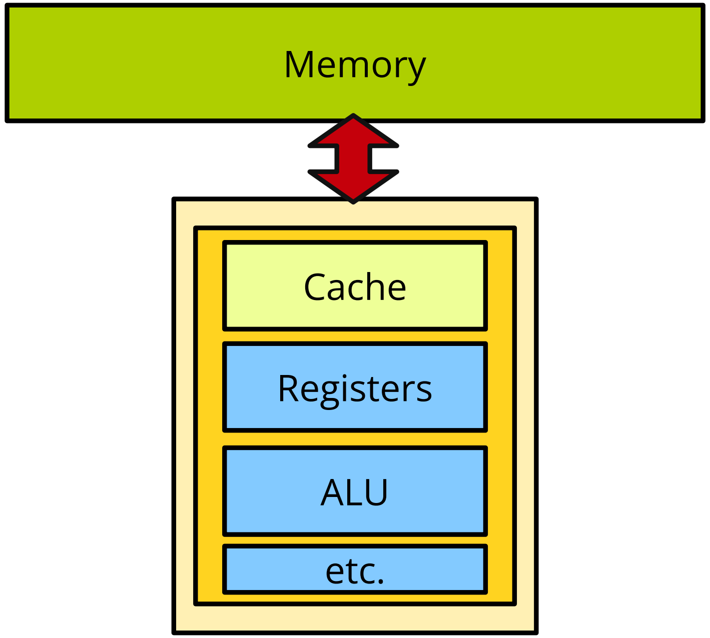
COMP35112 Chip Multiprocessors

Introduction 1

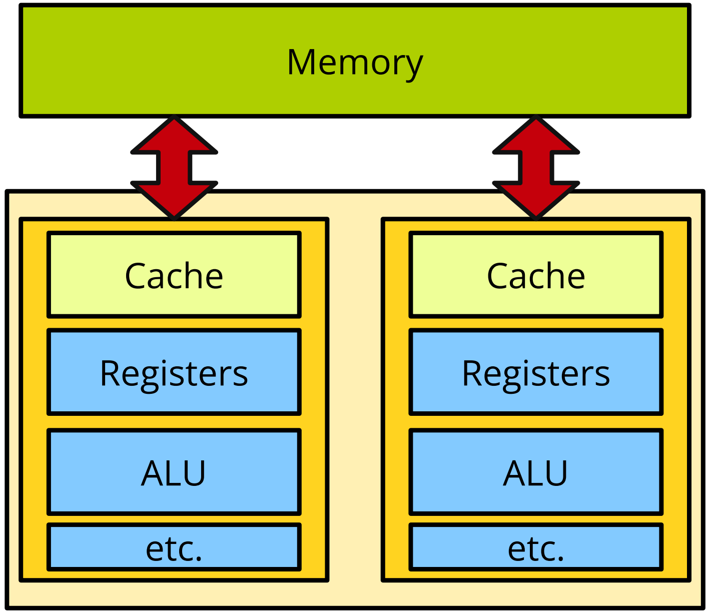
Pierre Olivier

Chip Multiprocessors?



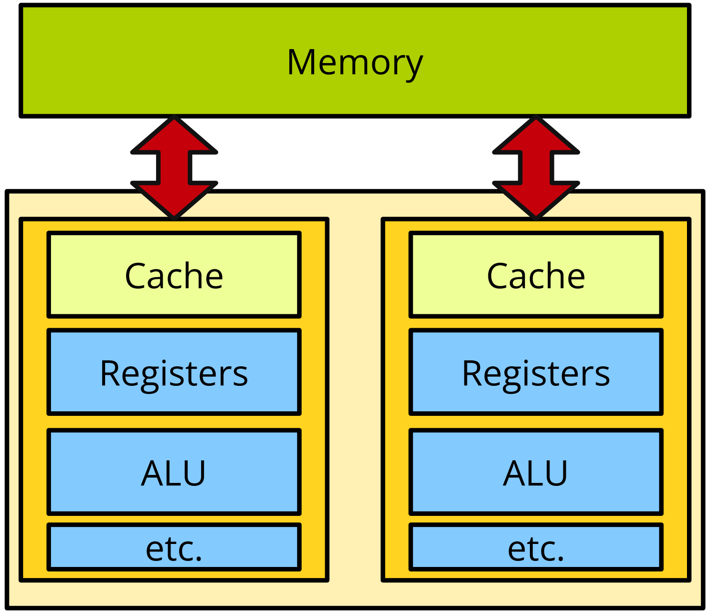
Uniprocessor or single-core CPU

Chip Multiprocessors?



