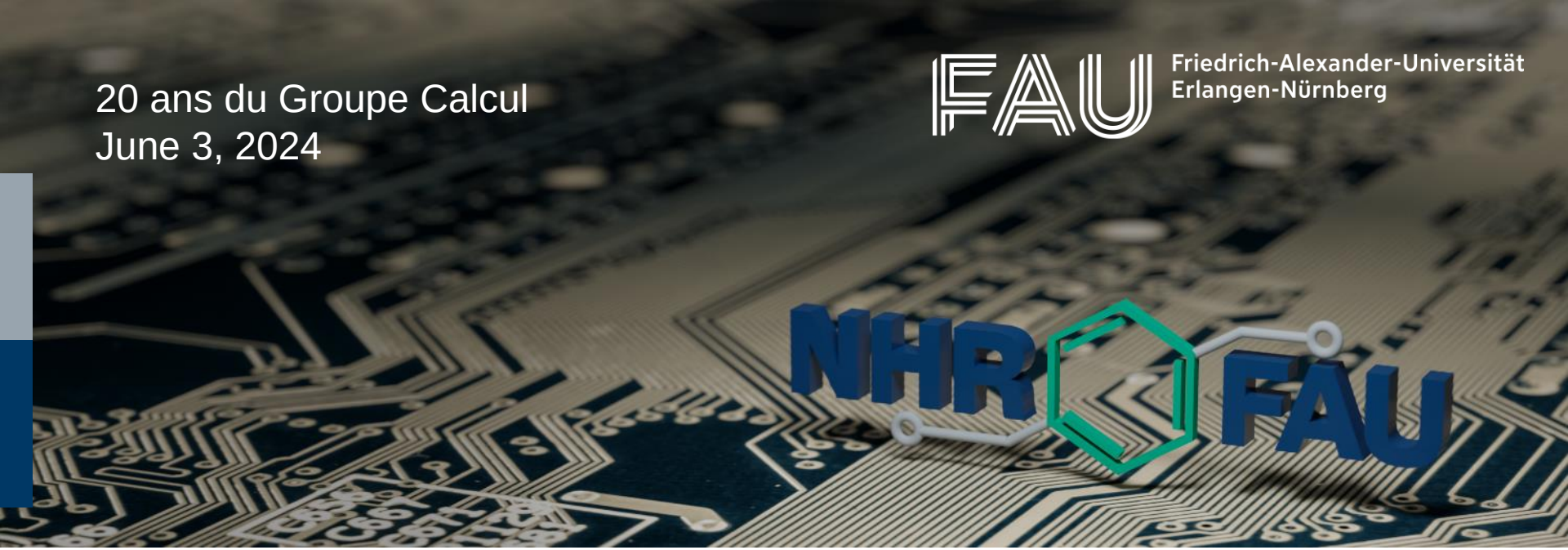
20 ans du Groupe Calcul
June 3, 2024



Hardware Evolution from an HPC Point of View

Georg Hager

Erlangen National High Performance Computing Center (NHR@FAU)

Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany

Supercomputing

“A supercomputer is a computer that is just one generation behind the requirements of the large-scale users”

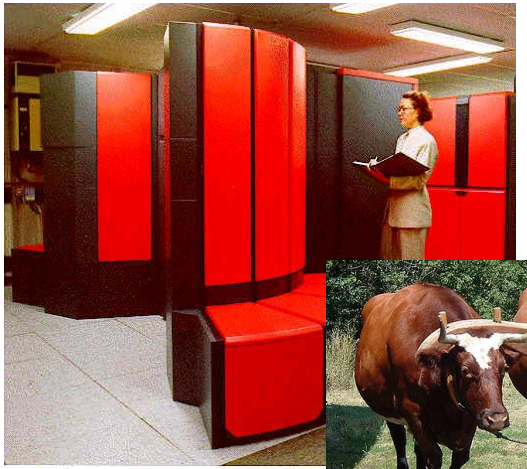
Neil Lincoln (CDC), 1981

If you were plowing a field, which would you rather use? Two strong oxen or 1024 chickens?

Seymour Cray

High Performance Computing History

Old-school stuff: oxen (1976-)



Enter the chickens (1985-)

High Performance Computing History

Commodity galore and
the reign of “good enough” (1995-)

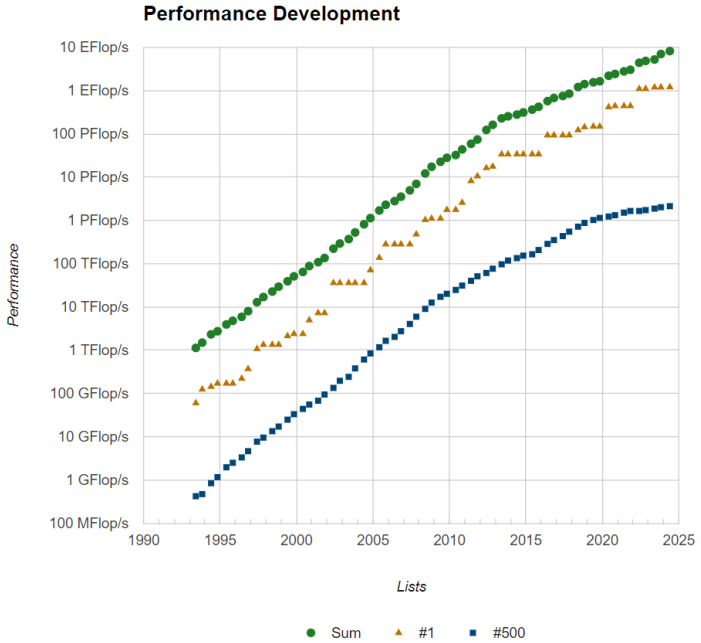
© D. Bader

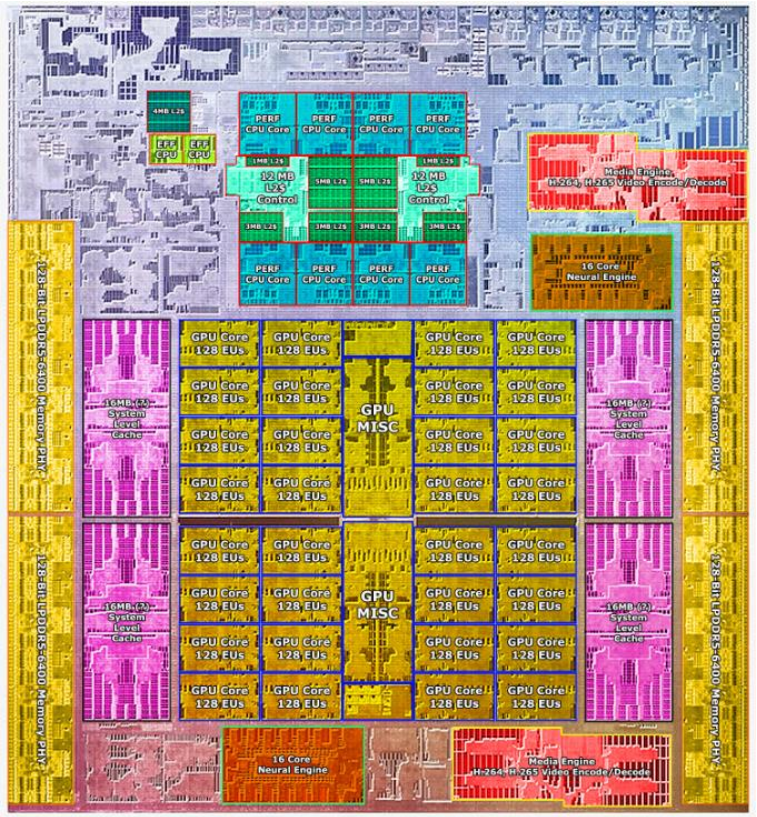
© OLCF

Graphics goes HPC (2007-)

