UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

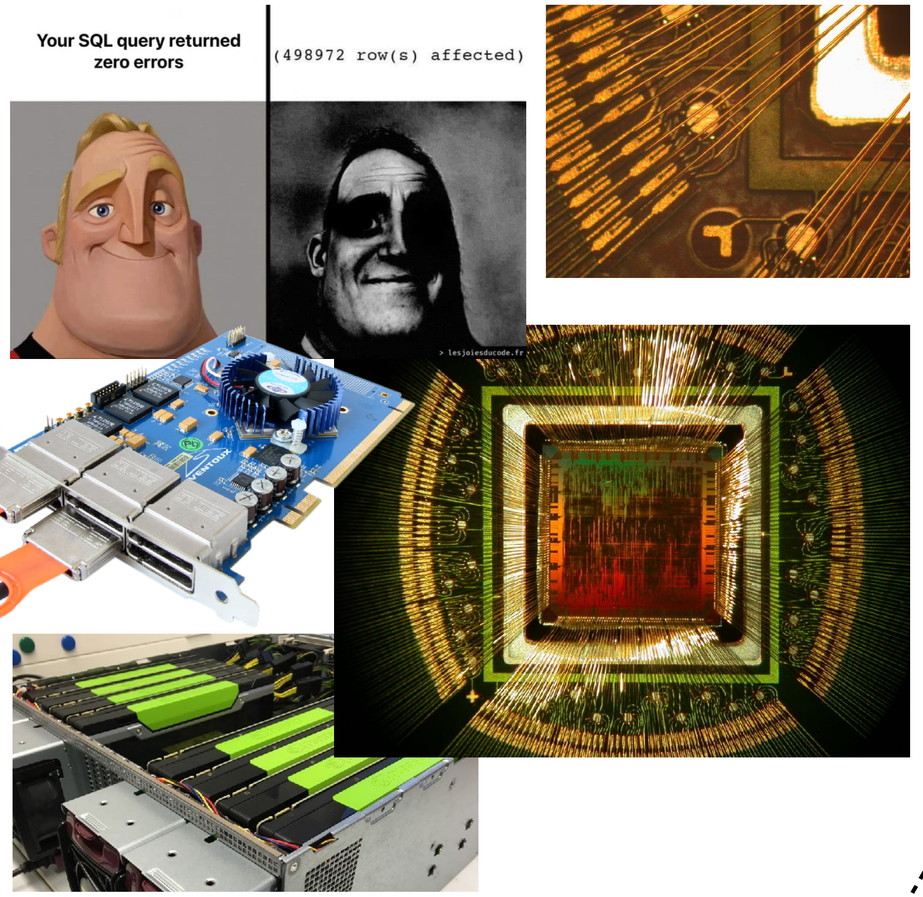
FPGAS AND PICOJOULES: A TALE OF LOVE AND HATE

Hendrik Borras, Daniel Barley, Paul Kupper, Holger Fröning - hendrik.borras@ziti.uni-heidelberg.de
Hardware and Artificial Intelligence (HAWAII) Lab - hawaii.ziti.uni-heidelberg.de
Heidelberg University

Workshop on "FPGA in (High Performance) Computing. Quo vadis", Mar 13, 2026, ZIB, Berlin

GROUP BACKGROUND: FROM HW TO ACCELERATORS TO ML

From: database engineer, HW designer (ASICS, FPGA), HPC



$$x^l = \Phi(\mathbf{W} \oplus \mathbf{x}^{l-1} + b^l)$$

Neural Architectures

tvm

Compiler

Plethora of HW

$$perf[\frac{\text{ops}}{\text{s}}] = p[Watt] \cdot e[\frac{\text{ops}}{\text{J}}]$$

$$P = afCV^2 + VI_{leakage}$$

To: vertically integrated approach to efficient ML => HW systems for AI

HARDWARE & ARTIFICIAL INTELLIGENCE

HAWAII [hə'waːiː] LAB

Holger Fröning
Prof. Dr.

Andrea Seeger
Office Assistance

Emma Seeger
Assistance's Assistance

Shigehiko Schamoni
M.A.

Kazem Shekofteh
Postdoc

Bernhard Klein
PhD Student

Alexandra Stehle
PhD Student

Hendrik Borras
PhD Student

Daniel Barley
PhD Student

Aleksandra Poreba
PhD Student

Xiao Wang
Researcher

Robin Janssen
PhD Student

Andrej Meininger
Master Research Assistant

Anusha Chattopadhyay
Graduate Research Assistant

Arjan Siddhpura
BScI Thesis

Christian Heusel
BScI Thesis

Dennis Jakob
Bachelor Research Assistant

Dominik Gausepohl
MScDACS Thesis

Elizaveta Mironovich
TA

Fangling Du
MScSC Thesis

Hao Zhang
MScSC Thesis

Ilya Belkin
MScDACS Thesis

Jonathan Bernhard
MScDACS Thesis

Jonathan Leis
MScDACS Thesis

Leandro Borzyk
MScTI Thesis

Lukas Rapp
MScDACS Thesis

Max Mielke
Master Research Assistant, MScTI Thesis

Maximilian Burr
Master Research Assistant

Niklas Summ
Master Research Assistant, TA

Nils Kochendörfer
MScTI Thesis

Paul Kupper
MScDACS Thesis

Philia Jankov
Master Research Assistant

Philipp Hematty
MScTI Thesis

Runan Duan
TA

Shijie Sun
MScSC Thesis

Theo Stempel-Hauburger
MScDACS Thesis

Vimala Bauer
TA

CONTINUED DEMAND FOR MORE HARDWARE PERFORMANCE

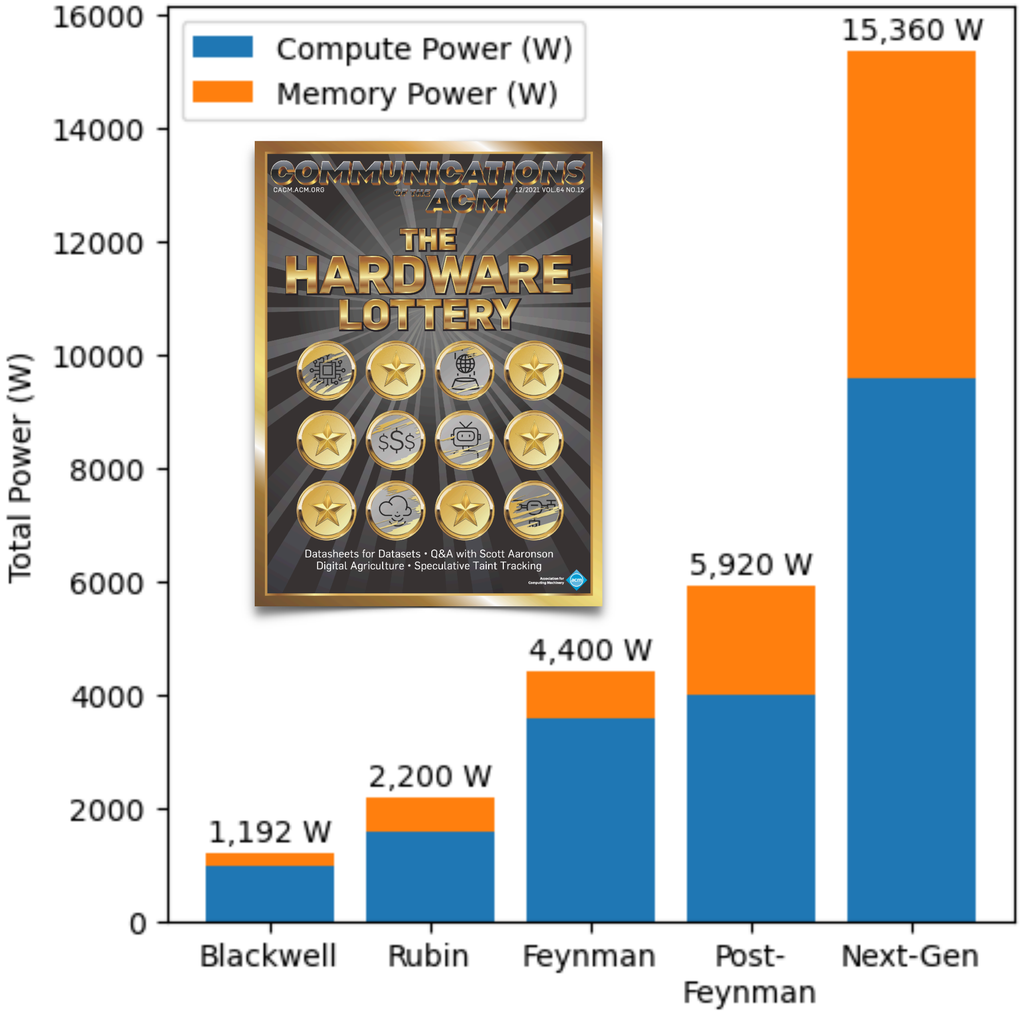
GPUs are close to perfect for standard DNNs and LLMs, but can we do better?

- Future scaling of GPUs comes at tremendous costs in terms of power consumption

Emerging HW often comes with imperfections

- Noise, non-linearities, saturation effects, etc.
- Analog computations (electrical, photonic), resistive memory

Power scales GPU performance



KAIST Teralab, HBM Roadmap Ver 1.7 Workshop, https://drive.google.com/file/d/1wdGvyAYM0SOjlweJcgPDKwXlk6czbuZo/view

POST-DENNARD CMOS PERFORMANCE SCALING

Peak-compute throughput

$$\text{perf}\left[\frac{\text{ops}}{\text{s}}\right] = \text{IPC} \cdot f$$

Power-limited throughput

$$\text{perf}\left[\frac{\text{ops}}{\text{s}}\right] = P[\text{W}] \cdot \eta\left[\frac{\text{ops}}{\text{J}}\right]$$

Energy is additive

$$= P \cdot \left(\frac{\text{ops}}{\epsilon_{\text{op}} + \epsilon_{\text{mem}}}\right)$$

Operation ← → Data movement

$$\epsilon_{\text{op}} = f_{\text{comp}}(t, b)\ [pJ]$$

GPU FPGA CPU

CMOS-BASED. ARCH DEP.

$$\epsilon_{\text{mem}} = f_{\text{mem}}(l, d, b)\ [pJ]$$

SRAM HBM DRAM WIRES

TECHNOLOGY DEP.

Frequency f, efficiency ϵ, data type $t[\{\texttt{float}, \texttt{int}\}]$, bit width b, distance $d[\text{mm}]$, memory level l

CMOS TECHNOLOGY TRENDS 45NM (2014) VS 7NM (2021)

	picoJoules	
Integer	45nm	7nm
Add		
8 bit	0,03	0,007
32 bit	0,1	0,03
Mult		
8 bit	0,2	0,07
32 bit	3,1	1,48

	picoJoules	
Float	45nm	7nm
FAdd		
16 bit	0,4	0,16
32 bit	0,9	0,38
FMult		
16 bit	1,1	0,34
32 bit	3,7	1,31

64-bit	picoJoules	
Mem	45nm	7nm
SRAM		
8kB	10	7,5
32kB	20	8,5
1MB	100	14
DDR4	1300 - 2600	1300 -
HBM2		250 - 450

