Technischer Aufbau und allgemeine Funktionsweise eines Computers

Jannek Squar

Proseminar CiS Physik

01.11.2011

Technischer Aufbau und allgemeine Funktionsweise eines Computers

Was ist ein Computer?

„Computer (Lateinisch: „computare“: ausrechnen) sind die elektronische Weiterentwicklung der mechanischen Rechenmaschinen. Daten werden in den Computer eingegeben, um dort weiterverarbeitet zu werden, worauf im Anschluss Ergebnisse ausgegeben werden. Durch die Verbindung von relativ einfachen Befehlen können so umfangreiche und komplexe Problemstellungen bearbeitet und/oder gelöst werden“ (frei nach „Computer Lexikon 2008“)

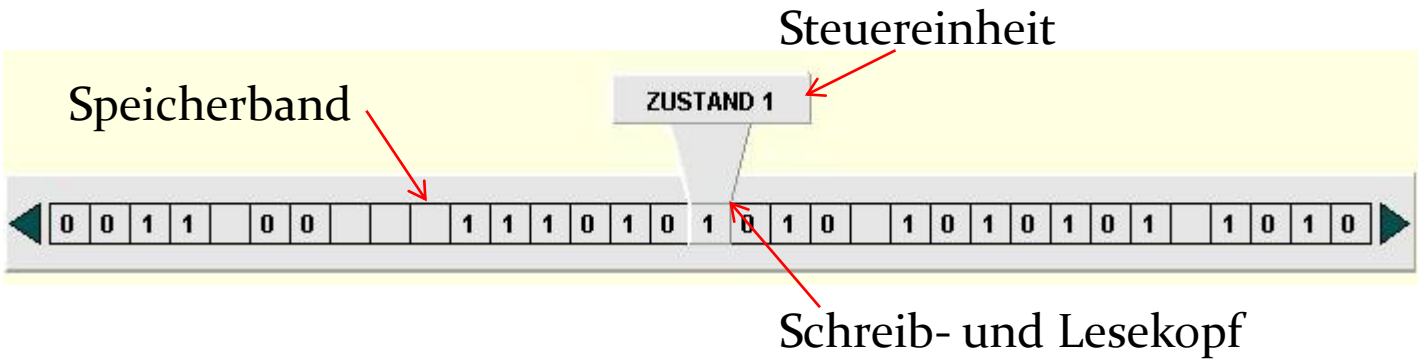
Was ist ein Computer?

- Grundlegende Definition über die Turingmaschine (1936)

-„*Die Klasse der Turing-berechenbaren Funktionen ist genau die Klasse der* ***intuitiv berechenbaren*** *Funktionen*“ (Church-Turing-These)
- Oder: „*Wenn man etwas berechnen kann, kann es auch die Turing-Maschine*“

- Church-Turing-These wird aufgrund fehlender Gegenbeweise akzeptiert
- Möglichkeiten der Turingmaschine stellen die Grenzen des Computers dar, mit Algorithmen Lösungen zu finden (Halteproblem)

Beispiel einer Turingmaschine



(Zustand x, gelesen) ---> (neuer Zustand y , geschrieben, Kopfbewegung)

(z1, 1) ---> (z1, 1, R)
(z1, 0) ---> (z1, 1, R)
(z1, _) ---> (z1, _, H)

(z1, 1) ---> (z1, 1, R)
(z1, 0) ---> (z2, 1, L)
(z2, 0) ---> (z2, 0, L)
(z2, 1) ---> (z2, 1, L)
(z2, _) ---> (z2, _, H)

Beispiel einer Turingmaschine

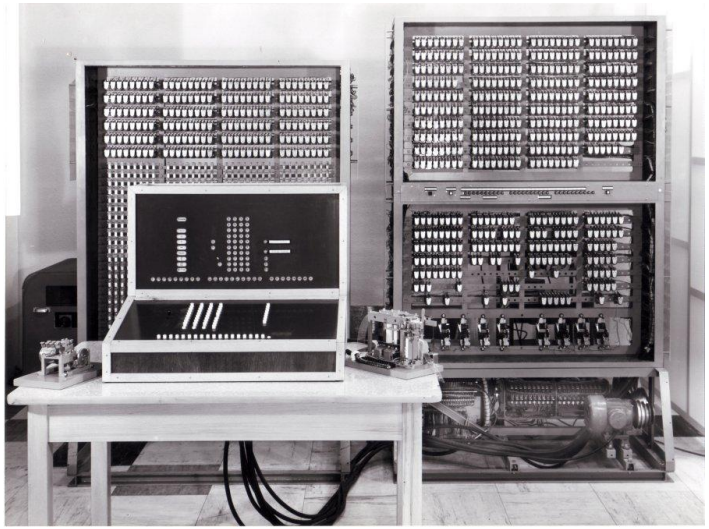
Addition zweier unärer Zahlen:

Programm:

- (z1 ,+) ---> (z1 ,1,R)
- (z1 ,1) ---> (z1 ,1,R)
- (z1 ,=) ---> (z2 ,_,L)
- (z2 ,1) ---> (z2 ,_,H)

Historischer Überblick

1941: Konrad Zuse stellt in Berlin die Z3 fertig

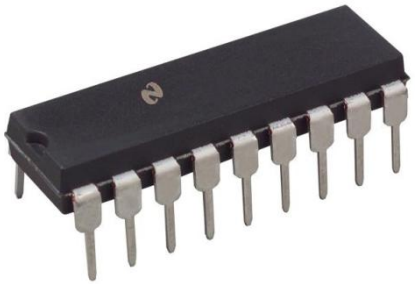


-Erster funktionsfähiger Computer
-Binäre Datenverarbeitung
-Voll programmierbar

1946: Vorstellung des ENIAC in den USA

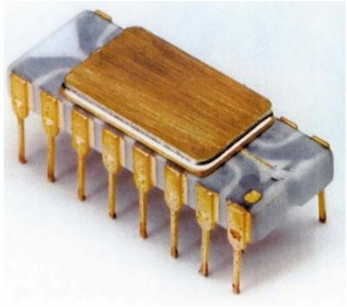
-Erster Computer, der sowohl turingmächtig, programmierbar als auch vollständig elektronisch war
-Dezimale Datenverarbeitung

1964: Patentierung des integrierten Schaltkreises (IC)



-Meilenstein in der Weiterentwicklung der Hardware

Historischer Überblick



1974: Intel präsentiert den „4004“-Mikroprozessor

-Erster Mikroprozessor, der in Serie ging und frei erhältlich war

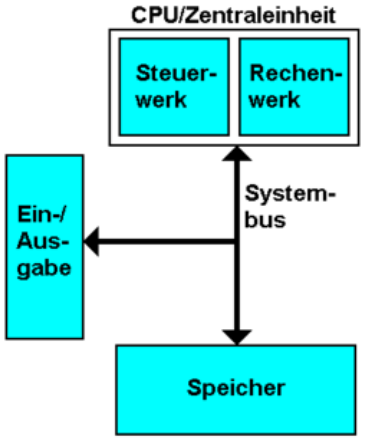
1982: Commodore stellt den „C-64“ vor

- Sowohl als Spielekonsole als auch in der Softwareentwicklung sehr populär
=> Der Computer zieht als Gebrauchsgegenstand in großem Stil in die Haushalte ein

Von-Neumann-Rechner

- Referenzmodell eines modernen Computers, hervorgegangen aus der Turingmaschine
- Unterteilung des Rechners in drei voneinander getrennte Funktionseinheiten:

1) Ein-/ Ausgabewerk
2) Speicherwerk
3) Zentraleinheit / CPU
    i) Steuerwerk
    ii) Rechenwerk (ALU)



- Verbindung der Funktionseinheiten miteinander über den Systembus

-„Konkurrenz“-Modell: Harvard-Architektur

Von-Neumann-Rechner

Ein- / Ausgabewerk
- Verarbeitung der Ein- und Ausgabe (z.B. Tastatur, Monitor)

