RISC-V and FPGAs: Open Source Hardware Hacking

Hackaday Superconference 2019-11-16

Megan Wachs, PhD

RISC-V : “The Free and Open ISA”

An ISA is the language a computer speaks

“Instruction Set Architecture”

Defines how 1s and 0s get turned into instructions the computer can execute.

What’s an ISA?

Let’s consider “adding”:

C = A + B

All (?) computers know how to do it…

What's an ISA?

High Level Code (e.g. C)

```
int c = a + b;
```

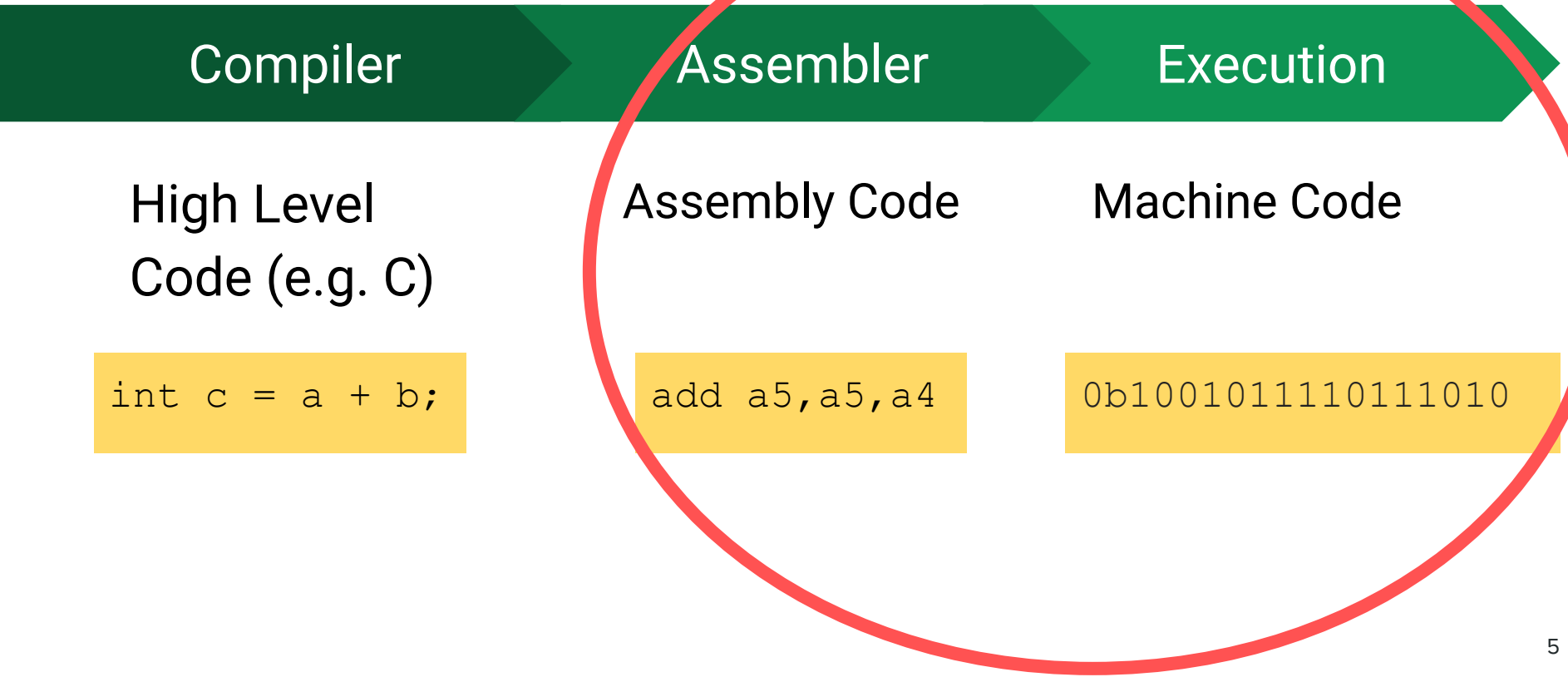
Assembly Code

```
add a5,a5,a4
```

Machine Code

```
0b1001011110111010
```

What's an ISA?


Compiler
Assembler
Execution
High Level Code (e.g. C)
Assembly Code
Machine Code
int c = a + b;
add a5,a5,a4
0b1001011110111010

What's an ISA?

x86 Instruction Set Reference

ADD

Add

Opcode	Mnemonic	Description
04 ib	ADD AL, imm8	Add imm8 to AL
05 iw	ADD AX, imm16	Add imm16 to AX
05 id	ADD EAX, imm32	Add imm32 to EAX
80 /0 ib	ADD r/m8, imm8	Add imm8 to r/m8
81 /0 iw	ADD r/m16, imm16	Add imm16 to r/m16
81 /0 id	ADD r/m32, imm32	Add imm32 to r/m32
83 /0 ib	ADD r/m16, imm8	Add sign-extended imm8 to r/m16
83 /0 ib	ADD r/m32, imm8	Add sign-extended imm8 to r/m32
00 /r	ADD r/m8, r8	Add r8 to r/m8
01 /r	ADD r/m16, r16	Add r16 to r/m16
01 /r	ADD r/m32, r32	Add r32 to r/m32
02 /r	ADD r8, r/m8	Add r/m8 to r8
03 /r	ADD r16, r/m16	Add r/m16 to r16
03 /r	ADD r32, r/m32	Add r/m32 to r32

https://c9x.me/x86/html/file_module_x86_id_5.html

What's an ISA?

ADD

Home » ARM and Thumb Instructions » ADD

10.10 ADD

Add without Carry.

Syntax

```
ADD { S }{ cond } { Rd },  Rn ,  Operand2
ADD { cond } { Rd },  Rn , # imm12  ; Thumb, 32-bit encoding only
```

http://www.keil.com/support/man/docs/armasm/armasm_dom1361289861747.htm

What does an ISA get you?

Software is hard. And annoying to port.

