



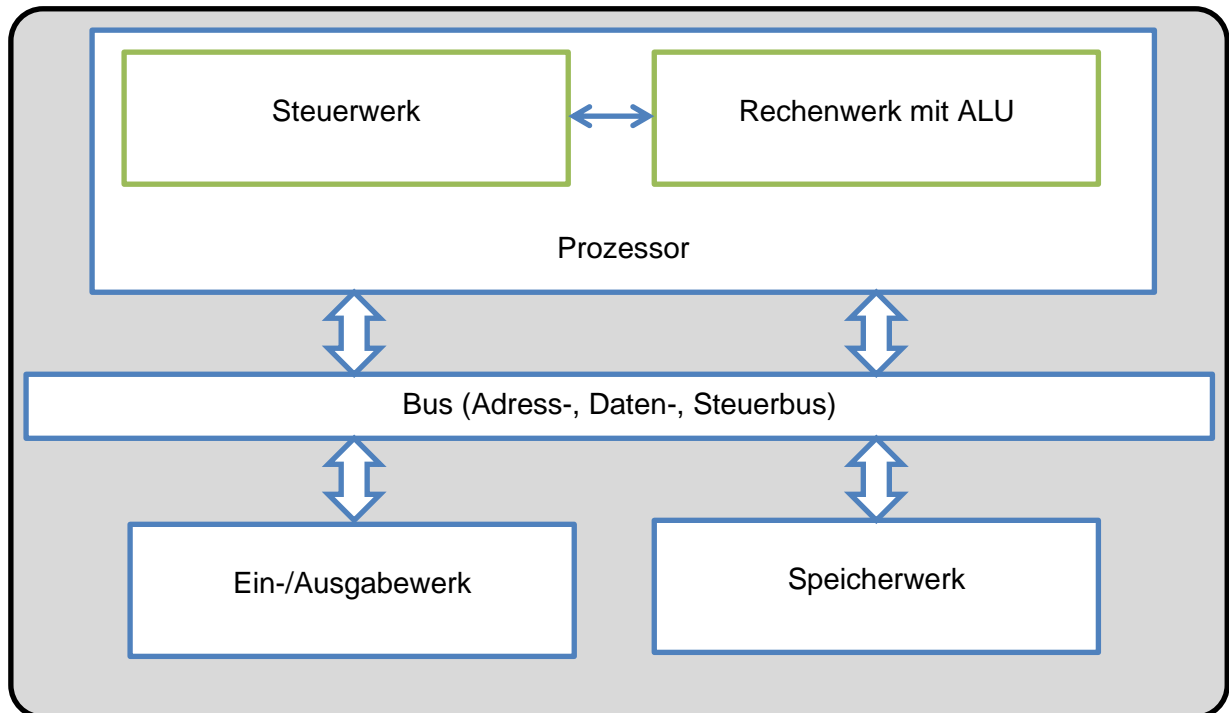
John-von-Neumann-Rechner

Name:

Vorname:

Klasse:

Aufbau



Prinzipien

- 1) Der Rechner besteht stets aus:
 - a) dem **Rechenwerk** (ALU - Arithmetic Logic Unit)
führt Rechenoperationen und logische Verknüpfungen durch
 - b) dem **Steuerwerk** (Control Unit)
interpretiert die Anweisungen eines Programms und steuert die Befehlsabfolge
 - c) dem **Speicherwerk** (Memory Unit)
speichert Programme und Daten
 - d) dem **Eingabe-/Ausgabewerk** (I/O Unit)
steuert die Ein- und Ausgabe von Daten zum Anwender (Tastatur, Bildschirm) oder zu anderen Systemen (Schnittstellen)
 - e) dem **Verbindungssystem** (Busbar - Sammelschiene)
verbindet die Komponenten des Rechners untereinander
- 2) Der Rechner ist universell einsetzbar und löst eine Vielzahl von Problemen. Dazu ist dem Gerät von außen ein Programm in den Speicher einzugeben.
- 3) Im Speicher liegen Programme, Daten und Ergebnisse.
- 4) Der Speicher besteht aus gleichgroßen, fortlaufend nummerierten Zellen. Über die Nummer (Adresse) lässt sich die Speicherzelle direkt lesen oder schreiben.
- 5) Ein Programm ist eine Folge von Anweisungen, das entsprechend seiner Abarbeitungsfolge hintereinander im Speicher steht.
- 6) Sprungbefehle erlauben Abweichungen von der sequenziellen Abarbeitung der Befehle.
- 7) Es gibt mindestens
 - a) *arithmetische Befehle*
für Addition, Multiplikation, Konstanten laden, ...
 - b) *logische Befehle*
für Vergleiche, sowie logisches not, and, or,



John-von-Neumann-Rechner

Name:

Vorname:

Klasse:

- c) *Transportbefehle*
für Bewegungen von Daten zwischen Speicherwerk und Rechenwerk sowie zur Ein- und Ausgabe
 - d) *bedingte Sprünge*
für die Implementierung von Wiederholungen und Bedingungen
 - e) *sonstige Befehle*
für Bitschieben, Unterbrechen, Warten, ...
- 8) Alle Daten werden binär codiert. Dem Inhalt einer Speicherzelle kann man nicht ansehen, ob es sich um einen Befehl, ein Datum oder eine Adresse handelt.

Arbeitsweise

Die Befehlsabarbeitung geschieht in den Phasen:

