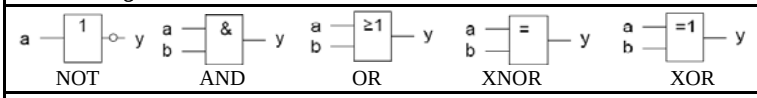
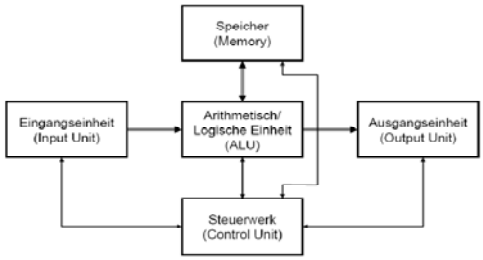
Moores Law (1965 geäußerte Vermutung)

- Anzahl der Transistoren je Prozessorchip verdoppelt sich alle 2 Jahre
- Verarbeitungsleistung der Prozessoren verdoppelt sich alle 1,5 jahre
- Speichergröße vervierfacht sich alle 3 Jahre
- Speicherleistung verdoppelt sich alle 10 Jahre
- Zum gleichen Preis die doppelte Leistung in weniger als 2 Jahren

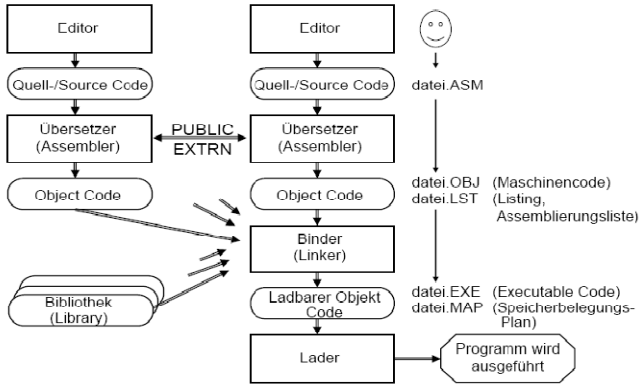
Boolesche Algebra



Blockschaltbild: Von-Neumann-Architektur



Programmentwicklung Assembler



Assembler und Hochsprachen

- Import von Assemblermodulen beim Linken (obj-Dateien)
- Verwendung des inline-Assemblers des jeweiligen Compilers. (asm, __asm)
- Zsf. mehrerer Befehle zu einem Block durch Klammerung (asm{ … })

Interruptverarbeitung

1) HW- oder SW-Interrupt innerhalb eines Progs ist aufgetreten; der gerade in Abarbeitung befindliche Befehl wird zu Ende ausgeführt
2a) Interrupt-Nummer wird mit vier multipliziert
2b) Adr. Des nächsten Programmbefehls des Hauptprog. (Rücksprungadr.) und die Flags werden auf den Stack gesichert.
3) Zugriff auf die Int.-Vektortabelle mit der in Schritt 2a berechneten Adr.
4) Int.-Vektortabelle enthält für alle Interrupts die zugehörigen Einsprung-adressen(Programm n: ISR n)
5) Am Ende des Interrupt-Prog befindet sich ein IRET-Befehl. IRET liest die Flags u die Rückkehradresse vom Stack.
6) Der Prozessor kann die Abarbeitung des Progs mit dem auf den Interruptbefehl folgenden Befehl fortsetzen.

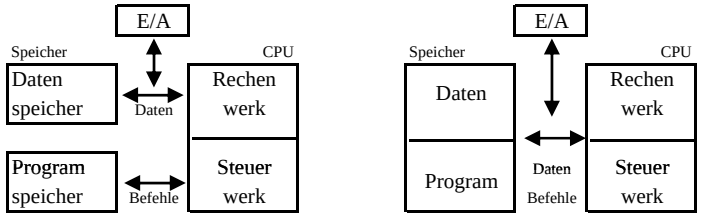
Lokale Prozessor-Klassifikation

- Stack-Architektur
- Akkumulator-Architektur
- Allgemeinzweck-Register-Architekturen (GPR)
  - Register-Register-Maschinen - RISC
  - Register-Speicher-Maschinen - CISC
  - Speicher-Speicher-Maschinen

Schichtenmodell eines Rechnersystems

Anwendungsprogramme	Word, PowerPoint
Höhere Programmiersprachen	Java, C++
Assemblersprachen, Maschinencode	Assembler
festverdrahtete Steuerung	
Funktionseinheiten	ALU
Logikebene	Gatter
Physikalische Ebene	Transistoren

