

MULTICORE- UND GPGPU- ARCHITEKTUREN

Korbinian Pauli - 17. November 2011

Seminar „Multicore Programmierung“, WS11, Universität Passau

Einleitung

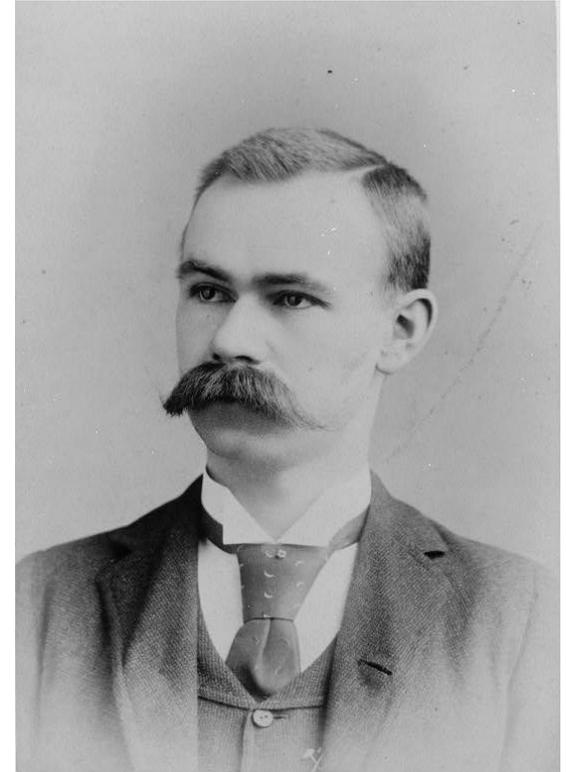
- Klassisches Problem der Informatik: riesige Datenmenge!
- Volkszählung USA: alle 10 Jahre
- Problem 1880: Abbruch der Auswertung nach 7 Jahren um die 1890er-Zählung vorzubereiten. Auswertung würde etwa **12 Jahre** dauern.
- Idee: Maschinelle Zählung
- Tatsächlich: 43 Maschinen, die den Zensus von 1890 in **3 Jahren** schafften

Einleitung

- Herman Hollerith (Student MIT)
→ Hollerith-Maschine
- Sortieren und Auswerten von Lochkarten (mit Daten der Bürger)
- Algorithmus: Bucketsort

- **Parallel arbeitende Maschinen!**

- Tabulating Machine Company
→ später: IBM



Herman Hollerith
(1860 – 1929)

Übersicht für heute:

- Begriffserklärungen / Parallelität
 - CPUs
 - GPGPUs
 - Exascale
 - Prognose

Begriffserklärungen

- Thread
 - SW: *Teil* eines Prozesses
 - HW: *Ausführungsstrang* bzw. *Reihenfolge* in der Abarbeitung eines Programms.
- Kern / Core
 - Recheneinheit eines Prozessors, der *eigenständig* einen Thread ausführen kann.
 - → Zwei Threads auf verschiedenen Kernen *beeinflussen sich nicht*
- Multicore-Prozessor
 - Prozessor mit mehr als einem Kern („Mehrkernprozessor“)
 - → Kann *gleichzeitig* mehrere Threads *parallel* ausführen

Parallelität / Flynn'sche Klassifizierung

	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	MIMD

- SIMD
 - *Zentraler* Programmspeicher, geteilter oder exklusiver Datenspeicher
 - → gleiche Instruktion synchron je Schritt
 - „Arrayprozessor“, „Vektorprozessor“, z.B. GPU
- MIMD
 - Asynchrone und unabhängige Datenverarbeitung
 - → Synchronisation auf Softwareebene da kein einheitlicher Kontrollfluss
- SPMD (Subkategorie von MIMD)
 - Programm wird mehrmals unabhängig und asynchron voneinander ausgeführt
 - Eingabedaten können unterschiedlich sein
 - Instruktionsablauf kann auf jeder Ausführungseinheit *unterschiedlich* sein
 - skalierbar

Simultaneous Multi-Threading

- „SMT“, „Hardwareseitiges Multi-Threading“
- *Mehrere* Threads (2-4) auf *einem* Kern durch virtuelle Kerne
- Betriebssystem erkennt mehr Kerne als vorhanden
- Nur bestimmte Einheiten mehrfach vorhanden, z.B. Register, Pipelines
- Beispiel: Intel Pentium 4: „Hyperthreading“



