

Geräteentwurf mit Mikroprozessoren 1

**Vorlesung am Institut für Elektronik der TU Graz
Dipl.-Ing. Dr. Gerhard Stöckler
SS 2003**

Vorausgesetzte Kenntnisse:

- **Grundlagen der Digitaltechnik**
- **Binäre Informationsdarstellung**

Inhalt:

Kapitel 1 Allgemeines

Kapitel 2 Halbleiterspeicher

Kapitel 3 Mikroprozessor 8086

Kapitel 4 Peripheriebausteine

Kapitel 5 Embedded Processor 80C186

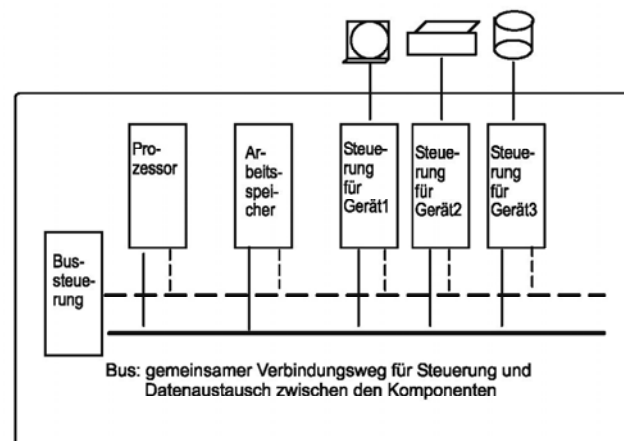
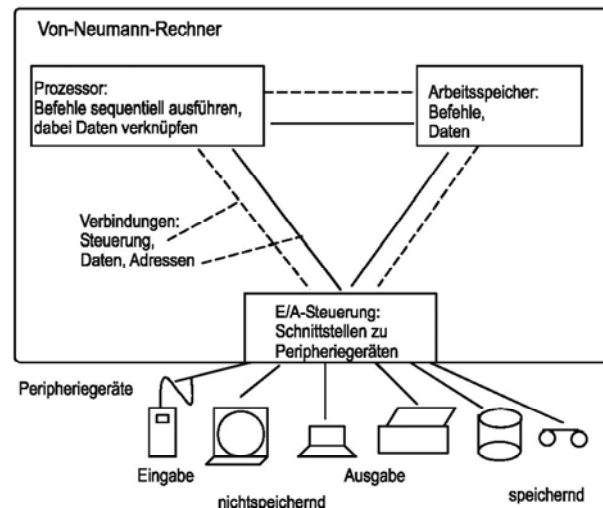
Kapitel 6 Multi-Master Systeme

Kapitel 7 Physikalische Eigenschaften von Signalleitungen

1. Allgemeines

Prinzip eines Digitalrechners:

- Zentraleinheit (CPU)
- Speicher
- Ein-/Ausgabe



Kennzeichen der v. Neumann Architektur:

- Gemeinsamer Speicheradressraum für Programm (Maschinencode) und Daten. Im Gegensatz dazu besitzt z.B. eine Harvard Architektur getrennte Adressräume (und Busse) für Programm und Daten.
- Zugriff auf Speicherinhalte über Speicheradressen. Der Speicher kann als Tabelle linear adressierter Speicherworte (meist 8 bit) angesehen werden → virtueller Speicher.
- Sequentielle Abarbeitung der Maschinenbefehle (elementare Anweisungen)

Ablauf eines Befehlszyklus':

- Einholen des Maschinenbefehls
- Dekodieren und Ausführen des Befehls

Typische Arten von Maschinenbefehlen:

- Datentransferbefehle (MOV, etc.)
- Arithmetische Befehle (ADD, SUB, MUL, DIV, etc)
- Vergleichsoperationen (CMP, etc.)
- Bool'sche Operationen (AND, OR, XOR, etc.)
- Programmverzweigungen (bedingte und unbedingte Sprünge, Unterprogrammaufrufe, etc.)

