A Runtime Manager Integrated Emulation Environment for Heterogeneous SoC Design with RISC-V Cores

H. Umut Suluhan[1], Serhan Gener[1], Alexander Fusco[1], Joshua Mack[1], Ismet Dagli[2], Mehmet Belviranli[2], Cagatay Edemen[3], and Ali Akoglu[1]

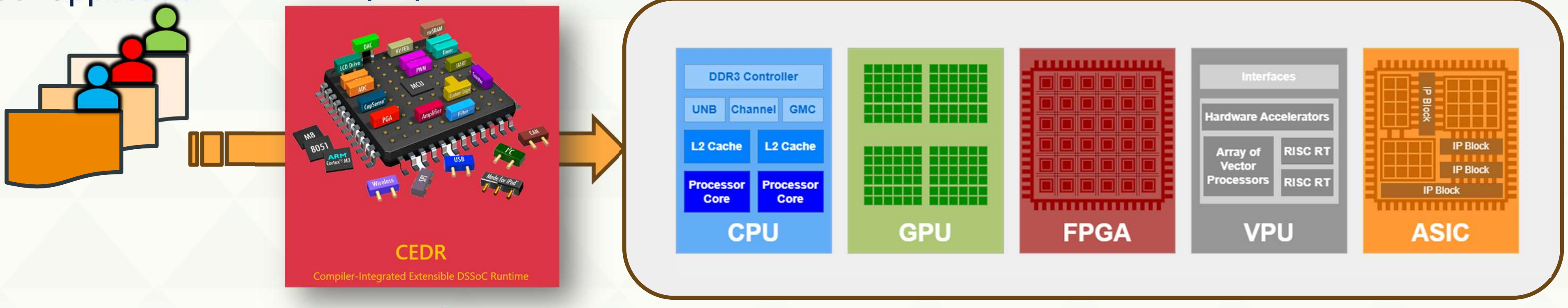
[1]Department of Electrical and Computer Engineering, University of Arizona
{**suluhan**, gener, jmack2545, afusco1, akoglu}@arizona.edu
[2]Computer Science Department, Colorado School of Mines, Golden, CO, USA
{ismetdagli,belviranli}@mines.edu
[3]Electrical and Electronics Engineering Department, Ozyegin University, Istanbul, Turkey
cagatay.edemen@ozyegin.edu.tr

COLLEGE OF ENGINEERING
Electrical & Computer Engineering

Reconfigurable Computing Lab

Motivation

User application | Deployment | Heterogeneity from SoC to HPC Scale



- As systems become more heterogenous, stress on runtime system increases
  - Task scheduling and resource management
    - Runtime overhead and scalability
- RISC-V offers ISA and datapath customization
  - Enables balancing performance and energy efficiency
    - Facilitates workloads that need hardware tailored for specific performance goals
  - **Can also be utilized for resource management with light-weight RISC-V cores**

COLLEGE OF ENGINEERING
Electrical & Computer Engineering

*Mack et al. , "CEDR-API: Productive, Performant Programming of Domain-Specific Embedded Systems," HCW'23 DOI:10.1109/IPDPSW59300.2023.00016
*Kumbhare et al., "User-Space Emulation Framework for Domain-Specific SoC Design," HCW'20 http://dx.doi.org/10.1109/IPDPSW50202.2020.00016

