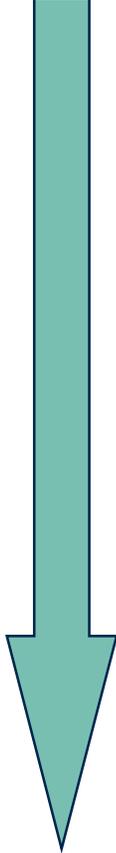




Der “von Neumann” Computer

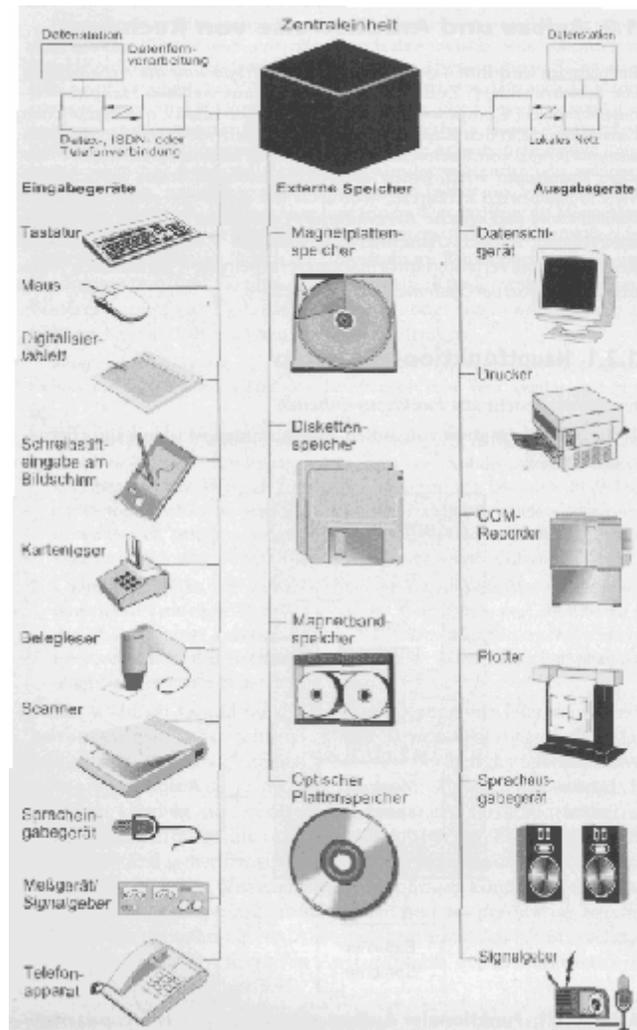
Grundlagen moderner Computer Technologie

Der moderne Computer ein weites Spektrum



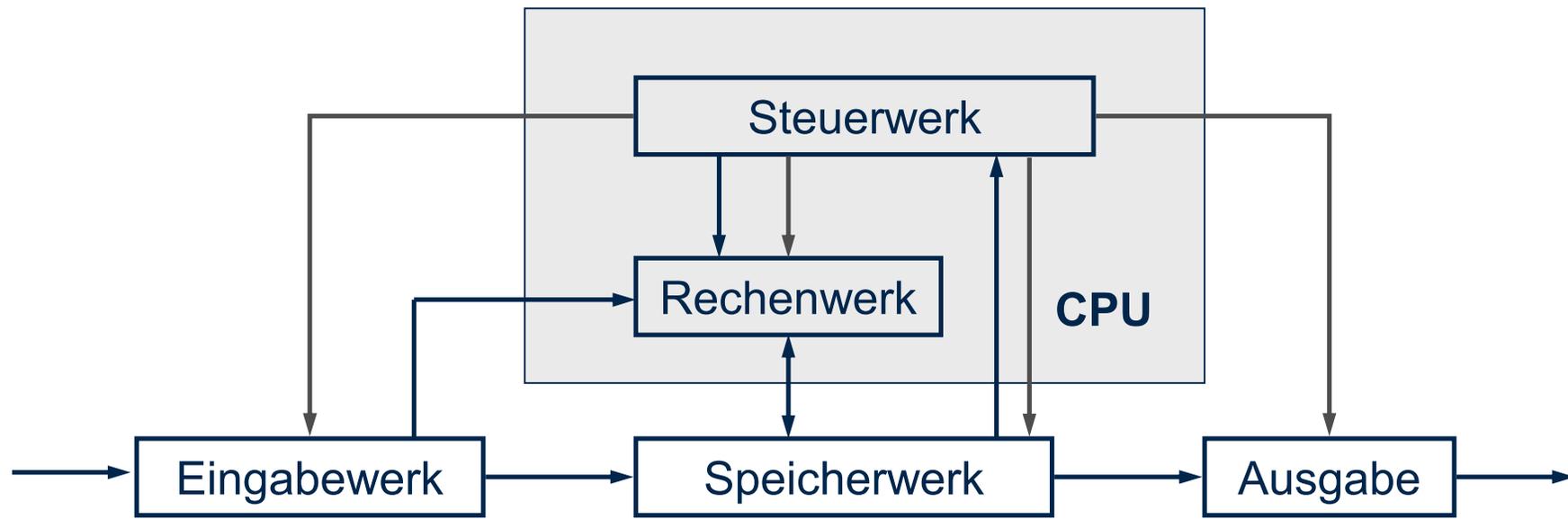
<u>Typ</u>	<u>Preis</u>	<u>Anwendungsbeispiel</u>
§ „embeded“ Computer	10-20 \$	in Autos, Uhren,...
§ Spielcomputer	100-200\$	Xbox, Cube,..
§ Personalcomputer	1.000\$	Desktop, Notebook
§ Server	10.000\$	Netzwerk Server
§ Cluster (von Servern)	100.000\$	Abteilungsrechner Business kritische Anwendungen
§ Großrechner (Mainframe, High end Server)	1.000.000\$	Bank, SAP,...
§ Supercomputer	10.000.000\$	Simulation, Wettervorhersage

