

RISC-V

A REDUCED INSTRUCTION SET COMPUTER

INTRODUCTION

It is a free and open instruction set architecture (ISA) that provides alternative to closed ISAs used in X86 and ARM CPUs



BACKGROUND

- The concept of RISC-V has roots in earlier RISC (Reduced Instruction Set Computing) architectures that emerged in the 1980s
- In 2010, the RISC-V project was formally launched at the University of California, Berkeley. aim to create an open free source ISA (instruction set architecture) based on RISC principles.
- In 2010-2015 ,The initial focus was on defining the core ISA and developing basic tools for design and simulation
- The RISC-V Foundation was established in 2015 to foster collaboration and accelerate adoption, companies like Western Digital and SiFive led to the development of RISC-V based processors for various applications.
- RISC-V adoption has gained significant traction in recent years, particularly in the embedded systems domain (e.g., wearables, IoT devices).

What makes RISC-V come to market?

- End of Moore's Law
- Free and open source

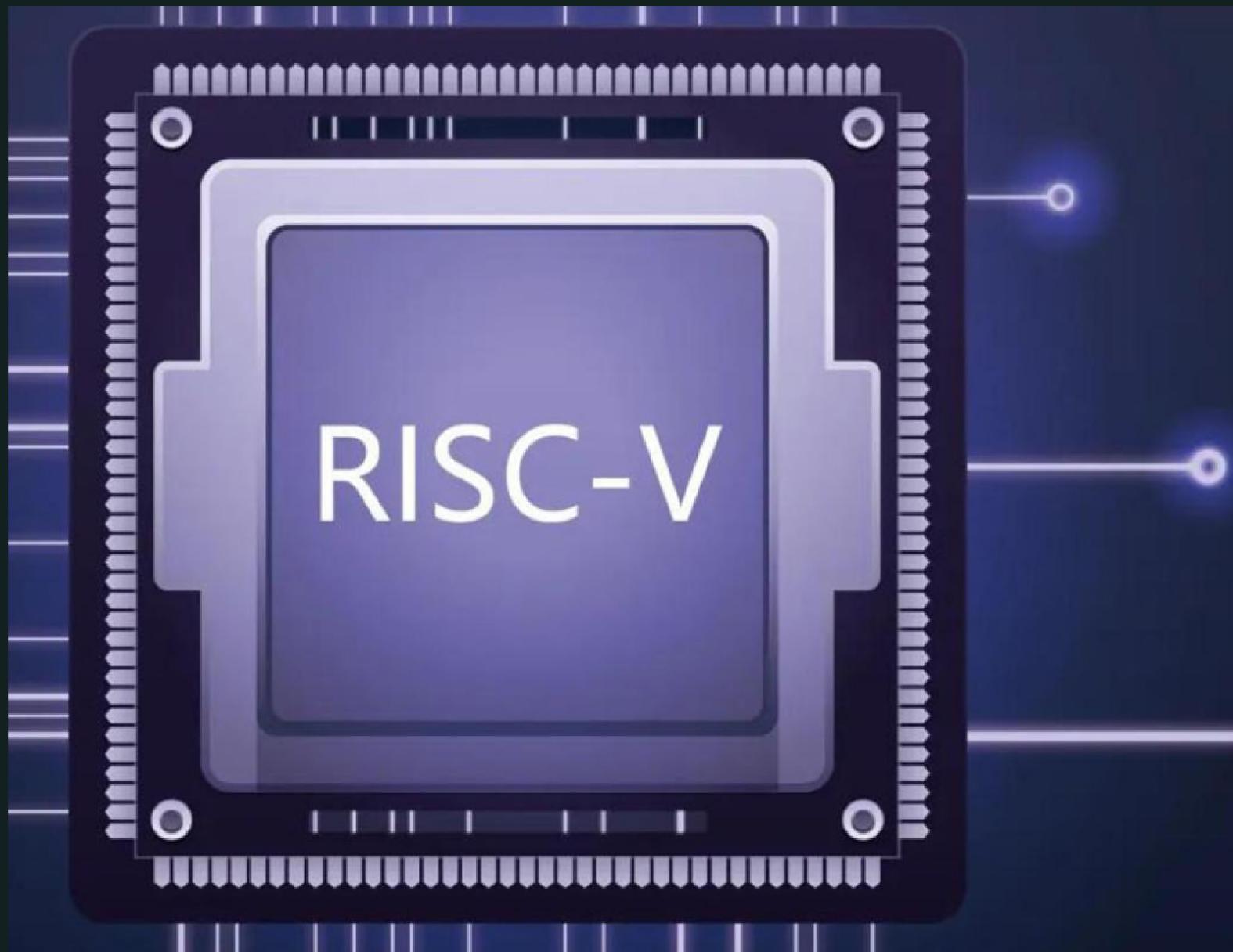
Why is it growing??

