Helping *Python*istas Become Microarchitects

Using Jupyter Notebooks and CIRCT/MLIR/LLVM

Zhiyang Ong

Department of Electrical and Computer Engineering
College of Engineering,
Texas A&M University
College Station, TX

December 1, 2023

1. Motivations for Python-based RISC Processor Design
2. Problems in Computer Architecture
3. New Golden Era of Computer Architecture, EDA, and Compiler Design
4. Python-based IC Design

Table of Contents

1. Motivations for Python-based RISC Processor Design
2. Problems in Computer Architecture
3. New Golden Era of Computer Architecture, EDA, and Compiler Design
4. Python-based IC Design

Motivations for Python-based RISC Processor Design

People want to get good pay

- Regardless of location or socioeconomic status
- Jane Street Capital
  - Hardcaml/OCaml for FPGA design
  - **US$100.00/hour, or US$20,000/month**

Machine learning for EDA

- BOOM Explorer: Use deep learning for design space exploration of processor architectures

Easy to attract laypeople to RISC processor design

- And, RISC-V -based System-on-Chip design
- Jump on trends for data science & computational thinking
- Use Python-like HCL/HDL for design + verification (MERL/UITU students/interns using PyUVM)

Table of Contents

1. Motivations for Python-based RISC Processor Design
2. Problems in Computer Architecture
3. New Golden Era of Computer Architecture, EDA, and Compiler Design
4. Python-based IC Design

Problems in Computer Architecture

Specifically with General-Purpose Processor Architectures

Golden Era of Computer Architecture (1980s till early 2000s):

- **Memory Wall** [Wulf1995] [Hennessy1990] [Horowitz2023] [Solihin2002]
- **End of Dennard's scaling** [Dennard1974] [Haensch2006] [Chen2006] [Dennard2007] [Calhoun2008] [Iwai2009] and **Power Wall** [Keshavarzi2007]
- **Dark Silicon** [Esmaeilzadeh2011] [Esmaeilzadeh2012] [Rahmani2017] [Hurson2018]
- **ILP Wall** $\rightarrow$ **limitations of** [Hennessy2019, §1.11, pp. 39]
- **impending doom of Moore's law** [Duranton2019] [Kelleher2022]
- **decline of general-purpose processors** [Thompson2018]
- **Hardware Accelerator Wall** [Fuchs2019]

