

Informatik

Vorlesung

Grundlagen Hardware und Betriebssystem

Prof. Dr.-Ing. Thomas Wiedemann
Fachgebiet Informatik / Mathematik



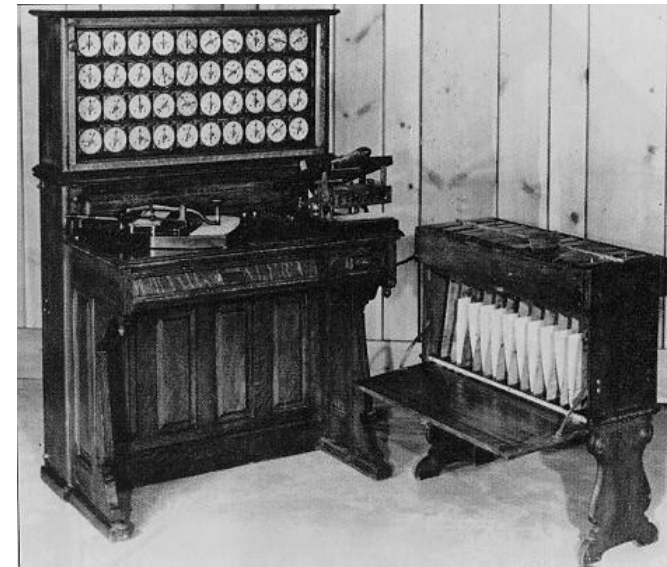
- Historischer Rückblick
- Grundlagen der Logikschaltungen
- Aufbau komplexerer Schaltungen
- Prinzipielle Funktionsweise gegenwärtiger Rechner
- Betriebssystem

Kurze Historie des Rechnens:

- „am Anfang waren die Finger zum Zählen und Rechnen
- zusätzliche Hilfsmittel für mehr als 10 Objekte: Knoten, Steine ...
- wichtige Neuerung: Position der Zahl bestimmt den Gesamtwert und nicht das Symbol selbst (Abkehr vom römischen System)
- einfache mechanische Hilfsmittel: Abakus, Tafeln, Rechenschieber
- mechanische Geräte zum Rechnen ab 18. Jhd. (alle Grundrechenarten)

Erste maschinelle Datenverarbeitung

- Auswertung der USA-Volkszählung 1886 durch IBM
- Erfassung auf Lochkarten, elektrische Kontakte und Zähler (V = einige Karten pro Sekunde)
- ab 1936 erster mechanische Rechner Z1 durch Konrad Zuse



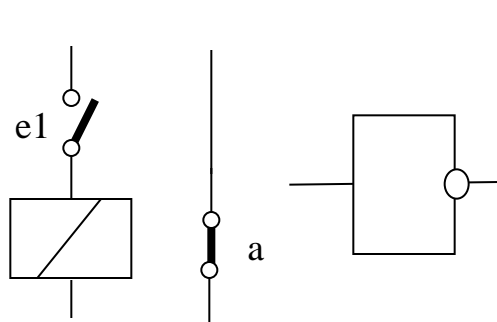
Quelle: 8bit-museum.de

Elektrische und elektronische Grundschaltungen

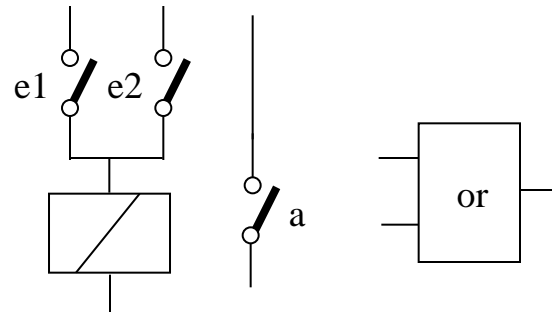
- Im Vergleich zu rein mechanischen Geräten (Masse -> Trägheit) sind elektrische und elektronische Schaltelemente bis zu 10^9 mal schneller
- erste Rechner auf Relais-Basis (ca. 20-200 Schaltop. pro s)
- von 1940 bis 1970 mit Röhren (ca. 2000-500.000 Schaltop pro s)
- ab 1970 mit Transistoren (ab 1975 mehrfach pro Chip, bis 20 GHz)
- **Grundlegende Abbildung der logischen Grundfunktionen**

(prinzipiell analog für alle Technologien - Bsp. Relais-technik, e-Eingänge a-Ausgang)

