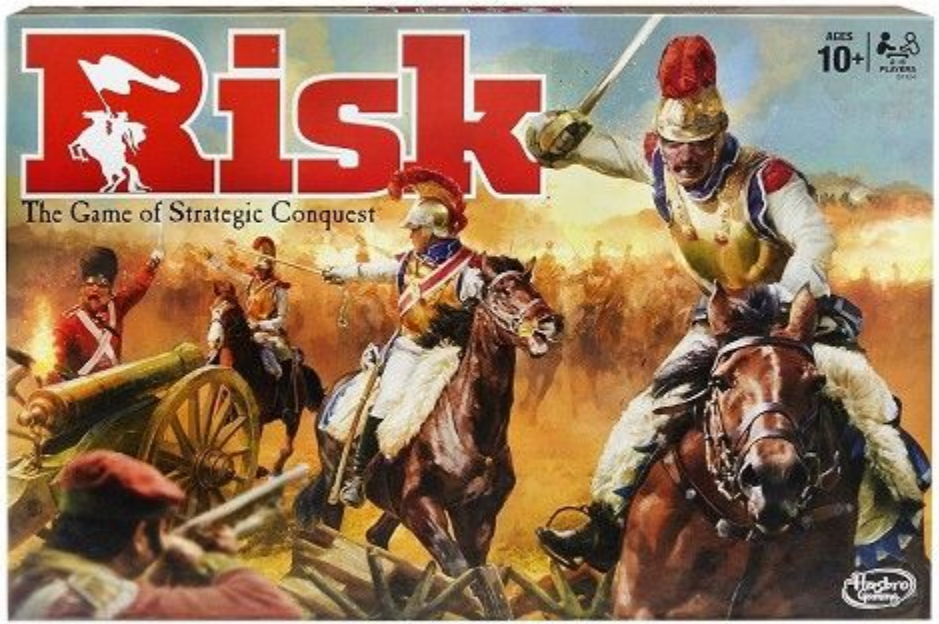
An introduction to RISC-V and a discussion of Speculative Execution Attacks

Carl Perry
caperry@edolnx.net
Mastodon: @edolnx@kind.social
Most other things: edolnx

Pronounced Like This



But really, it’s this

But what is RISC-V?

- The Fifth Iteration (hence -V) of the RISC Architecture, pulling from all the previous RISC implementations going back to the 80s
- Free and Open Source! https://github.com/riscv
- Already shipping, and you're probably already using it
- 32, 64, and 128 bit architecture variants supported
- All the base instructions are explained on a single piece of paper
- riscv.org for more info

Don’t call it a comeback, we’ve been here the whole time

RISC in a CISC world

Origins of RISC

RISC = Reduced Instruction Set Computing

RISC-V is an implementation of RISC, but not all RISC computers are using RISC-V, or RISC-IV, or RISC-III, or RISC-II, or RISC-I, or PA-RISC, or ARM

Also, some ARM processors are CISC

Are you appropriately confused now?