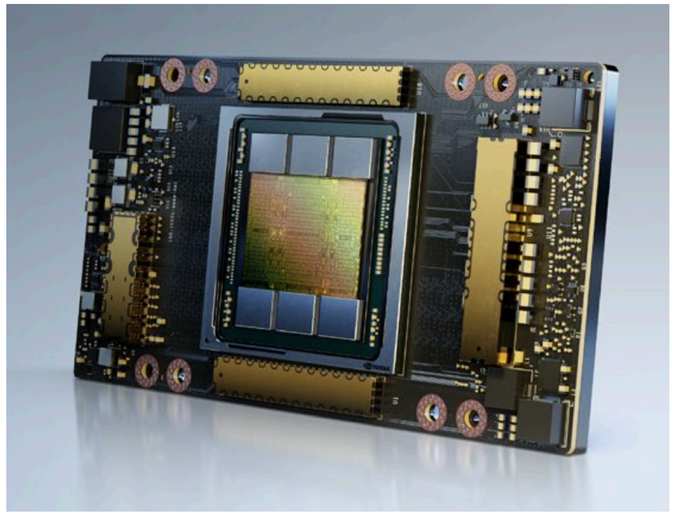
M. Horowitz, "1.1 Computing's energy problem (and what we can do about it)," 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC). doi: 10.1109/ISSCC.2014.675732

Norman P. Jouppi, et al. 2021. Ten lessons from three generations shaped Google's TPUv4i. ISCA. https://doi.org/10.1109/ISCA52012.2021.00010

A100 NUMBER CHECK

NVIDIA A100 SXM, 7nm, 400W

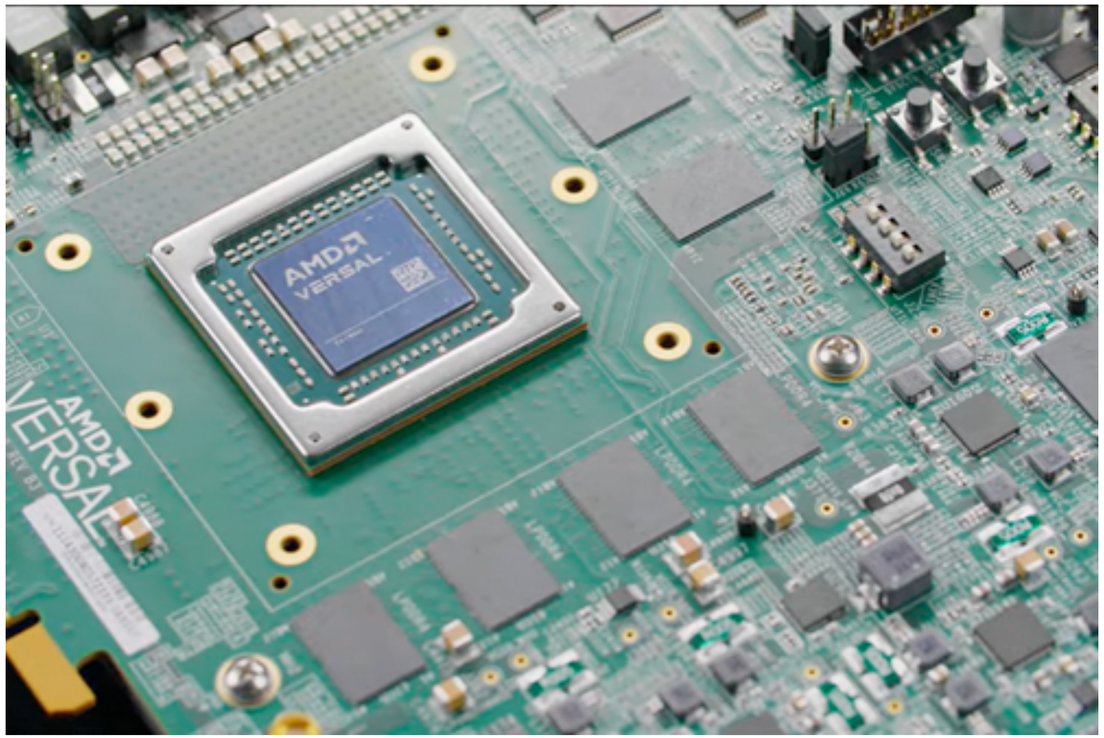
	A100 TOPS/s	A100 pJ/OP	Theory pJ/OP	Residual = memory pJ/bit
Vector FP32 [TF/s]	19,5	20,51	1,31	0,20
Matrix FP32 [TF/s]	156	2,56	1,31	0,01
Matrix FP16 [TF/s]	312	1,28	0,34	0,02
Matrix INT8 [TOP/s]	624	0,64	0,07	0,02



7nm Memory Energy

	64bit	1bit
SRAM 8kB	7,5	0,12
SRAM 32kB	8,5	0,13
SRAM 1MB	14	0,22
DDR4	1300	20,31
HBM2	450	7,03

WHERE DO WE PLACE FPGAS IN THIS?



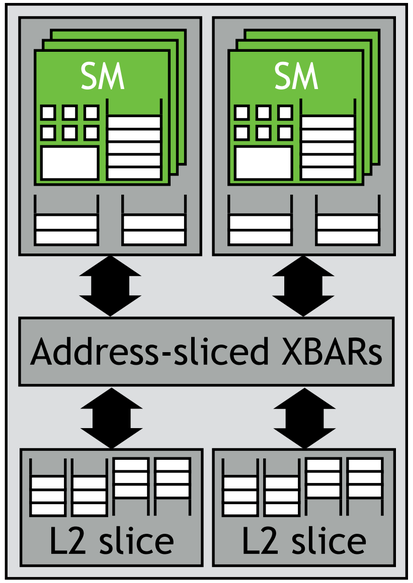
AMD Versal VEK280

	TOPS/s	pJ/OP	pJ/OP	Residual = memory pJ/bit
Vector FP32 [TF/s]	?	?	?	?
Matrix FP32 [TF/s]	?	?	?	?
Matrix FP16 [TF/s]	?	?	?	?
Matrix INT8 [TOP/s]	?	?	?	?