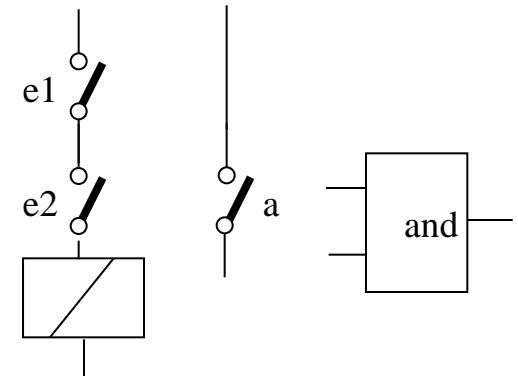
NOT (Inversion)



OR (logisches Oder)



AND (logisches Und)

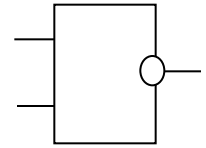


- Abstraktion von der konkreten Realisierung mittels universeller Schaltsymbole

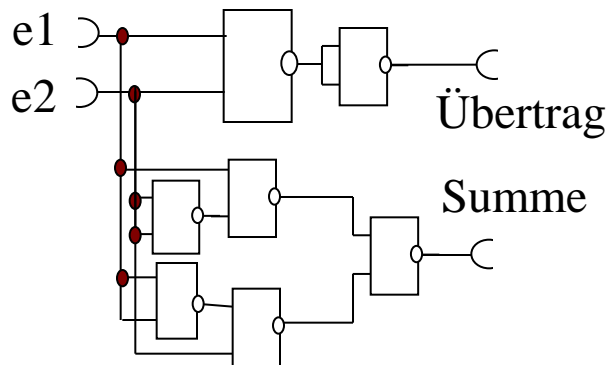
Aufbau komplexerer Funktionsblöcke aus einfachen Logikblöcken

- NAND-Gatter als ein universelles Element :

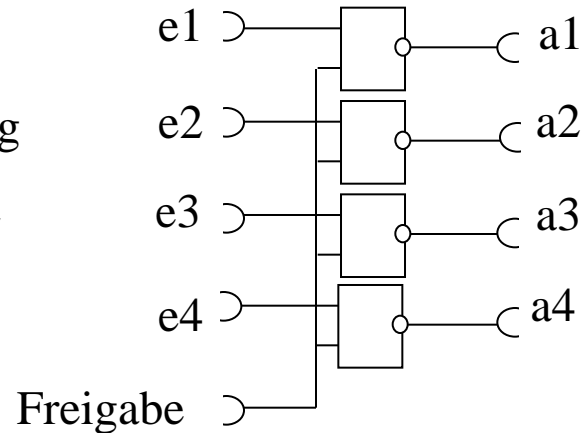
- entspricht UND-Verknüpfung mit anschließender Invertierung
- alle anderen Logikfunktionen ableitbar !! (?? Testat)



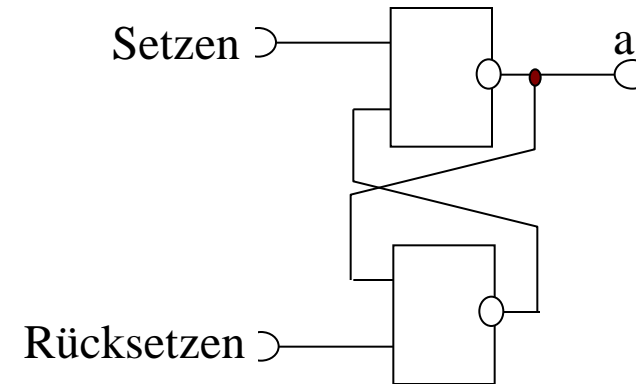
Einfache Addition



Bus-Gatter



1bit-Speicher (Flip/Flop)



Eine Addition von N-bits kann durch eine Kombination von 2 x N dieser Addierer erfolgen (immer 3bit pro Stelle addieren: 2 Bits + Übertrag)

Diese Schaltung dient zum Freigeben und Sperren von Daten und kann auf beliebige Bitanzahlen erweitert werden (16 / 32 / 64 bit)

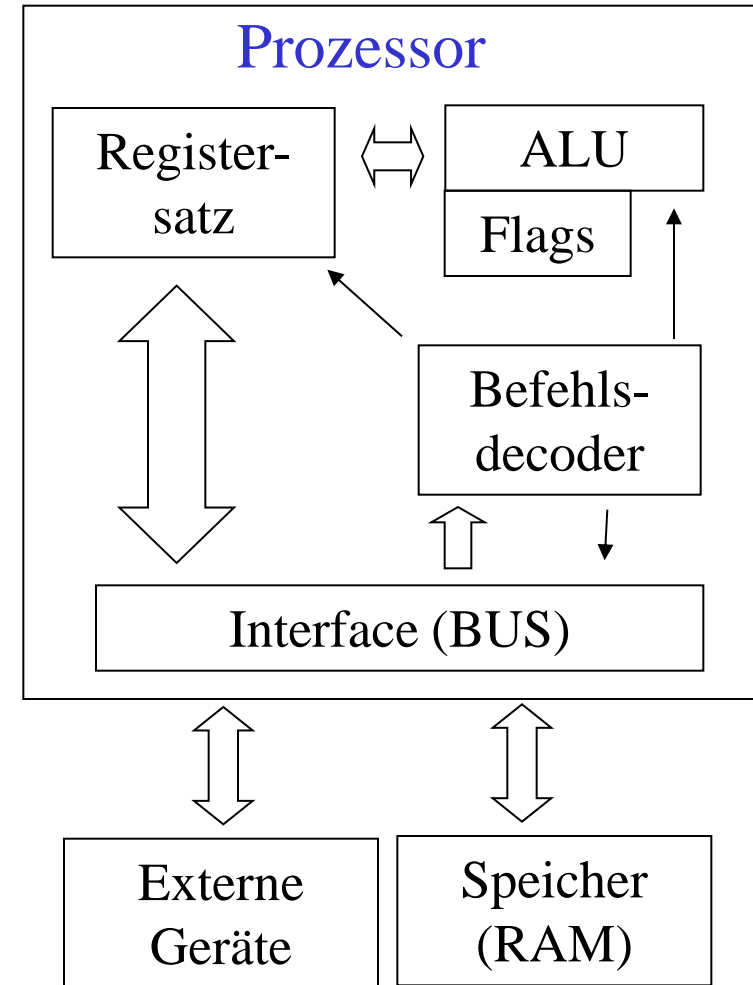
Mit dieser Schaltung werden Bits dauerhaft gespeichert (=statisches RAM und Prozessorregister)

Arbeitsprinzipien von Computern

- Erste Rechner wurden extern "programmiert", d.h die Arbeitsvorschrift wurde von außen eingestellt (verdrahtet)
 - sehr aufwendig und fehleranfällig
- Neues Prinzip eingeführt durch John von Neumann (1946)
 - gemeinsame Ablage von Daten und Verarbeitungsvorschriften
 - Compiler ist Programm, welches ein neues Programm erzeugt
 - Interpretation als Daten oder Operation ergibt sich allein aus Reihenfolge
 - Abarbeitung erfolgt **SEQUENTIELL** nach folgendem Prinzip :
 - **HOLE** Befehl
 - Interpretiere Befehle und falls notwendig :
 - **HOLE** Daten
 - **VERARBEITE** Daten
 - **SPEICHERE** Daten
 - **Erhöhe Befehlszähler und fahre mit nächsten Befehl fort**
- Das Holen und Speichern von Daten erfolgt meist nicht direkt, sondern über die ADRESSE der Daten im Speicher !
- Bei modernen Prozessoren wird versucht, einige der Aktionen parallel durchzuführen, solange sich keine Konflikte ergeben.

