Programmability, Scalability, and Security for Heterogenous Computing with FPGAs

Deming Chen

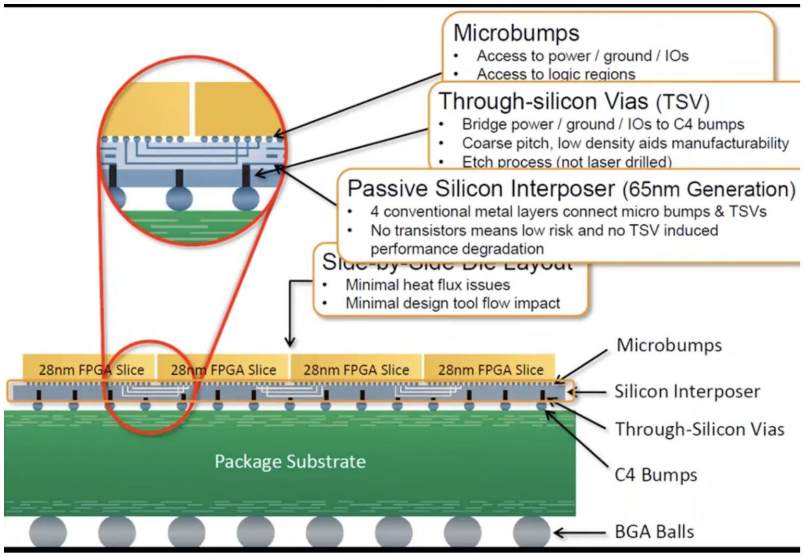
Abel Bliss Professor of Engineering

University of Illinois at Urbana-Champaign

DAC, 7/10/2023

Xilinx Led the 2.5D Device Development

- Xilinx started investigating 3D IC technology around early 2000
  - They built a true 3D, active on active, device in the labs
  - They also built a 2.5D implementation using a passive interposer
- "*Both devices worked, but we had to consider materials, connection types, architectures, designability, and manufacturability and cost…there were many considerations before we decided to go with SSI and agree on what we needed to bring the SSI technology to a fully commercially available product.*"
  - Ivo Bolsens, CTO, Xilinx, 2012.

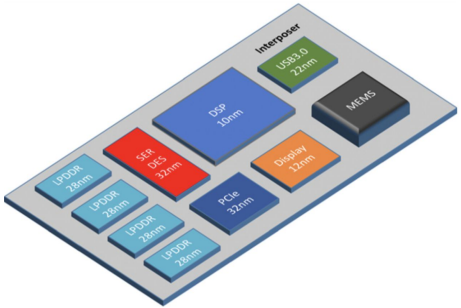


Xilinx Virtex-7 2000T FPGA, 2011

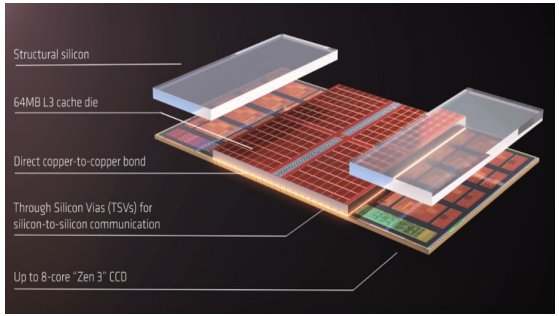
A common technology now: NVidia, Intel, AMD, …

https://www.eetimes.com/2-5d-ics-are-more-than-a-stepping-stone-to-3d-ics/

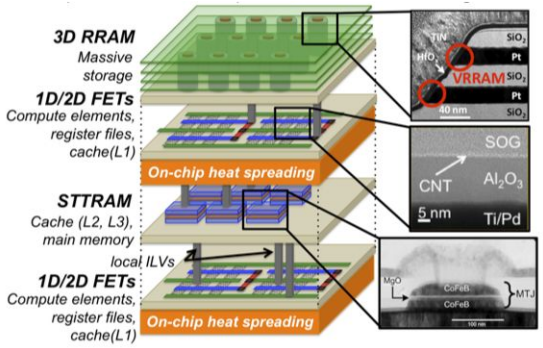
Examples of the Current State-of-the-Art



Chiplet-based system using an interposer.
Source: Cadence



AMD's 3D V-Cache technology stacks the cache on a processor.
Source: AMD

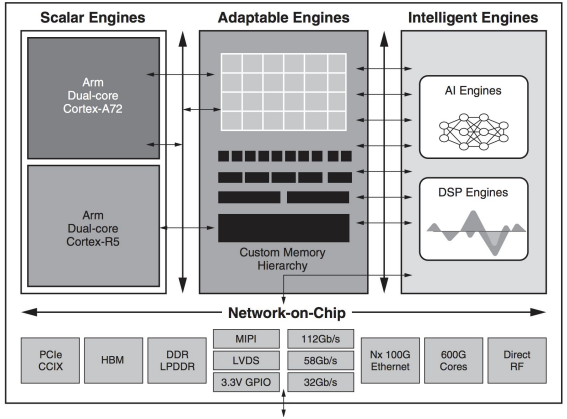


Monolithic 3D computing system.
Source: S. Mitra/Stanford

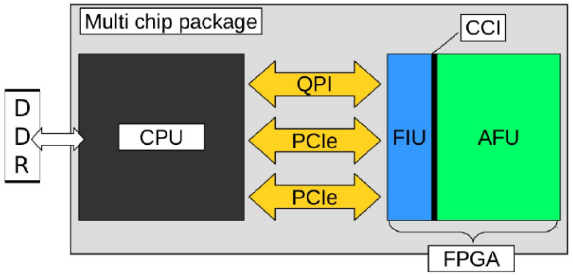
- Chiplet allows integration of a mix of different chips, each separately fabricated at suitable node
- Next-generation 3D-chip packaging and monolith 3D integration
- Challenges remain: thermal, yield, cost, reliability, testing, etc.

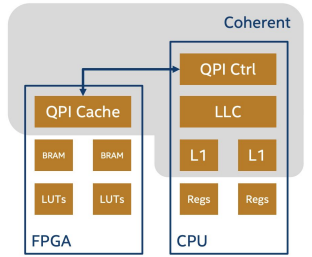
This Talk Focuses on FPGAs

- FPGAs try to blend the flexibility of software with the performance and efficiency of custom hardware
  - The principal difference with CPUs is the ability to make substantial changes to the datapath itself in addition to the control flow.
  - The main difference with custom hardware, i.e., ASICs, is the possibility to adapt the hardware during reconfiguration time or runtime by "loading" a new circuit on the reconfigurable fabric.



AMD-XILINX





INTEL

Why FPGAs?

- Low NRE cost
- Field programmable
- Fast time to market
- More flexible than ASICs
  - Less efficient than ASICs
- Shorter latency and lower power/energy compared to CPUs and GPUs
  - Depending on applications
  - Typically cannot compete with GPUs on compute throughput and memory bandwidth
- Efficient for intensive and parallelizable computations
  - Video and image processing
  - Network processing
  - DNA alignment
  - Encryption and compression
  - Deep Neural Network inferencing
  - …

- Intel acquired Altera in 2015 for $16.7B (the largest acquisition ever by Intel)
- AMD acquired Xilinx in 2022 for $49B (the largest in the semiconductor industry)
- FPGA starts to emerge in cloud computing services (AWS, Alibaba, Microsoft, …)

UIUC HACC Infrastructure

Cluster 1: 6 primary nodes:

- 10 Alveo FPGA cards
- 2 SN1000 SmartNICs
- 1 Titan RTX
- 4 Tesla V100
- 3 Versal VCK190

100G networking between all nodes

SLURM job manager to flexibly share resources

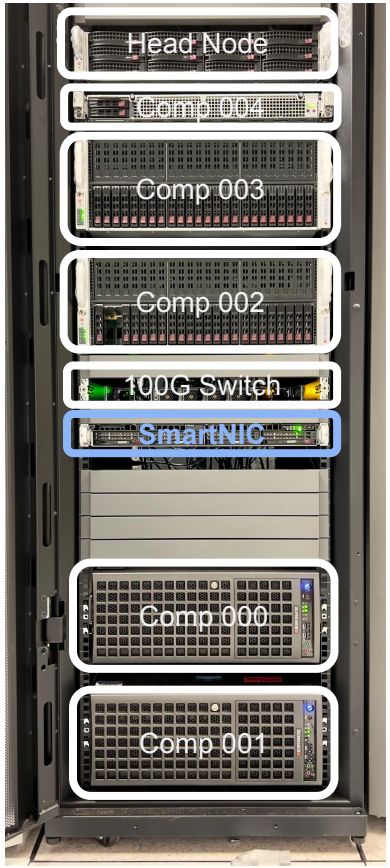
Unique features:

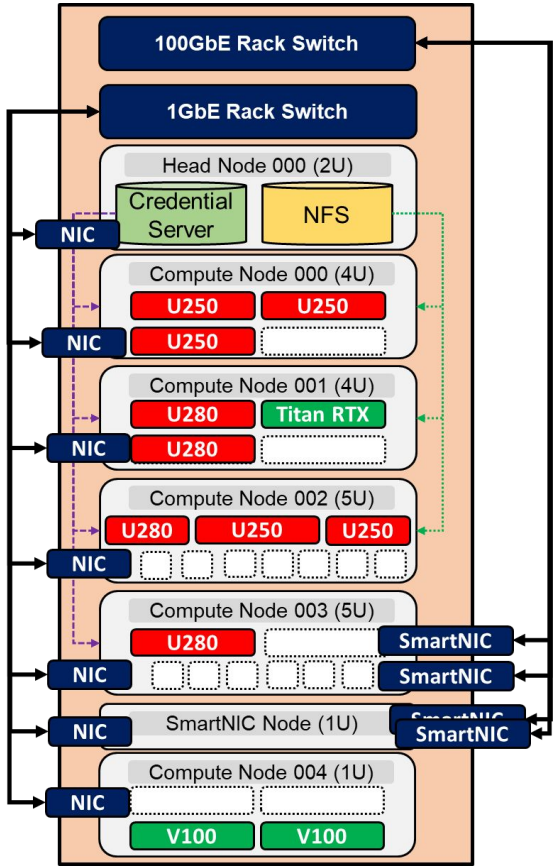
- Heterogenous: U250+U280; FPGA+GPU
- High speed: 100 GbE switch and link
- Flexible: job on single FPGA or networked FPGAs
- Smart: NIC & SmartNIC across multiple nodes
- Adaptive!

