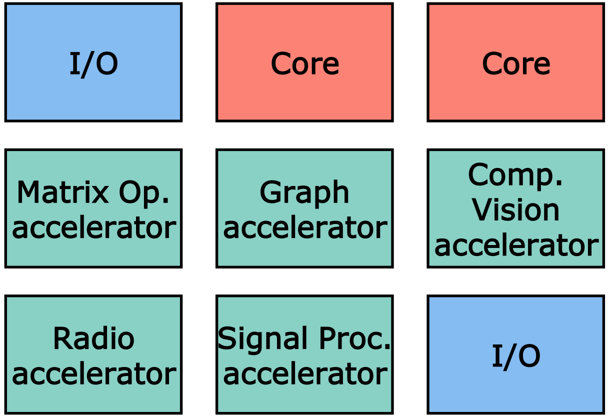
Collaborative System-on-Chip Design with the Open-Source ESP Platform

Luca P. Carloni

ModSim 2024
Seattle, WA
August 2024

Columbia University
in the City of New York

The Age of Heterogeneous Computing



- **State-of-the-art SoC architectures integrate increasingly diverse sets of components**
  - different CPUs, GPUs, hardware accelerators, memory hierarchies, I/O peripherals, sensors, reconfigurable engines, analog blocks…
- **The migration towards heterogeneous SoC architectures will accelerate, across almost all computing domains**
  - IoT devices, mobile devices, embedded systems, automotive electronics, avionics, data centers and even supercomputers
- **The set of heterogeneous SoCs in production in any given year will be itself heterogeneous!**
  - no single SoC architecture will dominate all the markets!

Heterogeneity Increases Design Complexity

- Heterogeneous architectures produce higher energy-efficient performance, but make more difficult the tasks of design, verification and programming
  - at design time, diminished regularity in the system structure, chip layout
  - at runtime, more complex hardware/software and management of shared resources
- With each SoC generation, the addition of new capabilities is increasingly limited by engineering effort and team sizes
  - [Khailany2018]
- The biggest challenges are (and will increasingly be) found in the complexity of system integration

[L. P. Carloni. The Case for Embedded Scalable Platforms, Invited Paper at DAC 2016]

Open-Source Hardware (OSH)

- An opportunity to reenergize the innovation in the semiconductor and electronic design automation industries
- The OSH community is gaining momentum
  - many diverse contributions from both academia and industry
  - multi-institution organizations
  - government programs

Image Sources:
https://riscv.org/
https://github.com/nvdla
https://github.com/lnis-uofu/OpenFPGA
https://pulp-platform.org/
https://vortex.cc.gatech.edu/
https://parallel.princeton.edu/openpiton/
https://fastmachinelearning.org/hls4ml/
https://chipyard.readthedocs.io/en/stable/
https://chipsalliance.org/
https://www.openhwgroup.org/

The Open Challenge of Open-Source Hardware

- To date, most OSH projects are focused on the development of individual SoC components, such as a processor core, a GPU, or an accelerator
- This leaves open a critical challenge:

How can we realize a complete SoC for a given target application domain by efficiently reusing and combining a variety of independently developed, heterogeneous, OSH components, especially if these components are designed by separate organizations for separate purposes?

The Concept of Platform

- Innovation in SoC architectures and their design methodologies is needed to promote design reuse and collaboration
  - Architectures and methodologies must be developed together
- *Platform = architecture + methodology*
  - An SoC architecture enables design reuse when it simplifies the integration of many components that are independently developed
  - An SoC methodology enables design collaboration when it allows designers to choose the preferred specification languages and design flows for the various components
- An effective combination of architecture and methodology is a platform that maximizes the potential of open-source hardware
  - by scaling up the number and type of components that can be integrated in an SoC and by enhancing the productivity of the designers who develop and use them

ESP : An Open-Source Platform for SoC Design

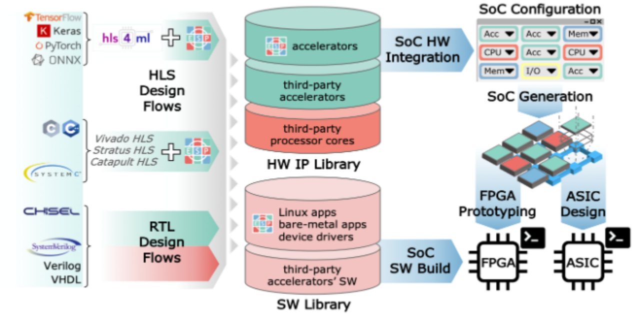
Home Release Resources News Press Team Contact

ESP
the open-source SoC platform

esp.cs.columbia.edu

The ESP Vision

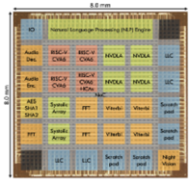
ESP is an open-source research platform for heterogeneous system-on-chip design that combines a scalable tile-based architecture and a flexible system-level design methodology.



ESP provides three accelerator flows: RTL, high-level synthesis (HLS), machine learning frameworks. All three design flows converge to the ESP automated SoC integration flow that generates the necessary hardware and software interfaces to rapidly enable full-system prototyping on FPGA.

Overview

Latest Posts



ESP at ISSCC!

Check out our second chip based on ESP, the open-source SoC platform.

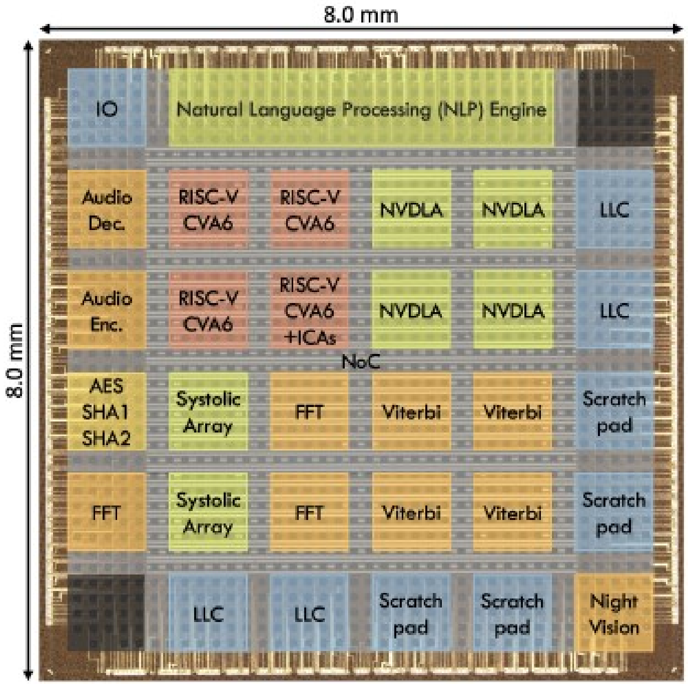
Read more

Published: Mar 16, 2024

Release 2024.1.0

A new GitHub Release

ESP is Silicon Proven: The EPOCHS-1 SOC



Technology	12nm FinFET
Area	$64mm^2$
#IOs	340
Power Domains	23
Clock Domains	35
Power	83mW – 4.33W
Total SRAM	8.4MB
Max Frequency Range	680MHz – 1.6GHz
Example Application Domain	Collaborative Autonomous Vehicles

14.5 A 12nm Linux-SMP-Capable RISC-V SoC with 14 Accelerator Types, Distributed Hardware Power Management and Flexible NoC-Based Data Orchestration

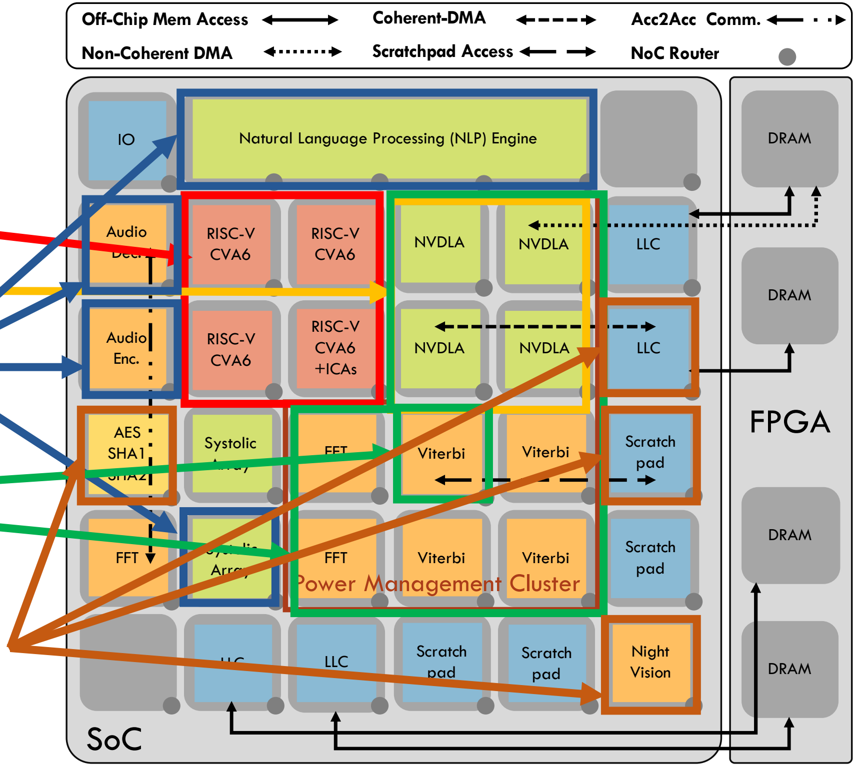
Maico Cassel dos Santos*[1], Tianyu Jia*[2], Joseph Zuckerman*[1], Martin Cochet*[3], Davide Giri[1], Erik Jens Loscalzo[1], Karthik Swaminathan[3], Thierry Tambe[2], Jeff Jun Zhang[2], Alper Buyuktosunoglu[3], Kuan-Lin Chiu[1], Giuseppe Di Guglielmo[1], Paolo Mantovani[1], Luca Piccolboni[1], Gabriele Tombesi[1], David Trilla[3], John-David Wellman[3], En-Yu Yang[2], Aporva Amarnath[3], Ying Jing[4], Bakshree Mishra[4], Joshua Park[2], Vignesh Suresh[4], Sarita Adve[4], Pradip Bose[3], David Brooks[2], Luca P. Carloni[1], Kenneth L. Shepard[1], Gu-Yeon Wei[2]