Reconfigurable Computing Lab

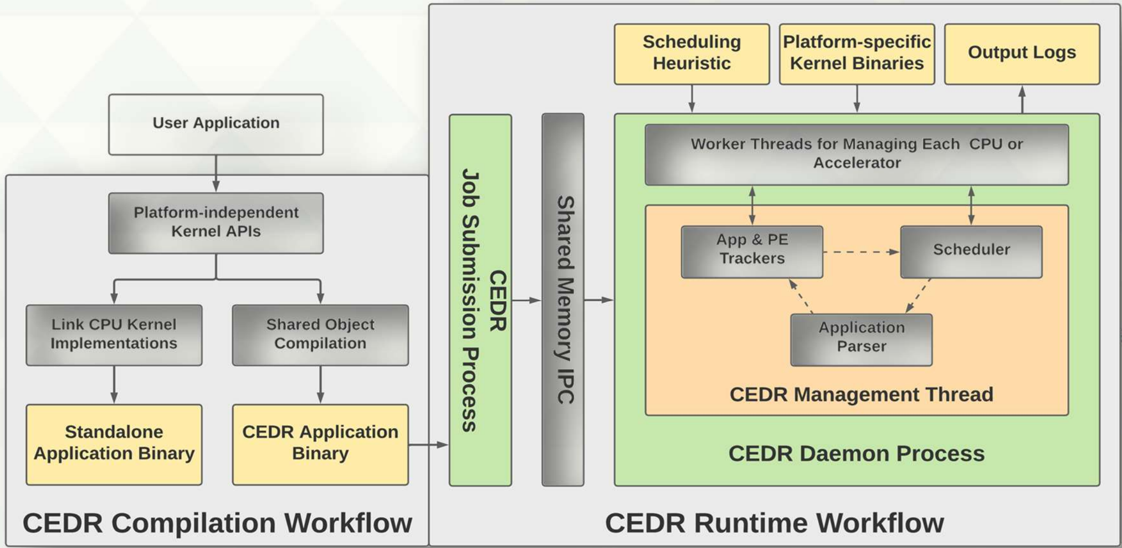
Contributions

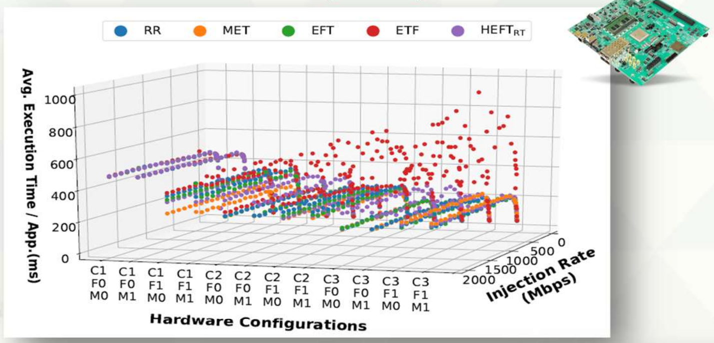
- Leveraged customizable RISC-V cores to realize a scalable runtime system
- Demonstrated ability to deploy dynamic workloads on a RISC-V integrated heterogenous system using CEDR *

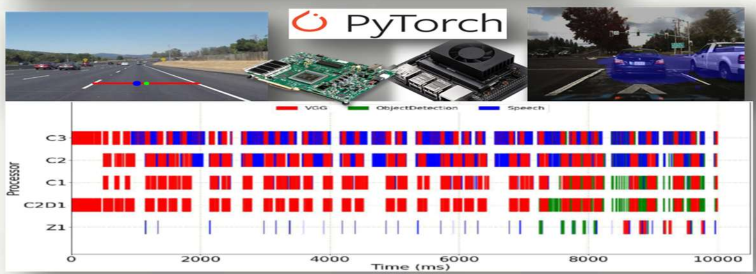
Link to the open source CEDR ecosystem

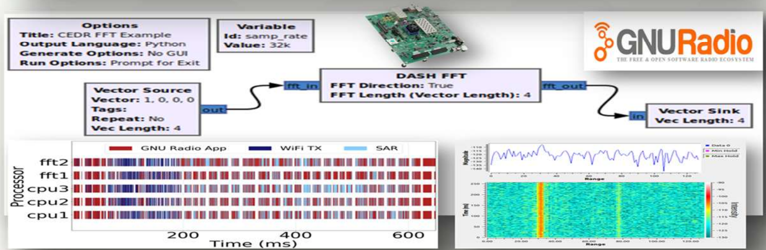
*J. Mack, S. Hassan, N. Kumbhare, M. Castro Gonzalez, and A. Akoglu, “CEDR: A compiler-integrated, extensible DSSoC runtime,” *ACM Trans. Embed. Comput. Syst.*, vol. 22, no. 2, Jan. 2023, issn: 1539-9087. doi: 10.1145/3529257