Cluster 1: set up in 2020
Cluster 2: set up in 2023

Support users from more than 20 US universities

https://xilinx-center.csl.illinois.edu/

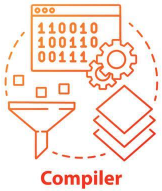


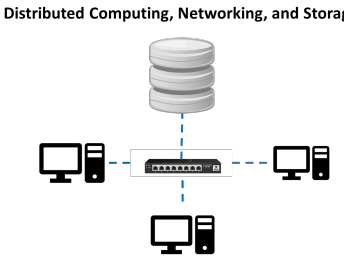


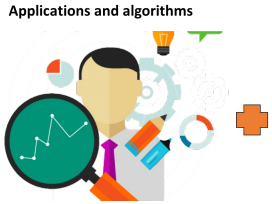
Cluster 1

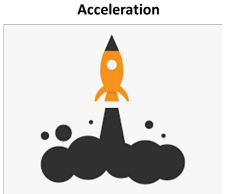
Research Overview

- Research Center + HACC investigating:
  - **System solutions** for high-performance computing (HPC), distributed computing, machine learning (ML) and other computation intensive applications.
  - Create new opportunities in **data center and HPC domains**, with a focus on increasing the flexibility of the memory hierarchy and exploiting parallelism at a large scale with **networked FPGAs and other types of accelerators.**
- Four research thrusts:
  - Compilers and languages
  - Systems solutions for heterogenous computing
  - Distributed computing, networking, and storage
  - Applications and algorithms acceleration



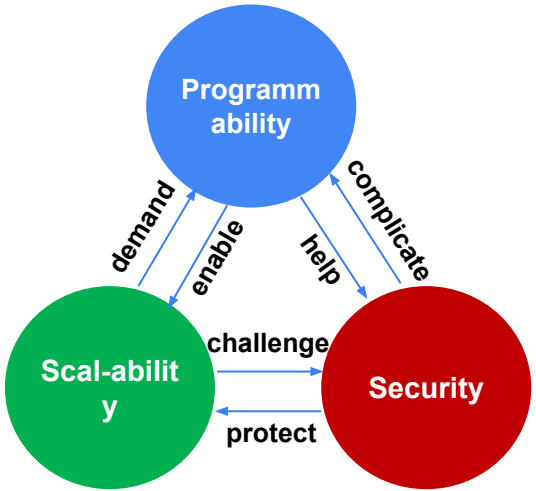






But, it is still not mainstream in the cloud yet

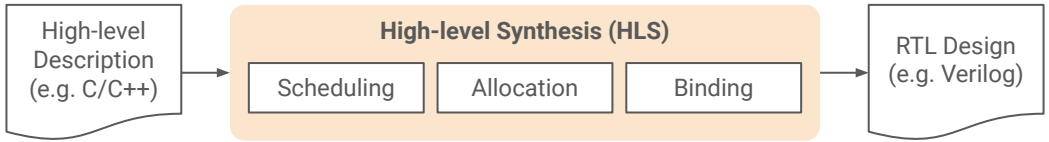
- Why?
  - **Programmability challenge**
    - Conventional designs still rely on Verilog code and hardware design expertise
    - Still use physical addresses managed by programmers (i.e., hardware designers)
  - **Scalability challenge**
    - Difficult to build and manage an FPGA cluster
    - No direct FPGA-to-FPGA communication
    - Difficult to virtualize FPGAs
    - Relatively small device memory and on-chip memory
  - **Security challenge**
    - Security for hardware accelerators is still in its infancy
  - Others
    - Floating point computation is relatively limited
    - Not a commodity product yet (may come soon)



The Rest of the Talk

- Programmability through HLS
- Scalability through SVM
- Security through TEE
- Future work
  - Focus on Chiplet/3D designs

Programmability: Promises and Challenges of HLS



High-level Synthesis (HLS) is great

- **Reduce design complexity:** Code density can be reduced by 7x - 8x moving from RTL to C/C++ [1]
- **Improve design productivity:** Get to working designs faster and reduce time-to-market [1]
- **Identify performance-area trade-offs:** Implement design choices quickly and avoid premature optimization [2]

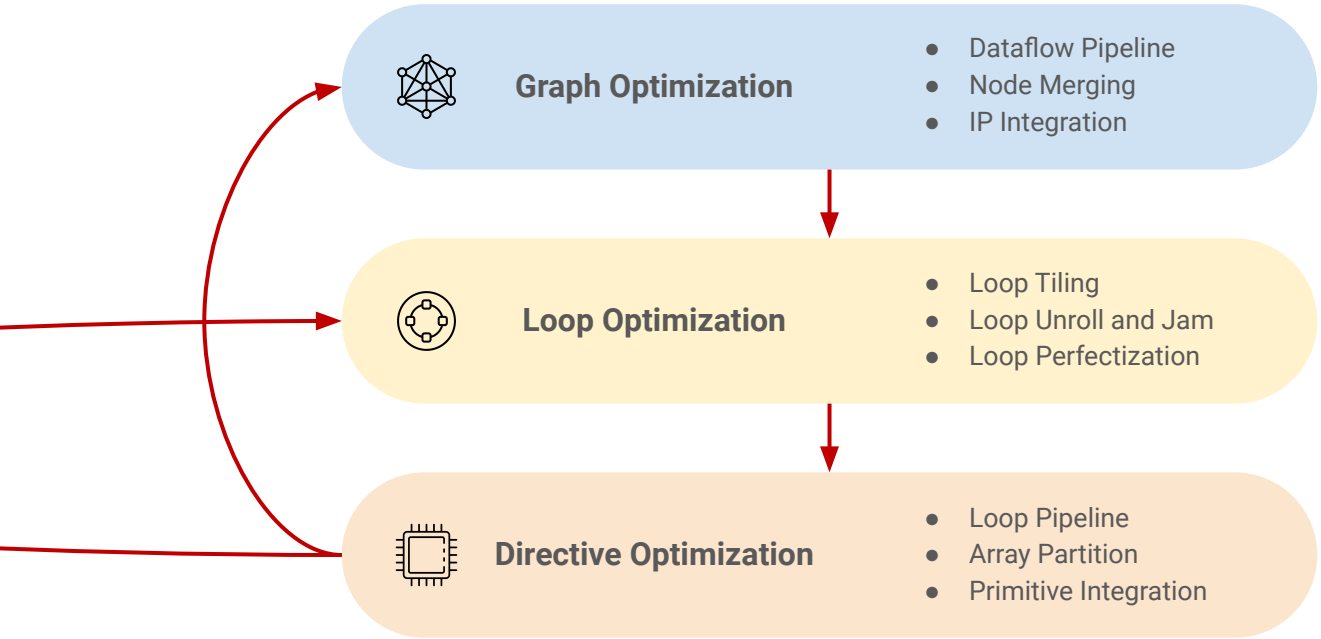
Designing HLS accelerators is non-trivial

- **Friendly to experts:** Rely on the designers writing 'good' code to achieve high design quality [3]
- **Large design space:** Many different combinations of applicable optimizations for large designs [2]
- **Correlation of design factors:** Difficult for human to discover complicated correlations for large designs [4]

We need a scalable compilation flow to automatically explore the design space and optimize HLS designs.

[1] J. Cong, et al. High-Level Synthesis for FPGAs: From Prototyping to Deployment. 2011. TCAD.
[2] B. C. Schafer, et al. High-Level Synthesis Design Space Exploration: Past, Present, and Future. 2020. TCAD.
[3] A. Sohrabizadeh, et al. AutoDSE: Enabling Software Programmers to Design Efficient FPGA Accelerators. 2022. TODAES.
[4] M. Yu. Chimera: A Hybrid Machine Learning-Driven Multi-Objective Design Space Exploration Tool for FPGA High-Level Synthesis. 2021. IDEAL.

