

Vom Von-Neumann-Prinzip zu heutigen Rechnerarchitekturen

Marcel Waldvogel

Geschichte

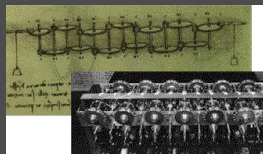


- Knochenritzen (~20000 v. Chr.)
- Primzahlen (8500 v. Chr.)
- Ziffernsystem (Babylon, ~1900 v. Chr.)
- Abakus (Babylon, ~1000 v. Chr.)
- Null (Babylon, ~300 v. Chr.)
- Negative Zahlen (Indien, ~600 n. Chr.)

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 2

Geschichte 2

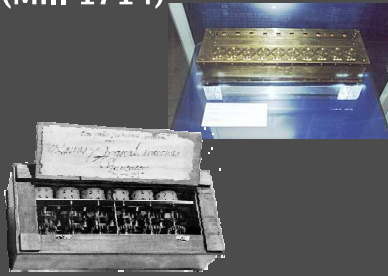
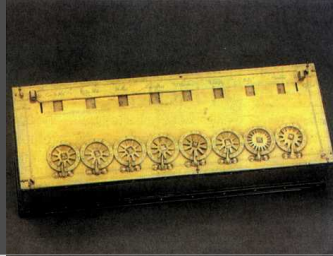
- Arithmetische Maschine (Da Vinci, 1500)
- Rechenschieber (Oughtred/Napier 1621)
- Zahnradcomputer (Schickard 1623)



Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 3

Geschichte 2(a)

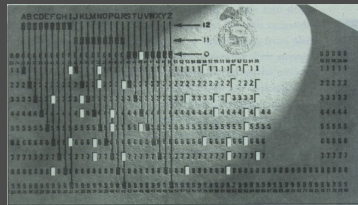
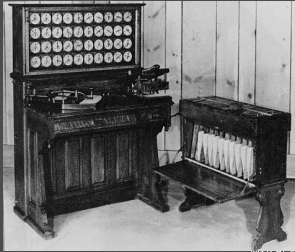
- Addiermaschine (Pascal 1640)
- Staffelwalzen, Dualzahlen, Quadratwurzeln (Leibniz 1672)
- Schreibmaschine (Mill 1714)



Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 4

Geschichte 3

- Lochkarte (Jacquard 1800)
- Programmierung (Idee; Babbage 1822-33)
- Programm (Lovelace, ~1830)
- Boole'sche Algebra (Boole, ~1850)
- Lochkartenspeicher (Hollerith 1884)



Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 5

Geschichte 4

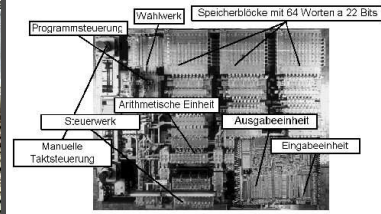
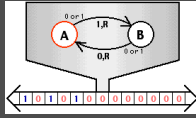
- Röhren (Edison 1883, Fleming 1900, de Forest 1906)
- Roboter (Buch; Capek 1921)
- Transistor (Lilienfield 1926)
- Analogrechner (Bush 1927)
- Elektrische Logik (Burack 1936)



Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 6

Geschichte 4(a)

- Turingmaschine (Turing 1937)
- Binärrechner (Z1; Zuse 1938)
- Programmiersprachen (Plankalkül 1945)



Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 7

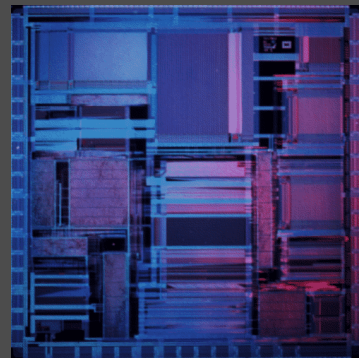
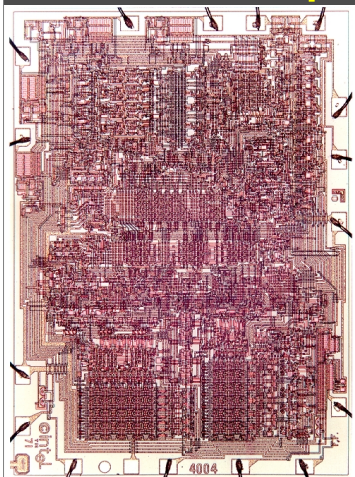
Geschichte 5

