Open Source HW in 2030

Why Architects Need It and It Needs Them

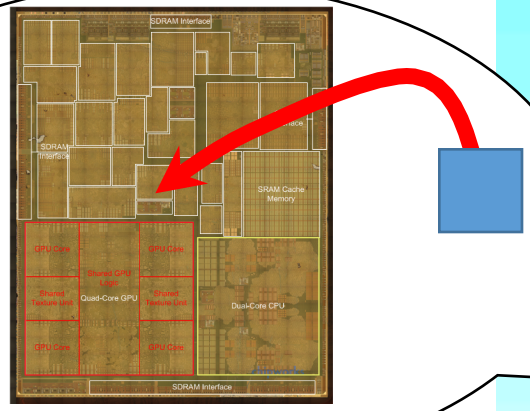
Michael Bedford Taylor

UC San Diego

The Fate of Computing Today

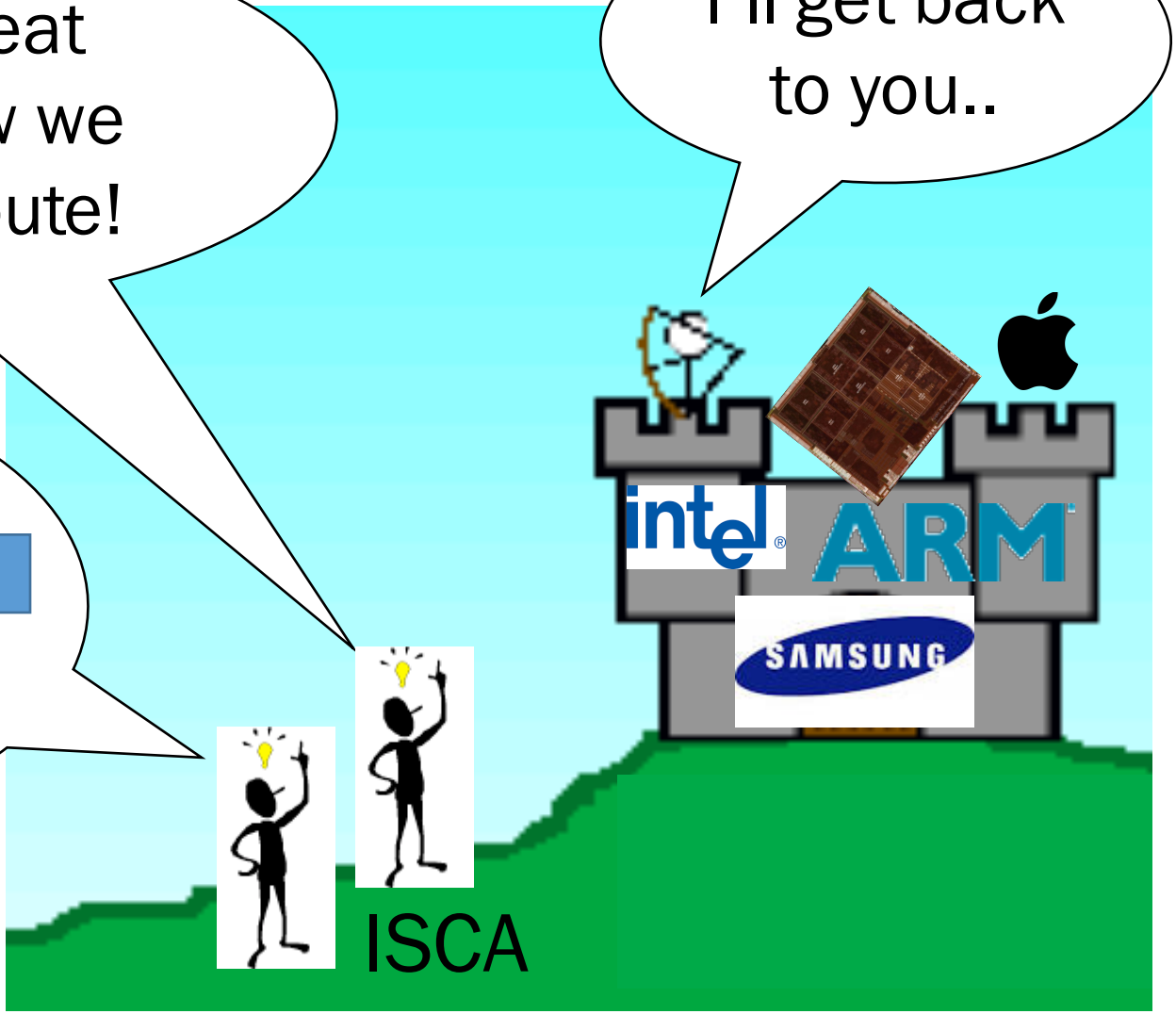
Is determined by a small number of companies…

We have great ideas for how we should compute!



16 nm

I'll get back to you..



What prevents adoption of our ideas?

1. Our methodology is pragmatic but broken

What prevents adoption of our ideas?

1. Our methodology is pragmatic but broken

performance:

```
repeat (modify_c_simulator())
 until (perf>=10%
        || sim_bug_in_my_favor
        || overtrained_on_my_10_benchmarks
        )
```

What prevents adoption of our ideas?

1. Our methodology is pragmatic but broken

performance:

```
repeat (modify_c_simulator())
 until (perf>=10%
        || sim_bug_in_my_favor
        || overtrained_on_my_10_benchmarks
        )
assert(it_would_really_work_in_hw)
```

What prevents adoption of our ideas?

1. Our methodology is pragmatic but broken

performance:

```
repeat (modify_c_simulator())
 until (perf>=10%
        || sim_bug_in_my_favor
        || overtrained_on_my_10_benchmarks
       )
assert(it_would_really_work_in_hw)
```

power:

```
assert(we_used_McPat && no_space_to_describe)
```

What prevents adoption of our ideas?

2. Our stuff works great ... for *our* CPU/GPU microarch

... but theirs is different

What prevents adoption of our ideas?

3. We didn't solve all of the important problems

"*Context switching happens .. out of band ..*"

What prevents adoption of our ideas?

4. “We regret to inform you that your idea was too revolutionary for us to consider as the successor to Core2Duo in our roadmap”

“This paper entirely rethinks …

Our cycle-accurate trace simulator…”

What prevents adoption of our ideas?

5. Chicken-and-Egg

16 nm SoCs need huge volume to amortize costs;

your emerging app that needs your accelerator is not already in use by many users; too risky to dedicate that much die area on iPhone 7

→ *No tech transfer*

What prevents adoption of our ideas?

6. The Last Mile

Your idea is great, but probably only you have the will and patience to adapt it to their system … and you don't work there.

What prevents adoption of our ideas?

7. Your awesome needle in the ISCA/MICRO/… haystack

   Everybody shows good results, but unbeknownst to all, yours is actually worth doing!