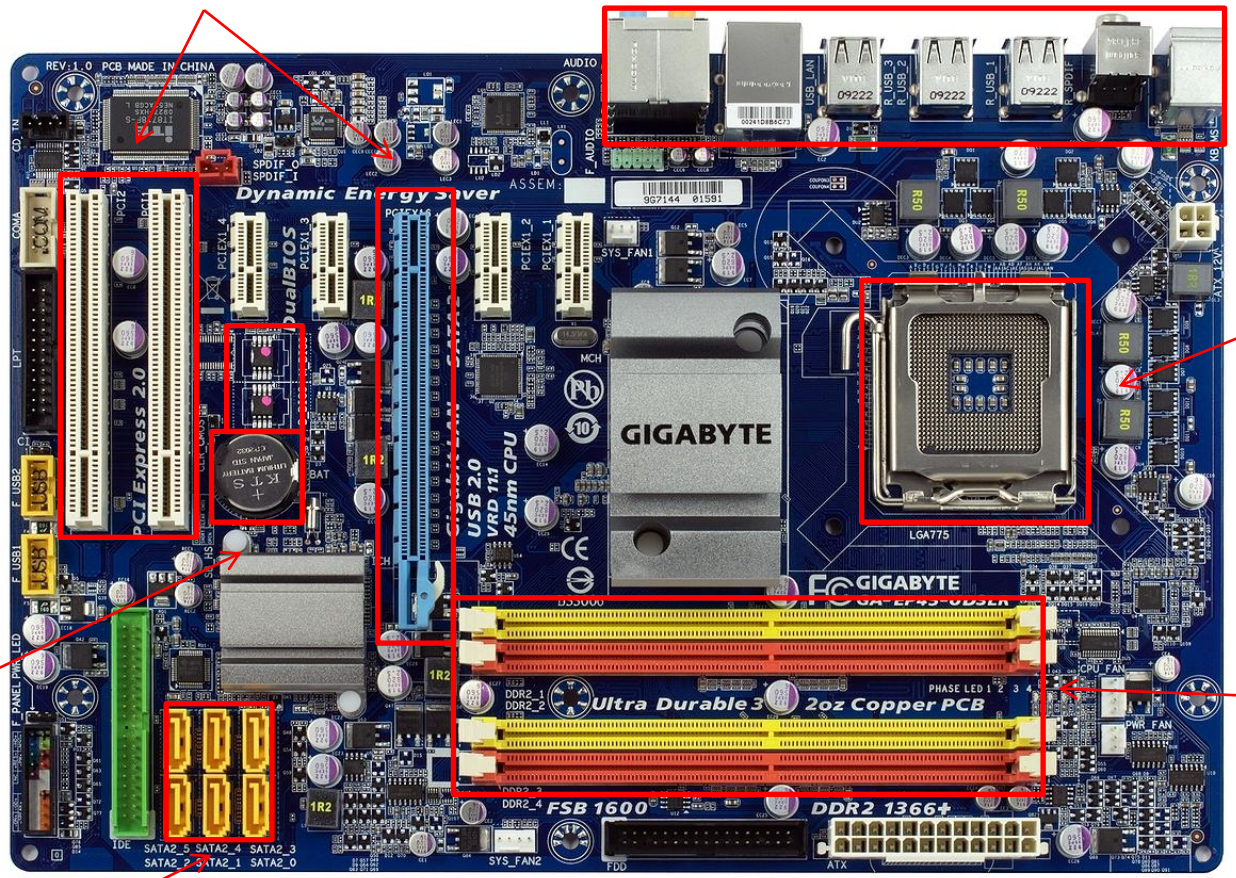
Speicherwerk
- Arbeitsspeicher
- Register (auf der CPU)
    - Befehlszählregister
    - Interrupt-Steuerregister
    - Adressregister
    - usw.

Zentraleinheit / CPU
- Steuerwerk: lädt Befehle, initialisiert das Rechenwerk
- Rechenwerk: Durchführung von Rechenoperationen, logischen Verknüpfungen und Operationsbefehlen (z.B. Laden, Speichern)

Die Hardware

Erweiterungskarten

I/O-Schnittstellen



CPU

BIOS
+
Batterie

RAM

Festplatten

Die Hardware

- Kommunikation mit dem Computer mittels einer Programmiersprache
  - Binärcode:
    - kann direkt vom System ausgeführt werden
    - auf eine bestimmte Plattform zugeschnitten (z.B. Prozessor)

  - Assamblercode:
    - dank Symbolschreibweise einfacher
    - wird vom Assambler nahezu unverändert in Binärcode übertragen