Der moderne Computer der innere Aufbau



- § Eingabeeinheiten
 - ∅ Tastatur, Maus,
- § Ausgabeeinheiten
 - ∅ Bildschirm,...
- § Speicher
 - ∅ Memory, (Disks), Band,..
- § Datenwege
 - ∅ Bus (USB, PCI,)
- § CPU
 - ∅ Rechenwerk
 - ∅ Leitwerk

Der von Neumann Computer (abstrakte Sicht)



CPU = Central Processing Unit = Zentraleinheit

—————> : Datenfluß

—————> : Steuerfluß

(nach Burks, Goldstine und von Neumann, 1946)

Der von Neumann Computer - Das Konzept

1. Der Computer lässt sich räumlich und logisch in die folgenden Bestandteile einteilen

⇒ **Speicherwerk:**

Speicherung von Programmen und Daten

⇒ **Rechenwerk:**

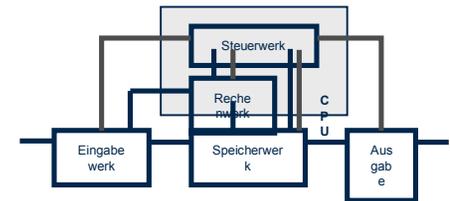
verarbeitet die Daten. D.h. Ausführung arithmetischer und logischer Operationen.

⇒ **Steuerwerk (auch Steuereinheit oder Leitwerk genannt):**

Steuerung des Programmablaufs.

⇒ **Eingabe- und Ausgabewerk:**

Eingabe von Programmen / Daten in den Speicher; Ausgabe von Ergebnissen vom Speicher „nach außen“



Der von Neumann Computer - Kernaussagen

2. Struktur eines Computers ist **unabhängig von den zu bearbeitenden Problemen:**

- **Für jedes Problem** wird eine Bearbeitungsvorschrift (**Programm**) „von außen“ eingegeben und im Speicher abgelegt. Dieses Programm macht den Computer erst arbeitsfähig.

3. **Programme und Daten im selben Speicher**

Der von Neumann Computer - Kernaussagen

4. Der Zentralspeicher ist in Zellen gleicher Größe eingeteilt, die durch fortlaufende Nummern (Adressen) bezeichnet werden

- **Byte:** Speicherelement der Größe 8 Bit
- **Wort:** unterschiedliche Verwendung