What prevents adoption of our ideas?

8. Smaller and smaller number of commercial architects have less and less time to find a home for our ideas

Current Tech Transfer Pipeline

```
repeat (modify_c_sim())
    until (perf>=10%
        | | sim_bugr
        | | overtrained
        )
```

C sim
50K LOC

ASIC Clouds: Specializing the Datacenter

Ikuo Magaki[†], Moein Khazraee, Luis Vega Gutierrez, and Michael Bedford Taylor

[†]UC San Diego, Toshiba UC San Diego

ABSTRACT

GPU and FPGA-based clouds have already demonstrated the promise of accelerating computing-intensive workloads with greatly improved power and performance.

In this paper, we examine the design of ASIC Clouds, which are purpose-built datacenters comprised of large arrays of ASIC accelerators, whose purpose is to optimize the total cost of ownership (TCO) of large, high-volume chronic computations, which are becoming increasingly common as more and more services are built around the Cloud model. On the surface, the creation of ASIC clouds may seem highly improbable due to high NREs and the inflexibility of ASICs. Surprisingly, however, large-scale ASIC Clouds have already been deployed by a large number of commercial entities, to implement the distributed Bitcoin cryptocurrency system.

We begin with a case study of Bitcoin mining ASIC Clouds, which are perhaps the largest ASIC Clouds to date. From there, we design three more ASIC Clouds, including a YouTube-style video transcoding ASIC Cloud, a Litecoin ASIC Cloud, and a Convolutional Neural Network ASIC Cloud and show 2-3 orders of magnitude better TCO versus CPU and GPU.

Among our contributions, we present a methodology that given an accelerator design, derives Pareto-optimal ASIC Cloud Servers, by extracting data from place-and-routed circuits and computational fluid dynamic simulations, and then employing clever but brute-force search to find the best jointly-optimized ASIC, DRAM subsystem, motherboard, power delivery system, cooling system, operating voltage, and case design. Moreover, we show how data center parameters determine which of the many Pareto-optimal points is TCO-optimal. Finally we examine when it makes sense to build an ASIC Cloud, and examine the impact of ASIC NRE.

1. INTRODUCTION

In the last ten years, two parallel phase changes in the computational landscape have emerged. The first change is the bifurcation of computation into two sectors: cloud and mobile; where increasingly the heavy lifting and data-intensive codes are performed in warehouse-scale computers or datacenters; and interactive portions of applications have migrated to desktop-class implementations of out-of-order superscalars in mobile phones and tablets.

The second change is the rise of dark silicon [1, 2, 3, 4] and dark silicon aware design techniques [5, 6, 7, 8, 9, 10] such as specialization and near-threshold computation, each of which help overcome threshold scaling limitations that prevent the full utilization of transistors on a silicon die.

Accordingly, these areas have increasingly become the focus of the architecture research community. Recently, researchers and industry have started to examine the conjunction of these two phase changes. GPU-based clouds have been demonstrated as viable by Baidu and others who are building them in order to develop distributed neural network accelerators. FPGA-based clouds have been validated and deployed by Microsoft for Bing [11], by JP Morgan Chase for hedgefund portfolio evaluation [12] and by almost all Wall Street firms for high frequency trading [13]. In these cases, companies were able to ascertain that there was sufficient scale for the targeted application that the upfront development and capital costs would be amortized by a lower total cost of ownership (TCO) and better computational properties. Already, we have seen early examples of customization, with Intel providing custom SKUs for cloud providers [14].

At a single node level, we know that ASICs can offer order-magnitude improvements in energy-efficiency and cost-performance over CPU, GPU, and FPGA. In this paper, we extend this trend and consider the possibility of ***ASIC Clouds***. ASIC Clouds are purpose-built datacenters comprised of large arrays of ASIC accelerators, whose purpose is to optimize the TCO of large, high-volume chronic computations that are emerging in datacenters today. ASIC Clouds are not ASIC supercomputers that scale up problem sizes for a single tightly-coupled computation; rather, ASIC Clouds target workloads consisting of many independent but similar jobs (eg., the same function, but for many users, or many datasets), for which standalone accelerators have been shown to attain improvements for individual jobs.

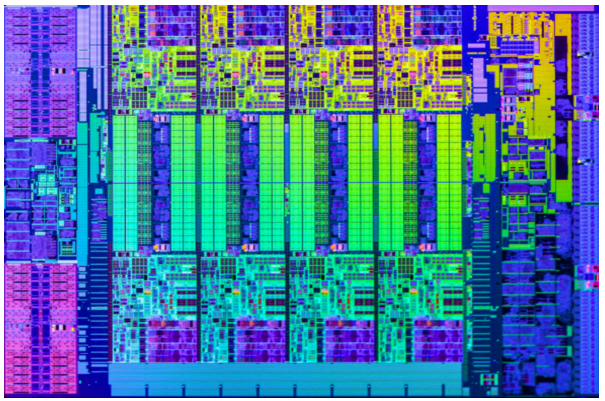
As more and more services are built around the Cloud model, we see the emergence of planet-scale workloads. For example, Facebook's face recognition algorithms are used on 2 billion uploaded photos a day, each requiring several seconds on a GPU [15], Siri answers speech queries, genomics will be applied to personalize medicine, and YouTube transcodes all user-uploaded videos to Google's VP9 format. As computations of this scale become increasingly frequent, the TCO improvements derived from the reduced marginal hardware and energy costs of ASICs will make it an easy and routine business decision to create ASIC Clouds.

ASIC Clouds Exist Today. This paper starts by examining the first large-scale ASIC Clouds, Bitcoin cryptocurrency mining clouds, as real-world case studies to understand the key issues in ASIC Cloud design. Bitcoin clouds implement the consensus algorithms in Bitcoin cryptocurrency systems. Although much is secretive in the Bitcoin mining industry, today there are 20 megawatt facilities in existence, and 40 megawatt facilities are under construction [16], and the global power budget dedicated to ASIC Clouds, large and small, is estimated by experts to be in the range of 300-500 megawatts. After Bitcoin, the paper then examines other applications including YouTube-style video transcoding, Litecoin mining and Convolutional Neural Networks.

Specializing the ASICs. At the heart of every ASIC Cloud is an ASIC design, which typically aggregates a number of accelerators into a single chip. ASICs achieve large reductions in silicon area and energy consumption versus CPUs,

The first two authors contributed equally to this paper.

ISCA
12-pages



Intel
7-nm
5 million units

Maybe we need a few more intermediate points?