RISC vs CISC

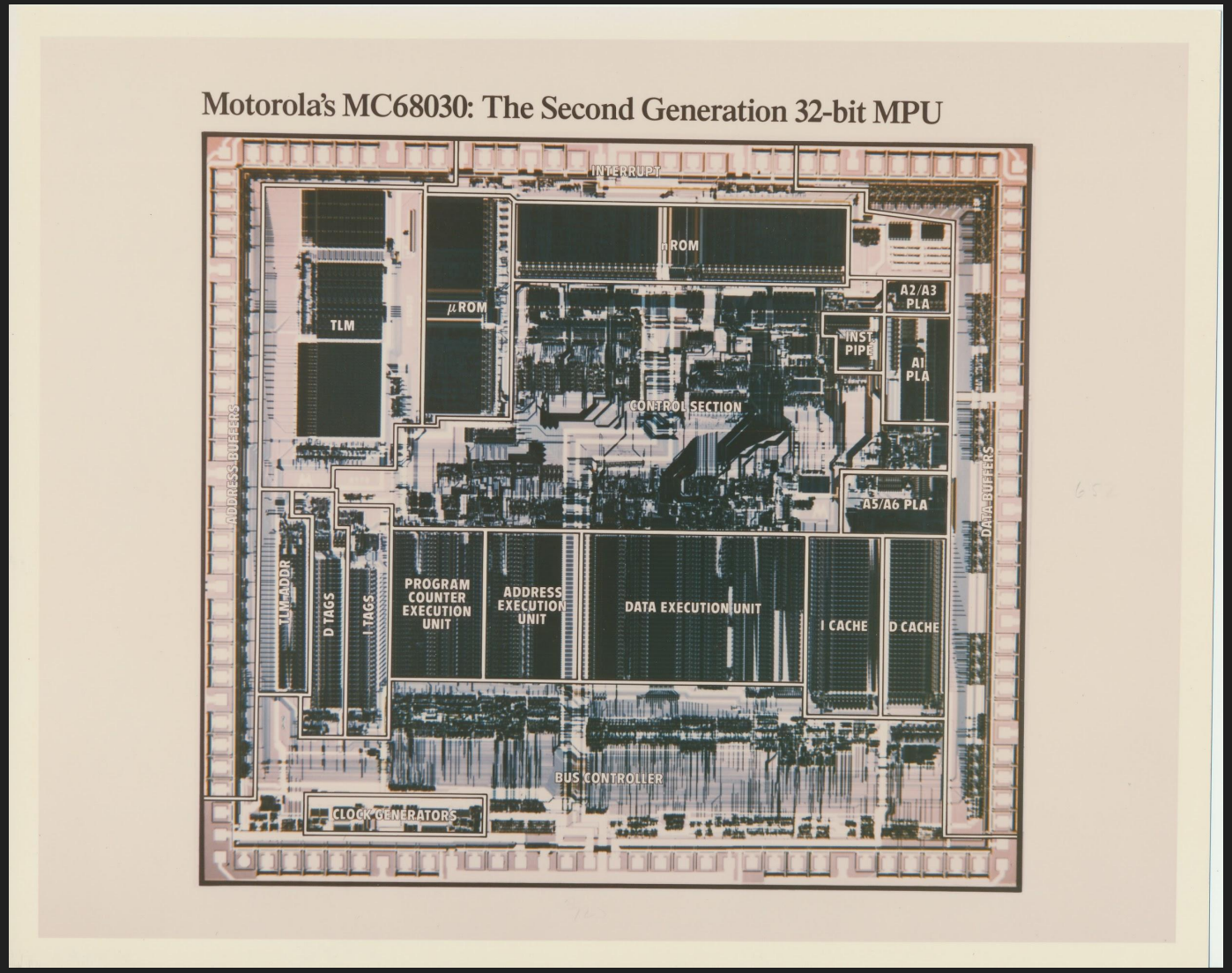
Let’s simplify: it comes down to two schools of thought

