ARM
(and performance monitoring)

Michael Williams, November 2013

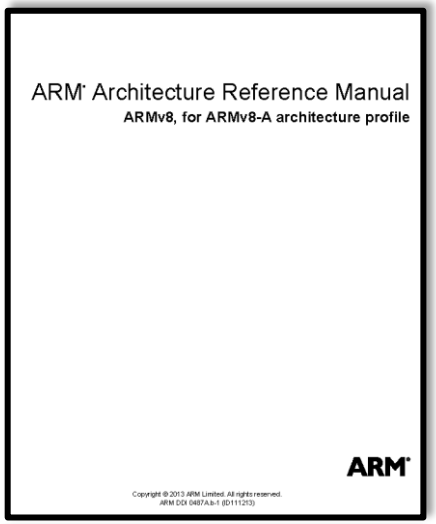
ARM in numbers

- ARM “the company” : **>23 years**
- Processors shipped in 2012 : **~8.7 Bu (~4.9 Bu in 1H’13)**
- Processors shipped in total : **>45 Bu**
- Processor licensees : **~1040**
- Semiconductor partners : **310**
- Foundry partners : **5+**
- Process technology : **14 - 250nm**
- Connected community members : **1000+**

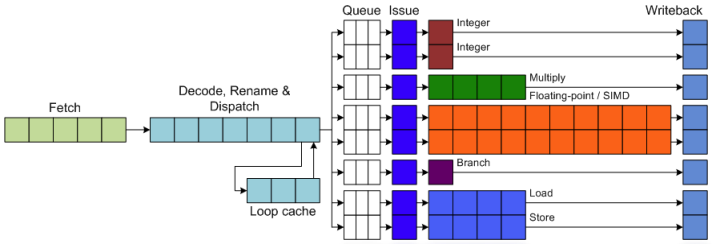
Architecture to implementation

- Architecture is the contract between hardware and software

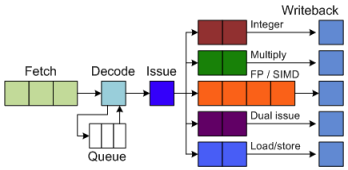
One architecture "ARMv8"



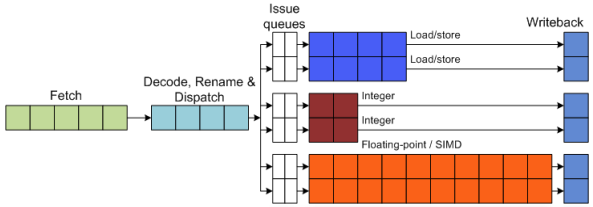
Many implementations
(ARM and licensees)



Cortex-A57 "Atlas"

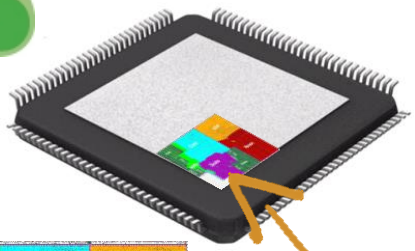


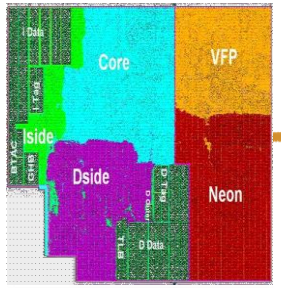
Cortex-A53 "Apollo"



