Life Post Moore’s Law: The New CAD Frontier

Mark Horowitz
Stanford University
The Technology Landscape Is Changing

- For over half a century
  - Information technology has been driven by technology scaling

- That scaling made computation cheaper
  - We have grown to expect that trend will continue

- Unfortunately, that scaling is now broken
  - Which has a great effect on future systems, and future system design
This Expectation Is Pervasive

- Making algorithms more complex is OK
  - In tools or applications

- Since the future computer can handle it
  - In fact need the complexity
    - To take advantage of future hardware
Driver of This Expectation: Moore’s Law

From Electronics, Volume 38, Number 8, April 19, 1965

Stanford University
Average Transistor Cost ($) – Moore’s Law

No Exponential is Forever... but We Can Delay 'Forever', Moore ISSCC 2002
When Transistor Cost Scaled

- Making the same product in the new technology was cheaper
  - You *always* moved high-volume products to newest technology
    - You make *more* money that way

- All high-volume (or growing volume) parts
  - Were in the latest technology
  - All fab development was in the advanced generation

- Notice that this isn’t happening anymore
MOORE’S LAW HAS ENDED
Transistor Cost Scaling Has Stopped

https://eri-summit.darpa.mil/docs/Mollenkopf_Steve_Qualcomm_Final.pdf

Two-year cadence
One-year cadence
<table>
<thead>
<tr>
<th>TSMC Wafers</th>
<th>Pricing</th>
<th>Increase in Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>7nm FinFET Wafer</td>
<td>$10,000 USD</td>
<td>2x increase from 8nm to 7nm</td>
</tr>
<tr>
<td>5nm FinFET Wafer</td>
<td>$16,000 USD</td>
<td>1.8x increase from 7nm to 5nm</td>
</tr>
<tr>
<td>3nm FinFET Wafer</td>
<td>$20,000 USD</td>
<td>1.3 increase from 5nm to 3nm</td>
</tr>
</tbody>
</table>

https://www.siliconexpert.com/blog/tsmc-3nm-wafer/

Stanford University
Technology Scaling Of 2x Over Time

- **Moore + Dennard Scaling**
  - 4x the number of functions/$ (also mm²)
  - Gates get 2x faster, Energy/op decreases 8x (W/mm² stays constant)
    - Actually we never really did that
    - Increased gate speed faster, and power went up

- **Moore**
  - 4x the number of functions/$ (also mm²)
  - Gates get a little faster, but mostly lower Vdd to keep power in check
  - Energy/gate scales by 2x, but power/ mm² scales by 2x
Scaling 2x Today

- Technology numbers are really a marketing label
  - Features are not scaling at this rate

- Transistors get more expensive (initially)

- Energy scales down a little
Consumers Expect Cost Scaling of Computation

- And the technology is no longer giving that to you
  - So you have a problem!

- Transistors are no-longer free
  - Need to use the ones we are paying for

- Moving of your products to latest technology doesn’t make sense
  - Even moving all of your hottest product might not make sense
Need to Increase Efficiency

- Efficiency generally implies specialization
  - Need to generate more product SKUs
  - More SKUs imply smaller market/SKU

- Need to decrease NRE/SKU

- Need to optimize $/function
  - Different technologies for different parts of the “chip”

- Did someone say chiplets?
CHIPLETS ARE NOT THE ANSWER
Types of Chiplets

© WikiChip

Stanford University
Chiplets

- Are generally mounted on a interposer
  - This surface has very high interconnect density
  - Initially it was made of silicon ($$)

- Allow different parts of the system to use different technology
  - AMD keep the I/O in an older technology
    - Less NRE, and cheaper, since I/O doesn’t scale well
  - Can add other “interesting” technology as well
    - Can you say photonics?

- But it increases the total cost of the system on a per transistor basis!
HOUSTON, WE HAVE A PROBLEM
Who Is Going To Do This Application Optimization?
We Are Burdened By Our Own Success
Complex Systems Are Expensive To Design

Chip Design and Manufacturing Cost under Different Process Nodes: Data Source from IBS
Leads To Industry Consolidation

Consolidation in the semiconductor industry

160 COMPANIES  97 COMPANIES

10 Years Ago  Today

And Lower Student Interest In Hardware
Who Is Going To Innovate?

https://adct.org.za/let-a-thousand-flowers-bloom/
Paradox

- Application Optimization
  - Requires radical thinking

- Most radical thinking
  - Requires fresh workers
  - Doesn’t work
A Killer of Innovation …

Chip Design and Manufacturing Cost under Different Process Nodes: Data Source from IBS
Only Approach (I can think of):

- Make task exciting
  - Bring in new people

- Make task cheap
  - Possible for small teams to accomplish
THE ADVANTAGE OF BEING OLD
I Have Seen This Rodeo Before
ASIC Design

Enabled Logic Designers

To build chips
Create Many New CAD Tools

Std Cells

Placers

Routers

Synthesis
To Create A New Market!