Multiprocessor or multicore CPU

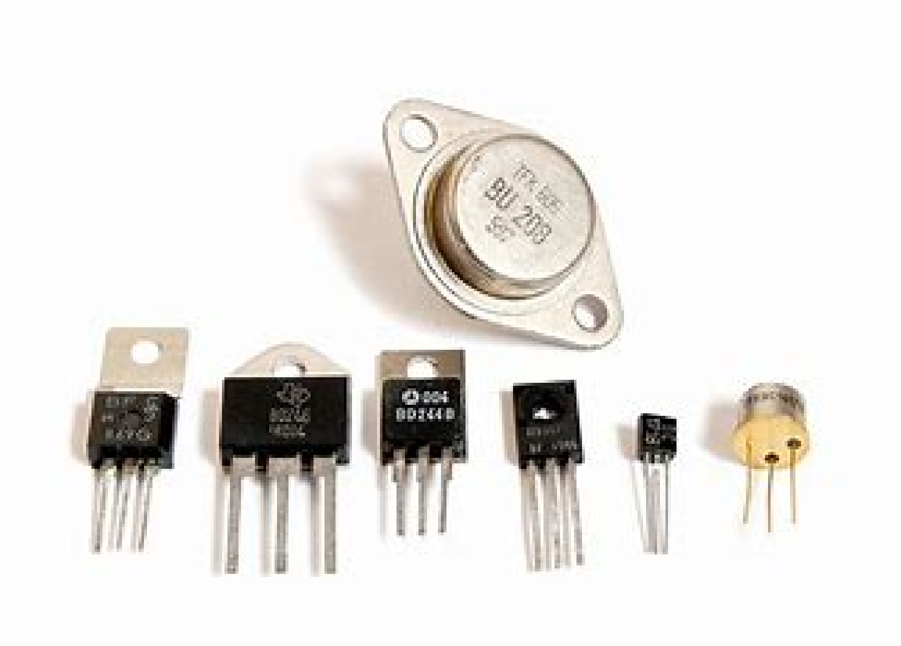
Chip Multiprocessors?



Multiprocessor or multicore CPU

Why was this invented?

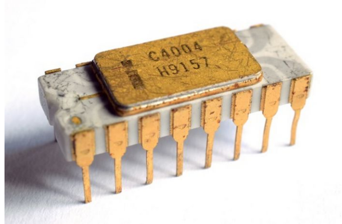
Chip Multiprocessors?



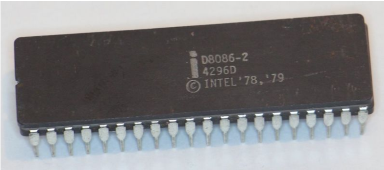
Chip Multiprocessors?

- Computing power of a CPU is function of the number of transistors it integrates

Time 1971 1978 1994 2001

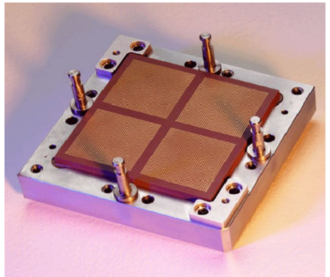


Intel 4004
1st commercial IC processor,
2,300 transistors



Intel 8086,
30K transistors

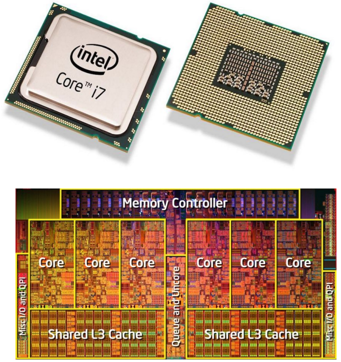
ARM7,
600K transistors



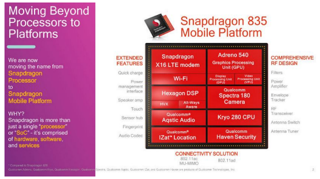
IBM Power4,
170M transistors,
2 cores

Chip Multiprocessors?

