



PULP PLATFORM

Open Source Hardware, the way it should be!

HERO: A Heterogeneous Research Platform to Explore HW/SW Codesign and RISC-V Manycore Accelerators

Luca Bertaccini

<lbartaccini@iis.ee.ethz.ch>

ETH zürich



<http://pulp-platform.org>



[@pulp_platform](https://twitter.com/pulp_platform)



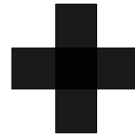
https://www.youtube.com/pulp_platform



Heterogeneous Systems-on-Chip (HeSoCs)

HOST

- General-purpose
- Linux-capable
- Versatility
- Programmability



PMCA

- Parallel Manycore Accelerator (PMCA)
- Domain-specialized
- Energy-efficient



Domain Specialization & Heterogeneity

Energy efficiency challenges
(post-Moore era)



Domain Specialization

Low versatility and
programmability for highly
specialized design



Heterogeneous Systems



Industrial HeSoCs

- Qualcomm Snapdragon 888 (mobile processor – 5nm)
- Apple M1 (MacBook Air, MacBook Pro – 5nm)
- NVIDIA Grace (CPU for AI and HPC – 5nm)
- And many more...



Picture from qualcomm.com



Picture from apple.com

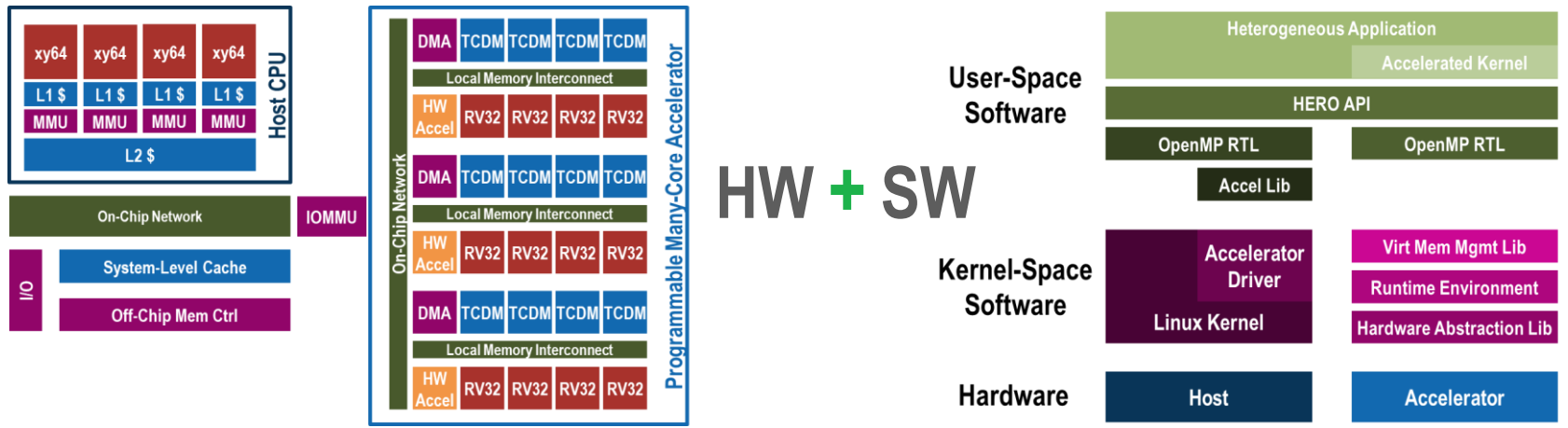


Picture from nvidia.com



HERO: Overview and Goals

HERO = Heterogeneous Research Platform



HW + SW

Enables research and development on heterogeneous computers:

- Algorithms and Applications
- Programming Models, Task Distribution, Scheduling
- Manycore Architectures, Hardware Accelerators, Core Microarchitecture
- Memory Organization, Communication, Synchronization

Memory Organization, Communication, Synchronization ← Focus of this talk





HERO: Hardware

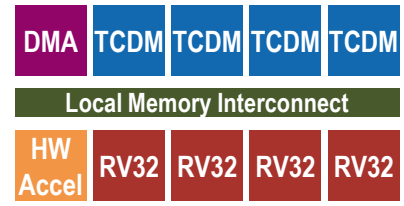
**Integration Host-PMCA
Support for efficient communication**