Allgemeiner Aufbau eines Maschinenbefehls:

Opcode	Adresse des 1. Operanden	Adresse des 2. Operanden	Adresse des Resultates	Adresse des nächsten Befehls
--------	--------------------------	--------------------------	------------------------	------------------------------

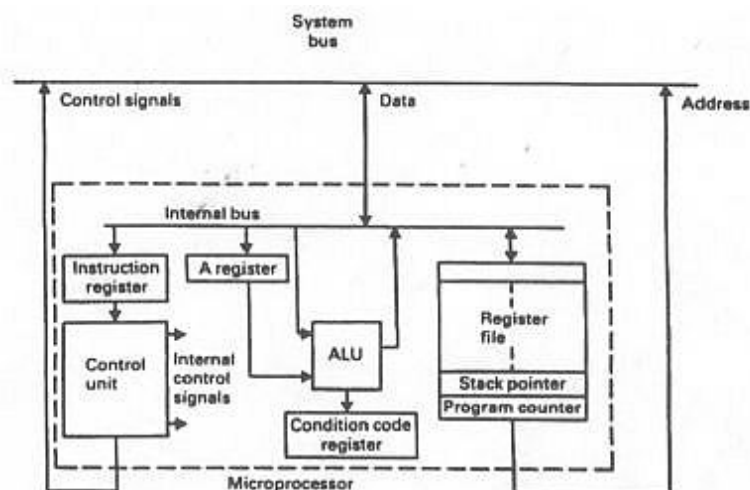
Legt man fest, dass die einzelnen Befehle in richtiger Reihenfolge im Speicher stehen, fällt die Angabe der Adresse des nächsten Befehls weg (Dreiadressmaschine). → Befehlszähler (PC, IP, etc.)

Wird weiters festgelegt, dass das Resultat an die Adresse des 1. Operanden geschrieben wird, entfällt die Angabe der Adresse des Resultats (Zweiadressmaschine).

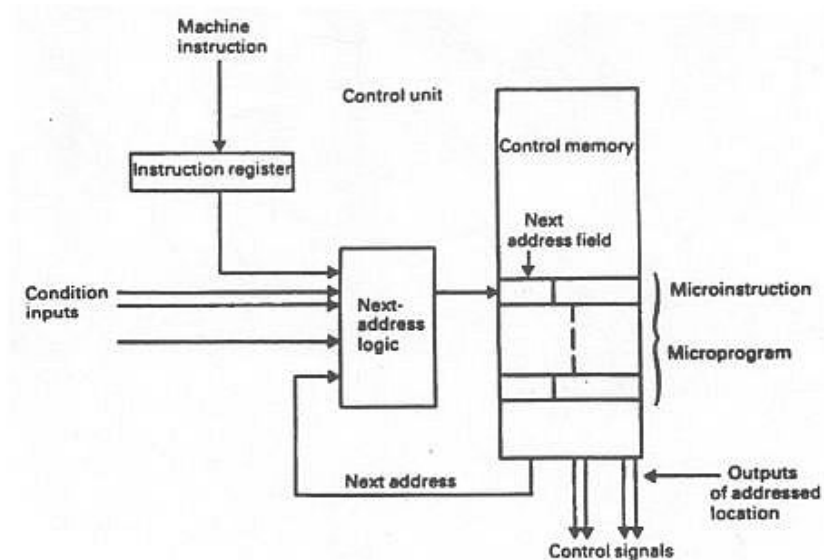
Befindet sich der 1. Operand vor Ausführung des Befehls in einem bestimmten Register (Akkumulator) entfällt auch die Angabe der Adresse dieses Operanden (Einadressmaschine). Das Resultat steht nach Befehlsausführung im Akkumulator.

Bei einer sog. Nulladressmaschine ist überhaupt keine Adressangabe nötig. Die Operanden befinden sich an der Spitze eines Stapelspeichers.

Zentraleinheit (CPU, Mikroprozessor)



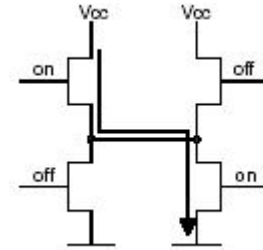
- Rechenwerk(ALU): Im einfachsten Fall Binäraddierer und Logik für Bool'sche und Verschiebeoperationen.
- Steuerwerk: Erzeugt alle notwendigen internen und externen Steuersignale. Zwei unterschiedliche Arten der Realisierung:
 - Micro programmed design: Die zeitliche Verlauf der Steuersignale wird durch die Abfolge sog. Mikroinstruktionen bestimmt. Das Bitmuster innerhalb einer Mikroinstruktion stellt den logischen Zustand der Steuersignale zu einem bestimmten Zeitpunkt dar. Wird bei CISC – Prozessoren verwendet.
 - Random logic design: Festverdrahtete sequentielle Logikschaltung. Wird bei RISC – Prozessoren verwendet.



