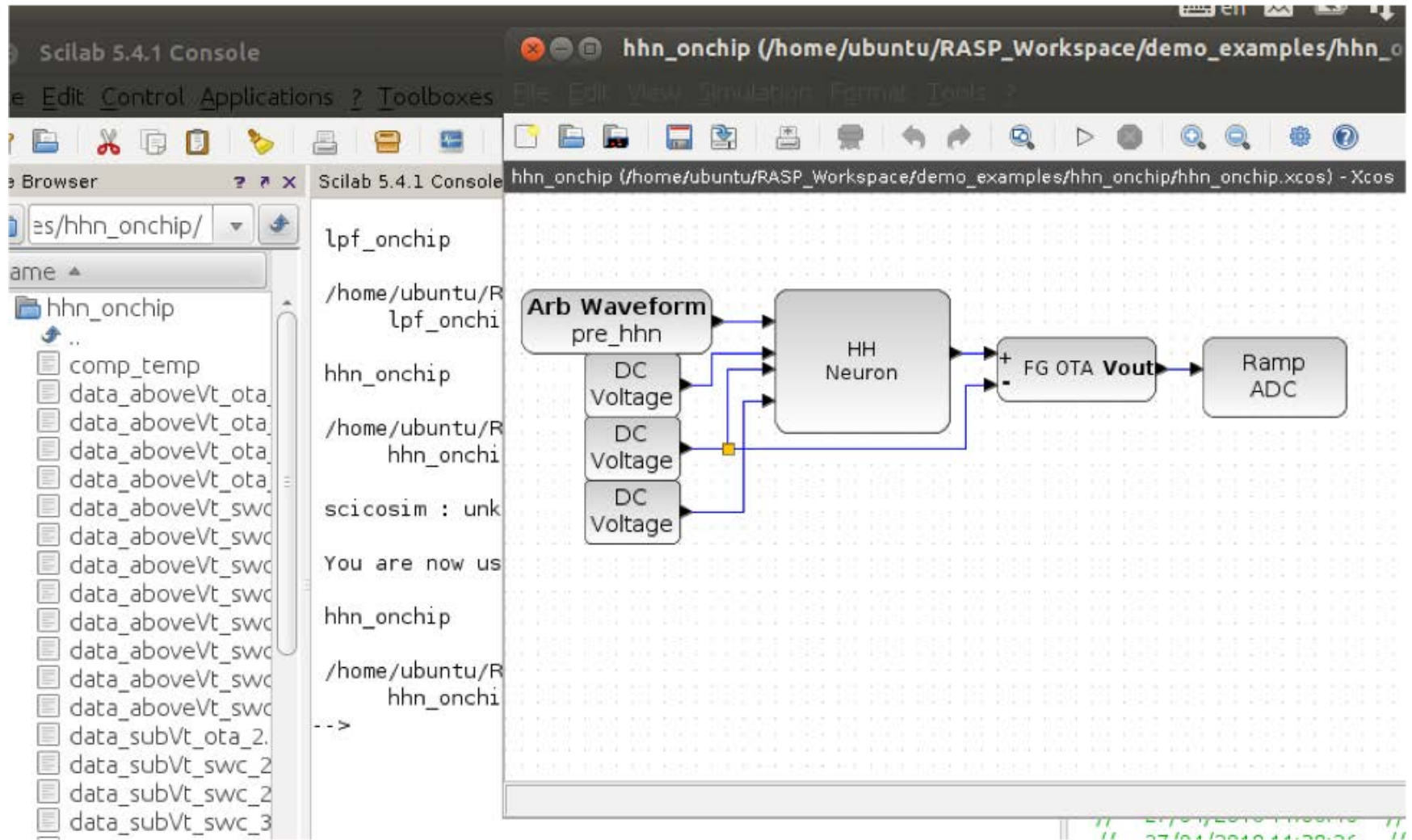
# FPAA Toolflow and Examples (3)