Time 2011 2017 2019 2023



6 cores Intel i7, ~1.1 Billion transistors



Moving Beyond Processors to Platforms

We are now moving the name from Snapdragon Processor to Snapdragon Mobile Platform

WHY? Snapdragon is more than just a single "processor" or "SoC" - it's comprised of hardware, software, and services

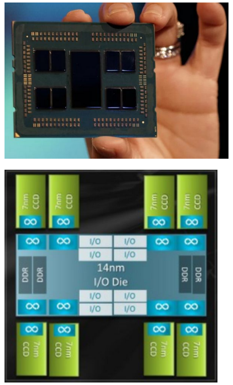
Snapdragon 835 Mobile Platform

EXTENDED FEATURES
Quick charge
Power management interface
Speaker amp
Touch
Sensor hub
Fingerprint
Audio Codec
Snapdragon X16 LTE modem
Wi-Fi
Hexagon DSP
HVX
All-Ways Aware
Qualcomm® Aqstic Audio
Qualcomm® IZat™ Location
Adreno 540 Graphics Processing Unit (GPU)
Display Processing Unit (DPU)
Video Processing Unit (VPU)
Qualcomm Spectra 180 Camera
Kryo 280 CPU
Qualcomm Haven Security
COMPREHENSIVE RF DESIGN
Filters
Power Amplifier
Envelope Tracker
RF Transceiver
Antenna Switch
Antenna Tuner
CONNECTIVITY SOLUTION
802.11ac MU-MIMO
802.11ad

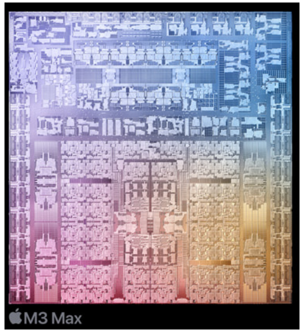
* Compared to Snapdragon 820

Qualcomm Adreno, Qualcomm Kryo, Qualcomm Hexagon, Qualcomm Spectra, Qualcomm Aqstic, Qualcomm IZat, and Qualcomm Haven are products of Qualcomm Technologies, Inc.

Qualcomm Snapdragon 835 SoC (Quad core, GPU, DSP...), ~3 Billion transistors



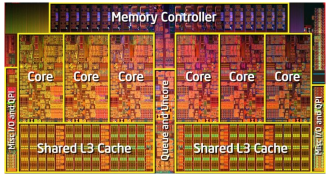
64 cores AMD EPYC Rome, 40 Billion transistors



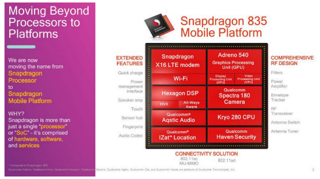
14 cores CPU + 18 cores GPU Apple M3 Max, 92 Billion transistors

Chip Multiprocessors?

Time 2011 2017 2019 2023



6 cores Intel i7, ~1.1 Billion transistors



Moving Beyond Processors to Platforms

We are now moving the name from Snapdragon Processor to Snapdragon Mobile Platform

WHY?
Snapdragon is more than just a single "processor" or "SoC" - it's comprised of hardware, software, and services

Snapdragon 835 Mobile Platform

EXTENDED FEATURES
Quick charge
Power management interface
Speaker amp
Touch
Sensor hub
Fingerprint
Audio Codec

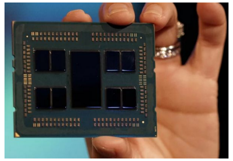
Snapdragon X16 LTE modem
Wi-Fi
Adreno 540 Graphics Processing Unit (GPU)
Display Processing Unit (DPU)
Video Processing Unit (VPU)
Hexagon DSP
HVX
All-Ways Aware
Qualcomm Spectra 180 Camera
Qualcomm® Aqstic Audio
Kryo 280 CPU
Qualcomm® IZat™ Location
Qualcomm Haven Security

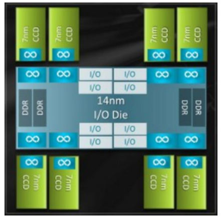
COMPREHENSIVE RF DESIGN
Filters
Power Amplifier
Envelope Tracker
RF Transceiver
Antenna Switch
Antenna Tuner