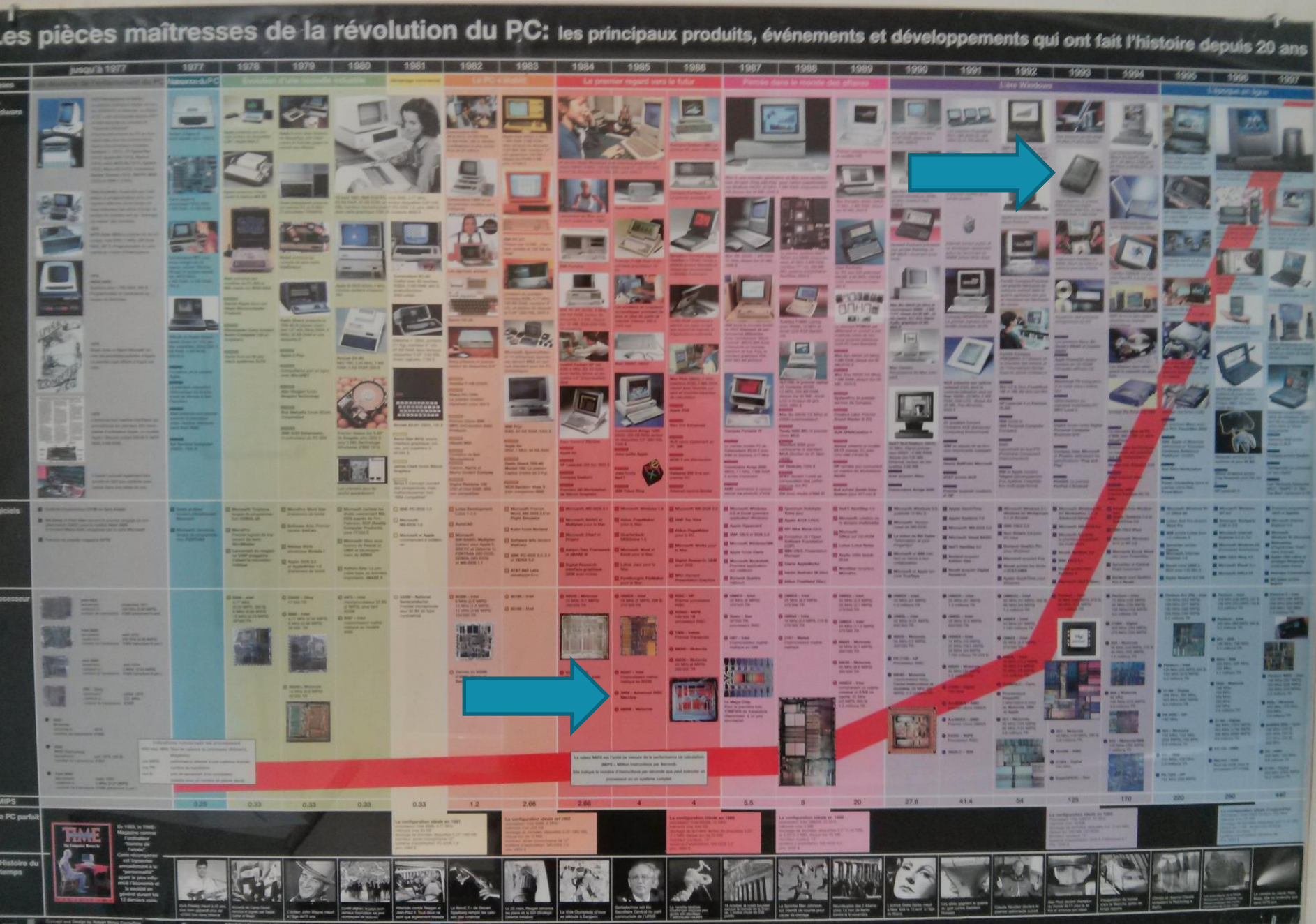
Cortex-A12

Billions of devices



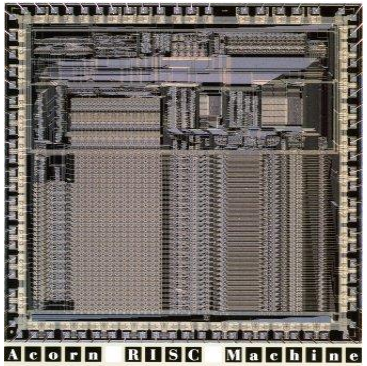


Architecture ***Microarchitecture*** ***Implementation***


Les pièces maîtresses de la révolution du PC: les principaux produits, événements et développements qui ont fait l'histoire depuis 20 ans
jusqu'à 1977
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
Naissance du PC
Le premier regard vers le futur
Percée dans le monde des affaires
L'ère Windows
L'époque en ligne
MIPS
0.25
0.33
0.33
0.33
0.33
1.2
2.66
2.66
4
4
5.5
8
20
27.6
41.4
54
125
170
220
260
440
TIME

Evolution of the ARM architecture

- Original ARM architecture (1985)
  - 32-bit RISC architecture, “Acorn RISC Machine”
  - 15 general-purpose 32-bit registers
    - Banked registers for nested exception handling
  - Conditional execution on all instructions
  - Load/Store Multiple operations
    - Good for code density
  - Shifts available on data processing and address generation
  - Optional floating-point coprocessor in separate socket



ARM1 3μm 6K gates

7mm × 7mm = 49mm^2

- 32-bit virtual address space (AArch32)
  - Original architecture had 26-bit address space only
  - 32-bit addressing came early (ARMv3)
- Basic 3 stage microarchitecture used by ARM7TDMI
  - Still ships billions of units each year

Evolution of the ARM architecture (2)

- T32 (Thumb) instruction set was the next big step (1995)
  - ARMv4T architecture (ARM7TDMI)
  - Introduced a 16-bit instruction set alongside the 32-bit instruction set
- ARMv5, ARMv6, ARMv7 evolved the 32-bit ARM architecture:
  - “Thumb-2” - variable length instruction set
  - Floating-point, SIMD, and DSP operations
  - Multiprocessing
  - Architectural virtual memory system (VMSA32)
  - Physical address extension
  - TrustZone security
  - Virtualization
- Microarchitecture evolves alongside architecture
  - ARM7: 3 stage, single issue
  - Cortex-A15: ~20 stages, three issue, out-of-order, quad-core