Hardware Agnostic Application Development and Deployment*







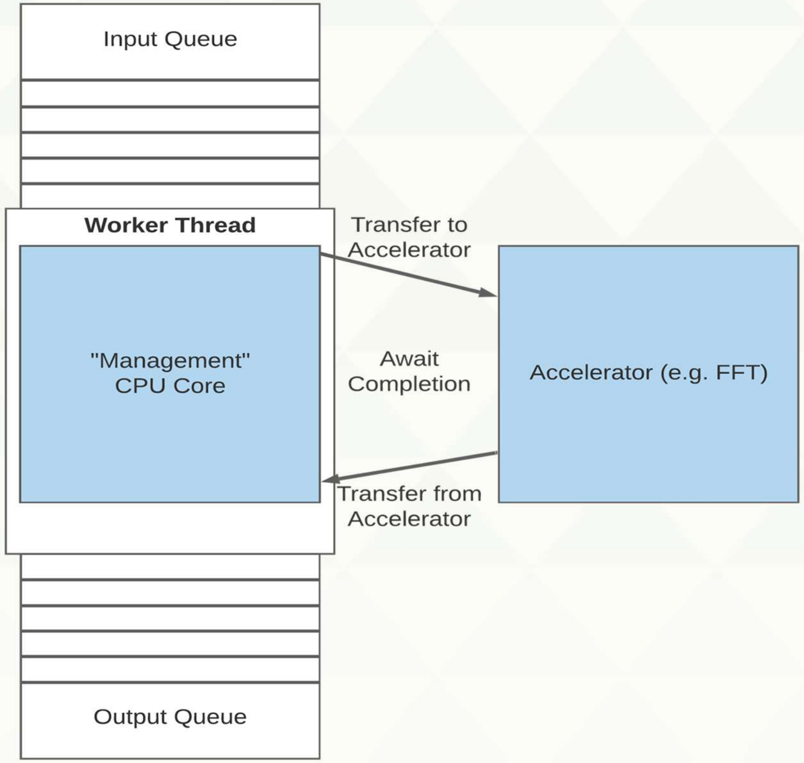


ARM'19 → GNU Radio'20 → FOSDEM'20 → FOSDEM'21 → GNU Radio'22 → ESWEEK'23 → ISFPGA'24

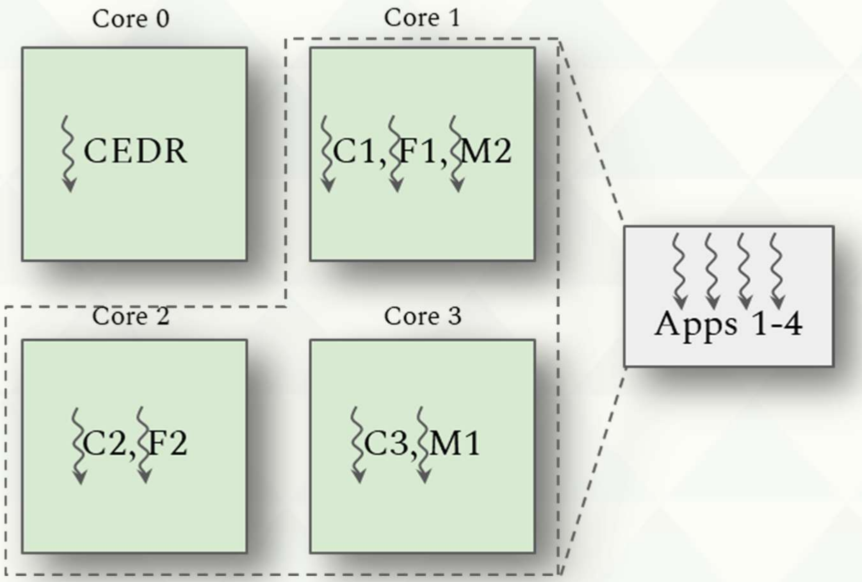
*Mack et al. , "CEDR-API: Productive, Performant Programming of Domain-Specific Embedded Systems," HCW'23 DOI:10.1109/IPDPSW59300.2023.00016