Agreeing on the ISA gets you lots of good software:

- Tools that speak the language.
- Compilers, assemblers, debuggers.
- Visualizers
- Emulators, Simulators
- Documentation, programmers guides, application notes...

An ISA is a standard.

Standards Make a Difference!

Field	*Open Standard*	*Free, Open Implement.*	*Proprietary Implement.*
Networking	Ethernet, TCP/IP	Many	Many
OS	Posix	Linux, FreeBSD	M/S Windows
Compilers	C	gcc, LLVM	Intel icc, ARMcc
Databases	SQL	MySQL, PostgresSQL	Oracle 12C, M/S DB2
Graphics	OpenGL	Mesa3D	M/S DirectX
ISA	**??????**	----------	x86, ARM, IBM360

Companies and their ISAs Come and Go

- Digital Equipment Corporation
  - **PDP-11**, **VAX**, **Alpha**
- Intel
  - **i960**, **i860**, **Itanium**
- **MIPS**
  - Sold to Imagination, then bought by Wave AI startup, now ~~open sourcing~~?
- **SPARC**
  - Was opened by Sun, acquired by Oracle, now closed down
- **ARM**
  - Sold to Softbank at >40% premium
  - Now 25% sold off to Abu Dhabi investment fund

How many ISAs does a chip speak?

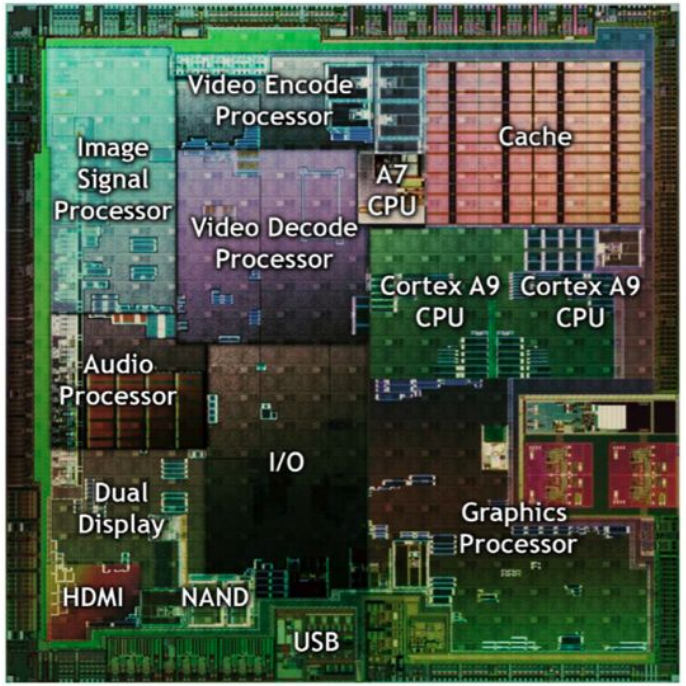
NVIDIA Tegra SoC

By Poeggi - using sample chips with a photo-camera at workPreviously published: n/a, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=24648921 / cropped

SOC: System on a Chip

How many ISAs does a chip speak?

- Applications processor (usually ARM)
- Graphics processors
- Image processors
- Radio DSPs
- Audio DSPs
- Security processors
- Power-management processor
- *> dozen ISAs on some SoCs – each with unique software stack*



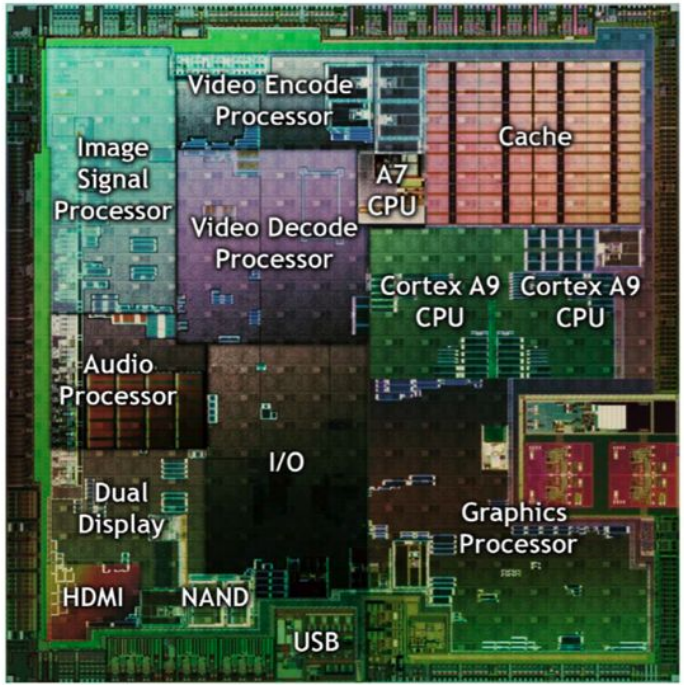
NVIDIA Tegra SoC

SOC: System on a Chip

How many ISAs does a chip speak?

Why?

- Apps processor ISA too big, inflexible for accelerators
- IP bought from different places, each proprietary ISA
- Engineers build home-grown ISA cores