Marry HLS and MLIR ➡ **ScaleHLS**

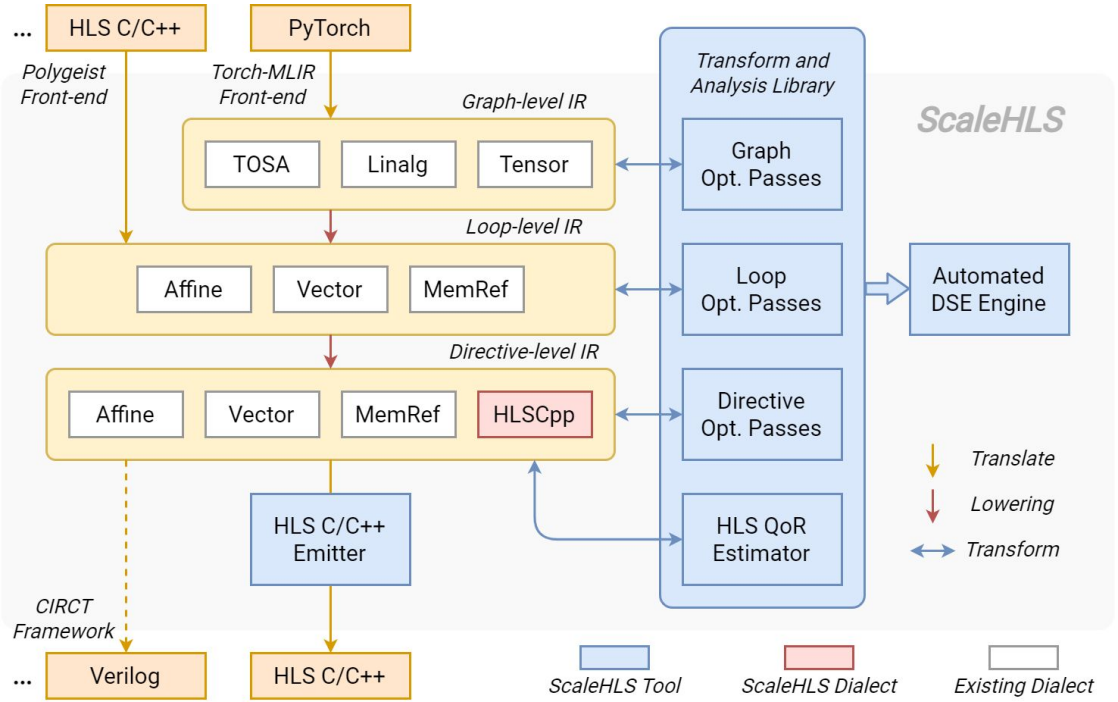


MLIR [1]

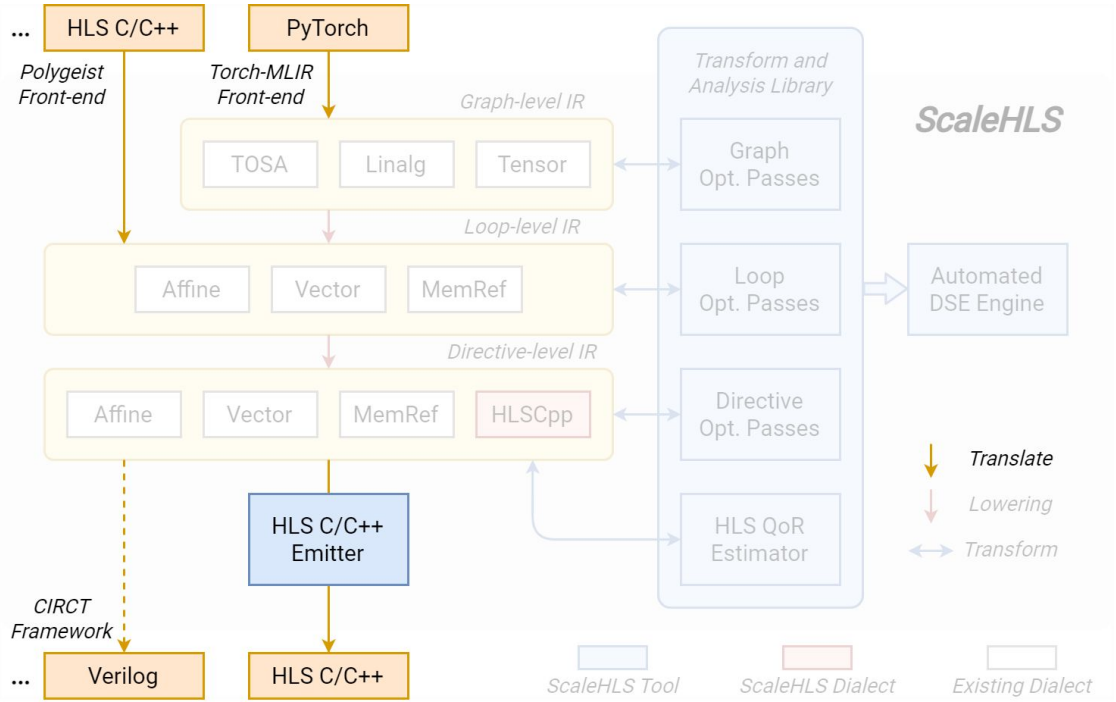
https://mlir.llvm.org

[1] C. Lattner, et al. MLIR: Scaling Compiler Infrastructure for Domain Specific Computation. 2021. CGO.

ScaleHLS: Integration



ScaleHLS: Integration



Inputs

C/C++ Polygeist [1]

PyTorch Torch-MLIR [2]

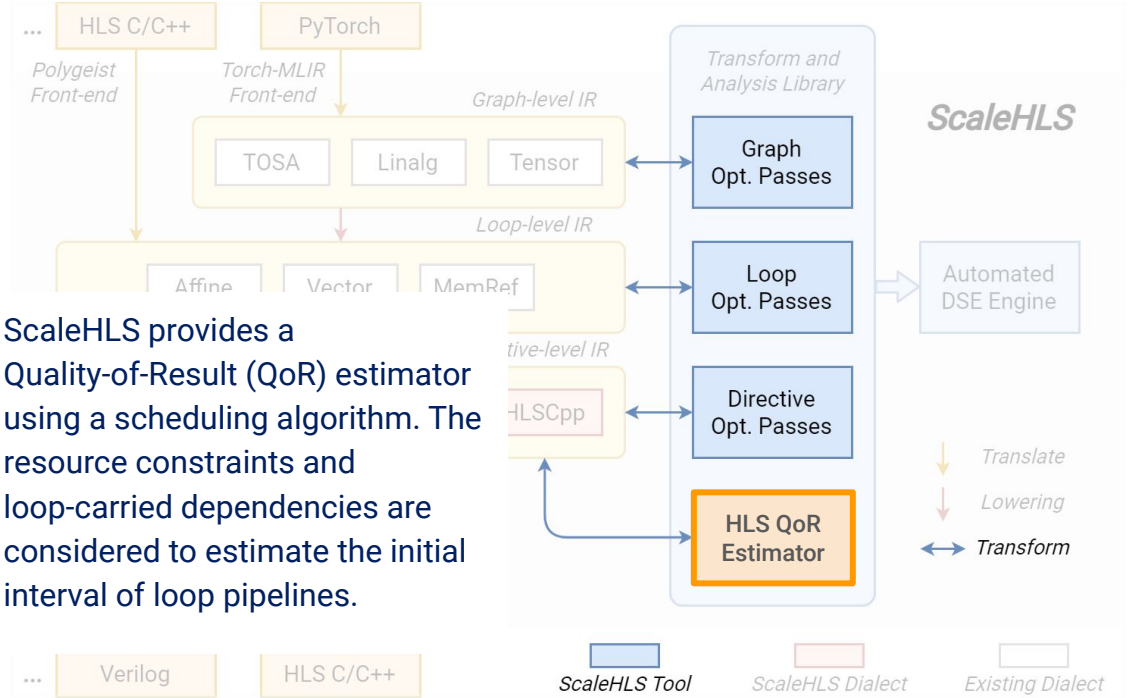
Outputs

Vitis HLS C/C++
C/C++ Emitter

Verilog CIRCT [3]
(work-in-progress)

[1] Polygeist: https://github.com/wsmoses/Polygeist [2] Torch-MLIR: https://github.com/llvm/torch-mlir [3] CIRCT: https://github.com/llvm/circt

ScaleHLS: Optimization Overview



ScaleHLS provides a Quality-of-Result (QoR) estimator using a scheduling algorithm. The resource constraints and loop-carried dependencies are considered to estimate the initial interval of loop pipelines.

Level	ScaleHLS Passes

Graph	`-simplify-tosa-graph`
	`-tosa-node-fusion`
	`-legalize-dataflow`

Loop	`-affine-loop-perfectization`
	`-remove-variable-bound`
	`-affine-loop-tile`
	`-affine-loop-order-opt`
	`-affine-loop-unroll-jam`
	`-simplify-affine-if`

Memory	`-affine-store-forward`
	`-simplify-memref-access`
	`-reduce-initial-interval`

Directive	`-loop-pipelining`
	`-function-pipelining`
	`-array-partition`
	`-create-hlscpp-primitive`
	`-qor-estimation`

Evaluation of C/C++: Single Loop Band

Kernel	Prob. Size	Speedup	LP	RVB	Perm. Map	Tiling Sizes	Pipeline II	Array Partition Factors
BICG	4096	41.7×	No	No	[1, 0]	[16, 8]	43	A:[8, 16], s:[16], q:[8], p:[16], r:[8]
GEMM	4096	768.1×	Yes	No	[1, 2, 0]	[8, 1, 16]	3	C:[1, 16], A:[1, 8], B:[8, 16]
GESUMMV	4096	199.1×	Yes	No	[1, 0]	[8, 16]	9	A:[16, 8], B:[16, 8], tmp:[16], x:[8], y:[16]
SYR2K	4096	384.0×	Yes	Yes	[1, 2, 0]	[8, 4, 4]	8	C:[4, 4], A:[4, 8], B:[4, 8]
SYRK	4096	384.1×	Yes	Yes	[1, 2, 0]	[64, 1, 1]	3	C:[1, 1], A:[1, 64]
TRMM	4096	590.9×	Yes	Yes	[1, 2, 0]	[4, 4, 32]	13	A:[4, 4], B:[4, 32]