CISC = Provide all the instructions a software developer needs

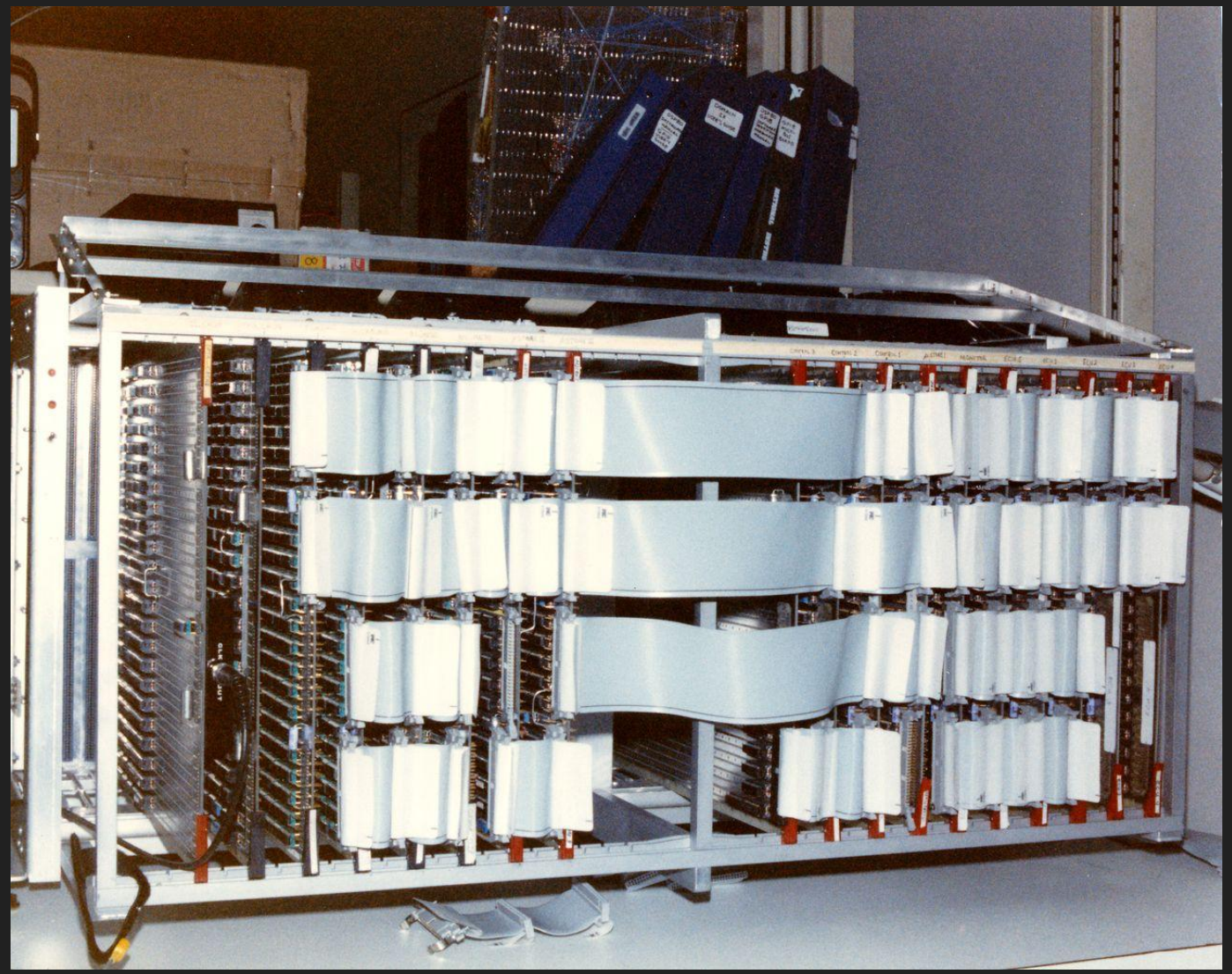
RISC = Make the building blocks instructions, and leverage the compiler to combine them