ETH zürich





HERO: Hardware Architecture



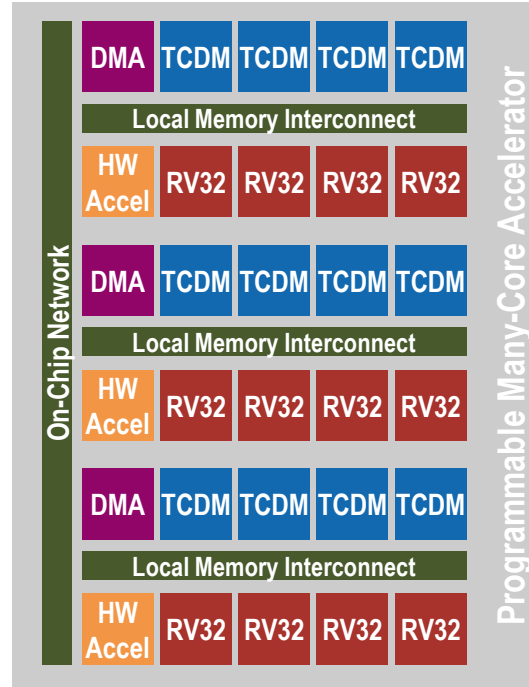
ETH zürich





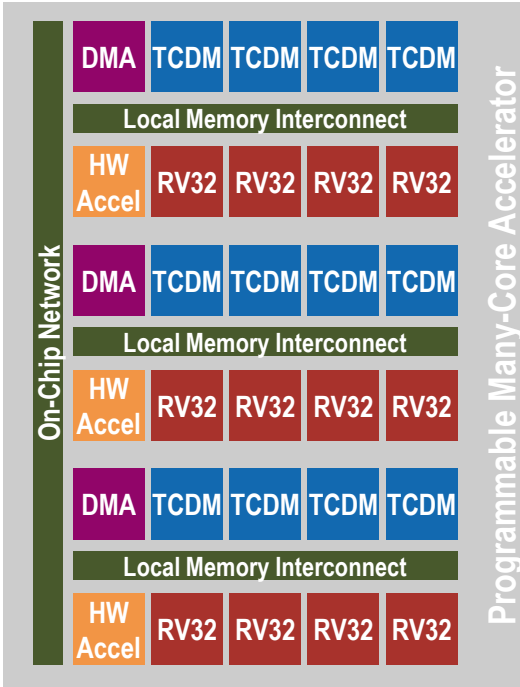
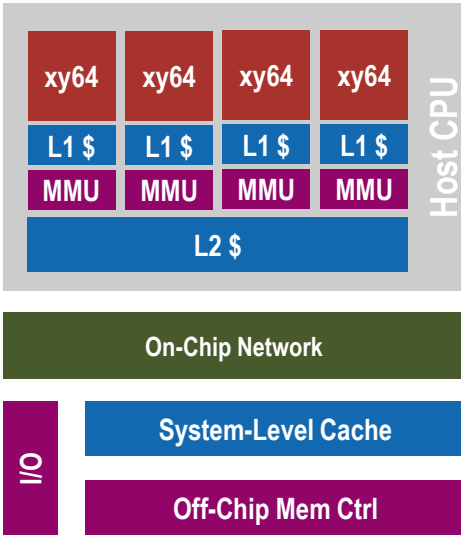
HERO: Hardware Architecture