*Kumbhare et al., "User-Space Emulation Framework for Domain-Specific SoC Design," HCW'20 http://dx.doi.org/10.1109/IPDPSW50202.2020.00016

Scalability Limitation of Runtime System



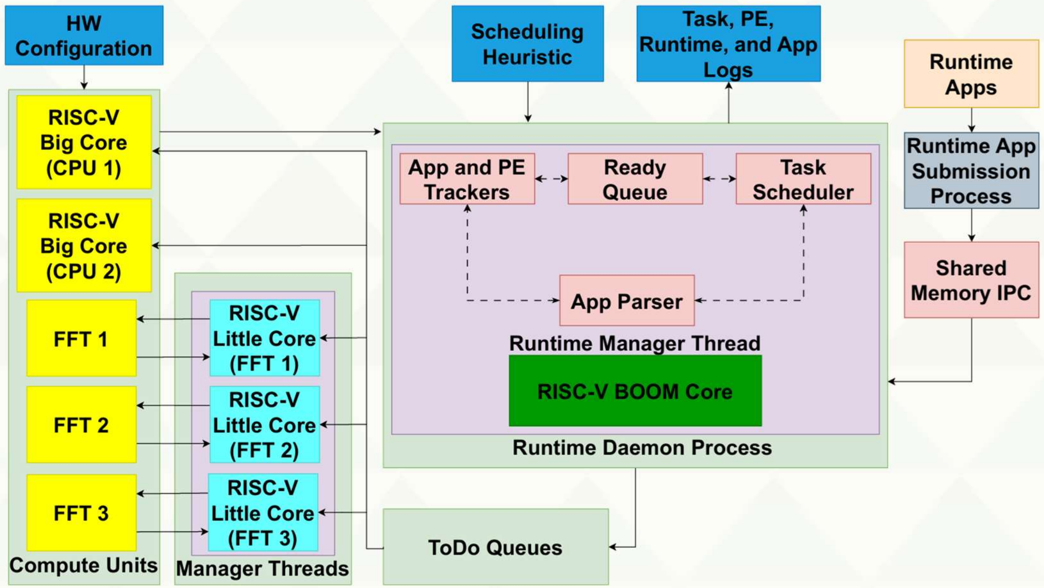
CEDR Worker Thread Overview



<u>Scenario:</u> 3 Cores (C1-C3), 2 FFTs (F1, F2), 2 MMULTs (M1,M2) as worker threads for 4 concurrent apps

➔ Cores 1-3: 3.67 threads/Core

CEDR RISC-V Integration



CEDR workflow with BOOM core for runtime manager thread, 3 ”little cores” for managing the 3 FFT accelerators, and two ”big cores” as CPU compute resources.

Hardware Unit	Parameter	BOOM	Big	Little
CPU	Branch Prediction	✓	✓	✗
	Floating Point Unit	✓	✓	✗
	Out-of-Order	✓	✗	✗
DCache/ICache	nSets	64/64	64/64	32/32
	nWays	4/4	4/4	1/1
	nTLBSets	1/1	1/1	1/1
	nTLBWays	8/32	32/32	8/8
	blockBytes	64/64	64/64	64/64

Configuration of RISC-V Cores

- Little cores handle accelerator management
  - low overhead execution
  - improves scalability

Experimental Setup

Hardware Unit	Number of Units	LUT Utilization	LUTRAM Utilization	DSP Utilization
FFT	3	1.68%	1.06%	1.21%
BOOM Core	1	21.70%	0.76%	1.29%
Big Core	2	8.60%	0.27%	0.54%
Little Core	3	3.96%	0.15%	0.21%