# FPAA Toolflow and Examples (4)

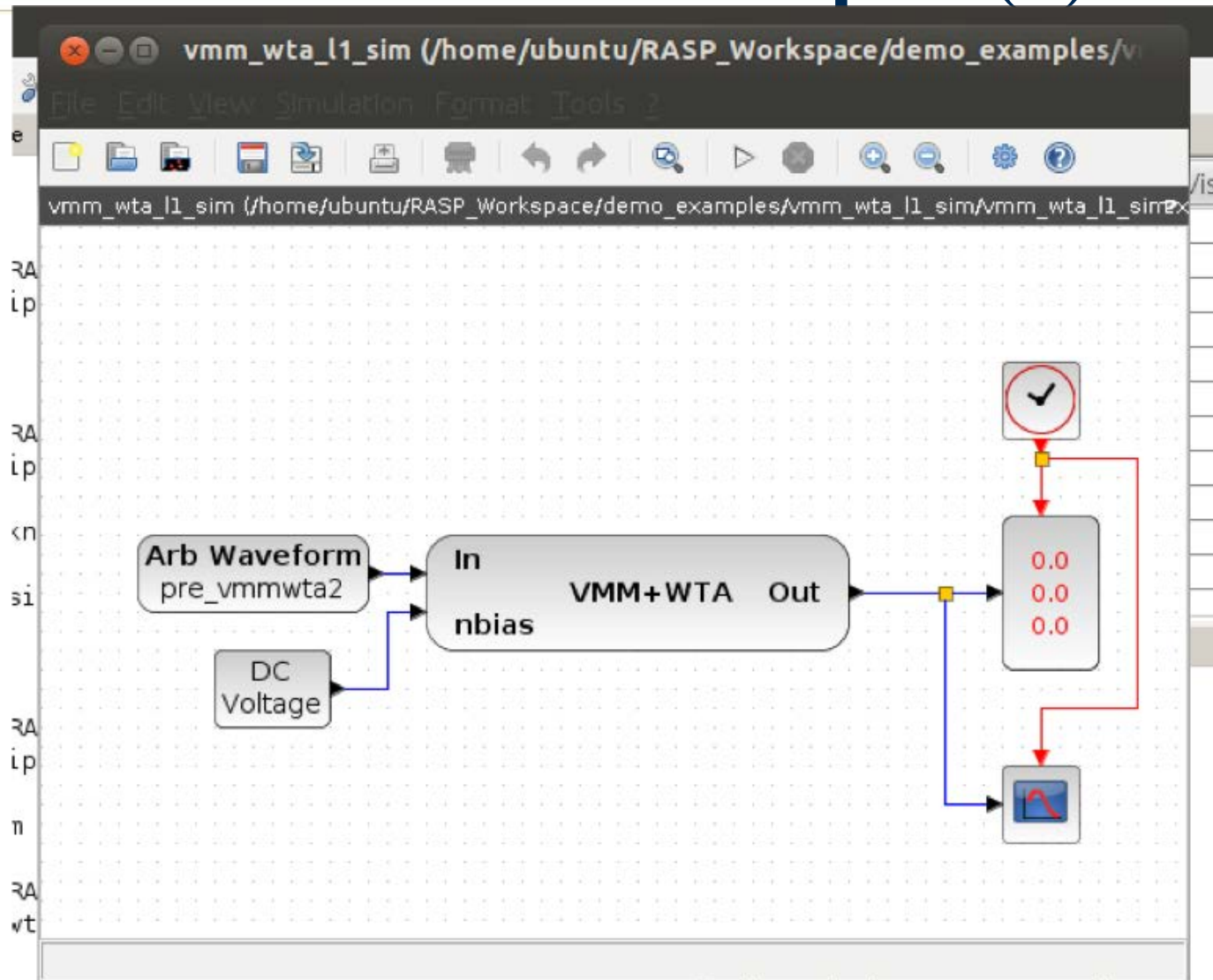


# FPAA Toolflow and Examples (5)





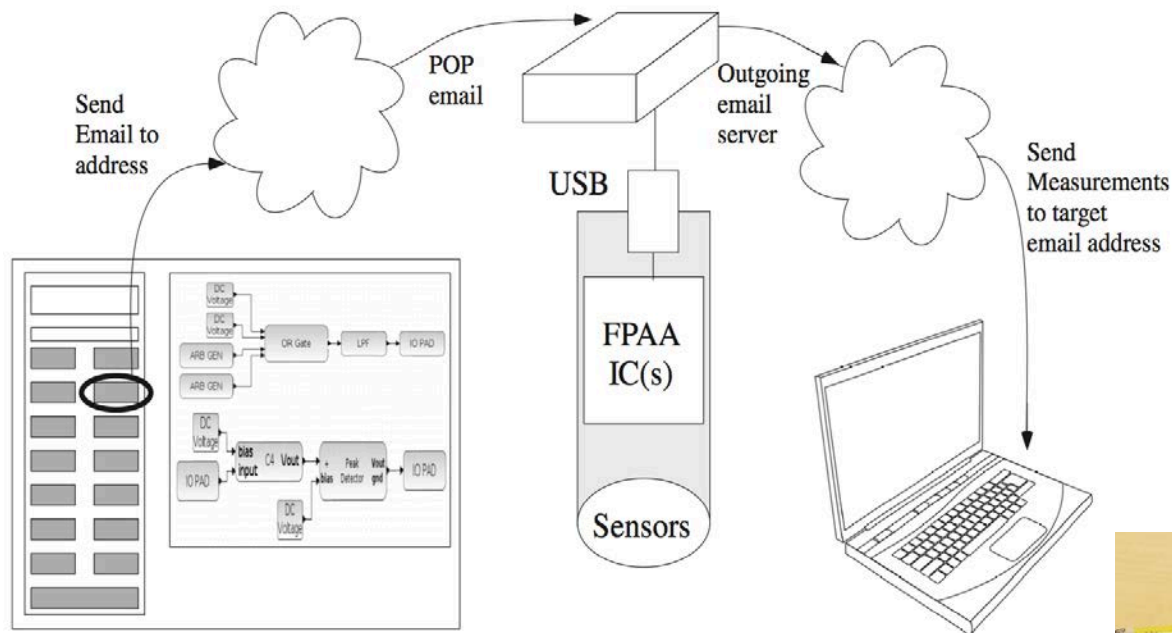
# FPAA Toolflow and Examples (6)



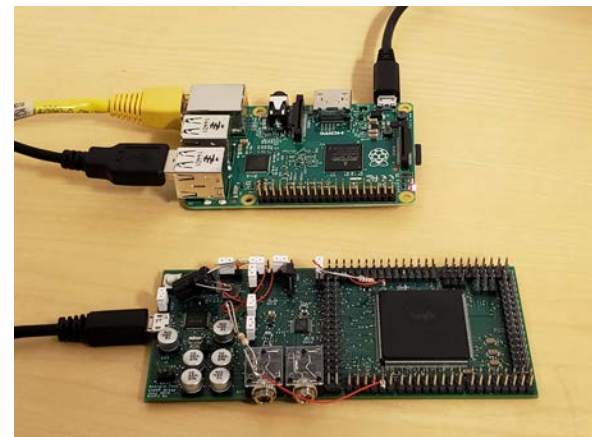
# FPAA Infrastructure Details



## Remote FPAA System



## Pi-hosted FPAA





# FPAA for Neuromorphic Computing

Near-term work is looking at interfacing with other neuromorphic APIs like EONS (Evolutionary Optimized Neuromorphic Systems)

We are also looking to host a prototype of UTK's DANNA architecture, which allows for testing neuromorphic architectures.