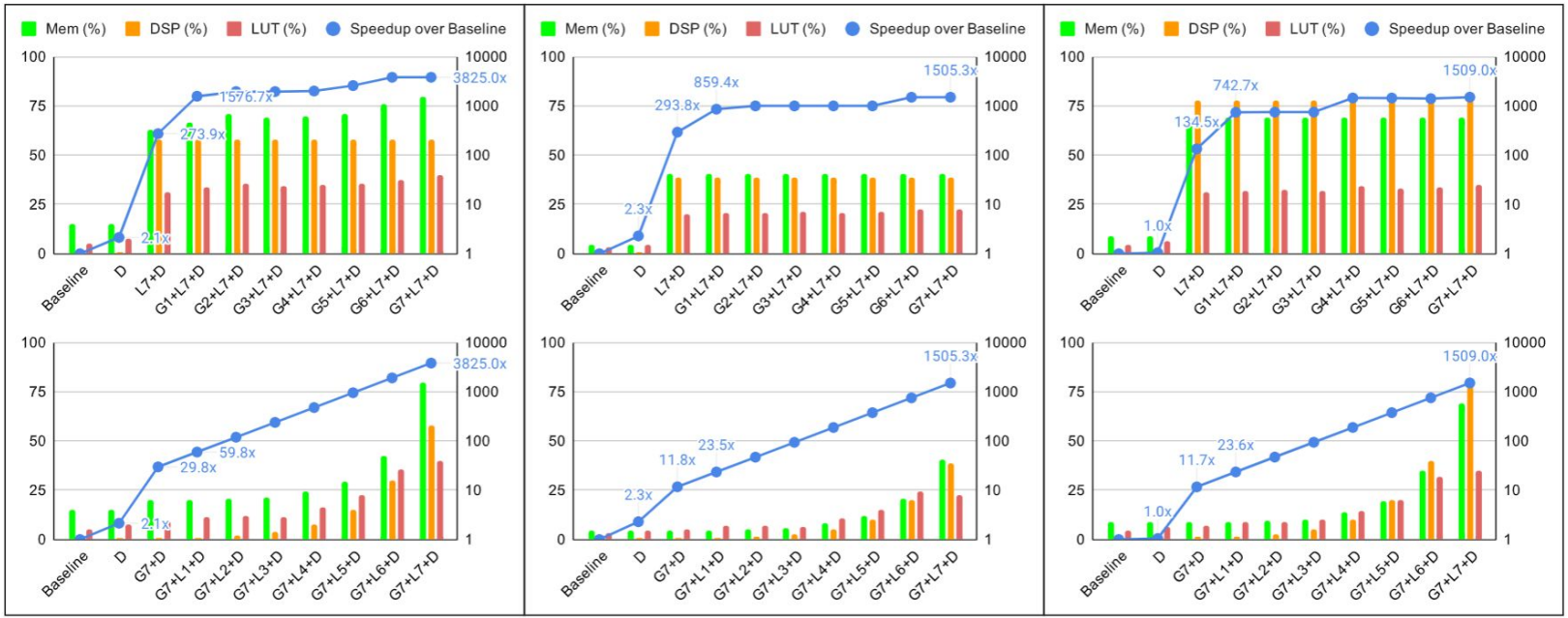
- The target platform is AMD-Xilinx XC7Z020 FPGA. Benchmarks are from PolyBench-C.
- Optimization parameters are automatically selected by the design space exploration (DSE) engine.
- The speedups are compared to the original computation kernels without DSE or any manual optimizations.

Evaluation of PyTorch: Multi-Level Optimization

Model	Speedup	Runtime (seconds)	Memory (SLR Util. %)	DSP (SLR Util. %)	LUT (SLR Util. %)	Our DSP Effi. (OP/Cycle/DSP)	DSP Effi. of TVM-VTA [49]
ResNet-18	3825.0×	60.8	91.7Mb (79.5%)	1326 (58.2%)	157902 (40.1%)	1.343	0.344
VGG-16	1505.3×	37.3	46.7Mb (40.5%)	878 (38.5%)	88108 (22.4%)	0.744	0.296
MobileNet	1509.0×	38.1	79.4Mb (68.9%)	1774 (77.8%)	138060 (35.0%)	0.791	0.468

- The target platform is one SLR (super logic region) of AMD-Xilinx VU9P FPGA.
- The PyTorch models are parsed into ScaleHLS and optimized at multiple levels, including graph, loop, and directive optimizations.
- The speedups are compared to the baseline designs, which are compiled from PyTorch to HLS C/C++ through ScaleHLS but without the multi-level optimization applied.

Evaluation of PyTorch: Ablation Study



(a) ResNet-18 **(b)** VGG-16 **(c)** MobileNet

D, *L{n}*, and *G{n}* denote directive, loop, and graph optimizations, respectively. Larger *n* indicates larger loop unrolling factor and finer dataflow granularity for loop and graph optimizations, respectively.

ScaleHLS is Open Source

ScaleHLS GitHub Repository
https://github.com/hanchenye/scalehls

ScaleHLS has around 30,000 views and 2,500 downloads since 2022 🔥🔥

For HLS Researchers

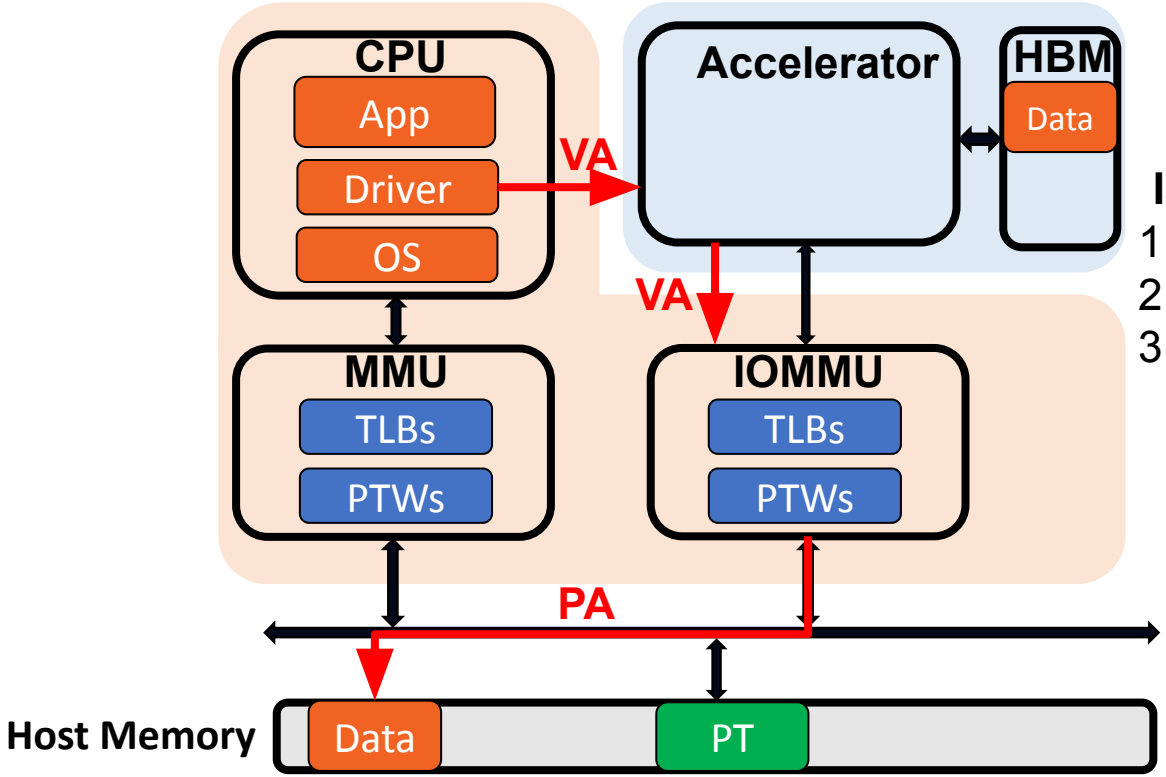
1. Rapidly implement new HLS optimization algorithms on top of the multi-level IR
2. Investigate new DSE algorithms using the transform and analysis library
3. Rapidly build an end-to-end HLS optimization flow and demonstrate your awesome works!

For HLS Users

1. Optimize HLS designs using the multi-level optimization passes
2. Avoid premature design choices by using the QoR estimator to estimate the latency and utilization
3. Find optimized HLS designs with the automated DSE engine

[1] H. Ye, et al., “ScaleHLS: A New Scalable High-Level Synthesis Framework on Multi-Level Intermediate Representation”, HPCA, 2022.
[2] H. Ye, et al., “Invited: ScaleHLS, a Scalable High-Level Synthesis Framework with Multi-level Transformations and Optimizations”, DAC, 2022.
[3] H. Ye, et al., “High-level Synthesis for Domain Specific Computing”, ISPD, 2023.

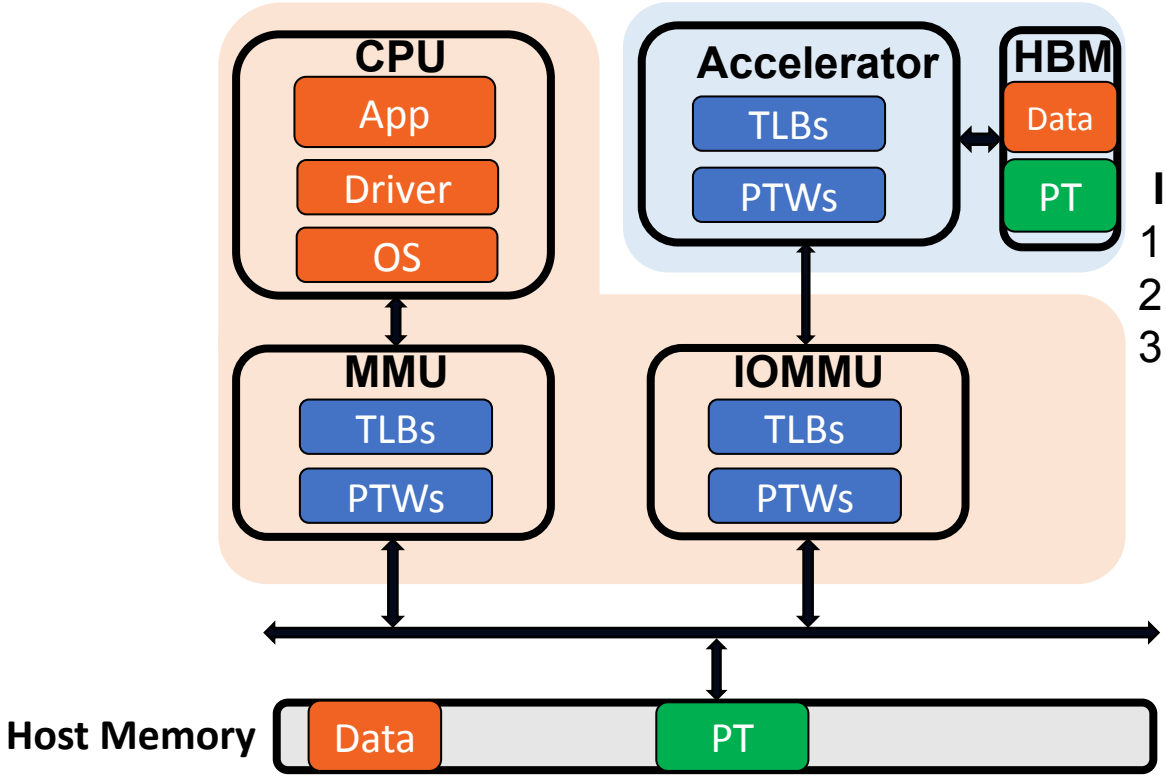
Scalability: Shared Virtual Memory (SVM) with FPGAs