Problems in Computer Architecture

Specifically with General-Purpose Processor Architectures

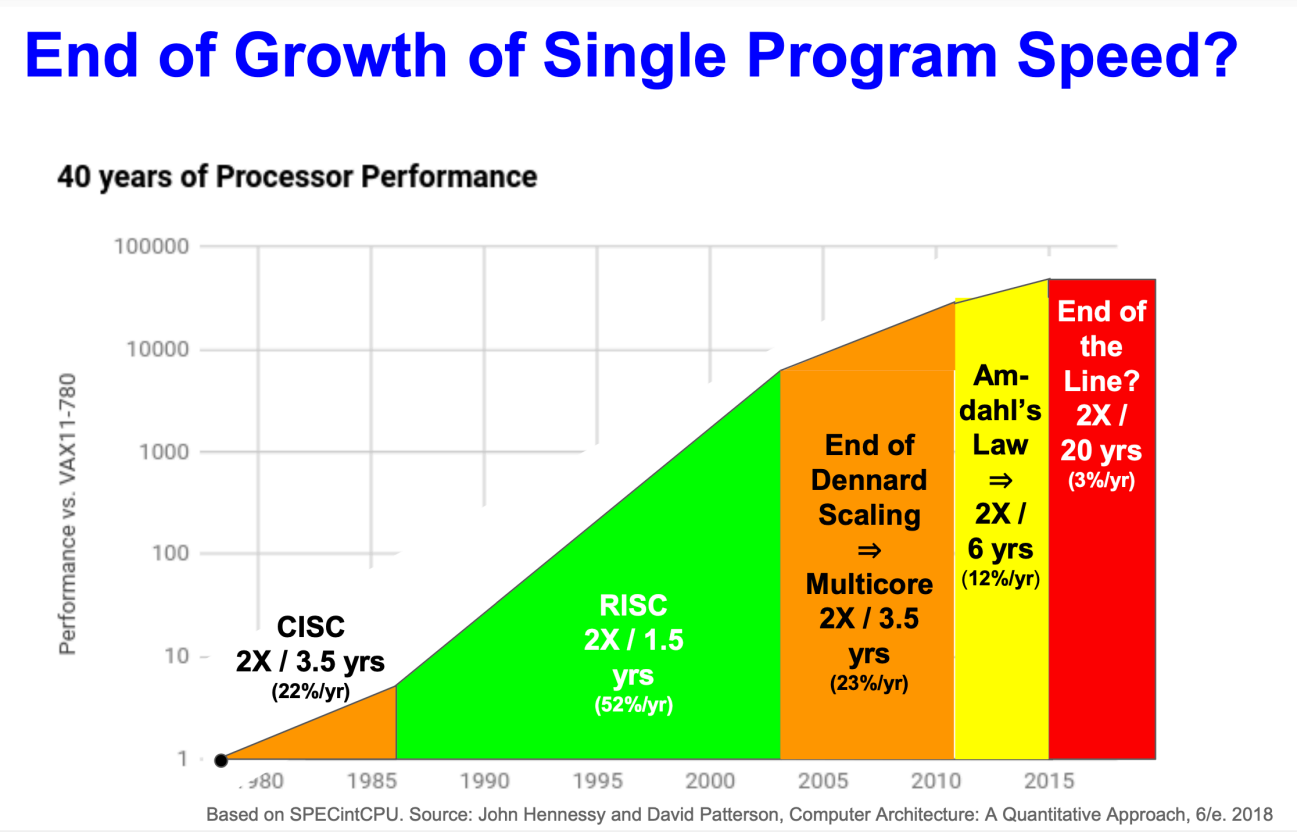

Figure: Plot of the performance of general-purpose processors over time, from 1980 till the late 2010s [Hennessy2018]

Table of Contents

1. Motivations for Python-based RISC Processor Design
2. Problems in Computer Architecture
3. New Golden Era of Computer Architecture, EDA, and Compiler Design
4. Python-based IC Design

New Golden Era of Computer Architecture (1)

And, also for EDA and Compiler Design

Problems → Opportunities [Hennessy2019a]

Domain-Specific Computing [Hennessy2019] → **Heterogeneous System Architectures** [HSAFoundationAdministration2016] [Hwu2016] [Duranton2019]

Hardware Security [Gruss2017] [Szefer2018] [Duranton2021]

Open-Source ISA [Patterson2018b], and support ecosystem across the hardware/software stack

New Golden Era of Computer Architecture (2)

And, also for EDA and Compiler Design

Agile IC Design Methodologies [Gerstlauer2001] [Hennessy2018] [Johnson2018a] + **Python-based IC Design**

Domain-Specific Compilers [Lattner2021a] + Compilers for Heterogeneous Systems

System-Technology Co-Optimization [Wu2021]:

- **system** $\rightarrow$ computer systems $\rightarrow$ hardware/software co-design
- **semiconductor manufacturing technology** (including semiconductor device engineering)

New Golden Era of Computer Architecture (3)

And, also for EDA and Compiler Design

Recent **Non- von Neumann Computing Paradigms and Technology Trends**:

- **in-memory computing** [Zhu2013] [Paul2014] [Williams2017a] [Theis2017] [Imani2020] [Wu2021]
- **hyperdimensional computing** [Imani2020] [Wu2021]
- **photonic ICs** [Topaloglu2015]
- **wafer-scale computing**
- **3-D ICs** and **high-bandwidth memory interconnects**
- **chiplet-based System-in-Package design**

Table of Contents

1. Motivations for Python-based RISC Processor Design
2. Problems in Computer Architecture
3. New Golden Era of Computer Architecture, EDA, and Compiler Design
4. Python-based IC Design