Utilization per Hardware Unit

Resource Pool

- 2 CPUs
- 3 FFT Accelerators

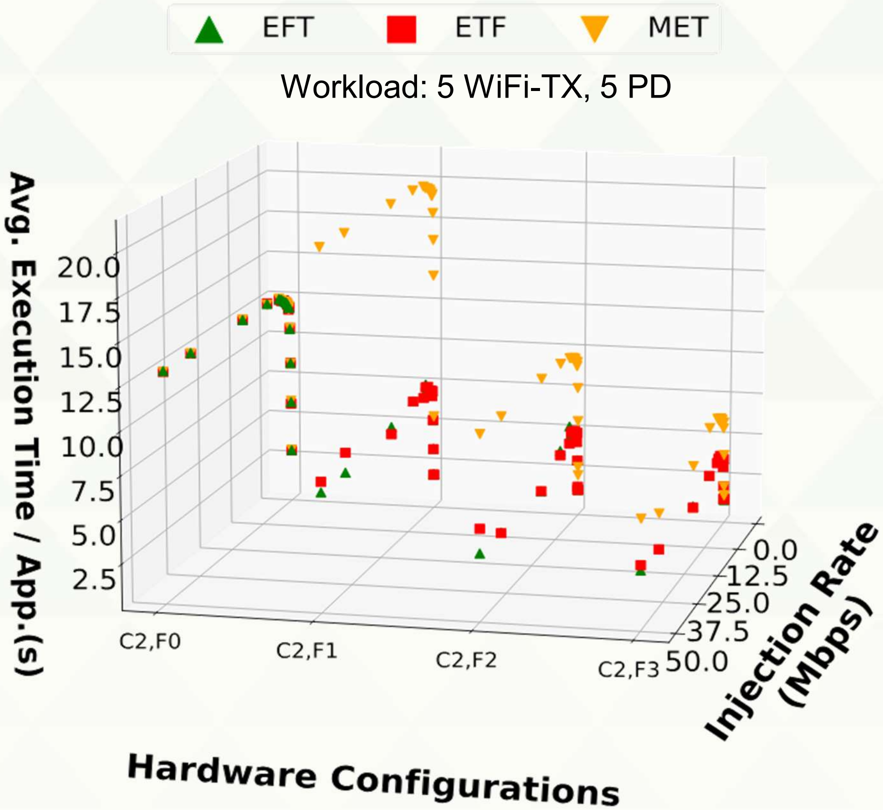
Workload Composition

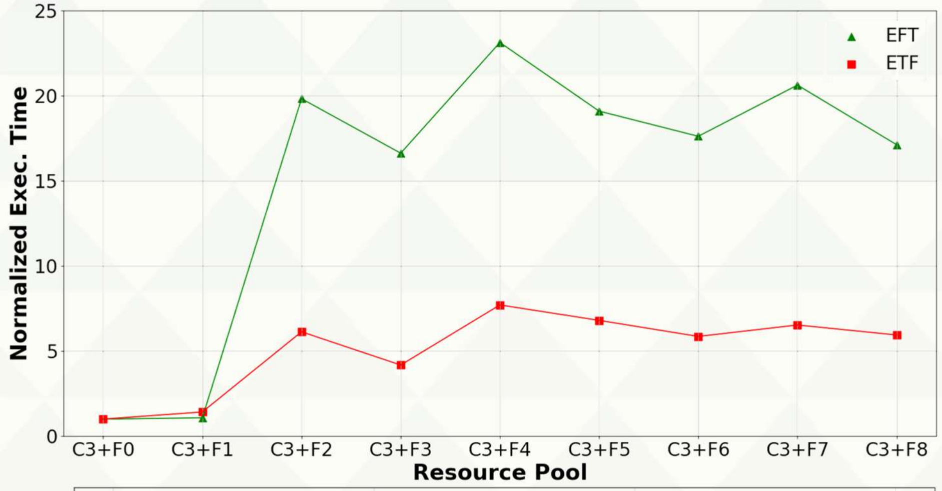
- Radar Correlator
- Pulse Doppler
- WiFi-TX
- Synthetic Aperture Radar
- Lane Detection

