ARM Architecture Overview

1. RISC and CISC
2. ARM Architectures/Micro-architecture

History of ARM

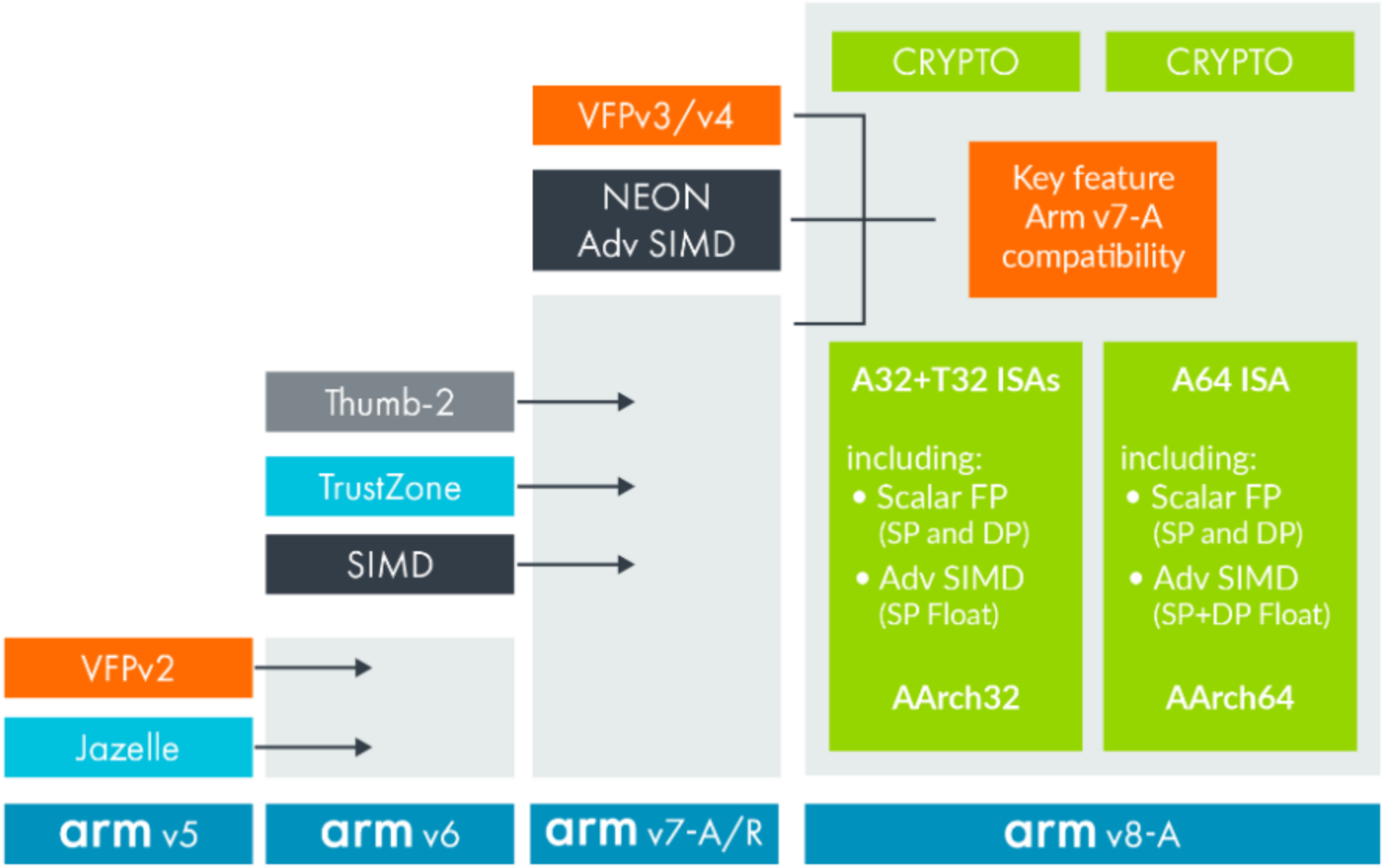
- ARM (Acorn RISC Machine) started as a new, powerful, CPU design for the replacement of the 8-bit 6502 in Acorn Computers (Cambridge, UK, 1985)
- First models had only a 26-bit program counter, limiting the memory space to 64 MB (not too much by today standards, but a lot at that time).
- 1990 spin-off: ARM renamed Advanced RISC Machines
- ARM now focuses on Embedded CPU cores
  - IP licensing: Almost every silicon manufacturer sells some microcontroller with an ARM core. Some even compete with their own designs.
  - Processing power with low current consumption
    - Good MIPS/Watt figure
    - Ideal for portable devices
  - Compact memories: 16-bit opcodes (Thumb)
- New cores with added features
  - Harvard architecture (ARM9, ARM11, Cortex)
  - Floating point arithmetic
  - Vector computing (VFP, NEON)
  - Java language (Jazelle)