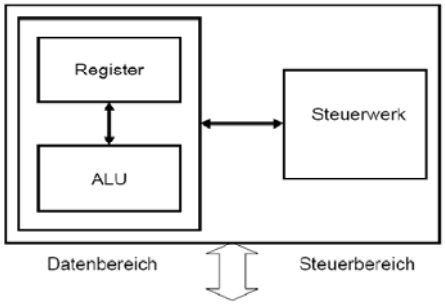
Grundarchitekturen



- Befehle und Daten in getrennten Speichern
- Je ein Befehls- und Datenbus
- Schneller gleichzeit. Zugriff auf Code u Daten
- Hauptanwendung: Signalprozessoren

- Befehle u Daten im gemeinsamen Speicher
- Nur ein Bus für Befehle und Daten
- Flexible Aufteilung des Speichers zw Code und Daten
- Hauptanwendung: Allgemeine Computer

Blockschaltbild: CPU



Einteilung Interrupts

Ein Interrupt ist eine Unterbrechung. Das laufende Prog. wird unterbrochen und ein spezielles Prog. Gestartet, die so genannte Interrupt Service Routine. Nach Beendigung der ISR wird an der ursprünglichen Position im Prog. fortgefahren.
Die Interrupts des 8086 können eingeteilt werden in:

- Software-Interrupts
  Werden vom Programmierer ausgelöst (zB. INT 21H)
- Exceptions
  Werden in Ausnahmesituationen durch den Prozessor selbst ausgelöst
  zB. Auslösen von Interrupt Nr.0 bei Division durch 0.
- Externe (asynchrone) Interrupts
  Werden durch externe bausteine bzw. Ereignisse ausgelöst. zB. Timer

Interrupt-Vektortabelle

Ist ein großer Block reservierter Speicherplätze (1024 Kbyte)
Dort liegen die Interruptvektoren (Einsprungadr) für die 256 möglichen Interrupts. Jeder Interruptvektor besteht aus Segment u Offset, belegt 4 Byte

Parallelrechner-Architekturen

SISD	MISD
Single Instruction Single Data	Multiple Instruction Single Data
zB Von-Neumann	zB Pipelining bei Unix
SIMD	MIMD
Single Instruction Multiple Data	Multiple Instruction Multiple Data
zB MMX, 3D-Now	zB. Mehrprozessorsys.

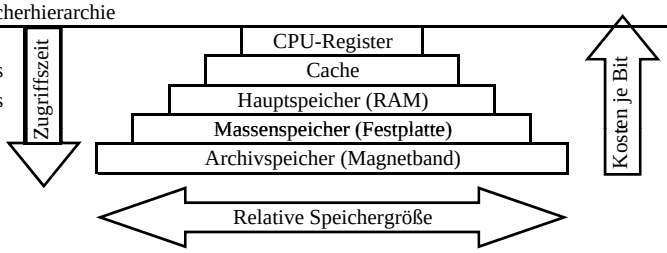
GPR-Architektur

Complex Instruction Set Computer (CISC)	Reduced Instruction Set Computer (RISC)
Mischung aus Akkumulator und Load-Store-Architektur	Load-Store-Architektur
Komplexe Instruktionen, Ausführung in mehreren Taktzyklen	Einfache Instruktionen, Ausführung in einem Taktzyklen
Jede Instruktion kann auf den Speicher zugreifen	Nur Lade- und Speicherbefehle greifen auf den Speicher zu
Kein oder wenig Pipelining	Intensives Pipelining
Instrultionen werden vom Mikroprogramm interpretiert	Instruktionen werden durch festverdrahtete HW ausgeführt
Instruktionsformat variabler Länge	Instruktionen alle mit festen Länge
Kompakter Code	Code weniger kompakt
Viele Instruktionen und Adressierungsarten	Wenige Instruktionen und Adressierungsarten
Die Komplexität liegt im Mikroprogramm / mächtige Befehle	Die Komplexität liegt im Compiler
Einfacher Registersatz	Umfangreiche Registersätze (32 - 512)

Operation Modes 80386 bis Pentium

- *Real-Adress Mode*
  Stellt die Programmierumgebung des 8086 zur Verfügung u zusätzlich die Möglichkeit in den Protected oder System Managment Mode zu wechseln. Prozessor arbeitet als hochgetakteter 8086.
- *Virtual-8086 Mode*
  "quasi-operating mode". Erlaubt die Ausführung von 8086-Software in einem protected, multitasking environment
- *Protected Mode*
  Originärer Mode des Prozesors. Alle Befehle und Architektur-eigenschaften sind nutzbar.