ARMv8

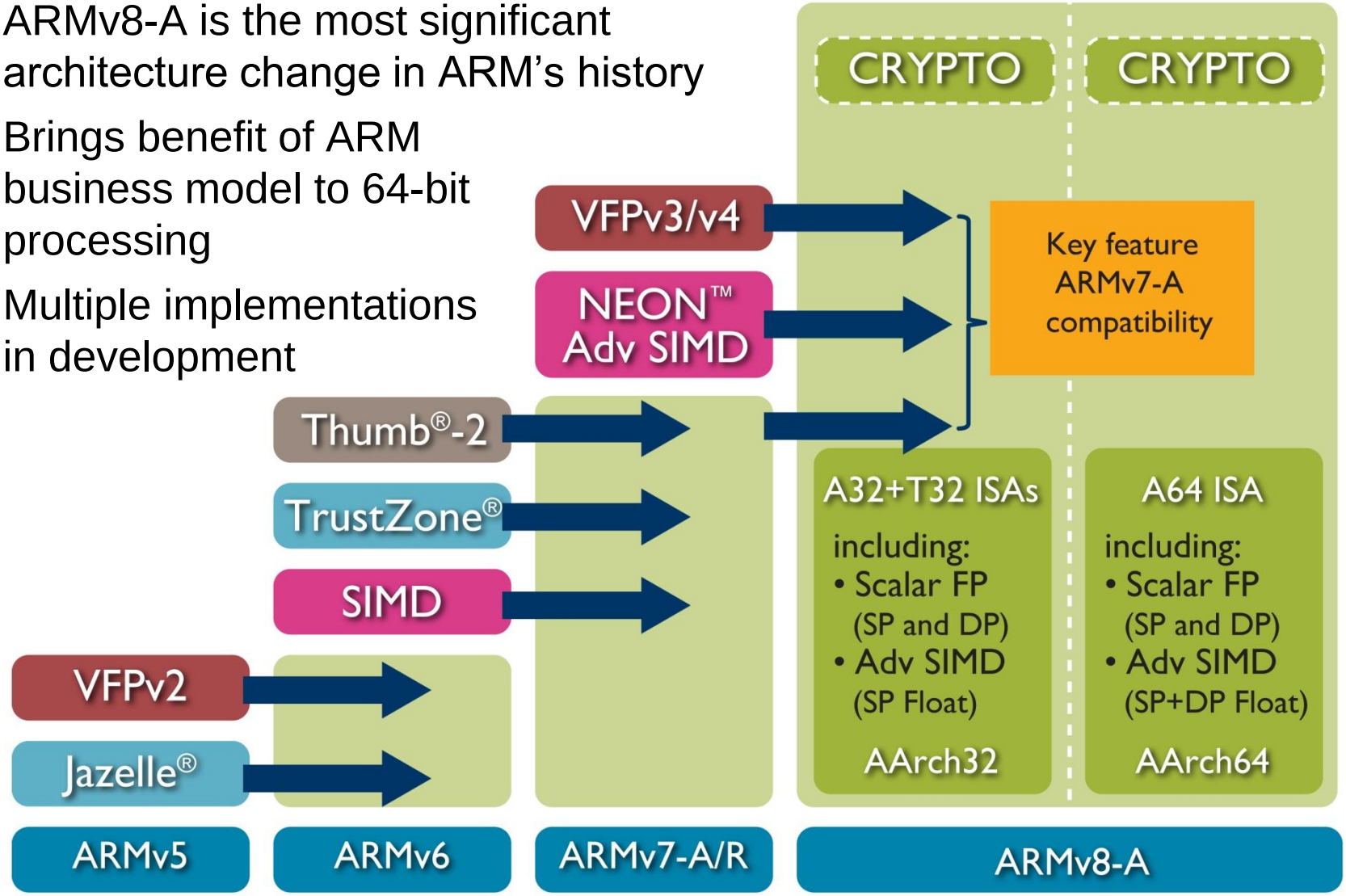
- 64-bit architecture alongside 32-bit
  - AArch64 state alongside AArch32 state
    - Modern instruction set for 64-bit processing
    - 31 general-purpose 64-bit registers (2 × AArch32)
    - 32 SIMD&FP 128-bit registers (2 × AArch32)
  - Further instruction set evolution for new workloads
    - Load Acquire/Store Release instructions
    - IEEE 754-2008 enhancements
    - Cryptographic instructions (SHA/AES)
    - Cyclic Redundancy Check (CRC32) instructions

- Enhancements carried into AArch32
  - Relatively small scale additions taken from AArch64
  - Maintaining full compatibility with ARMv7

