**NYU Tandon School of Engineering**

**Fall 2021, ECE 6913**

**Homework Assignment 8**

*Instructor: Azeez Bhavnagarwala,* email: ajb20@nyu.edu

*Course Assistant Office Hour Schedule (Room 808, 370 Jay St: 9AM – 11AM)*

*Mondays & Tuesdays:* Haotian (Kenny) Zheng **hz2687@nyu.edu &** Shan Hao **sh6206@nyu.edu**,

*Wednesdays:* Karan Parikh **kap9580@nyu.edu**

*Thursday:* Sahil Chitnis **ssc9983@nyu.edu**

*Fridays:*Kewal Jani **kj2062@nyu.edu**

*Saturday:* Zhiming Fan **zf2035@nyu.edu**

[released Friday December 3rd 2021] [due\* **Friday December 10th 2021, *before* 11:55 PM**]

You *are allowed* to discuss HW assignments only with other colleagues taking the class. You are *not allowed* to share your solutions with other colleagues in the class. Please feel free to reach out to the Instructor during office hours or by appointment if you need any help with the HW.

Please enter your responses in this Word document after you download it from NYU Classes. *Please use the NYU Classes portal to upload your completed HW. Please do not upload images of handwritten sheets or PDFs of scanned sheets of handwritten solutions. Please be sure to type-in your solutions into Word or Google Docs and upload machine readable documents only*.

1. Please read the ISSCC 2014 Keynote Publication by Professor Mark Horowitz *“Computing’s*

*Energy Problem (and what we can do about it)”* [1]

* 1. How does Technology Scaling decrease the cost of Computing? How do reductions in the cost of manufacturing a transistor enable widespread use of computing devices?

Modern CMOS technology allows us to create chips with millions of transistors at nearly no cost, due to MOS transistor scaling and feature size scaling following a Moore’s law exponential growth, mainly following Dennard scaling up until the 2000s. During Dennard scaling, we continued to scale up clock frequencies faster than dictated by constant-field scaling while voltage is constantly scaled down. For this reason, CMOS cost drops twice every 2 years for the same function allowing massive increase in computing sales, thus making computing more available and eventually these devices became in widespread use.

* 1. Why did scaling processor clock frequency become more difficult in the last 15 years? How did Power dissipation become the primary constraint on server CPU performance?

Shortly after 2000, the rapid scaling of clock frequency slowed down due to processors hitting the power wall for air cooling as well as the slowdown of voltage scaling, due to rising leakage currents. Power is , and C scales with technology, so power dissipation eventually became a constraint when designers ran out of power saving techniques, after DVFS was created to optimize power use.

* 1. Why is Moore’s Law slowing down? Why did Dennard Scaling end?

Moore’s Law slowed down due to the ending of Dennard Scaling where rapid clock frequency is no longer possible when operating voltages could not be scaled proportionately with transistor geometry. Designers move on to DVFS, 3-D CMOS topologies, parallelism, and CPU cache structure to increase CPU performance while keeping power consumption constant. In the power limited world in which we now live, increasing performance means we need to decrease the energy/operation to keep the total power constant.

* 1. Why is the energy consumption by Memory substantial ?

DRAM uses a very energy-inefficient I/O which takes over 20 pJ/bit and require static power to keep the I/O active, which requests and data must travel a large distance on the processor and memory chips to reach.

* 1. What solutions to Computing’s Energy Problem does Professor Mark Horowitz’s envision?

1. Increased hardware specialization in an application-oriented fashion, i.e. development of more application accelerators and codecs on the CPU. More exploitation of data locality for energy efficiency.
2. Enabling of a larger group of application experts to participate in creating the efficient hardware/software systems that they require, as codifying the principles of efficient hardware implementations seems feasible. We need tools that allow application experts access to technology, but now at the level of building board-level systems from existing parts.
3. This assignment requires you to review several references on RISCV beginning with a summary transcript [2] of the Debate on Proprietary Vs Open-Source Instruction Sets at the 4th Workshop on Computer Architecture Research Directions, June 2015 sponsored by the ACM.