NVIDIA Tegra SoC

SOC: System on a Chip

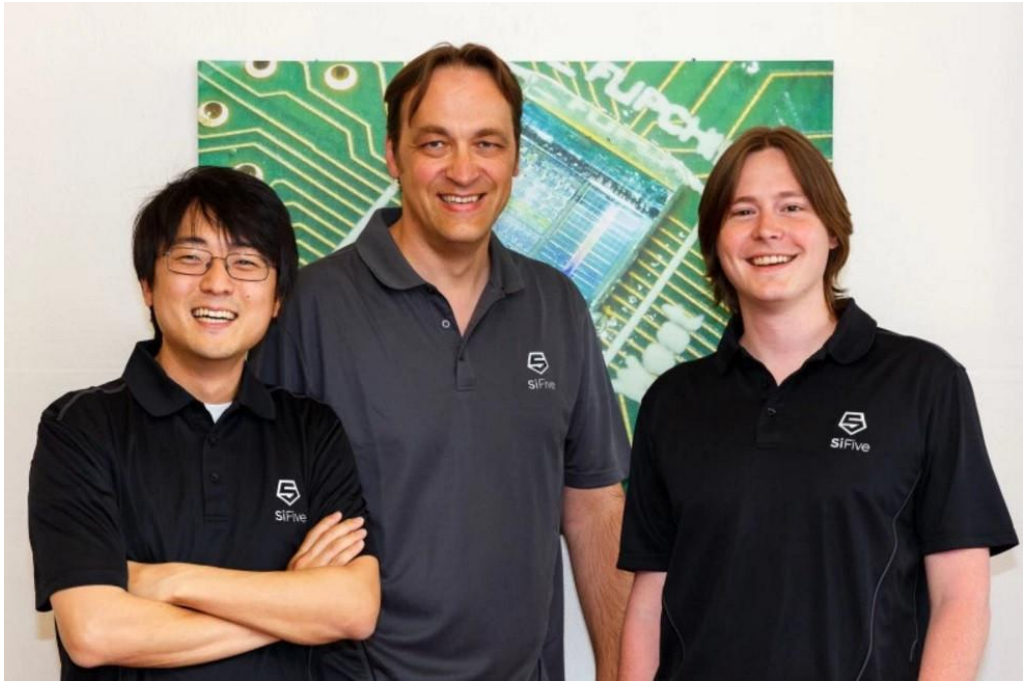
Do we need all these different ISAs?

Must they be proprietary?

Must they keep disappearing?

What if there was one stable free and open ISA everyone could use for everything?

Enter RISC-V



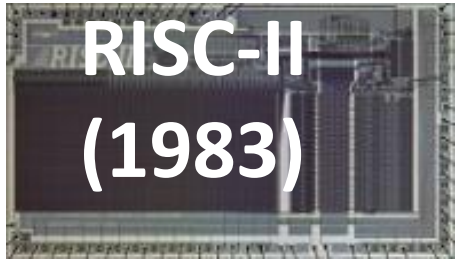
2010:

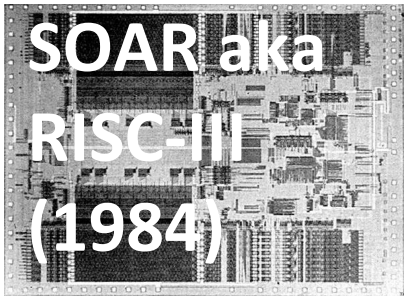
Students Andrew Waterman and Yunsup Lee working with Prof Krste Asonovic and Dave Patterson

Trying to get some cool Computer Architecture research done, but choosing an ISA to use was getting in the way

Enter RISC-V

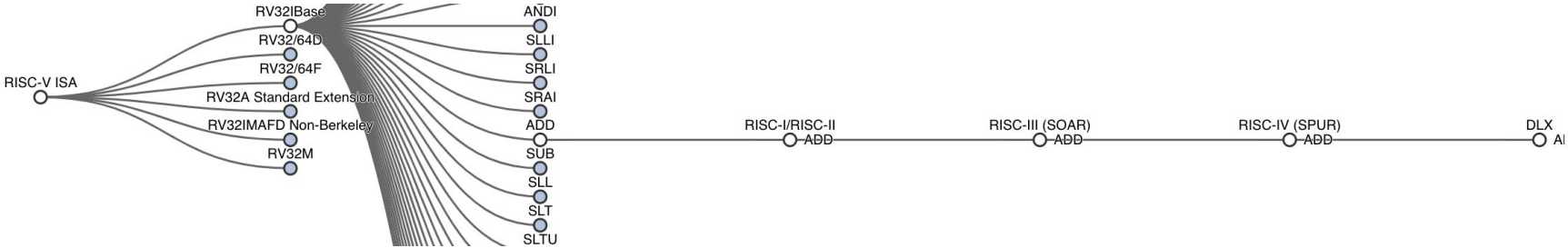
- Options for the Berkeley Team's research?
  - Proprietary options: licensing issues and too complex as a starting point
  - Open source options: limited and tightly coupled to actual designs
- They started "3-month project" to develop clean-slate ISA
- Name RISC-V (pronounced "risk-five") represents fifth major Berkeley RISC ISA





RISC-V Is Boring

- Goes “back to basics”
- Designed to be simple to learn and implement.
- But also makes it easy to add bells and whistles
    - Extensions and customizations assumed from the start



https://riscv.org/risc-v-genealogy/

Then what happened?

Berkeley Researchers developing the ISA were making changes

People outside of Berkeley started complaining… ????

Seems there was lots of demand for a simple, boring, easily extensible ISA.

May 2014, released frozen base user spec

- 4 years after the “3 month summer project” started
- many tapeouts and several research publications along the way

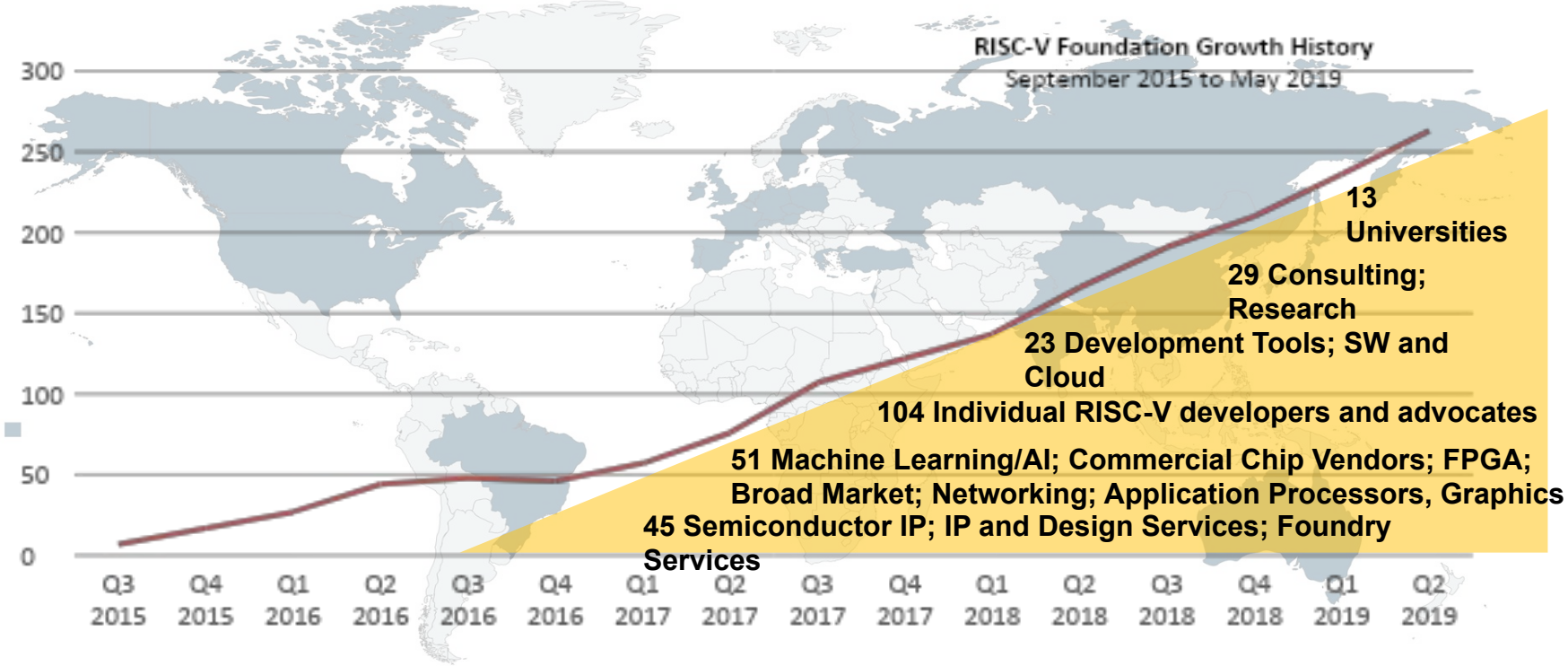
RISC-V Foundation created to maintain it as a free and open standard.

RISC-V Foundation *(2015-)*

- **RISC-V** is the open-source hardware Instruction Set Architecture (ISA)
- Frozen base user spec released in 2014, contributed, ratified, and openly published by the RISC-V Foundation

RISC-V Foundation

More than 250 RISC-V Members in 28 Countries Around the World



RISC-V Ecosystem

Software

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

Commercial software:
Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, Imperas, AntMicro, …

RISC-V Foundation

ISA specification | Golden Model | Compliance

Hardware

Open-source cores:
Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Swerv, Hummingbird, …

Commercial core providers:
Andes, Bluespec, Cloudbear, Codasip, Cortus, C-Sky, Nuclei, SiFive, Syntacore, …

Inhouse cores:
Nvidia, WDC, Alibaba, ...

Reminder: What is an ISA?

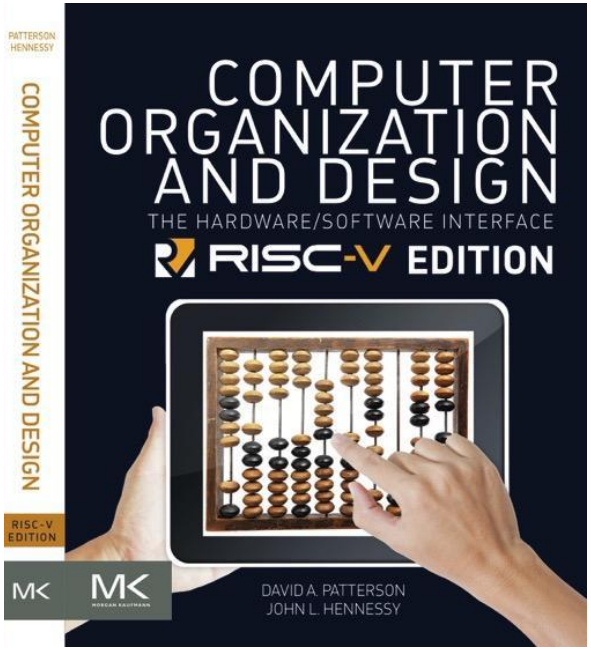
- Just the standard
- NOT an actual piece of silicon that can execute code
- NOT the tools and compilers that are needed to generate and debug that code
- NOT a spec for the platform
- NOT all the OSes, drivers, software written to that standard

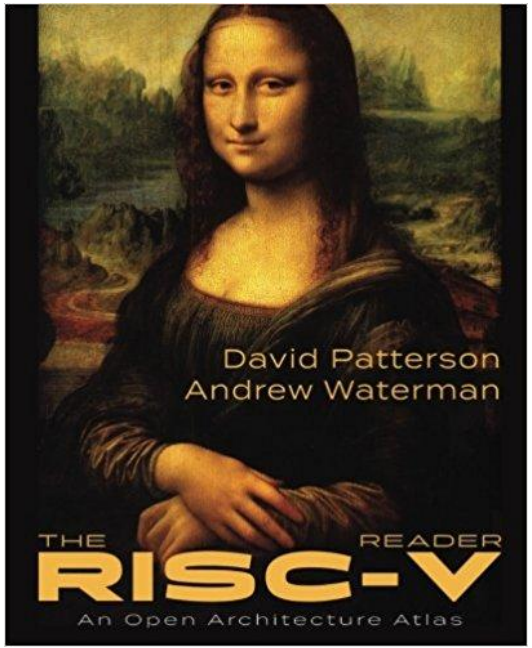
type of operation.