# FPGAs as Rogue Enablers

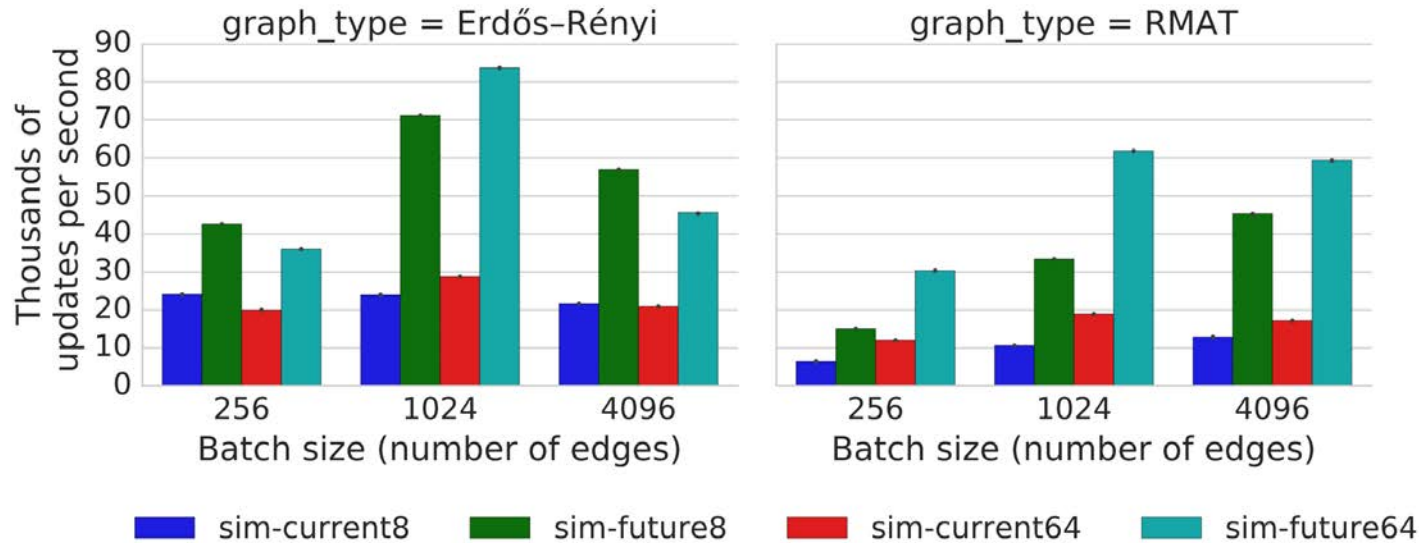


Boards with HBM enable high bandwidth for mapping algorithms, machine learning, and neuromorphic computing (Intel Loihi, DANNNA)





# RG Benchmarking - Emu Microbench (STINGER)



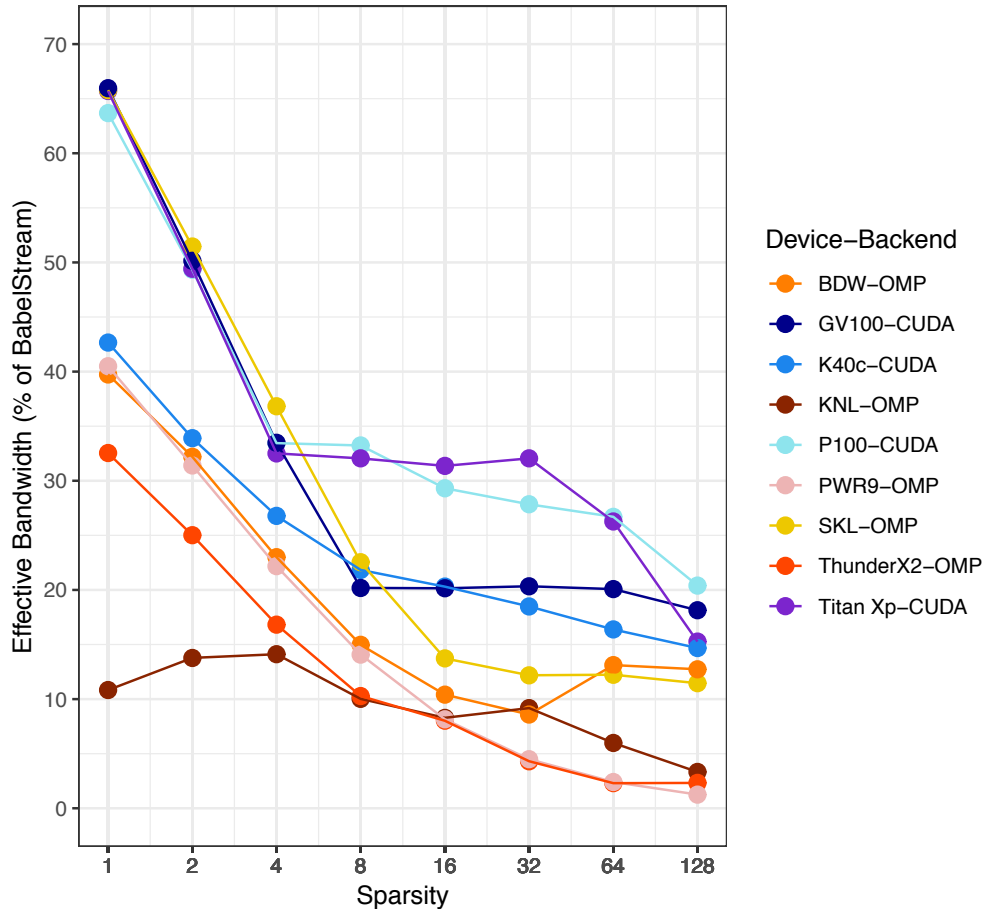
Focused on Streaming BFS

Code available at <https://github.com/ehein6/emu-microbench> and increasingly as part of Emu toolchain releases



# RG Benchmarking - Spatter

Impact of Access Sparsity  
GATHER kernel, Uniform Stride



**Code available at Spatter.io**

Spatter provides a new benchmark to evaluate gather/scatter accesses across different CPU and GPU platforms.

Emu and FPGA backends are in progress!