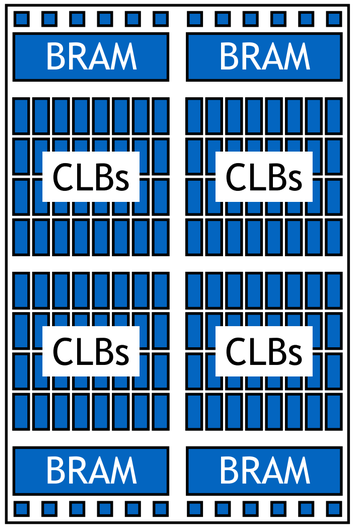
LET’S START MORE GENERAL

REVIEW OF MANY-CORE PROCESSORS

	GPU	FPGA
High concurrency at reduced frequency	y	y
Flat memory hierarchy	y	y
Scratch-pad memory	y	y
Programming using data-parallel kernels	y	partly
Latency tolerance	BSP-like	block data transfer
Main requirement	structured parallelism	predictability
Energy efficiency driver	extreme specialization	flexible specialization
NOC	non-blocking[1]	blocking[1]
3D die stacking	difficult	possible
Main benefit	**performance**	**wattage**
Main drawback	**amount of data movements**	**memory performance**



GPU



FPGA

[1] *based on best research effort (possibly outdated)*

SO WHAT CAN WE DO WITH THIS?