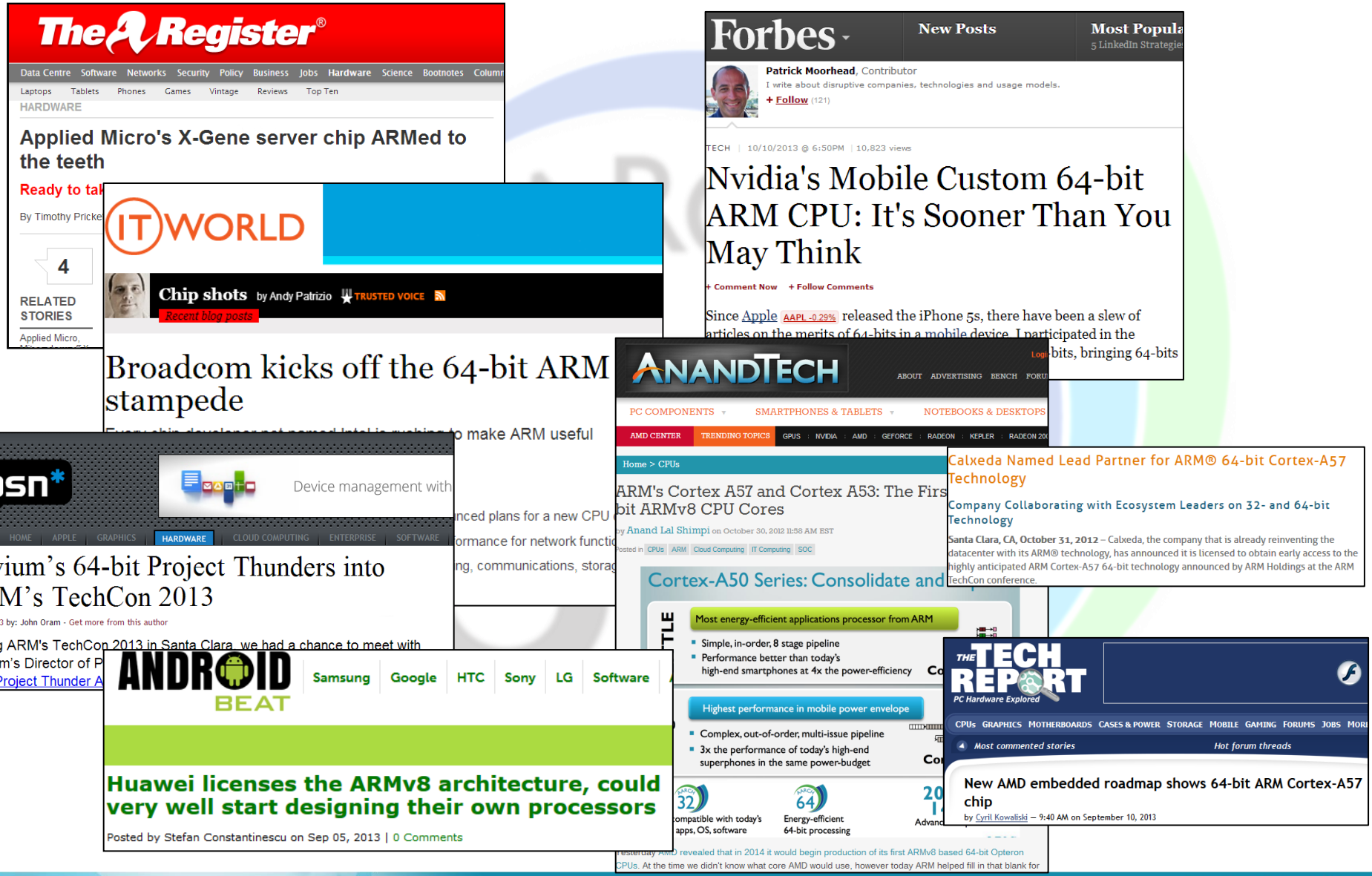
- Focus on power efficient architecture advantages in both states

Evolution of the ARM architecture (3)

- ARMv8-A is the most significant architecture change in ARM's history
- Brings benefit of ARM business model to 64-bit processing
- Multiple implementations in development