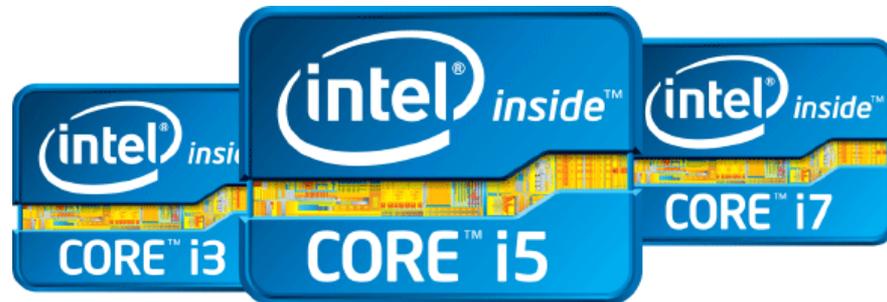
Simultaneous Multi-Threading

- Vorteile
 - Bessere Auslastung der CPU bzw. Vermeiden von Wartezeit
 - Weniger Transistoren notwendig
 - → Geringere Herstellungskosten
- Nachteile
 - Ressourcen (z.B. ALU, Cache) auf Threads aufgeteilt
 - Performanzgewinn geringer als bei „echtem“ Multicore
- Performanzgewinn
 - Intel: 10-20%
 - AMD: 60-70%

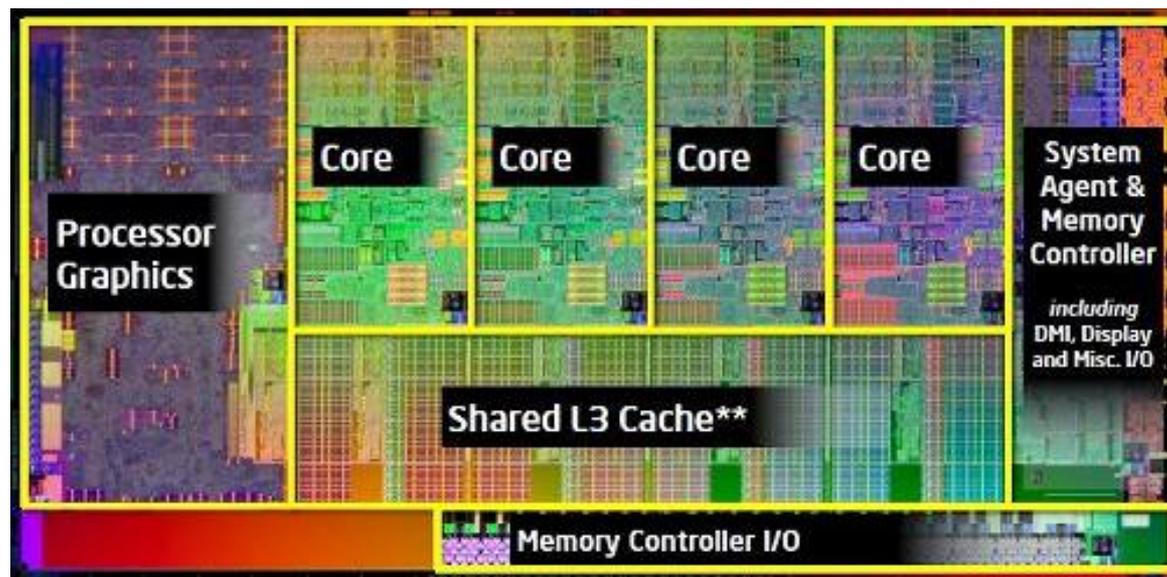
Intel Core-i-Serie / Sandy Bridge

- Architektur: Sandy Bridge (2011)
- Core-i-Prozessorfamilie i3, i5, i7 („2. Generation“)
- Verwendung: privat / Desktop-PC

- Kerne: 2-4
- SMT je nach Modell
- Taktfrequenz: 3,5 / 3,9 GHz (i7-2700K, Oktober 2011)



