

FireSim and Chipyard Tutorial: Welcome!

You must enter a valid email on this form! →

1. Fill out the form at [in-person only] now for EC2 instance access



2. You'll receive two emails. Follow insts to login, then wait.



Berkeley
Architecture
Research



FireSim and Chipyard Tutorial: Intro

Sagar Karandikar

UC Berkeley

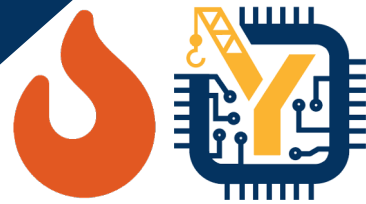
sagark@eecs.berkeley.edu



Berkeley
Architecture
Research



Presenters/Organizers



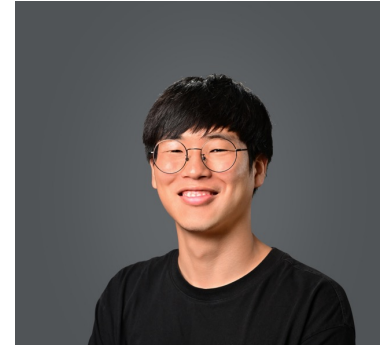
Sagar
Karandikar



Jerry
Zhao



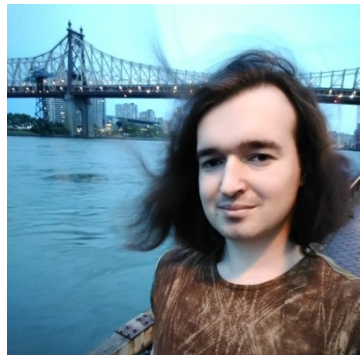
Vighnesh
Iyer



Joonho
Whangbo



JunSun
Choi



Dima
Nikiforov



Kris
Dong



Sophia
Shao



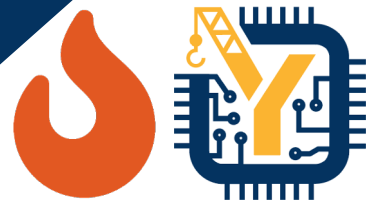
Bora
Nikolić



Krste
Asanović



Getting Started/Logistics (recap)



- Fill out the form at **[in-person only]** now for EC2 instance access
- You'll receive two emails. One from Google Forms and one that looks like →
- Follow the instructions in this one to login to your FireSim manager instance
- Finally, on the manager, run:

```
$ tail -n1 /tutorial-launchstatus.log  
+ echo 'DONE!'
```
- ^ this should be the output, otherwise complain

FireSim/Chipyard Tutorial User Info Inbox x



FireSim Tutorial User Registration <mailgun@mg.sagark.org> to sagark ▾

Welcome to the FireSim/Chipyard tutorial!

Your Instance IP is 3.86.98.198
Your Instance Username is centos

There are two steps to login:

1) Save the attached key. You will likely need to fix permissions on it like so:

```
chmod 0600 tutorial-user-0000-us-east-1.pem
```

2) Next, there are two options for logging in, choose one. Mosh is highly recommended.

2a) If you have mosh installed (or can install it) we highly recommend logging in with mosh. See <https://mosh.org/#getting>

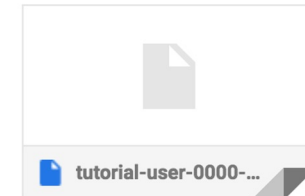
Once installed, to login with mosh, run:

```
mosh --ssh="ssh -i tutorial-user-0000-us-east-1.pem" centos@3.86.98.198
```

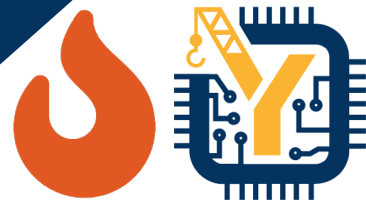
2b) If mosh is not available, login with a regular ssh client, then run screen on the remote instance.

```
ssh -i tutorial-user-0000-us-east-1.pem centos@3.86.98.198  
[now, start a screen on the remote instance]
```

Please let a presenter know if you have issues logging in.