Scheduling Heuristics

- Earliest Finish Time (EFT)
- Earliest Time to Finish (ETF)
- Minimum Execution Time (MET)

Results





HETEROGENEITY in COMPUTING WORKSHOP

2023

HETEROGENEITY in COMPUTING WORKSHOP

2024

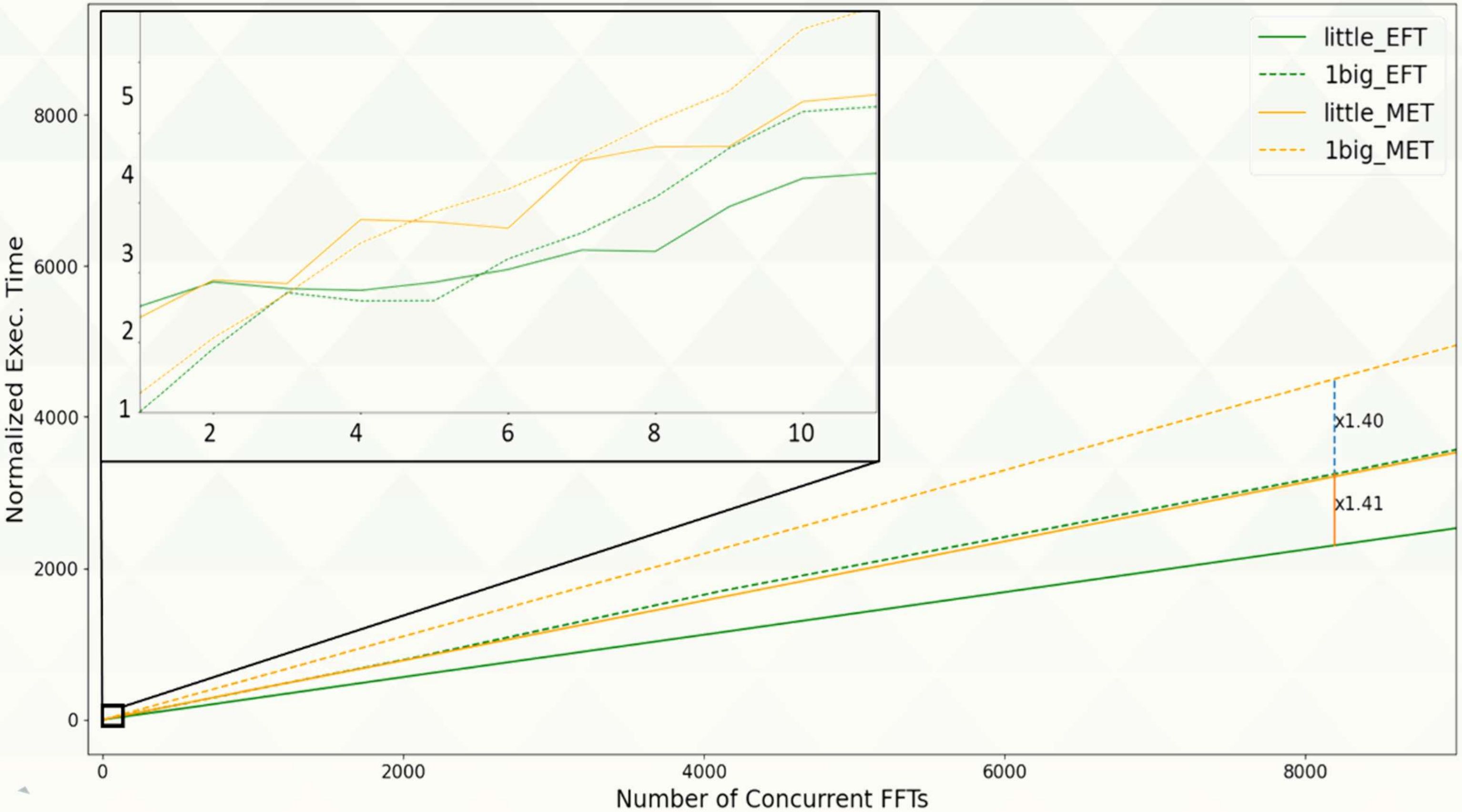
Light RISC-V cores enables execution time reduction with the increase in compute resources