0	0011 0011 1101 1100
1	0011 0101 1101 1100
2	0011 0000 1101 1100

- als Synonym zu Speicherelement

Fest Größe - Typisch: $k \in \{8, 16, 32, 64\}$

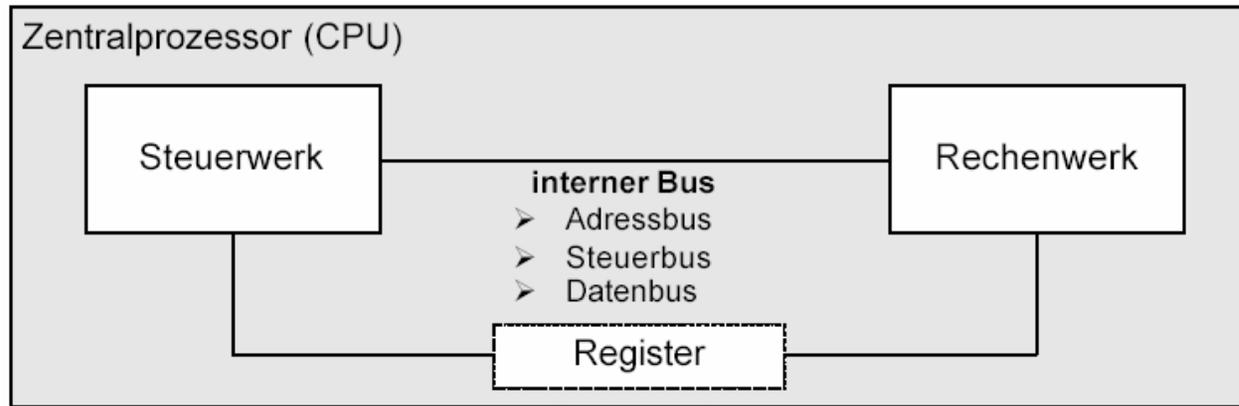
- als Bezeichnung für die Zusammensetzung von 2 oder 4 (aufeinanderfolgenden) Bytes

- Speicherelemente sind die kleinsten adressierbaren Einheiten
- Über die Adresse kann der Speicherinhalt direkt abgerufen werden

Der von Neumann Computer -Kernaussagen

5. Das Programm besteht aus einer Folge von Befehlen die im Allgemeinen hintereinander ausgeführt werden. (Dadurch wird das Holen des nächsten Befehls einfacher, weil auf diese Weise die Befehlsadresse nur um 1 erhöht werden muss)
6. Von dieser sequentiellen Ausführung kann auch abgewichen werden. Hierzu existieren bedingte und unbedingte Sprungbefehle die die Programmausführung an einer anderen Speicherzelle fortsetzen. Die bedingten Sprungbefehle hängen von gespeicherten Werten ab.
7. Die Maschine nutzt Binärcodes, Zahlen werden dual dargestellt.

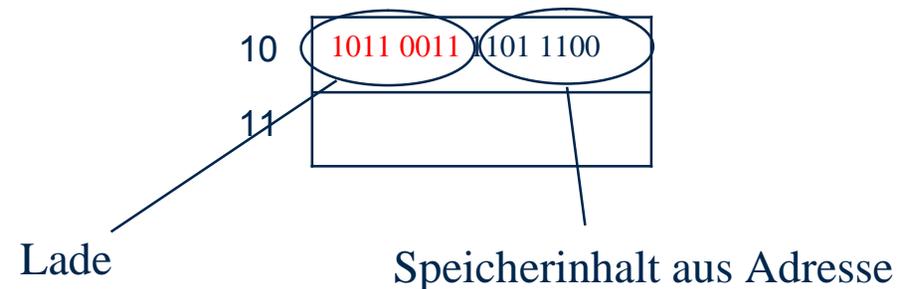
Der von Neumann Computer - CPU



Welche Aufgaben haben nun das Steuerwerk/Leitwerk und das Rechenwerk?