[1]Columbia University, New York, NY; [2]Harvard University, Cambridge, MA
[3]IBM Research, Yorktown Heights, NY; [4]University of Illinois, Urbana, IL
*Equally Credited Authors

ISSCC 2024 / SESSION 14 / DIGITAL TECHNIQUES FOR SYSTEM ADAPTATION, POWER MANAGEMENT AND CLOCKING / 14.5

The EPOCHS-1 SoC: Sources of OSH IPs

- Sources of Open-Source Hardware IPs:
  - 4 RISC-V CVA6 cores from ETH Zurich/OpenHW Group
  - 4 NVIDIA Deep Learning Accelerators
  - 4 Accelerators designed at Harvard
  - 1 Accelerator and Power Management designed at IBM Research
  - 3 Accelerators, Memory Hierarchy, and Network-on-Chip designed at Columbia

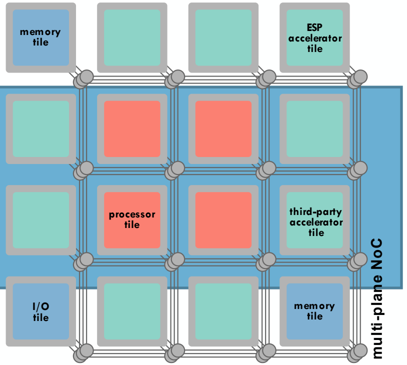


ESP

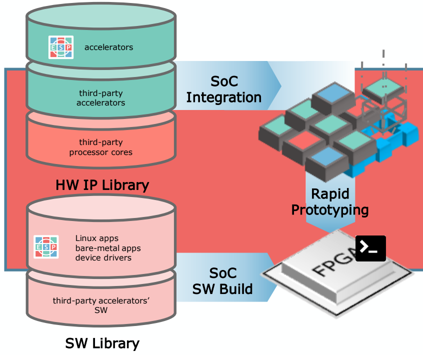
Columbia University in the City of New York

Outline

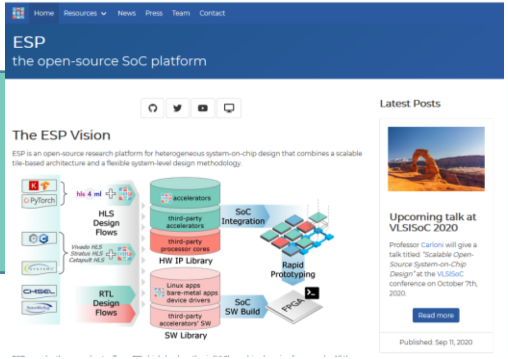
The ESP Architecture



The ESP Methodology



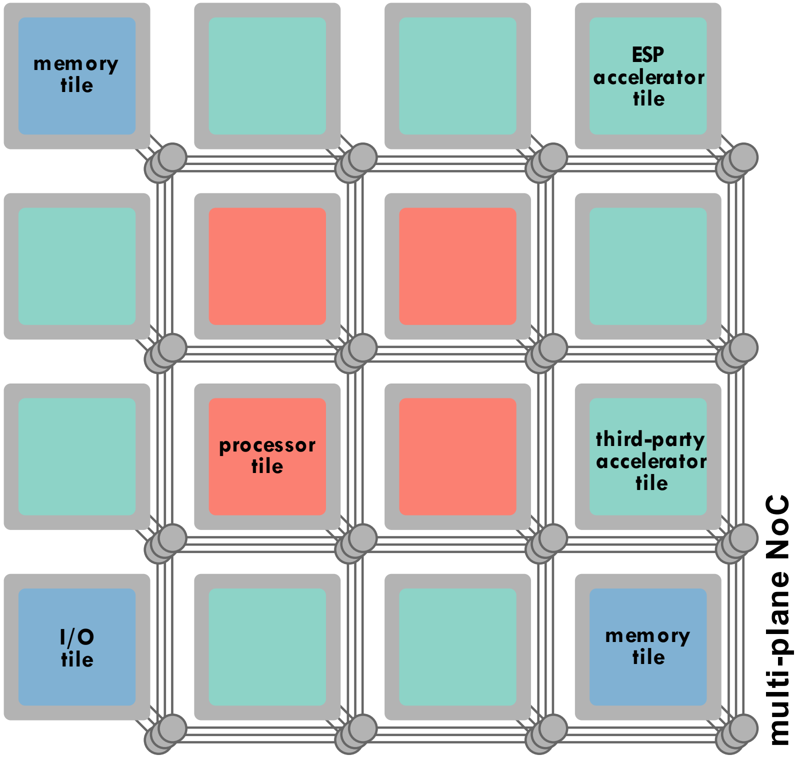
Scalable Collaborative SoC Design



ESP Architecture

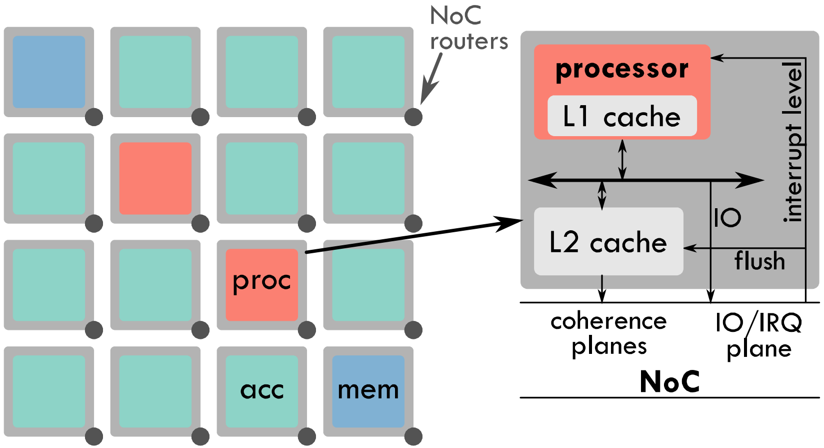
- RISC-V Processors
- Many-Accelerator
- Distributed Memory
- Multi-Plane NoC

The ESP architecture implements a **distributed** system, which is **scalable**, **modular** and **heterogeneous**, giving processors and accelerators similar weight in the SoC