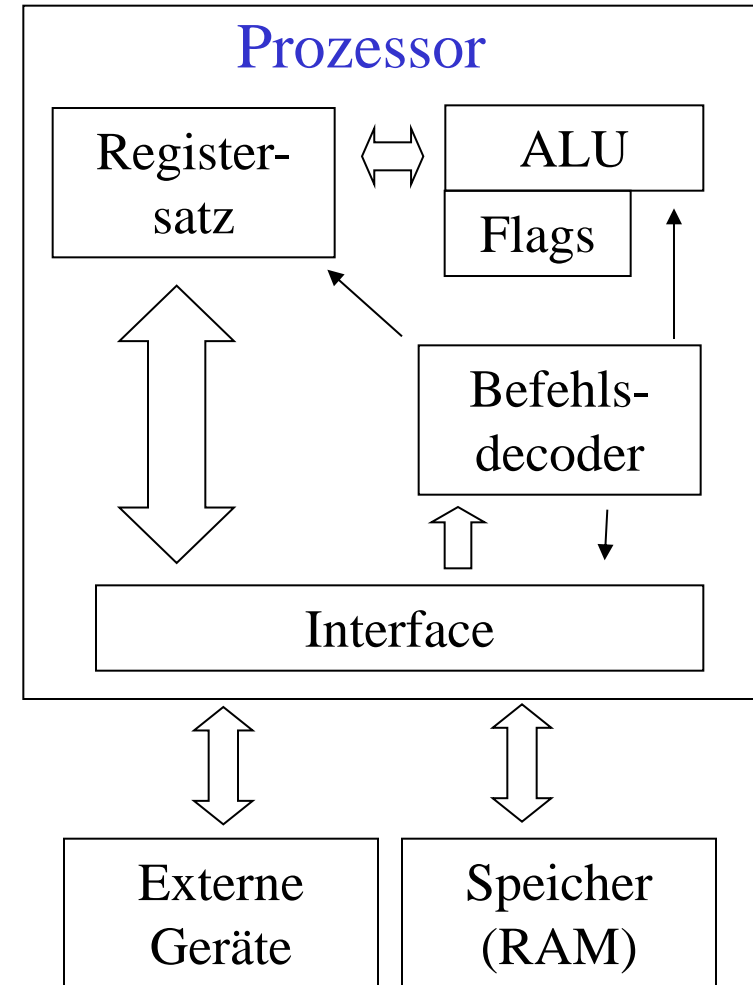
Aufbau eines "von Neumann" - Prozessors

- **Arithmetische Recheneinheit (ALU)**
 - führt alle mathematischen und logischen Operationen aus
- **Registersatz zur Datenhaltung**
 - Universalregister (für Daten)
 - Extrem schneller Speicher (< 1ns)
 - Spez. Adreßregister (zum Speichern und Manipulieren von Adressen)
 - Flagregister (enthält spez. Bits (=Flags) bzgl. der letzten ALU-Operation - Flags: ZERO
NEGATIVE CARRY (Übertrag/ Überlauf)
- **Befehlsdecoder und Steuereinheit (Control)**
 - Steuert alle Vorgänge (Transport der Daten von und zu allen Einheiten, Typ der Operation)
 - definiert durch Mikroprogramm (fest definiert)
 - Operationen werden ebenfalls als Zahlen kodiert
- **Interfaceschaltungen zur Außenwelt**