Issues:

1. Isolation
2. Programmability
3. Performance

Scalability: Shared Virtual Memory (SVM) with FPGAs



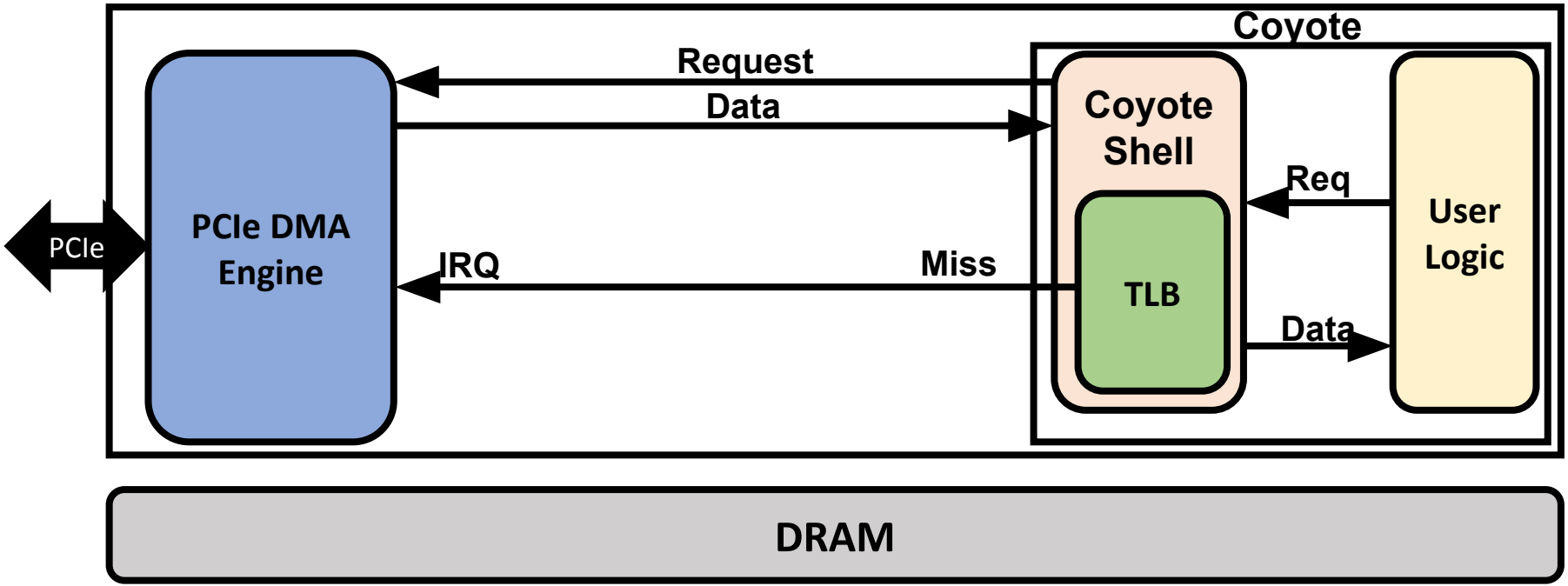
Issues:

1. Isolation
2. Programmability
3. Performance

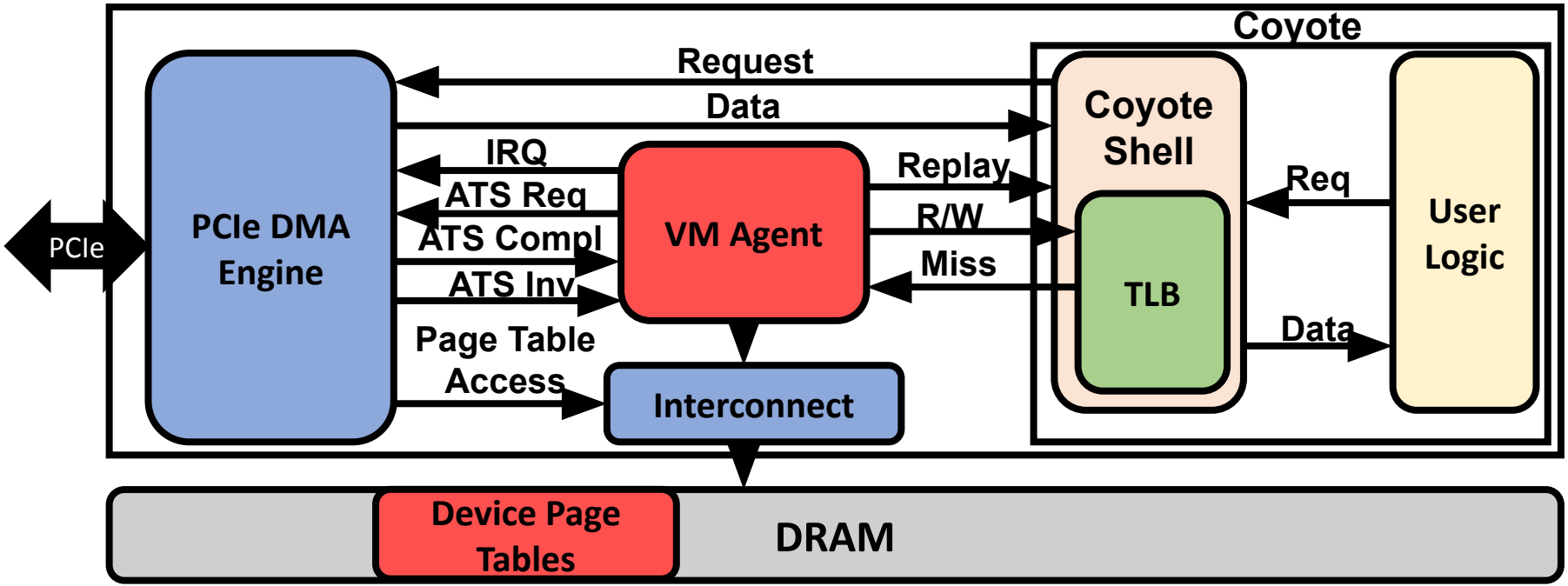
Our Solution: Qilin!

Options	High Speed	Realistic CPU and Interconnect Modeling	Transparent	Flexible
Industry SVM Implementation	✓	✓	✗	✗
Researchers need a system to provide fast, accurate, transparent, and flexible SVM systems!				
FPGA	✓	✓		

Qilin Systems Design (1/2)



Qilin Systems Design (2/2)