  - Höhere Programmiersprache
    - problemorientiert

```
01010000
01110010
01101111
01100111
01110010
01100001
01101101
01101101
00100000
01110110
```

```
mov ax, data
mov ds, ax
mov es, ax

mov Z1,230
mov Z2,257

mov ax,Z2
add ax,10
mul Z1
```

```
with ReplaceDialog1 do begin
  if frReplace in Options then begin
    i := length(TPlusMemo(PageControl1.ActivePage.Con
    if i > 0 then
      TPlusMemo(PageControl1.ActivePage.Controls[0]).
    ReplaceDialog1Find(FindNext1);
  end;
  if frReplaceAll in Options then begin
    while found do begin
      i := length(TPlusMemo(PageControl1.ActivePage.C
      if i = 0 then Found := False;
```

Die Instruktionsfolge

- Steuerung der Hardware mittels Programmen, welche aus mehreren Instruktionsfolgen bestehen
- Damit die CPU einzelne Instruktionsfolgen bearbeiten kann, gilt:
    1. Die Instruktionsfolge besteht aus Binärcode
    2. Länge und Aufbau der Instruktionsfolge sind festgelegt
    3. Die maximale Länge einer Instruktionsfolge überschreitet nicht die Kapazität des Registers und des Systembusses (üblicherweise 32bit/64bit)

Beispiel einer Additions-Instruktion:

Operationscode	Quellregister	Zielregister	Konstante
100000	00010	00011	0000 0000 0000 1000

-weitere Möglichkeiten: Ladeinstruktion, Sprunginstruktion, usw.

Die Instruktionsfolge

Die Verarbeitung einer Instruktionsfolge erfolgt in fünf Stufen:

1. **Instruction Fetch** *(Pflicht!)*
   - Laden der Instruktion, auf die der Befehlsregisterzähler zeigt, in die CPU

2. **Instruction Decode (ID)** *(Pflicht!)*
   - Ermittelung des Operationscodes, Initialisierung des Rechenwerks

3. **Execution Stage (EX)** *(Optional!)*
   - Durchführung der Rechenoperation im Rechenwerk

4. **Memory Access (MEM)** *(Optional!)*
   - Zugriff auf den Arbeitsspeicher

5. **Write Back (WB)** *(Optional!)*
   - Speichern des Ergebnisses in das Zielregister

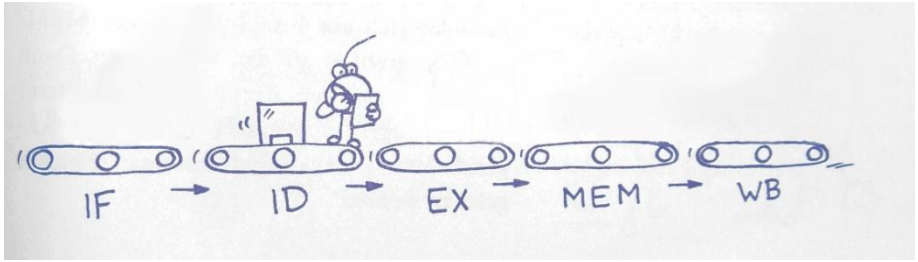
Die Instruktionsfolge

-Der Taktgeber gibt das Arbeitstempo der CPU an

-Messung der Taktfrequenz in Hertz (Hz)
- 1 Hz = 1 durchgeführte Instruktionsfolge / Sekunde

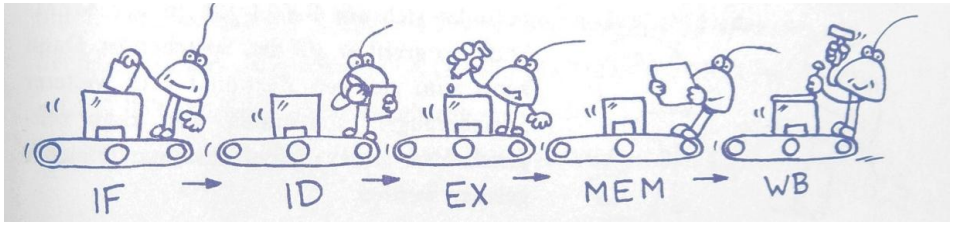
- früher: 1 Takt = die maximal benötigte Zeit für einen kompletten Durchlauf aller fünf Stufen der Instruktionsfolgenbearbeitung

- Problem?