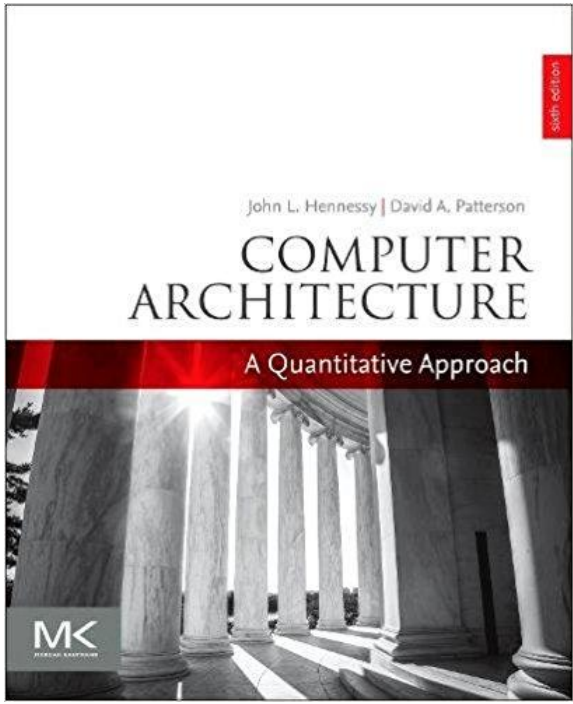
31 25	24 20	19 15	14 12	11 7	6 0
funct7	rs2	rs1	funct3	rd	opcode
7	5	5	3	5	7
0000000	src2	src1	ADD/SLT/SLTU	dest	OP
0000000	src2	src1	AND/OR/XOR	dest	OP
0000000	src2	src1	SLL/SRL	dest	OP
0100000	src2	src1	SUB/SRA	dest	OP

ADD and SUB perform addition and subtraction respectively. Overflows are ignored and the low XLEN bits of results are written to the destination. SLT and SLTU perform signed and unsigned compares respectively, writing 1 to *rd* if *rs1* < *rs2*, 0 otherwise. Note, SLTU *rd*, *x0*, *rs2* sets *rd* to 1 if *rs2* is not equal to zero, otherwise sets *rd* to zero (assembler pseudo-op SNEZ *rd*, *rs*). AND, OR, and XOR perform bitwise logical operations.