High Performance Computing History

The death of Moore's Law (whenever)





© @Locuza_

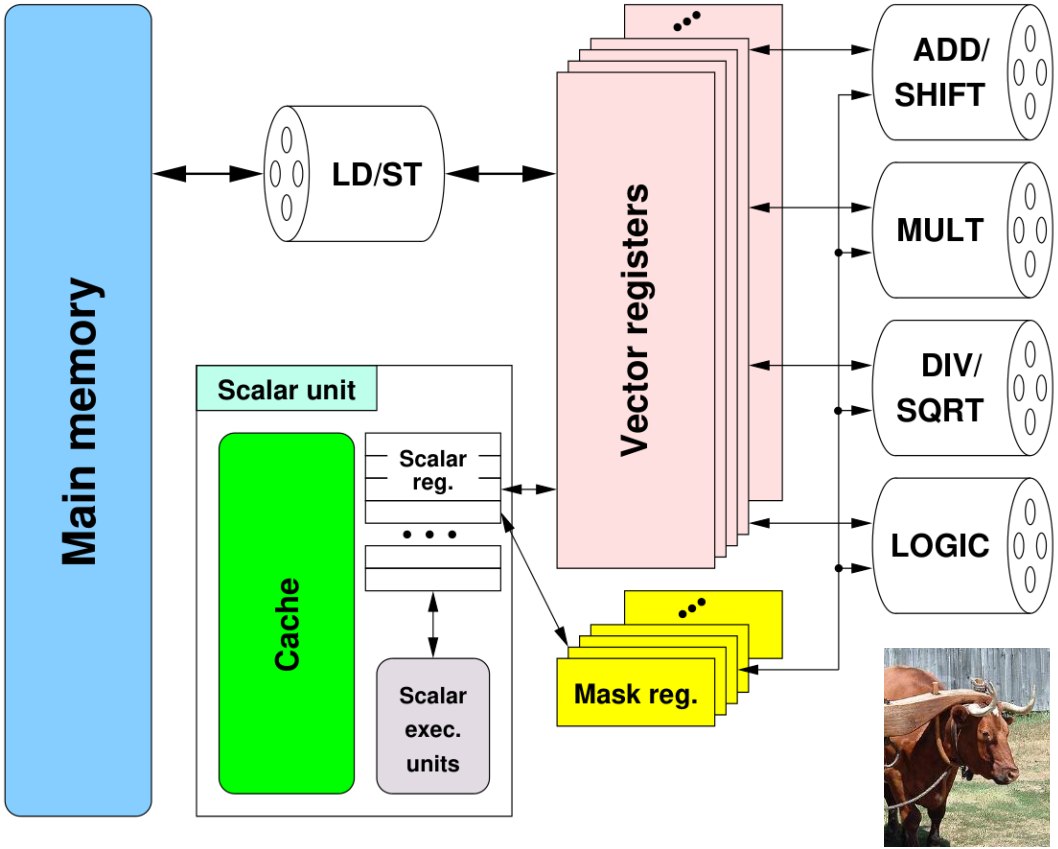
The Oxen

Vector CPUs

- SIMD instruction set
- Multi-(track-)pipelines
- Considerable memory bandwidth
- High clock speeds (originally)

Design decisions

- Memory-memory vs. register machines
- Pipeline chaining
- Vector cache



Famous vector machines

ILLIAC-IV (~1975)

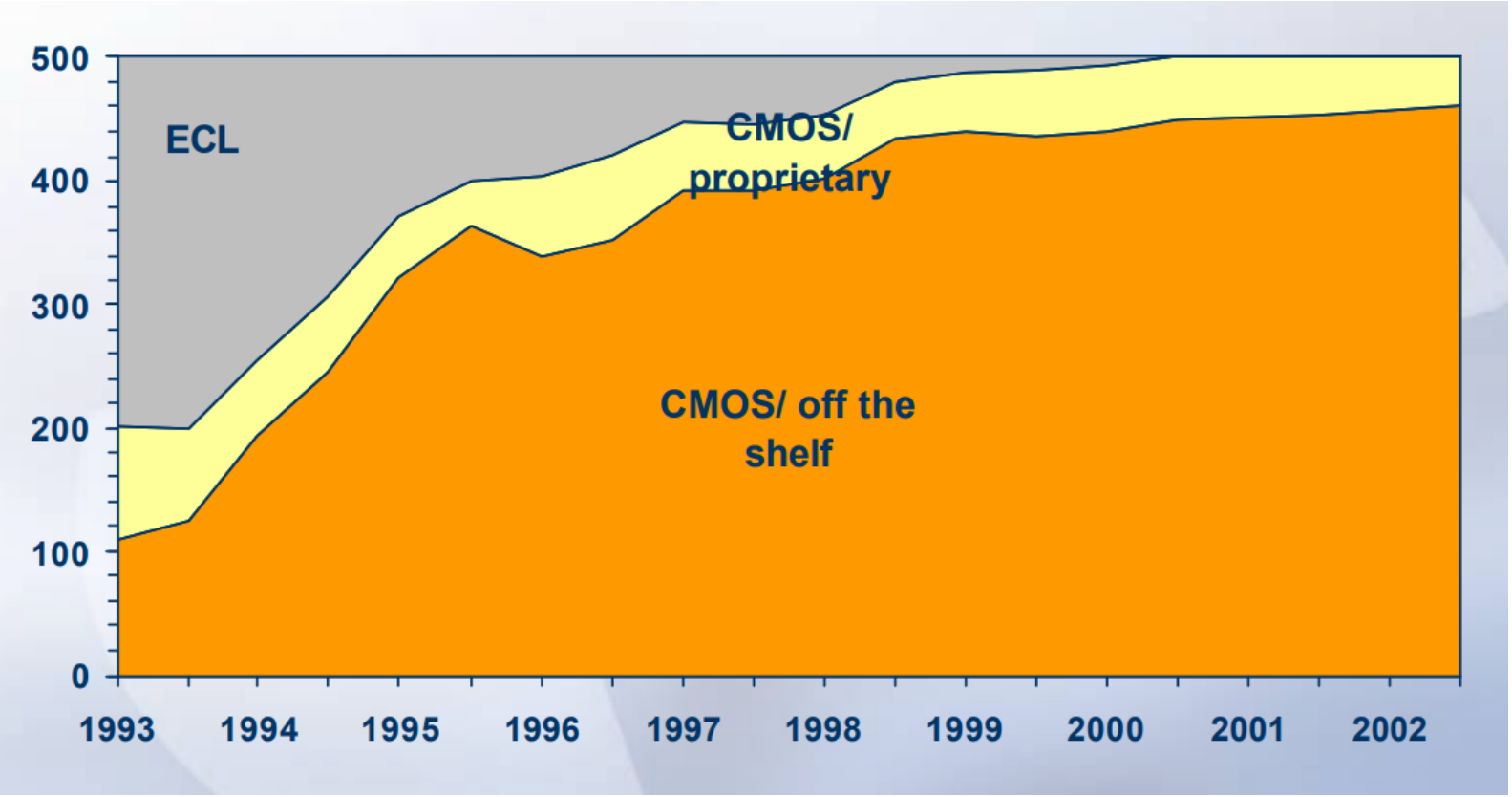
Cray Y-MP (1988)