Speicherhierarchie



MMX (= Multimedia Extensions)

MMX-Prozessor kann als Vektorrechner im Kleinen bez. werden (SIMD)
8 zusätzliche register der FDU mit einer speziell auf Multimedia ausgerichteten Arithmetik.
Sättigungsarithmetik : Bei einer Ton- und Bildverarbeitung ist eine seperate Arithmetik für Multimedia sinnvoll
Sättigung : ein kl. Wert für eine Farbintensität kann problemlos auf 0 gesetzt werden, bzw. ein Überlauf im Blauton für ein Bild ist harmlos, wenn einfach der maximale Wert eingestellt wird.

VGA (=video Graphics Array)

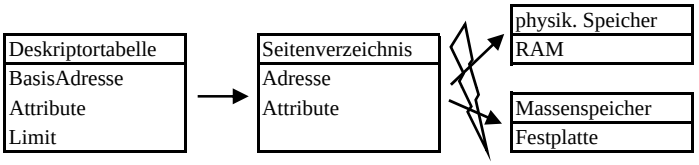
- 640 x 480 Pixel Auflösung, 16 oder 256 Farben
- auf allen Grfikkarten verfügbar
- aktiv in der Boot-Phase eines PC sowie im Reperatur- und Notbetrieb

Ansprechvarianten:

- über das BIOS der Grafikkarte (oder seltener: des PC), zB. INT 10H
  pro: einfach zu programmieren, contra: langsam
- direkter Zugriff auf den Bildspeicher (Planes) durch den Grafikcontroller
  pro: schnell, contra: aufwändige Programmierung

Virtuelle Speicherverwaltung

- Ein virtuelle Speicher bildet ein komplexes Programm ab.
- Ist eine so genannte Erweiterung des RAM.



Attribute:
- Present-Bit
- Zugriffsrechte

Interrupt:
MMU der CPU erkennt, dass Seite nicht im RAM ist.
-> BS lagert die fehlende Seite ein und aktualisiert die Seitentabllen.

Strategien für den Seitenwechsel:
1) First In First Out
2) Least Recently Used (LRU)
3) Least Frequently Used (LFU)

Cache (Kaschierer,Verdecker)

- schneller Pufferspeicher, der den nachgeordneten langsamen Speicher kaschiert / verdeckt (liegt zwischen CPU und RAM)

Assoziativspeicher: Inhaltsadressierbarer Speicher, die Adressierung erfolgt teilweise über die gesuchten Daten selbst

1) Index adressiert mittels Dekoder eine Zeile im Tag-Ram
2) Tag wird dann mit Inhalt der betreffenden Zeile im Tag-Ram verglichen: identisch (Hit), ungleich (Miss)

Schreibzugriff Cache:

1) Durchschreibeverfahren: CPU schreibt Datum gelichzeitig in Cache u Hauptspeicher. Vorteil: Daten sind immer konsistent
   Nachteil: Zugriffszeit ist durch Zugriff auf Hauptspeicher bestimmt
   -> Cache bringt keinen Vorteil beim Schreiben
2) Rückschreibeverfahren: CPU schreibt Datum zunächst nur in Cache -> schneller Zugriff. Geschriebene Daten werden durch so genanntes Dirty Bit gekennzeichnet. Rückschreiben erfolgt durch Cache controller wenn das datum in den hauptspeicher ausgelagert werden muss -> CPU arbeitet parallel weiter. Vorteil: Schneller Schreib-/Lesezugriff
   Nachteil: Datenkonsistenzproblem bei Zugriff weiterer Einheiten auf Hauptspeicher.

Rendering:

Bilderzeugung am Computer aus geometrischen Objekten.

1. Application
   - Interaktion mit Anwender
   - Kollisionsberechnung
   - Generierung der Rendering-Primitiven
   - "Culling": Zugriff auf vorgefertigte Graphikobjekte
2. Geometry
   - 2.1. Transformation [Modell- und Sichttransformation (Kameraperspektive) in 3D-Weltkoordinaten]
   - 2.2. Lighting [Beleuchtungsberechnung (Lichteffekte, Reflexionen)
   - 2.3. Projection [Transformation in 3D-Kamerakoordinaten]
   - 2.4. Clipping [Abtrennen aller Objekte außerhalb des Sichtbereichs der Kamera]
   - 2.5. Screen Mapping [Transformation in 2D-Kamerakoordinaten]
3. Rasterizer
   - Sichtbarkeitsberechnung (Z-Buffer)
   - Einrechnen der Texturen ("Tapezieren" der 2D-Polygone)
   - Setup: Umsetzen der 2D-Polygone in Pixel