Intel Core-i-Serie / Sandy Bridge



- Integrierte GPU (850 MHz – 1,35 GHz)
- Jeder Kern: eigener L1- und L2-Cache (64, 256 KB)
Für alle Kerne + GPU: L3-Cache (8 MB)
- Kommunikation der Komponenten: Ring-Bus (1000 Signalleitungen, insg. 300 GByte/s Transferrate (2,4 TBit/s))

Intel Core-i-Serie / Sandy Bridge

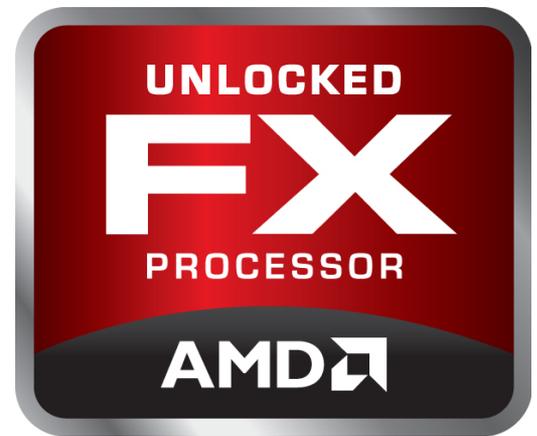
- „Turbo Boost 2.0“: automatisches *Übertakten* von Kernen um bis zu 666 MHz, um „ältere Software“ schneller ausführen zu können
- Wegen Hitzeentwicklung nur für kurzen Zeitraum möglich
- Komplette Abschaltung von Prozessorkernen

Intel® Turbo Boost Technology 2.0



AMD FX / Bulldozer

- Architektur: Bulldozer (Oktober 2011)
- Verwendung: privat / Desktop-PC (Konkurrenz zu Intel i)
- Kerne: 2-4 mit je zwei Integer Cluster
- Taktfrequenz: 3,6 / 4,2 GHz (FX-8150)



AMD FX / Bulldozer

- „Application Power Management“: ähnlich Intel's „Turbo Boost“ für automatisches Übertakten
- Erstmals bei AMD: „Core Multithreading Technologie“ = Kombination von „echtem“ Multithreading und SMT:
 - Nur *Decode*- und *Fetch*befehle müssen unter den Threads aufgeteilt werden
 - Da jeder Kern *zwei* Integer-Cluster besitzt
 - Leistungssteigerung um 60-70%
- beworben von AMD: „[...]weltweit erste unlocked Eight-Core-CPU [...]“
- Übertakten: Weltrekord mit 8,585 GHz (Andre Yang)

Tilera TILE-Gx

- TILE-Gx-Familie mit unterschiedlich vielen Kernen
- Verwendung: professioneller Bereich / Server
- Kerne: 16, 36, 64, 100
- Taktfrequenz: 1,5 GHz