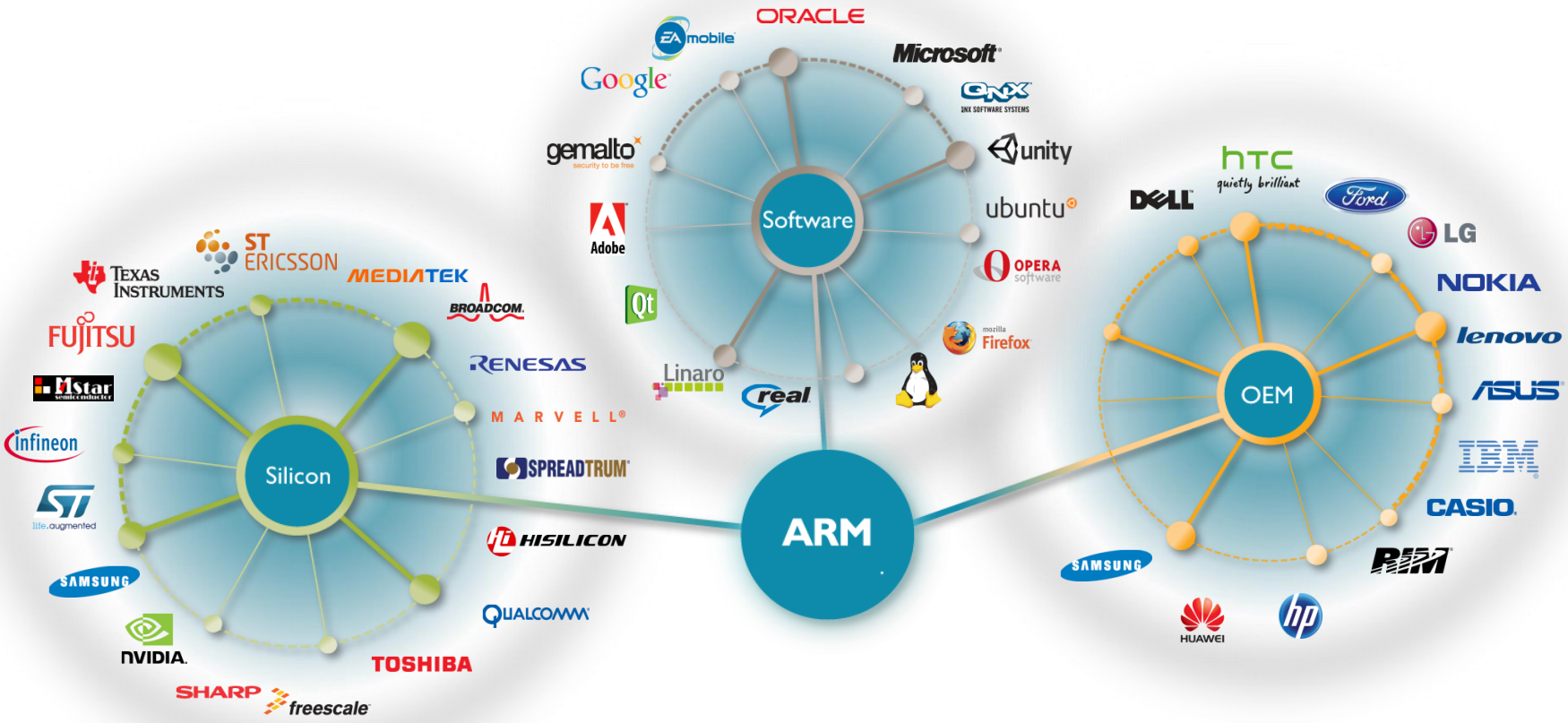


ARMv8 diversity

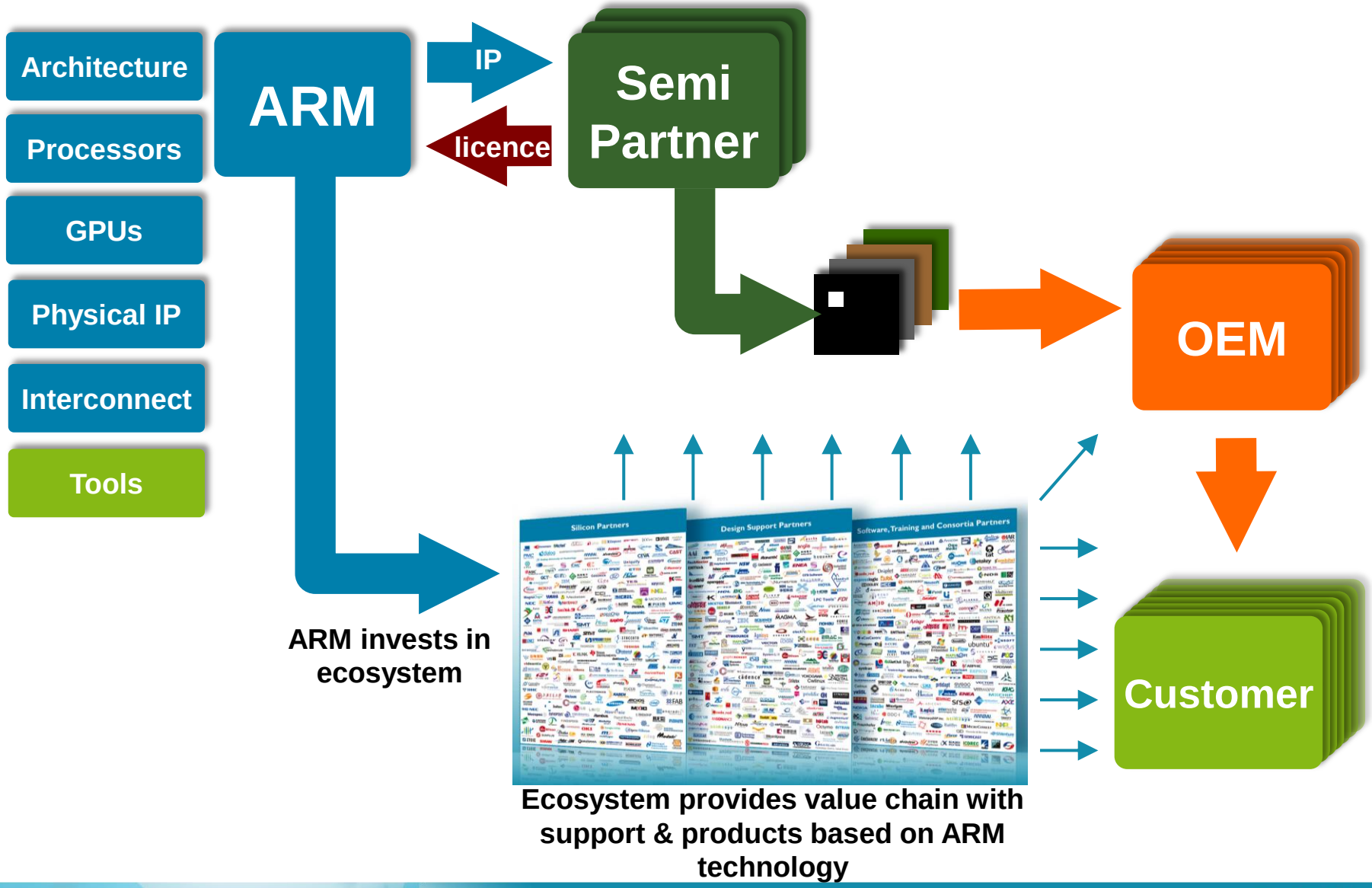


Diversity through partnership

Partnership drives successful ecosystems



ARM business models



Enabling efficiency everywhere

applied micro
AMD
CALXEDA
Power your tomorrow™
CAVIUM
NVIDIA
MARVELL®

BOSTON
Servers · Storage · Solutions
DELL
HUAWEI
hp
PENGUIN COMPUTING
EXPERTS IN HIGH PERFORMANCE COMPUTING SOLUTIONS
MiTAC
zt Systems

redhat
OPEN
Compute Project
openstack
CLOUD SOFTWARE
ubuntu

