

# Aufbau eines von-Neumann-Rechners

Marcel Waldvogel

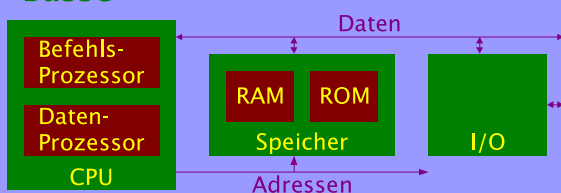
## Übersicht

- Bausteine
- Design
- Speicher
- Busse
  - Adressen, Daten, Steuersignale
- Ein-/Ausgabe
- Interrupts

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

## Grundbausteine

- Zentraleinheit (CPU)
  - Daten-, Befehlsprozessor
- Arbeitsspeicher
  - ROM/RAM
- Ein-/Ausgabesysteme (I/O Unit)
- Busse



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

## Design-Elemente

- **Unabhängig von Programmen**
- **Speicher**
  - **bekannter Wortlänge**
  - **mit festen Adressen**
- **Vielseitige Datenwerte**
  - **Binär**
  - **Nicht selbstidentifizierend/-beschreibend**
  - **Identisches Format für Befehle, Daten und Adressen**

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

## Aufbau und Arbeitsweise der CPU

- **Datenprozessor**
  - **Rechenwerk (ALU)**
  - **Akkumulator**
  - **Puffer-Register (MBR)**
- **Befehlsprozessor**
  - **Befehlsregister (IR)**
  - **Speicheradressregister (MAR)**
  - **Befehlszähler (PC)**
  - **Befehlsdecodierer**

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

## Operation

- **Genau ein Befehl mit genau einem Datenwert (SISD)**
- **Speicherzellen kontextabhängig eingesetzt als Befehl, Daten oder Adressen**
- **Zwei Phasen**
  - **Interpretation (Fetch/Decode)**
  - **Ausführung (Execution)**
    - **Meist lineare**
- **von-Neumannscher Flaschenhals**

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

## Speicher

- **ROM**
  - Start, Systemdienste
- **RAM**
  - Applikationen, Daten
- **Speicher (relativ) langsam**
  - Register, Cache, Prefetch, Write-behind
- **Speicher (relativ) klein**
  - Externer Massenspeicher, Datenhaltung, Virtual Memory
- **Grösse von MAR und MBR**

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

## Busse

- **Datenaustausch zwischen Komponenten**
  - Adressen
  - Daten
  - Steuersignale
  - Mehrfachzugriff

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

## Ein-/Ausgabe

- **Sehr langsam**
  - Tastatur/Terminal, Soundkarte, Netzwerk, Disk
- **Unabhängige Operation**
  - E/A-Gerät oder CPU beschäftigt
- **Buffer**
- **Interrupts**
- **Direkter Speicherzugriff (DMA)**

Marcel Waldvogel, IBM Zürich Research Laboratory, Universität Konstanz, 15.10.2001, 9

## Interrupt

1. CPU hat neuen Auftrag für beschäftigtes E/A-Gerät: Auftrag in Warteschlange, Weiterarbeit
2. E/A-Gerät beendet aktuellen Auftrag: Interrupt an CPU
3. CPU unterbricht aktuelle Tätigkeit
4. CPU erfragt Status des letzten Auftrags; kopiert evt. Daten in Arbeitsspeicher
5. CPU konfiguriert Gerät für nächsten Auftrag und startet Gerät
6. CPU fährt mit unterbrochener Tätigkeit fort

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

## Interne Grössen

- 8 Bit = 1 Byte (heute)
- SI-Einheiten
  - Kilo =  $10^3$ ; Mega =  $10^6$ ; Giga =  $10^9$
  - Kibi =  $2^{10}$ ; Mebi =  $2^{20}$ ; Gibi =  $2^{30}$
  - 1024, 1'048'576, 1'073'741'824
- 2 kByte = Schreibmaschinenseite, 1 MByte = dickes Buch
- Zweierpotenzen bis  $2^{16}$ , sowie  $2^{24}$  und  $2^{32}$  wissenswert

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

## Rechnerstart

- Definiertes Initialzustand des Prozessors
- Geschichte des Systemstarts
  - Vorzeit: Programmieren via Schalter
  - Gesamtes Betriebssystem im (EEP)ROM
    - Heim-/Spielcomputer, dedizierte Rechner
  - Bootloader und Grundfunktionen im ROM
    - CP/M: BIOS und BDOS
  - Mehrstufiger Bootloader
    - Bootmanager, Entpacker, zu grosse Disks
  - Minimalsystem als Bootloader
    - OS benutzt Funktionen des Laders nur anfänglich
  - Betriebssystem selbst als Bootloader
    - Linux-BIOS

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