© cray-history.net

Cray 1 (1976)

Cray X1 (2003)

From ECL to CMOS



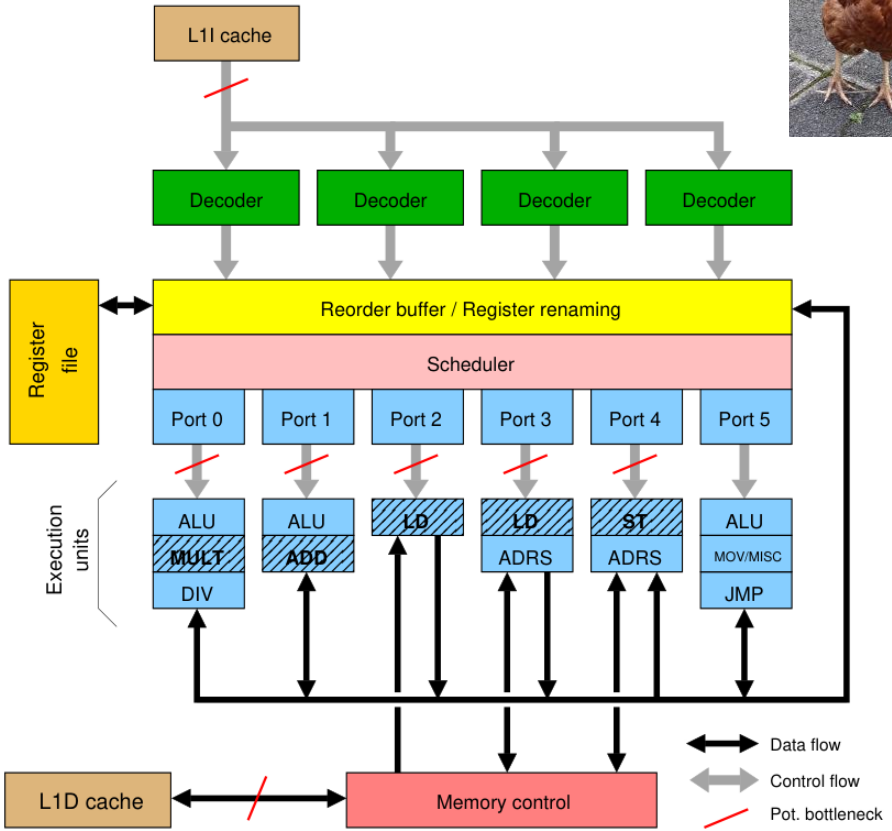
The Chickens

"Scalar" CPUs

- No or little vector capability
- Weak memory bandwidth
- Large caches

Improvements

- Out-of-order, superscalar, speculative execution
- Multicore
- "Wider" short-vector SIMD



Massively parallel computing

- 1985: CM-1
- 1991: CM-5
- 1993: IBM SP
- 1995: Cray T3E
- 1997: ASCI Red (Intel/SNL)
- 2000: Hitachi SR-8000
- 2003: Red Storm (Cray)
- 2004: Blue Gene (IBM)
- 2008: Roadrunner (IBM/LANL)

© J. McCranie

© ANL

© cray-cyber.org

Commodity Computing: The Beowulf

BEOWULF:
A PARALLEL WORKSTATION FOR SCIENTIFIC COMPUTATION

Thomas Sterling Donald J. Becker
Center of Excellence in Space Data
and Information Sciences
Code 930.5 NASA Goddard Space Flight Center
Greenbelt, MD 20771
{tron, becker}@cesdis.gsfc.nasa.gov

Daniel Savarese
Department of Computer Science
University of Maryland
College Park, MD 20742
dfs@cs.umd.edu

John E. Dorband
NASA Goddard Space Flight Center

Udaya A. Ranawake Charles V. Packer
Hughes STX Corp.

Abstract – *Network-of-Workstations technology is applied to the challenge of implementing very high performance workstations for Earth and space science applications. The Beowulf parallel workstation employs 16 PC-based processing modules integrated with multiple Ethernet networks. Large disk capacity and high disk to memory bandwidth is achieved through the use of a hard disk and controller for each processing module supporting up to 16 way concurrent accesses. The paper presents results from a series of experiments that measure the scaling characteristics of Beowulf in terms of communication bandwidth, file transfer rates, and processing performance. The evaluation includes a computational fluid dynamics code and an N-body gravitational simulation program. It is shown that the Beowulf architecture provides a new operating point in performance to cost for high performance workstations, especially for file transfers under favorable conditions.*

1 INTRODUCTION

development time and incurring increased cost. An alternative approach, adopted by the Beowulf parallel workstation project, recognizes the particular requirements of workstation oriented computation workloads and avoids the use of any custom components, choosing instead to leverage the performance to cost benefits not only of mass market chips but of manufactured subsystems as well. The resulting system structure yields a new operating point in performance to cost of multiple-processor workstations.

2 BEOWULF ARCHITECTURE

The Beowulf parallel workstation project is driven by a set of requirements for high performance scientific workstations in the Earth and space sciences community and the opportunity of low cost computing made available through the PC related mass market of commodity subsystems. This opportunity is also facilitated by the availability of the Linux operating system [7], a robust Unix-like system environment with source code that is targeted

PVM
16x Intel 486DX
10 Mbit Ethernet hub

D. J. Becker et al., ICPP 1995

Competitive Commodity Clusters from the late 90s

- Reasonable networking options (Myrinet)
- x86 gaining traction (Pentium II/III/4, AMD)
- Open-source software availability
  - MPI via MPICH
  - GCC
  - OpenPBS/Torque
  - ATLAS

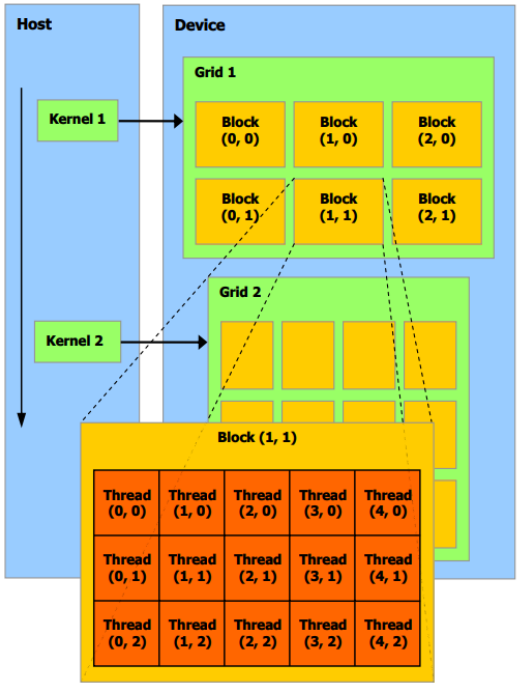
- Computing continuum spanning from small workstation clusters to Top500 systems