A *Golden Age* in Computer Architecture



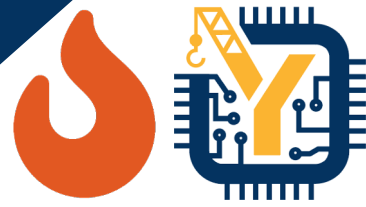
- No more traditional scaling...
- An architect's dream: everyone wants custom microarchitectures and HW/SW co-designed systems
- Also, a golden age to have *direct impact* as researchers
 - Exploding open-source hardware environment
 - An open-ISA that can run software we care about



<https://cacm.acm.org/magazines/2019/2/234352-a-new-golden-age-for-computer-architecture/fulltext>



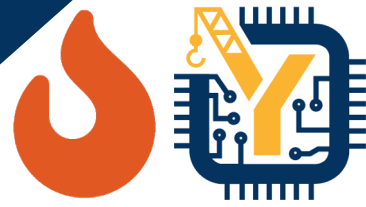
A *Dark Age* in Computer Architecture tools



- What do we need to do good architecture research?
 - Need tools that let us evaluate designs on a variety of metrics:
 - Functionality
 - Performance
 - Power
 - Area
 - Frequency
 - Especially in small teams (grad students, startups), these tools need to be *agile*
 - Historically, without good open IP, had to build abstract arch/uarch simulators out of necessity
 - But now, we have much better IP and software compatibility, so what's stopping us?



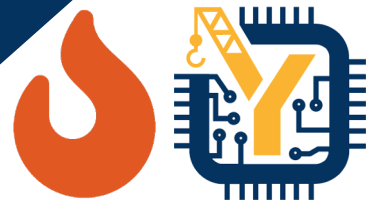
A *Dark Age* in Computer Architecture tools



- Designed to be operated by hundreds of engineers
- Not, 10s of engineers or 1s-10s of grad students
- Three hard questions:
 - Where do I get a collection of well-tested hardware IP + complex software stacks that run on it?
 - How do I quickly obtain performance measurements for a novel HW/SW system?
 - How do I get ASIC QoR feedback and tape-out a design, with portability across tools and processes?



Three hard questions, answered!



- Where do I get a collection of well-tested hardware IP + complex software stacks that run on it?



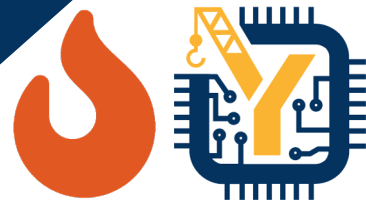
- How do I quickly obtain performance measurements for a novel HW/SW system?



- How do I get ASIC QoR feedback and tape out a design, with portability across tools and processes? (and open-source and proprietary flows)



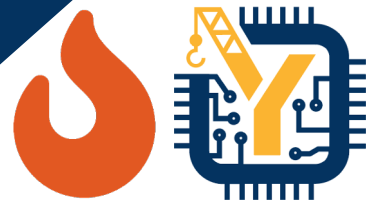
Three hard questions, answered!



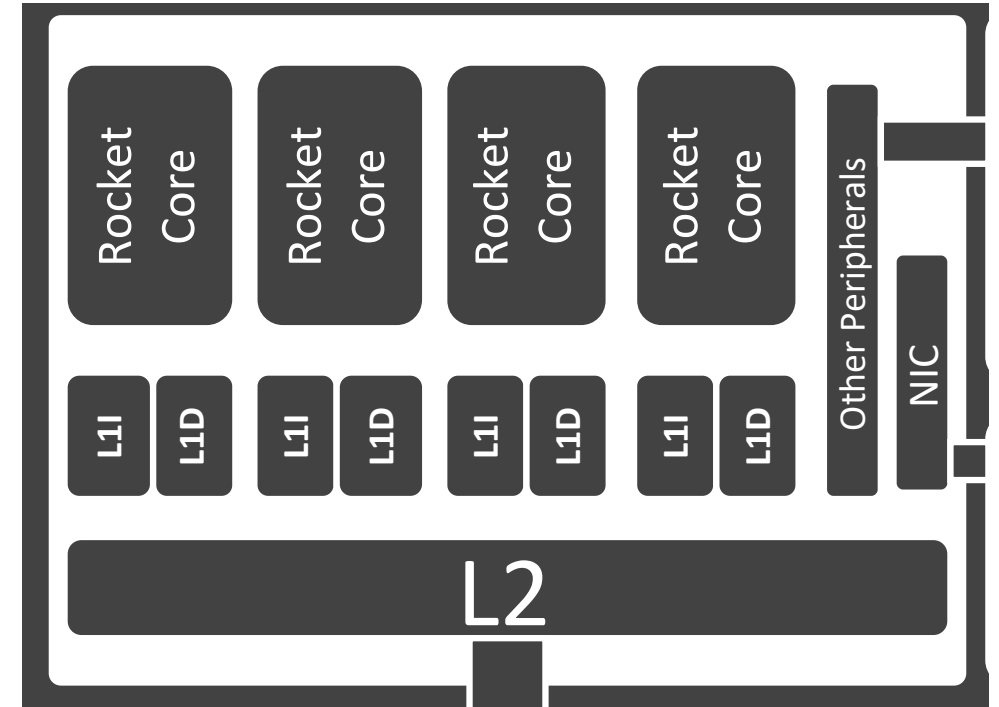
**Measure Functionality, Performance, Power,
Area, Frequency *for real HW/SW systems,*
quickly and easily, with small teams of engineers**