What is the Harvard architecture?

Facts

- 32-bit CPU
- 3-operand instructions (typical): ADD Rd,Rn,Operand2
- RISC design…
  - Few, simple, instructions
  - Load/store architecture (instructions operate on registers, not memory)
  - Large register set
  - Pipelined execution, Hardwired Control Units
- Although with some CISC touches…
  - *Multiplication* and *Load/Store Multiple* are complex instructions (many cycles longer than regular, RISC, instructions)
- And some very specific details
  - Not stack, Link register instead, is used to keep return address.
  - PC as a regular register
  - Conditional execution of all instructions
  - Flags altered or not by data processing instructions (selectable)
  - Concurrent shifts/rotations (at the same time of other processing)
  - …

Development of ARM Architectures



CISC (Complex Instruction Set Computer)

- Example: i386, i486, ….
- Compiler simplification?
  - Disputed…
  - Complex machine instructions harder to exploit
  - Optimization more difficult
- Smaller programs?
  - Program takes up less memory but…
  - Memory is now cheap
  - May not occupy less bits, just look shorter in symbolic form
    - More instructions require longer op-codes
    - Register references require fewer bits
- Faster programs?
  - More complex control unit
    - Microprogram control store larger
      thus simple instructions take longer to execute

Reduced Instruction Set Computer (RISC)

- Key features
  - Limited and simple instruction set
    - Fixed Instruction Formats and Fixed Instruction Execution Time
      - Hardwired Control Units
    - Arithmetic Logic Operations: register-to-register
    - Specific load/store instructions
  - Large number of general purpose registers
    - or use of compiler technology to optimize register use
    - Improve the performance of procedure calls
  - Emphasis on optimising the instruction pipeline
    - Increasing the processor clock frequency

Use of RISC architectures

- Use of RISC architectures
  - Mobile systems and Entertainment systems
    - ARM → Android, iOS devices; Industrial Microcontrollers;
      - dominates the market for low power and low cost embedded systems
    - MIPS
      - Used in the PlayStation, PlayStation 2, Nintendo 64, PlayStation Portable game consoles, and residential gateways like Linksys WRT54G series.
    - Hitachi's SuperH, Atmel AVR
  - High end RISC and supercomputing
    - MIPS, by Silicon Graphics (ceased making MIPS-based systems in 2006).
    - SPARC, by Oracle (previously Sun Microsystems), and Fujitsu. → K computer, 日本神戸
    - IBM's Power Architecture, used in many of IBM's supercomputers, midrange servers and workstations. → IBM Sequoia: top1 supercomputer 2015
    - Hewlett-Packard's PA-RISC, also known as HP-PA (discontinued at the end of 2008).
    - Alpha, used in single-board computers, workstations, servers and supercomputers from Digital Equipment Corporation, Compaq and HP (discontinued as of 2007).
    - RISC-V, the open source fifth Berkeley RISC ISA

For 2020 top 10 supercomputer:

https://www.analyticsinsight.net/top-10-fastest-supercomputers-world-watch-2020/

ARM Architecture and Micro-architecture

ARM Architecture

- The Arm architecture is one of the most popular processor architectures in the world today, with several billion Arm-based devices shipped every year.
- There are three architecture profiles: A, R and M.

A-profile (Applications)	R-profile (Real-time)	M-profile (Microcontroller)
• High performance	• Targeted at systems with real-time requirements.	• Smallest/lowest power. Small, highly power-efficient devices.
• Designed to run a complex operating system, such as Linux or Windows.	• Commonly found in networking equipment, and embedded control systems.	• Found at the heart of many IoT devices.