© D. Bader 1999

MPICH+PVM
128x dual Intel Pentium II
Myrinet/SAN

Graphics Goes HPC

2007: CUDA 1.0 released



© NVIDIA

We Were Mystified…

Demystifying GPU Microarchitecture through Microbenchmarking

Henry Wong, Misel-Myrto Papadopoulou, Maryam Sadooghi-Alvandi, and Andreas Moshovos
Department of Electrical and Computer Engineering, University of Toronto
{henry, myrto, alvandim, moshovos}@eecg.utoronto.ca

***Abstract*—Graphics processors (GPU) offer the promise of more than an order of magnitude speedup over conventional processors for certain non-graphics computations. Because the GPU is often presented as a C-like abstraction (e.g., Nvidia's CUDA), little is known about the characteristics of the GPU's architecture beyond what the manufacturer has documented. This work develops a microbechmark suite and measures the CUDA-visible architectural characteristics of the Nvidia GT200 (GTX280) GPU. Various undisclosed characteristics of the processing elements and the memory hierarchies are measured. This analysis exposes undocumented features that impact program performance and correctness. These measurements can be useful for improving performance optimization, analysis, and modeling on this architecture and offer additional insight on the decisions made in developing this GPU.**

I. Introduction

The graphics processor (GPU) as a non-graphics compute processor has a different architecture from traditional sequential processors. For developers and GPU architecture and compiler researchers, it is essential to understand the architecture of a

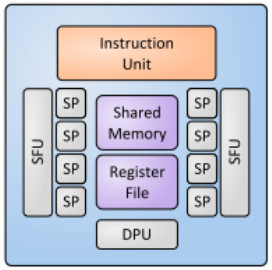


Fig. 1: Streaming Multiprocessor with 8 Scalar Processors Each

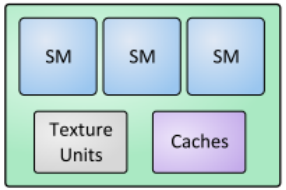
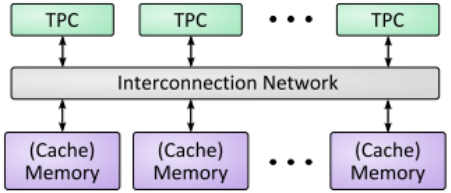
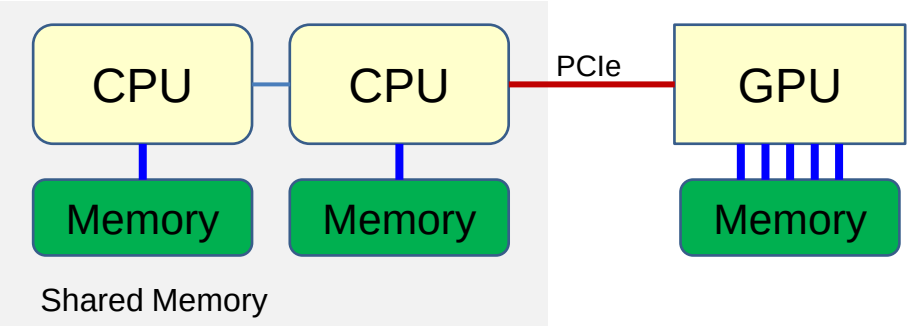


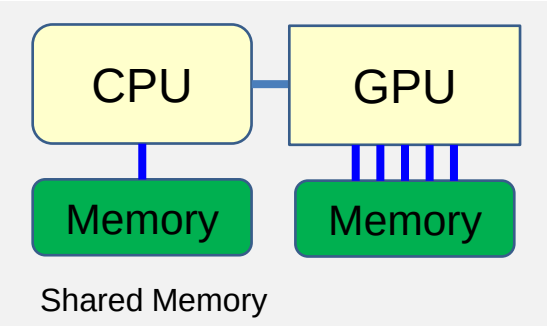
Fig. 2: Thread Processing Cluster with 3 SMs Each



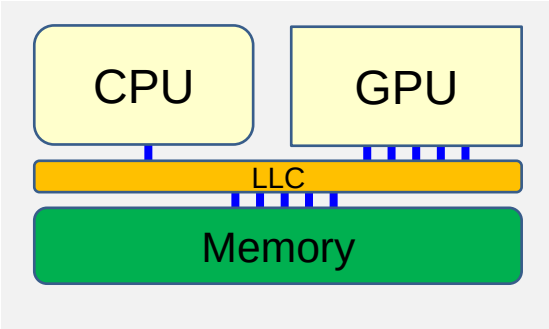
[H. Wong et al., 2010]

Evolution of CPU-Accelerator System Architecture





NVIDIA Grace Hopper (2023)



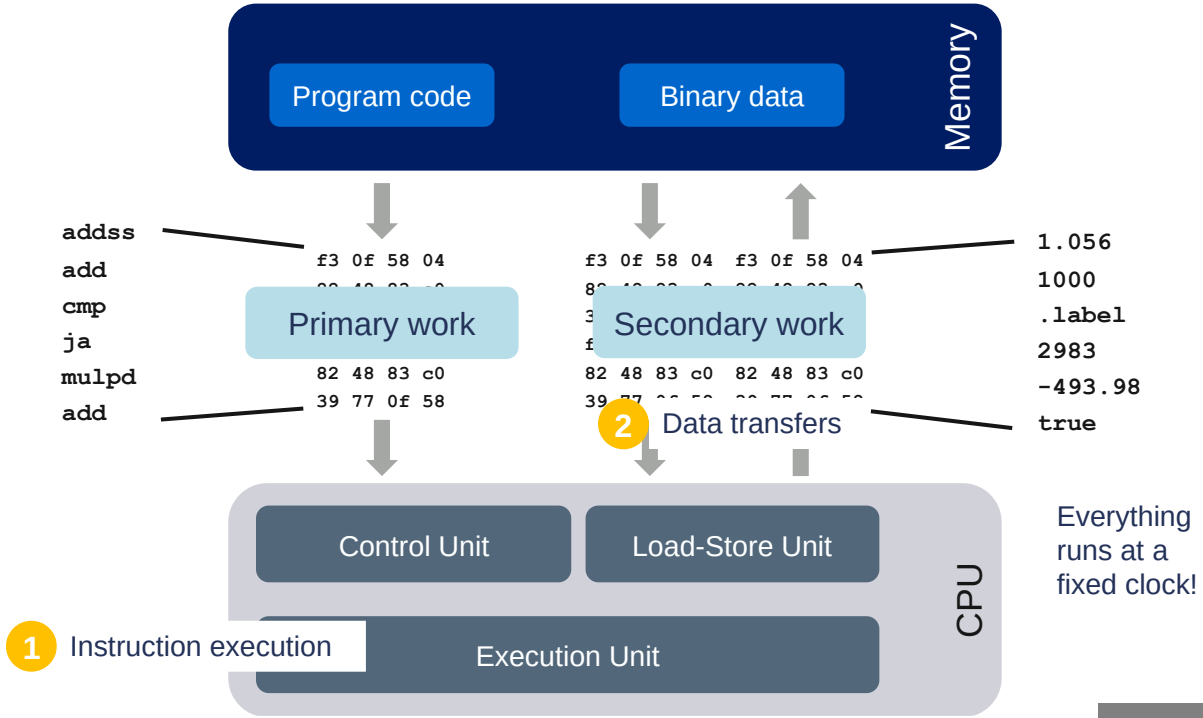
AMD MI300A (2023)

NHR FAU

FAU Friedrich-Alexander-Universität Erlangen-Nürnberg

A Closer Look at the Evolution of HPC Chips

A bird's eye view



Everything runs at a fixed clock!