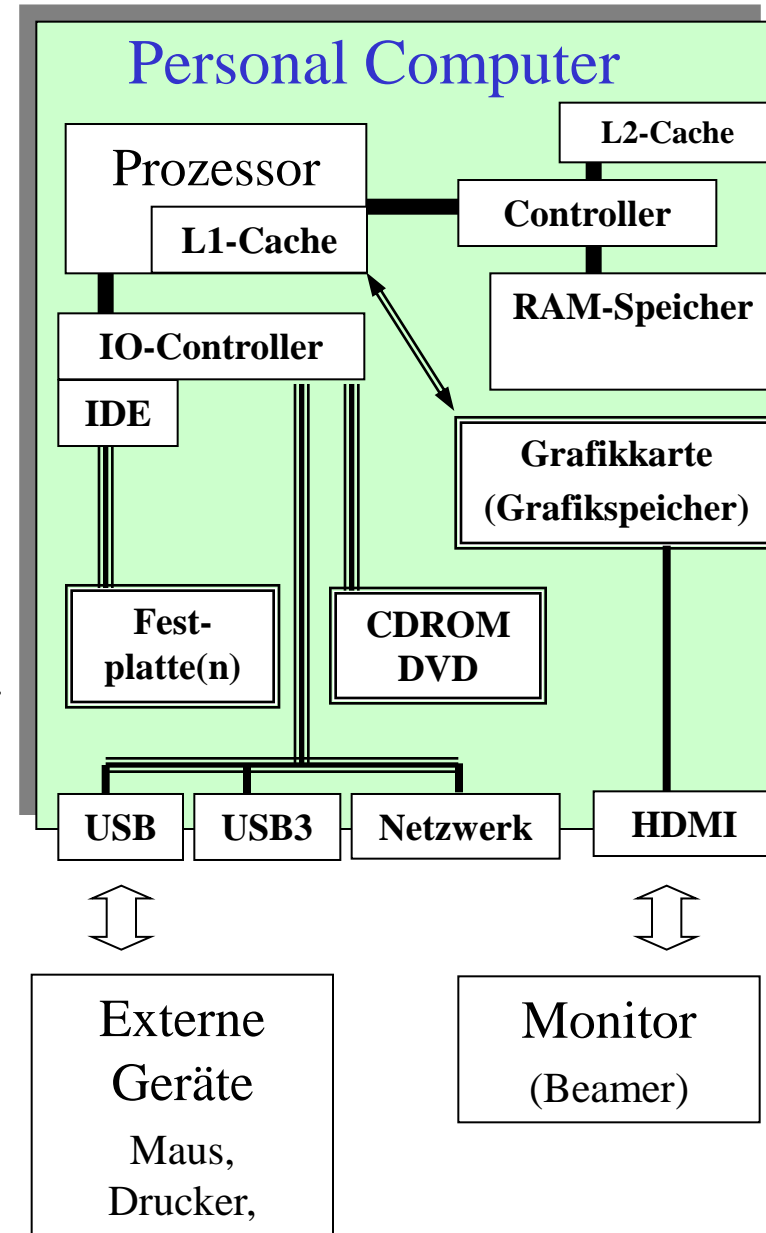
Grundlegende Operationen im Prozessor

- Transportbefehle
 - Daten aus und zum Speicher
 - Daten aus und zu externen Geräten /Grafikkarte/ Tastatur / Diskette ...)
 - Dabei auch indirekte Adressierung über Register (hole Inhalt von Speicherzelle, dessen Adresse in Register 3 steht)
- ALU-Operationen
 - Operation mit Register / Speicher
- Sprungbefehle (unbedingt / bedingt)
 - Springe zu Adresse (unbedingter Sprung)
 - Springe wenn FLAG (z.B. ZERO) zu Adresse
 - Springe NICHT wenn ZERO zu Adresse
 - Rufe Unterprogramm auf (springe zu Adresse und speichere aktuelle Adresse für Rücksprung)



Aufbau eines Rechners

- Prozessor (siehe Folien vorher)
- Controller: zuständig für Timing und Logik aller Operationen außerhalb des Prozessors (heute 1 Chip, früher mehrere)
- Cache- extrem schneller Zwischenspeicher (puffert gelesene Daten)
- (langsamer, großer Halbleiter-)Speicher RAM (Random Access Memory)
- Grafikkarte mit direktem Speicherzugriff
- IO-Controller für Steuerung von
 - Festplatten (IDE-Interface), Diskette
 - Maus
 - Drucker (früher Parallel, heute USB !!!)
 - Modem (früher Seriell, heute USB !!!)



Betriebssystem (BS)

- **Software als Schnittstelle zwischen Anwendungsprogramm und der konkreten Gerätetechnik**
- **ohne Betriebssystem ist die Hardware nicht nutzbar**

Aufgaben

- **Maximale Abstraktion** von den spezifischen Eigenschaften der Geräte (Größe, Art und Eigenschaften der Geräte sollen keine Auswirkungen auf die Programme haben, ansonsten muß jedes Programm angepaßt werden)
- **Flexible Anpassung** an neue Hardware (BS darf nicht monolithisch oder starr für einen Rechner sein, sondern muß dessen Umkonfiguration unterstützen)
- **Wahlfreie Ausführung verschiedener BS** auf einem Rechner (heute Windows, morgen LINUX)
- **Organisation des parallelen Ausführens verschiedener Anwendungsprogramme** (Multitasking) und mehrerer Nutzer (Multiuser)

Maximale Abstraktion:

- Bereitstellung sehr allgemeiner Zugriffsregeln - und konzepte (logisches Laufwerk „A“=Diskette1, „C“=Festplatte1 , Internet-URL)
- Organisation der Daten in allgemeinen, wiederverwendbaren Strukturen (Datei, Verzeichnis, Web)

Flexible Anpassung

- **Treiberkonzept:** kleine Spezialprogramme für spezifische Geräte (auf der Eingangsseite Datei- oder Verzeichniskonzept, auf der Ausgangsseite sehr spezielle Hardwaresignale oder Bitmuster)
- **Gliederung des BS** in: BIOS (unterste Ebene auf Mainboard-Chip), Festplatten BIOS-unterste Ebene auf Massenspeicher, DOS (Disk Operation System)
- **hierarchischer Aufbau:** beginnend mit ganz einfachen Befehlen für Basisdatenaustausch (Lese/Schreibe einen Block mit 512 Byte Daten, über Dateioperationen (Öffne /Lese / Schreibe Datei), bis zu komplexen Operationen (Durchsuche alle Dateien nach dem Wort „Rechnung“)