# RG Libraries – ParTI (Parallel Tensor Infrastructure)



Hardware

Multicore  
CPUs

GPUs

Distributed  
Memory

Emu

**ParTI! Library**

<https://github.com/hpcgarage/ParTI>

Sparse

Baseline Tensor  
Routines

Tensor  
Decomposition

Algorithms



# RG Libraries - ParTI

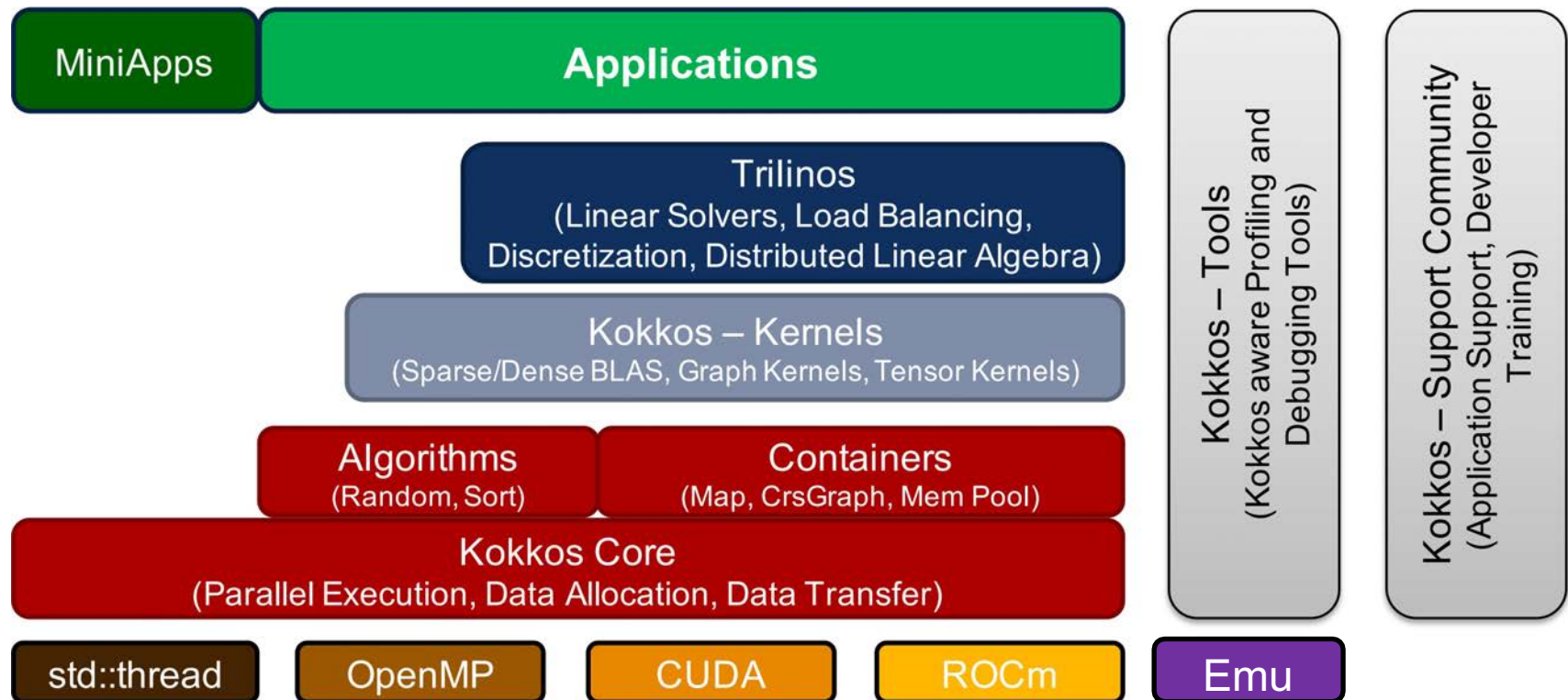
Data Structures/ Platforms	Algorithms	Multicore CPUs	GPUs	Distributed Systems	Emu
COO	CP	✓	✓	✓	✓
	Tucker	✓	✓		
HiCOO	CP	✓	✓	✓	
	Tucker				



# RG Tools - Kokkos



- Kokkos is a C++ library that is focused on ***performance portability*** through the mapping of parallel patterns