Proposed Tech Transfer Pipeline

You

OOO Processor

Multicore

GPU

Community-maintained
Open Source Cores
(Like Linux)

Proto
Implementation

You


ASIC Clouds: Specializing the Datacenter

ISCA
12-pages

You

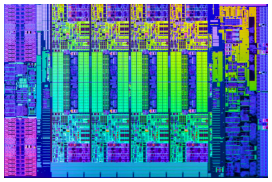
Integrated
Into Community
Open Source

Quadcopter
65 nm ASIC

Samsung
TV 32 nm

Talking
Toy 90 nm

Intel
7-nm
5M units



Maybe we need a few more intermediate points?
This would address all of the 8 problems I showed…
Reproducible, flushed out, real results, ….

Switching gears to a different facet of open source…

What will the Hardware workforce look like in 15 years?

- *Good news*: enrollment in undergrad Computer Architecture: 30→400

What will the Hardware workforce look like in 15 years?

- *Good news*: enrollment in undergrad Computer Architecture: 30→400

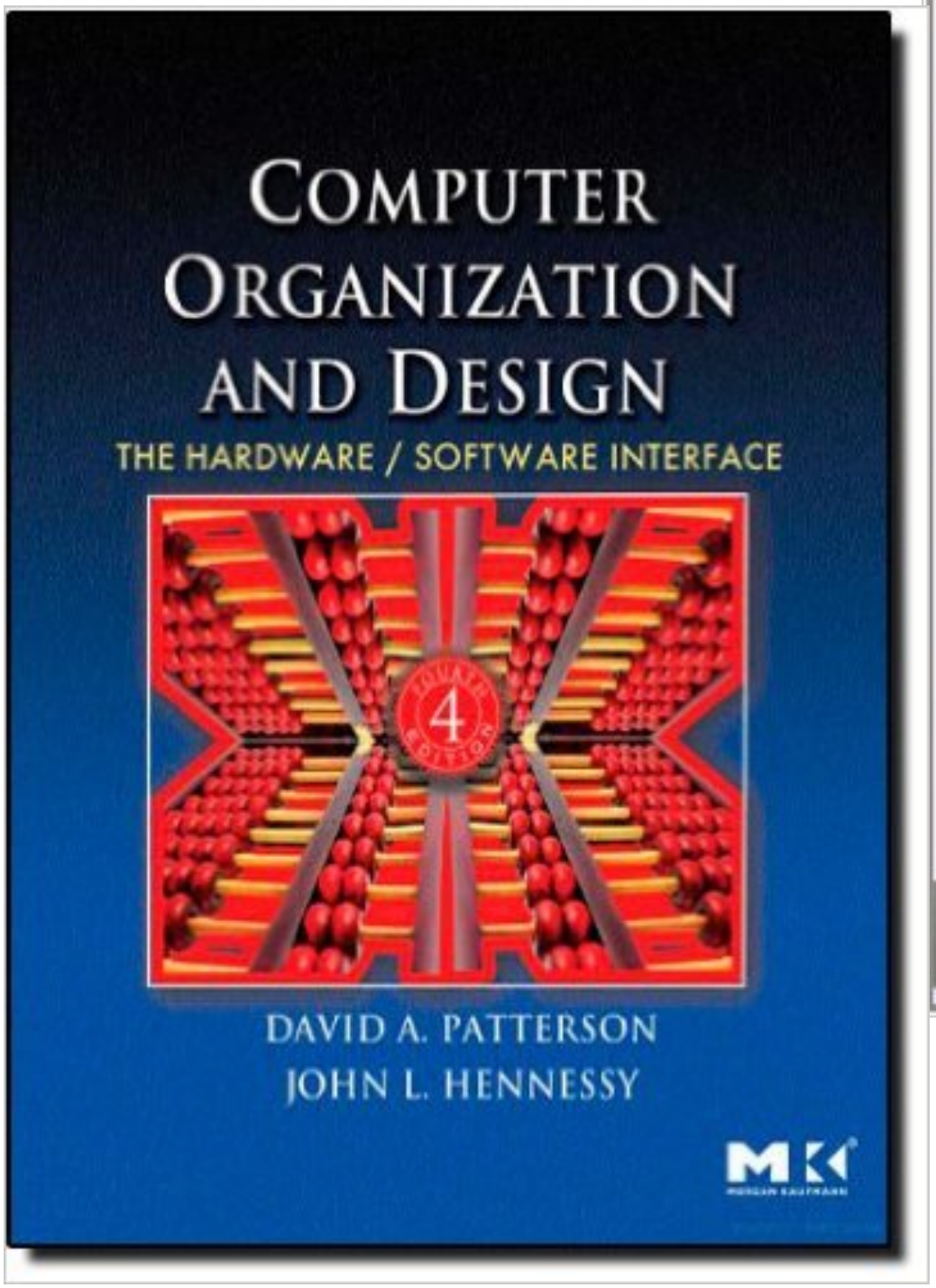
- *Bad news*: "professor, which chapter of Patterson & Hennessy covers Apps ?"

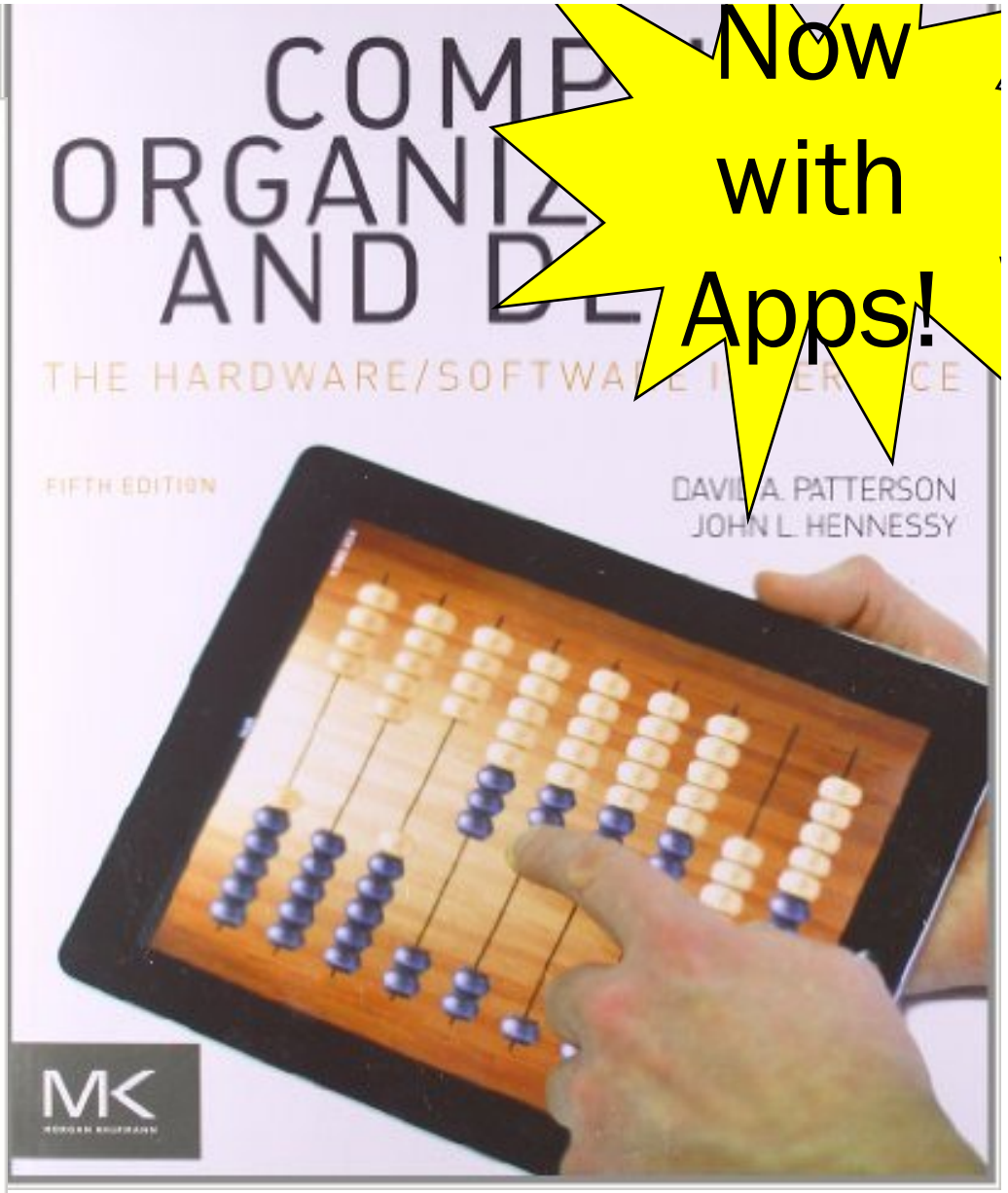
What will the Hardware workforce look like in 15 years?

