A New Era of Open-Source Hardware

Christopher Batten

Computer Systems Laboratory
Electrical and Computer Engineering
Cornell University
Accelerators for Machine Learning in the Cloud

**NVIDIA DGX Hopper**
- Graphics processor specialized just for accelerating machine learning
- Available as part of a complete system with both the software and hardware designed by NVIDIA

**Google TPU v4**
- Custom chip specifically designed to accelerate Google’s TensorFlow C++ library
- Tightly integrated into Google’s data centers

**Microsoft Catapult**
- Custom FPGA board for accelerating Bing search and machine learning
- Accelerators developed with/by app developers
- Tightly integrated into Microsoft data center’s and cloud computing platforms
Accelerators for Machine Learning at the Edge

Amazon Echo
- Developing AI chips so Echo line can do more on-board processing
- Reduces need for round-trip to cloud
- Co-design the algorithms and the underlying hardware

Facebook Oculus
- Starting to design custom chips for Oculus VR headsets
- Significant performance demands under strict power requirements

Movidius Myriad 2
Top-five software companies are all building custom accelerators

- **Facebook**: w/ Intel, in-house AI chips
- **Amazon**: Echo, Oculus, networking chips
- **Microsoft**: Hiring for AI chips
- **Google**: TPU, Pixel, convergence
- **Apple**: SoCs for phones and laptops

Chip startup ecosystem for machine learning accelerators is thriving!

How can we **accelerate innovation** in **accelerator-centric** hardware design?
Software Innovation Today

Like climbing an iceberg – much is hidden!

Your proprietary code
- Instagram
- $500K seed with 13 people → $1B

Open-source software
- Python
- Django
- Memcached
- Postgres/SQL
- Redis
- nginx
- Apache, Gnuicorn
- Linux
- GCC

"What Powers Instagram: Hundreds of Instances, Dozens of Technologies"
https://goo.gl/76fWrM

Adapted from M. Taylor, “Open Source HW in 2030,” Arch 2030 Workshop @ ISCA’16
Hardware Innovation Today

Like climbing a mountain – nothing is hidden!

What you have to build
- New machine learning accelerator
- Other unrelated components, anything you cannot afford to buy or for which COTS IP does not do

Closed source
- ARM A57, A7, M4, M0
- ARM on-chip interconnect
- Standard cells, I/O pads, DDR Phy
- SRAM memory compilers
- VCS, Modelsim
- DC, ICC, Formality, Primetime
- Stratus, Innovus, Voltus
- Calibre DRC/RCX/LVS, SPICE

Adapted from M. Taylor, “Open Source HW in 2030,” Arch 2030 Workshop @ ISCA’16
Chip Costs Are Skyrocketing

Adapted from M. Taylor, “Open Source HW in 2030,” Arch 2030 Workshop @ ISCA’16; original: International Business Strategies & T. Austin.
A New Era of Open-Source Hardware

Minimum Viable Product/Prototype

Adapted from M. Taylor, “Open Source HW in 2030,” Arch 2030 Workshop @ ISCA’16; original: International Business Strategies & T. Austin.
Minimum Viable Product/Prototype

Can we use open-source software/hardware to address remaining costs?

Adapted from M. Taylor, “Open Source HW in 2030,” Arch 2030 Workshop @ ISCA’16; original: International Business Strategies & T. Austin.
How can HW design be more like SW design?