“Love” side

FPGAS AND ML WORKLOADS

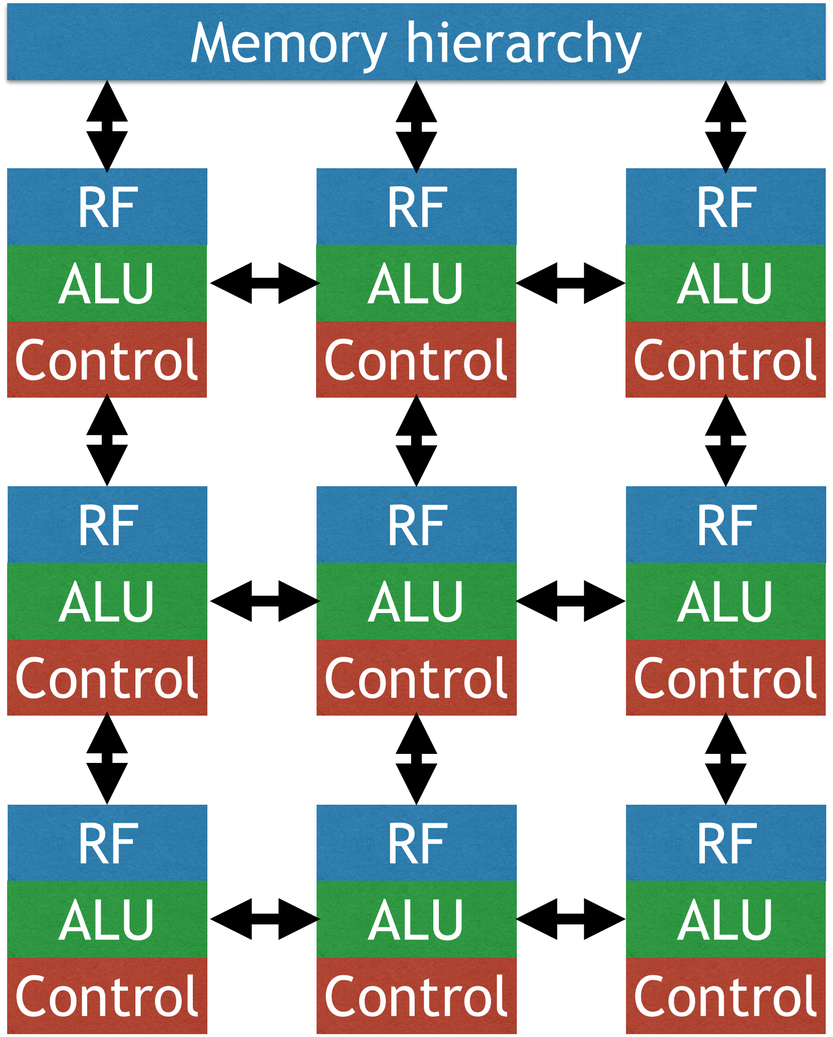
Low energy though low frequency

High degree of parallelization

Predictability is at the core of DNNs

No unnecessary overhead of predictive HW

- No speculative execution
- No register state caching



Array-based processor

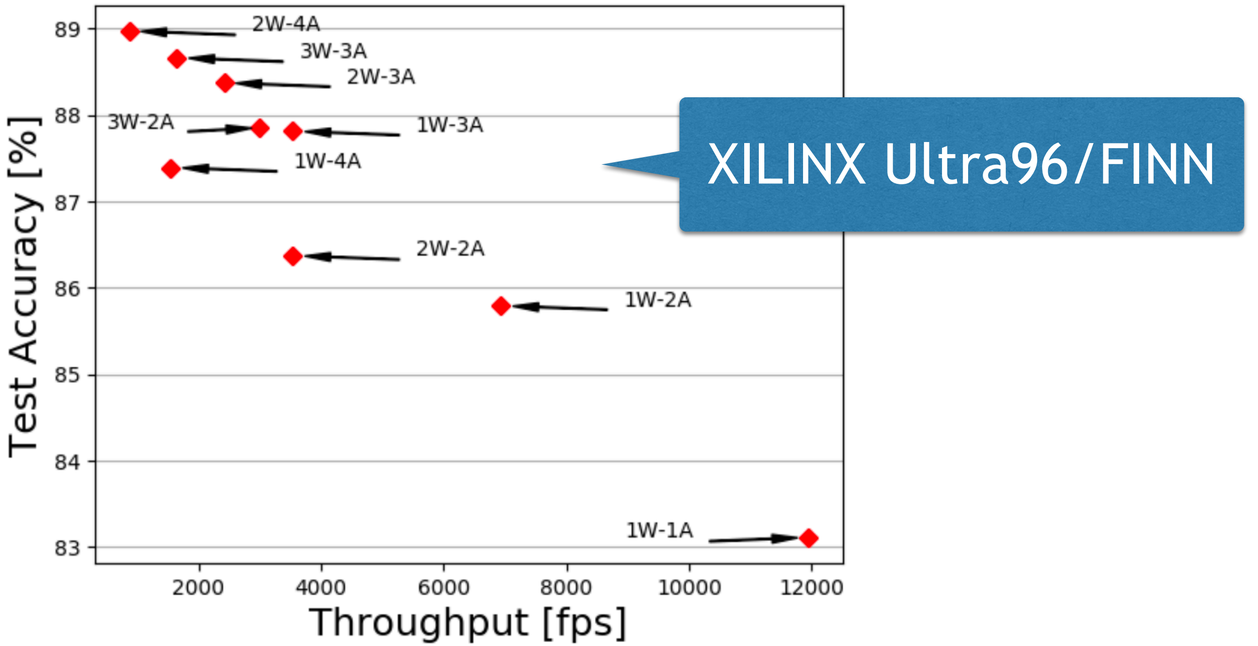
SO, WHAT'S OUT THERE?

High-level, HLS based frameworks:

hls4ml, FINN, hPipe

Low-level: LUT-based

LogicNets, LUTNet, NullaNet, PolyLUT



Throughput-accuracy trade-off of different quantized VGG models on the CIFAR-10 task for an FPGA data-flow architecture

Wolfgang Roth, Günther Schindler, Bernhard Klein, Robert Peharz, Sebastian Tschiatschek, Holger Fröning, Franz Pernkopf, Zoubin Ghahramani, Resource-Efficient Neural Networks for Embedded Systems, JMLR 25(50):1–51, 2024. [pdf]

WHAT DOSEN’T WORK?

“Hate” side

HOW DO WE ARGUE ABOUT ENERGY USAGE?

picoJoule information is fundamentally missing

Good reasons for this

Makes comparisons to other architectures very difficult

AMD Versal VEK280

	TOPS/s	pJ/OP	pJ/OP	Residual = memory pJ/bit
Vector FP32 [TF/s]	?	?	?	?
Matrix FP32 [TF/s]	?	?	?	?
Matrix FP16 [TF/s]	?	?	?	?
Matrix INT8 [TOP/s]	?	?	?	?

PROBLEMS WITH HIGH-THROUGHPUT MEMORY

HBM boards exist, but:

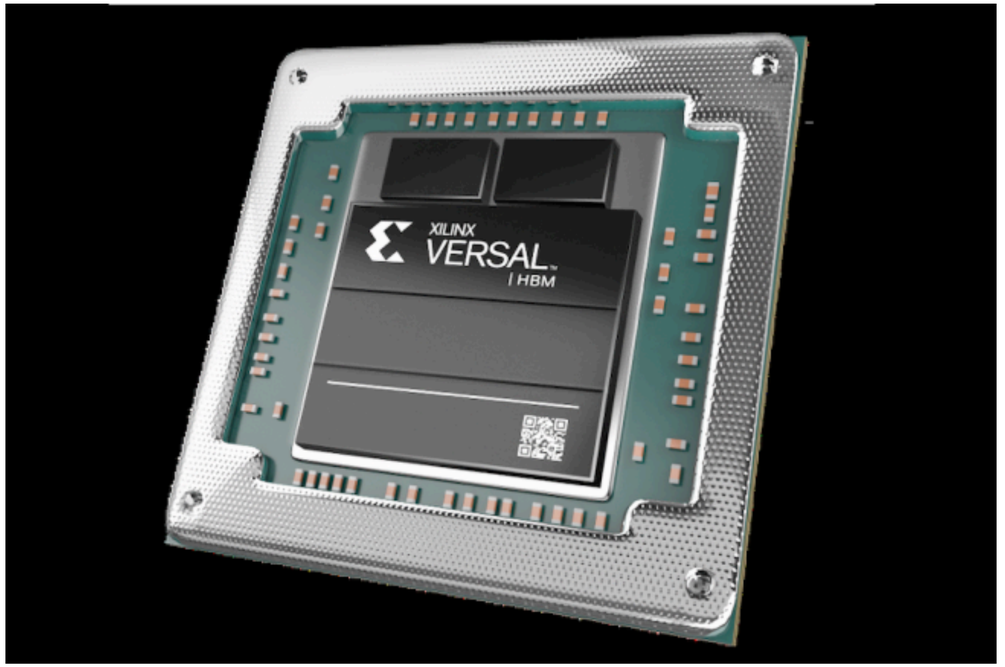
Getting them to run is highly non-trivial