Methodology – Experimental Platform

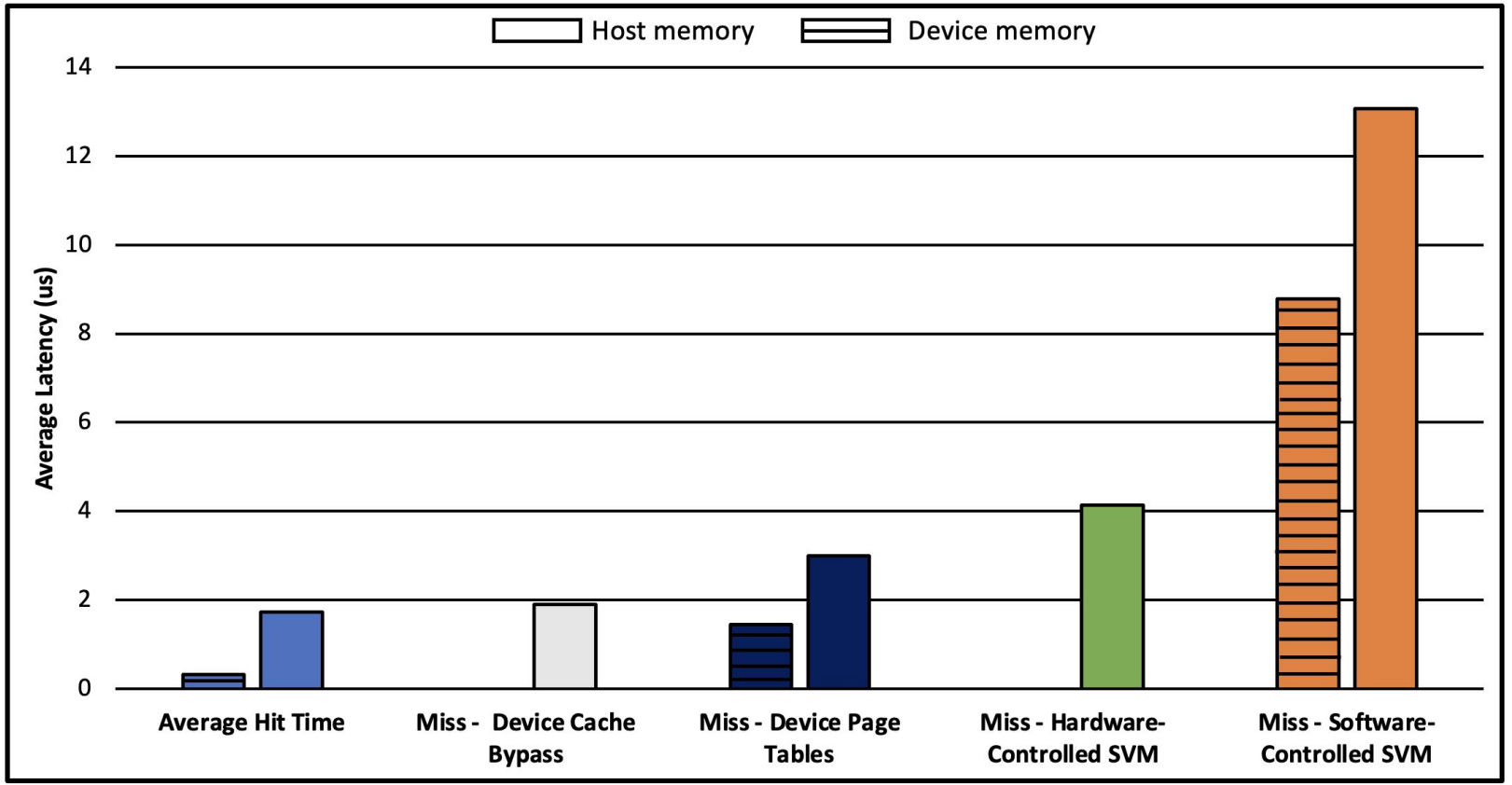
Platform Specs

Category	Specification
CPU	AMD Ryzen 9 3900X 12C/24T 3.8 GHz
Memory	DDR4 3200MHz 32GB in Dual Channel Mode
FPGA	Alveo U280 using x16 PCIe 3.0
OS	Ubuntu 18.04 Kernel 5.4.0-100-generic
Tools	Xilinx Vivado 2020.2.1

Test Platform

E. Richter and D. Chen, “Qilin: Enabling Performance Analysis and Optimization of Shared-Virtual Memory Systems with FPGA Accelerators,” ICCAD, 2022.

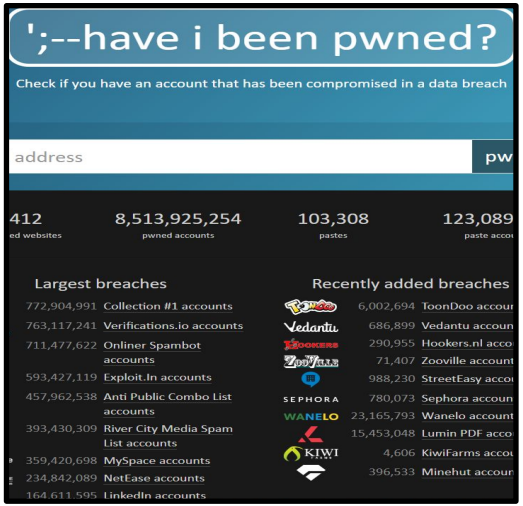
Host Memory vs Device Memory



Security: Even “Secure” Cloud Is Not Free of Data Breaches

Verizon

LinkedIn



Pwned?

Challenge: there is no well-established framework targeting at the protection of host-FPGA systems

Trusted Execution Environment (TEE)

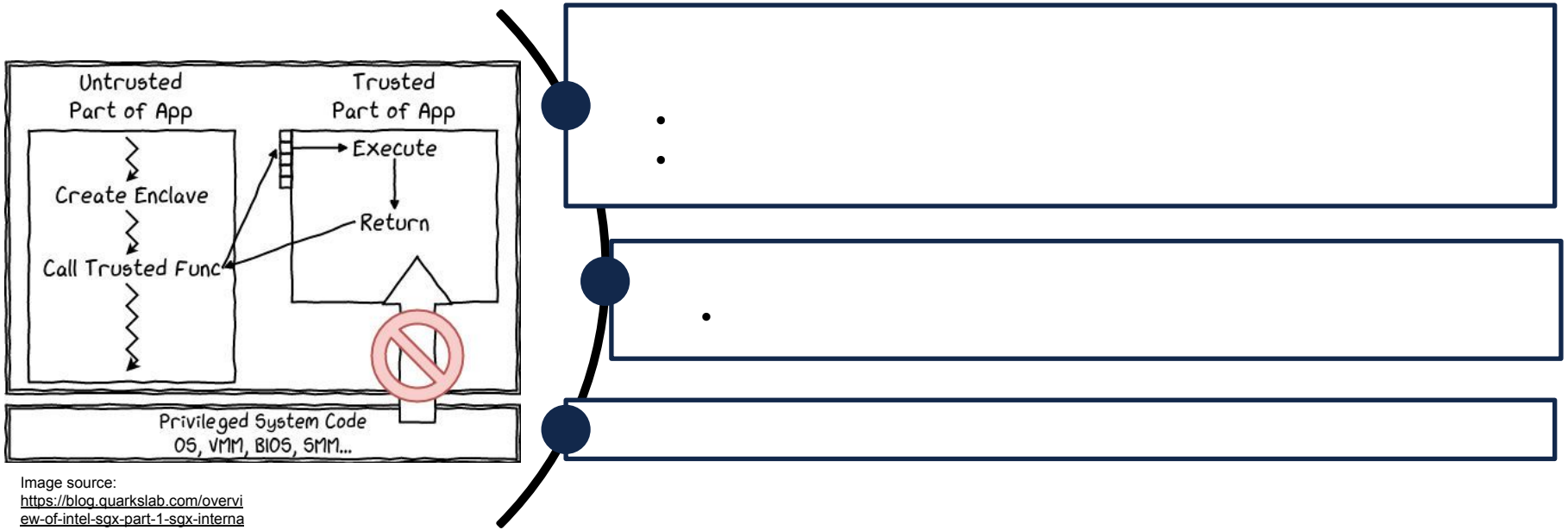
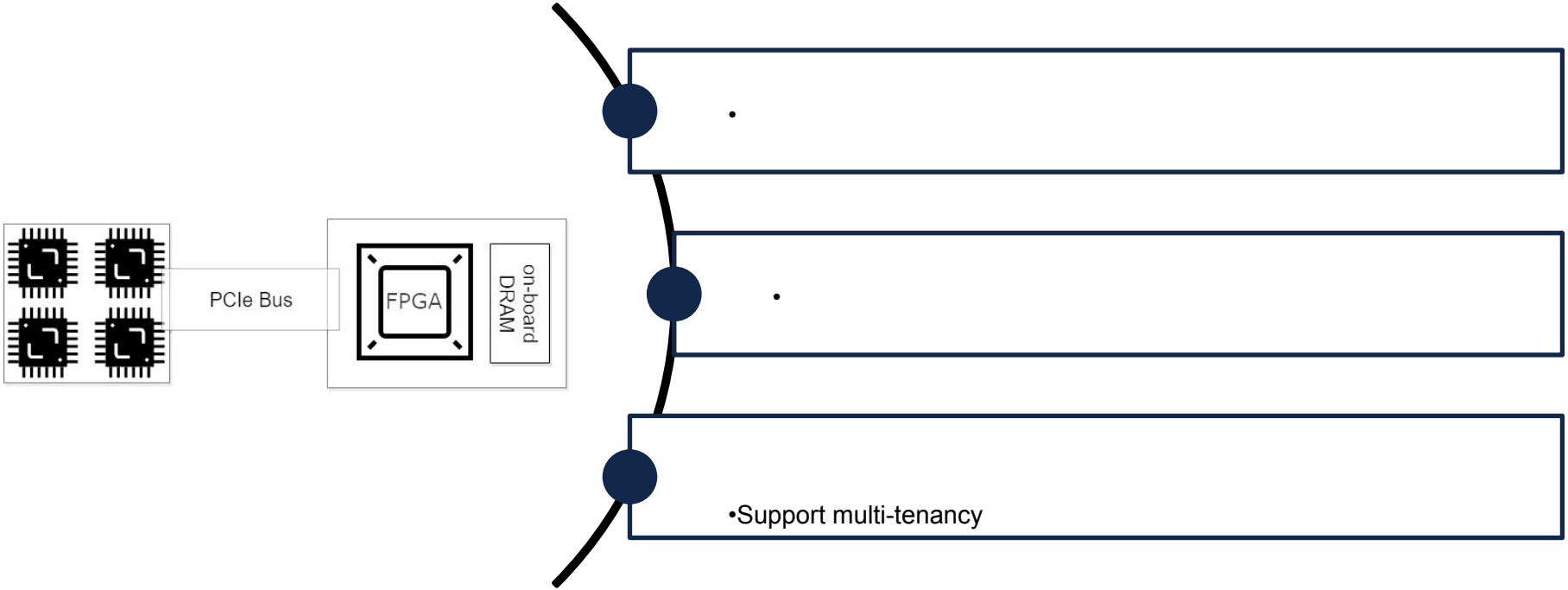


Image source:
https://blog.quarkslab.com/overview-of-intel-sgx-part-1-sgx-internals.html

