Wrangling Rogues: A Case Study on Managing Experimental Post-Moore Architectures

Will Powell, Jason Riedy, Jeffrey Young, Tom Conte

1 August 2019

Center for Research into Novel Computing Hierarchies at Georgia Tech

Outline

Cyber-security	Identify anomalies, malicious actors
Health care	Find outbreaks, population epidemiology, similar patient association
Social networks	Advertising, searching, grouping
Intelligence	Decisions at scale, regulating markets, smart & sustainable cities
Systems biology	Understanding interactions, drug design
Power grid / Smart cities	Disruptions, conservation, prediction

Irregular data access. Changing data.

High-Performance Data Analysis (HPDA)

Novel applications:

- Data at scale and speed needs new ideas for computing analysis.
- "Big data" platforms fare poorly v. a single thread plus large SSD even for static data sets. (McSherry, Isard, Murray. "Scalability! But at what COST?" HotOS XV, 2015.)
- Many high-level codes are written and re-written to answer one question: need flexibility.
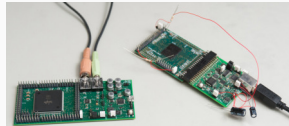- *Some* primitives may be tuned and re-used.

Rogue: Someone who goes their own way, who breaks away from the crowd.

- Current architectures are hitting limits on manufacturing, heat dissipation, memory latency...
- What happens when novel prototypes hit reality?
- Designers need feedback, a software ecosystem, and **trained students**.

Hardware! "I'll tell you later."

Introducing the CRNCH Rogues Gallery

CRNCH Rogues Gallery

A physical & virtual space for hosting novel computing architectures, systems, and accelerators since fall 2017.

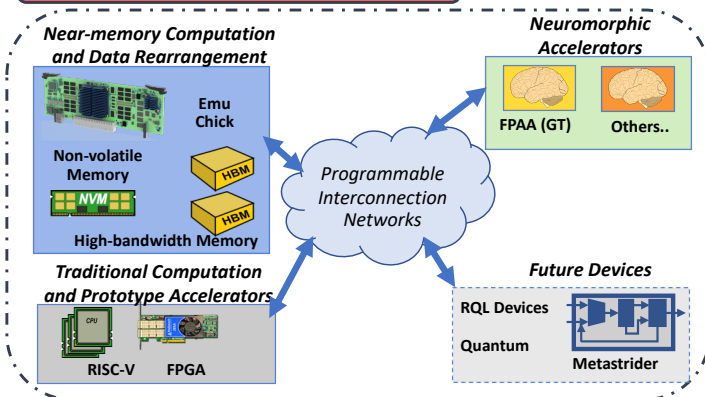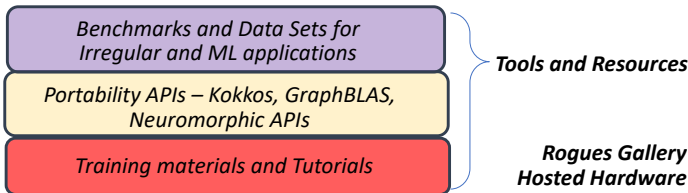Host / manage remote access for novel architectures to

- kick-start software ecosystems (*e.g.* Kokkos),
- leverage real applications to train students, and
- provide rapid feedback to architects.

Amortize effort and cost of trying novel architectures.
Break the "but it's too much work" barrier.
http://crnch.gatech.edu/rogues-gallery

Rogues Gallery summary

Current Rogues

Current Rogues

Emu Chick

3D Stacked Memories and FPGAs

Neuromorphic / Analog Hardware (FPAA)

Emu Technology's Chick

- "Migratory Memory Side Processing" to exploit *weak locality*.

- Data for graph edge attributes, documents / medical records, *etc.* reside nearby even if accessed irregularly.