Stored Program Computer

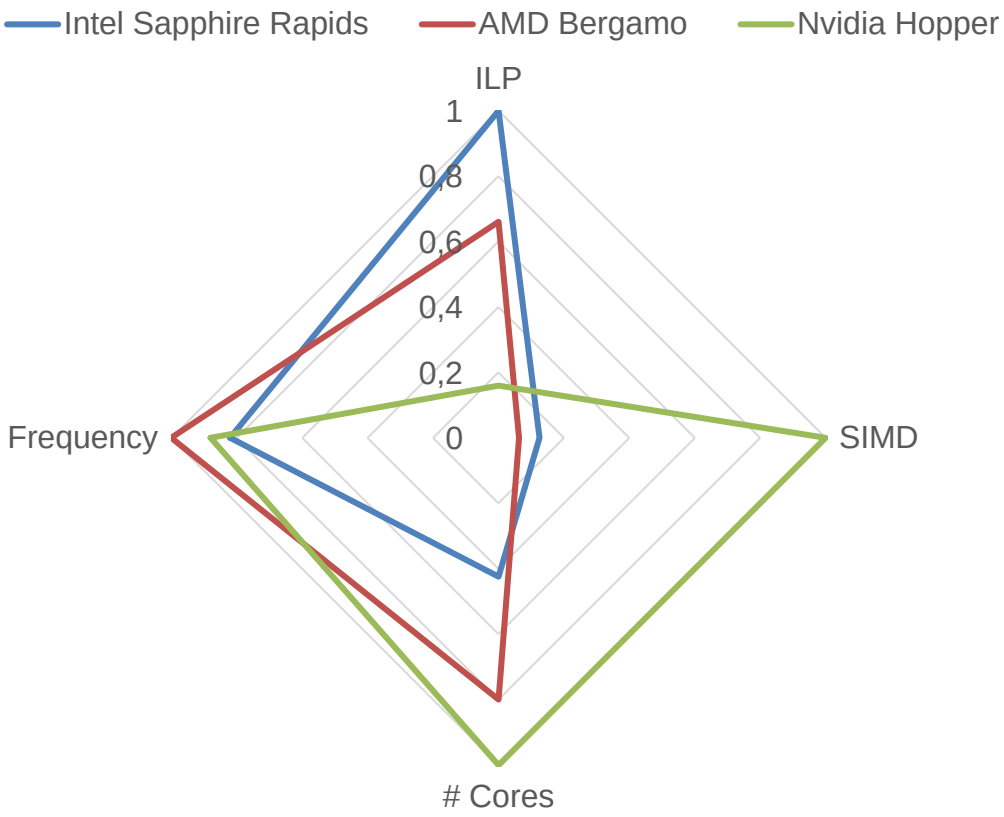
Focus on relevant software, technical opportunities, economic concerns, marketing concerns

Optimization strategies

- Increase clock speed
- Parallelism
- Specialization

Most hardware optimizations make assumptions towards the software!

Compromise within given power envelope



The chase for higher frequencies (1990 – 2005)

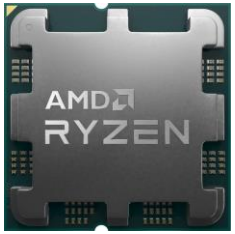
Enabled by advances in ILP technology and manufacturing processes

1990 x486
33 MHz
1000nm

1996 P200
200 MHz
350nm 15W

1999 PIII
1.1 GHz
180nm 33W

115x in 15 years!



2023 AMD Ryzen
4.5 GHz
5nm 170W 16c

2002 P4
2.5 GHz
130nm 61W

2005 P4
3.8 GHz
90nm 115W

Intel Pentium 4 introduced DP FP SSE2 (2001) and Hyperthreading (2002). You had to use SIMD instructions and parallelize your code!

SIMD in commodity CPUs (1999 – 2018)

A very old idea…

Today: “Short Vector SIMD”

- 1997: MMX (64bit, Integer only)
- 1998: 3DNow! (64bit, SP FP)
- 1999: SSE (64bit, SP FP)
- 1999: Altivec/VMX (128bit, SP FP)
- 2001: SSE2 (128bit, DP FP)
- 2011: AVX (256bit)
- 2016: AVX-512 (512bit)

Single-core DP floating-point performance

$$P_{core} = n_{super}^{FP} \cdot n_{FMA} \cdot n_{SIMD} \cdot f$$

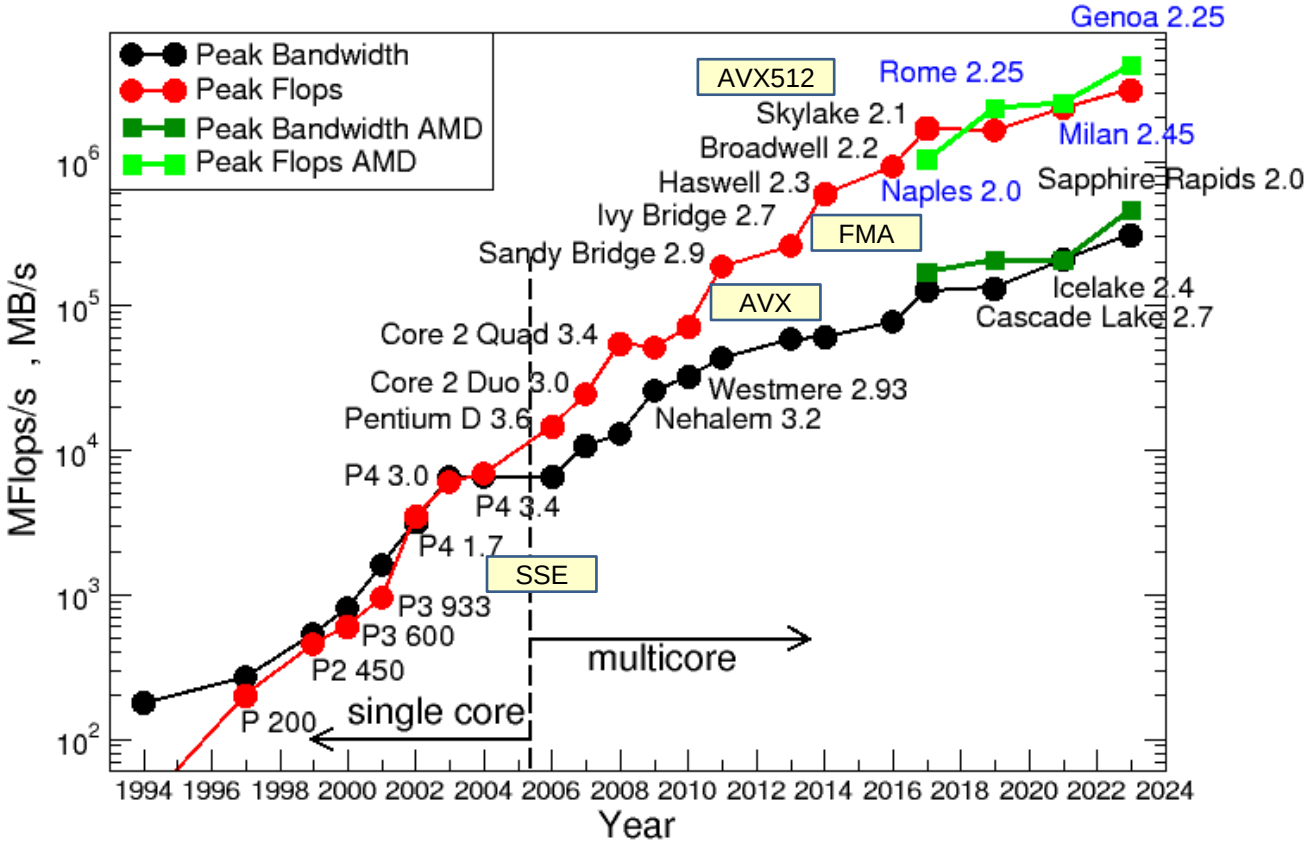
Super-scalarity | FMA factor | SIMD factor | Clock Speed