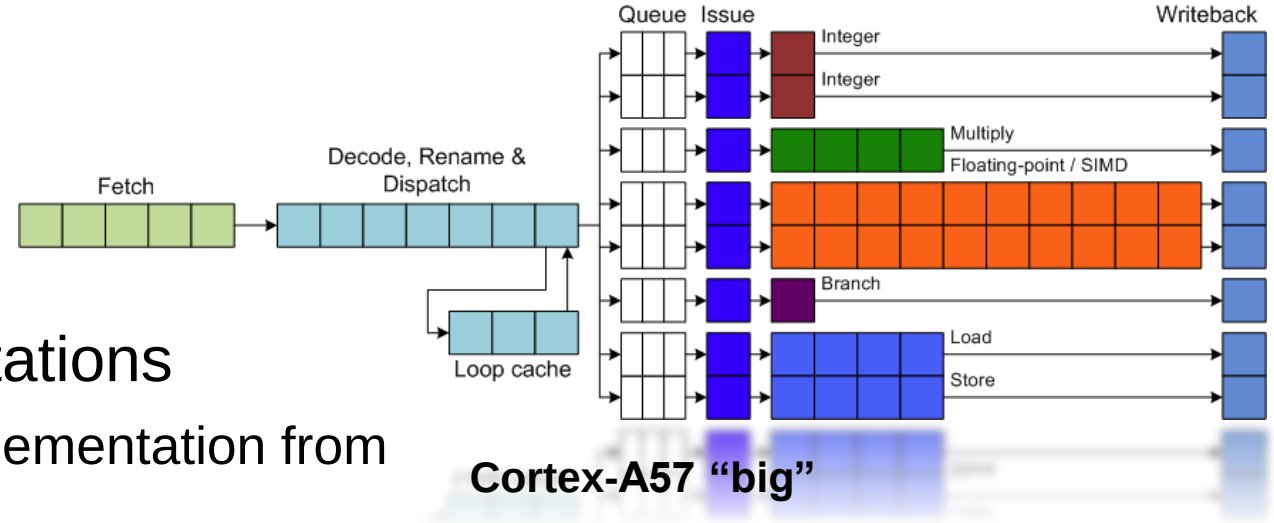
freescale
HUAWEI
LSI
MINDSPEED
Texas Instruments

"ARM Chips to leap from Smartphone to Networks that run them"

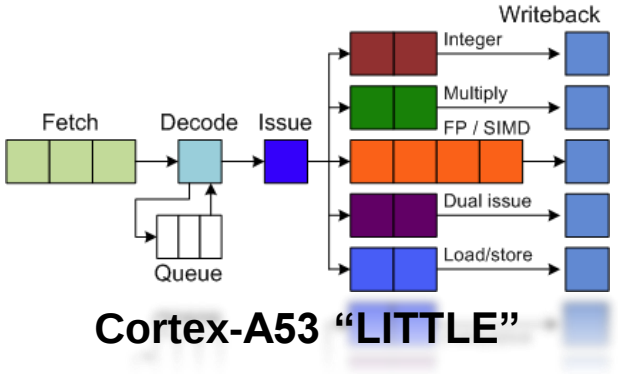
Bloomberg News

ARMv8 diversity

- Many implementations
  - big.LITTLE implementation from ARM
  - Implementations from ARM architecture partners
- Performance tuning challenge
  - HPC workloads more targeted at “big” core
  - Sea of “LITTLE” cores for servers and highly parallel workloads



Cortex-A57 “big”



Cortex-A53 “LITTLE”