- *Good news*: enrollment in undergrad Computer Architecture: 30→400

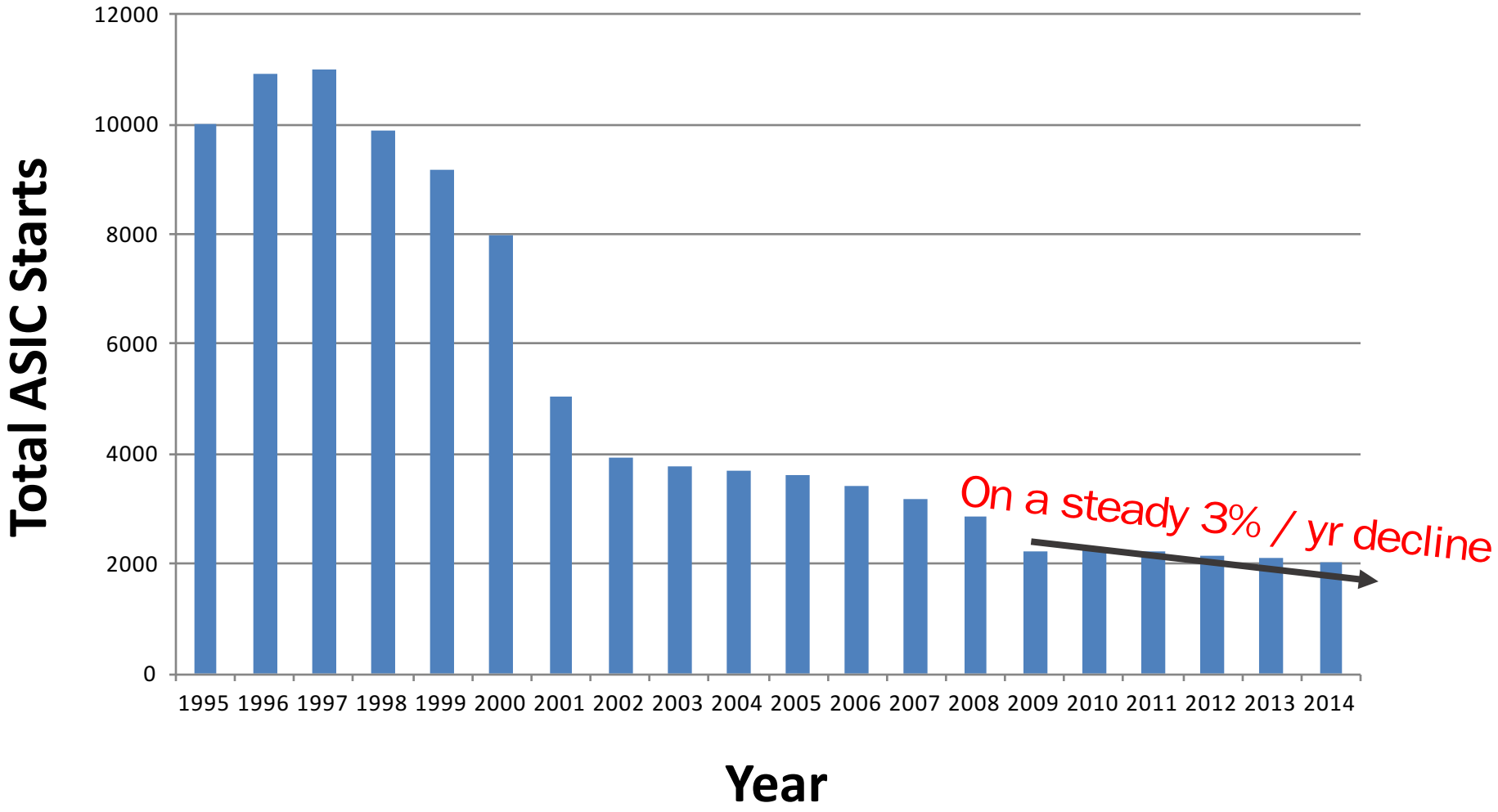
- *Bad news*: "professor, which chapter of Patterson & Hennessy covers Apps ?"

- *Students don't want to design hardware at a stodgy old HW company, they want to start the next Instagram!*
- *Attracting the best talent is a serious problem for the vibrancy of our HW industry*


COMPUTER
ORGANIZATION
AND DESIGN
THE HARDWARE / SOFTWARE INTERFACE
4
DAVID A. PATTERSON
JOHN L. HENNESSY
MK


COMP
ORGANIZ
AND D
THE HARDWARE/SOFTWA
FIFTH EDITION
DAVID A. PATTERSON
JOHN L. HENNESSY
MK
Now
with
Apps!