CONNECTIVITY SOLUTION
802.11ac MU-MIMO
802.11ad

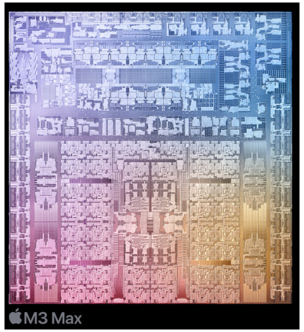
* Compared to Snapdragon 820
Qualcomm Adreno, Qualcomm Kryo, Qualcomm Hexagon, Qualcomm Spectra, Qualcomm Aqstic, Qualcomm IZat, and Qualcomm Haven are products of Qualcomm Technologies, Inc.
2

Qualcomm Snapdragon 835 SoC (Quad core, GPU, DSP...), ~3 Billion transistors





64 cores AMD EPYC Rome, 40 Billion transistors



14 cores CPU + 18 cores GPU Apple M3 Max, 92 Billion transistors

Why this increase in number of compute units (cores) per chip?

Core Count Increase

- For decades we have seen a continual increase in single-core processors **speed**
- Transistors get (significantly) smaller, circuits get (a bit) bigger, **clock frequency increase**
    - Program is slow? Wait for the next generation CPU to get an automatic speed boost ➡️ free lunch for the programmer

Core Count Increase

- For decades we have seen a continual increase in single-core processors **speed**
- Transistors get (significantly) smaller, circuits get (a bit) bigger, **clock frequency increase**
    - Program is slow? Wait for the next generation CPU to get an automatic speed boost ➡️ free lunch for the programmer
- Power consumption and heat dissipation → **clock speed stalled**
- **Architectural approaches to increase single processor speed have been exhausted**

Core Count Increase

- Processors don't get faster, but we can still put more transistors in a single integrated circuit
- Solution: **more cores**! Several closely coupled compute units, i.e. several processors, working together, on a single circuit

Core Count Increase

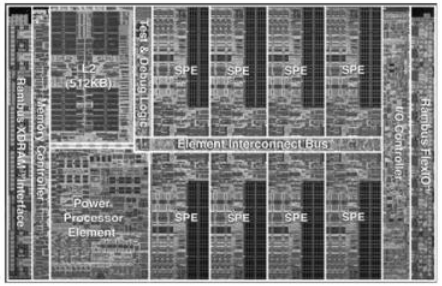
- Processors don't get faster, but we can still put more transistors in a single integrated circuit
- Solution: **more cores**! Several closely coupled compute units, i.e. several processors, working together, on a single circuit
- This has several implications and **2 x 3GHz != 6GHz**

Core Count Increase

- Processors don't get faster, but we can still put more transistors in a single integrated circuit
- Solution: **more cores**! Several closely coupled compute units, i.e. several processors, working together, on a single circuit
- This has several implications and **2 x 3GHz != 6GHz** - **Hardware/architectural issues**: what kind of processor(s) to use? How are processors connected? How is memory organised?

Core Count Increase

- Processors don't get faster, but we can still put more transistors in a single integrated circuit
- Solution: **more cores**! Several closely coupled compute units, i.e. several processors, working together, on a single circuit
- This has several implications and **2 x 3GHz != 6GHz** - **Hardware/architectural issues**: what kind of processor(s) to use? How are processors connected? How is memory organised?
    - **Software issues**: how do we program this thing?



Moore's Law

- Not really a law, more an observation/prediction
- **Transistor count on a chip doubles every 18 months**
- What drove this increase?

Moore's Law

- Not really a law, more an observation/prediction
- **Transistor count on a chip doubles every 18 months**
- What drove this increase?
    - Feature size (basically transistor size)
        - Intel 386 (1985): 1.5 um
        - Apple M1 Max (2021): 5 nm
        - Decrease factor: 300x
    - Die (chip) size
        - 386: 100 mm^2 (275K transistors)
        - M1 Max: 420.2 mm^2 (57B transistors)
        - Increase factor: 4x

Moore's Law

- Not really a law, more an observation/prediction
- **Transistor count on a chip doubles every 18 months**
- What drove this increase?
    - Feature size (basically transistor size)
        - Intel 386 (1985): 1.5 um
        - Apple M1 Max (2021): 5 nm
        - Decrease factor: 300x
    - Die (chip) size
        - 386: 100 mm^2 (275K transistors)
        - M1 Max: 420.2 mm^2 (57B transistors)
        - Increase factor: 4x
- Increase in transistors/chip mostly due to **transistor size reduction**

Smaller Meant Faster, but not Anymore

- Due to electric properties:
    - Smaller transistors have a faster switch delay
    - **They can be clocked at a higher frequency**
    - Circuit gets faster, programmer's free lunch