Performance optimization on ARM

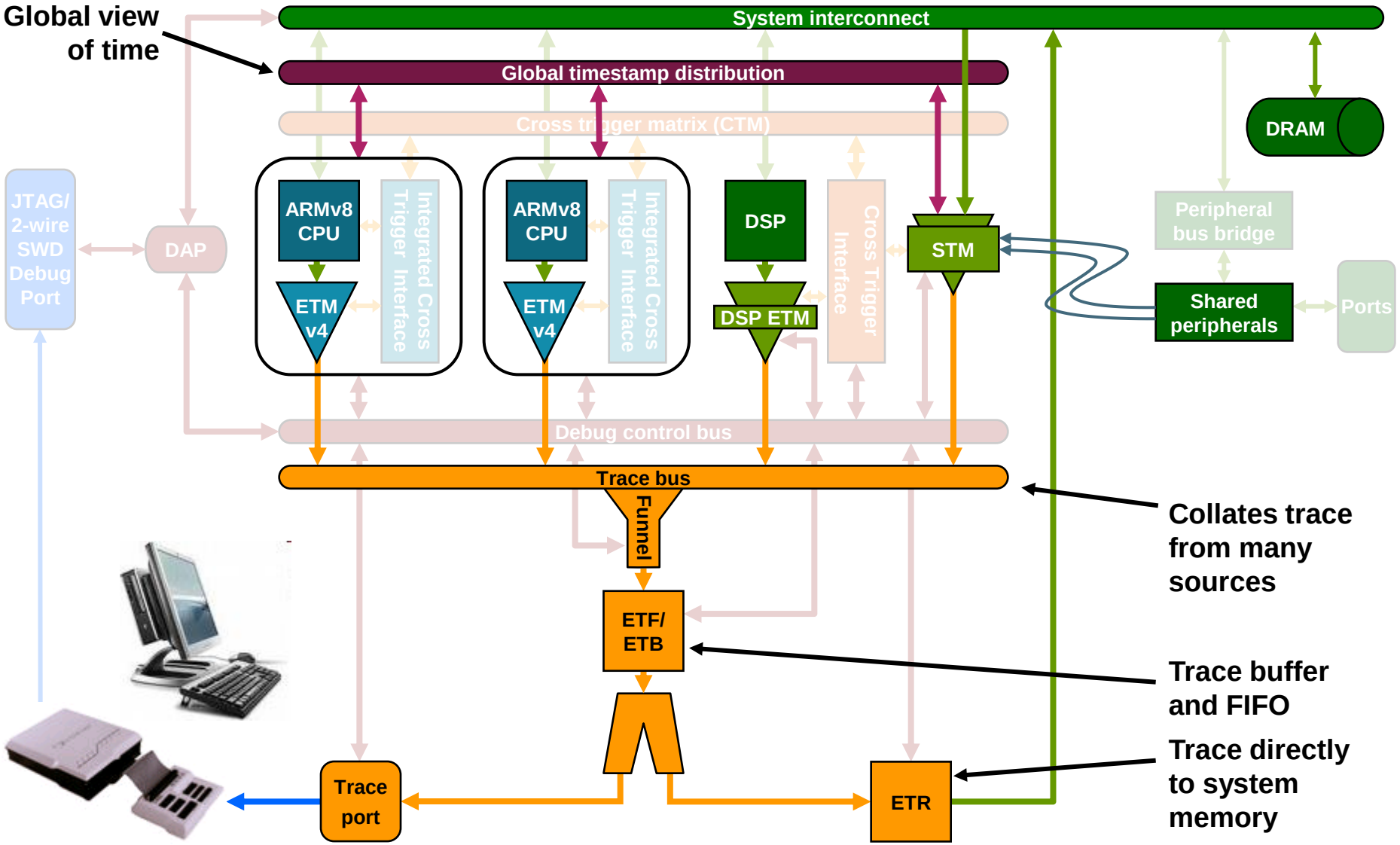
- The ARM architecture has mobile in its DNA
  - Low-power
- Basic mobile is an embedded system
  - SoC with many accelerators and coprocessors
  - Expensive JTAG probe debug tools
  - Full observability of system behavior
  - Small code footprints
  - Focus on code density over performance and features
- Higher performance requires a different development style
  - Smart phones, smart TVs, smart appliances, …
  - Enterprise systems, backhaul routers, HPC, …

Embedded trace

- Embedded trace gives full visibility over instruction flow
  - Trace every instruction or every branch
  - Cycle counting
  - Complex filtering and triggering
- Allows for trace based profiling
  - Detailed analysis of function run times and coverage
  - Graphical analysis of variables changing over time
- Widely used in mobile and real-time platforms

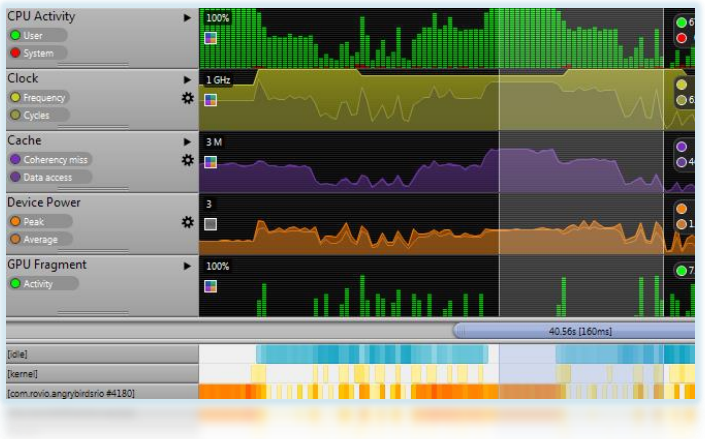
- But does not scale to large systems
  - ~0.5 bits of trace per instruction → ~25Gbyte/s for 128 cores @ 3GHz
  - Even if you could get data off chip (you can't), you could not process it in real time
    - (Perhaps CERN could – most people could not)
  - Best case for trace is as a sampling tool