Wahlfreie Ausführung verschiedener BS

- Konzept des BOOTSTRAP (davon kommt auch der Begriff Booten)
- BIOS-liest den ersten SEKTOR der Festplatte ein (512 Byte) und startet ein kleines Programm in diesen Daten (Bootstraploader)
- dieses Programm liest von einer anderen (genau definierten) Stelle der Festplatte den eigentlichen Betriebssystemkern und startet es
- der BS-Kern beginnt mit dem Laden aller weiteren Bestandteile und der notwendigen Treiber
- Weitere Anwendungsprogramme können automatisch gestartet werden (Datei Autoexec.bat bzw. Autostart-Ordner)

Achtung:

- die Datenstrukturen zur Organisation von Dateien können bei den BS unterschiedlich sein (Austausch dann nur durch spezielle Konvertierungsprogramme)

Multitasking-Konzept

- Gleichzeitiges, paralleles Ausführen mehrerer Programme
- Grundprinzip:
 - Jedes Programm erhält nur einen Teil der Rechenzeit (Zeitscheibe) und wird ansonsten inaktiviert
 - Zuteilung der Zeitscheiben erfolgt nach Priorität des Programms oder auch nach Bedarf
- Zwei Arten des Multitasking
 - Kooperatives Multitasking: Wechsel hängt von Mitarbeit der Programme ab, inkooperative Programme können das System lahm legen (Windows 3.x)
 - Preemptives Multitasking : BS selbst vergibt die Rechenzeit (ab Win95, UNIX) und kann Programme auch zur Pause zwingen
 - altes DOS war nicht direkt multitaskfähig (nur über spezielle Treiber mit Interrupts)

Multiuser-Konzept

- Aufteilung der Rechenzeit für verschiedene Nutzer analog zu Multitasking
- Zusätzlich Berücksichtigung von Zugriffsrechten und Nutzerprioritäten
- Windows verwaltet zwar verschiedene Nutzer, ist aber nicht direkt Multiuser-fähig

Historische Entwicklung der Betriebssysteme

Anfang der Rechentechnik

- BS nur für Tastatur und Bildschirm (Festplatte war Aufgabe des Programmierers)

DOS (Disk Operation System = Festplatten-basierte Betriebssysteme)

- bis auf den BIOS-Loader alles auf Festplatte (oder Diskette oder CD-ROM)
- Bereitstellung relativ komfortabler Funktionen zur Verwaltung der Massenspeicher als Systemprogramme (copy, dir, find) oder als Systemfunktionen für Anwendungsprogramme

Grafische Betriebssysteme (WINDOWS...)

- volle Unterstützung aller grafisch-interaktiven Handlungen (einheitliche Dialoge, standardisierte Darstellungen der Rahmen und Befehlselemente)
- Bessere Unterstützung der Peripherie (einheitliches Treibermodell für Drucker, Soundkarten,)
- Speicherverwaltung : Bereitstellung virtuellen Speichers, d.h. per Festplatte wird den Anwendungsprogrammen ein größerer Speicher bereitgestellt, inaktive Programme werden auf den Festplattenbereich ausgelagert (Performanceproblem)
- Ausnahmebehandlung (Abfangen von Fehlern bei der Programmausführung)

Historische Entwicklung der Betriebssysteme II

Erste Betriebssysteme von Großrechnern

- IBM 360 , PDP 11 (Fa. DEC), BS1000 (SIEMENS)

IBM - PC-basierte Betriebssysteme

- MS-DOS (Microsoft) entwickelt 1981 aus QDOS der Fa. Seattle Computer Products
- Konkurrenzprodukte : DR-DOS, CP/M , Novell NetWare (für Server)

Grafische Betriebssysteme auf PC-Basis

- Windows 1.x (1985)-Windows 3.11 (1995) – 16 bit Systeme (ohne Schutzmanagement)
- Win95 (ab 1995) , Win98, WinMe - 32 bit System mit starken Verbesserungen bzgl. Stabilität, Fehlerrobustheit und Dateiverwaltung (lange Dateinamen) , noch auf DOS-Basis
- WinNT (New Technology) -> Win NT 4.0, Win 2000 als profess. bzw. Server-BS
- WinXP, Win 2003, Win Vista, Win7, Win 8, Win10 – Neuentwicklung ohne DOS-Basis

Unix-basierte Betriebssysteme

- Unix - entwickelt durch Thomson/Ritchie (auch C) bei Fa. Bell Labs um 1970
- Verschiedene kommerzielle Versionen : AIX (IBM), HP-UX (HP), Solaris (Sun)
- Linux - Unix-ähnliches, portables BS mit offenem Sourcecode (meist auf PC-Basis)

Weitere Betriebssysteme

- Mac-OS (für Apple-Rechner) , Android (von Google auf Linux-Basis)