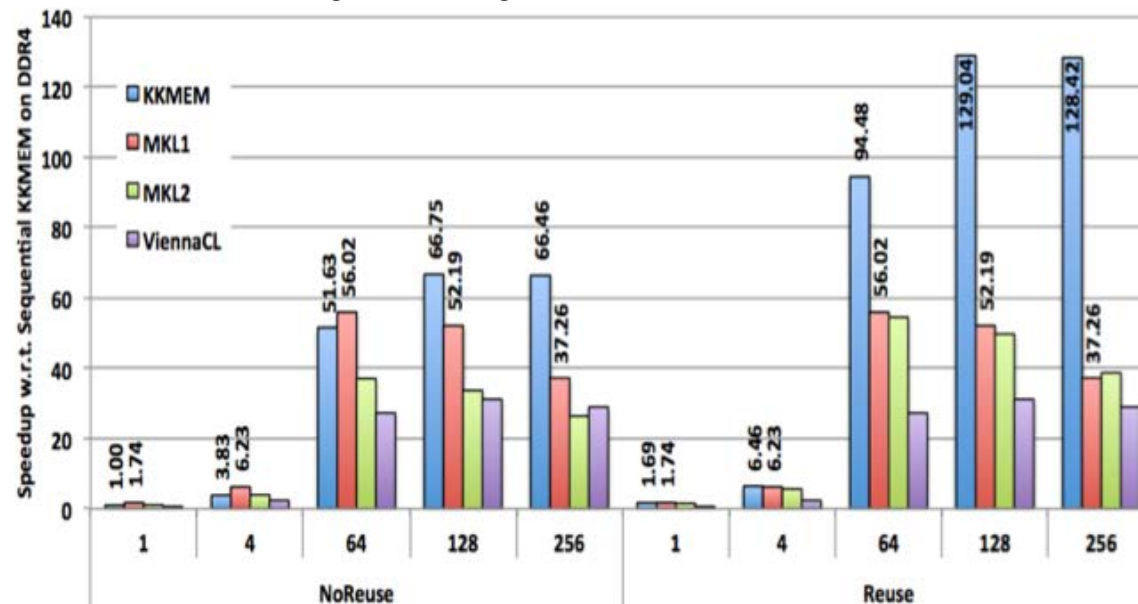


# RG Tools - Kokkos



- Kokkos Kernels provide implementations of common sparse, dense linear algebra, and graph-related operations
  - BLAS, SPARSE, AXPY, graph coloring, tensor contraction
  - Common front-end implementations benefit from mapping to Kokkos

SPGEMM Results Comparing Kokkos Kernel implementation with MKL and ViennaCL on Knight's Landing



From Rajamanickam, et al. KokkosKernels: Compact Layouts for Batched Blas and Sparse Matrix-Matrix multiply, Batched BLAS Workshop, 2017



# How to Work with the Rogues Gallery



GT Home > Home > Rogues Gallery

## Request Rogues Access

Our intention is that the Rogues Gallery will provide an open environment for experimentation and collaboration on unique, new request access to the Rogues Gallery please fill out the following form.

Start

Name \*

Email \*

Sign up for an account at [crnch.gatech.edu](http://crnch.gatech.edu) – it's free!

Join our Slack or email lists and look out for CRNCH-sponsored events like tutorials and yearly summits in Atlanta, GA

# Acknowledgments



- Emu – Janice McMahon and many others!
- Dr. Jennifer Hasler
- Srinivas Eswar (GT CSE)
- Dr. Eric Hein (GT ECE  $\Rightarrow$  Emu)
- Patrick Lavin (GT CSE)
- Jiajia Li (GT CSE  $\Rightarrow$  PNNL)
- Abdurrahman Yaşar (GT CSE)
- Dr. Ümit Çatalürek (GT CSE)
- Dr. Tom Conte (GT CS/ECE)
- Dr. Vivek Sarkar (GT CS/ECE)
- Dr. Bora Uçar (ENS Lyon CNRS)
- Dr. Rich Vuduc (GT CSE)

## Code:

- `crnch-rg.gitlab.io` – front-end for public resources
- <https://github.com/ehein6/emu-microbench>
- Spatter.io – Spatter benchmark
- ParTI - <https://github.com/hpcgarage/ParTI>





# Tutorial Feedback

- Feel free to email us:
  - Jason - [jason.riedy@cc.gatech.edu](mailto:jason.riedy@cc.gatech.edu)
  - Jeff - [jyoung9@gatech.edu](mailto:jyoung9@gatech.edu)
- We also have a survey for you to fill out on the event webpage at  
***[crnch-rg.gitlab.io/asplos-2019](http://crnch-rg.gitlab.io/asplos-2019)***