Our goal – provide isolation and privacy guarantee to FPGA accelerators

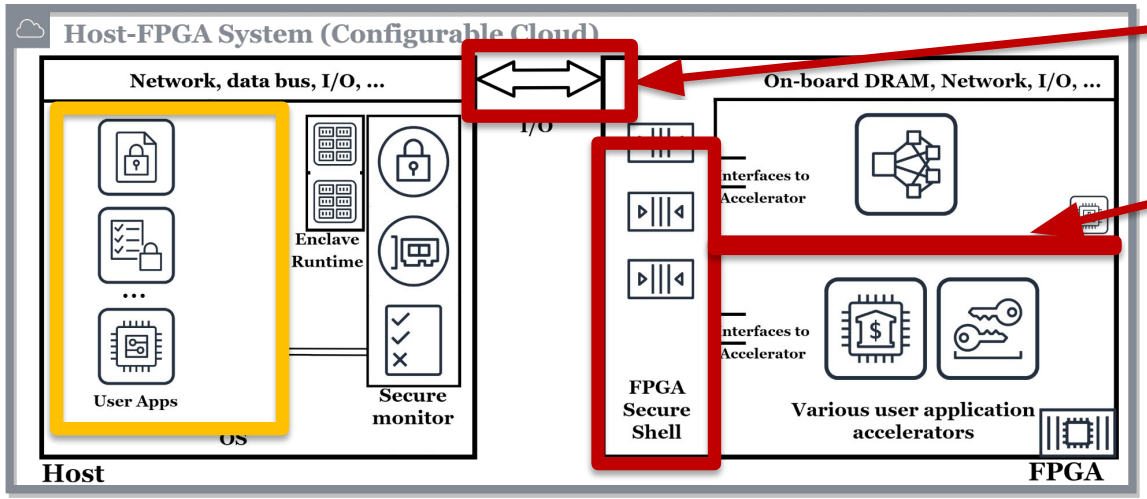
Our Contributions



[1] W. Ren, J. Pan, and D. Chen, “AccGuard: Secure and Trusted Computation on Remote FPGA Accelerators”, iSES, 2021.
[2] W. Ren, et al., “AccShield: a New Trusted Execution Environment with Machine-Learning Accelerators,” DAC, 2023.

Isolation for Security – FPGA Side

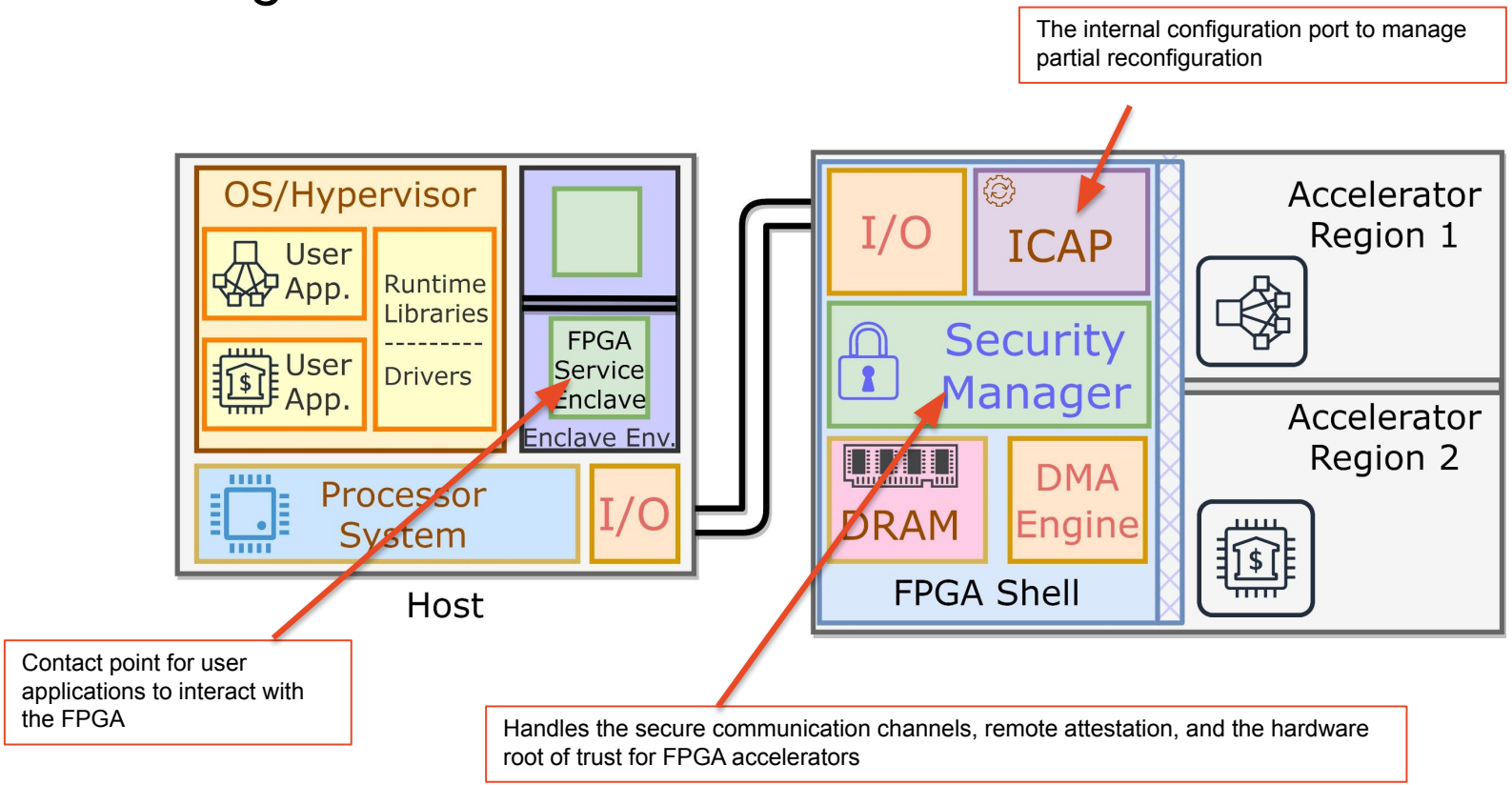
- Secure communication channels
- Physical (design) isolation – partial reconfiguration
- Logical isolation – Secure Monitor (SM) and FPGA Security Manager
    - Enforce strict resource and access control