ETH zürich





HERO: Hardware Architecture

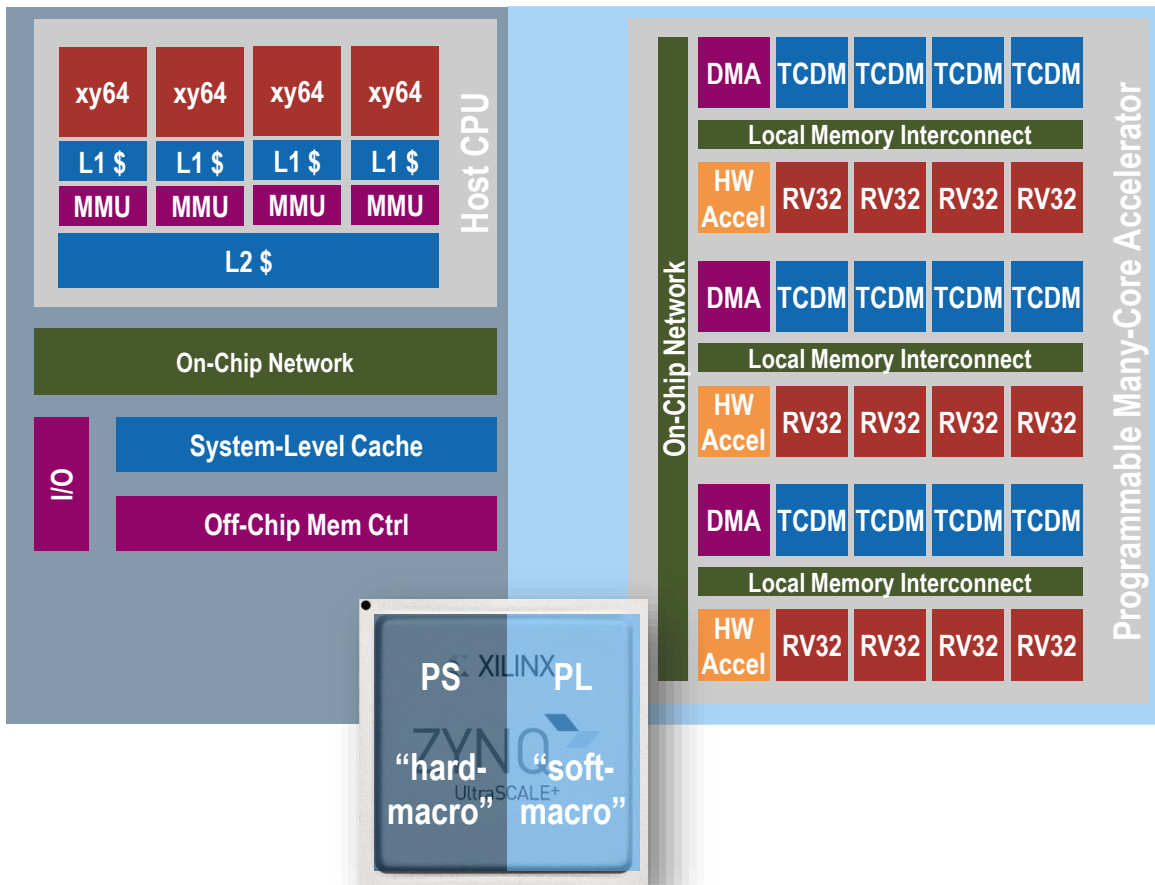


ETH zürich





HERO: Hardware Architecture



ETH zürich





HERO: Hardware Architecture

Virtual Memory
Coherent Cache

Physically-Addressed
Scratchpad Memory

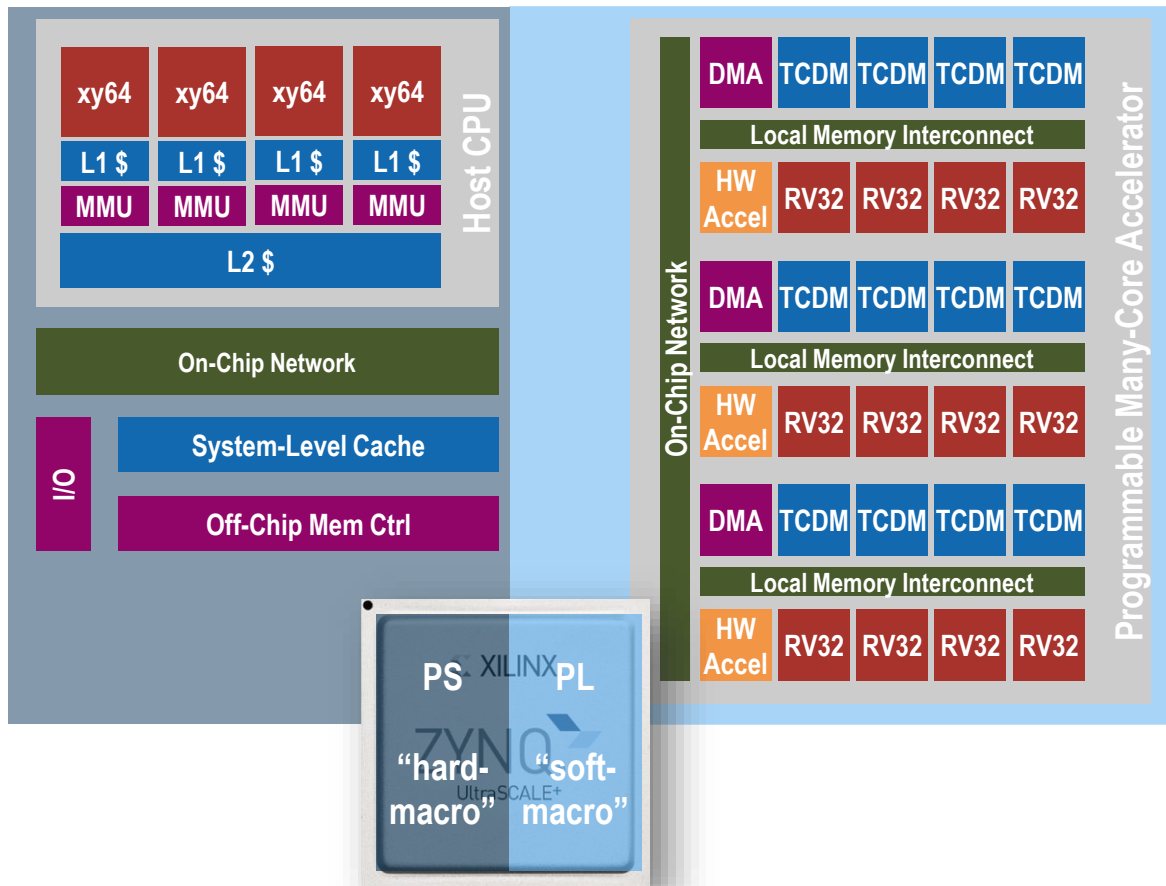


ETH zürich





HERO: Hardware Architecture

