Wie werden Computer schneller?

M. Anton Ertl
anton@mips.complang.tuwien.ac.at
http://www.complang.tuwien.ac.at/anton/
Institut für Computersprachen
Technische Universität Wien

Mehr Bauelemente

- Gordon Moore 1975 (1965): Die Anzahl der Bauelemente pro Chip verdoppelt sich alle 18 (12) Monate.

Gründe

- Größere Chips
- Kleinere Strukturen

Auswirkung auf die Geschwindigkeit

- Die Schaltzeiten sinken mit dem Quadrat der Strukturgröße.
- Die Übermittlungszeiten bleiben gleich (wenn man sonst nichts ändert).

Prozessoren

Jahr	Strukturgröße	Taktfrequenz	Codename	offizieller Name
	0.5μm	150MHz	P6	
1995	0.35μm	300MHz	Klamath	Pentium II
1997	0.25μm	600MHz	Katmai	Pentium III
1999	0.18μm	1000MHz	Coppermine	Pentium III
2001	0.13μm			

Hauptspeicher

Jahr	Größe	Durchsatz	Zugriffszeit
1982	64Kb	2MB/s	≈500ns
2000	256Mb	1066MB/s	100ns–200ns

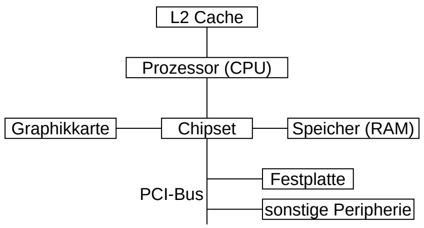
Festplatten

Jahr	Größe	Durchsatz	Zugriffszeit
1993	340MB	20MB/s	
1996	8GB	7MB/s	13ms
2000	73GB	38MB/s	5ms

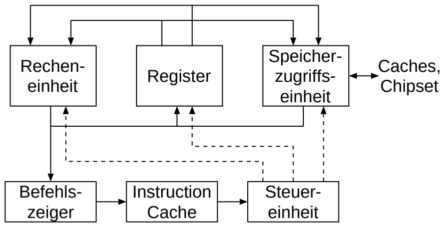
Was macht man mit mehr Bauelementen?

- Mehr Funktionen
- Mehr Integration, weniger Chips, geringere Kosten
- Leistungssteigernde Techniken

Aufbau eines aktuellen PCs



Aufbau eines Prozessors



Ein Maschinenprogramm

```
131, 198, 4, 133, 219, 117, 20, 3, 109, 0, 139, 30

add    $0x4,%esi        131, 198, 4
test   %ebx,%ebx        133, 219
jne    0x804a120        117, 20
add    0x0(%ebp),%ebp   3, 109, 0
mov    (%esi),%ebx      139, 30
```

Architektur

- Auf Maschinenprogrammebene sichtbar
- Nur nutzbar durch Programmänderung
- Beispiel: SIMD-Erweiterungen (MMX, 3DNow, SSE, Altivec)
- Beispiel: IA-64

Mikroarchitektur

- Nur durch Änderung der Geschwindigkeit sichtbar
- Programmänderungen helfen oft
- Beispiel: Caches, Pipelines, superskalar.

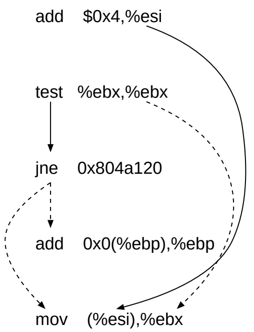
Lokalität der Zugriffe

- Zeitlich: die gleiche Stelle mehrmals
- Räumlich: nah beieinanderliegende Stellen

Speicherhierarchie

	Größe	Zugriffszeit	Durchsatz
Register	96B–512B	0–1T	50GB/s
L1 cache	8KB–64KB	2–3T	10GB/s
L2 cache	256KB–8MB	10–20T	16GB/s
Hauptspeicher	64MB–1.5GB	100–200ns	1GB/s
Festplatte	15GB–80GB	5ms–16ms	38MB/s

Parallelismus auf Maschinenbefehlsebene



- alle Abhängigkeiten: Parallelismus≈2
- nur Datenflußabhängigkeiten: Parallelismus>100

Befehlsabarbeitung

- Befehl holen
- Befehl dekodieren (Steuersignale erzeugen)
- Register auslesen
- Operation ausführen
- Resultat in Register schreiben

Fließbandverarbeitung (Pipelining)

- Pro Zyklus ein Befehl
- Bypasses
- Branch prediction
-

Prozessor	Stufen
486, Pentium	5
K6	7
P6, Athlon	≈10
Pentium IV	≈20

Superskalar

- Mehrere Befehle pro Zyklus

Umordnung von Befehlen

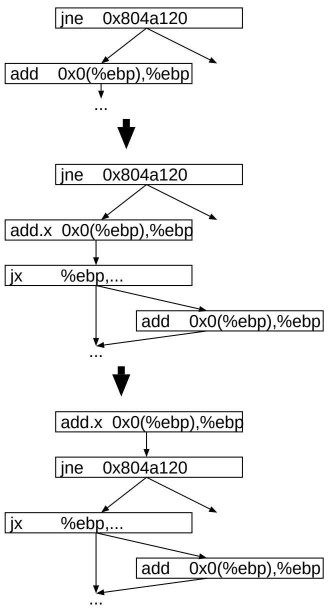
- Abschluß der Befehle in ursprünglicher Reihenfolge
- Sprungvorhersage
- Vorauseilende Ausführung
- Umbenennung von Registern

Prozessor	Befehle/ Zyklus	Umordnung
486	1	nein
Pentium	2	nein
K6	2	ja
P6, Athlon	3	ja
Pentium-IV	3	ja

IA-64 (64-bit Intel Architektur)

- Umordnung in Software statt in Hardware
- Architektur unterstützt Umordnung in Software
- Software kann besser umordnen
- Software hat andere Grenzen
- Weniger komplexe Steuereinheit, schneller?
- Beispiel: Umbenennung von Registern:
  128 Register, software-gesteuerte Umbenennung

Steuerungsabhängigkeiten



Warum werden Computer langsamer?

- Neue Software braucht mehr Leistung weil
  - sie mehr kann, oder
  - sie so entwickelt wurde

Ziele bei kommerzieller Software

- kommerzieller Erfolg
- wenig Entwicklungskosten
- schnelle Entwicklung
- "besser" als Konkurrenz/Vorgänger
- nicht zu viele Fehler
- nicht zu langsam

Produkt	Jahr	Zeilen	Jahr	Zeilen
MS Basic	1975	4 000	1996	500 000
MS Word	1982	27 000	1996	2 000 000

Zusammenfassung

- Miniaturisierung
- Lokalität, Parallelismus
- Mikroarchitektur:
  Caches, Pipelining, superskalar, Umordnung
- Architektur:
  mehr Register, SIMD, Umordnung in Software
- Software passt sich der Hardware an