HW diversity of computing devices is dwindling…

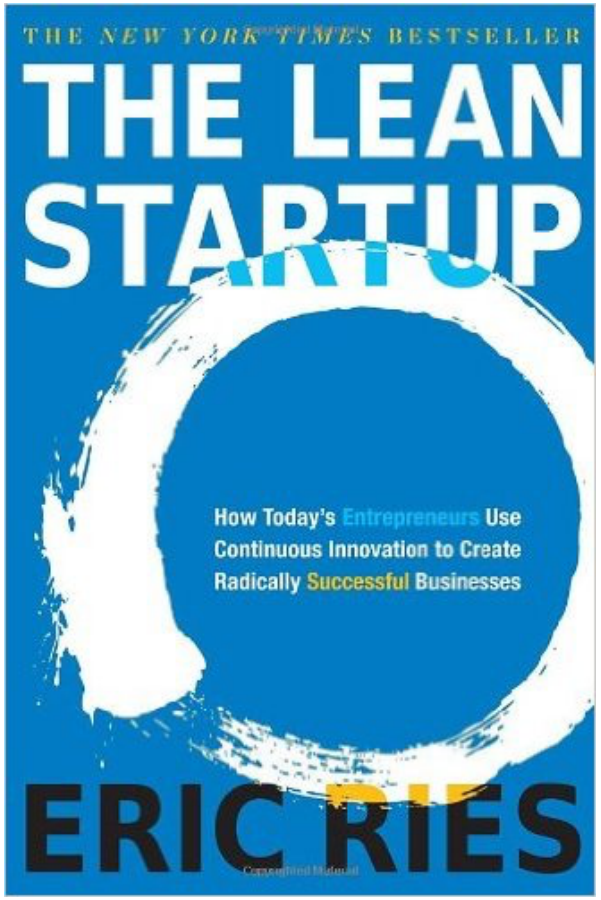


How can open source revitalize the HW field in general?

Source: Gartner Group, T. Austin

Can we make hardware design exponentially leaner so we can have more startups exploring more ideas?

Can we get to a “Minimum Viable Product*” with a few people years of effort?

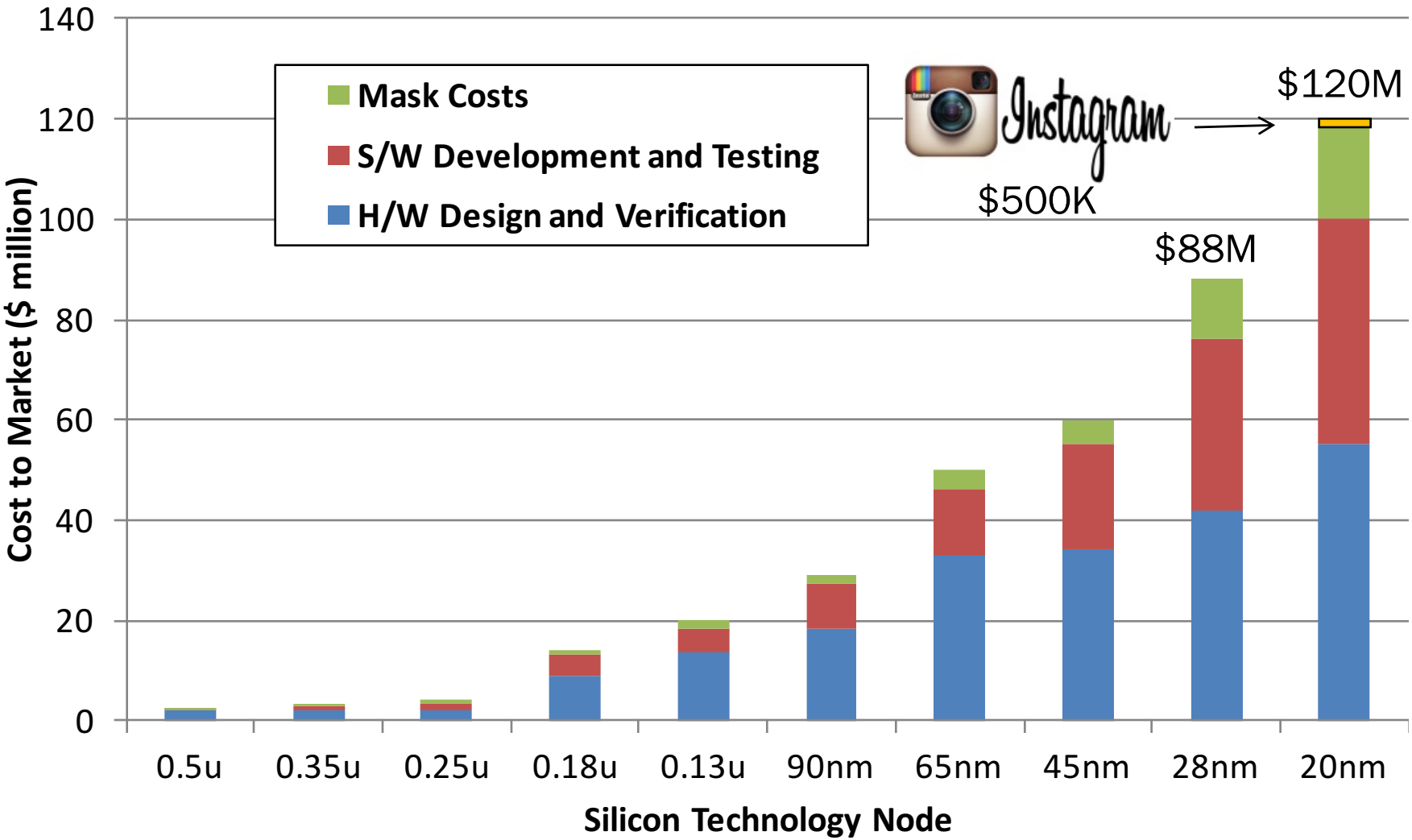


Under the Pillow of Our CS Undergrads…

Is it possible?