Python-based IC Design: Options

Possible options:

- **MyHDL** (old)
- **PyMTL** (Cornell University)
- **PyRTL** (University of California, Santa Barbara)
- **Jupyter Notebook + Python** -based IC design flow (supported by Google Colab)
- **CIRCT: Circuit IR Compilers and Tools** [Lattner2021]
  - LLVM (initially, Low Level Virtual Machine) [Lopes2014] [Pandey2015] [Sarda2015]
  - Multi-Level Intermediate Representation, MLIR (extension of LLVM ecosystem for domain-specific computing)

Jupyter Notebook + Python -based IC Design Flow (1)

Supported by Google Colab

DSLX

1. Update the `mul4` function below to use the DSLX standard library functions to implement a 4-bit multiplier (don't forget the `std::` prefix).
2. Generate the verilog for the design.
3. Run the OpenLane flow up until synthesis.
4. Observe the change in the complexity of the graph.
5. Compare to the results w/ the previous adder design.

```
[ ]  %%bash -c 'cat > user_module.x; interpreter_main user_module.x'
     import std

     fn mul4(a: u4, b: u4) -> u8 {
       u8:0 // TODO(YOU) implement mul4
     }

     fn user_module(io_in: u8) -> u8 {
       mul4(io_in[0:4], io_in[4:8]) as u8
     }

     #[test]
     fn test() {
       let _ = assert_eq(mul4(u4:8, u4:8), u8:64);
       let _ = assert_eq(user_module(u8:0b1000_1000), u8:0b0100_0000);
       _
     }
```

Figure: DSLX -based Design: Python-like hardware construction language (HCL), or HDL

Jupyter Notebook + Python -based IC Design Flow (2)