Processors used in a cell phone

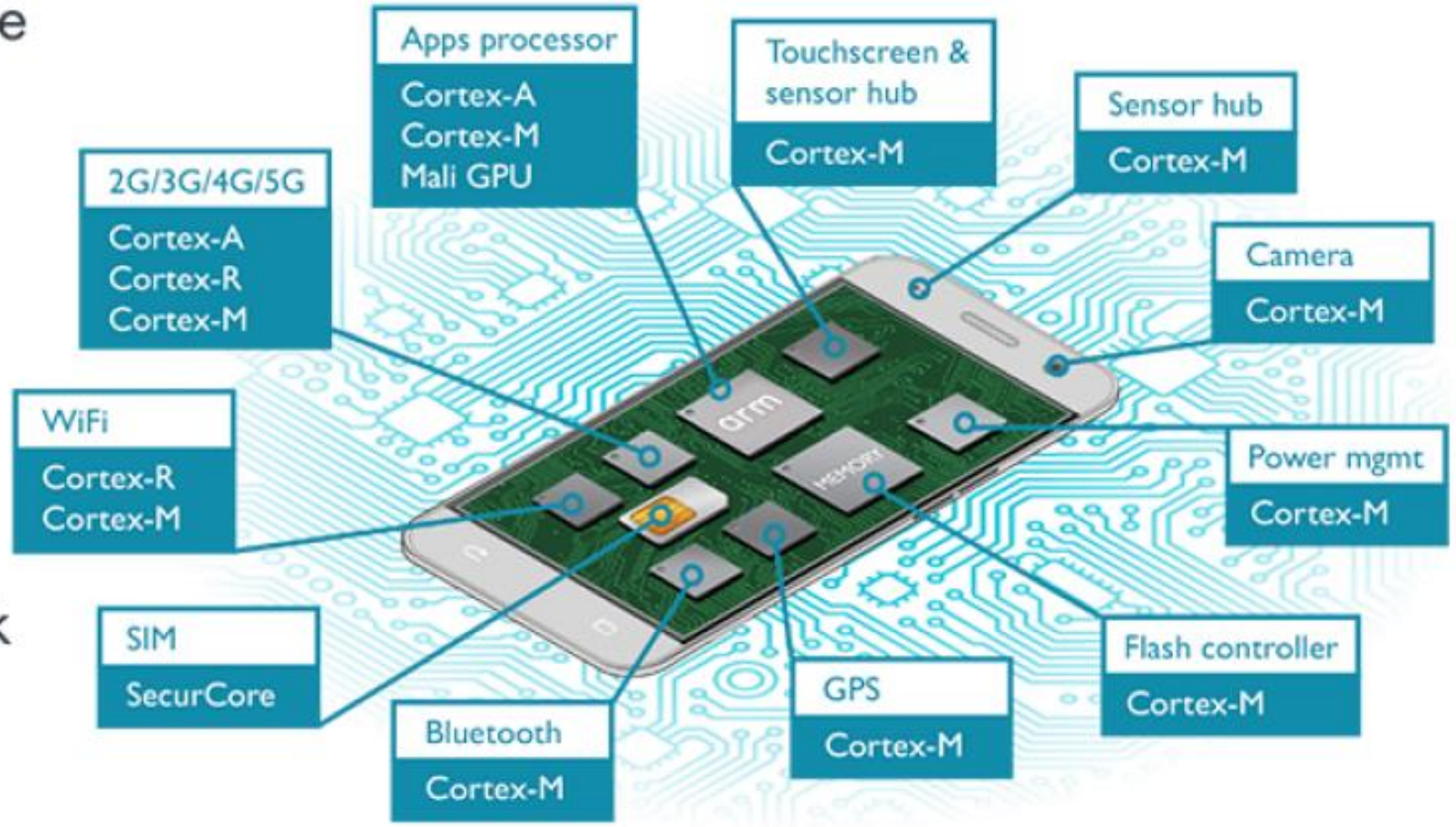
arm TrustZone

arm Artisan

arm Mali

arm Cortex

arm CoreLink



What do we mean by architecture?