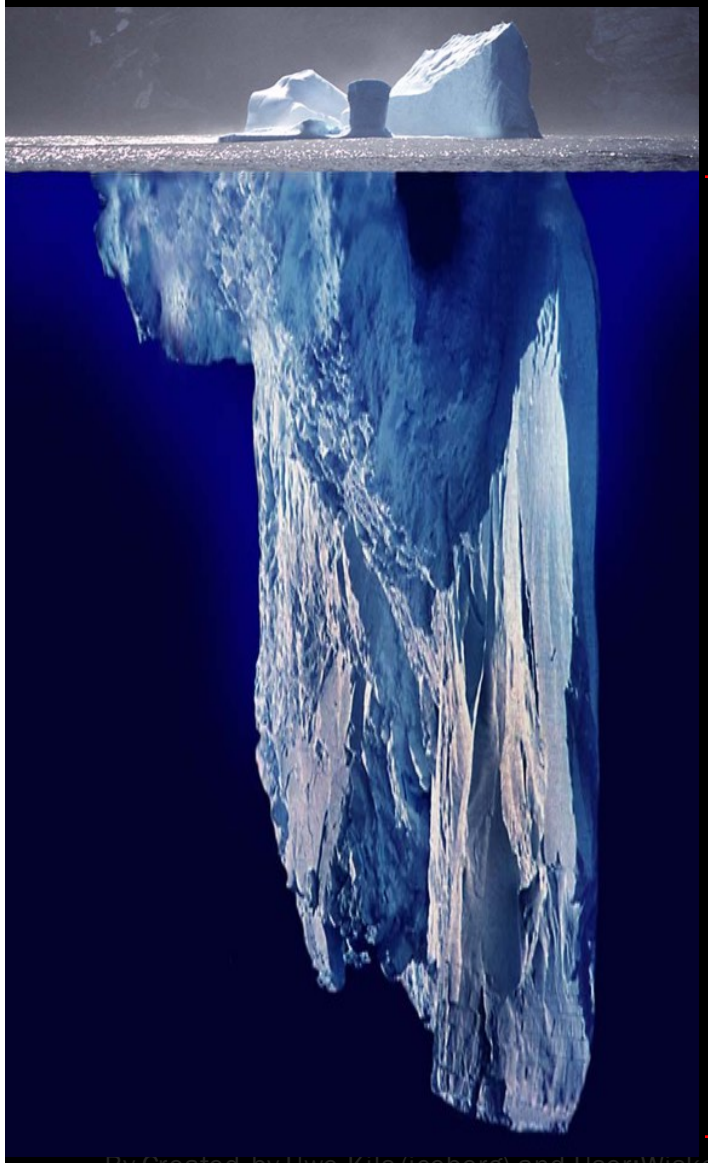
- Most basic version of your product that customers actually pay for/use or in terms of research, show a ”real” design

Costs of Latest Nodes Are Skyrocketing



Source: International Business Strategies, T. Austin

Software Innovation Today



Instagram

Proprietary Code
500K-->13 people & $1B

Open Source

Python

Django

Memcached

Postgres/SQL

Redis

Apache

Linux

GNU *

GCC

Hardware: Where is the Open Source?

Your Secret Sauce

Closed Source ($$)

ARM A57, A7, M4, M0...