- Moving threads to data on reads: all reads are local, so lower latency.
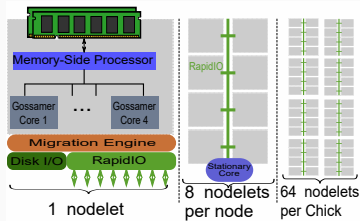
Emu Chick: Ongoing Results

Pointer chasing
Sandy Bridge Xeon vs Emu Chick (8 nodelets)
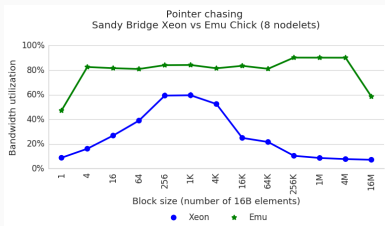
Platform:

- Application direction.
- Demo-able code.
- Debugging.
- Connections.
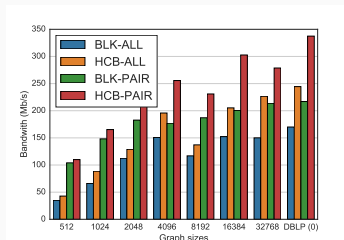
People:

- GT → Emu: One employee, one intern.
- Emu → GT: One graduate student.

3D Stacked Memory and FPGAs

- FPGA + HMC / DRAM: Enable experiments with "near-memory" and memory-centric processing.
- FPGA platforms prototype non-traditional accelerators like Automata, sparse data engines, *etc.*
- Current work is supported in part by Micron hardware donation.

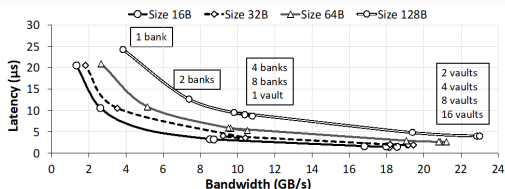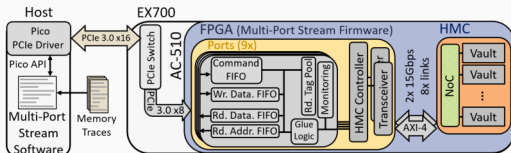Hadidi, Asgari, Young, Mudassar, Garg, Krishna, Kim. "Performance Implications of NoCs on 3D-Stacked Memories: Insights from the Hybrid Memory Cube (HMC)," ISPASS 2018

· Characterizations with FPGA and Hybrid Memory Cube show latency/bandwidth tradeoff.

· Other FPGA work is focused on compilers, HPC prototyping, and sparse algorithms for Intel and Xilinx FPGAS.

Neuromorphic systems

- Field-Programmable Analog Array (FPAA) System-On Chip, designed in the lab of Dr. Jennifer Hasler.
- Analog + digital to achieve unprecedented power and size reductions.
- Potential on-chip/package accelerator.
- Adding other neuromorphic systems

Ultra-low energy computation

< 23μW power
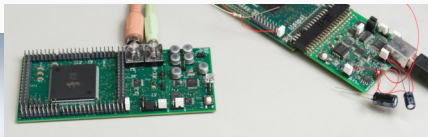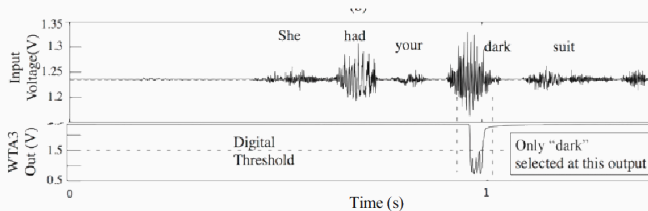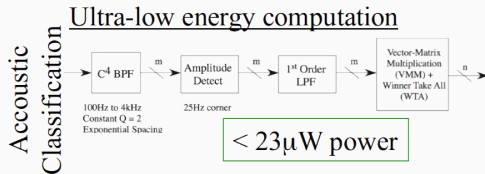
Embedded learning & classification: 20-30μW
on full, 1s Nzero database (GOMAC 2016)