- Provide flexibility to designer
- Standard platform
- Can be freely extended
- Versatility

IMPLEMENTATION

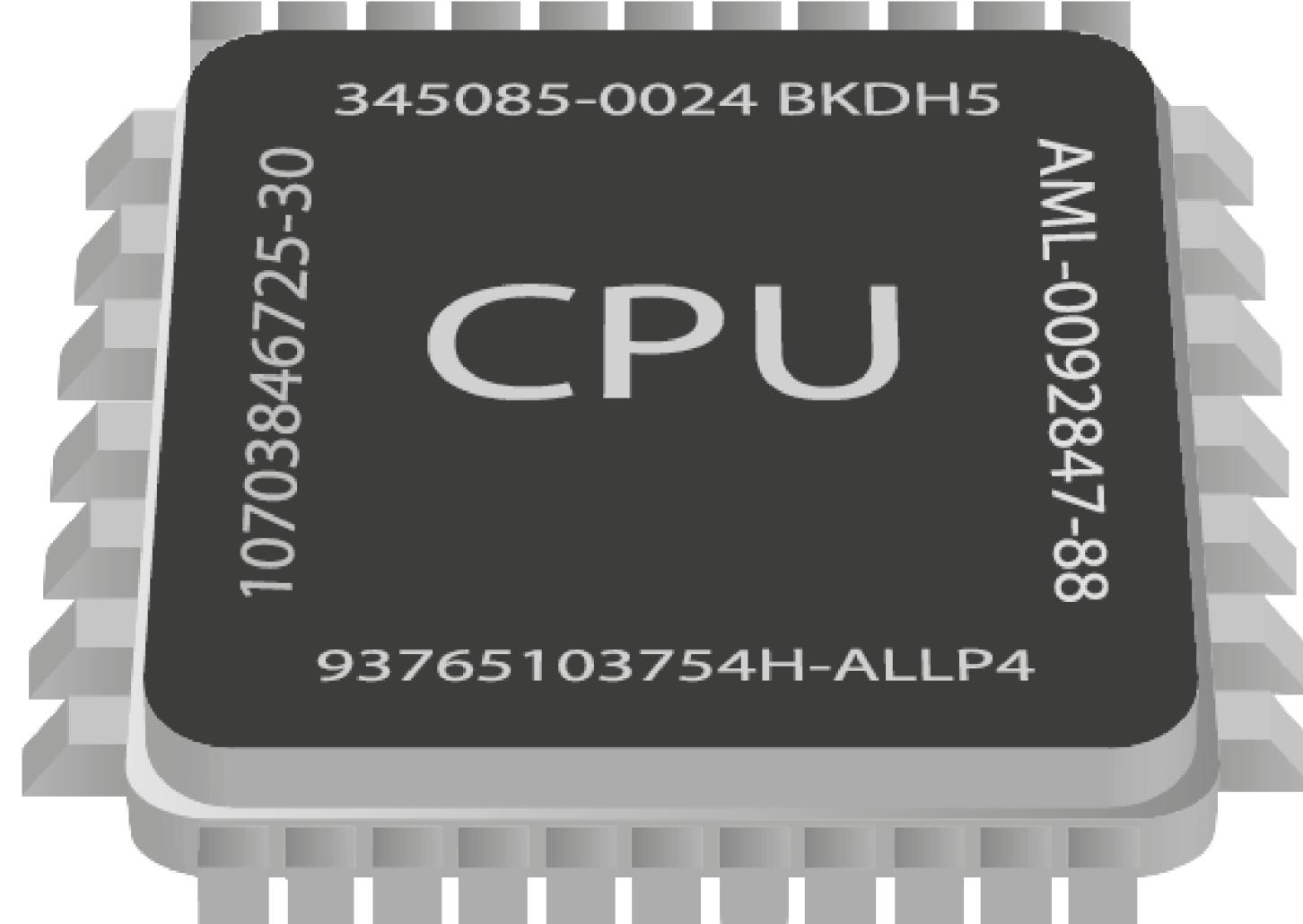
- OBJECT DETECTION
- AUTO DRIVING
- MEDICAL FIELD
- IMAGE CLASSIFICATION

Angella White





ADVANTAGES OF RISC-V



Cost-Effective

RISC-V's open-source nature eliminates the need for licensing fees associated with proprietary ISAs (instruction set architectures).

Flexibility

RISC-V provides a core set of instructions with the ability to add optional extensions for specific functionalities. This empowers developers to create customized processors that perfectly align with their application's requirements.