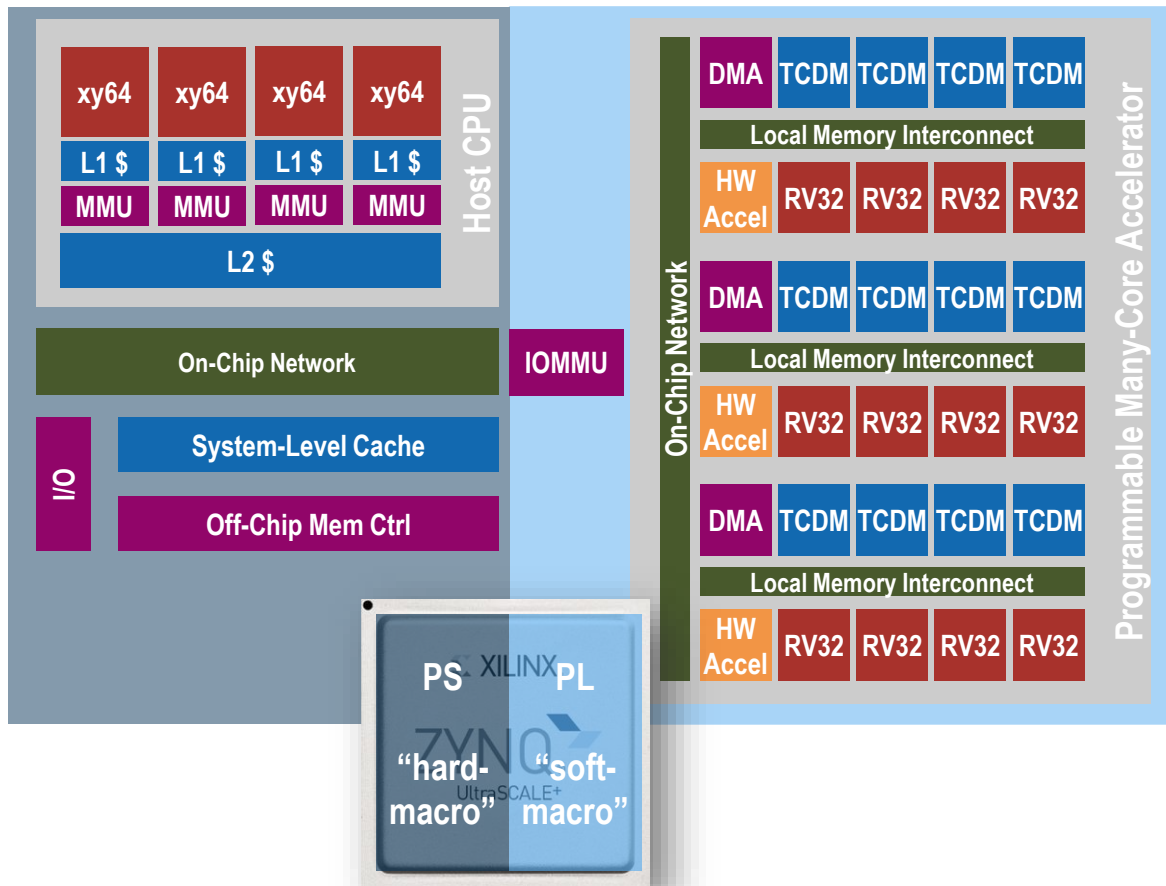


ETH zürich





HERO: Hardware Architecture

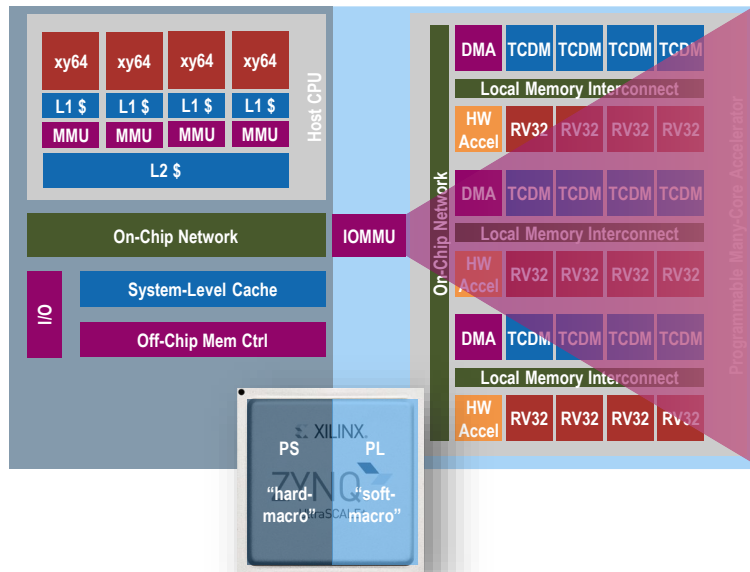


ETH zürich





HERO: Hybrid IOMMU



Hybrid IOMMU

- To bridge the gap between the different memory systems
- SW-controlled TLB
- TLB prefetching
- Shared Virtual Memory (SVM) accessible by DMA transfers without additional buffers in the IOMMU