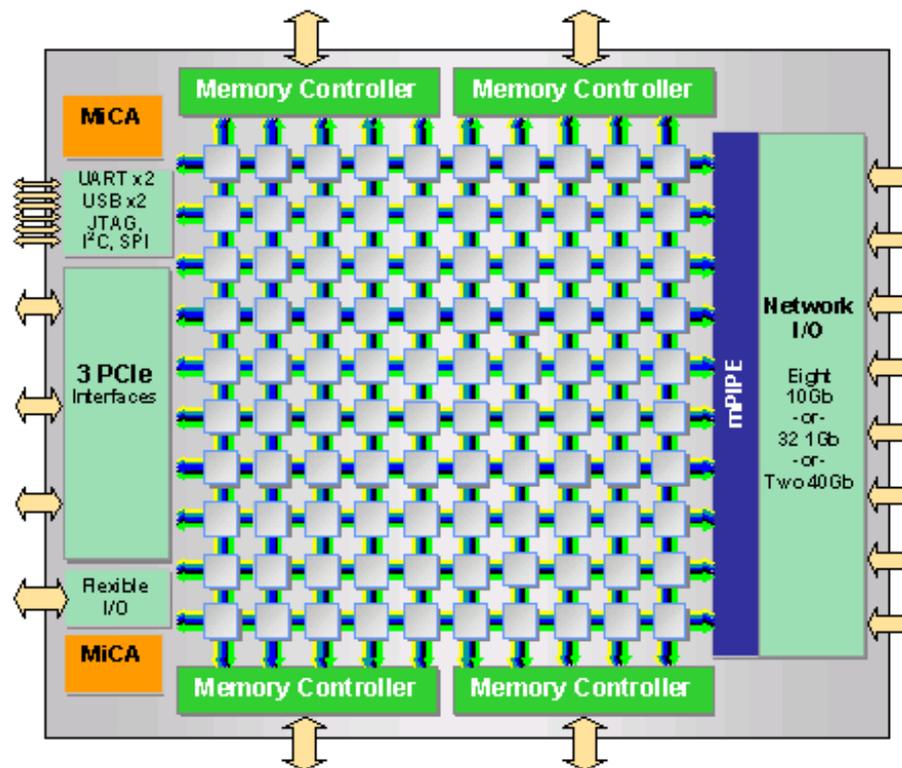


Tilera TILE-Gx

- Startup: 2004, USA
- Ankündigung für 2012: 64Bit-Prozessor mit **100** Kernen
- 64 KB L1- und 256 KB L2-Cache für jeden Kern
- Insgesamt 32 MB an L1- und L2-Cache
- Jeder Kern („Tile“) kann unabhängig von den anderen ein Betriebssystem ausführen (oder zusammengefasst)
- → Bis zu 100 Betriebssysteme gleichzeitig!
- **Problem:** Kerne schnell mit Daten / Befehlen zu versorgen

Tilera TILE-Gx

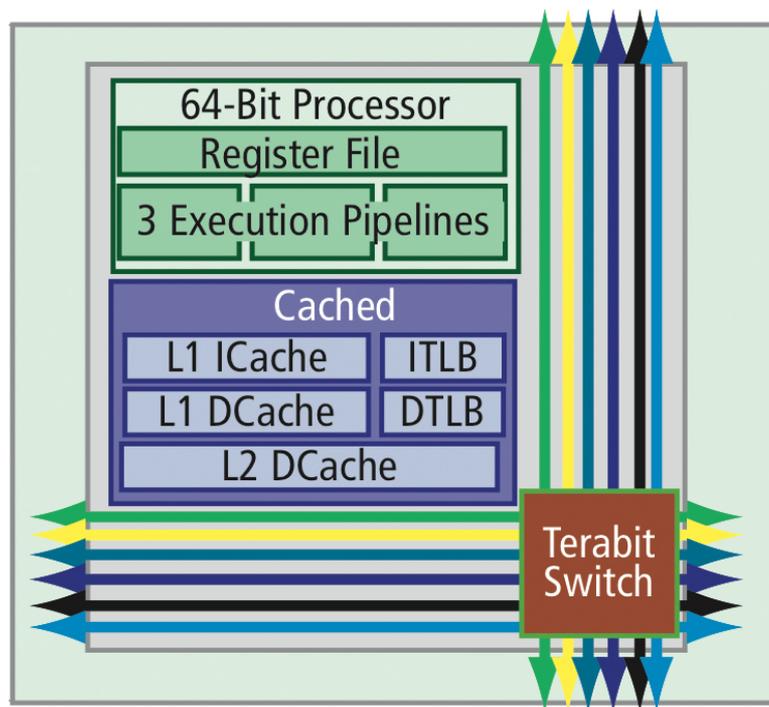
Lösung: „iMesh“ + „Terabit Switch“ + quadratische Anordnung



Quadratische Anordnung der Kerne:

- Hop(max) = 19
- Hop(avg) = 10

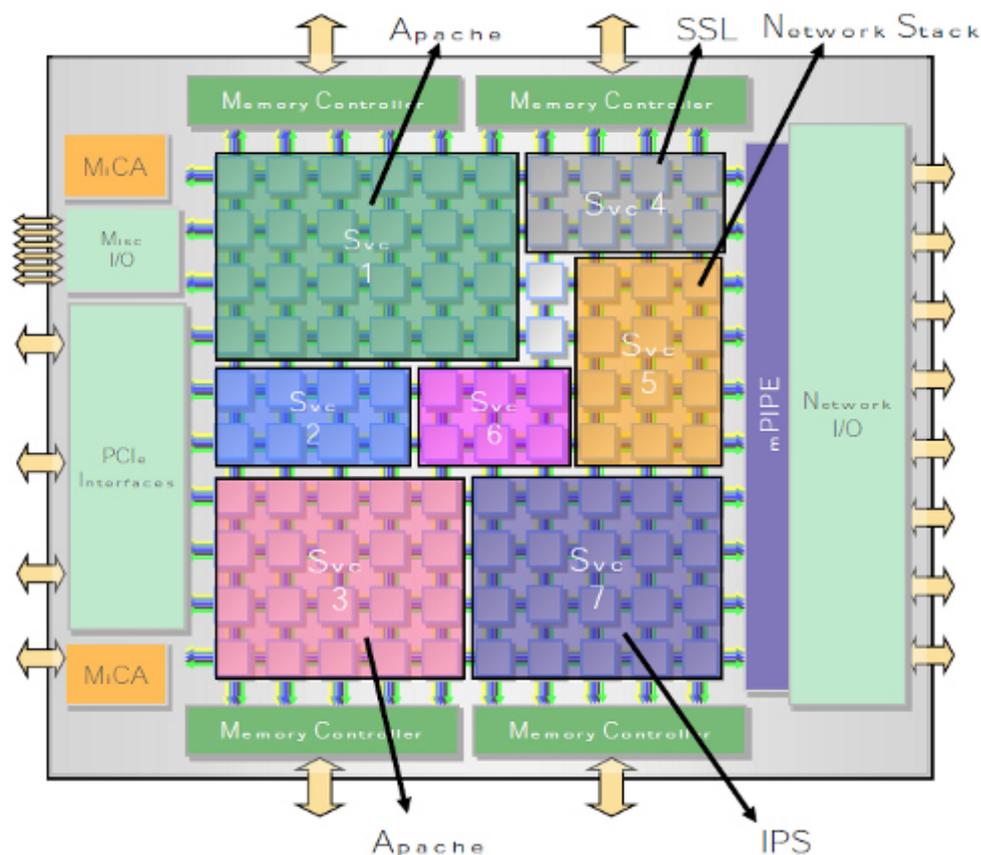
Tilera TILE-Gx



- Terabit Switch:
 - Bussystem aus 5 Kanälen
 - Verschiedene Zwecke je Kanal:
 - Speicher-/Cacheoperationen
 - Copy zwischen Kernen
 - Datentransfer von RAM zu Kern
 - I/O-Netzwerk
 - Benutzerprozesse: Daten auf Kernen verschieben
 - Vorteil: Operationen blockieren sich nicht gegenseitig
 - Durchsatz: 200 Terabit/s
 - vgl. Sandy Bridge: 2,4 Terabit/s

Tilera TILE-Gx

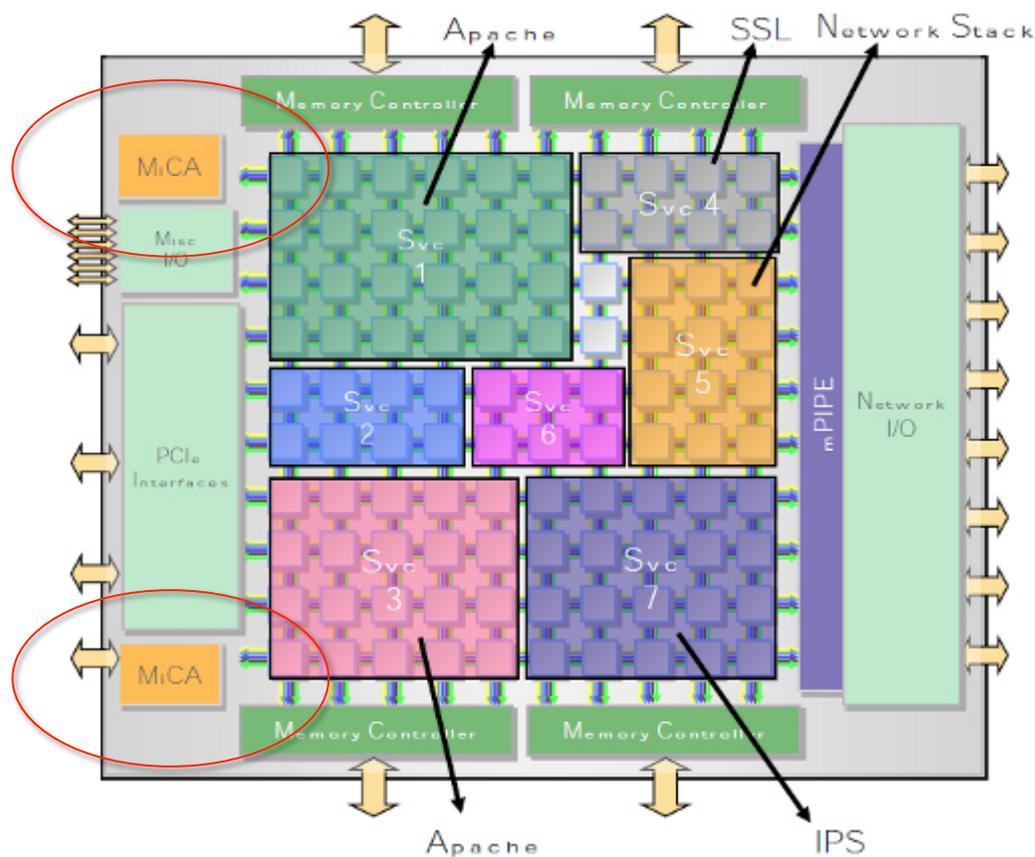
- Partitionierung:



- Aufteilung der Kernzuordnungen nach Anwendungsbereich
- Lokal nebeneinanderliegende Kerne für eine Applikation
- Verringerung der Busauslastung
- Bestimmte Schnittstellen exklusiv für eine Anwendung

Tilera TILE-Gx

- MiCA:



- „Multistream iMesh Crypto Accelerator“

- Für bestimmte kryptografische Anwendungen

- z.B. RSA-Verschlüsselung (1024b Key) mit **40 Gigabit/s**

Tilera TILE-Gx

- Manko: kein x86-Befehlssatz
- Software muss angepasst werden!

• Stromverbrauch?

- Bei „typischen“ Anwendungen: **55 Watt**
- vgl. Intel Xeon, 6 Cores @ 3,46 GHz: 130 Watt

- Neben 100-Kern Version: 16, 36, 64 – Kerne
- Bereits erhältlich: 16, 36

GPGPU

- **General Purpose Computation on Graphics Processing Unit**
 - = Grafikkarte für allgemeine Berechnungen nutzen
 - = Schnittstelle für Programmierer
- Oft herkömmliche Grafikkarte
- Bekannte Schnittstellen: CUDA, OpenCL, DirectX

- Einsatz: wissenschaftlicher Bereich
- Anwendungsbereich: stark parallelisierbare Probleme ab 1 Million *gleichartiger* Berechnungen

GPU/GPGPU

	NVidia Fermi / Tesla 20	AMD FireStream	AMD Radeon HD 6000
Typ	GPGPU	GPGPU	GPU
Modelle	M2090*, M2070 ...	9370*, 9350, ...	6990*, 6970, ...
Einsatzgebiet	High Performance Computing (HPC)		Desktop / Games
Kerne	512	1600	2x 1536 (=3072)
Taktfrequenz	1,3 GHz	825 MHz	830 MHz
Speicher	6 GB (177 GB/s)	4 GB (147 GB/s)	2x 2GB (320 GB/s)
Rechenleistung	665 GFlops / 1,3 TFlops	528 GFlops / 2,6 TFlops	1,27 TFlops / 5,1 TFlops
Stromverbrauch	225 Watt		350 – 450 Watt

- Daten beziehen sich auf das jeweilig leistungsstärkste Modell (*)

GPU/GPGPU Besonderheiten

- NVidia Tesla 20:
 - Error Correction on Code (ECC)
 - C, C++, OpenCL, DirectCompute, Fortran
- AMD FireStream
 - *Kein* ECC → Fehlerhafte Berechnungen möglich
 - JavaCL, OpenCL, DirectCompute, Fortran
- AMD Radeon HD 6990
 - Derzeit leistungsfähigste Grafikkarte für private Zwecke
 - „Far Cry 2“: 1920x1080 Pixel bei 135 FPS
 - Leistungsschalter: + 50 MHz / 450 Watt / 5,4 Teraflops
 - DirectX 11, OpenGL

Exascale



- Tianhe-1A (China):
 - ca. 7200 NVidia Tesla M2050 GPUs
 - ca. 14.000 Intel Xeon Serverprozessoren (6 Cores)