High Performance

RISC-V's core principle of using simpler instructions leads to efficient execution. Because the instructions are more concise, they can be decoded and processed by the processor faster.

Growing Ecosystem

The open-source nature of RISC-V has fostered a rapidly growing ecosystem of tools, software, and hardware resources.

CHALLENGES:

1) Toolchain Development

The RISC-V toolchain (compilers, debuggers, simulators) is still under development compared to more mature architectures

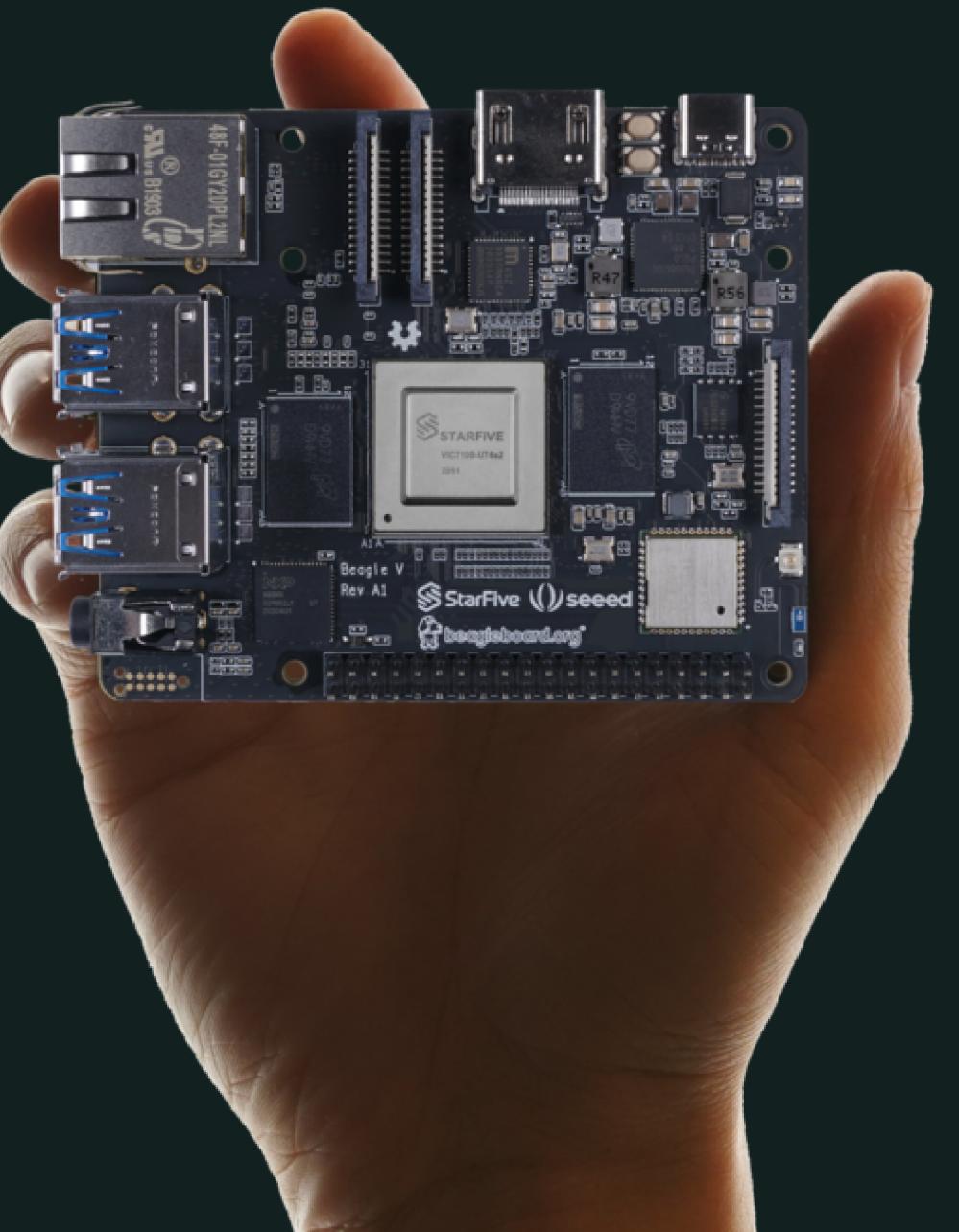
2) Hardware Design

The need for custom design can lead to higher upfront costs compared to readily available processors from other architectures.

3) Security

The relative newness of RISC-V compared to established architectures means there's potentially less established security knowledge and practices.

What's Next in RISC-V??





Do you have
any questions?