- Registersatz: Universalregister zur Speicherung von Operanden und spezielle Register:
 - Befehlszähler (program counter PC, instruction pointer IP): Enthält immer die Adresse des als nächstes einzuholenden Programm Code; wird automatisch inkrementiert.
 - Stapelzeiger (stack pointer SP): Zeigt immer auf die Spitze des Stapelspeichers (frei wählbarer Bereich des Arbeitsspeichers) → Speicherzugriff ohne Adressangabe; bei Verzweigung zu Unterprogrammen oder Interrupt Service Routinen automatische Abspeicherung der Rücksprungadresse.
 - Programmstatuswort (PSW, flag register, condition code register): Die einzelnen Bits geben Auskunft über das Ergebnis einer ausgeführten arithmetischen oder logischen Operation (z.B. zeroflag, carryflag, etc.).

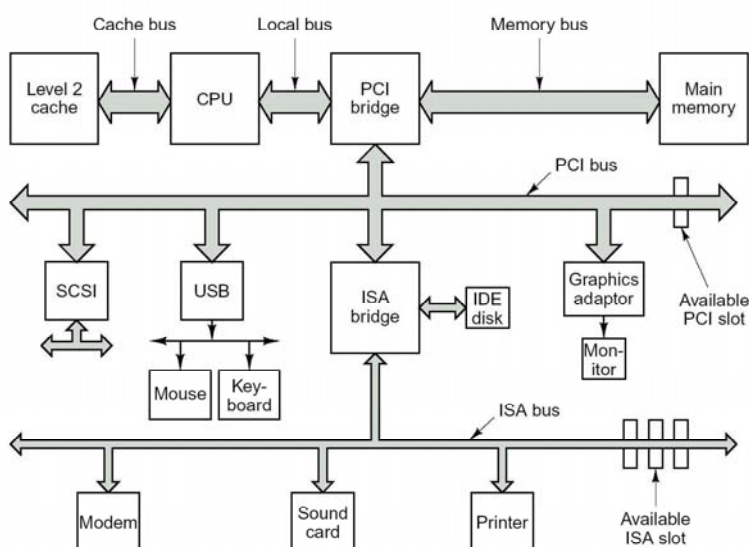
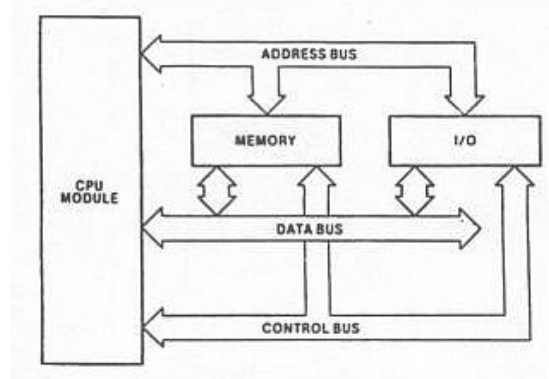
Bussystem

Schar elektrischer Leitungen zur Kommunikation zwischen den einzelnen Bausteinen bzw. Baugruppen. Alle Teilnehmer sind parallel angeschlossen und es muss dafür gesorgt werden, dass zu einem bestimmten Zeitpunkt immer nur ein Teilnehmer den Zustand am Bus bestimmt → Buskonflikt.



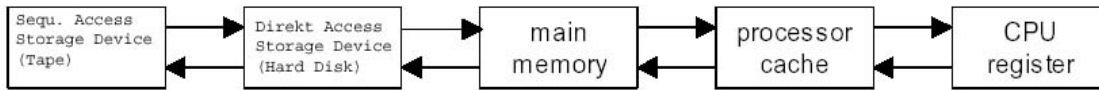
Signalpegel: TTL, LVTTTL, GTL, etc.

- Datenbus (4, 8, 16, 32, 64 Bit)
- Adressbus (16, 20, 24, 32, 36 Bit)
- Steuerbus



Architecture of a typical Pentium II system.