Supported by Google Colab

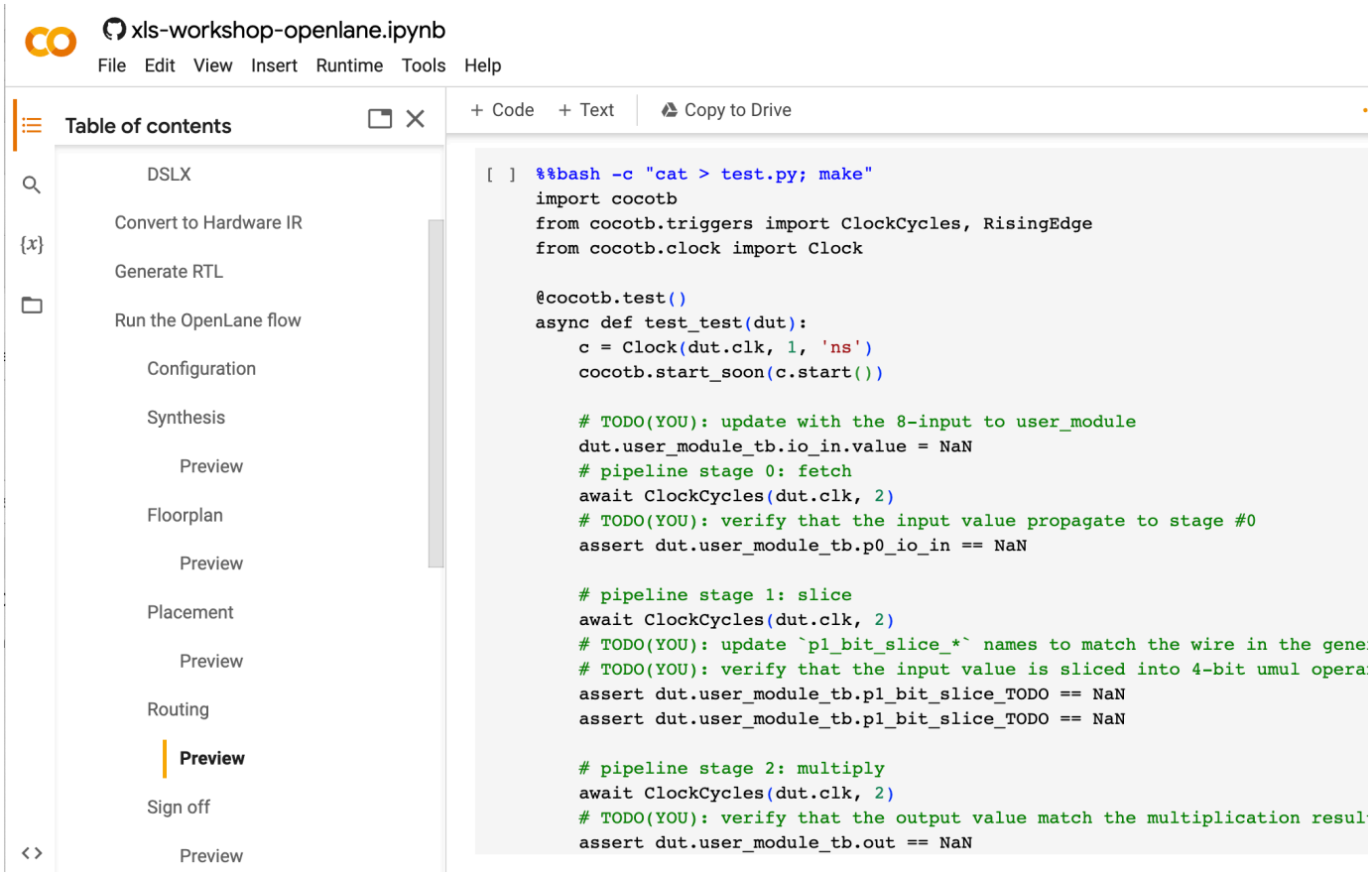

Figure: Python-based VLSI Functional Verification: Using cocotb for HDL-based Simulation... Can use *PyUVM*

CIRCT -based IC Design Flow

Supported by MLIR and LLVM

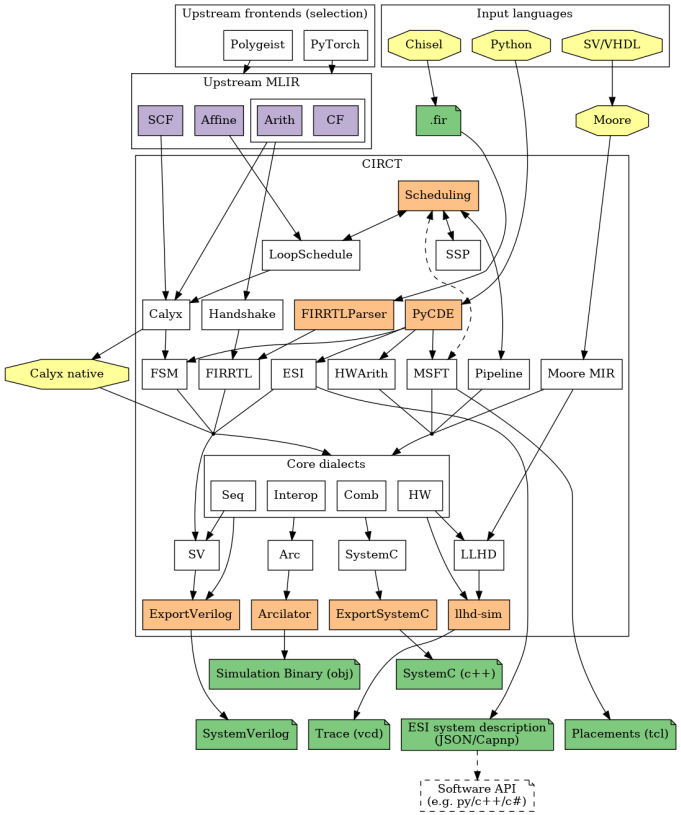

Figure: CIRCT IC design flow + MLIR + LLVM [Lattner2021]

References (1)

[Calhoun2008] Benton H. Calhoun, Yu Cao, Xin Li, Ken Mai, Lawrence T. Pileggi, Rob A. Rutenbar, and Kenneth L. Shepard, "Digital Circuit Design Challenges and Opportunities in the Era of Nanoscale CMOS", *Proceedings of the IEEE*, Volume 96, Number 2, pp. 343–365, IEEE Press, Piscataway, NJ, February, 2008. DOI: https://dx.doi.org/10.1109/JPROC.2007.911072.