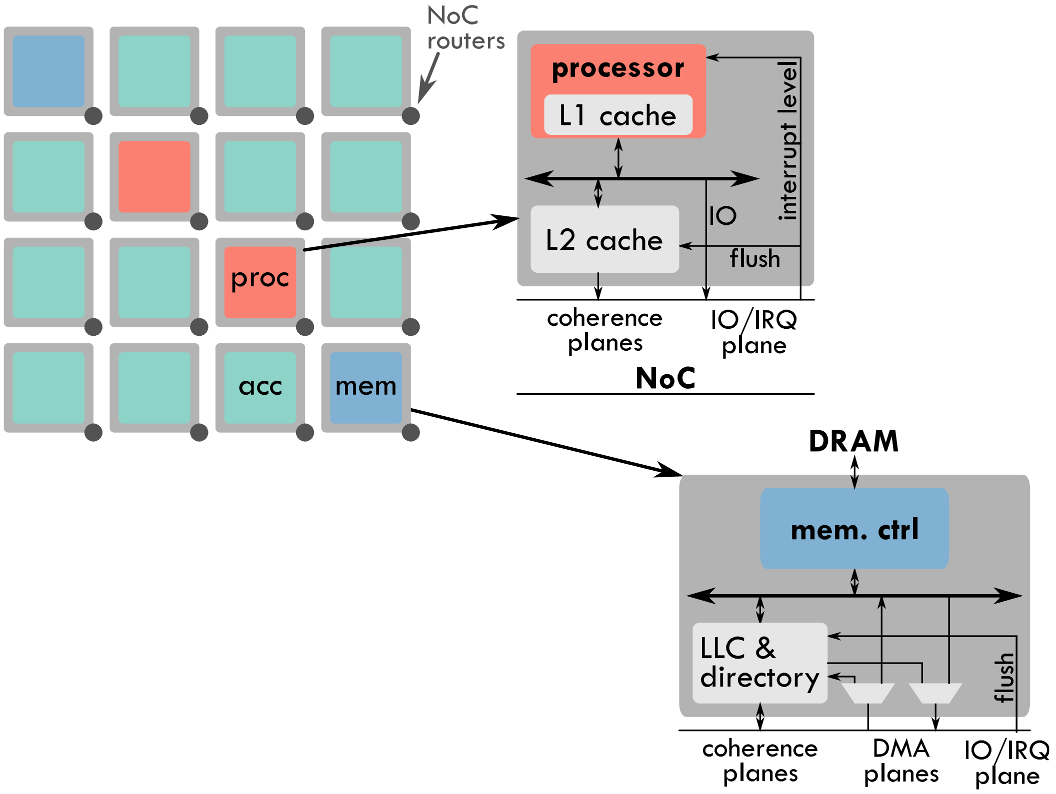
ESP Architecture: Processor Tile

- Processor off-the-shelf
  - RISC-V CVA6-Ariane (64 bit) SPARC V8 Leon3 (32 bit)
  - RISC-V IBEX (32 bit)
  - L1 private cache
- L2 private cache
  - Configurable size
  - MESI protocol
- IO/IRQ channel
  - Un-cached
  - Accelerator config. registers, interrupts, flush, UART, …



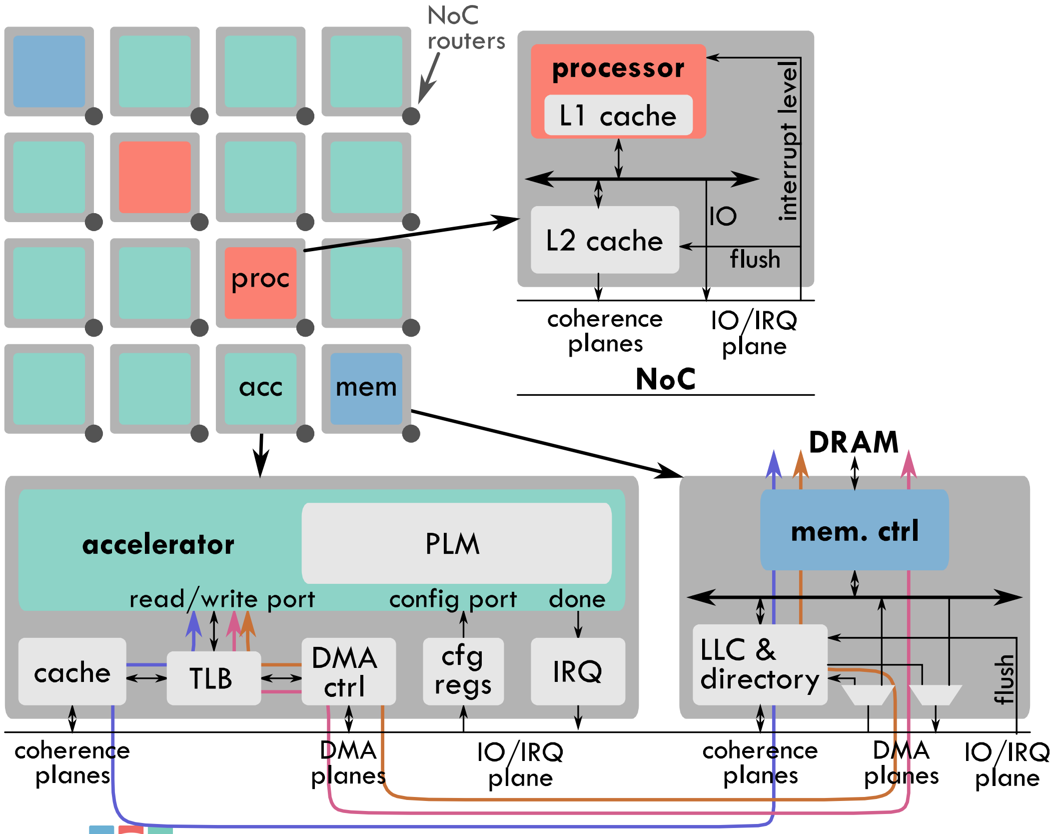
ESP Architecture: Memory Tile

- **External Memory Channel**
- LLC and directory partition
  - Configurable size
  - Extended MESI protocol
  - Supports coherent-DMA for accelerators
- DMA channels
- IO/IRQ channel

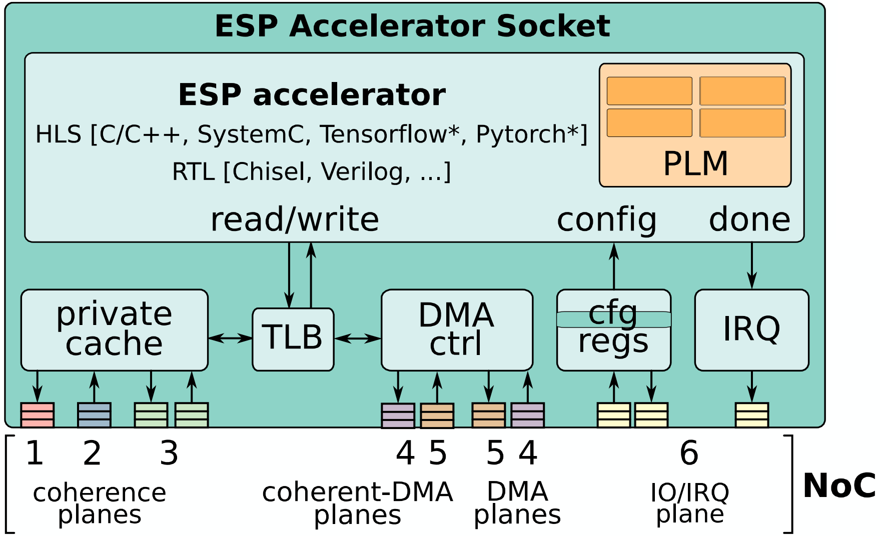


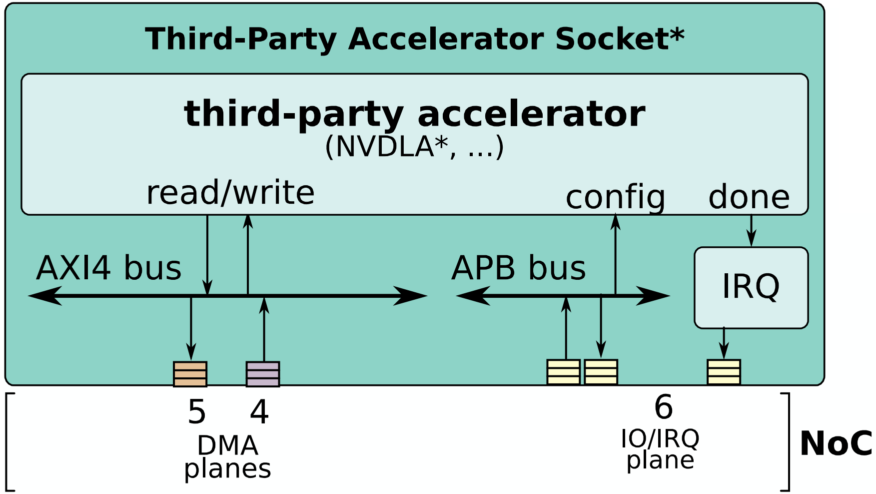
ESP Architecture: Accelerator Tile

- Accelerator Socket w/ Platform Services
  - Direct-memory-access
  - Run-time selection of coherence model:
    - Fully coherent
    - LLC coherent
    - Non coherent
  - User-defined registers
  - Distributed interrupt