Die Instruktionsfolge

- Lösung? Pipelining!
- Überlappende Ausführung der Instruktionen
- Länge eines Taktes wird auf die maximale Ausführungsdauer einer Stufe herabgesetzt
-Eine Instruktion an sich wird nicht schneller ausgeführt, aber die Zahl an in der gleichen Zeit bearbeiteten Instruktionen verfünffacht sich bestenfalls
- Gleichzeitiger Zugriff auf eine Ressource wird durch Einfügen von *nop (no operation)* verhindert



BIOS (Basic Input Output System):

-Wird noch vor dem Betriebssystem geladen
- Speichert systemrelevante Parameter (z.B. Uhrzeit, Bootreihenfolge, CPU Taktfrequenz)
- Listet die angeschlossene Hardware auf und initialisiert diese
- Überprüfung der Hardware auf Fehler
    - tritt ein Fehler auf, ertönt über den internen Systemlautsprecher ein bestimmter Piep-Code, mithilfe dessen sich der Fehler eingrenzen lässt
- Anschließende Übergabe der Kontrolle an den „Urlader“
    - ggf. Booten von externem Speichermedium (z.B. Diskette, CD, etc.)
    - Booten von Festplatte des gewünschten Betriebssystems

Software

Betriebssystem:

-Bietet eine Schnittstelle für die Hardwarekomponenten und Betriebsmittel unterschiedlicher Firmen
    - Software kann dank Treiber auf die Schnittstellen der Hardware zugreifen
-Verwaltet den Zugriff ausgeführter Software auf die Hardware
    - Verhindert gleichzeitigen Zugriff von Software auf ein Betriebsmittel
- Organisiert die Ausführung von Software
    - Laden notwendiger Variablen aus dem Arbeitsspeicher in die Register
- Abfrage des Interrupt-Steuerregisters
    - Unterbrechung des laufenden Prozesses bei Ablauf einer Zeitscheibe (sog. Timer-Interrupt) oder bei Ereignis seitens einer Hardwarekomponente (sog. I/O-Interrupt)

Multitasking:

-Scheinbar gleichzeitige Ausführung von Prozessen, indem zwischen den verschiedenen Prozessen schnell gewechselt wird
-Prozesse werden ihrer Priorität entsprechend bevorzugt ausgeführt
- Zuweisung sog. „Zeitscheiben“ an die Prozesse

=> Quasiparallele Ablaufplanung

Multiprocessing:

-Echte Gleichzeitigkeit
- Aufteilung der Instruktionsbearbeitung auf Mehrkernprozessoren

- Vorteile:
    - kostengünstiger, da mehrere schwächere Prozessoren
    - leichter mehr Leistung zu erzielen
    - Arbeitsteilung nach Gebieten (z.B. ein Kern für Physikberechnung)

- Nachteile:
    - Software kann ausbremsen, da Instruktionen parallelisiert werden müssen

Ausblick (gegenwärtig)

-Ausbau des Multiprocessing durch den Einsatz von mehr Kernen
- Voranschreitende Miniaturisierung der Bauelemente
    - Prozessorkerne in der Größe von roten Blutkörperchen (ca. 7,5 µm) in knapp 10 Jahren (25.10.11, Mike Muller, Chief Technical Officer von ARM)
- Mooresches Gesetz: Verdoppelung der Anzahl der Transistoren pro Flächeneinheit alle 18 bis 24 Monate
- Grid-Computing: Zusammenschluss mehrerer Computer zu einem einzigen „Supercomputer“
    - Bsp: Blizzard im DKRZ
        - 8448 Prozessoren
        - über 20 TeraByte Speicher
        - Leistung von 158 TeraFLOPS (Floating Operations Per Second)
        - Zum Vergleich: K Computer (Japan) wird nach Fertigstellung eine Leistung von ca. 10 PetaFLOPS aufweisen
-Cloud-Computing: Auslagerung von Ressourcen
- Erweiterung der I/O-Schnittstelle (z.B. Bewegungssteuerung, augmented reality)

Ausblick (zukünftig)

- Einsatz von Licht als Informationsträger

- Biocomputer (Einsatz von DNS als Speichermedium)

- Quantencomputer

Empfohlene Literatur / Quellen

-Johannes Magenheim, Thomas A. Müller: Informatik macchiato. Pearson Studium, 2009

-Peter Winkler: Computer Lexikon 2008. Markt+Technik, 2007

- Scott Müller: PC-Hardware Superbibel. Markt+Technik, 2005

- „Matheprisma“, http://www.matheprisma.uni-wuppertal.de/index.htm, 29.10.11

- Boris Jakubaschk, „Computergeschichte“, www.computergeschichte.de, 29.10.11