HBM is ill-supported over the lifetime length of FPGAs

- HBM: 2~3 years
- FPGAs: 5~6 years

Makes high throughput accelerators difficult

Makes model compression non-optional



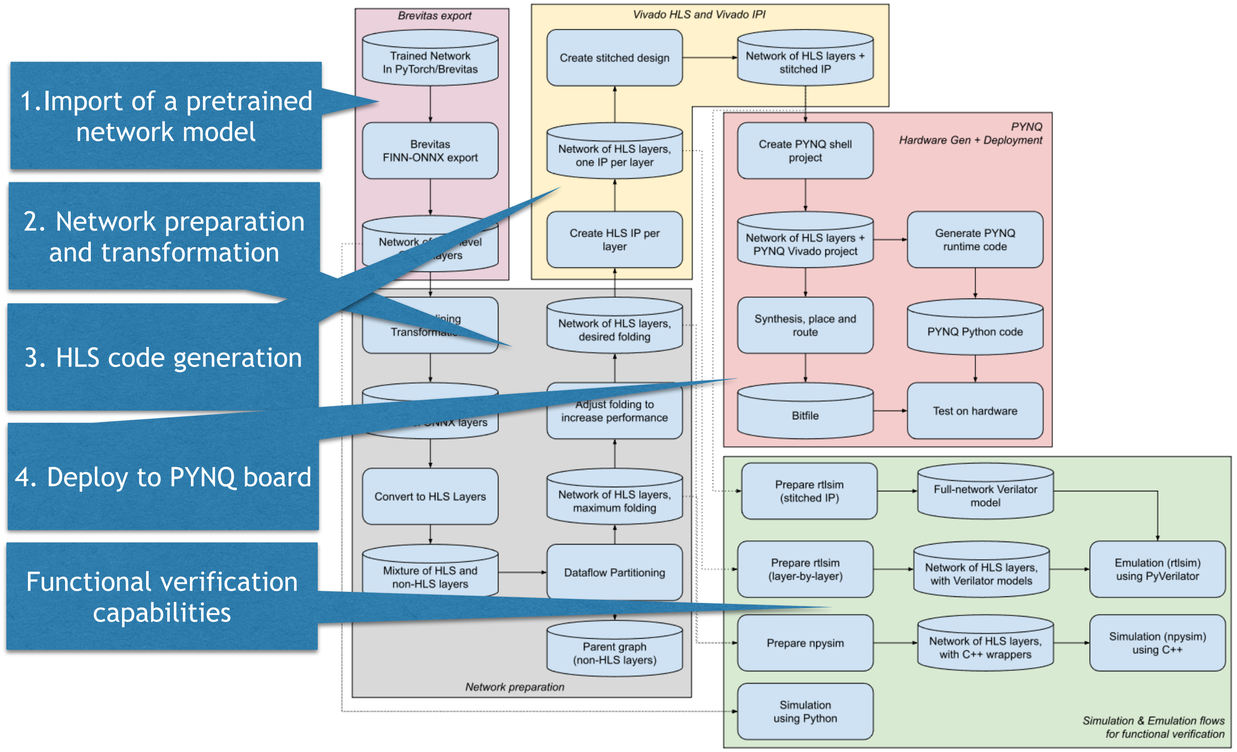
AMD Versal HBM announcement

TOOLING IS HIGHLY NON-TRIVIAL

Tooling itself is complicated

- Involved setup
- Requires deep HW understanding
- Often finicky

Even "simple" HLS tool-flows are comparatively complicated



HLS tool-flow of FINN

WHERE DOES THIS LEAVE US?

CAN ENERGY-TO-COMPLETION BE A COMPLETELY NEW ROLE FOR FPGAS?