Management lessons learned

Management lessons learned

> Helpful points
>
> Painful points

Rogues Gallery structure

User Resources

- emu-dev → emu-chick
- login / notebook → emu-dev
- login / notebook → fpga-dev-1 ..N
- login / notebook → fpga-intel
- login / notebook → fpga-hmc
- fpga-dev-1 → fpga-intel
- fpga-dev-1 → fpga-hmc
- login / notebook → fpaa-host
- fpaa-host --- fpaa-dev
- login / notebook → power-host
- power-host → nvidia-tegra-1 / nvidia-tegra-N

Scheduling, Tools, and Admin

- rg-adm **Slurm Ctl**
- toolbox (NFS)
- rg-db **Slurm DBD**

Key:
- Schedulable Resource
- Physical Resource
- VM
- USB device

Management lessons learned

- Invest in rogues, but realize some technology may be short-lived.
    - Minimize custom management effort.
- Physical hardware resources not dedicated to rogues should be kept to a minimum.
    - Don't spend $ on non-rogues.
- Collaboration and commiseration is key.
    - Rogues need a community to succeed.
- Licensing and appropriate identity management are tough but necessary challenges.
    - Use network isolation when needed.

Helpful points

- Network isolation provides security.
    - Well, enough given limited usefulness.
- Singularity is great for build environments.
    - HW start-ups cannot afford supporting every OS/arch.
    - IT cannot afford supporting every OS/arch.
    - Companies must be friendly...
- Inspired undergrads are wonderful!
    - Modernizing tools (FPAA)
    - Building out demonstrations
    - `http://www.vip.gatech.edu/teams/rogues-gallery`

Painful points

- SLURM aspects:
    - Managing slurmd.conf.
    - Building on all the OS/arch combos.
- Few light-weight management options.
    - salt-ssh, ansible on *some*
- Hardware access for rebooting, reseating.
- Many programming interfaces, few people
    - Kokkos, TENNLab, more...
- Still need to tackle "sensitive" data, including some FPGA IP
- Reproducible / replicable / audit-able results

- Integrating FPAAs and toolchain
- Tight development loop with Emu
- Active research projects and publications
- Community building via tutorials & talks
- New approaches to benchmarking, quantum software stacks, neuromorphic toolchains, …

CRNCH Rogues Gallery connects researchers *and students* with novel architectures and architects with upcoming applications.

Let us host / manage your neat stuff!
`http://crnch.gatech.edu/rogues-gallery`

Acknowledgments

Fantastic students and colleagues:

- Srinivas Eswar (GT CSE)
- Dr. Eric Hein (GT ECE ⇒ Emu)
- Patrick Lavin (GT CSE)
- Dr. Jiajia Li (GT CSE ⇒ PNNL)
- Abdurrahman Yaşar (GT CSE)
- Chunxing Yin (GT CSE)

- Dr. Jeffrey S. Young (GT CS)
- Dr. Tom Conte (GT CS/ECE)
- Dr. Vivek Sarkar (GT CS)
- Dr. Ümit Çatalürek (GT CSE)
- Dr. Bora Uçar (ENS Lyon CNRS)
- Dr. Rich Vuduc (GT CSE)

Code (ideally will have links from `crnch.gatech.edu`):

- `https://gitlab.com/crnch-rg`
- `https://github.com/ehein6/emu-microbench`

Other testbeds:

- ORNL: ExCL
- PNNL: CENATE

- Argonne
- Sandia HAAPS

- Berkeley: AQCT
- (others?)

External Image Credits

- "What's that watermelon doing there?": copyright MGM, used for identification
- Oscar Wilde: public domain, obtained from Wikipedia
- Edna St. Vincent Millay: public domain, obtained from Wikipedia
- Dread Pirate Roberts: copyright 20th Century Fox, used for identification
- Mary Jackson, Katherine Goble Johnson, Dorothy Vaughan (Hidden Figures): copyright 20th Century Fox, used for identification
- Malcolm Reynolds: copyright Universal Pictures, used for identification
- Rogue One: copyright Walt Disney Studios Motion Pictures, used for identification
- The Story of Karrawingi, the Emu (cover): copyright estate of Leslie Rees, used for identification
- Big Hero 6: copyright Walt Disney Studios Motion Pictures, used for identification