ETH zürich





HERO: TLB misses handling

ETH zürich



PMCA's
DMA issues
a transaction
that
generates a
TLB miss

IOMMU
responds
with an error
and drops
the
transaction

One of the
PMCA cores
walks the
page table,
adds the
new TLB
entry, and
notifies the
DMA

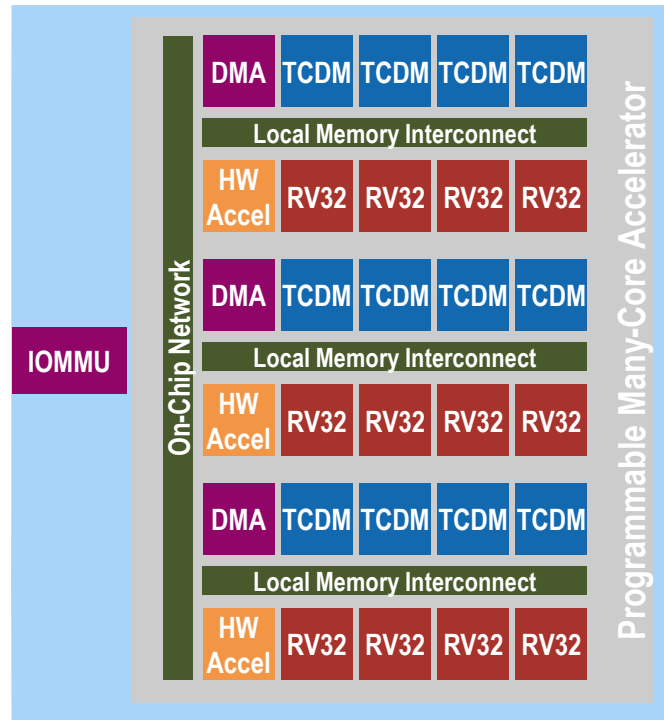
PMCA's
DMA
re-issues the
transaction
whose TLB
miss has
been
resolved



HERO: TLB prefetching

Worker Threads (**WTs**) and Prefetching Helper Threads (**PHTs**). The cores are statically allocated to WTs or PHTs

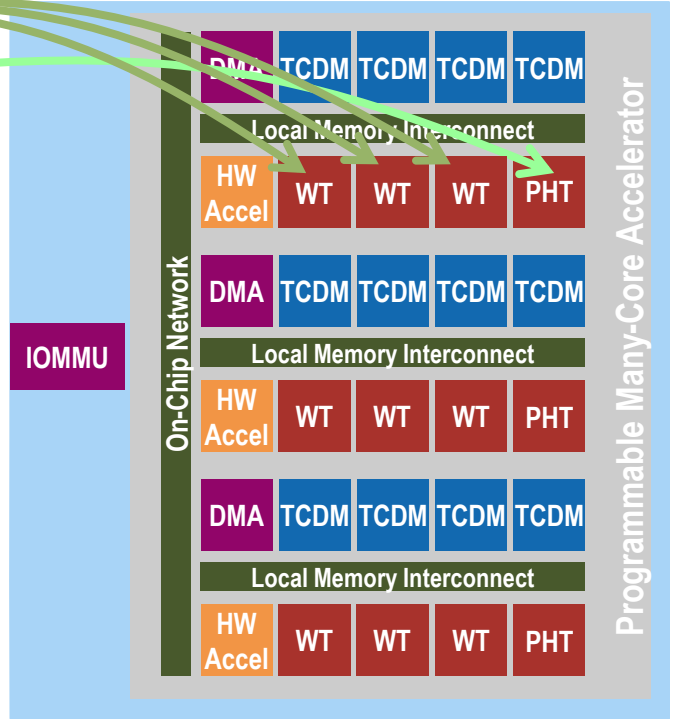
ETH zürich





HERO: TLB prefetching

Worker Threads (WTs) and Prefetching Helper Threads (PHTs). The cores are statically allocated to WTs or PHTs