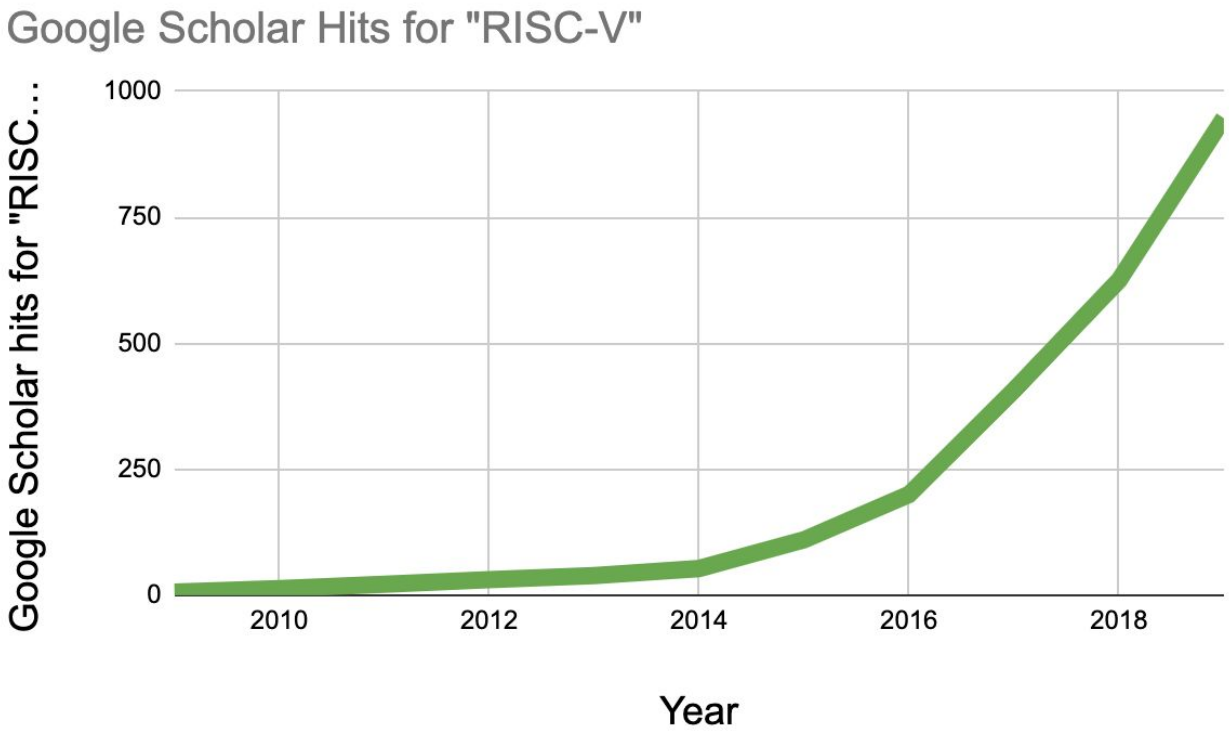
RISC-V: ISA of choice for education







RISC-V: ISA of choice for researchers



RISC-V: The ISA of choice for ~~researchers~~ hackers

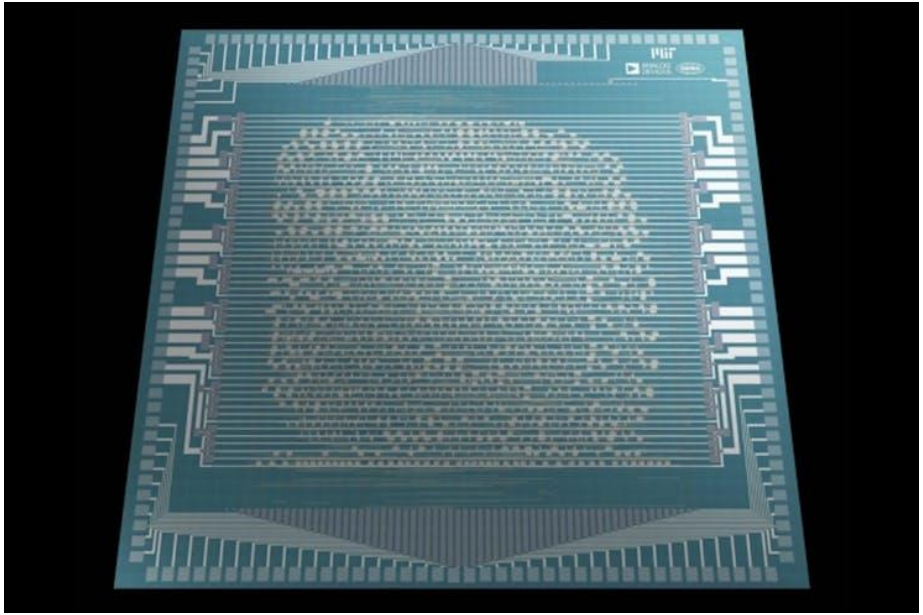
nature

Article | Published: 28 August 2019

Modern microprocessor built from complementary carbon nanotube transistors

RISC-V USES CARBON NANOTUBES

by: Al Williams



RISC-V makes it easy to focus on the cool part of your project

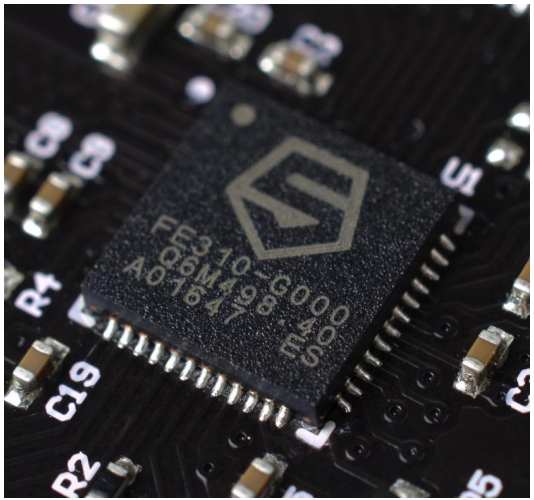
Meanwhile…

- In 2016 I was working at a cryptographic hardware company
  - Super cool work, but… all very much closed source
- The RISC-V inventors were starting a company to actually build RISC-V Silicon

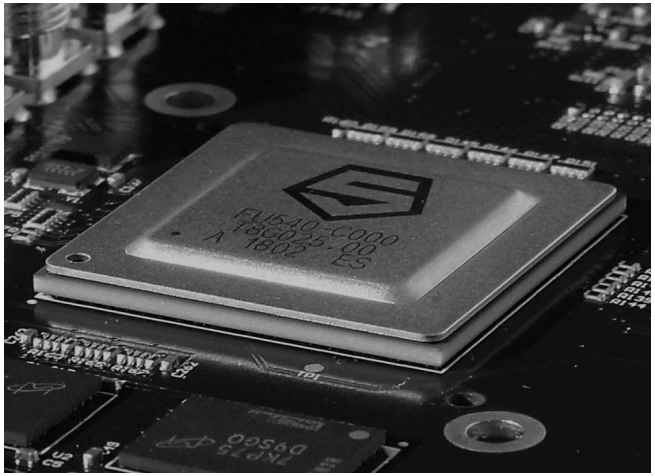
- I had evaluated RISC-V for a project, but it wasn't "ready yet".

Solve a problem I was facing in industry?
By getting involved in Open Source projects?
Sign me up!

First Commercially Available RISC-V Silicon



FE310-G000
Embedded RISC-V MCU:
2016



FU540-C000
Linux Capable RISC-V
Multicore
2018

Free and Open ISA == Open Source Hardware?

- No!
- But it ALLOWS open source hardware implementations
- Or closed source. Or a mixture.

RISC-V: Open Source Hardware (that's practical)

- People Build Open Source Cores & Platforms
- And get to use an open source compiler
- And get to run an open source OS
- And get to run a bunch of Open Source Apps




Hacking on Open Source Hardware

32-bit Embedded Rocket RISC-V Core:
https://github.com/chipsalliance/rocket-chip

FE310-G000 Silicon:
https://github.com/sifive/freedom

On HiFive1 PCB:
https://www.sifive.com/boards/hifive1

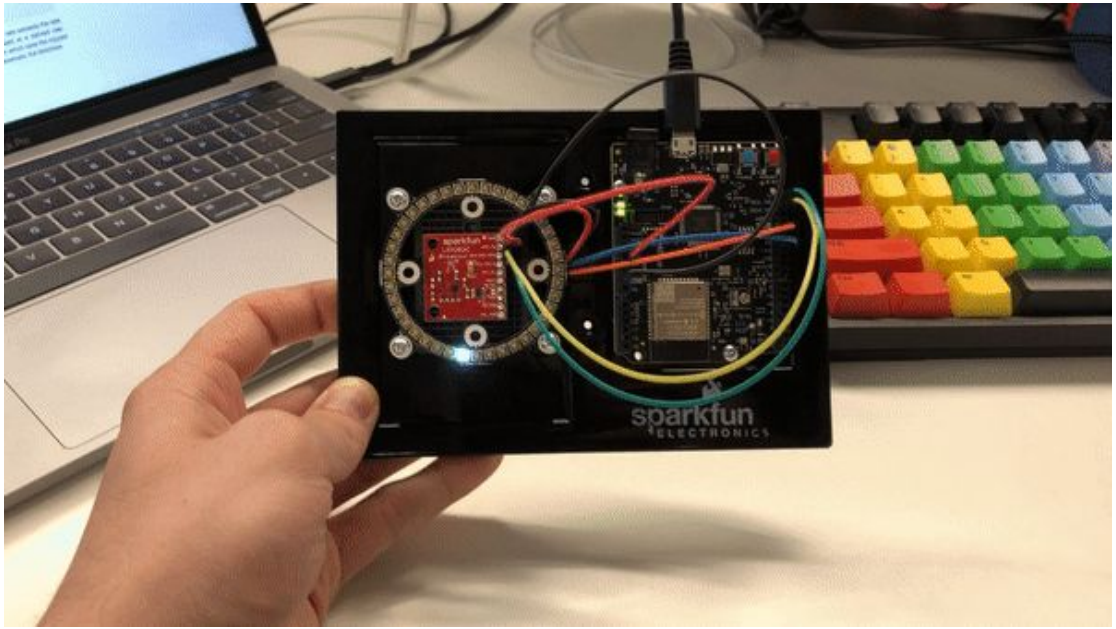
On Open-Source Arduino:
https://github.com/sifive/cinco