This Debate between Professor David Patterson (author of the textbook you are using) and Dave

Christie of AMD highlights all of the key technical and business arguments for and against an Open-Source ISA such as RISC V as of 2015 (the same year the RISC V Foundation was established). A Technical Report from EECS UC Berkeley highlights the technical reasons for Open ISAs [3] providing a more detailed discussion on the advantages offered by open-source ISAs

1. Articulate *your* views on the topics debated in [2]. Justify your views.

While Dave Christie recognizes the standardization, stability, and strong ecosystems that proprietary ISAs provide, the debate topic in question is “Might open ISAs accelerate the innovation of computer hardware?” There is no denying that open-source ISAs, although failed 20 years ago, will spur innovation and reuse due to the emergence of 21st-century architecture and minimal modular ISA, and even the ending of Moore’s Law which shifted designers’ attention from increasing clock speed to other factors such as the optimization of ISAs. Similar to how Apple’s Newton PDA was a commercial failure in 1993, while the iPhone, which took much of its influence from the Newton device, revolutionized the mobile device industry and became the success it is known for today. One of the reasons certain new ventures fail is simply that it is way ahead of its time.

Granted, there are risk to every potential open-source ISA that does not yet have an ecosystem, but that is the nature of every free and open product. It is exactly for this reason, as David Patterson said, open-source ISAs would produce greater innovation because more people would get to design them, not just the engineers at Intel, ARM, and so on. Dave Christie has good points on why commercials ISAs are “good for business” and purchasing ISA licenses make more sense in terms of lowering the risk of failure for a commercial application of software development, that is also true of every other proprietary implementation of technology. These points have no relevance to whether open-source ISAs will accelerate the innovation of computer hardware. Furthermore, limitation of proprietary ISAs have their own problems such as the ARMv8 not having 16-bit instructions, so “the code size is very large, even bigger than x86. As such, running the code will result in higher instruction cache miss rates. To compensate, their current cores have larger instruction caches.”

I agree with David Patterson in that just like an OS or a graphics API, open-source ISAs like RISC-V will eventually develop its own standard and ecosystem. In his words, “Obviously, this is like the Innovator’s Dilemma. It’s incomplete yet very promising, but it doesn’t have everything you need. In this open-source world, we think volunteers will help supply the missing pieces and build that ecosystem.” In this fashion, with the ISA being free and open to everyone who wishes to not only use it but also to share their improvements on it, the rate at which innovation takes place far exceeds that of commercial ISAs where innovation is only competition and market-driven, because like Coca-Cola, there is little incentive to innovate on x86 or ARM if their cash cow status aren’t threatened.

1. Review and summarize technical reasons for Open-Source ISAs in [3].
2. Unlike OSs, ISAs change very slowly, whereas algorithmic innovations and new application demands force continual OS evolution. It is also an interface standard like TCP/IP, thus simpler to maintain and evolve than an OS.
3. Commercial ISAs have many limitations on their technology as IPs are developed behind closed doors and not shared and released to the public for improvement. Open-Source ISAs allow for greater innovation via free-market competition from more designers.
4. Proprietary ISA companies cannot verify ISA compatibility and their ISAs are not guaranteed to last.
5. Transparency from shared open core designs means shorter time to market, lower cost from reuse, and fewer errors.
6. Technical reasons for RISC-V specifically:
   1. The base-plus-extension structure: i) a small core set of instructions that compilers and OS’s can depend upon; ii) standard but optional extensions for common ISA additions to help customize the SoC to the application; and iii) space for entirely new opcodes.
   2. Compact instruction set encoding.
   3. Quadruple-precision (QP) as well as SP and DP floating point.
   4. 128-bit addressing as well as 32-bit and 64-bit.

**References**

1. M Horowitz, *“Computing’s Energy Problem (and what we can do about it)”* Plenary Session

1.1, 2014 ISSCC Digest of Tech. papers, Feb 2014 [*PDF attached*]

1. M Hill et al, “Proprietary Versus Open Instruction Sets” 4th Workshop on Computer Architecture Research Directions, June 2015, ACM [*PDF attached*]
2. K Asanovic et al, “Instruction Sets Should Be Free: The Case For RISC-V”, EECS, University of California Berkeley, Technical Report No. UCB/EECS-2014-146, Aug 6 2014 [*PDF attached*]