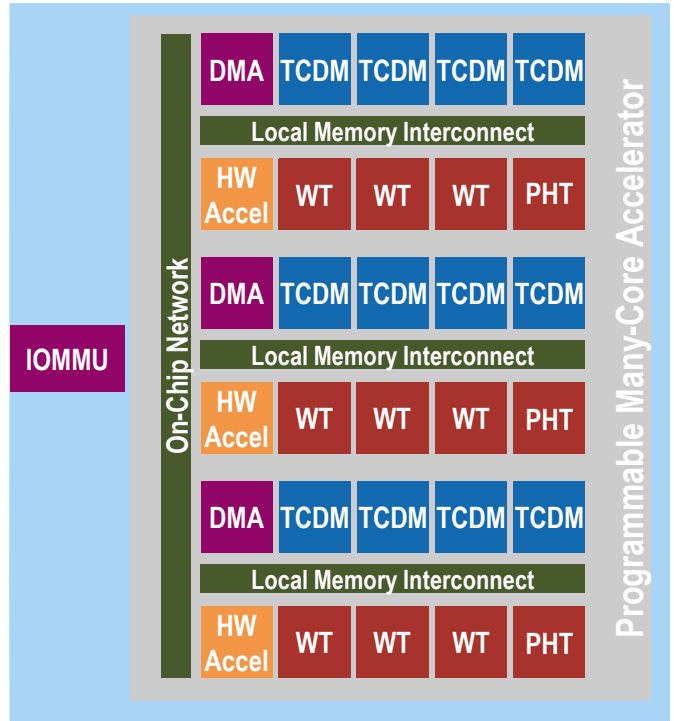




HERO: TLB prefetching

The PHTs are automatically generated by the compiler which checks for SVM accesses in the code

ETH zürich

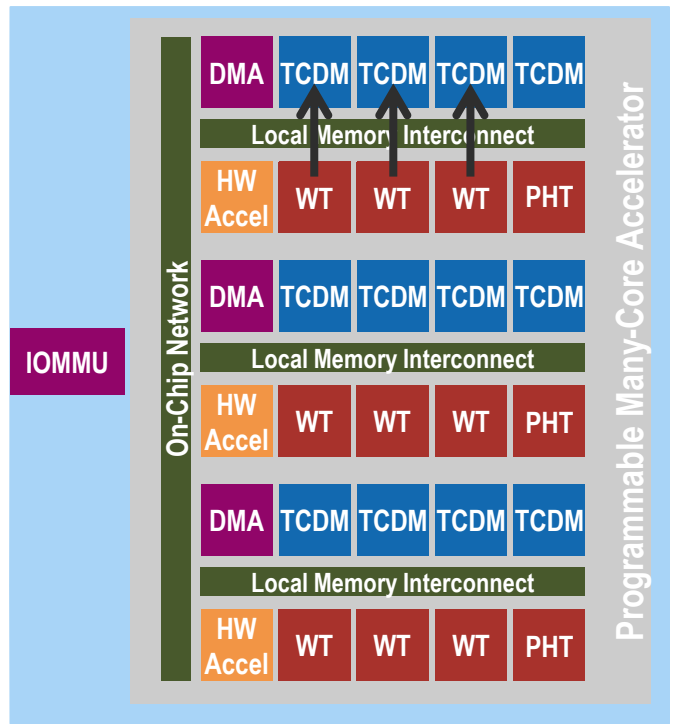




HERO: TLB prefetching

The PHTs are automatically generated by the compiler which checks for SVM accesses in the code

WTs contain additional store instructions to the L1 SPM to share the execution state while PHTs contain additional load instructions



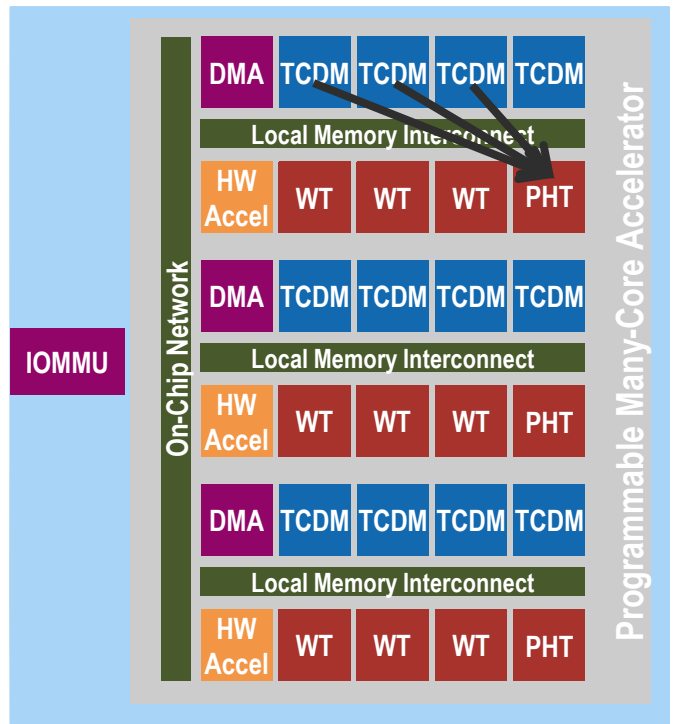


HERO: TLB prefetching

The PHTs are automatically generated by the compiler which checks for SVM accesses in the code