Running Adafruit OpenSource NeoPixel library:
https://github.com/mwachs5/Adafruit_NeoPixel

Hacking on Open Source Hardware

Demo running on the
RISC-V Port of Zephyr OS:

https://github.com/sifive/hifive1-revb-pendulum



Hacking on Open Source Hardware

Platform Assembly by ABOpen:
https://abopen.com/news/building-a-risc-v-pc/

64-bit Multicore-Linux Capable Rocket RISC-V::
https://github.com/chipsalliance/rocket-chip

On Freedom Unleashed PCB:
https://www.sifive.com/boards/hifive1

Running Linux from Western Digital:
https://github.com/westerndigitalcorporation/RISC-V-Linux/

Where do FPGAs Come In?

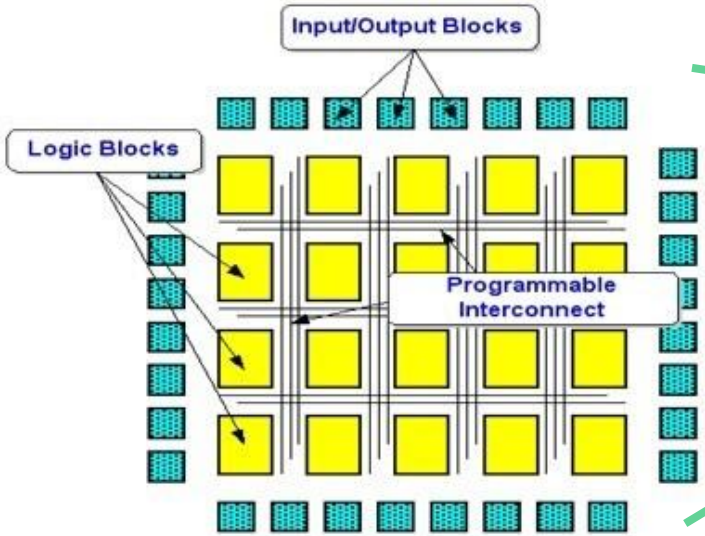
Cool thing about an open source core or chip is understanding and **changing** it.

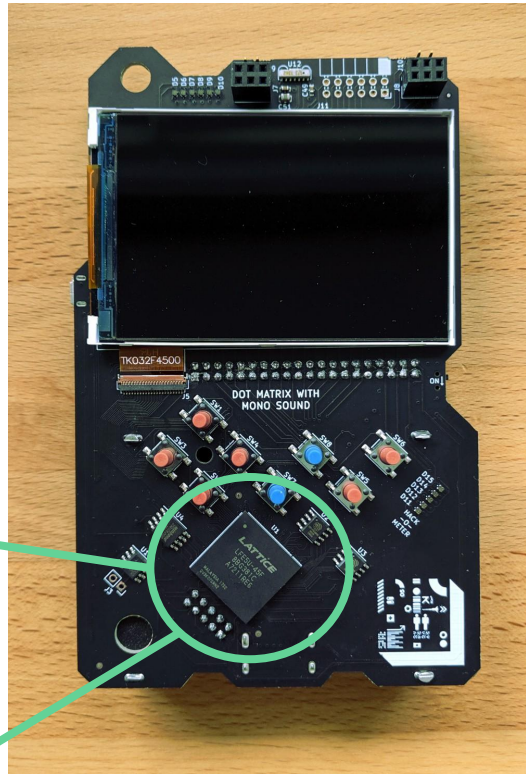
But it takes weeks/months to build a new chip assuming you already know what you want to build

Months - years if you need to figure that out first.

Where do FPGAs Come In?

- Field Programmable Gate Array
- Change a chip's HARDWARE in a few minutes
- Make it act like a new chip!





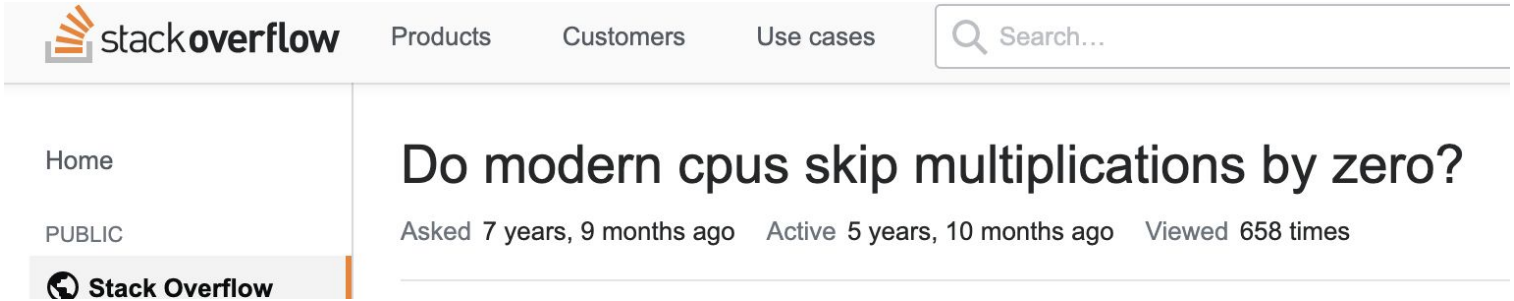
5 Ways to modify a RISC-V Based Design

1
Tune the CPU

- 32 bit? 64 bit? 128 bit?
- Cache sizes
- Branch Prediction
- Pipeline Depth
- ...