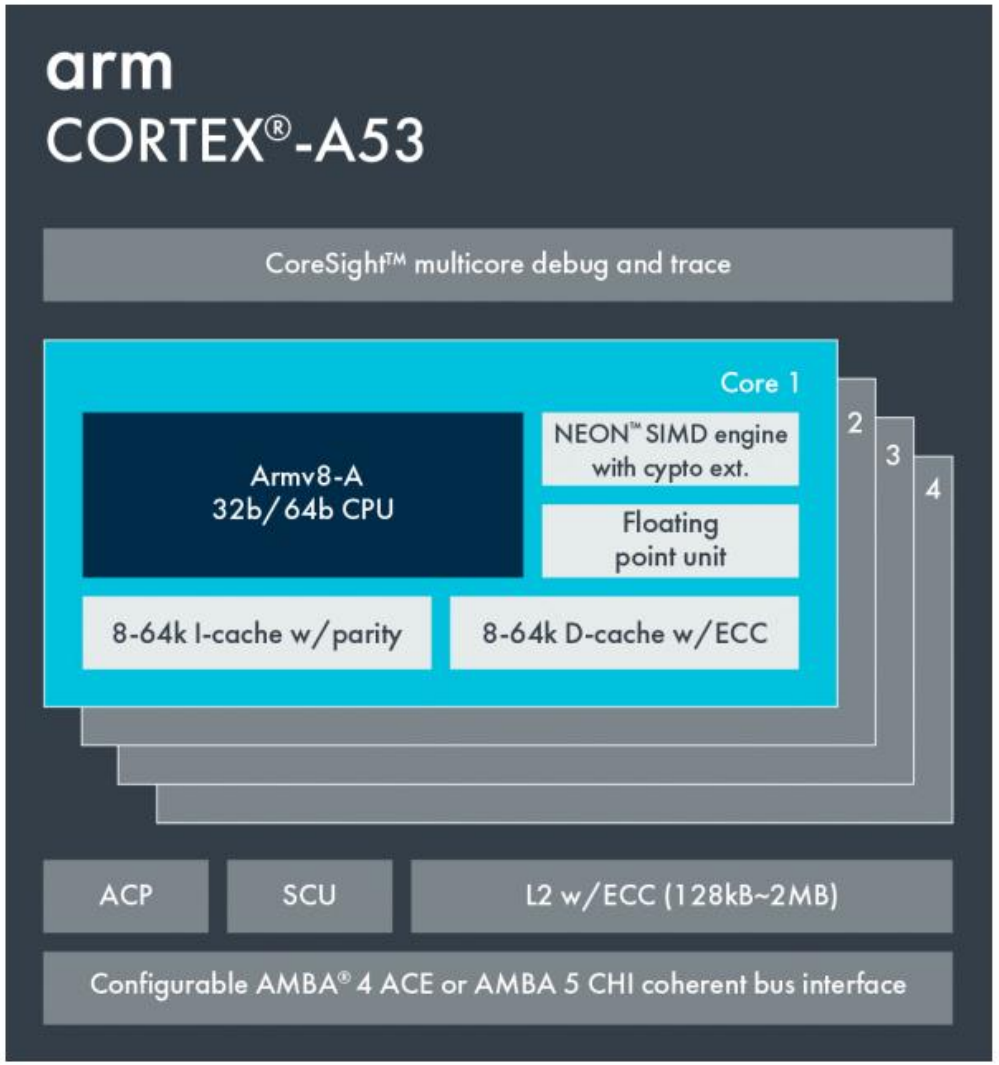
- An architecture specifies what a processor will behave, such as what instructions it has and what the instructions do.