WTs contain additional store instructions to the L1 SPM to share the execution state while PHTs contain additional load instructions

ETH zürich





HERO: Software

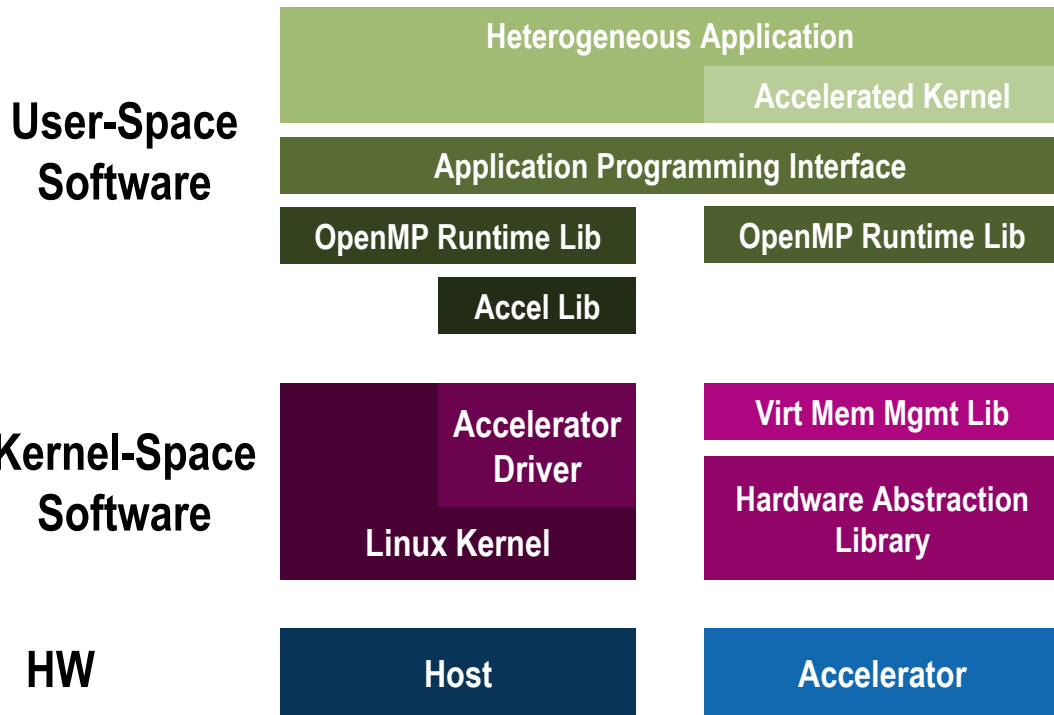
SW stack
Efficient offloading

ETH zürich





HERO: Software Stack



ETH zürich





HERO: OpenMP support

Copy-Based Shared Memory

- **Data is copied** to and from a physically contiguous, uncached section in main memory, and **physical pointers** are passed to the PMCA

Shared Virtual Memory

- It enables zero-copy offloads, directly passing **virtual pointers** to the PMCA



HERO: Programming Model and API

Principle: **single-source** heterogeneous programming. Example:

```
void host_function(unsigned n_elems, const float a, const float* x, float* y) {  
    #pragma omp target map(to: n_elems, a, x, y) device(HERO_DEVICE_PULP)  
    {  
        float buf_x[BUF_ELEMS], buf_y[BUF_ELEMS];  
        for (unsigned offset = 0; offset < n_elems; offset += BUF_ELEMS) {  
            const unsigned cur_n_elems = min(n_elems - offset, BUF_ELEMS);  
            const size_t cur_memcpy_size = cur_n_elems * sizeof(float);  
            hero_memcpy_host2dev(buf_x, x+offset, cur_memcpy_size);  
            hero_memcpy_host2dev(buf_y, y+offset, cur_memcpy_size);  
            hero_dblas_saxpy(cur_n_elems, a, buf_x, buf_y);  
            hero_memcpy_dev2host(y+offset, buf_y, cur_memcpy_size);  
        }  
    }  
}
```



HERO: Heterogeneous Compilation

Single-source, single-binary heterogeneous compilation ...