<table>
<thead>
<tr>
<th>Open-Source</th>
<th>Software</th>
<th>Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>high-level languages</td>
<td>Python, Ruby, R,</td>
<td>Chisel, PyMTL, PyRTL, MyHDL, JHDL,</td>
</tr>
<tr>
<td></td>
<td>Javascript, Julia</td>
<td>Clash, Calyx</td>
</tr>
<tr>
<td>libraries</td>
<td>C++ STL, Python std libs</td>
<td>BaseJump</td>
</tr>
<tr>
<td>systems</td>
<td>Linux, Apache, MySQL, memcached</td>
<td>Rocket, Pulp/Ariane, OpenPiton,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Boom, FabScalar, MIAOW, Nyuzi</td>
</tr>
<tr>
<td>standards</td>
<td>POSIX</td>
<td>RISC-V ISA, RoCC, TileLink</td>
</tr>
<tr>
<td>tools</td>
<td>GCC, LLVM, CPython, MRI, PyPy, V8</td>
<td>Icarus Verilog, Verilator, qflow, Yosys, TimberWolf, qrouter, magic, klayout, ngspice</td>
</tr>
<tr>
<td>methodologies</td>
<td>agile software design</td>
<td>agile hardware design</td>
</tr>
<tr>
<td>cloud</td>
<td>IaaS, elastic computing</td>
<td>IaaS, elastic CAD</td>
</tr>
</tbody>
</table>
```bash
# Ubuntu Server 16.04 LTS (ami-43a15f3e)
% sudo apt-get update
% sudo apt-get -y install build-essential qflow
% mkdir qflow && cd qflow
% wget http://opencircuitdesign.com/qflow/example/map9v3.v
% qflow synthesize place route map9v3 # yosys, graywolf, qrouter
% wget http://opencircuitdesign.com/qflow/example/osu035_stdcells.gds2
% magic # design def/lef -> magic format
>>> lef read /usr/share/qflow/tech/osu035/osu035_stdcells.lef
>>> def read map9v3.def
>>> writeall force map9v3
% magic # stdcell gds -> magic format
>>> gds read osu035_stdcells.gds2
>>> writeall force
% magic map9v3
>>> gds write map9v3 # design + stdcells magic format -> gds
% sudo apt-get -y install libqt4-dev-bin libqt4-dev libz-dev
% wget http://www.klayout.org/downloads/source/klayout-0.24.9.tar.gz
% tar -xzvf klayout-0.24.9.tar.gz && cd klayout-0.24.9
% ./build.sh -noruby -nopython
% wget http://www.csl.cornell.edu/~cbatten/scmos.lyp
% ./bin.linux-64-gcc-release/klayout -l scmos.lyp ..../map9v3.gds
```
“I’m doing a (free) operating system
(just a hobby, won’t be big and professional like gnu)
for 386(486) AT clones.”

— Linus Torvalds, 1991
A New Era of Open-Source Hardware

RISC-V: The Open era of computing

RISC-V Announces First New Specifications of 2022, Adding to 16
Ratified in 2021 | RISC-V International

Efficient Trace, Supervisor Binary Interface,
Unified Extensible Firmware Interface, and
Zmmul Multiply-Only Extension Accelerate
Embedded- and Large-System Design. Six
Additional Specifications Already In the
Pipeline As Development Extends Into Vertical...

Check out our local language pages!

Get in touch!

Press: press@riscv.org
Analysts: analysts@riscv.org
General: info@riscv.org
Industry Interest in RISC-V is Growing
### RISC-V Hardware and Software Ecosystem

#### Open-source software:
- Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

#### Commercial software:
- Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC ...

#### RISC-V Foundation
- ISA specification
- Golden Model
- Compliance

#### Hardware

<table>
<thead>
<tr>
<th>Open-source cores:</th>
<th>Commercial core providers:</th>
<th>Inhouse cores:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Swerv, Hummingbird, ...</td>
<td>Andes, Bluespec, Cloudbear, Codasip, Cortus, C-Sky, InCore, Nuclei, SiFive, Syntacore, ...</td>
<td>Nvidia, +others</td>
</tr>
</tbody>
</table>
The OpenROAD Project:
Unleashing Hardware Innovation
Andrew B. Kahng⇤,† and Tom Spyrou⇤,‡
(*) CSE and (†) ECE Depts., UC San Diego
‡ Precision Innovations, Inc.

Abstract
The OpenROAD project develops an open-source framework that targets digital IC design flow, enabling portable, flexible, and scalable design software that is open to the broader research community. OpenROAD is built on top of the OpenTitan SoC, a large open-source silicon platform that is produced by the OpenTitan project. OpenROAD is designed to be a platform for research and education, with the goals of accelerating EDA innovation, promoting open-source hardware, and driving new machine learning approaches in hardware design.

OpenROAD: The Future of Open-Source EDA

OpenROAD flow, built around the integrated OpenROAD tool.