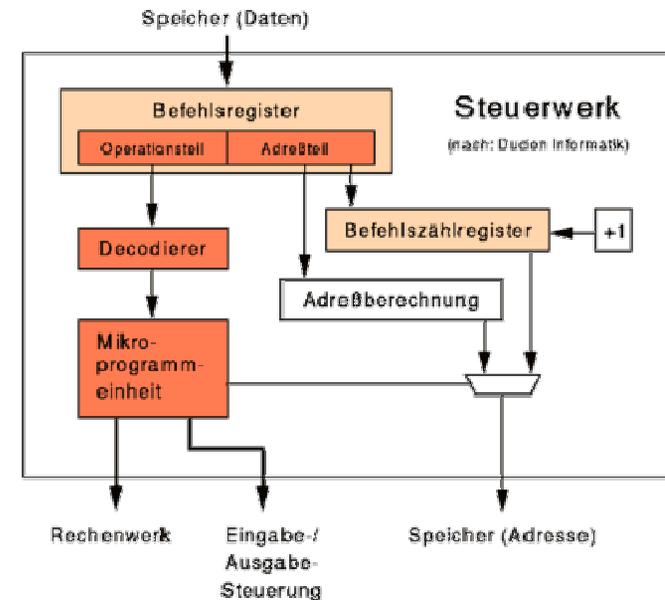
Der von Neumann Computer - Steuerwerk

- § Das Steuerwerk ist für die Steuerung und Koordinierung zuständig
- ∅ Es steuert die Reihenfolge in der die Befehle eines Programms ausgeführt werden. In einem speziellem Register (Befehlszähler) merkt es sich die Speicheradresse aus welcher der Befehl gelesen werden soll. Nach dem lesen des Befehls wird der Befehlszähler um 1 erhöht.
 - ∅ Der Befehl selbst wird im Befehlsregister gespeichert.
 - ∅ Jeder Befehl (Maschinenbefehl) besteht aus einem
 - Operationsteil (Was?)
 - Operandenteil (Womit?)



Der von Neumann Computer - Steuerwerk (2)

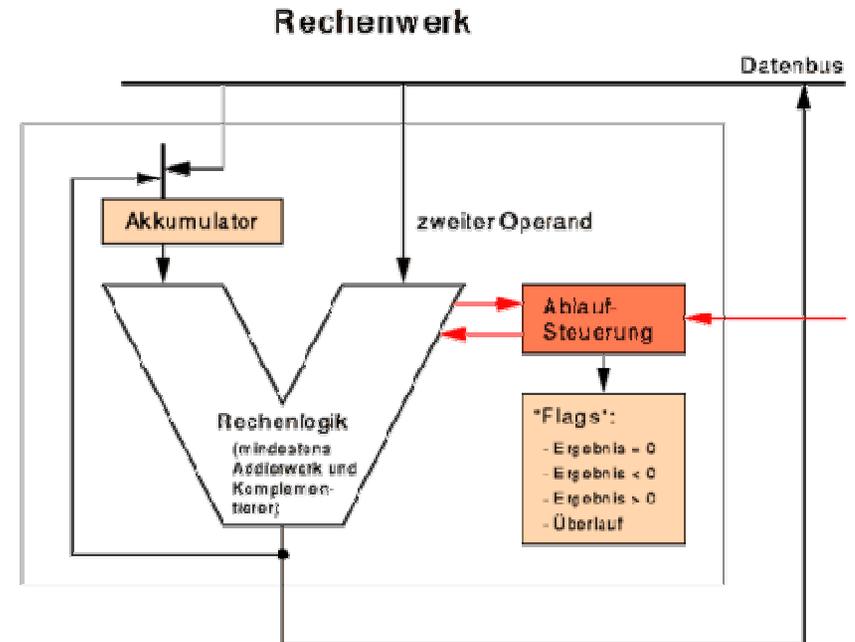
- ∅ In der **Befehlsdecodiereinrichtung** wird der Operationsteil entschlüsselt und die zugehörigen Operandenadressen werden errechnet



- ∅ Es werden dann die zugehörigen Signale mit den benötigten Informationen an die zuständigen Register des Leitwerks, den Arbeitsspeicher, u.s.w. weitergeleitet
- ∅ Der Status in dem sich das Programm befindet wird in einem Statusregister festgehalten.

Der von Neumann Computer - das Rechenwerk

- ∅ Das **Rechenwerk** verarbeitet die vom Leitwerk bezeichneten Daten
- ∅ Das Rechenwerk ist in der Lage
 - Arithmetische Funktionen (+, -) auszuführen und
 - Logische Verknüpfungen (vergleichen, verschieben) durchzuführen



- ∅ Das Rechenwerk besteht im wesentlichen aus
 - Registern (z.B. Akkumulator) und
 - Binären Schaltwerken (Rechenlogik oder ALU) (Addition einzelner Binärstellen)

Zusammenarbeit von Steuerwerk und Rechenwerk

- § Das Steuerwerk (oder Leitwerk) koordiniert der Programmablauf, während das Rechenwerk Verknüpfungen und Rechenoperationen durchführt
- § Das Steuerwerk veranlasst das Rechenwerk, die im Operationsteil des Befehls angegebene Operation mit dem im Operandenteil angegebenen Operanden auszuführen.
 - ∅ Es kann durchaus mehr als einen Operanden geben!
- § Das Steuerwerk entschlüsselt die Befehle und das Rechenwerk führt sie aus.