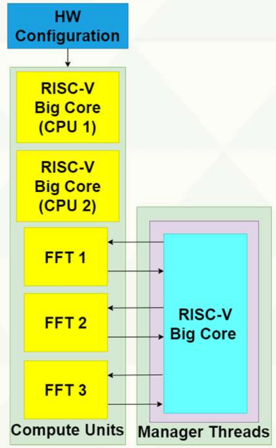
COLLEGE OF ENGINEERING Electrical & Computer Engineering

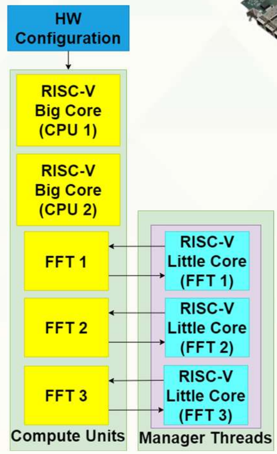
Reconfigurable Computing Lab

* J. Mack et al. "CEDR-API: Productive, Performant Programming of Domain-Specific Embedded Systems," IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2023,

Results





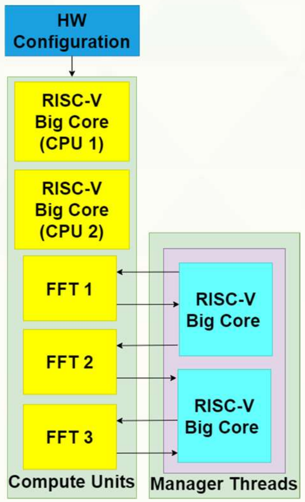


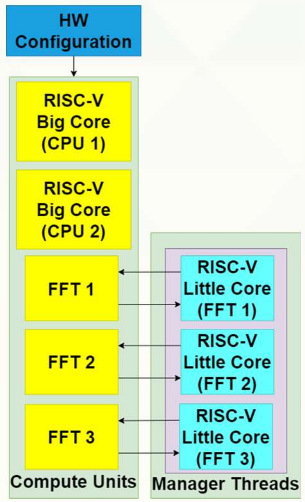
Stress test with FFT only tasks: single big core vs. three little cores

Results

	Number of Tasks	Concurrency Degree	2big (sec)	3little (sec)	Speedup	Overhead Reduction
RC	3	1x3	0.191	0.202	0.95x	3.29%
WiFi TX	100	10x10	3.291	2.939	1.12x	29.55%
PD	512	256x1|128x2	12.590	7.395	1.70x	36.61%
SAR	2,305	256x3|512x3	17.641	8.642	2.04x	47.31%
LD	6,148	256x24	46.03	20.35	2.26x	41.99%

Big cores vs Little cores for distributed accelerator management





Conclusions and Future Work

- Designed and developed an ecosystem that will serve application engineers and hardware architects for rapidly experimenting with RISC-V integrated heterogeneous systems.
- Showcased the versatility of CEDR framework capable of managing resources for RISC-V integrated heterogeneous systems
- Investigated the trade-off between centralized and distributed accelerator management

- Future Work
  - Integrate richer set of RISC-V cores with customized datapaths to support pipelined and dataflow-centric computations
  - Utilize customized RISC-V cores for handling accelerator-to-accelerator data flow management

Thank you!

Contact:
{suluhan, gener, akoglu}@arizona.edu

COLLEGE OF ENGINEERING
Electrical & Computer Engineering

Reconfigurable Computing Lab