Typical representatives	n_{super}^{FP} [inst./cy]	n_{FMA}	n_{SIMD} [ops/inst.]	@market	Ex. model	f [Gcy/s]	P_{core}[GF/s]
Nehalem	2	1	2	Q1/2009	X5570	2.93	11.7
Sandy Bridge	2	1	4	Q1/2012	E5-2680	2.7	21.6
Haswell	2	2	4	Q3/2014	E5-2695 v3	2.3	36.8
Skylake	2	2	8	Q3/2017	Gold 6148	2.0	64
AMD Zen	2	2	2	Q1/2017	Epyc 7451	2.3	18.4
AMD Zen2	2	2	4	Q4/2019	Epyc 7642	2.3	36.8
Fujitsu A64FX	2	2	8	Q2/2020	FX700	1.8	57.6
IBM POWER10	8	2	2	Q3/2020	-	3.5	112 (?)
Apple Silicon	4	2	2	Q1/2023	M2	3.5	56
NVIDIA Grace	4	4	2	Q3/2023	Grace Superchip	3.2	51.2

Data Parallelism and the DRAM Gap (2006 – 2022)

SIMD allowed to sustain performance increases!

Gap not widening anymore?

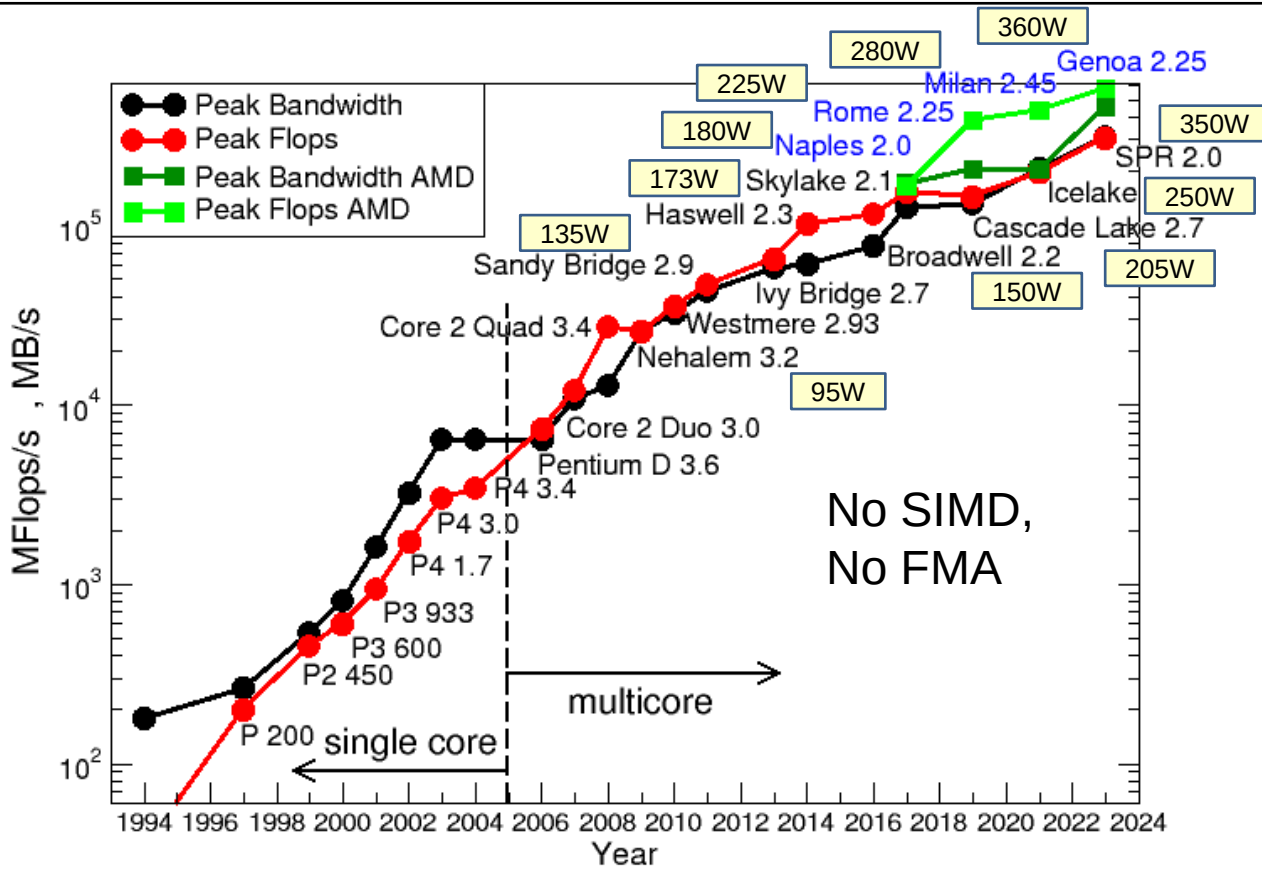


A Different View

"DRAM Gap" is driven by SIMD and FMA

Power dissipation becomes a major concern

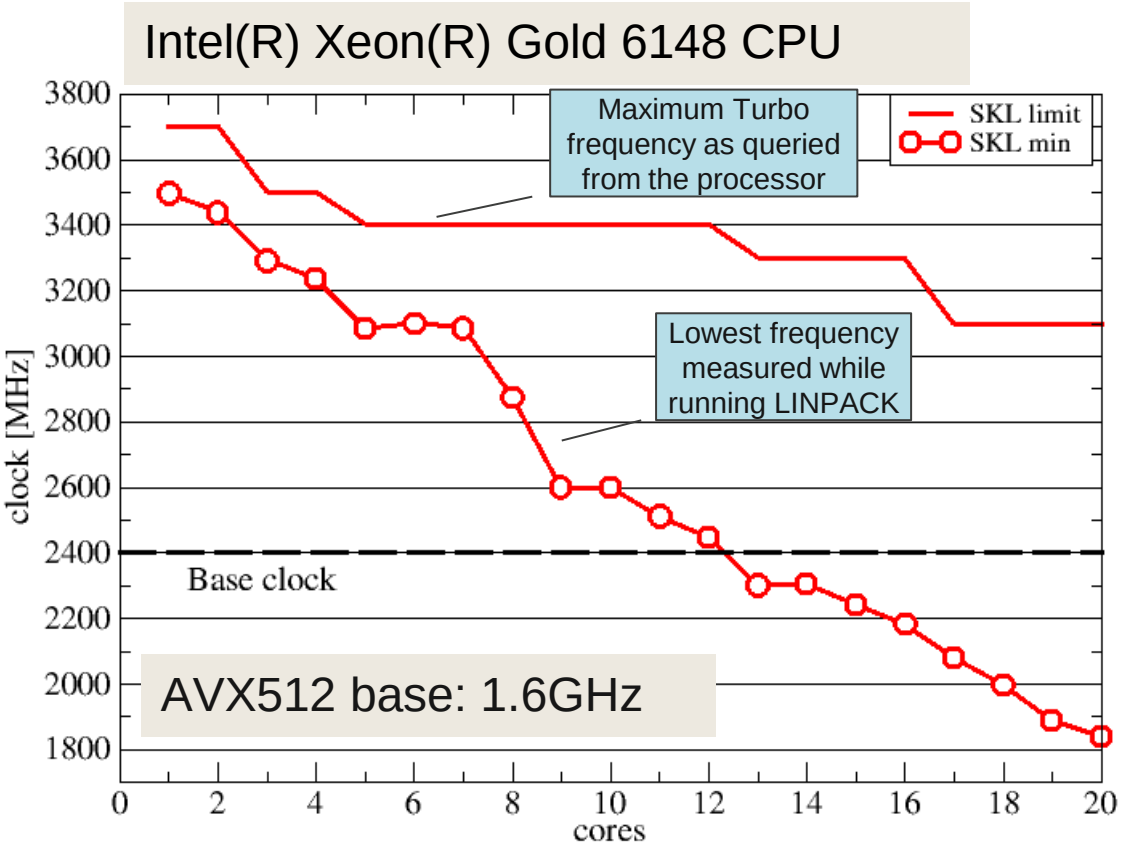
Multicore becomes manycore



Which clock?

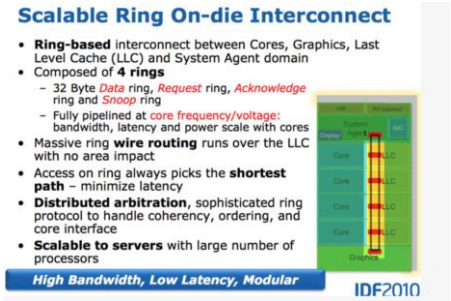
The processor dynamically overclocks to exploit more of the TDP envelope if fewer cores are active.

On Intel CPUs the base clock is meaningless!



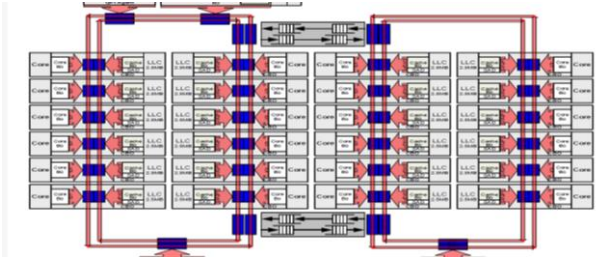