Encrypted communication – separate secure channels

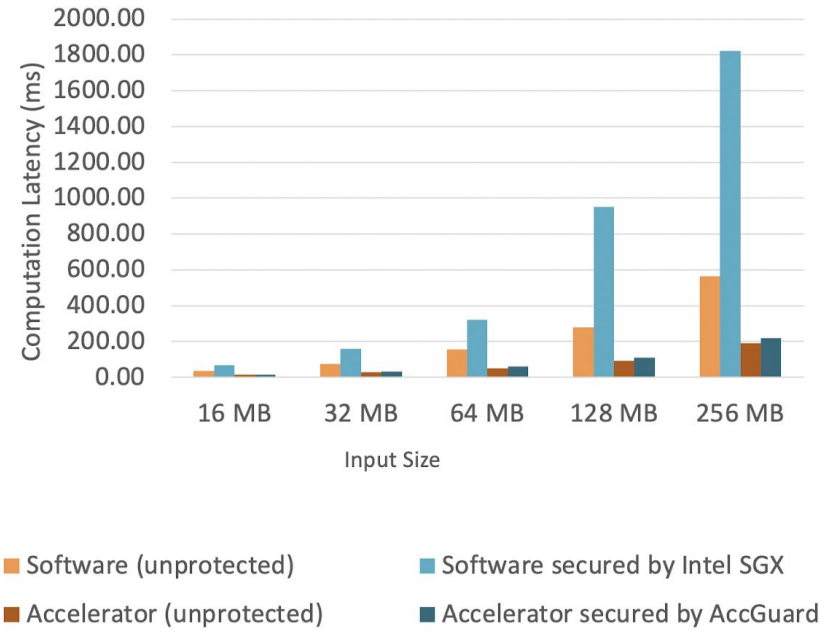
Isolation between accelerators

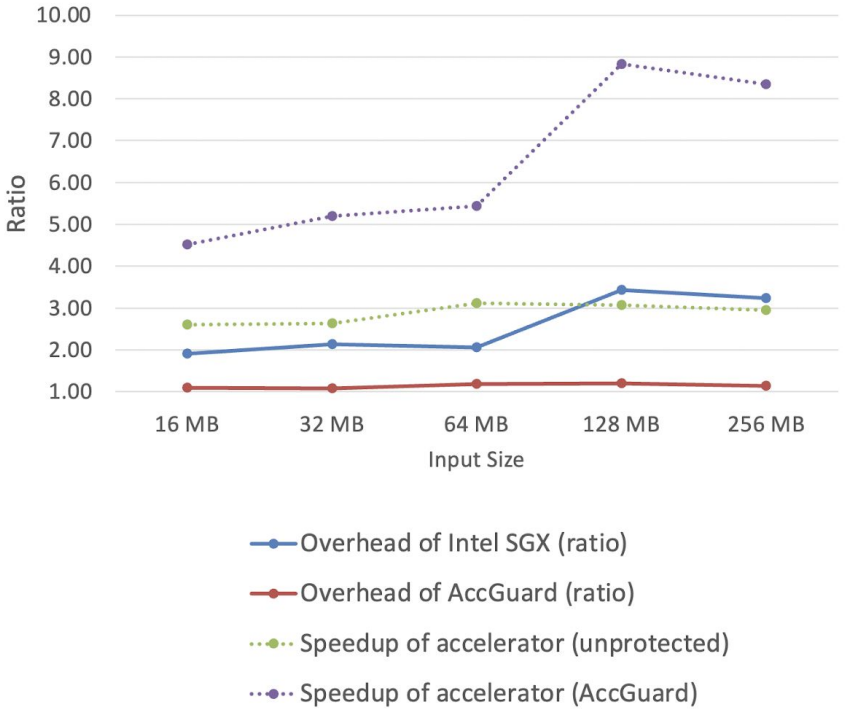
Block Diagram of AccGuard



Support hardware root of trust and remote attestation

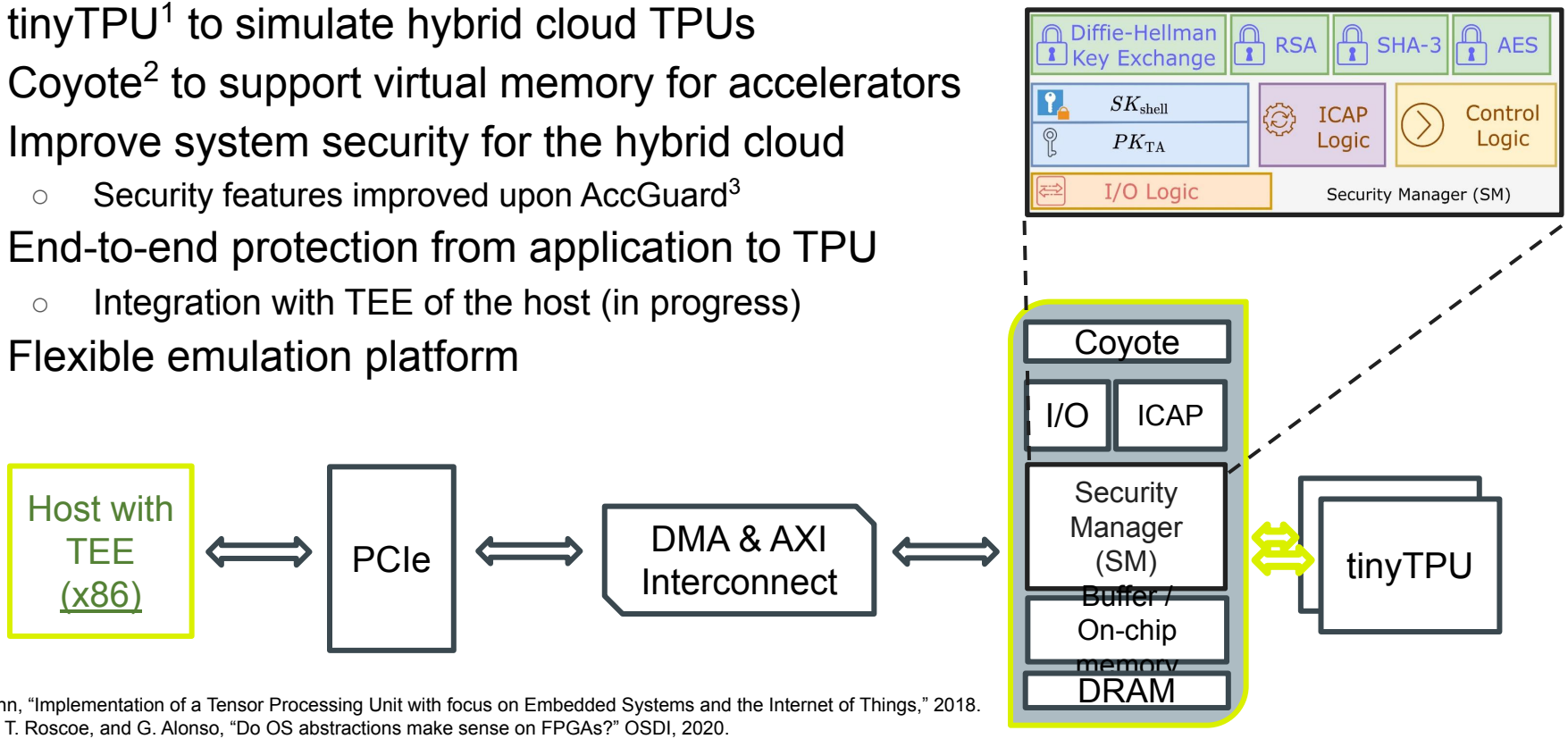
Results – Computation Latency & Overhead





What is AccShield [DAC'23]?

- tinyTPU[1] to simulate hybrid cloud TPUs
- Coyote[2] to support virtual memory for accelerators
- Improve system security for the hybrid cloud
  - Security features improved upon AccGuard[3]
- End-to-end protection from application to TPU
  - Integration with TEE of the host (in progress)
- Flexible emulation platform



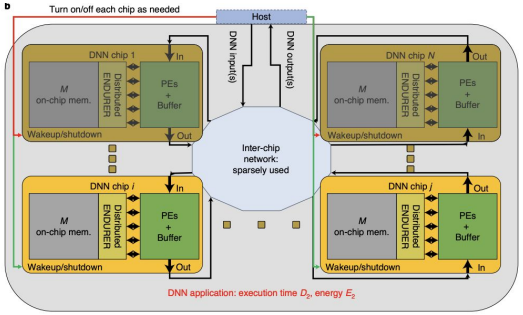
[1] Fuhrmann, "Implementation of a Tensor Processing Unit with focus on Embedded Systems and the Internet of Things," 2018.
[2] Korolija, T. Roscoe, and G. Alonso, "Do OS abstractions make sense on FPGAs?" OSDI, 2020.
[3] W. Ren, J. Pan and D. Chen, "AccGuard: Secure and Trusted Computation on Remote FPGA Accelerators," iSES, 2021.

- How would all these relate to next-generation chiplet-based designs or 3D designs?

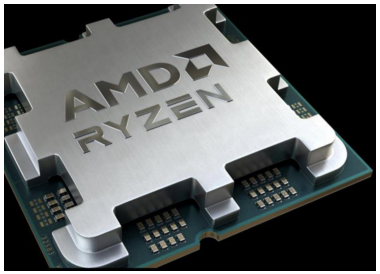
Demands for Next-Generation Chiplet/3D Chips

- Customized chiplet designs for cores, on-chip memories, off-chip memories, supporting different packaging:
  - e.g., TSV 3D, bonding, monolithic 3D, 2.5D interposers
- Customized and distributed communication systems to connect all the chiplets together efficiently.
- Performance maximization for such a large design while meeting other important design constraints such as robustness and energy efficiency.
- Need novel EDA tools for
  - chiplet-communication-packaging co-design
  - design scalability and productivity
  - programmability
  - security
  - exploration of system design tradeoffs
  - verification and testing
  - …

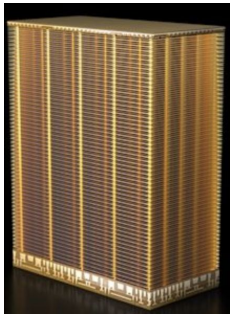
Strong Recent Developments for Chiplet/3D Chips



Illusion. [Nature Electronics, 2021]. Stanford



AMD 3D V-Cache Zen 4 CPUs

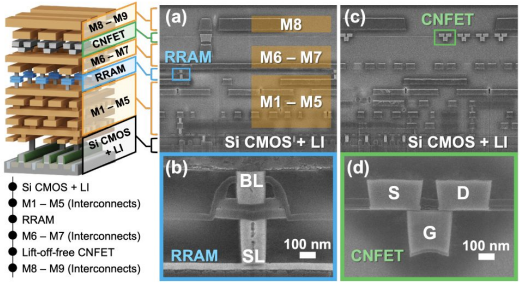


Micron's 232-Layer NAND

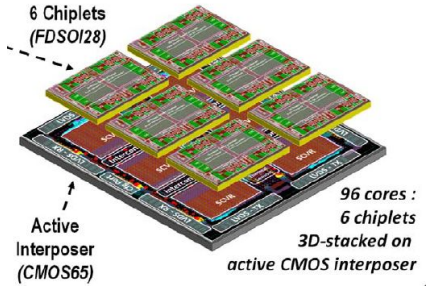
UCIe™

Universal Chiplet Interconnect Express™

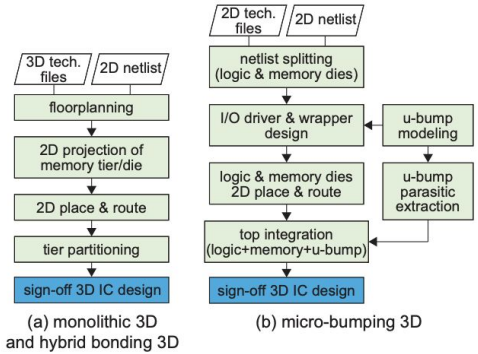
Building an open ecosystem of chiplets for on-package innovations



Monolithic 3D. [VLSI, 2023]. Stanford



96-Core, 6 Chiplets, 3D-Stacked. [ISSCC, 2020]. CEA-LETI



Design flows for 3D ICs. [DAC, 2021]. Georgia Tech.

Conclusions and Future Work

- **Programmability**: ScaleHLS is an open-source MLIR-based HLS compilation flow, which features multi-level representation and optimization of HLS designs.
- **Scalability**: Qilin is an open-source SVM system implemented on an FPGA and host CPU, which allows researchers and application developers to explore the SVM design space.
- **Security**: AccGuard is a unique hardware enclave framework, which supports isolation and remote attestation, leading to a stronger security guarantee of remote application accelerators and data integrity in the cloud.
- **Future work includes**
  - Support for concurrent multiple FPGAs
  - Formal verification of the security protocols
  - Verifiable high-level synthesis
  - Programmability + Scalability + Security
  - Target chiplet-based and 2.5D/3D designs

Acknowledgement

Collaborators:

Hanchen Ye, Cong Hao, Jianyi Cheng, Hyunmin Jeong, Jack Huang, Stephen Neuendorffer, Wei Ren, Junhao Pan, Edward Richter

Sponsors:

Booz | Allen | Hamilton®

AMD XILINX

NSF

SRC

C³SR center for cognitive computing systems research

The Grainger College of Engineering
IBM-Illinois Discovery Accelerator Institute

Thank you!

Questions?