What kinds of designs can I work with?



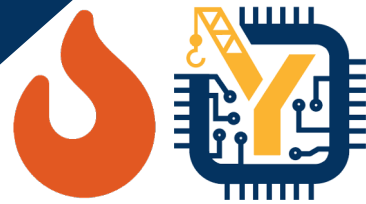
- RISC-V Cores:
 - Rocket Chip In-Order core, industry proven
 - SonicBOOM Out-of-Order Superscalar core
 - CVA6
 - Ibex
- Accelerators
 - Gemmini (Berkeley DNN Accelerator)
 - sha3 accelerator
 - NVDLA (NVIDIA Deep Learning Accelerator)
 - Hwacha Vector Accelerator
 - FFT Generator
 - Many more
- Peripherals/other IP
 - L2 Cache, UART, Disk, Ethernet NIC, etc.
- FPGA-Simulation Models
 - Large LLCs, large DDR3 memory systems



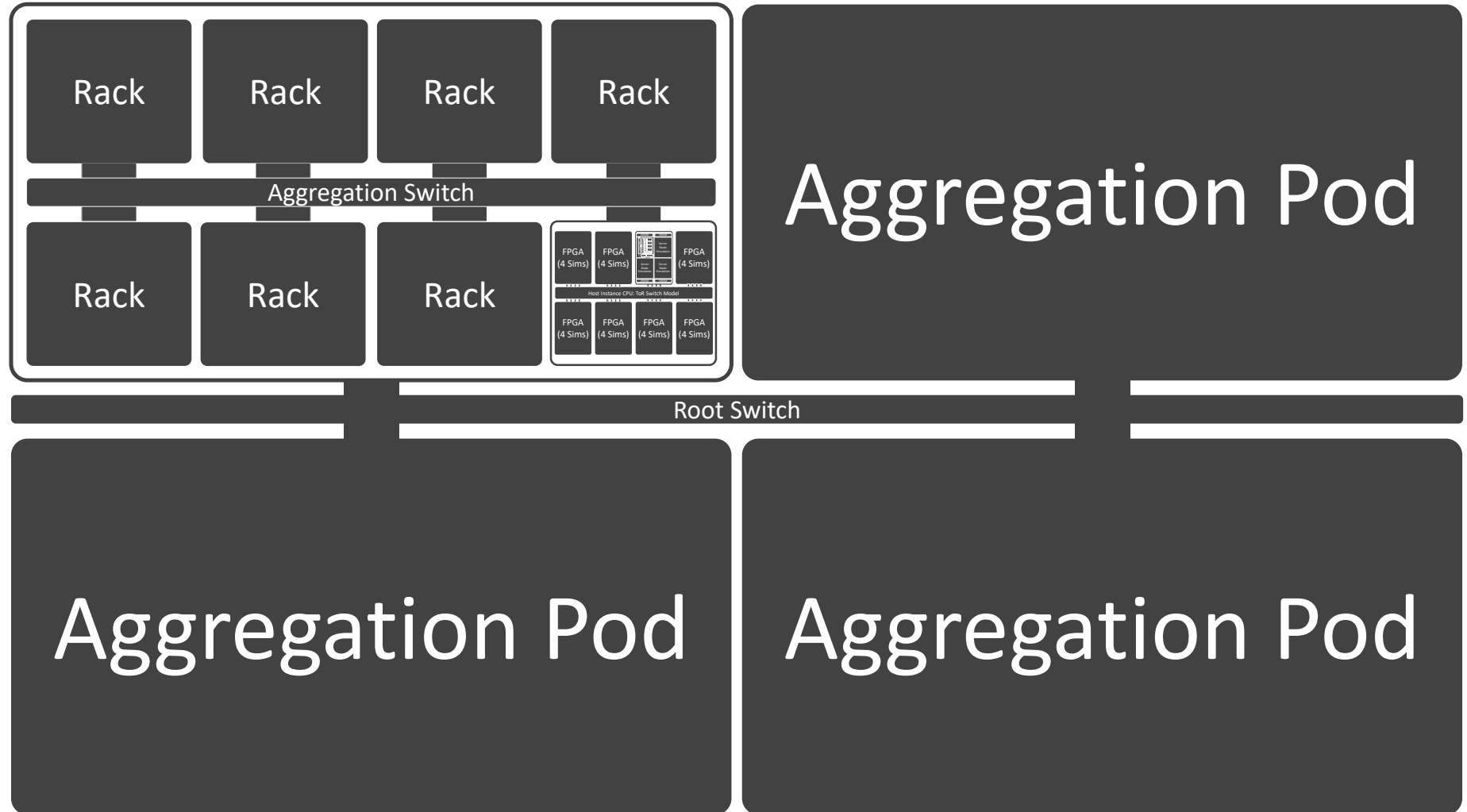
Complete Single-
SoC RISC-V System



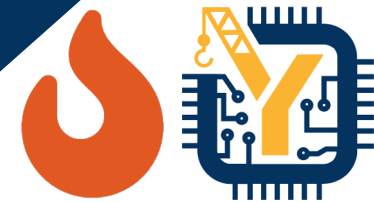
What kinds of designs can I work with?



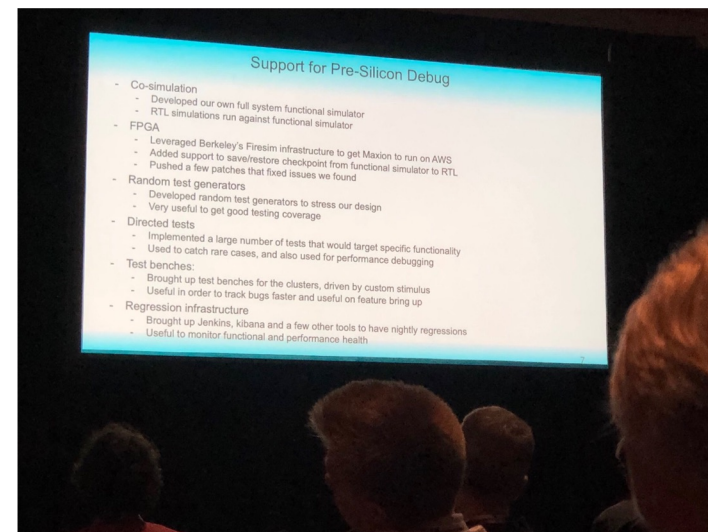
Ethernet-
Networked
1024 Node
(4096 Core)
System on
256 Cloud
FPGAs



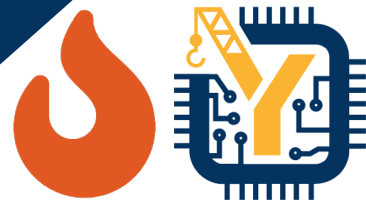
Join the FireSim Community! Open-source users and industrial users



- More than 200 mailing list members and 850 unique cloners per-week
- Projects with public FireSim support
 - Chipyard
 - Rocket Chip
 - BOOM
 - Hwacha Vector Accelerator
 - Keystone Secure Enclave
 - Gemmini
 - NVIDIA Deep Learning Accelerator (NVDLA):
 - NVIDIA blog post: <https://devblogs.nvidia.com/nvdl/>
 - BOOM Spectre replication/mitigation
 - Protobuf Accelerator
 - Too many to list here!
- Companies publicly announced using FireSim
 - SiFive validation paper @ VLSI'20
 - Esperanto Maxis ET
 - Intensivate IntenCore
 - Tenstorrent
 - Galois (DARPA SSITH/FETT)