- Optimal Solutions a.k.a
  - Started as a logic optimization company
    - Netlist to better netlist
  - When Verilog was a simulation language
  - Place and route was what you did on boards

- Tools were created for logic designers
  - Not chip designers
ASIC Chip Results Were Not Optimal

- No “chip designer” would use these tools
  - I know because I was one

- But the results were much better than the board designs
  - So they were good enough
Within 10 Years

- Created a vibrant new industry
  - Which drove many innovations

- Invented fabless semiconductor companies

- Tools improved
  - Which killed off custom design
IF WE WANT A REVOLUTION
To Create A New Market

Need to answer 4 questions:

- **Who** are the new designers?
- **What** abstractions do they use now?
- **Why** now?
- **How** to enable them to do design?
Who

- Need hardware / software co-design

Application Designers

- Yes, software people
What

- No knowledge of hardware
- Some interest in performance
  - Understanding of parallelism, locality
  - Performance tuning
Why

- Moore’s Law is dead
  - Their application is not improving with time

- If they need better performance/power/cost
  - They will need to do something
  - At some point just optimizing the code will be very hard

- Hardware/software co-design will be an interesting option
THE NEW CAD CHALLENGE (HOW)

APPS STORE FOR HARDWARE
Critical Insight: Why An Apps Store?

- Users are creating an application on a system, not the system
One Can’t Build This Cheaply
First Requirement – Base System(s)

Hardware + Software
+ APIs
App Store Other Advantages:

- Creates an open interface for everyone to use
  - While maintaining a proprietary (and revenue generating) platform

- Creates a zero support interface
  - If the interface doesn’t work for you, it is your problem
  - Remember the expected ROI on each design is negative

- But the creation and maintenance costs can be significant
CAD Problems

- Mapping application to hardware

- Scheduling
  - A.K.A. Design Space Exploration

- Defining clean API
  - Abstracts many hardware issues
  - Creates efficient implementations
    - And tools to create implementations from these abstractions

- Validation / Debugging
A Non Goal

- Help current chip designers
  - Working on building chips for billion dollar markets

- Initial tools are always not perfect
  - And to make it accessible and easier
  - It must work in a restricted space

- It may/will take over the world later
  - When the tools mature
Software-Hardware Optimization

- Application designer directed design space exploration
  - Need tools to evaluate performance
  - And tools to suggest possible program transformations

- Goal:
  - Reduce work and/or improve locality and parallelism

- By:
  - Tiling / Dependency breaking / Pipelining / Memoization / Data duplication
Converting Application Code to Hardware

- High-level synthesis

```cpp
int mult_add(int a[4], int b[4] {  
    int t = 0
    for(int i=0; i < 4; i++)
        t += a[i]*b[i];
    return t;
}
```
We Rarely Want The Application In Hardware

- Really want an engine optimized for this type of application
  - Which would be effective as I tune my application

- This tool needs to create a hardware/software combination
  - Hardware engine
  - Software tool which maps application to the hardware engine

- Want to be able to evolve both platforms.
API Support – A Playground For New Tools

- Power
  - States, gating, thermal throttling, Vdd
  - Data retention policies

- Initialization
  - Boot, boot ordering, redundancy
  - Power supply ordering

- Clocking
  - DVFS/power states/supply events

- Security
  - Level of paranoia
API Support, cont’d

- What are the abstractions
  › How to make them orthogonal

- Driver generation for the generated hardware
  › How to make the hardware software plumbing efficient

- Control and data transport
  › Including hardware generation in accelerator
Validation & Debugging

- Another advantage of the App Store framework
  - You are checking that your application works
    - Not that the hardware is perfect!
  - You are not building hardware that everyone will use

- But you are connecting it to a complex system
  - You will misunderstand the specs
    - And the system won’t work

- Source level debugging is critical
  - What does that look like for hardware?
Please Help Me Make This Happen

- Good news / bad news:

  It is a 0 billion dollar market now ...
A Possible Path; Need You Help

- Use some Chips money to bootstrap this effort
- I’m talking with vendors to get a base platform
- We need to create a community that creates these tools
  - Experiment with different API approaches
  - And different tool implementations