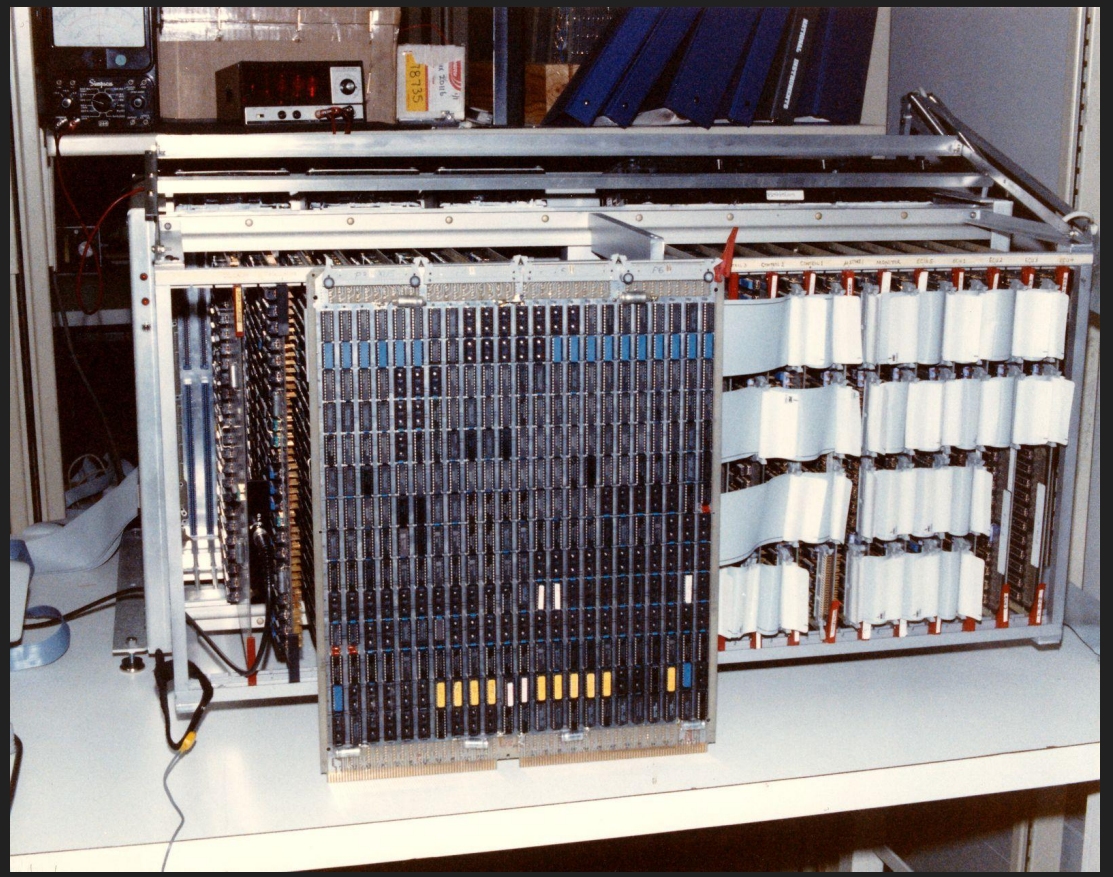
The problem with this is that CISC is very complex to implement

Annotated Die Shot of a Motorola 68030 (CISC)