ESP Accelerator Socket

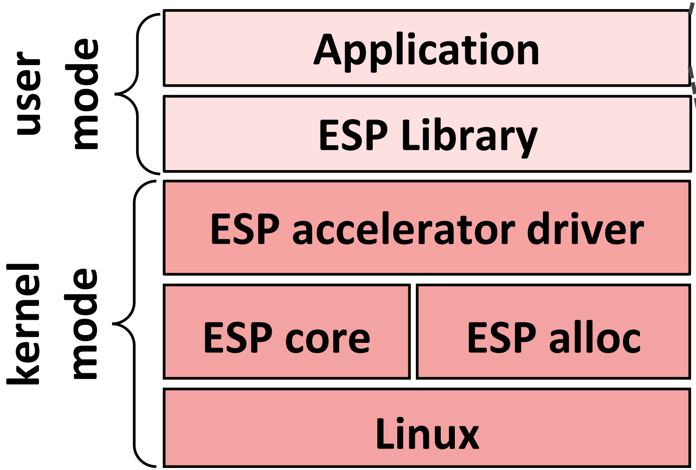




Columbia University in the City of New York

ESP Software Socket

- **ESP accelerator API**
  - Generation of device driver and unit-test application
  - Seamless shared memory




```
/*
 * Example of existing C application with ESP
 * accelerators that replace software kernels 2, 3,
 * and 5. The cfg_k# contains buffer and the
 * accelerator configuration.
 */
{
  int *buffer = esp_alloc(size);

  for (...) {

    kernel_1(buffer,...); /* existing software  */

    esp_run(cfg_k2);      /* run accelerator(s) */
    esp_run(cfg_k3);

    kernel_4(buffer,...); /* existing software  */

    esp_run(cfg_k5);
  }

  validate(buffer);       /* existing checks    */

  esp_free();           /* memory free        */
}
```

ESP Platform Services

Accelerator tile
- DMA
- Reconfigurable coherence
- Point-to-point
- ESP or AXI interface
- DVFS controller

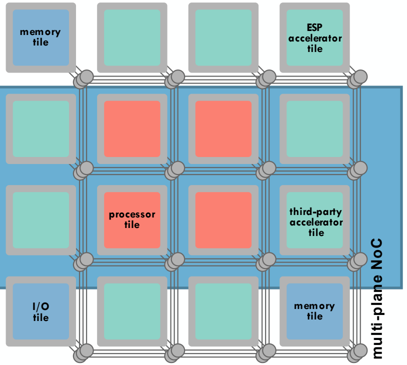
Processor Tile
- Coherence
- I/O and un-cached memory
- Distributed interrupts
- DVFS controller

Miscellaneous Tile
- Debug interface
- Performance counters access
- Coherent DMA
- Shared peripherals (UART, ETH, …)

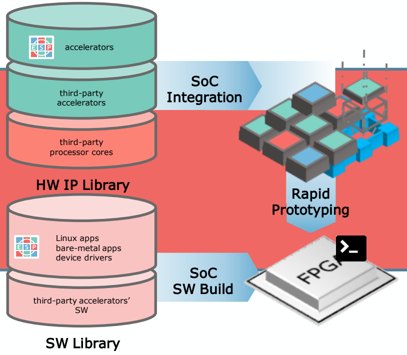
Memory Tile
- Independent DDR Channel
- LLC Slice
- DMA Handler