Exascale

- Linpack Benchmark:
 - **2,5** Petaflops Rechenleistung (Platz 2 Top 500)
- Peta = 10^{15} = 1.000.000.000.000.000
- Zum Vergleich:
 - AMD FireStream: 0,0026 Petaflops
 - AMD Radeon HD 6990: 0,005 Petaflops
- **Exascale: 1000 Petaflops = 10^{18} Flops**

Exascale

- Nutzen von Exascale?
 - Neue Erkenntnisse in den Wissenschaften!
 - Hochkomplexe Simulationen von weltweiten Klimaveränderungen
 - Vorgänge in Zellen (Tiere/Menschen/Pflanzen)
- **Aber:**
 - Enormer Stromverbrauch
 - Tianhe-1A mit bisherigen Komponenten auf 1 Exaflop aufgerüstet:
 - Hochrechnung v. Intel: Leistungsaufnahme von **1,6 Gigawatt**
 - Entspricht Stromverbrauch von **zwei Millionen** Privathaushalten
 - (Deutschland \approx 3,7 Mio. Privathaushalte)

Exascale

- Spezielle Technologien notwendig, z.B.
 - Intel: „Knights Ferry“ (früher: Grafikkartenprojekt „Larrabee“)
 - „Many Integrated Core“-Architektur (MIC)
 - Ziel: Exascale-Supercomputer bis 2018 mit **20 Megawatt**
- 32 1,2 GHz Rechenkerne, später bis zu 50 Kerne (22 nm)
- Kern: Pentium basierte CPU, keine GPU
 - allgemeinere Berechnungen möglich
- Knights Ferry – Karte als Beschleuniger für Serverprozessoren

Exascale



- Benchmark in Kombination mit Intel Xeon – Prozessor:
 - 1 Karte: 772 Gigaflops
 - 8 Karten: 7,4 Teraflops

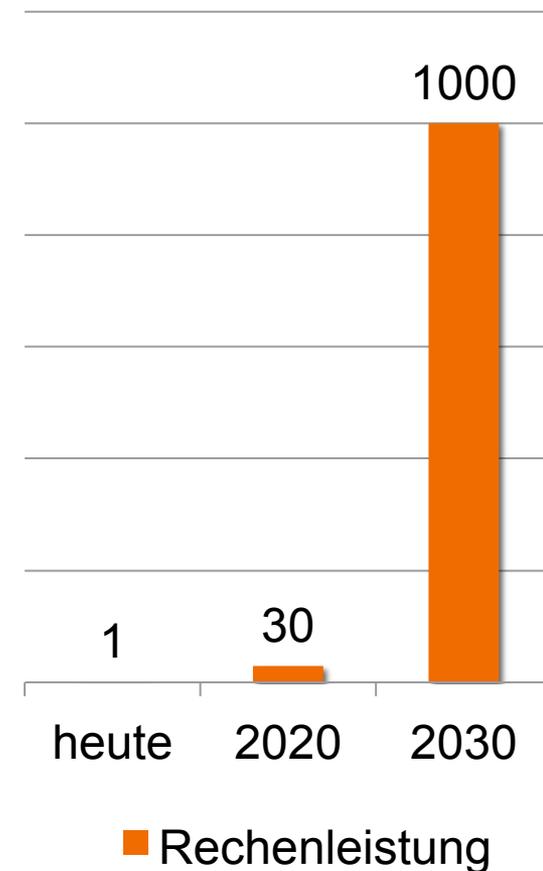
Prognose

- Rasante Entwicklung bei Prozessoren
→ Vorhersage schwierig!
- Trend: Mehr- bis Vielkernprozessor (siehe Tileria!)
- Keine signifikante Erhöhung der Taktfrequenz

Prognose

- Hochrechnung von Intel:
 - 2020: 30-fache Leistung
 - 2030: 1000-fache Leistung
- Annahme: immer kleinere Transistoren
 - Minimale Größe!
 - Transistor besteht nur noch aus wenigen Atomen
 - → gegenseitige Beeinflussung!
- Nicht nur mehr Transistoren auf einem Die sondern alternative Technologien (SMT)

Rechenleistung bis 2030



Vielen Dank!