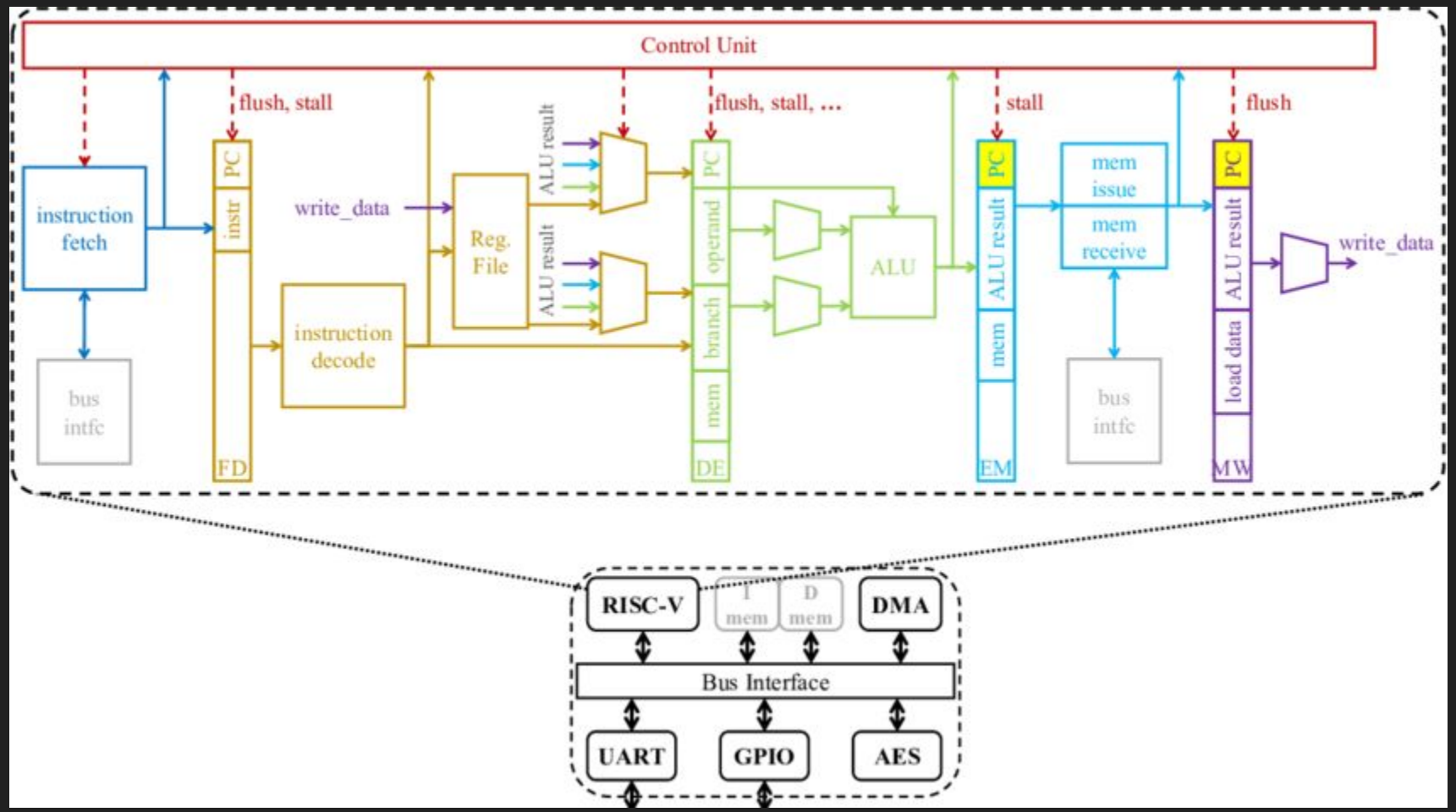


…and this is what the development breadboard looked like

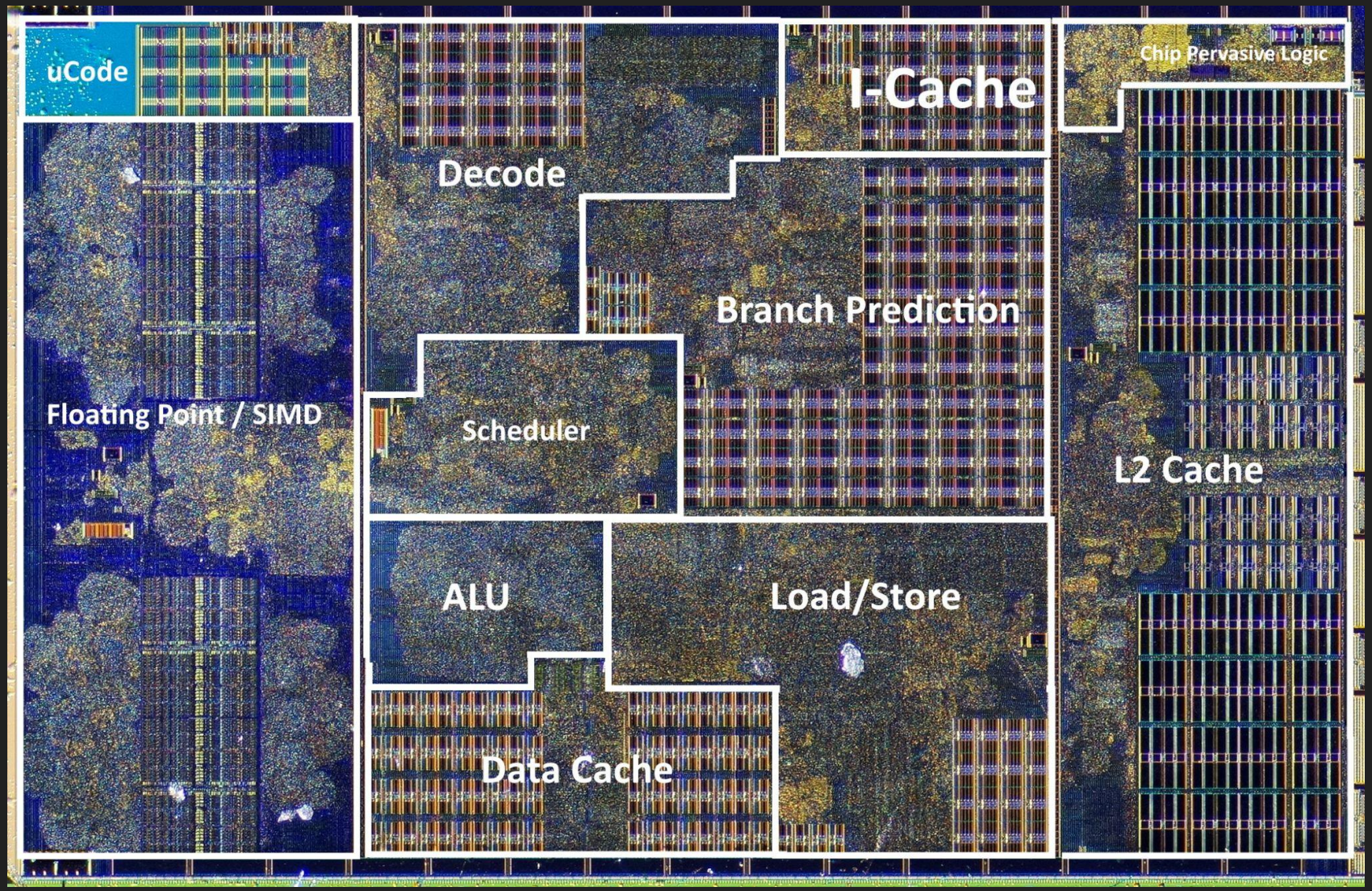




Let’s look at what makes up a RISC-V hart (aka core)



Now let’s compare that with a ZEN 2 core



Who Makes RISC-V?

Lots of companies, and a few folks

SiFive

Andes Technologies

Qualcomm

Microchip

Western Digital

Nvidia

T-Head

…and a bunch more

There are also a lot of smaller groups and academic institutions like:

OpenHW

ETH Zurich