ARM Interconnect

IO Pads

Standard Cells

DDR Phy

VCS

Design Compiler

IC Compiler

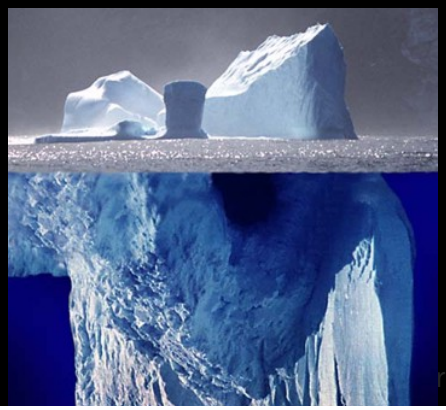
Spice

Formality

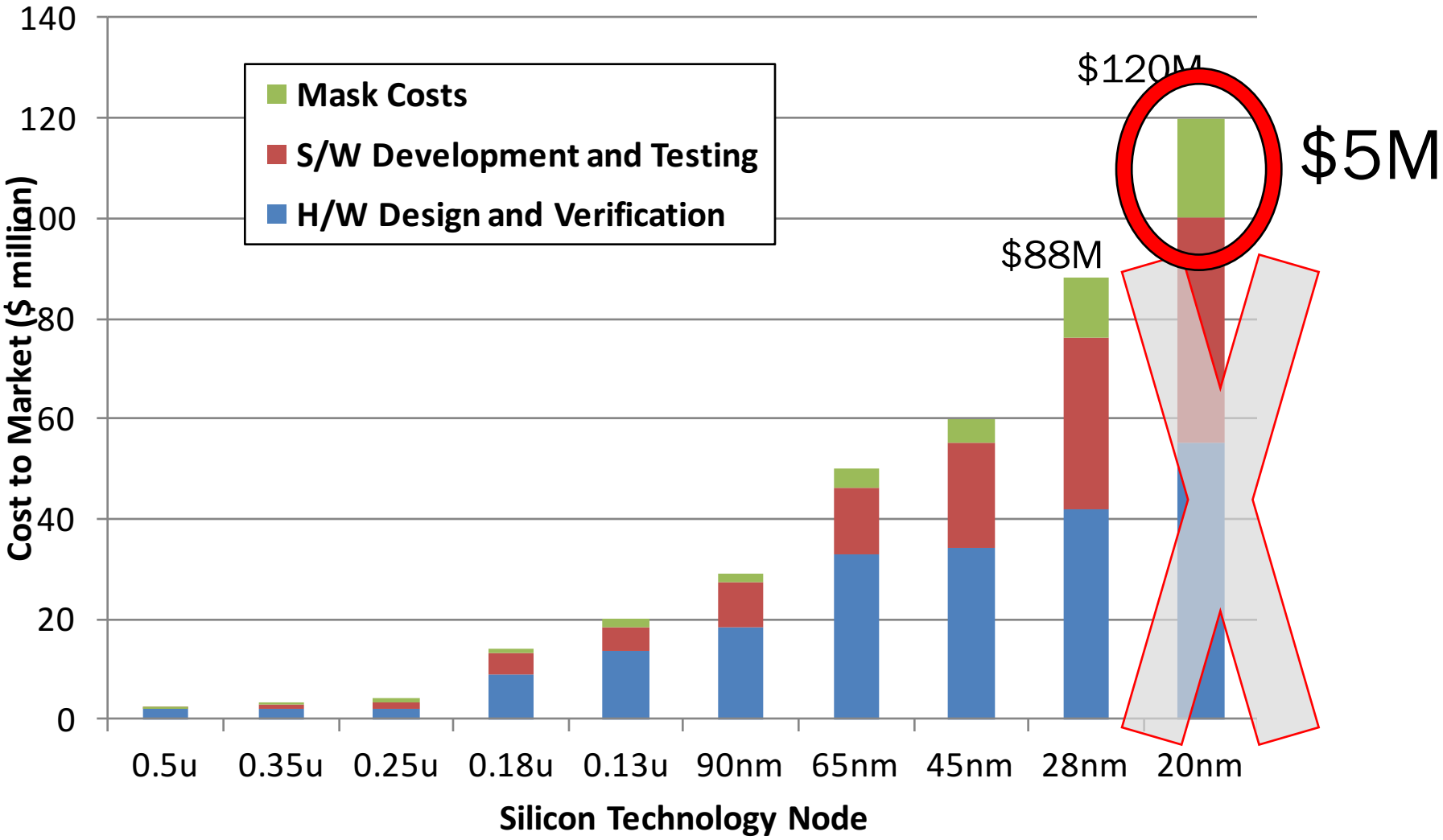
Calibre DRC/LVS

Open Source

Instagram

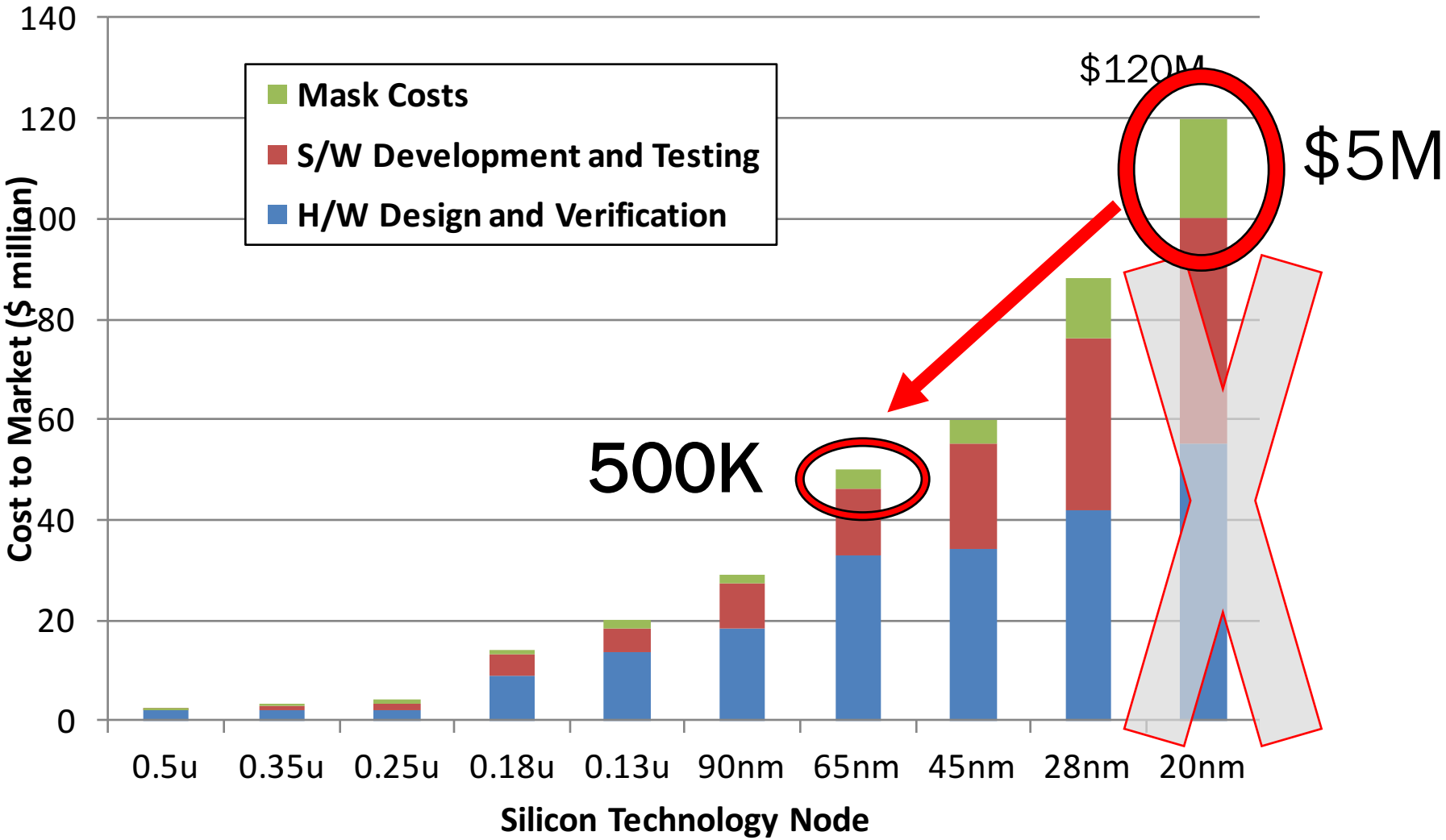


From $120M to $5M: Open Source Can Address most of the Cost



Source: International Business Strategies, T. Austin

And going back a few nodes can get us from $5M to $500K for a 4X perf. Penalty (post-Dennard scaling)



Source: International Business Strategies, T. Austin

How can Hardware Design Be More Like Software?

- Open source infrastructure allows us to create systems where we may only have to write 5% of the total code to create an entirely new product. → Leverage, not labor (and not IP $$$)
- Open source Languages and Libraries so we don't have to redesign everything every time. (like STL or Python or Java Libraries)
- Reduce the overhead of creating + testing new designs
  - Open Source CAD, Open Source Packages, Open Source Standard Cells, Open Source Testboards, NO NDA's.
- IAAS clouds allow us to scale quickly from small companies to large ones from 1 customer to 1 billion customers → Scaling ideas from the small to the big

The Open Source HW Vision

Think GNU/Linux, but for everything HW related:

Open Source CAD Tools (Like GNU)
- VLSI HLS, RTL to GDS …
- PCB Design and Simulation Tools

Open Source Chip Designs (Like Linux)
- Out-of-order
- In-order
- GPU
- FPGA

Open Source IP
- *PLLs, I/O, Standard Cells, DRAM Controllers…*

Emerging open source projects

Processors

- **ISA:** RISC-V
- **In-order:** Rocket, Pulpino, Leon3, OpenRISC
- **OOO Superscalar:** Boom, Fabscalar
- **GPU:** MIAOW, GPLGPU, Nyuzi
- **Manycore:** OpenPiton
- **Microcontroller:** OpenMSP430

CAD Tools (imagine if Linux did not have GCC)

- **Verilog to GDS:** Qflow
- **Verilog to Gate Level:** Yosys
- **Languages:** Chisel, PyMTL, myHDL, ...
- **FreePDK15:** Standard Cells

Motherboards

- **Commercial:** Facebook OpenCompute
- **Prototyping:** UCSD Basejump

But who will do this work?

We need people who:

- are idealistic
- have lots of free time
- will work for free

Who might that be?

But who will do this work?

We need people who:

- are idealistic
- have lots of free time
- will work for free

Who might that be?

Students! (Remember Linus Torvalds?)

An Experiment: CSE 190

CSE 190: The Open Source Hardware Movement with Prof. Michael Taylor:

The open source software movement has blossomed over the last 30 years, and is directly responsible for the current surge in the software industry, where developers can create large startups in which only 5% of the source base is their own code.

Recently, the open source hardware movement has been rapidly gaining ground. In this class, we will study the development of the movement, including progress in open-source processors (RISC-V), open-source GPUs (MIAOW), open-source FPGAs, and open-source libraries (opencores.org). In this class we will brainstorm about this movement, and students will engage in an open source hardware project of their choice to advance the state-of-the-art in open source hardware development. Prerequisites: A+ or A or A- in CSE 141L or ECE 11, or excellent knowledge of SystemVerilog, or Permission of Instructor.

CSE 190

First month of class has students presenting on various open source projects and estimating their important and trajectory.