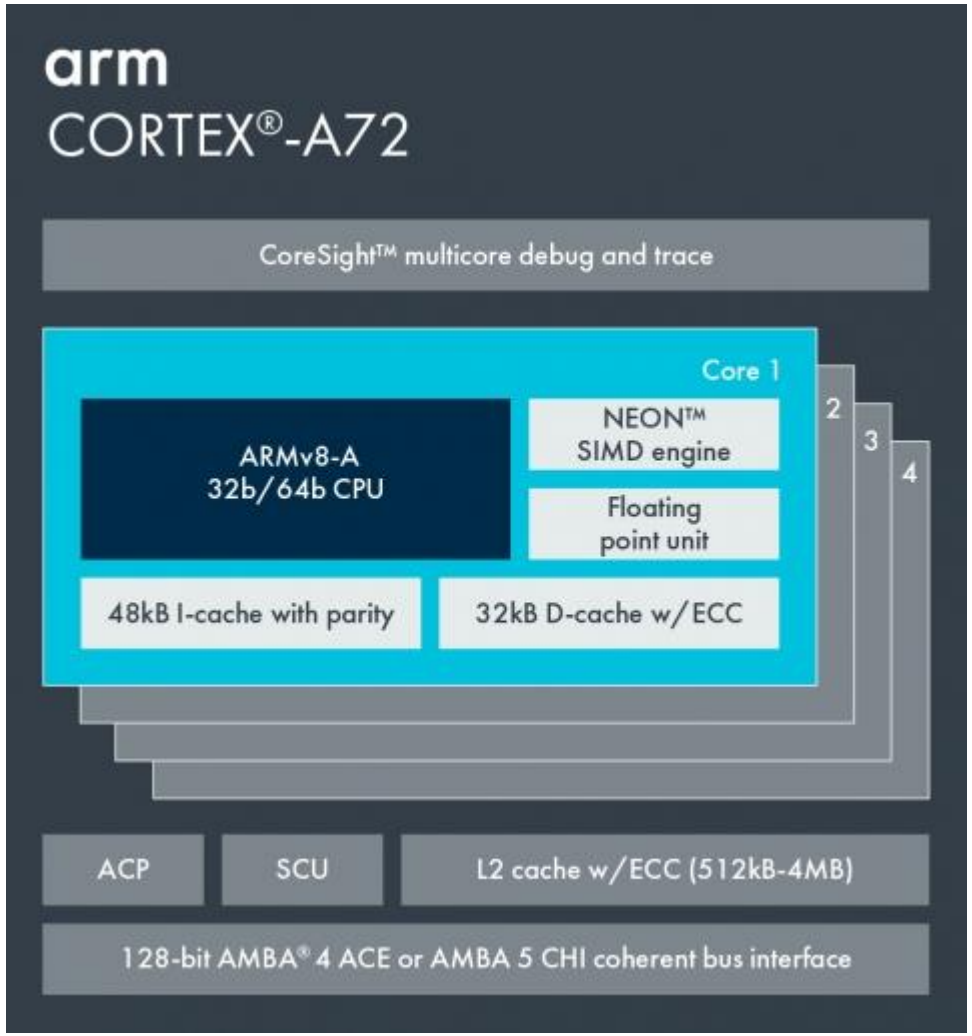
Instruction set	• The function of each instruction • How that instruction is represented in memory (its encoding).
Register set	• How many registers there are. • The size of the registers. • The function of the registers. • Their initial state.
Exception model	• The different levels of privilege. • The types of exceptions. • What happens on taking or returning from an exception.
Memory model	• How memory accesses are ordered. • How the caches behave, when and how software must perform explicit maintenance.
Debug, trace, and profiling	• How breakpoints are set and triggered. • What information can be captured by trace tools and in what format.

Architecture and micro-architecture

- Architecture does not tell you how a processor is built and works. The build and design of a processor is referred to as micro-architecture. Micro-architecture tells you how a processor works.
- Micro-architecture includes things like:
  - Pipeline length and layout.
  - Number and sizes of caches.
  - Cycle counts for individual instructions.
  - Which optional features are implemented.
- For example, Cortex-A53 and Cortex-A72 are both implementations of the Armv8-A architecture.
  - This means that they have the same architecture, but they have very different micro-architectures, as shown in the following image:

Cortex-A53 and Cortex-A72 (are both based on the Armv8-A architecture)