Enjoy Digital

(Most products out of these groups are fully open source under liberal licenses)

Questions?

Speculative Execution

Trading security for speed

How did we get here?

In the prep for this meeting I got a lot of questions about RISC-V vs other platforms for things like:

- Viruses
- Firewalls
- Spectre/Meltdown

So let’s cover these.

In-Order vs Out-Of-Order execution

In-Order

Traditional

Slow

Doesn’t scale

Secure!

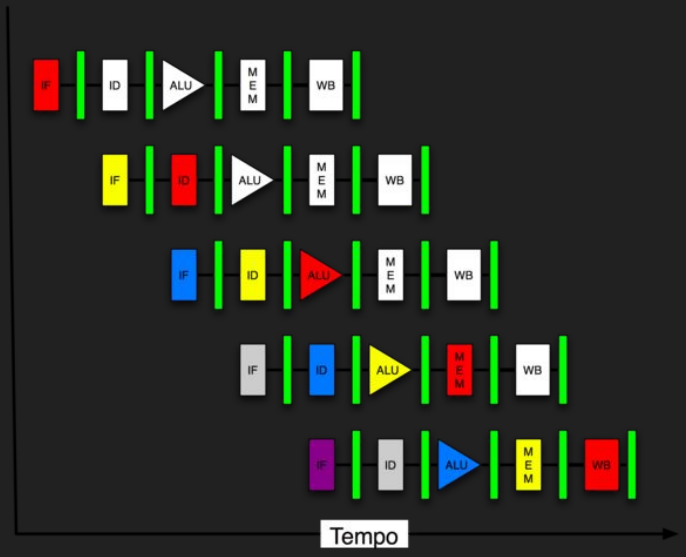
Out-Of-Order

Modern!