Students then work in teams. To get an A, they needed to have changes accepted to an Open Source Hardware project. (“To GIT you must commit!”)

Teaching

Search GitHub Pull requests Issues Gist

Steven Okai
s-okai

sokai@eng.ucsd.edu
https://www.linkedin.com/in/ste…
Joined on Oct 26, 2013

4	12	3
Followers	Starred	Following

Overview Repositories Public activity Follow Block or report

Popular repositories

- **riscv-boom** — Berkeley Out-of-Order Machine — 0 ★
- **riscv-boom-doc** — Documentation for the BOOM processor — 0 ★
- **rocket-chip** — Rocket Chip Generator — 0 ★

5 contributions in the last year

Jun Jul Aug Sep Oct Nov Dec Jan Feb Mar Apr May

M
W
F

Summary of pull requests, issues opened, and commits. Learn how we count contributions.

Less More

Research

Have your funded students use and commit to open source HW efforts during their research
... instead of "rolling your own" or using your own proprietary stuff (e.g. Raw)

Search GitHub Pull requests Issues Gist

Anuj Rao
anujnr

anr044@ucsd.edu
https://www.linkedin.com/in/anu...
Joined on Nov 11, 2015

0	7	5
Followers	Starred	Following

Overview Repositories Public activity Follow Block or report

Popular repositories

rocc-template 0 ★

trace-debug-submodule 0 ★

13 contributions in the last year

Jun Jul Aug Sep Oct Nov Dec Jan Feb Mar Apr May

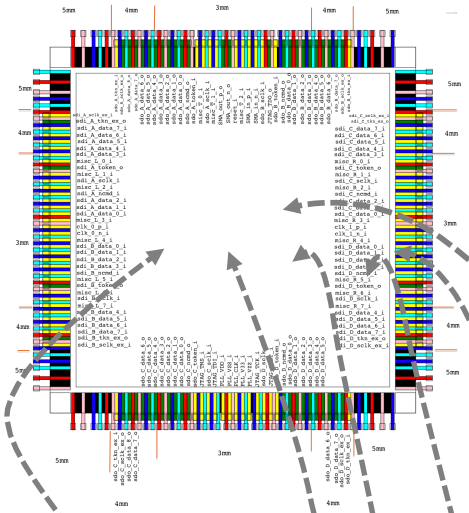
M
W
F

Summary of pull requests, issues opened, and commits. Learn how we count contributions. Less More

Contribution activity Period: 1 week

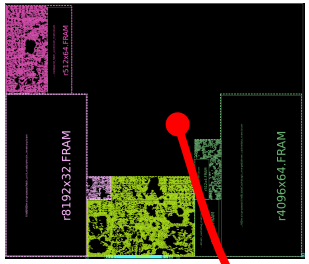
Basejump: A "Base Class" for Open Source HW

Basejump Skeleton

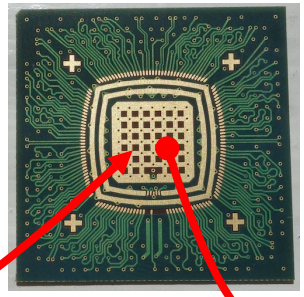


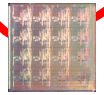
Basejump TSMC 28nm Flow

Your chip



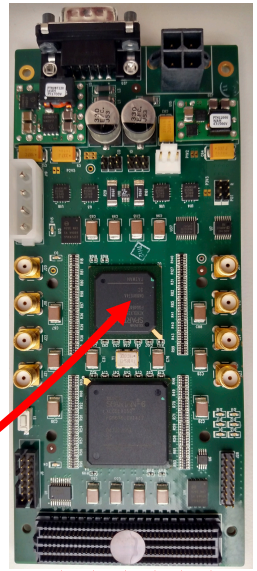
Basejump BGA



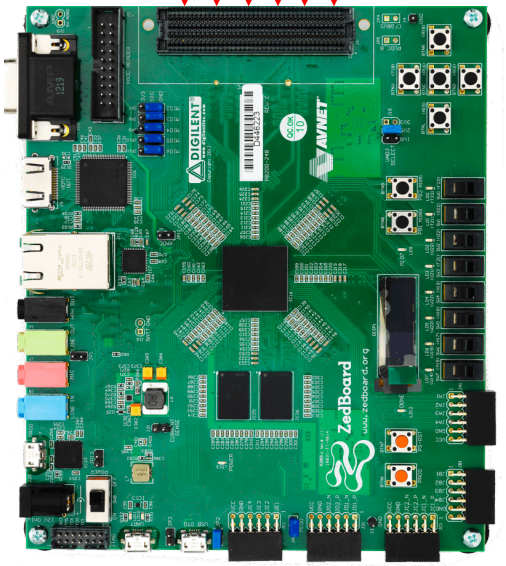


eSilicon

Basejump Motherboard



Click!!



Xilinx Zedboard

OpenCores
www.opencores.org

RISC-V

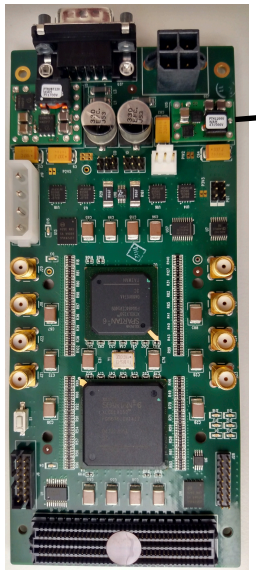
IAOW

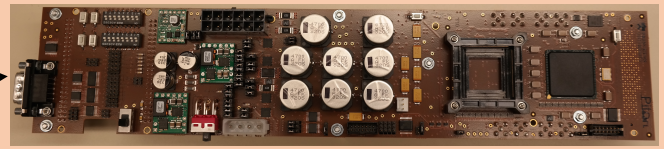
Basejump IP Cores (Clockgen, PLL, IO, ...)

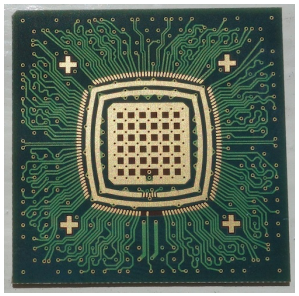
C-core generator (UCSD)

Your Application

Basejump: Early Adopters







PRINCETON UNIVERSITY

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Massachusetts Institute of Technology

UNIVERSITY OF CAMBRIDGE

UCSB

UCSD
NSF SaTC Med (Security)

UCSD

Ucla

UNIVERSITY OF MICHIGAN

CORNELL UNIVERSITY FOUNDED A.D. 1865

DARPA CRAFT (16nm)

UNIVERSITY of VIRGINIA

NYU TANDON SCHOOL OF ENGINEERING

UCSD
NSF SaTC Large (Crypto)

Thanks!