Smaller Meant Faster, but not Anymore

- Due to electric properties:
  - Smaller transistors have a faster switch delay
  - **They can be clocked at a higher frequency**
  - Circuit gets faster, programmer's free lunch
- **But we hit limits**
  - Power density increases, heating becomes a problem

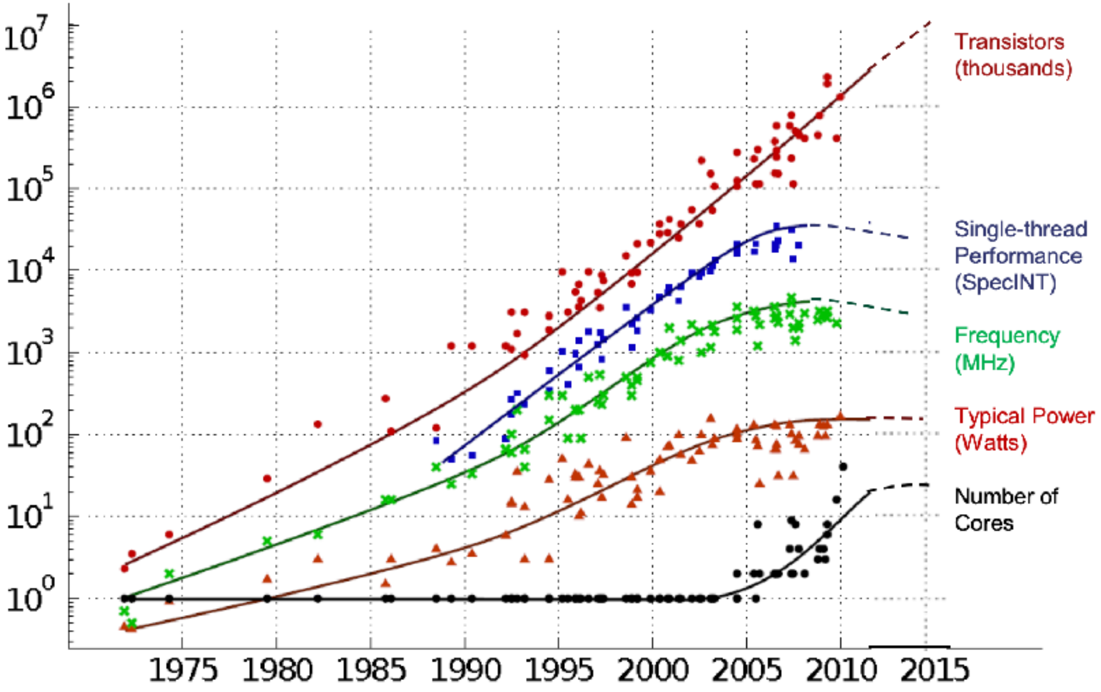
End of Dennard Scaling

- **Dennard Scaling** (1974)
    - Transistors get smaller, so they consume less power as we pack more on the same chip
    - Power density stay constant → no power consumption/heating problem 👍

End of Dennard Scaling

- **Dennard Scaling** (1974)
  - Transistors get smaller, so they consume less power as we pack more on the same chip
  - Power density stay constant → no power consumption/heating problem 👍
  - **Broke down in the mid-2000s** 🥲
    - Mostly due to the high current leakage with small transistor sizes