Scalable Core Interconnect?

Challenge: Implement scalable interconnect!

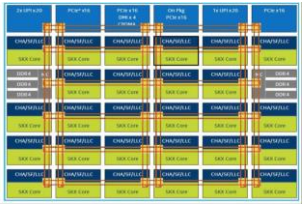


Solutions:

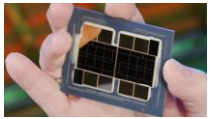
- Direct connection
- Ring bus
- Two of them



- Mesh



- Chiplets



100 billion transistors across 47 active tiles manufactured on five different process nodes

More games with factors …

uArch	n_{super}^{FP}	n_{FMA}	n_{SIMD}	n_{cores}	f_{simd}	Release	Model	P_{pkg} [GF/s]	TDP [W]	GF/W att
Sandy Bridge	2	1	4	8	2.9	2012	E5-2680	185	130	1.42
Haswell	2	2	4	18	2.0	2014	E5-2695-v3	576	120	4.80
Broadwell	2	2	4	22	2.2	2016	E5-2699-v4	774	145	5.3
Skylake	2	2	8	28	1.9	2017	8176	1702	165	10.3
Naples	2	2	4	32	2.0	2017		1024	180	5.7
Rome	2	2	4	64	2.25	2019		2304	225	10.2
Ice Lake	2	2	8	40	2.0	2021	8380	2560	270	9.5
Sapphire Rapids	2	2	8	56	1.9	2023	8470	3405	350	9.7
Bergamo	2	2	4	128	2.25	2023		4608	360	12.8
Apple Silicon	4	2	2	4+4	3.5	2023	M2	448	30	14.9
Grace	4	4	2	72	~3	2023	Gr. SC	3686	250*	14.7

What about main memory?

Same game with factors:

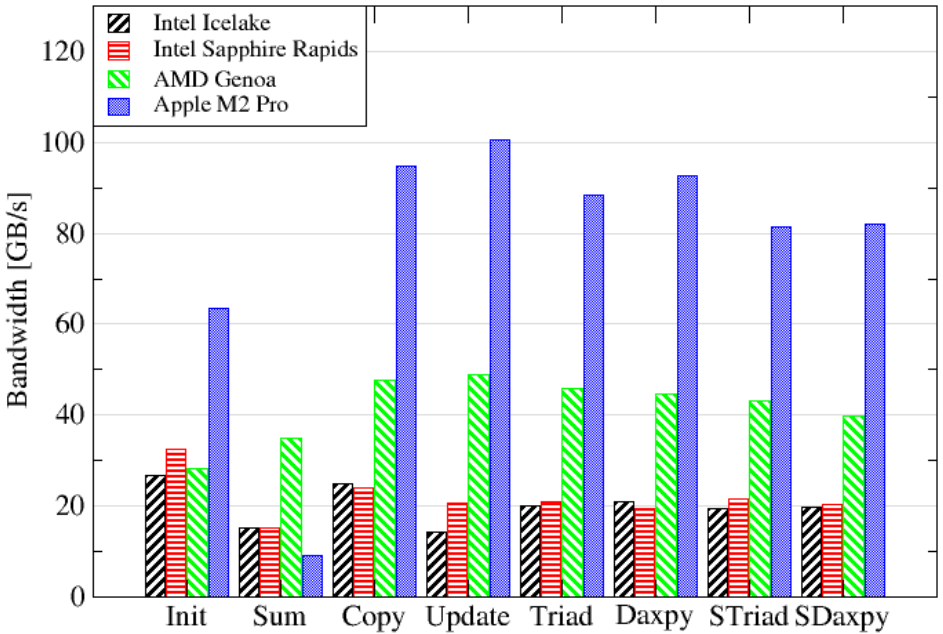
$$B_{pkg} = n_{interfaces} \cdot n_{channels} \cdot 8 \cdot f_{bus}$$