Speicherhierarchie:

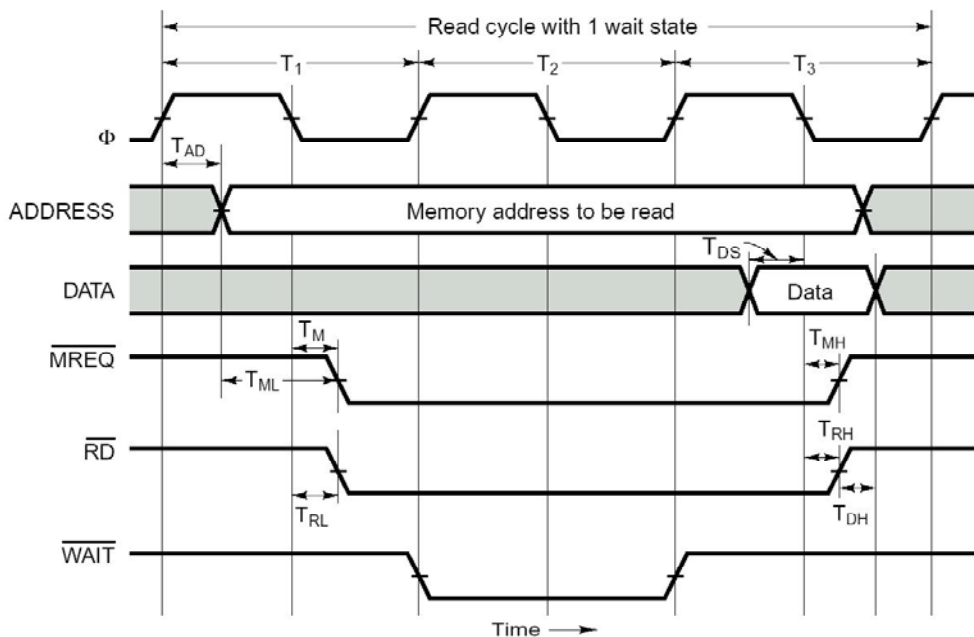


Bustransfer:

Die Bussteuereinheit (CPU, DMA-Controller, MMU) bestimmt:

- Adresse
- Transferrichtung (Schreiben: CPU → Speicher/IO, Lesen: Speicher/IO → CPU)
- Transferzeitpunkt

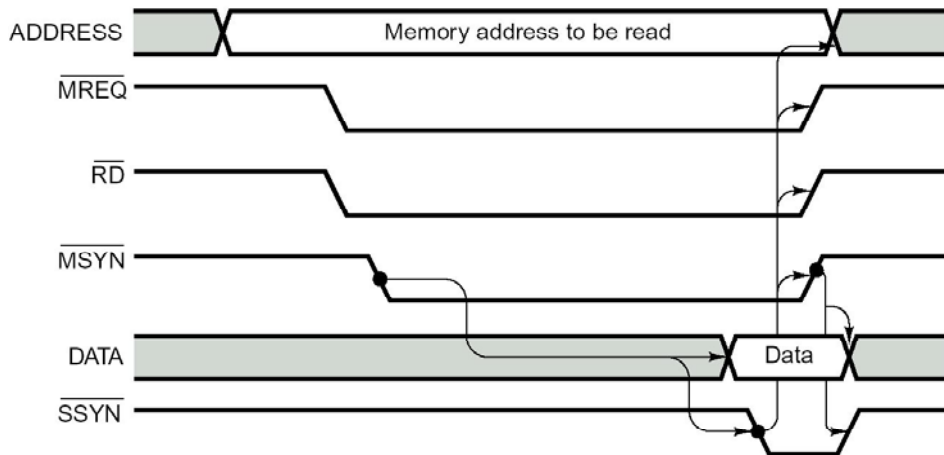
Synchroner Bustransfer:



Der Transfer ist an einen festen Takt gebunden. Der angesprochene Teilnehmer muss das Datum zu einem festen Zeitpunkt bereitstellen (Lesezyklus) oder übernehmen (Schreibzyklus) → Taktfrequenz richtet sich nach dem langsamsten Teilnehmer.

Abhilfe: Signal $\text{WAIT}\#$ ($\# \text{READY}$) → durch Einfügen von *wait states* wird die Zykluslänge individuell angepasst. „Wait state generator“ innerhalb der Bussteuereinheit oder extern.

Asynchroner Bustransfer:



Buszyklus beginnt wieder mit Aussenden der Adresse und Festlegung der Transferrichtung. Signal MSYN# (#AS) der Bussteuereinheit startet den Transfer, die angesprochene Einheit quittiert mit SSYN# (DTACK#) wenn sie das Datum bereitgestellt bzw. übernommen hat.

Burst-transfer: Synchroner Bustransfer, bei dem nur am Beginn des Zyklus' eine Adresse gesendet und dann automatisch auf die nachfolgenden Adressen zugegriffen wird. → schneller. Die Speicher müssen dafür geeignet sein.