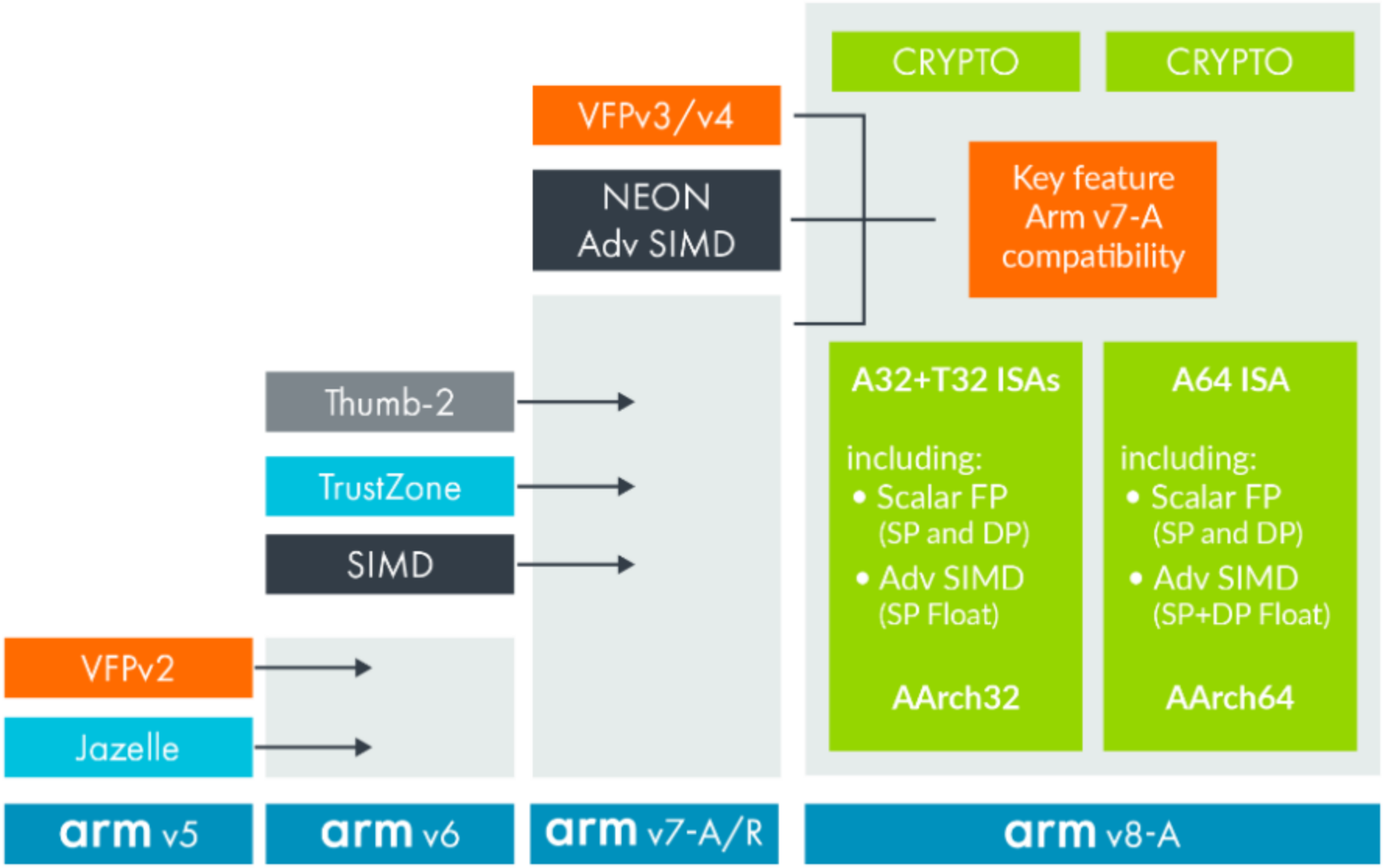


Architecture specifies functionality and instruction sets

- Software that is architecturally-compliant can run on either the Cortex-A53 or Cortex-A72 without modification, because they both implement the same architecture.

Target	Optimized for power efficiency	Optimized for performance
Pipeline	8 stages In-order	15+ stages Out-of-order
Caches	L1 I cache: 8KB - 64KB L1 D cache: 8KB - 64KB L2 cache: optional, up to 2MB	L1 I cache: 48KB fixed L1 D cache: 48KB fixed L2 cache: mandatory, up to 2MB

Development of ARM Architectures



Conclusions

- ARM is a 32/64-bit reduced instruction set computer (RISC) instruction set architecture (ISA) developed by ARM Holdings.
  - The ARM architecture is the most widely used 32-bit ISA in terms of numbers produced. The simplicity of ARM processors makes them suitable for low power applications.
- Competitor
  - RISC-V is an open standard instruction set architecture (ISA) based on established reduced instruction set computer(RISC) principles.
  - 晶心科技

  https://www.andestech.com/en/homepage/