AMD Genoa: 4 interfaces, 3 channels each, DDR5-4800 (2.4GHz)
460.8 GB/s

Intel Sapphire Rapids: 4 interfaces, 2 channels each, DDR5-4800 (2.4GHz)
307.2 GB/s

Apple M2 Ultra: 4 interfaces, 4 channels each, LPDDR5-6400 (3.2GHz)
819.2 GB/s

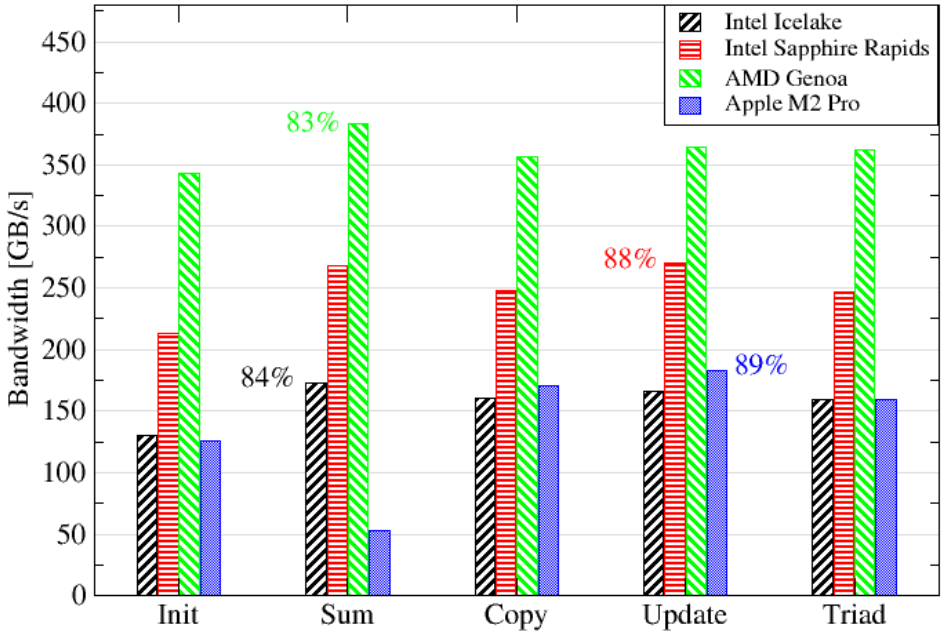
Single thread memory bandwidth



- Increasing number of load streams, with and without store miss.
- Intel architectures have history of low single thread bandwidth.
- AMD and Intel sequential bandwidth is a function of core architecture
- Apple achieves 50% of total bandwidth with single thread!

https://github.com/RRZE-HPC/TheBandwidthBenchmark

Package total memory bandwidth



- Total package bandwidth using all available cores
- AMD is far ahead of the competition (with DDR RAM)
- Apple bandwidth is on same level than previous generation Icelake

https://github.com/RRZE-HPC/TheBandwidthBenchmark

Accelerators

uArch	n_{super}^{FP}	n_{FMA}	n_{SIMD}	n_{cores}	f	Release	Model	P_{pkg} [GF/s]	TDP	GF/Watt
Sandy Bridge	2	1	4	8		2012	E5-2680	173	130	1,33
Pascal	1	2	32	56	1.480	2016	GP100	5304	300	17.68
Skylake	2	2	8	26	1.85	2017	8170	1581	165	9,58
Volta	1	2	32				GV100	8177	300	27.25
Ampere	1	2	32	1			GA100	9746	400	24.36
CDNA 1	1	2	32	1			MI100	11520	300	38.40
CDNA 2	1	2	64	2x110	1.70	2021	MI250x	47872	560	85.48
Hopper	2	2	32	132	1.980	2022	GH100	33450	700	47.78
Bergamo	2	2	4	128	2.25	2023	9754	4608	360	12.80
~~Ponte Vecchio~~	~~8~~	~~2~~	~~16~~	~~128~~	~~1.60~~	~~2023~~	~~Max 1550~~	~~52430~~	~~600~~	~~87.38~~

Double that in case you can use Tensor cores

Specialized execution engines

There already existed special purpose instructions for graphics and cryptography operations

Modern system-on-chip designs add execution engines

- Apple Neural Engine and Media Engine for video transcoding
- Nvidia Tensor cores
- Power 10 MMA engine
- Intel AMX

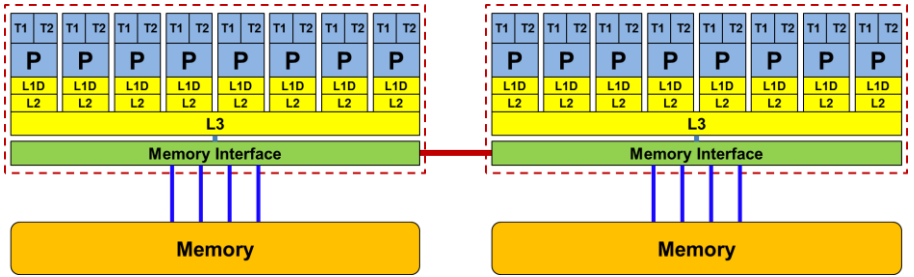
Going all in: Standalone accelerators for AI, autonomous driving and crypto

- Google TPU
- ARM MLP
- Tesla FSD

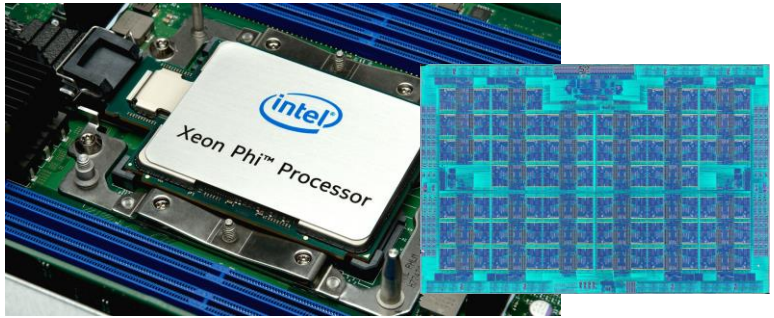
Notable success stories and failures

Itanium (2001 – 2020)

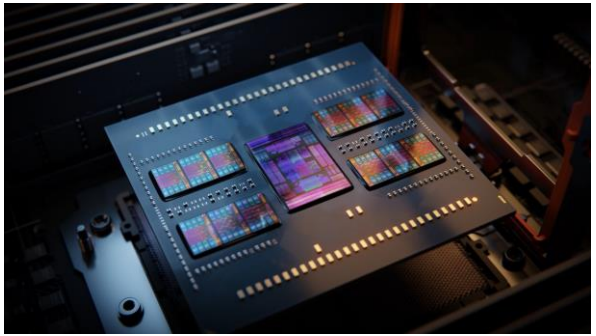
ccNUMA (2000 – now)



Xeon Phi (2012 – 2018)



Chiplets (2017 – now)



Outlook and predictions

- Extrapolation indicates that current technology is good for 2-3 more iterations
- Specialization will increase
- The boundary between “accelerator” and “CPU” will become more blurry on all scales
- “Traditional” PCIe-based accelerators will survive in the cost-effective regime

- Energy efficiency is driven by hardware innovation, not by playing around with clock speeds

NHR FAU

FAU Friedrich-Alexander-Universität Erlangen-Nürnberg

Thank You.