Outline

The ESP Architecture

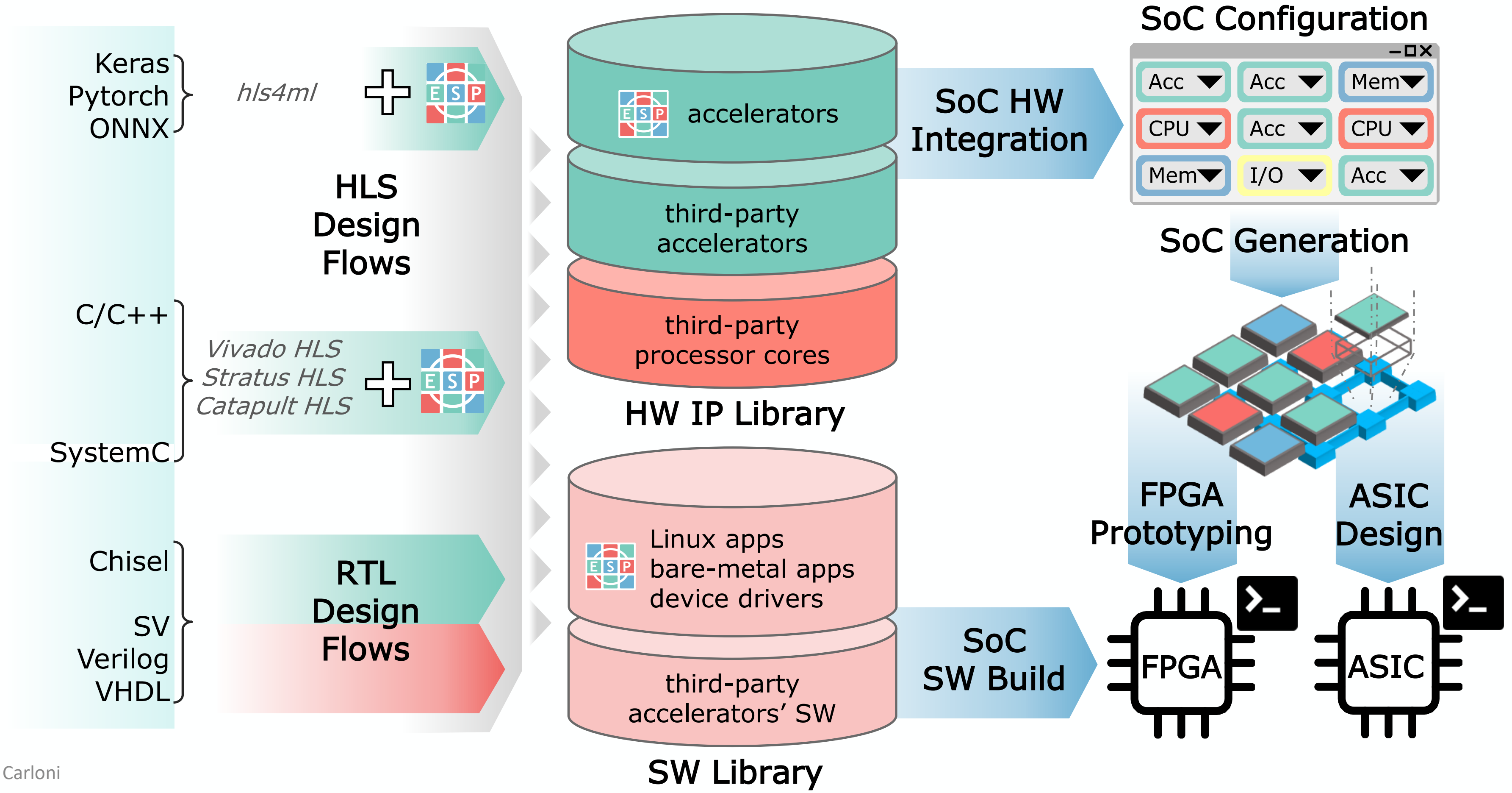


The ESP Methodology



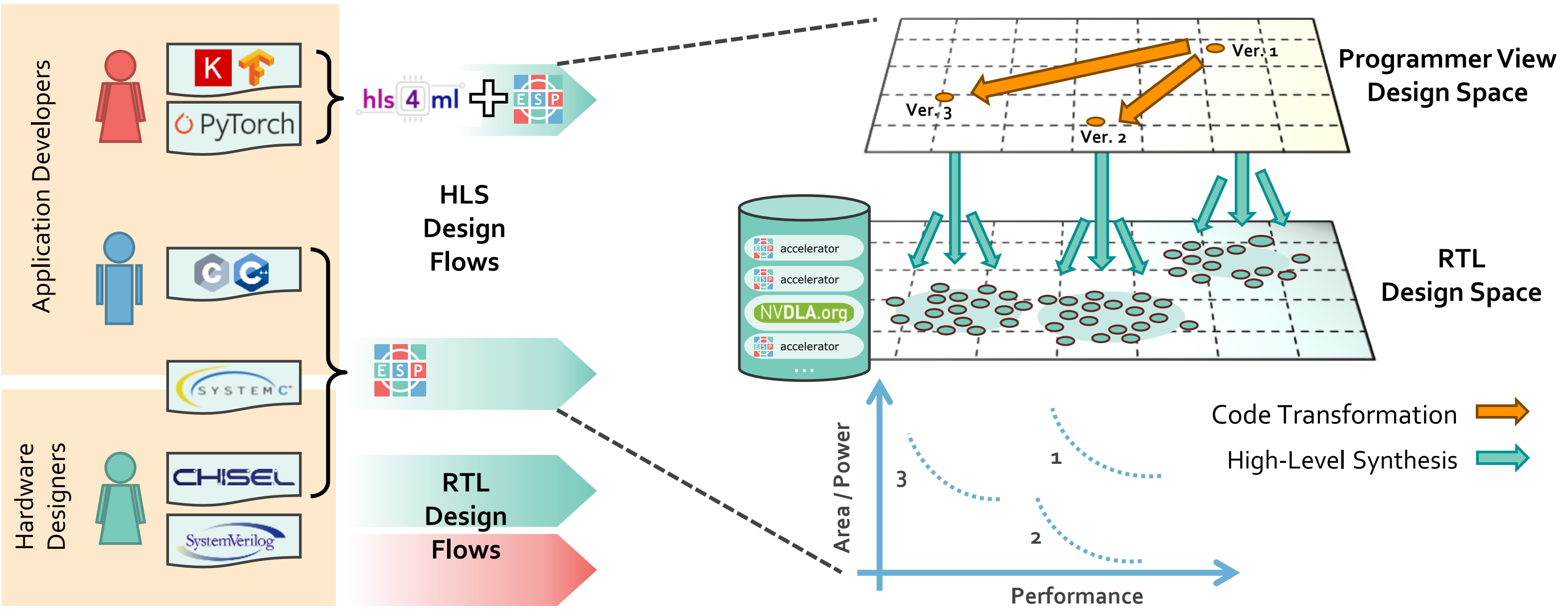
Scalable Collaborative SoC Design

The ESP Vision: Domain Experts Can Design SoCs

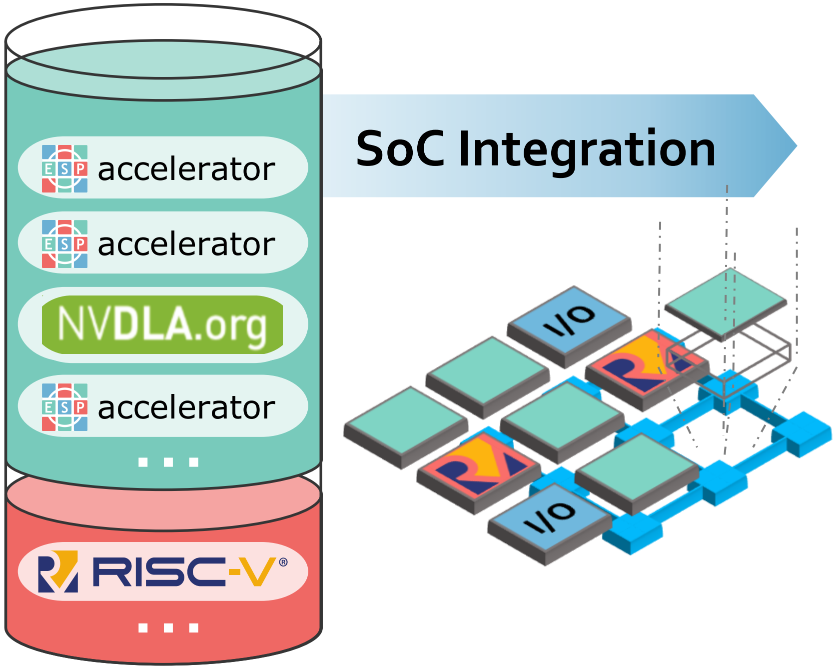


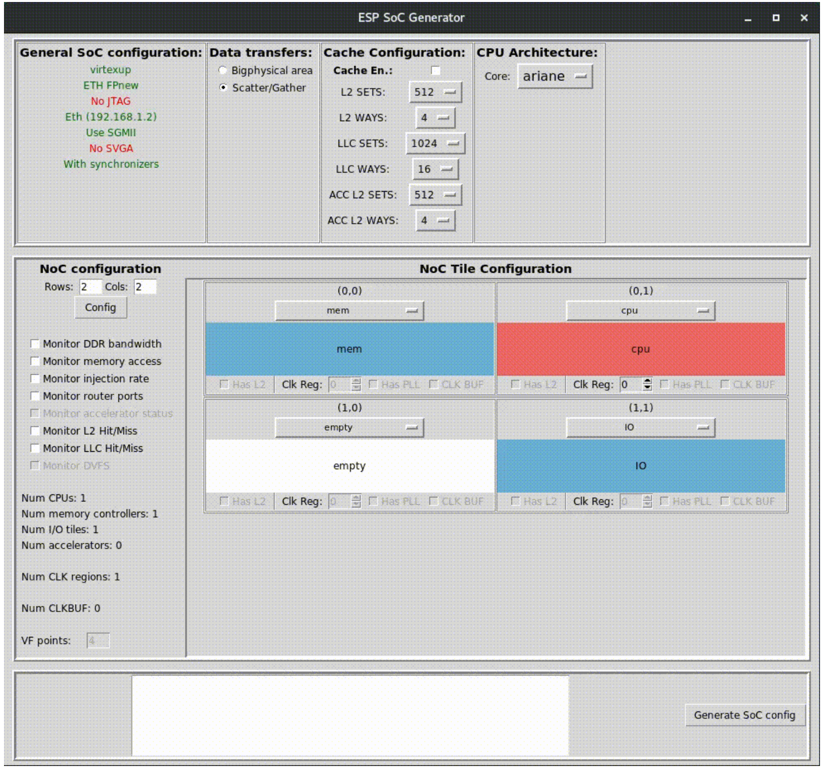
ESP Accelerator Flow

Developers focus on the **high-level specification**, **decoupled** from memory access, system communication, hardware/software interface