- Design: `design.v + .lib + .lef + .sdc + parameters.cfg`
- Flow Setup: `Flow parameters, Library, techfile preparation, Macro wrappers, Dont_use list`
- Logic Synthesis: `Logic optimization, Technology mapping, Buffering, sizing`
- OpenROAD v1.0: `Io placement, Mixed-size + macro placement, Tapcell insertion, PDN generation`
- Floorplanning: `Global placement, Placement-based optimization, Detailed placement`
- Placement: `Clock tree synthesis, CTS, hold, ERC repair, Placement legalization`
- CTS: `Global routing, Antenna check + repair, Detailed routing`
- Routing: `Filler cell + BEOL fill insertion, Merge wrapped macros, Merge GDS`

- Results: `result.def + ppa.rpt + drc.rpt + results.gds`

OpenTitan SoC
GF12LP
Google Partners with SkyWater and Efabless to Enable Open Source Manufacturing of Custom ASICs
A New Era of Open-Source Hardware

chipIgnite

Rapid IC Creation

Shuttle 2110C

$9750 per project

13 of 40 project slots reserved

Tapeout: November 26, 2021 23:59 PT
Deliver: April 01, 2022
A New Era of Open-Source Hardware

YOUR DESIGN

CARAVEL

10mm² User Space

INTEGRATED CHIP

OpenMPW Submissions

Run Capacity

Cornell University Christopher Batten
Colleagues,

Through a generous donor gift creating the Shen Fund for Social Impact we have the opportunity to fund multiple new engineering project teams. This program is designed to bring together new student teams under a faculty member’s mentorship to address significant social challenges through novel and/or advanced engineering solutions. Falling under the Project Team Umbrella, the program will fund up to three new teams per year, with each supported for a three-year period at $30K/yr. The teams will also be provided space and support to design and implement these projects.

Proposals may be submitted by either faculty looking to guide a group of students, or by students who will engage with a faculty member to form the teams.

Attached to this e-mail are three documents:

- Shen Fund FAQ Sp22.pdf: More fully describes the nature of the projects and the goals of the program (also copied to the e-mail below).
- Shen Fund Proposal Template Sp22.docx: Short project proposal form.
- Shen Funded Projects Summary_Sp22.pdf: A summary document of a currently funded teams.

The ideal project will likely develop through discussions with Lauren Stulgis (as director of the project teams) and me. Feel free to reach out to us with rough ideas and concepts and we can help to try to develop a viable proposal.

Proposals will be considered as they arrive, with discussions to strengthen each within the program constraints. The initial application is a simple document identifying the primary goals, technical challenges and plans, timeline and budget, and currently engaged personnel.

Proposals must be uploaded directly to Box by email to: Proposa.zeuyhp9wqg5p8teo@u.box.com. The first round of decisions will be made based on submissions received by 11:59pm on Sunday, March 13, 2022.

Again, please feel free to contact me or Lauren Stulgis with any questions or to discuss potential projects.

Prof. Alan Zehnder
Associate Dean for Undergraduate Programs
177 Rhodes Hall
Phone: (607) 255-9181
email: eng_ugdean@cornell.edu
C2S2: Cornell Custom Silicon Systems Project Team

Three-year student-led project team to tapeout a custom chip in SkyWater 130nm to implement a proof-of-concept system for a campus partner

- Open RISC-V ISA
- Open-Source VexRISCV microcontroller
- Open-Source OpenROAD chip flow
- Open PDK for SkyWater 130nm
- OpenMPW + ChipIgnite w/ efabless

100+ applications → 25 team members

- Digital & Verification Subteam
- Analog Subteam
- Software Subteam
- System Architecture Subteam

User’s Project
~10 mm²
(2.92mm x 3.52mm)
PyMTL

Python-based hardware generation, simulation, and verification framework which enables productive RTL design and multi-level modeling.
A Call to Action

- Open-source hardware needs developers who
  - ... are idealistic
  - ... have lots of free time
  - ... will work for free

- Who might that be?
  **Students!**

- Academics have a practical and ethical motivation for using, developing, and promoting open-source electronic design automation tools and open-source hardware designs