- Rechnerarchitektur (von Neumann 1945)
- EDVAC (Mauchly, Eckert 1946)
- Assembler (1949)
- Integrierte Schaltkreise (Dummer 1952, Kilby 1958)
- Paketnetze (Davies 1966)
- Speicherchips (1970)
- Mikroprozessor (Intel 1971)
- E-Mail (Tomlinson 1971)



Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 8

Mikroprozessoren



IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 9

Von Neumann-Prinzip

- 4 Komponenten: Hauptspeicher, Leitwerk, Rechenwerk, Ein-/Ausgabe
- Struktur des Rechners unabhängig vom Programm
- Programme und Datenstrukturen im selben Speicher
 - **Vergleiche Harvard-Architektur, DSPs**
- Hauptspeicher: Fortlaufend nummerierte Zellen gleicher Grösse

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 10

Von Neumann-Prinzip (Forts.)

- Programm: Folge von meist sequentiellen Befehlen
 - **Sequentialität implizite Fortschaltungsregel**
- Abweichungen davon durch bedingte oder unbedingte Sprungbefehle
- Maschine benutzt Binärcodes; stellt Zahlen dual dar

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 11

Unterschiede in modernen Architekturen

- Differenzierte Speicherhierarchie
 - **Register, (mehrere Stufen von) Caches, Hauptspeicher, Sekundärspeicher**
- Parallelität und Pipelining
 - **Parallelität auf Instruktionsebene (SIMD)**
 - **Multiprozessorsysteme**
- Getrennte Speicher für Daten und Programme
 - **Caches in Harvardarchitektur**

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 12

Typen von Rechnern

- Eingebettete Rechner
 - Palmtops, Liftsteuerungen, . . .
- Personal Computer (PC)
 - Einzelner Benutzer (früher durch geringe Leistung bedingte einfache Betriebssysteme)
- Arbeitsplatzrechner (Workstation)
 - Höhere Leistung, mehrere Benutzer
- Grossrechner (Mainframe)
 - Zuverlässigkeit, viele Benutzer, mehrere Prozessoren, Spezialhardware

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 13

Vergleichsmerkmale

- "Lies, dann lies, and benchmarks"
 - Leistung hängt immer von Anwendung ab
 - Dauerleistung vs. Reaktionsfähigkeit
- Rechenleistung
 - MIPS, FLOPS; standardisierte Benchmarks
 - MHz * CPI
- Speicherausbau
 - Grösse, Zugriffsgeschwindigkeit (Latenzzeit, Durchsatz)
- Systembus
 - Fähigkeiten, Durchsatz, Latenzzeit
- Fehlerrate

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 14

Rechnergenerationen

Generation	Technologie	Befehle/s	Programmierung
1. (bis 1960)	Röhren	1000	Maschinensprache
2. (bis 1970)	Transistoren Ferritkernspulen	10000	Erste höhere Sprachen (Fortran, COBOL, ...)
3. (bis 1980)	Halbleiter SSI, MSI 8-Bit-Prozessoren	500000	Höhere Sprachen Datenbanken Programmiersysteme
4. (heutige)	Halbleiter (V)LSI 16-, 32-, 64-Bit	>1000000	Entwicklungsumgebungen Parallele/verteilte Programmierung
5.	Optische, biologische/chemische und Quantencomputer (Hochparallele Systeme)		

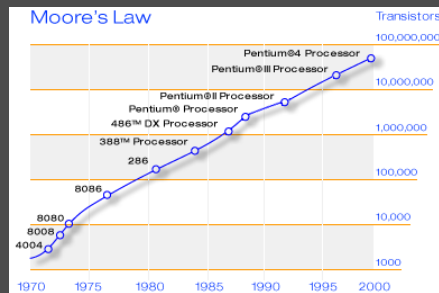
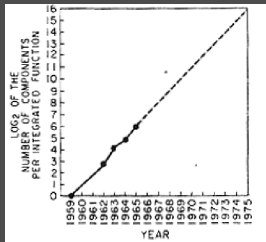
Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 15

Zukunftsperspektiven

■ Moore and Noyce (1965)