"Time-to-Solution" impractical

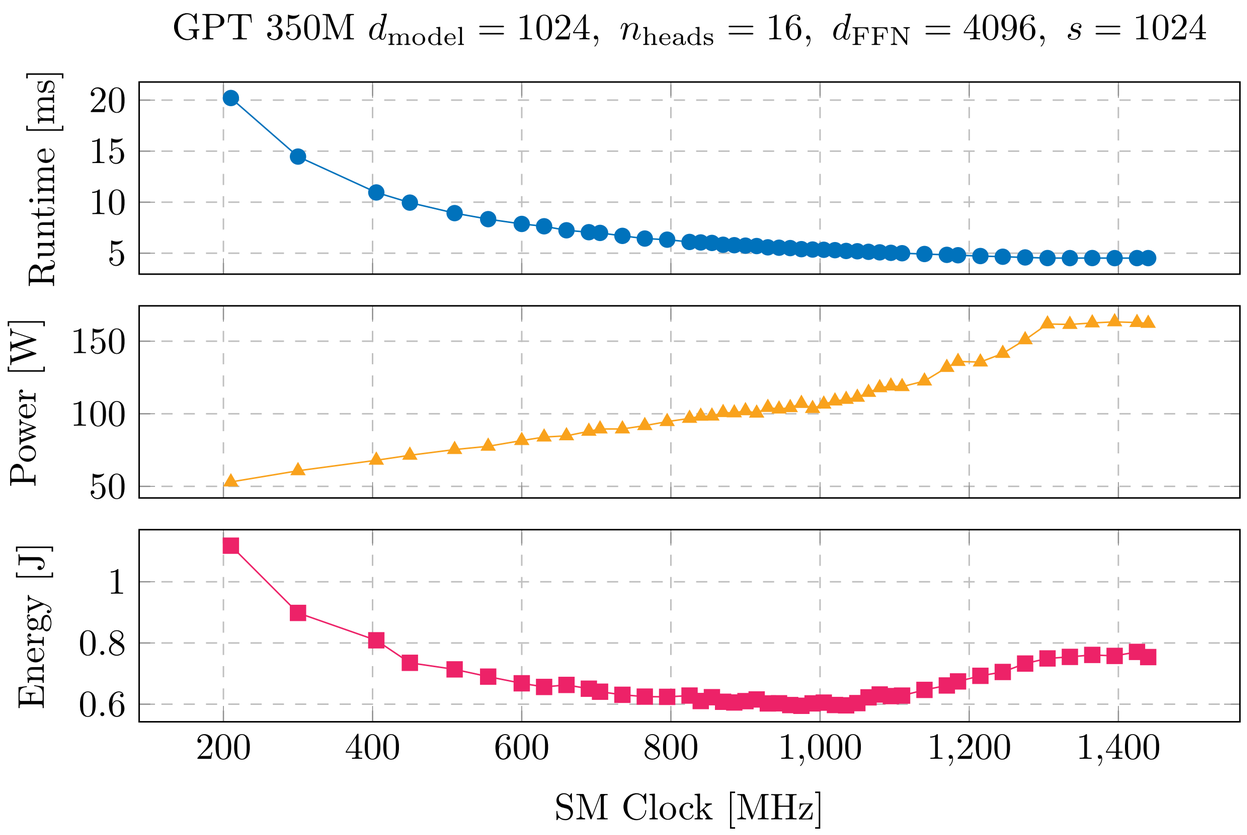
But: Promising energy efficiency

- Already works well on GPUs

However: FPGAs need to evolve

- LUT flexibility is often too expensive
- CGRAs offer a compelling compromise
- Memory bandwidth remains an issue

What do we do in the mean-time?



LLM energy scaling with frequency

Device: Nvidia A30

MAYBE MORE HETEROGENEOUS COMPUTE?

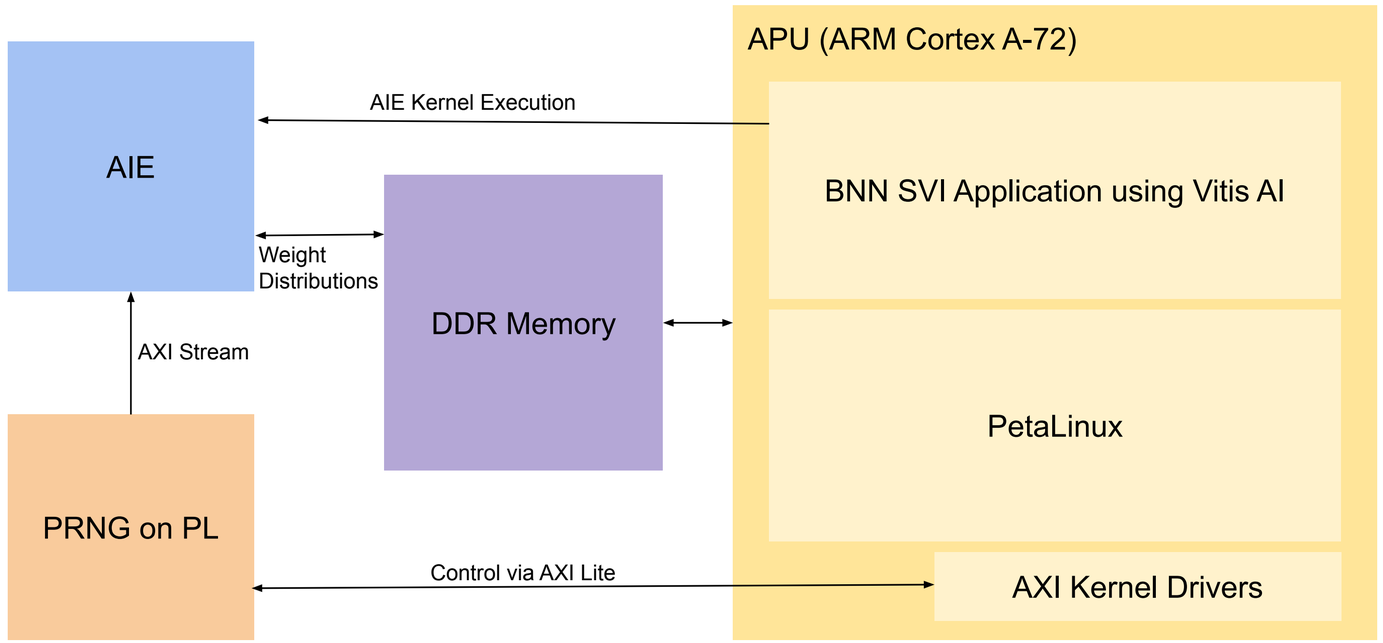
TARGETED HETEROGENEOUS APPLICATIONS

Workload: Bayesian Neural Networks

FPGA: AMD VEK280

- Offload-stochasticity to fabric
- Run DNN functions on AIEs

Promising in first evaluations



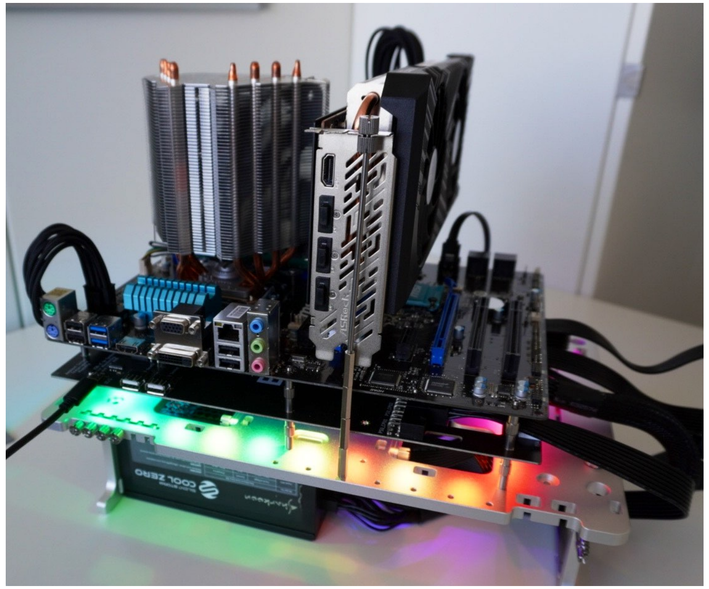
MAYBE WE SHOULD SIMPLY START MEASURING?