FireSim in DARPA FETT



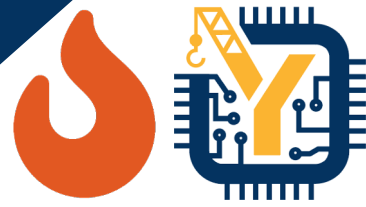
- DARPA SSITH: Building hardware defenses to address common software vulnerabilities
- DARPA FETT: How good are the defenses built in SSITH?
 - Multiple designs hosted for attack in FireSim [1]
- “Morpheus II: A RISC-V Security Extension for Protecting Vulnerable Software and Hardware”
 - Developed by UT Austin, U Mich., Agita Labs
 - Hosted on FireSim for FETT [2]
 - Over 500 attackers tried to break Morpheus II defenses, working for large bug bounties. None succeeded [3]



- [1] K. Hopfer. Leveraging Amazon EC2 F1 Instances for Development and Red Teaming in DARPA’s First-Ever Bug Bounty Program. AWS APN Blog. May 2021.
- [2] A. Harris, et. al., “Morpheus II: A RISC-V Security Extension for Protecting Vulnerable Software and Hardware”. In proceedings of the 2021 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), December 2021.
- [3] T. Austin., et. al., “Morpheus II: A RISC-V Security Extension for Protecting Vulnerable Software and Hardware”. In HotChips 33, August 2021.



Join the FireSim Community! Academic Users and Awards



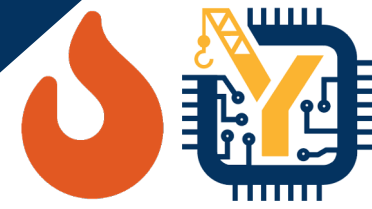
- **ISCA '18**: Maas et. al. HW-GC Accelerator (**Berkeley**)
- **MICRO '18**: Zhang et. al. “Composable Building Blocks to Open up Processor Design” (**MIT**)
- **RTAS '20**: Farshchi et. al. BRU (**Kansas**)
- **EuroSys '20**: Lee et. al. Keystone (**Berkeley**)
- **OSDI '21**: Ibanez et. al. nanoPU (**Stanford**)
- **USENIX Security '21**: Saileshwar et. al. MIRAGE (**Georgia Tech**)
- **CCS '21**: Ding et. al. “Hardware Support to Improve Fuzzing Performance and Precision” (**Georgia Tech**)
- **MICRO '21**: Karandikar et. al. “A Hardware Accelerator for Protocol Buffers” (**Berkeley/Google**)
- **MICRO '21**: Gottschall et. al. TIP (**NTNU**)
- **Over 20 additional user papers on the FireSim website:**
 - <https://fires.im/publications/#userpapers>

FireSim and User Awards:

- **ISCA '18 FireSim Paper:**
 - IEEE Micro Top Pick
 - CACM Research Highlights Nominee from ISCA '18
- **ISCA '18 Maas et. al.:**
 - IEEE Micro Top Pick
- **MICRO '18 Zhang et. al.:**
 - IEEE Micro Top Pick
- **MICRO '21 Gottschall et. al.:**
 - MICRO-54 Best paper runner-up
- **MICRO '21 Karandikar et. al.:**
 - MICRO-54 Distinguished Artifact winner
 - IEEE Micro Top Pick Honorable Mention
- **DAC '21 Genc et. al.:**
 - DAC 2021 Best Paper winner



Join the FireSim Community! Academic Users and Awards



- **ISCA '18**: Maas et. al. HW-GC Accelerator (**Berkeley**)
- **MICRO '18**: Zhang et. al. "Composable Building Blocks to Open up Processor Design" (**MIT**)
- **RTAS '20**: Fa...
- **EuroSys '20**:
- **OSDI '21**: Iba...
- **USENIX Sec**
- **CCS '21**: Ding Performance
- **MICRO '21**: K Buffers" (**Berkeley/Google**)
- **MICRO '21**: Gottschall et. al. TIP (**NTNU**)
- **Over 20 additional user papers on the FireSim website:**
 - <https://firesim.com/publications/#userpapers>