Der von Neumann Computer - von Neumann-Zyklus

§ Die folgenden Arbeitsphasen nennt man von Neumann-Zyklus

1. Befehl aus dem Hauptspeicher holen
 2. Befehl decodieren (Operanden-Adressen oder Sprungziel errechnen)
 3. Operanden holen
 4. Befehl ausführen, d.h. Operanden verarbeiten, ggf. Ergebnis speichern
 5. Befehlszähler erhöhen
- Die Phasen 1 bis 3 (Fetch & Decode) und 5 übernimmt das Rechenwerk,
 - die Phase 4 (Execute) das Rechenwerk

Der von Neumann Computer - von Neumann-Zyklus

- § Die Folge von:
 - Befehl holen $\langle \rangle$ Befehl ausführenwird ständig wiederholt bis das Programm beendet wird.
- § Aufgrund der Architektur werden die Befehle nacheinander (sequentiell) und nicht parallel (gleichzeitig) verarbeitet.
 - ∅ Dies bezeichnet man auch als den von Neumann'schen Flaschenhals

Der von Neumann Computer - Beispiel (in Assembler)

- § Anstelle alles in Maschinencode (binär) darzustellen bedient sich der Mensch gerne einer „höheren“ einfacheren Sprache.
- § Eine Sprache die recht nahe am Maschinencode ist, ist Assembler
 - ∅ Beispiel:

Adresse	Inhalt des Speichers	Assemblerbefehl	Bedeutung
01300	AD 13 60 (hex)	LDA \$1360	Lade den Akku mit Speicherzelle 1360

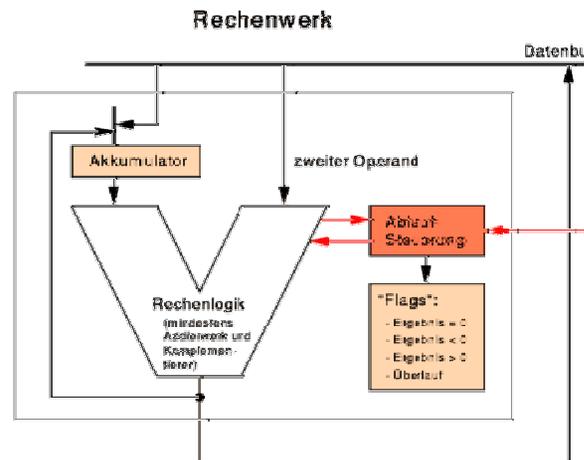
1010 1101 0001 0011 0110 0000

Die Bedeutung des Maschinencodes hängt natürlich von der verwendeten CPU ab. Der Assembler ist dagegen allgemeiner

Der von Neumann Computer - Beispiel

Speicheradr.	Inhalt des Speichers	Assemblerbefehl	Kommentar
01300	AD 13 60	LDA \$1360	; Akku erhält den Inhalt der Speicherzelle 1360
01318	6D 13 61	ADC \$1368	; Addiere zu Akku den Inhalt der Speicherzelle 1368
01330	8D 13 62	STA \$1370	; Schreibe Inhalt des Akku in Speicherzelle 1370
01348	00	BRK	; Beende Programm

Speicheradr.	Inhalt des Speichers
01360	00
01368	02
01370	00



Speicheradr.	Inhalt des Speichers
01360	00
01368	02
01370	02
:	:

Der von Neumann Computer - Maschinencode <> Assembler

- § Die CPU versteht einen bestimmten Satz von Befehlen: „Instruction Set“
- § Kennt man die Instruction Set Architektur (ISA) also den Befehlsatz, die Befehlsformate, Adressierungsarten, die Register und Speicherplätze die Programmierer verändern kann, dann:
 - ∅ Kennt man alles was ein Maschinensprach-Programmierer wissen muss, um ein korrektes Programm für diese Maschine schreiben zu können
- § Die ISA ist in der Regel in „Manuals“ beschrieben
 - ∅ Z.B. die IA32-Architektur wie sie von Pentium oder AMD K6 Prozessoren verwendet wird

Der von Neumann Computer - Und warum dann C++

- § Natürlich kann man alle Programme direkt in Maschinensprache (Assembler) implementieren
 - ∅ Nicht immer ist das so ganz einfach !!!!!
 - ∅ Oft nutzt man besser einen mehr logischen / objektorientierten Programmansatz statt eines sehr Hardware nahen Modells
- § Außerdem fehlt uns noch das Betriebssystem und seine Dienste wie:
 - ∅ Verwaltung von Festplatten, Zubehör wie Maus, Tastatur,...die man mit einer Hochsprache wie C++ ganz einfach nutzen kann:
„cin >> a;“
x = y + z;
if (i < 10)