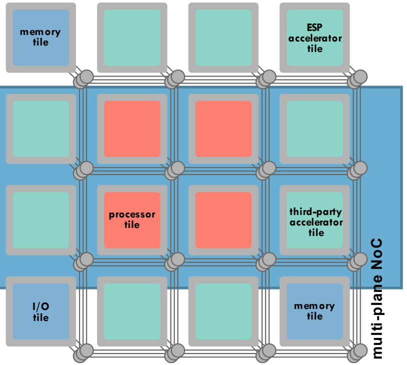


ESP Interactive Flow for SoC Integration

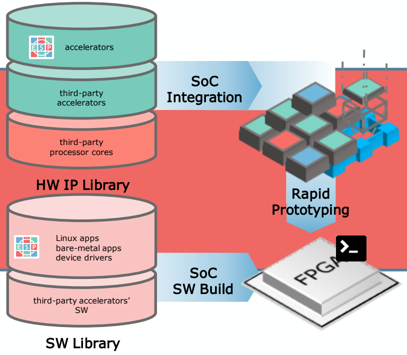




Outline

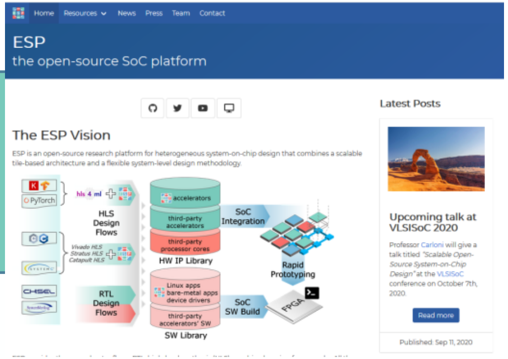


The ESP Architecture



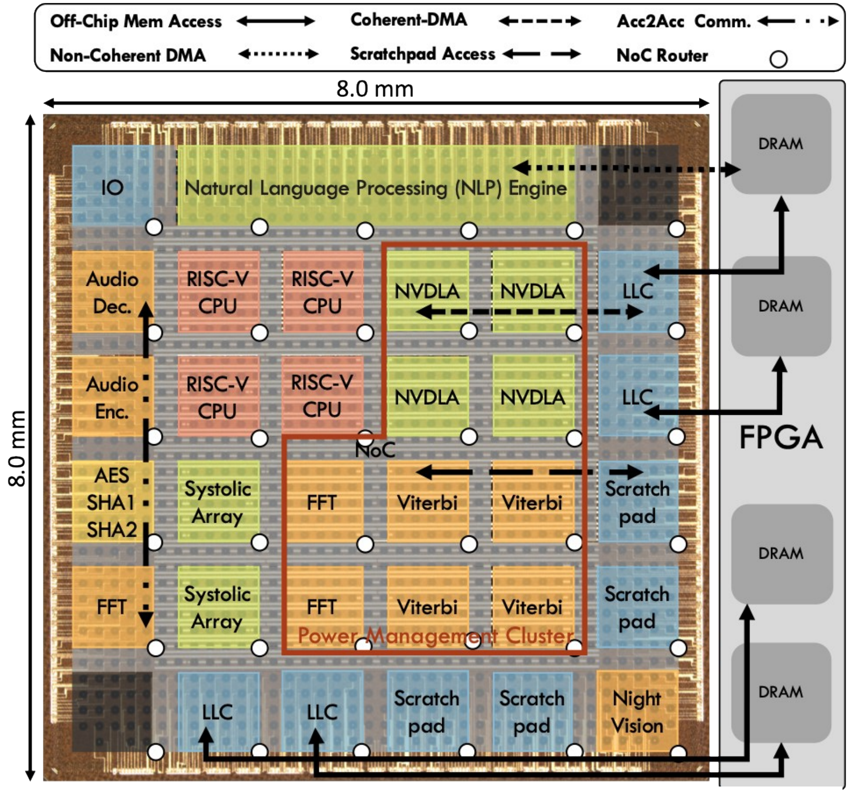
The ESP Methodology

Scalable Collaborative SoC Design



The EPOCHS-1 SoC: Chip Highlights

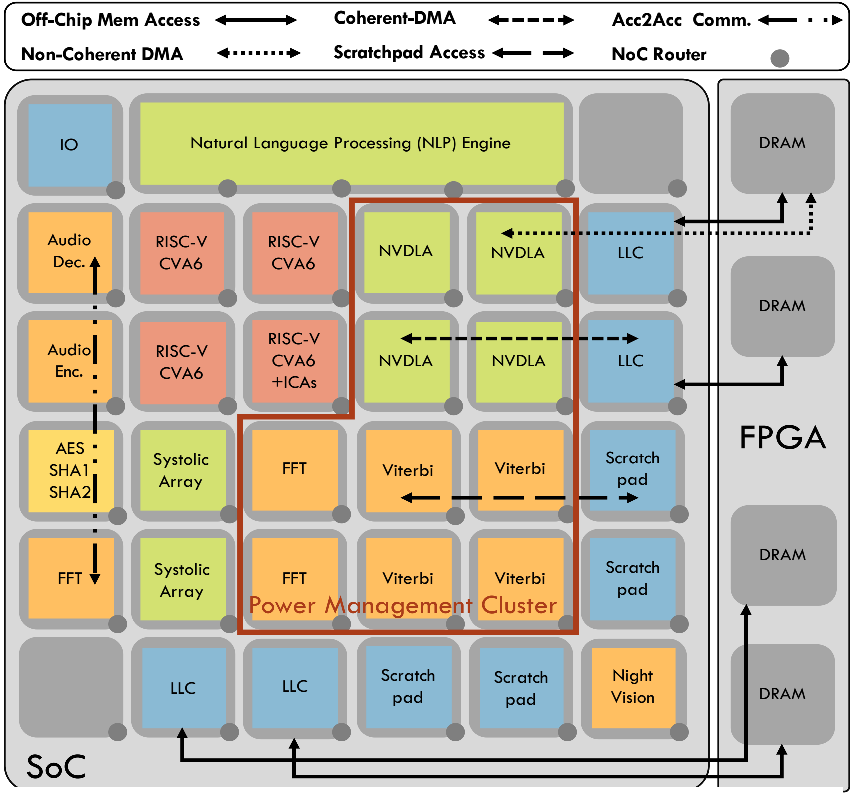
- 64 mm^2 SoC designed in 12 nm FinFET
- 35 clock domains; 23 power domains
- 8.4 MB on-chip SRAM memory
- Tile-based SoC architecture



[M. Cassel et al., A 12nm Linux-SMP-Capable RISC-V SoC with 14 Accelerator Types, Distributed Hardware Power Management and Flexible NoC-Based Data Orchestration, ISSCC 2024]

The EPOCHS-1 SoC: Chip Highlights

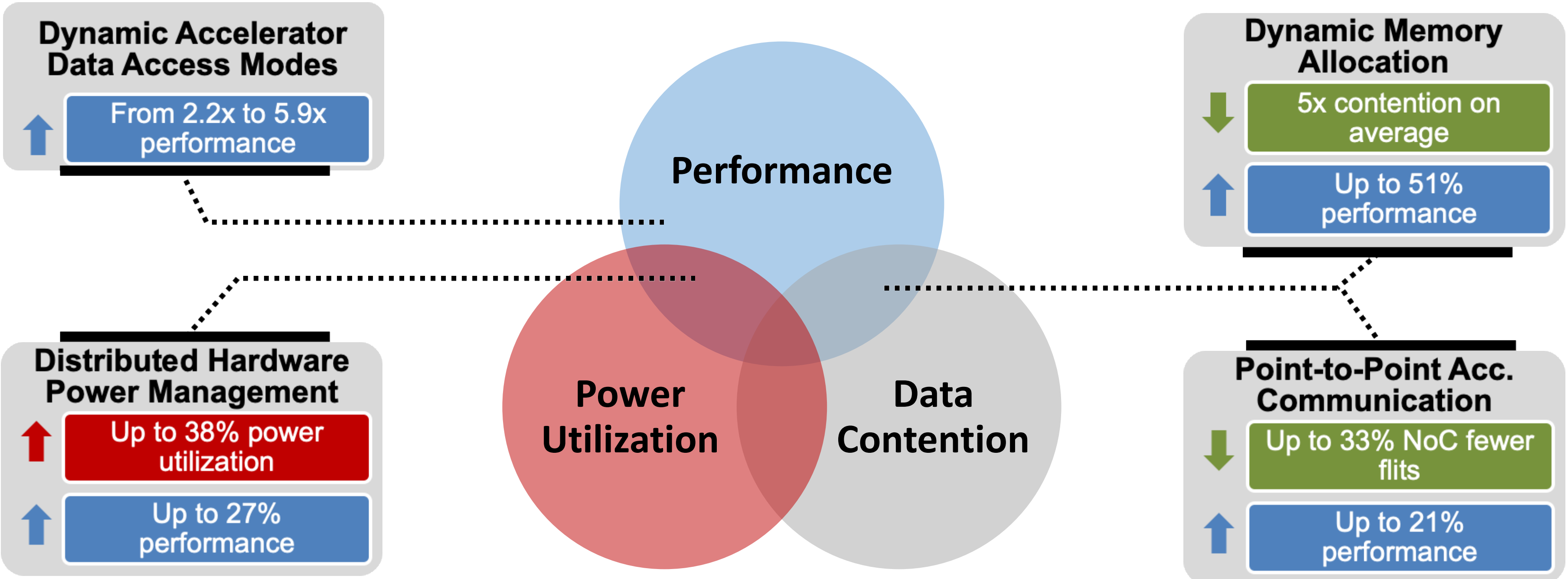
- 64 mm^2 SoC designed in 12 nm FinFET
- 35 clock domains; 23 power domains
- 8.4 MB on-chip SRAM memory
- Tile-based SoC architecture
- 34 tiles connected by a 6-plane 2-D mesh NoC
- The 74 Tbps NoC provides flexible orchestration of data
- 23 accelerators of 14 different types
- 10 accelerators compose a cluster demonstrating a novel distributed hardware power management scheme
- Designed by a small team of PhD students, postdocs, and industry researchers in 3 months with ESP, our open-source platform for agile SoC design



[M. Cassel et al., A 12nm Linux-SMP-Capable RISC-V SoC with 14 Accelerator Types, Distributed Hardware Power Management and Flexible NoC-Based Data Orchestration, ISSCC 2024]

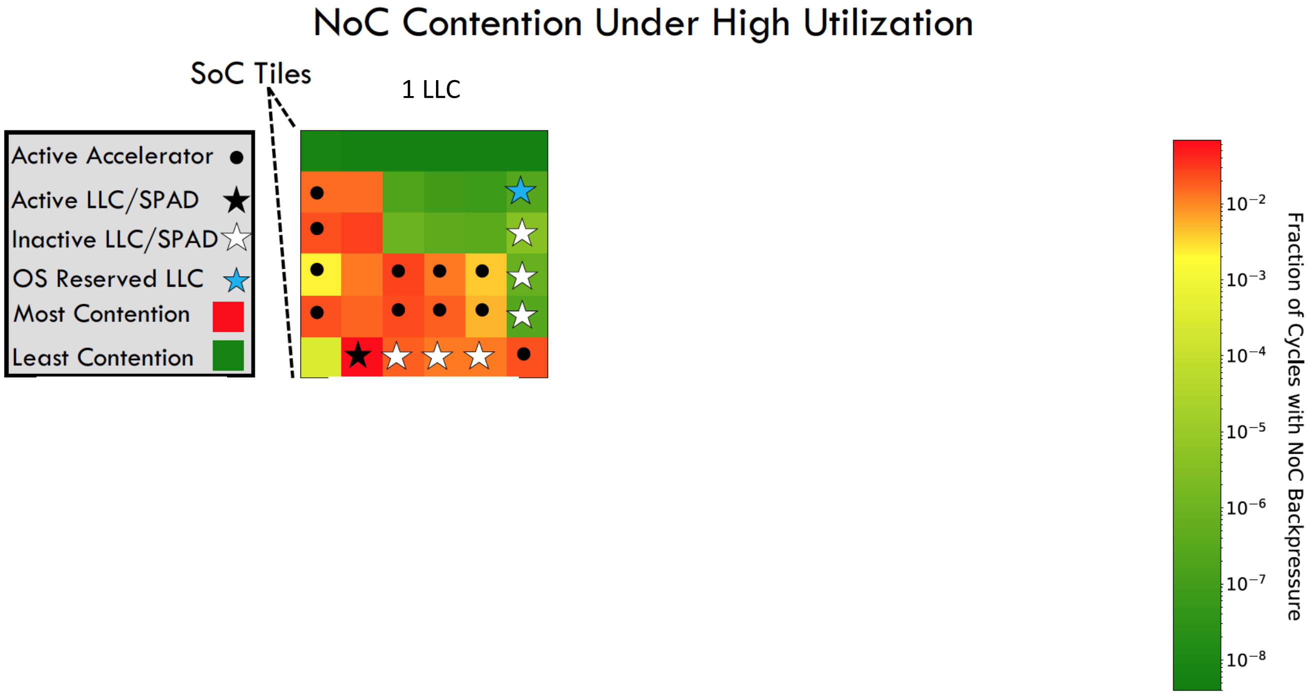
EPOCHS-1 Chip: Summary

- Managing resources in a large, heterogeneous SoC that runs multiple simultaneous applications is a difficult system-level challenge