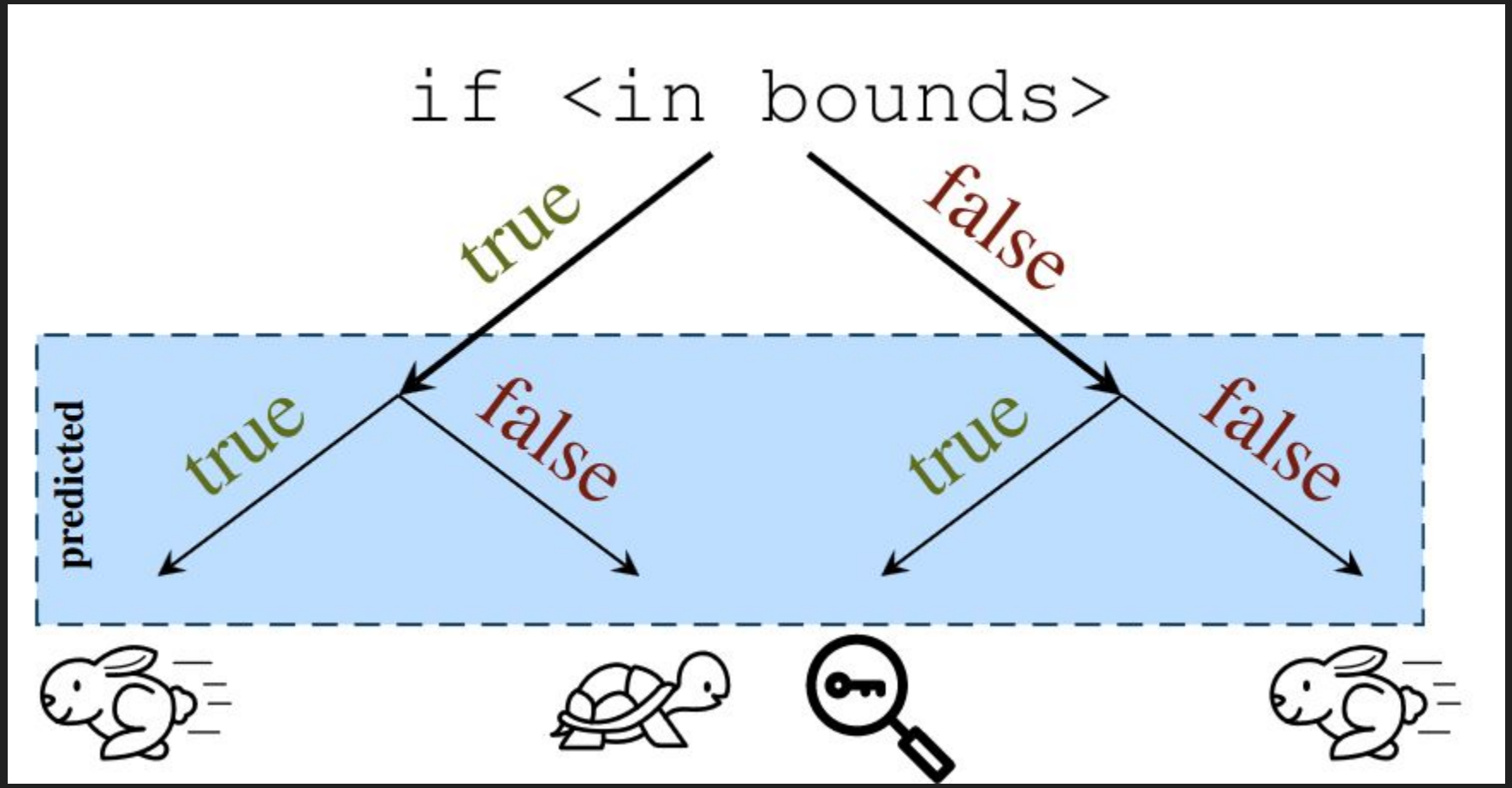
Fast!

Scalable (add more stages to your pipeline)

Has some security issues…



Speculative Execution Attacks



Questions?

It’s not all doom and gloom