CoreSight

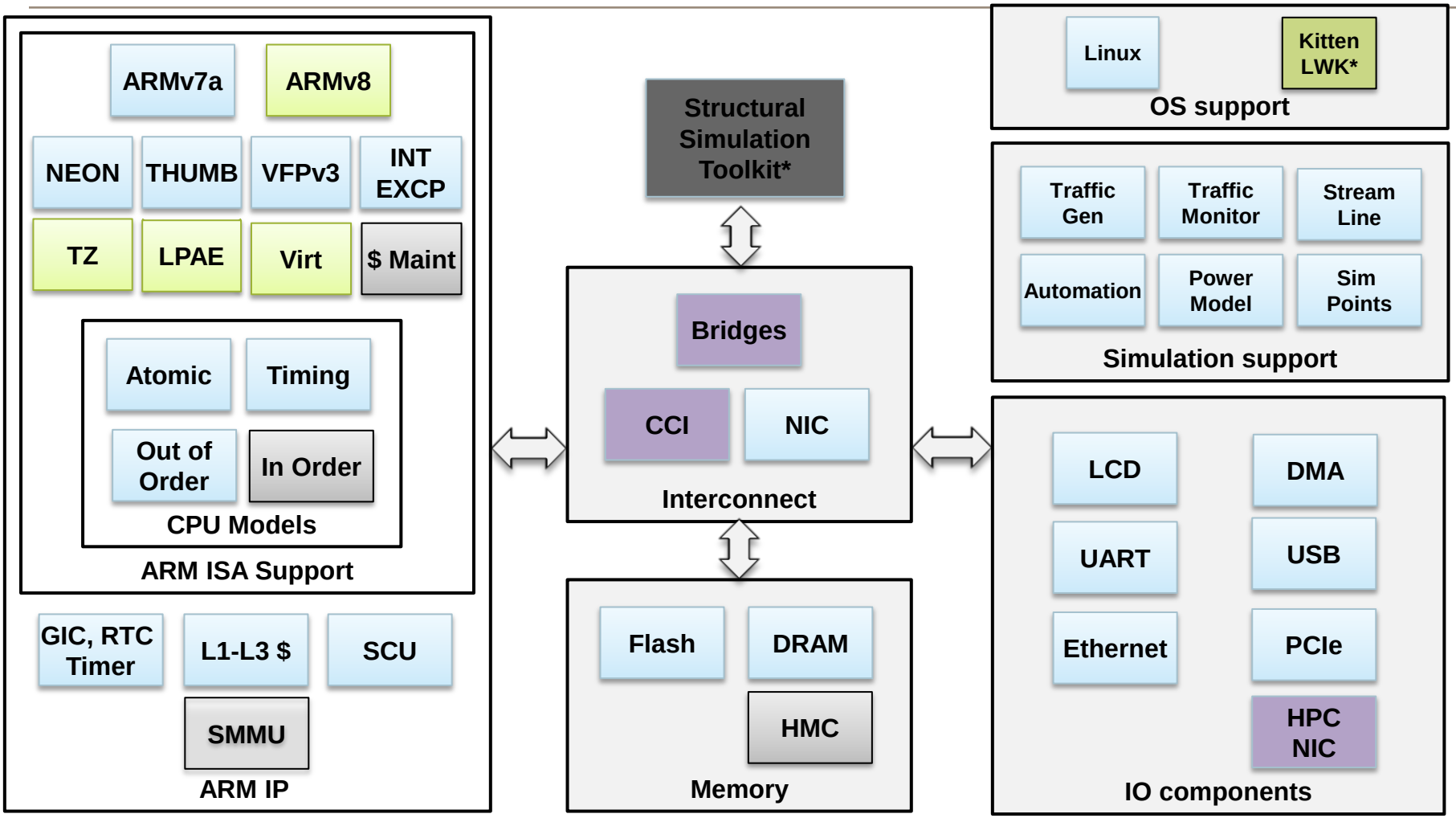


Performance monitoring on ARM

- ARMv8 defines first generation hardware performance counters
  - Between 2 and 31 (6 typical) 32-bit event counters + 64-bit cycle counter
    - (Configurable) interrupt on overflow
  - Each event counter is independently configurable (no constraints)
    - Type of event
    - Filtering by privilege level
  - ~32 architecturally-defined common (micro)architectural event types
    - Many implementation-defined event types
  - Operating system can choose to expose PMU to applications
- Supported in Linux perf
  - Support for uncore CCI-400 PMU
  - Exploring drivers for trace
- *Streamline Performance Analyzer* from ARM



Performance modeling with System Explorer



Done

In Process

Planned

Potential Areas for Future Work

More Info: gem5.org

*SST and Kitten are being developed by Sandia National Labs.

Evolving the ARM architecture

- ARMv8 is not the end point for the ARM architecture
  - Many areas of investigation for future development

- Performance measurement evolving as part of that architecture
  - Enhanced profiling of the CPU
  - Use of embedded trace as a profiling tool
  - Extending to cover complex SoC
    - Memory controllers, interconnects, system caches, etc.

Thank you

Questions?