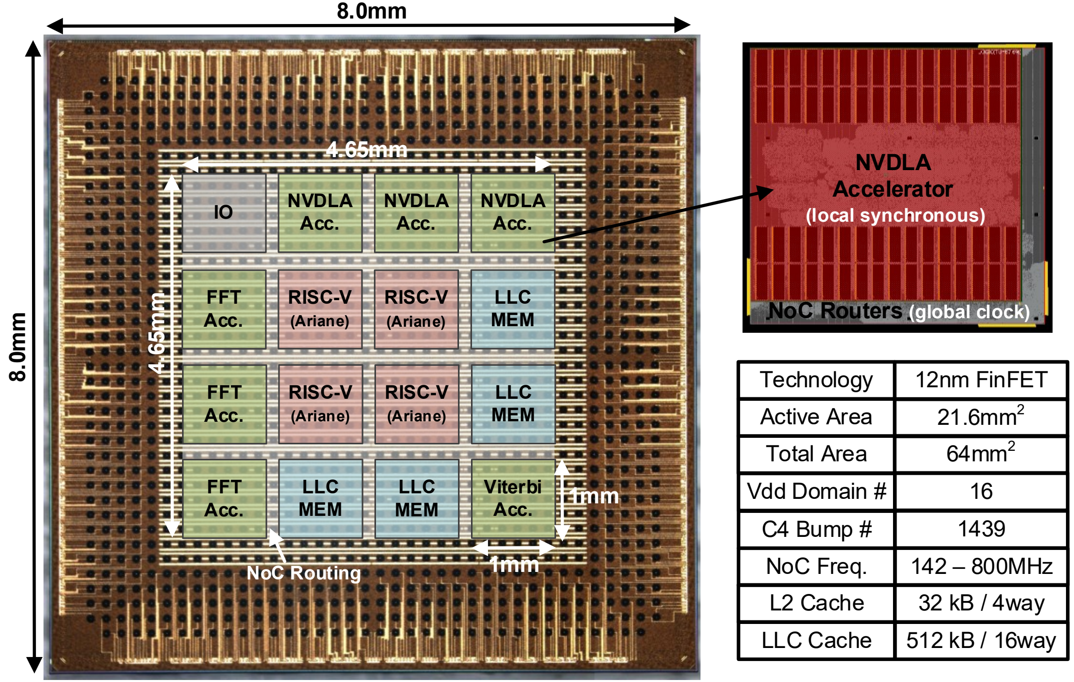


EPOCHS-1 SoC: NoC-Based Data Orchestration

- NoC traffic with 11 accelerators executing in parallel
  - "Contention" = # of cycles when a queue is full and asserts backpressure
- 7 different configurations of the memory hierarchy
- Scaling up the memory hierarchy alleviates contention and distributes traffic

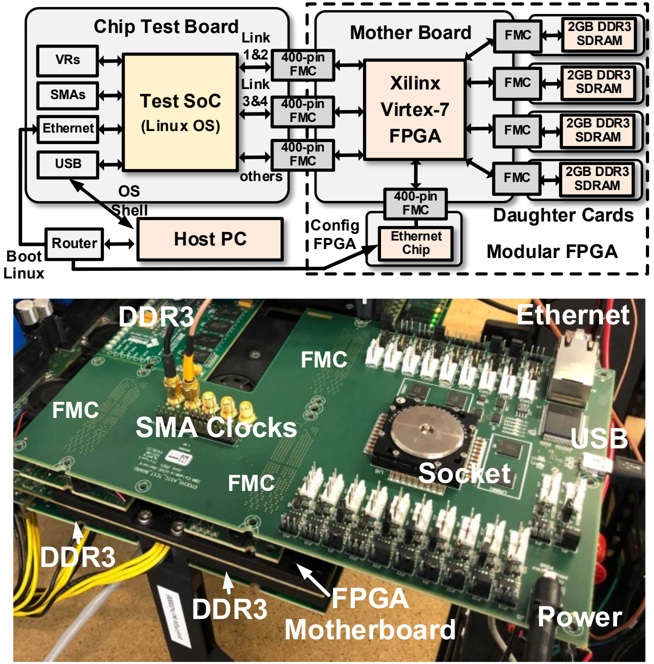


The EPOCHS-o Chip



Technology	12nm FinFET
Active Area	21.6mm^2
Total Area	64mm^2
Vdd Domain #	16
C4 Bump #	1439
NoC Freq.	142 – 800MHz
L2 Cache	32 kB / 4way
LLC Cache	512 kB / 16way

12nm FinFET test chip

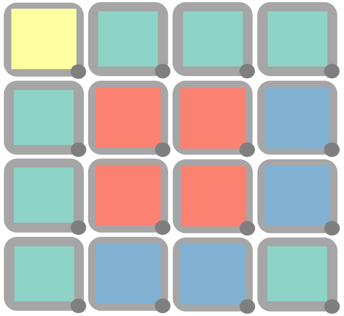


Test Setup

[T. Jia, et al. "A 12nm Agile-Designed SoC for Swarm-Based Perception with Heterogeneous IP Blocks, a Reconfigurable Memory Hierarchy, and an 800MHz Multi-Plane NoC, ESSCIRC 2022]

A Scalable Approach to Chip Design

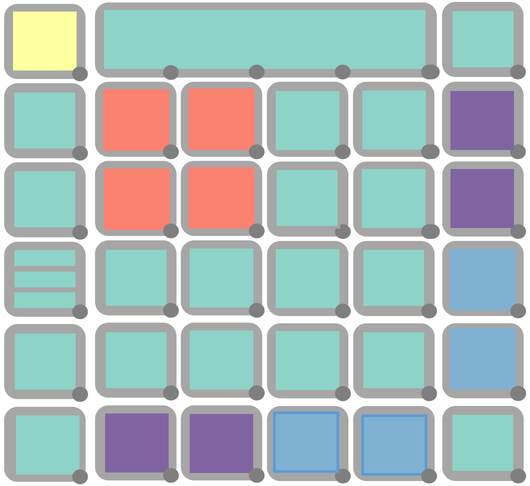
EPOCHS-0



- 4x4 tiles
- 21.62 mm^2
- 17 clock domains
- 16 power domains
- Tile: 12 hours in 16-core 64GB RAM machine
- Top: 51 hours in 64-core 376 GB RAM machine

7 new accelerators tiles
2.25x more tiles
2.18x more clock domains
2.25x more power domains
2.96x more area
Same tile imp. running time
+29% top imp. running time

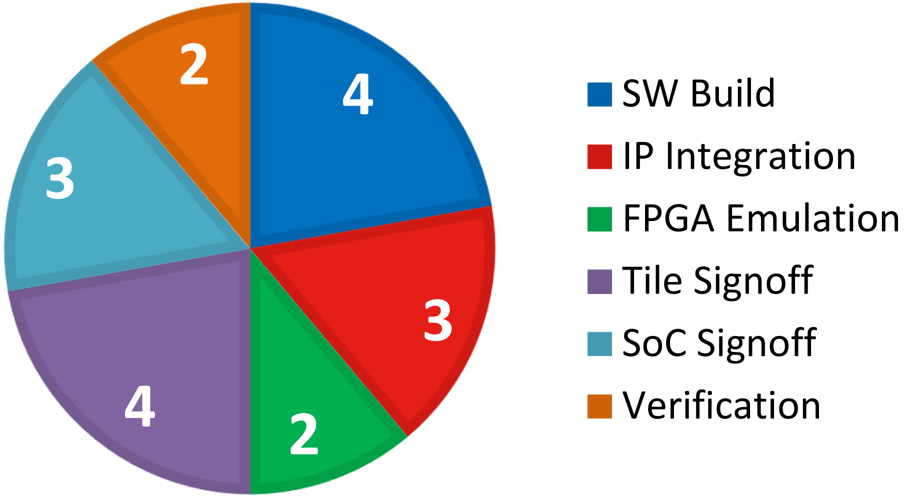
EPOCHS-1



- 6x6 tiles
- 64 mm^2
- 37 clock domains
- 23 power domains
- Tile: 12 hours in 16-core 64GB RAM machine
- Top: 66 hours in 64-core 376 GB RAM machine