5 Ways to modify a RISC-V Based Design

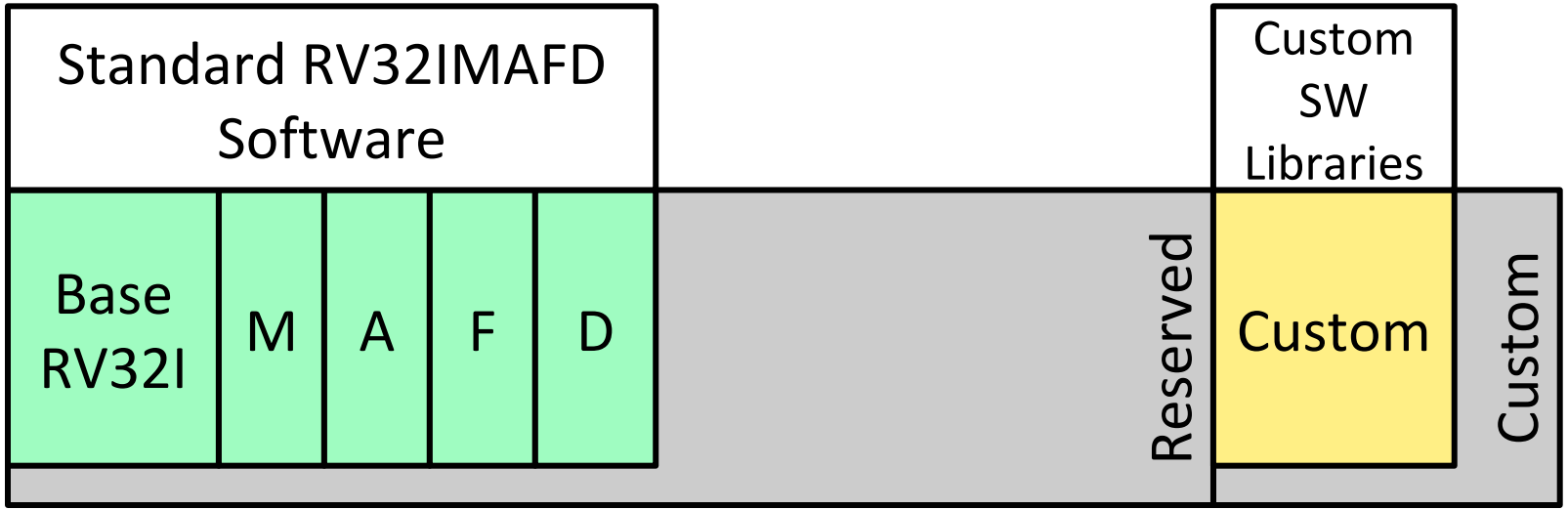
2
Tweak the CPU



Interesting performance tweak or Side Channel Red Flag?

5 Ways to modify a RISC-V Based Design

3
Customize your RISC-V ISA



5 Ways to modify a RISC-V Based Design

4
REPLACE the CPU

This is the true power of RISC-V.
First you pick the ISA, then pick the implementation.
Don’t find what you need?
Design your own!
Get stuck designing your own?
Can look for alternatives!

5 Ways to modify a RISC-V Based Design

5
Modify the PLATFORM

Drive thousands of NeoPixels?

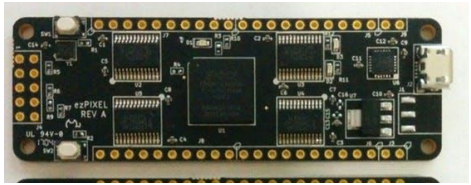


Photo Credit: Thomas Burke
http://makerlogic.com/ezpixel/

Add Deep Learning Acceleration?

NVDLA.org

nvdla

NVDLA Open Source Project

http://nvdla.org/ nvdla@nvidia.com

RISC-V Foundation FPGA Contests

2018: Efficient RISC-V CPU Challenge

- All entries were open-source FPGA implementations of base RISC-V
- Huge variety of languages and techniques

2019: Attack-resistant CPU Challenge:

- All entries were open source FPGA implementations
- RISC-V cores resistant to five security attack vectors

Stay tuned for the 2020 challenge!

Another Cool RISC-V + FPGA Project

- https://fires.im/
- Focuses on accurate performance numbers, not emulation speed
- Uses Amazon F1 FPGA instances in the cloud
- Can model an entire datacenter to enable software and hardware codevelopment

Want to apply your Passion to RISC-V?

Opcode Space Mgmt Standing Committee

Software Standing Committee

Base ISA Ratification Task Group

Privileged ISA Spec Task Group

UNIX-Class Platform Spec Task Group

Formal Specification Task Group

Trusted Execution Env Spec Task Group

B Extension (Bit Manipulation) Task Group

J Extension (Dynam. Translated Lang) Task Group

P Extension (Packed-SIMD Inst) Task Group

V Extension (Vector Ops) Task Group

Cryptographic Extension Task Group

Debug Specification Task Group

Fast Interrupts Spec Task Group

Memory Model Spec Task Group

Processor Trace Spec Task Group

Compliance Task Group

+ Security Committee and new Safety Task Group

https://riscv.org

Open Source Hardware: What Else is Needed?

- Put it on open source FPGAs?

- Design your chip only with open source tools?

- Design an SoC with only Open Source IPs?
- Build your chip in an open source fab?
- Software. Software. Software.

Let’s talk about your projects!

megan@sifive.com

riscv.org