- Verdoppelung der Transistoren auf einem Chip alle 12 Monate
- Heute: 18-24 Monate

- Transistoren
- Taktraten



Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 16

Moderne Rechnerarchitekturen

■ Virtuelle Rechnerarchitektur

- Mikroprogramme im ROM
- Umfangreichere, variable Befehlssätze

■ CISC

■ RISC

■ Eierlegende Wollmilchsau

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 17

CISC

■ Complex Instruction Set Computer

- >100, z.T. komplexe Befehle
- Viele Adressierungsarten
- Aufwärtskompatibles Familienkonzept
- Mikroprogramme
- Trennung von Befehlssatz, Implementationstechnik und Realisation

■ Beispiele

- Intel x86, Motorola m68k, DEC VAX, IBM-Grossrechner

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 18

RISC

- Ziel: schneller, höher, weiter
 - Befehle unterschiedlich häufig
 - Komplexe Befehle weniger häufig
 - "Optimizing the common case"
- Reduced Instruction Set Computer
 - Elementare, einfache Operationen
 - Addition, Subtraktion, Logik, Speicher, Kontrolle
 - Orthogonal, einfach für Compiler
 - Keine Adressrechnung im Speicherbefehl
 - Operanden nur in (vielen) Registern
 - Pipelining
 - Fest verdrahtetes Leitwerk
 - Wenige einheitliche Befehlsformate

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 19

RISC (2)

- Beispiele:
 - MIPS (Silicon Graphics, Embedded Devices)
 - David A. Patterson und John L. Hennessy: Computer Architecture: A Quantitative Approach
 - HP PA-RISC
 - Sun SPARC
 - IBM Power-PC
 - DEC Alpha AXP
- Workstations, Hochleistungsrechner

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 20

CISC vs. RISC

- Religionskrieg ca. 1990
- CISC
 - Bequem für Assemblerprogrammierung
 - Viele Befehle
 - Nachteilig für Geschwindigkeit
- RISC
 - Nur für Compiler
 - Wenige Befehle

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 21

RISC vs. CISC heute

■ RISC wird CISC

- Kompatibilität zu Performance-Befehlen vorheriger Generationen
- Moore's Law: Leistungssteigerungen machen Erweiterungen attraktiv (Marketing, Preis)
- Direkte Implementation als festverdrahtete Logik möglich (Platz dank Integrationsdichte)

■ Neue Befehle

- SIMD (MMX, VIS, ...)
- Grafikbefehle
- Fließkomma
- Multiplikation, Division
- Speicherverwaltung (Prefetch, Cache Control)

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 22

Prozessorbefehle

■ $x := x + k;$

■ Von-Neumann pur

- LOAD x
- ADD k
- STORE x

■ CISC

- ADD x, k

■ RISC

- LOAD R1, x
- LOAD R2, k
- ADD R1, R2, R3
- STORE R3, x

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 23

Heutige Architekturen und Tricks

■ Virtuelle Maschinen, virtuelle Hardware

- Motorola m68k
- IBM S/390 (zSeries) VM
- Emulation
- Transmeta Code-Morphing

■ Grössere Registersätze

■ Beispiel: Intel x86-Architektur (IA32)

- 8 spezialisierte Register
- Nicht-orthogonale Befehlsstruktur

Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 24

Pipelining

■ Von-Neumann

- Fetch-Decode-Execute-Zyklus
(Interpretation-Ausführung)

■ Verbesserungsmöglichkeiten?

■ Nachteile/Gefahren?

- Sprungbefehle
- Data Hazards

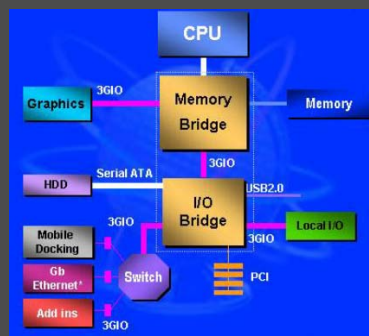
Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 25

Einbindung der Prozessoren

■ Bussysteme

- (E)ISA, Micro Channel, PCI, AGP
- PCMCIA, Cardbus
- SCSI, ATA(PI)
- USB, Firewire (IEEE 1394)

■ Seriell, parallel



Marcel Waldvogel, IBM Zurich Research Laboratory, Universität Konstanz, 15.10.2001, 26