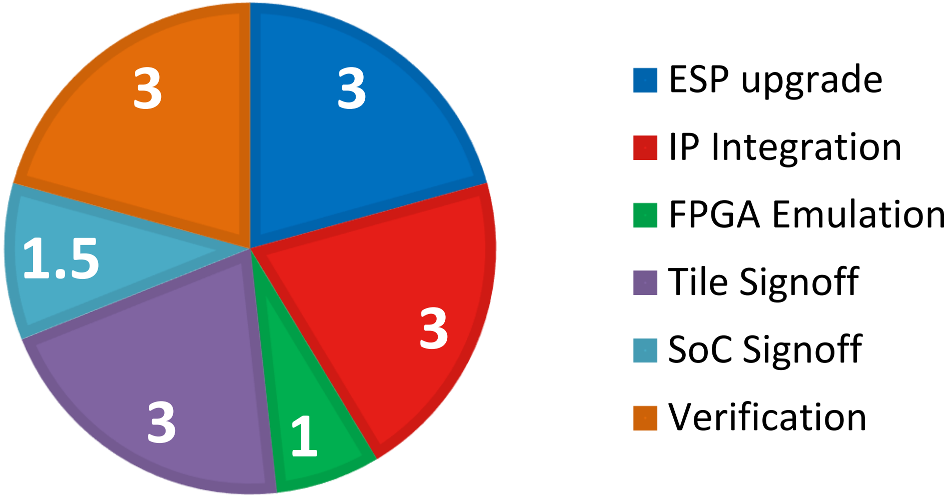
A Scalable Approach to Chip Design

EPOCHS-0 DESIGN CYCLE (WEEKS)



~ 4 months

EPOCHS-1 DESIGN CYCLE (WEEKS)

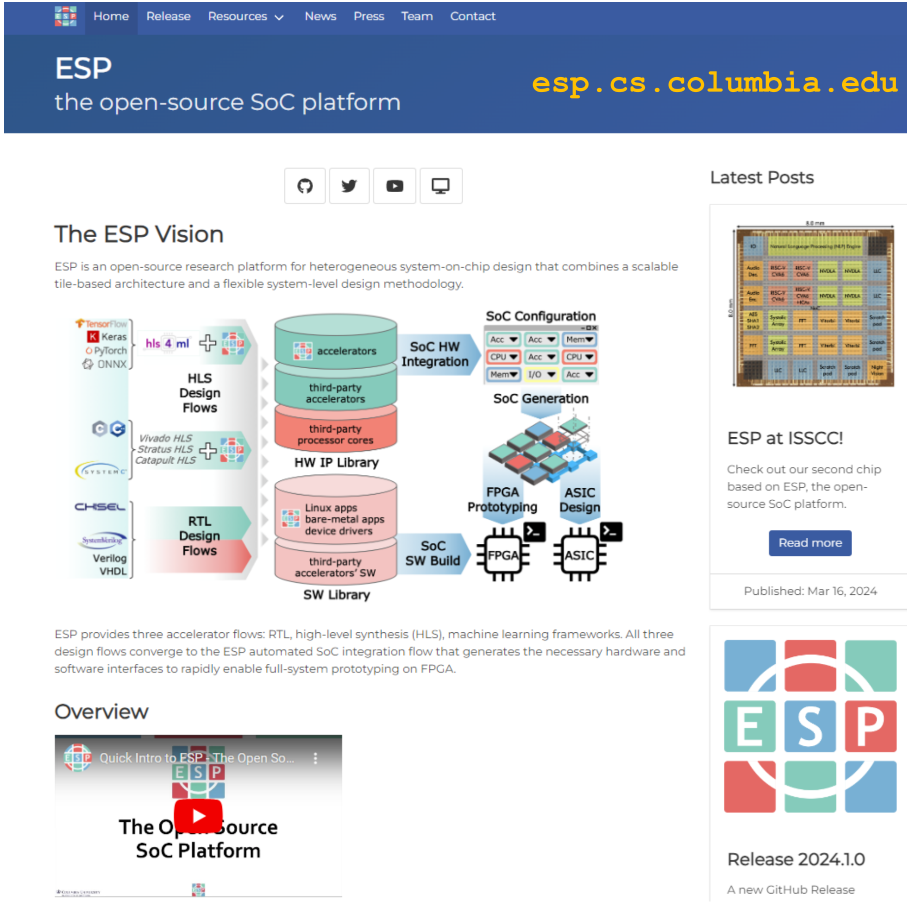


~ 3 months

COLUMBIA UNIVERSITY IN THE CITY OF NEW YORK

In Summary: ESP for Open-Source Hardware

- We contribute ESP to the OSH community in order to support the realization of
  - more scalable architectures for SoCs that integrate
  - more heterogeneous components, thanks to a
  - more flexible design methodology, which accommodates different specification languages and design flows
- ESP was conceived as a heterogeneous integration platform from the start and tested through years of teaching at Columbia University
- We invite you to use ESP for your projects and to contribute to ESP!

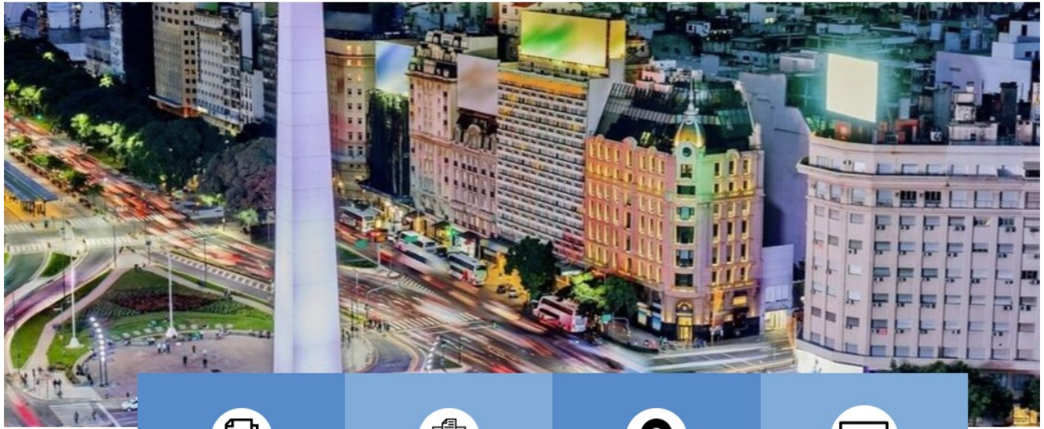


COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

The Third OSCAR Workshop

Open-Source Computer Architecture Research (OSCAR)

June 29, 2024 or Sunday, June 30, 2024 – Buenos Aires, Argentina (co-located with ISCA 2024)



Call for Abstracts | Program | Venue | Archive

Welcome to OSCAR 2024!

OSCAR 2024 is the third edition of a new workshop on open-source hardware which addresses the wide variety of challenges encountered by both hardware and software engineers in dealing with the increasing heterogeneity of next-generation computer architectures. By providing a venue which brings together researchers from academia, industry and government labs, OSCAR promotes a collaborative approach to foster the efforts of the open-source hardware community in this direction.

https://oscar-workshop.github.io/

Some Relevant Publications

1. M. Cassel dos Santos et al. A 12nm Linux-SMP-Capable RISC-V SoC with 14 Accelerator Types, Distributed Hardware Power Management and Flexible NoC-Based Data Orchestration. ISSCC 2024.
2. M. Cassel dos Santos et al. A Scalable Methodology for Agile Chip Development with Open-Source Hardware Components. ICCAD 2022 (Invited Paper).
3. T. Jia et al. A 12nm Agile-Designed SoC for Swarm-Based Perception with Heterogeneous IP Blocks, a Reconfigurable Memory Hierarchy and an 800MHz Multi-Plane NoC. ESSCIRC 2022.
4. J. Zuckerman et al. Cohmeleon: Learning-Based Orchestration of Accelerator Coherence in Heterogeneous SoCs IEEE/ACM International Symposium on Microarchitecture (MICRO-54), 2021.
5. D. Giri et al. Accelerator Integration for Open-Source SoC Design. IEEE MICRO, 2021
6. P. Mantovani et al. Agile SoC Development with Open ESP. ICCAD 2020 (Invited Paper).
7. L. P. Carloni et al. Teaching Heterogeneous Computing with System-Level Design Methods, WCAE 2019.
8. D. Giri et al. Accelerators & Coherence: An SoC Perspective. IEEE MICRO, 2018.
9. L. P. Carloni. The Case for Embedded Scalable Platforms DAC 2016. (Invited Paper).
10. C. Pilato et al. System-Level Optimization of Accelerator Local Memory for Heterogeneous Systems-on-Chip. IEEE Trans. on CAD of Integrated Circuits and Systems, 2017.
11. P. Mantovani et al. An FPGA-Based Infrastructure for Fine-Grained DVFS Analysis in High-Performance Embedded Systems. DAC 2016.
12. L. P. Carloni. From Latency-Insensitive Design to Communication-Based System-Level Design. The Proceedings of the IEEE, November 2015.

Thank you from the ESP team!

esp.cs.columbia.edu

github.com/sld-columbia/esp

System Level Design Group

CS@CU COMPUTER SCIENCE

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK