2.2 Rechnerorganisation: Aufbau und Funktionsweise

- Hardware, Software und Firmware
- grober Aufbau eines von-Neumann-Rechners
- Arbeitsspeicher, Speicherzelle, Bit, Byte
- Prozessor
- grobe Arbeitsweise eines Prozessors
- Memory - I/O Bussse

Hardware, Software und Firmware

Zur Begriffsklärung

- Hardware — elektronische Schaltungen, I/O-Geräte, Speicher, ...
- Software — durch Programme virtueller Rechner oberhalb der Maschinensprache beschriebene Algorithmen
- Firmware — Programme, die bei der Fabrikation eines Rechners in Hardware eingebettet wurden (sind in der Regel in Nurlesespeicher (ROM, PLA, ...) abgespeichert)

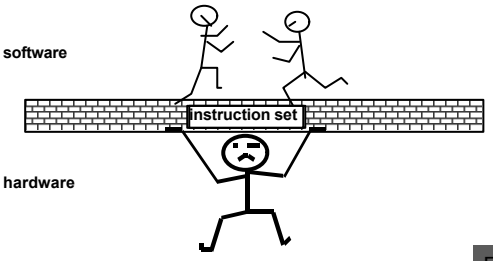
Hardware vs. Software (1)

- Früher Software, heute Hardware:
  - ganzzahlige Multiplikation, Division
  - Gleitkommaarithmetik
  - ...
- Aber auch: RISC-Konzept (Reduced Instruction Set)
  - Teile wieder mit Software ausgeführt.

    Beispiel: ganzzahlige Division in RISC-Prozessoren

Hardware vs. Software (2)

- Festlegung des Befehlssatzes eines Rechners entscheidet, welche Teilaufgaben durch Hardware und welche durch Software gelöst werden können.



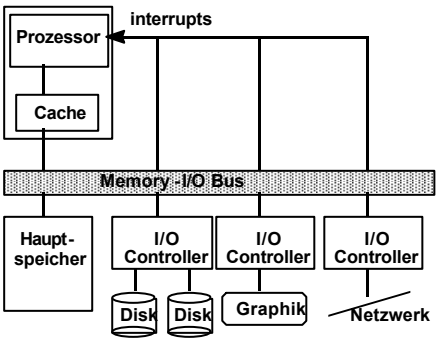
Beispiel
Realisiert man die Division als Hardware oder als Software?

Fragen der Kosten und der Performanz !

Exkurs: Hardware / Software Co-Design

- Anwendung insbesondere bei „eingebetteten Systemen", die nicht frei programmierbar sein müssen
- Welche Komponenten in Hardware realisiert bzw. in Software/Firmware implementiert werden, entscheidet der "Systemarchitekt" oder wird (halb-)automatisch anhand von Laufzeitanalysen der Software entschieden.
- Auslagern rechenzeitintensiver Teile in Hardware!

Prinzipieller Aufbau eines Rechners



Bestandteile:

- Prozessor (CPU)
- Hauptspeicher
- Externe Speicher
- Eingabegeräte (Tastatur, Maus)
- Ausgabegeräte (Bildschirm, Drucker, Plotter)
- Busse

John von Neumann (1903 - 1957)

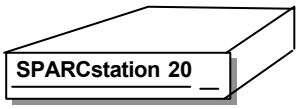
Der Aufbau der modernen Rechner wurde stark von dem Mathematiker John von Neumann beeinflußt.

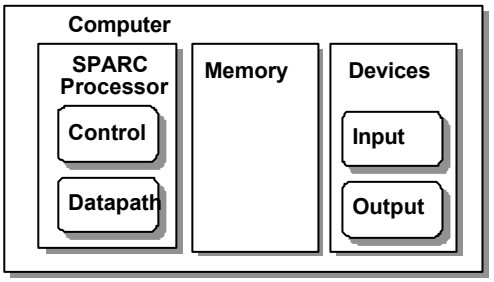
Das von Neumann Prinzip beruht auf folgenden 5 Punkten
- voll-elektrischer Rechner
- binäre Darstellung der Daten
- Benutzung einer arithmetischen Einheit
- Benutzung eines Steuerwerkes
- Interner Daten- und Programmspeicher

1903 geboren in Budapest
Studium in Budapest und Zürich
1933-1957 am Institute for Advanced Studies, Princeton
1945 mitverantwortlich für den Bau der ENIAC

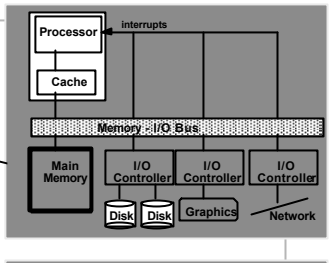
John von Neumann, 1952

SUN SPARCstation 20





Der Arbeitsspeicher



- ... erlaubt es, Daten (Programme, "übliche" Daten) abzuspeichern
- ... besteht aus n Speicherzellen
- Eine **Speicherzelle**
  - speichert eine Information bestehend aus k Bit. Der Wert k wird **Wortbreite des Speichers** genannt (k=8,16,32,64)
  - wird über eine Adresse (≈ Hausnummer) aus dem Bereich [0..n-1] angesprochen
  - ist kleinste adressierbare Einheit des Speichers

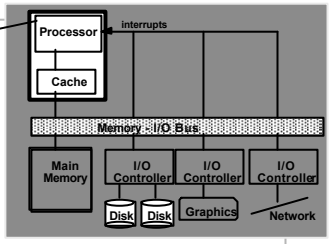
Kerngrössen
- Anzahl der Speicherzellen
- Wortbreite des Speichers
- Zugriffszeit

Ein **Bit** ist die kleinste Informationsmenge, die in einem System mit genau zwei Werte, abgespeichert werden kann.

Ein Bit ist entweder gleich 0 oder 1. Die Belegung eines Bits wird hierbei durch die angelegte Spannung definiert

Ein **Byte** besteht aus 8 Bit.

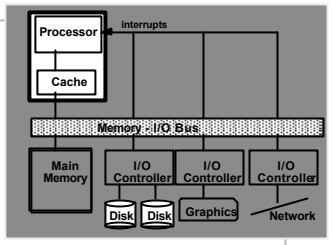
Der Prozessor



besteht in der einfachsten Form aus

- **Steuerwerk (Control Unit)**
  - holt durch Setzen geeigneter Signale den nächsten Maschinenbefehl aus dem Hauptspeicher in ein Register
  - analysiert durch Setzen geeigneter Signale das gelesene Wort
  - führt durch Setzen geeigneter Signale den gelesenen Maschinenbefehl aus
- **Recheneinheit (Arithmetic Logic Unit - ALU, Datenpfad)**
  - führt die Teilschritte (Vergleiche, Addition, ...) gemäß der gesetzten Signale aus
- **Registerbank**
  - Speicherelemente, sogenannte Register, auf die das Rechenwerk und das Steuerwerk ohne Wartezyklus schnell zugreifen kann.

Register des Prozessors



Vom Prozessor unbedingt benötigte Informationen

- aktuell abzuarbeitender Befehl → **Befehlsregister** (instruction register - IR)
- Adresse des nächsten abzuarbeitenden Befehls im Hauptspeicher → **Befehlszähler** (programm counter) - PC
- Adresse im Hauptspeicher, ab der Daten und Informationen gespeichert werden können → **Stackpointer** - SP
- von der ALU zu verarbeitende Daten → **Arbeitsregister** (general purpose register)

Prinzipielle Arbeitsweise eines Prozessors

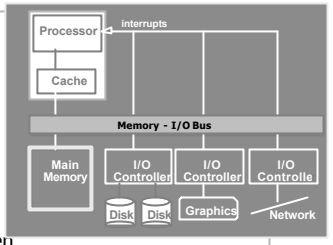
Fetch-Decode-Excecute-Zyklus

Fetch hole den nächsten Maschinensprachebefehl aus dem Arbeitsspeicher und speichere ihn im Befehlsregister IR ab. Die benötigte Adresse steht im Befehlszähler PC

Decode analysiere den Befehl und lade die benötigten Daten

Execute führe den Befehl aus und speichere das Ergebnis ab

Memory - I/O Busse



Was ist ein Bus ?

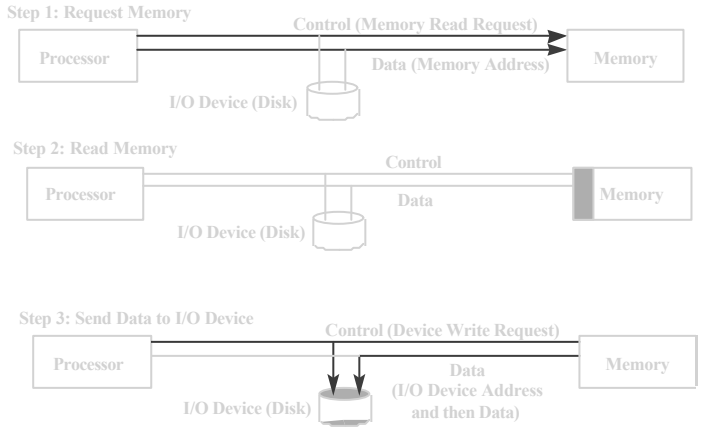
- Elektrische Leitungsbündel in einem digitalen Rechner
- Kommunikationsmedium, auf dem Daten fliessen können
  - zwischen CPU und Hauptspeicher
  - zwischen CPU und I/O-Geräten
  - zwischen Hauptspeicher und I/O-Geräten
- normierte Stecker
- standardisierte Interpretation der einzelnen Leitungen

Am häufigsten benutzte Bussysteme:

- ISA-Bus (Intel 8088, 20 Adressleitungen, synchrone Übertragung)
- EISA-Bus (Intel 80386, 32 Adressleitungen, synchrone Übertragung)
- PCI-Bus (Intel Pentium, 32 Adress/Datenleitungen, synchrone Übertragung)
- SCSI-Bus (Unix-Systeme, 8 Adress/Datenleitungen, asynchrone Übertragung)

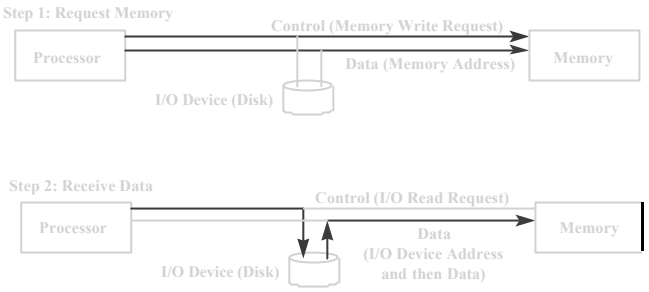
Prinzipieller Ablauf einer Kommunikation

Datentransfer von Hauptspeicher zu einer Platte



Prinzipieller Ablauf einer Kommunikation

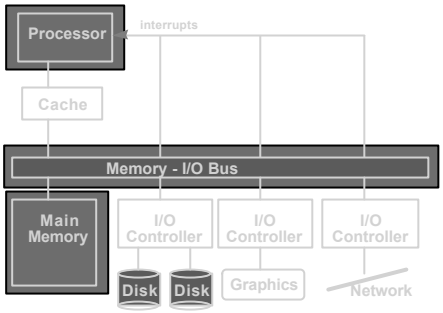
Datentransfer von einer Platte zum Hauptspeicher



Optionen eines Busses

- Separate Adress- und Datenleitungen sind schneller
- Multiplexen von Adress- und Datenleitungen ist billiger
- Breiter ist schneller
- Schmaler ist billiger
- Mehrere Busmaster erlaubt (erfordert Schiedsrichter)
- Nur ein Busmaster (erfordert keinen Schiedsrichter)
- synchrone Übertragung
- asynchrone Übertragung

Zusammenfassung



Wie geht es weiter?

„Optimierung der Verarbeitung"

Wichtige Methoden:

- **Pipelining** (Kap. 3)
- **Speicherhierarchie, Cache** (Kap. 4)
- **Parallelverarbeitung** (Kap. 5)