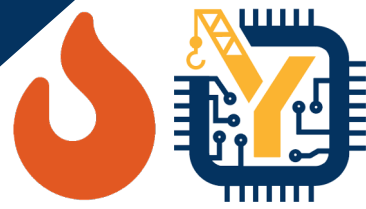
FireSim has been used in published work from authors at over 20 academic and industrial institutions*

**actually used, not only cited*

FireSim and User Awards:

- ISCA '18 FireSim Paper:
 - IEEE Micro Top Pick
 - Distinguished Artifact Winner
 - Distinguished Artifact Nominee
- ... al.:
- ... runner-up
- ... et. al.:
- MICRO-54 Distinguished Artifact winner
- IEEE Micro Top Pick Honorable Mention
- DAC '21 Genc et. al.:
- DAC 2021 Best Paper winner

Learn more about FireSim use-cases, directly from the users!



- Videos from the First FireSim/Chipyard Workshop, co-located with ASPLOS 2023, are now available on YouTube!
 - 10 great talks from users across academia and industry
- Links available at: <https://firesim.org/workshop-2023/>

First FireSim and Chipyard User and Developer Workshop at ASPLOS 2023

March 26, 2023 - Vancouver, BC, Canada

Table of Contents

1. Overview	3. Program/Schedule	5. Workshop Organizers	7.
2. Keynote	4. Registration	6. Is there also a tutorial?	8.

Overview

The FireSim and Chipyard user and developer community has experienced rapid growth, with developer collaborations. This full-day workshop at ASPLOS 2023 aims to bring together the future direction of this ecosystem and spawn new collaborations.

This workshop will feature talks from academic and industrial users of FireSim and Chipyard architecture, systems, programming languages, and VLSI research/development. We hope to inspire lively discussion of FireSim/Chipyard governance, feature roadmaps, outreach and more.

Keynote

FireSim in High-Profile Action—FETT: DARPA's First Ever Bug Bounty Program
[Joe Kiniry](#), Principal Scientist, Galois



Bio: Dr. Kiniry is a Principal Scientist at Galois and the Research Assurance Secure Hardware/Firmware Design and Verification, Research Assurance Model-Based Systems and Software Engineering with Verifiable Elections, High-assurance Cryptography, and Audits-for-CEO and Chief Scientist of Free & Fair, a Galois spin-out focusing on technologies and services.

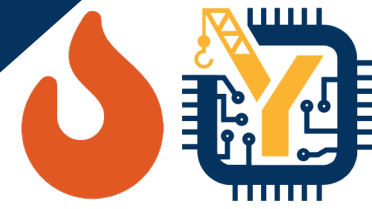
Abstract:

Joe will talk about FETT, DARPA's first ever bug bounty program, and how FireSim played a role in the program. Information about FETT is found here: <https://fett.darpa.mil/>. FETT was a part of the DARPA [Research Assurance Model-Based Systems and Software Engineering with Verifiable Elections](#).

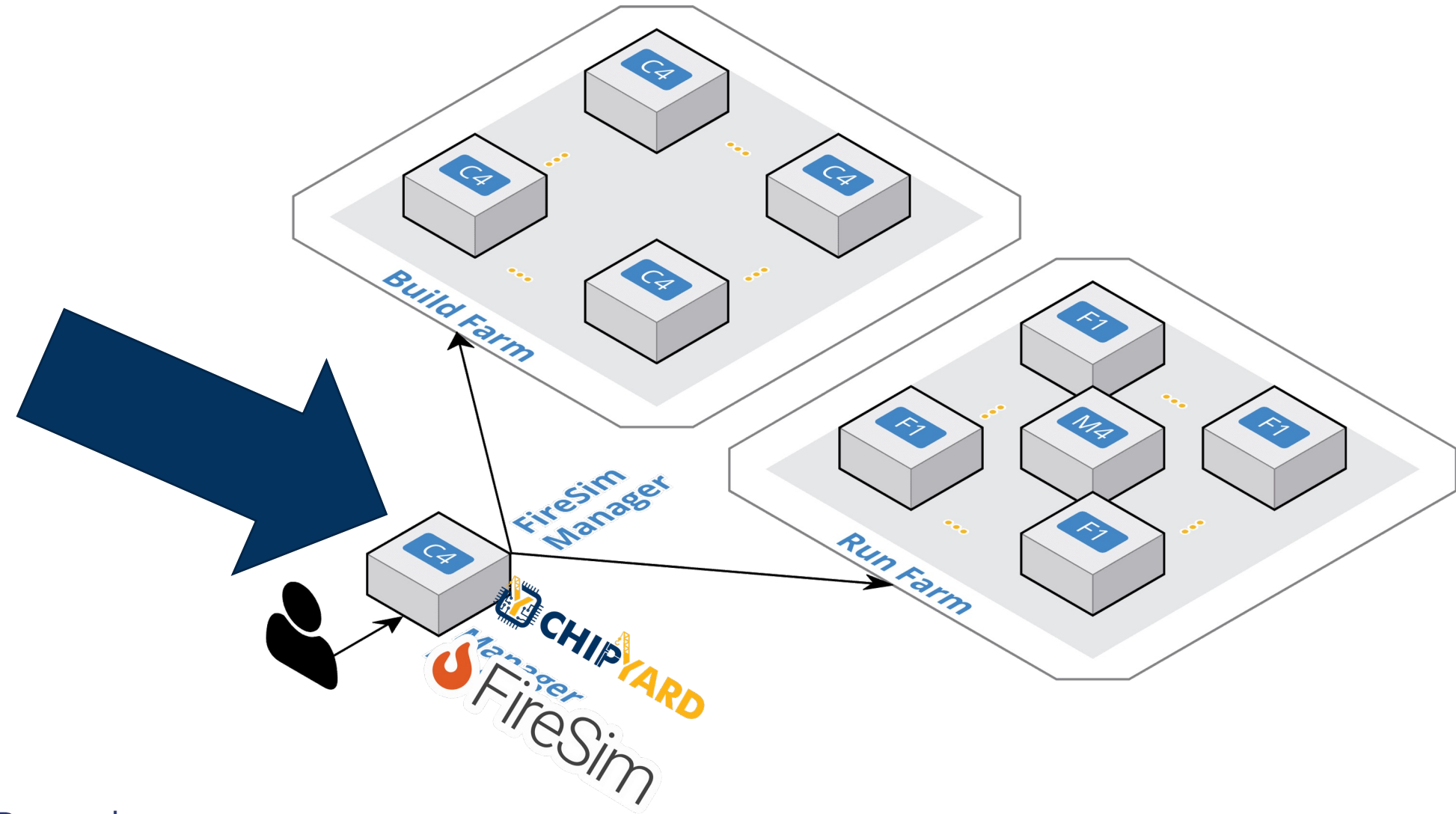
Program/Schedule

Time (PDT)	Talk Title and Authors	Slides (coming soon)
9:00am - 9:30am	Intro and Welcome Workshop Organizers	PDF
9:30am - 10:20am	Keynote: "FireSim in High-Profile Action—FETT: DARPA's First Ever Bug Bounty Program" Joe Kiniry (Galois, Inc.)	PDF
10:20am - 10:40am	Coffee Break	
10:40am - 11:05am	"TraceDoctor: Versatile High-Performance Tracing for FireSim" Björn Gottschall (Norwegian University of Science and Technology), Magnus Jahre (Norwegian University of Science and Technology)	PDF
11:05am - 11:30am	"Integrating a high performance instruction set simulator with FireSim to cosimulate operating system boots" Jiahua Zhang (Tenstorrent Inc.), Varun Koyyalagunta (Tenstorrent Inc.), Joe Rahmeh (Tenstorrent Inc.), Divyanshu Agrawal (Tenstorrent Inc.)	PDF
11:30am - 12:00pm	"Developing and Evaluating the nanoPU and nanoSort using Chipyard and FireSim" Stephen Ibanez (Stanford University & Intel), Theo Jepsen (Stanford University & Intel)	PDF
12:00pm - 1:40pm	Lunch	
1:40pm - 2:05pm	"MIRAGE: Mitigating Cache Attacks with a Randomized Fully-Associative Cache" Gururaj Saileshwar (NVIDIA Research & University of Toronto), Mohammed Qureshi (Georgia Tech)	PDF
2:05pm - 2:30pm	"ChipShop: A Cloud-Based GUI for Accelerating SoC Design" Shahzaib Kashif (Usman Institute of Technology), Talha Ahmed (Usman Institute of Technology), Farhan Ahmed Karim (Universiti Kebangsaan Malaysia)	PDF
2:30pm - 2:55pm	"Profiling an Architectural Simulator" Johnson Umeike (University of Kansas), Alex Manley (University of Kansas), Neel Patel (University of Kansas), Mohammad Alian (University of Kansas)	PDF
2:55pm - 3:20pm	"Berkeley eXtensible Environment: A Cloud-Based Open-Source Computer Architecture Simulation Environment" Farzad Fatollahi-Fard (Lawrence Berkeley National Laboratory), Nirmalendu Patra (Lawrence Berkeley National Laboratory), Angelos Ioannou (Lawrence Berkeley National Laboratory), John Shalf (Lawrence Berkeley National Laboratory)	PDF
3:20pm - 3:40pm	Coffee Break	
3:40pm - 4:05pm	"FireSim on Xilinx U250 and Other Custom Host Platforms" David Christoph Metz (Norwegian University of Science and Technology), Magnus Sjölander (Norwegian University of Science and Technology)	PDF
4:05pm - 4:30pm	"Ocelot Vector Unit and Integrating SV-based Modules in BOOM" Dongjie (DJ) Xie (Tenstorrent Inc.), Srikanth Arekapudi (Tenstorrent Inc.)	PDF
4:30pm - 5:00pm	Wrap-up and Discussion Workshop Organizers/Attendees	PDF

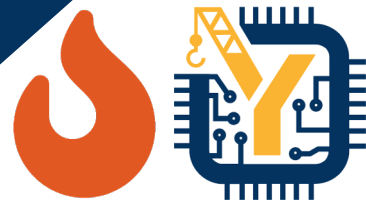
Today's Logistics



You are here



Running a FireSim FPGA Build



- This will take a while, so we will run this in the background:

```
tmux new -s fpgabuild # this will give you a persistent
                        # session you can reattach to
firesim managerinit --platform f1
```

[When prompted, enter your email address to get a build completion notification]

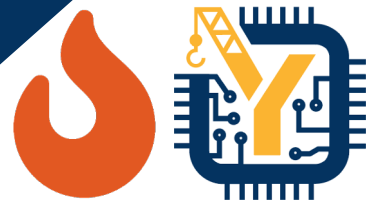
```
# runs the HW build, all the way to AGFI
firesim buildbitstream
```

[Lastly, detach from tmux with “ctrl-b d”. We will return to this build later.]

[this will build a design called firesim_rocket_singlecore_no_nic_l2_lbp]



Today's Agenda



9:00am: Introduction/Overview, Amazon EC2 Instance Setup, Logistics - Sagar

9:30am: Chipyard Basics – Jerry

10:00am: Building Custom RISC-V SoCs in Chipyard – Jerry

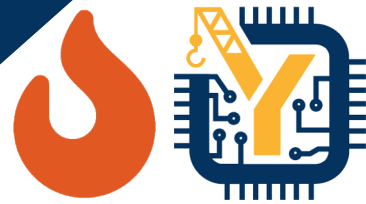
11:00am: Coffee break

11:20am: Hammer VLSI flow – Vighnesh

11:50am: FireSim Introduction – Sagar

12:30pm – 2:00pm: Lunch

Today's Agenda



2:00pm: Building Hardware Designs in FireSim – JunSun

2:30pm: Building Software Workloads with FireMarshal – Jerry

3:00pm: Running a FireSim Simulation: Booting Linux and Running Hardware Accelerated ResNet-50 – Joonho

3:30pm: Coffee break

4:00pm: Debugging and Profiling FireSim-Simulated Designs – Sagar

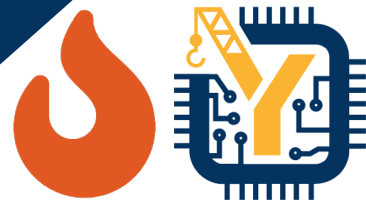
4:30pm: FireSim Local (On-Prem) FPGA Demo – Sagar

4:45pm: RoSÉ: Full Stack Pre-Silicon Evaluation of Robotics SoCs using FireSim – Dima & Kris

5:15pm: Conclusion – Sagar

5:30pm: End of Tutorial

Thanks to AWS, Xilinx, and SLICE/ADEPT Lab Sponsors



SLICE Lab Sponsors:



National Science Foundation
WHERE DISCOVERIES BEGIN

NSF Award #2016662
CCRI: ENS: Chipyard