[Chen2006] Tze-Chiang Chen, "Overcoming Research Challenges for CMOS Scaling: Industry Directions", *Proceedings of the* 8^{th} *International Conference on Solid-State and Integrated Circuit Technology (ICSICT '06)*, pp. 4–7, IEEE Press, Shanghai, China, October 23–26, 2006. DOI: https://dx.doi.org/10.1109/ICSICT.2006.306040.

References (2)

[Dennard1974] Robert H. Dennard, Fritz H. Gaensslen, Hwa-Nien Yu, V. Leo Rideout, Ernest Bassous, and Andre R. LeBlanc, "Design of ion-implanted MOSFET's with very small physical dimensions", *IEEE Journal of Solid-State Circuits*, Volume 9, Number 5, pp. 256–268, IEEE Press, New York, NY, October, 1974. DOI: https://dx.doi.org/10.1109/JSSC.1974.1050511.

[Dennard2007] Robert H. Dennard, Jin Cai, and Arvind Kumar, "A perspective on today's scaling challenges and possible future directions", *Solid-State Electronics*, Volume 51, Number 4, pp. 518–525, Elsevier Ltd., Oxford, U.K., April, 2007. DOI: https://dx.doi.org/10.1016/j.sse.2007.02.004.

References (3)

[Duranton2019] Marc Duranton, Koen De Bosschere, Bart Coppens, Christian Gamrat, Madeleine Gray, Harm Munk, Emre Ozer, Tullio Vardanega, and Olivier Zendra, "HiPEAC Vision 2019: High Performance and Embedded Architecture and Compilation", Ghent, Belgium, January, 2019. Available online from *HiPEAC: HiPEAC Vision: Archive: 2019 – HiPEAC Vision 2019* at: `https://www.hipeac.net/vision/#/archive/` and `https://www.hipeac.net/vision/2019/`; April 7, 2021 was the last accessed date.

References (4)

[Duranton2021] Marc Duranton, Koen De Bosschere, Bart Coppens, Christian Gamrat, Thomas Hoberg, Harm Munk, Catherine Roderick, Tullio Vardanega, and Olivier Zendra, "HiPEAC Vision 2021: High Performance Embedded Architecture and Compilation", Ghent, Belgium, January, 2021. Available online from *HiPEAC: HiPEAC Vision: HiPEAC Vision 2021* at: `https://www.hipeac.net/vision/#/latest/` and `https://www.hipeac.net/vision/2021/`; April 7, 2021 was the last accessed date.

References (5)

[Esmaeilzadeh2011] Hadi Esmaeilzadeh, Emily Blem, Renée St. Amant, Karthikeyan Sankaralingam, and Doug Burger, "Dark Silicon and the End of Multicore Scaling", *Proceedings of the* 38th *Annual International Symposium on Computer Architecture (ISCA '11)*, pp. 365–376, ACM Press, San Jose, CA, June 4–8, 2011. DOI: `https://dx.doi.org/10.1145/2000064.2000108`.

[Esmaeilzadeh2012] Hadi Esmaeilzadeh, Emily Blem, Renée St. Amant, Karthikeyan Sankaralingam, and Doug Burger, "Dark Silicon and the End of Multicore Scaling", *IEEE Micro*, Volume 32, Number 3, pp. 122–134, IEEE Computer Society Press, Los Alamitos, CA, May–June, 2012. DOI: https://dx.doi.org/10.1109/MM.2012.17.