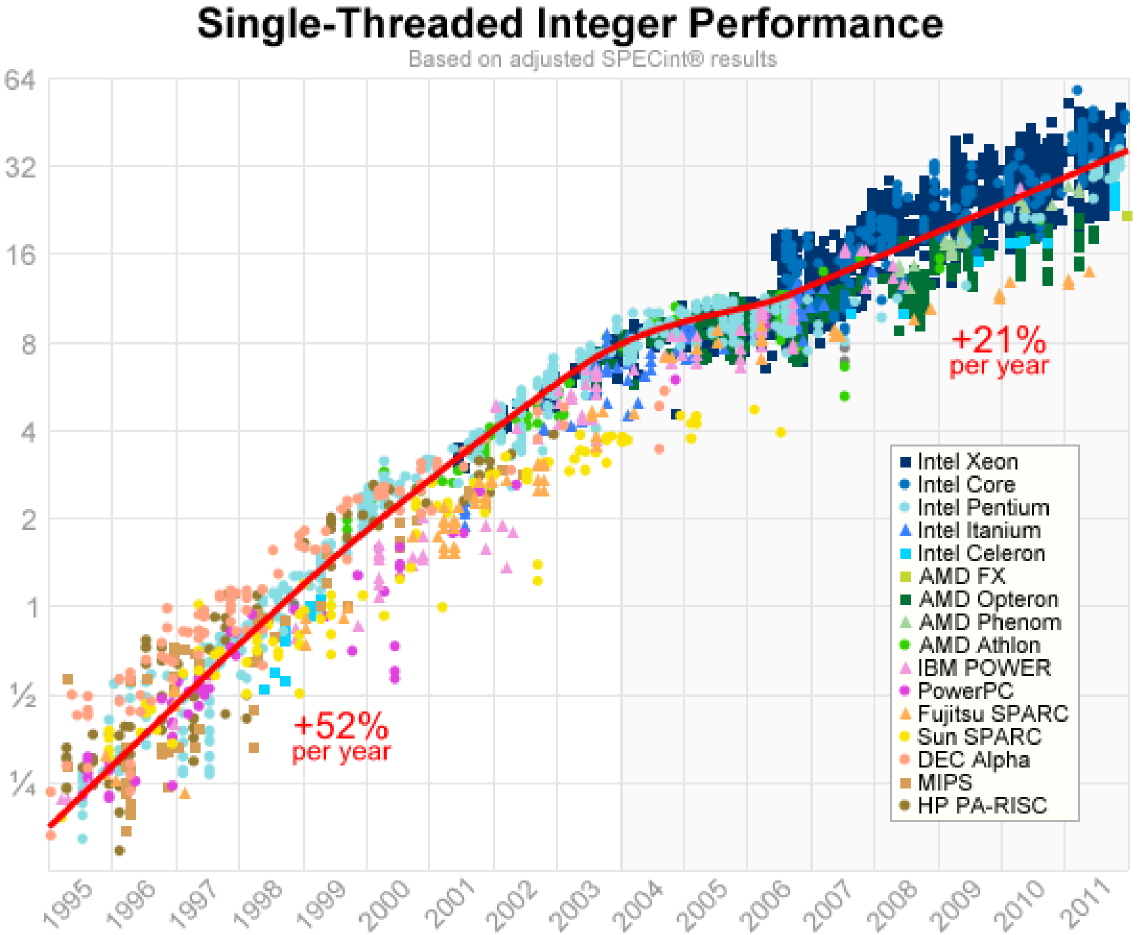
End of Dennard Scaling



Original data collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond and C. Batten
Dotted line extrapolations by C. Moore

End of Dennard Scaling: Frequency can't keep increasing because power can't be removed from the chips, they'd melt!

Single Core Performance



How to Go Faster?

- How to use these extra transistors to get faster?

How to Go Faster?

- How to use these extra transistors to get faster?
    - Can we build ***faster single-core processors?***

How to Go Faster?

- How to use these extra transistors to get faster?
  - Can we build ***faster single-core processors?***
    - **More parallel pipelines** to exploit Instruction Level Parallelism?
      - ILP has diminished returns beyond ~4 pipelines
    - **Bigger caches**:
      - Payback for bigger caches also diminishes rapidly

The "Solution": Multiple Cores

- Put multiple CPUs (cores) on a single integrated circuit (chip)
    - **"Multicore chip" or "Chip Multiprocessor"**
- Use these CPUs in parallel to achieve higher performance
- Simpler to design vs. increasingly complex single core CPUs
- Need more computing power? Add more cores ...
    - ... not that simple in practice, 2 * 3GHz != 6GHz

Multicore "Roadmap"

- Year, cores per chip, feature size

- 2006, ~2 cores, 65nm
- 2008, ~4 cores, 45nm
- 2010, ~8 cores, 33nm
- 2012, ~16 cores, 23nm
- 2014, ~32 cores, 16nm
- 2016, ~64 cores, 12nm
- 2018, ~128 cores, 8nm
- 2020, ~256 cores, 6nm
- 2022, ~512 cores, 4nm
- 2024, ~1024 cores, 3nm
- 2026, ~2048 cores, 2nm
- *scale discontinuity?*
- 2028, ~4096 cores
- 2030, ~8192 cores
- 2032, ~16384 cores

Summary

- Single core performance hit a plateau in the mid 2000s
- To address that issue constructors started to pack more cores on a single chip
- Next: **how to efficiently exploit the resulting parallelism**?