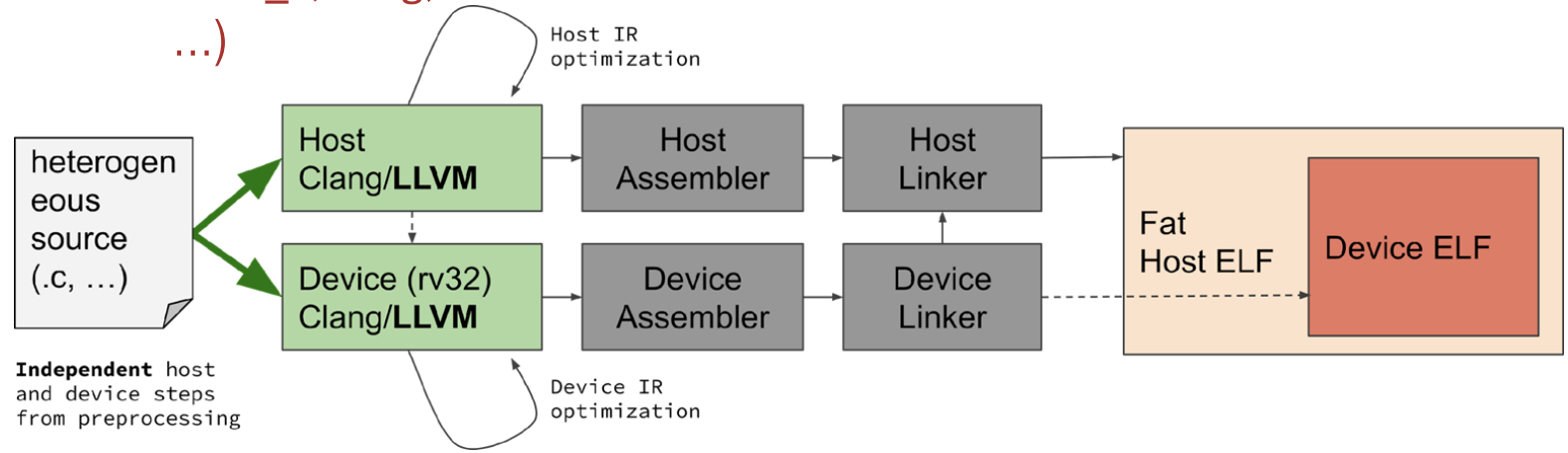
Single Source,
Common API,
Specialized
Code

Separate data models
(width of pointers,
size_t, long,
...)

Separate ISAs

Separate
Libraries

Single Binary,
Nested
Executables



... provides first-class support for heterogeneous programming!





HERO: Implementation and Results

FPGA implementations
Results

ETH zürich





HERO on FPGA

HERO implementations have been deployed on different FPGA platforms:

- Zynq UltraScale+ MPSoC ZCU102 (Xilinx)
- Zynq ZC706 Evaluation Kit (Xilinx)
- Juno development board (Arm)
- Virtex UltraScale+ HBM VCU128 (Xilinx)

Larger programmable logic available on the FPGA enables research on larger PMCAs or multi-cluster PMCAs.



Results: Zero-Copy vs. Copy-Based

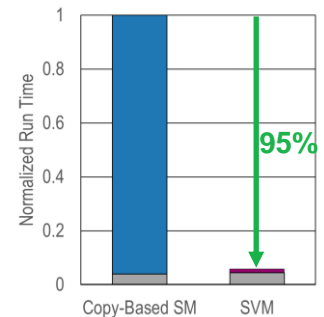
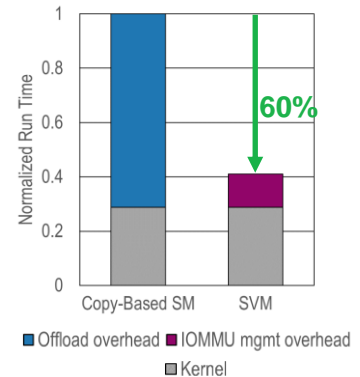
The main motivation for shared virtual memory (SVM) is programmability. However, SVM can also significantly improve performance!

PageRank (algorithm to analyze graph connectivity):

- The overhead of manipulating pointers at offload-time in the **copy-based** approach is higher than the overhead introduced by translating pointer with **SVM**

MemCopy (Copy a large array from DRAM to PMCA and back)

- The host copy phase takes much more time than letting the PMCA access data directly with high-bandwidth DMA transfers





HERO: Ongoing efforts

HOSTs

ARM

ARM

 **Ariane** 

 **Ariane** 



PULP



Snitch



PULP



Snitch



PMCAs

ETH zürich





Conclusion

- **HERO** is a research platform to explore HW/SW codesign of heterogeneous systems
- HERO enables full-system exploration of RISC-V manycore accelerators
- HERO achieves **efficient collaboration** between **host** and **PMCA** through a shared virtual memory enabled by its hybrid IOMMU
- OpenMP plugin allows for **transparent accelerator programming**



Thank you for your attention!



ETH zürich