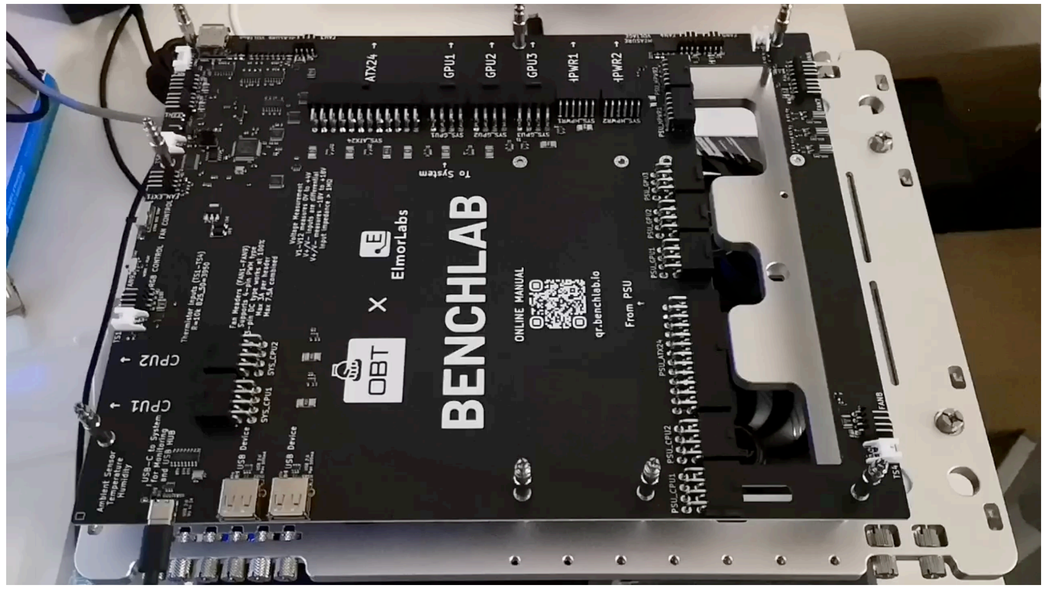
ENERGY-EFFICIENT SYSTEMS AND AI LAB (ESAIL)

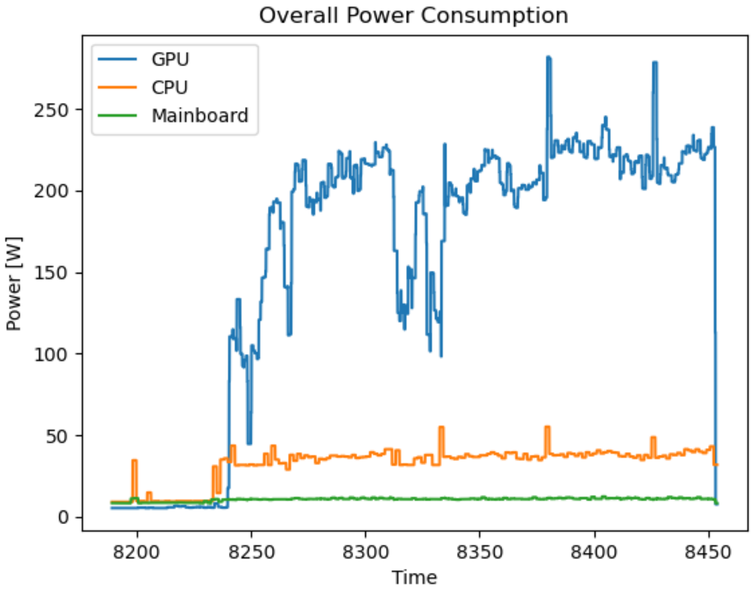
Explore optimality

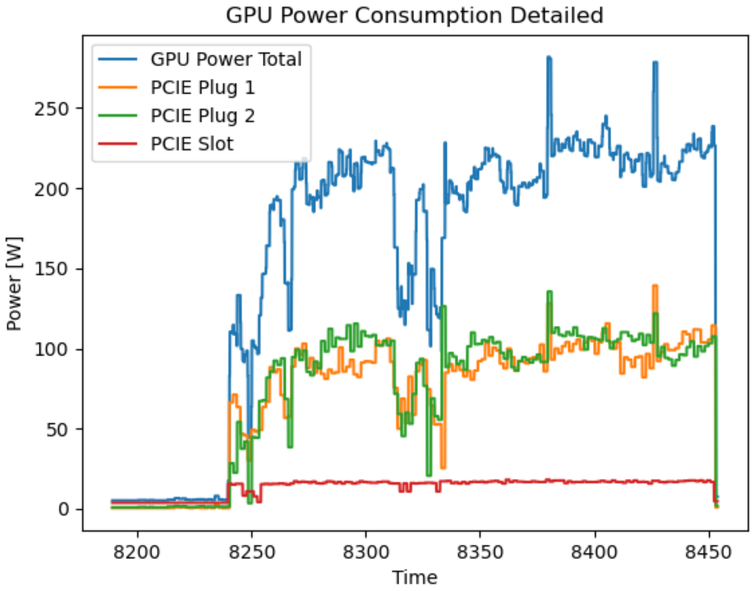
- GPU type (consumer/ server)
- DVFS
- Other accelerators
- System overhead

2x for comparisons









ENERGY-TO-COMPLETION FOR FPGAS?

Can we make energy efficiency the defining advantage of FPGAs?

What do we need to get there?

- Ways to reason about FPGA energy
  - Even if it's just case studies
- An evolving FPGA landscape
- Approachable tooling