

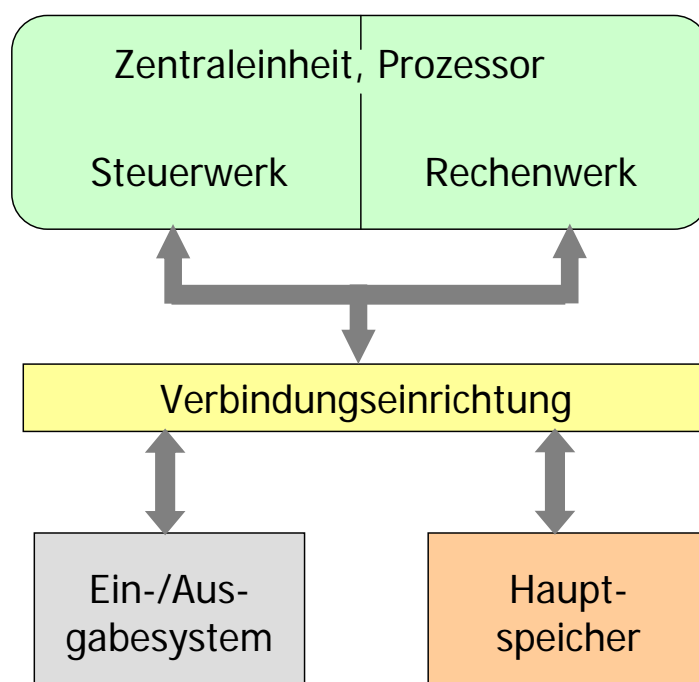
Kapitel 3

Ein grundlegendes Rechnermodell

- Organisationsprinzip des von Neumann Rechners
- Aufbau eines einfachen Mikroprozessors
 - Steuerwerk (Leitwerk),
 - Rechenwerk
 - Speicherwerk
 - Ein-Ausgabewerk
 - Verbindungsstrukturen
- Maschinenbefehlszyklus



3.1 Organisationsprinzip des von Neumann Rechners



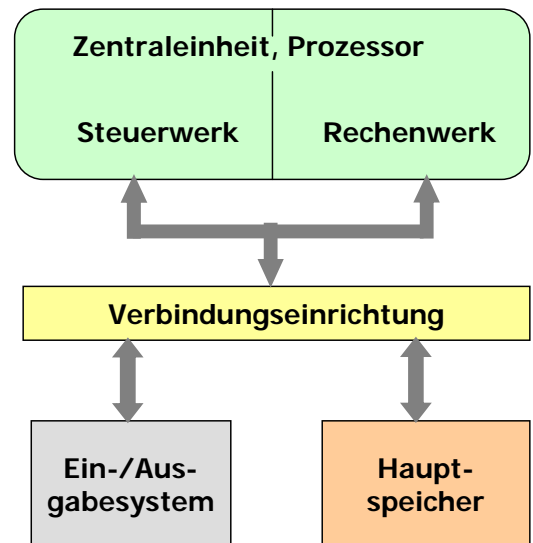
Komponenten des von Neumann Rechners

□ Zentraleinheit

(central processing unit, CPU, Prozessor)

Verarbeitet Daten gemäß eines Programms. Sie besteht aus Leitwerk und Rechenwerk:

- **Leitwerk** (Steuerwerk, control unit, CU)
 - Holt die Befehle eines Programms aus dem Speicher
 - entschlüsselt sie und
 - steuert ihre Ausführung in der verlangten Reihenfolge durch Steuer- und Synchronisier-Signale.

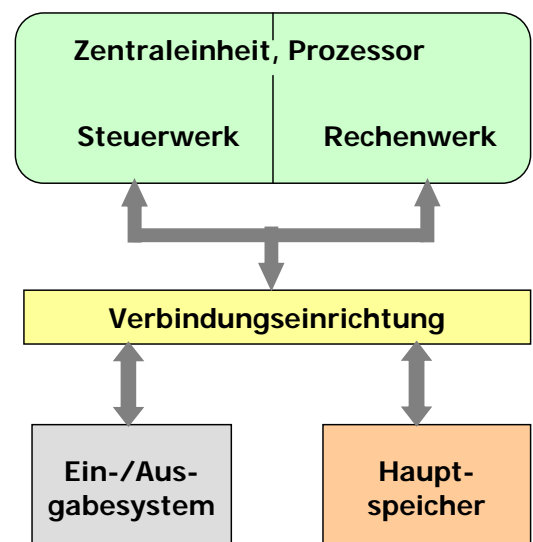


Komponenten des von Neumann Rechners

□ Zentraleinheit

Verarbeitet Daten gemäß eines Programms. Sie besteht aus Leitwerk und Rechenwerk:

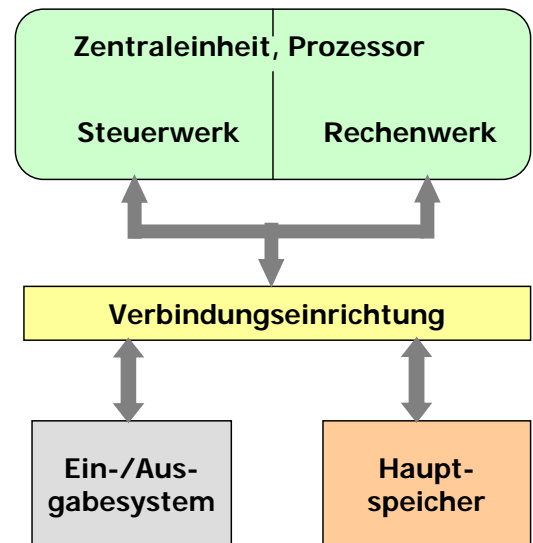
- **Rechenwerk (Operationswerk, Ausführungseinheit, ALU)**
 - Führt arithmetisch/logische Operationen aus.
 - Wird durch Steuersignale des Leitwerks beeinflusst und
 - liefert seinerseits Meldesignale an das Leitwerk zurück.



Komponenten des von Neumann Rechners

□ Hauptspeicher

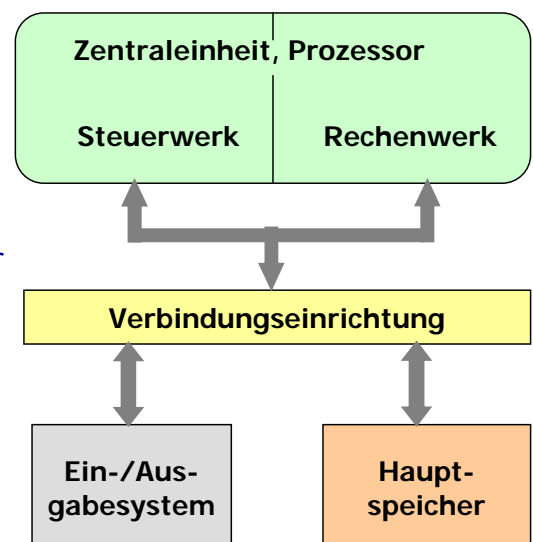
- Jede Speicherzelle ist eindeutig durch ihre Nummer (Adresse) identifizierbar.
- Dort werden Programme und Daten aufbewahrt (von-Neumann-Konzept).
- Den einzelnen Speicherzellen ist nicht anzusehen, welchen Typ von Information sie enthält
- Alternativ: **Harvard-Architektur** mit getrenntem Programm- und Datenspeicher.
- Inhalt nach Abschaltung des Rechners flüchtig.



Komponenten des von Neumann Rechners

□ Verbingungsstruktur (BUS)

- **Adreßleitungen:** Leitungen, auf denen die Adressinformation transportiert wird (unidirektional).
- **Datenleitungen:** Transportieren Daten und Befehle von/zum Prozessor (bidirektional).
- **Steuerleitungen:** Geben Steuerinformationen von/zum Prozessor (uni- oder bidirektional).



□ Bus (Sammelschiene)

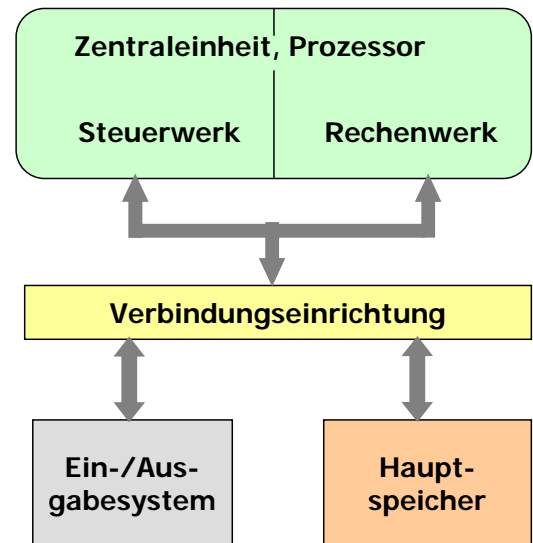
Systembus = Adreßbus + Datenbus + Steuerbus



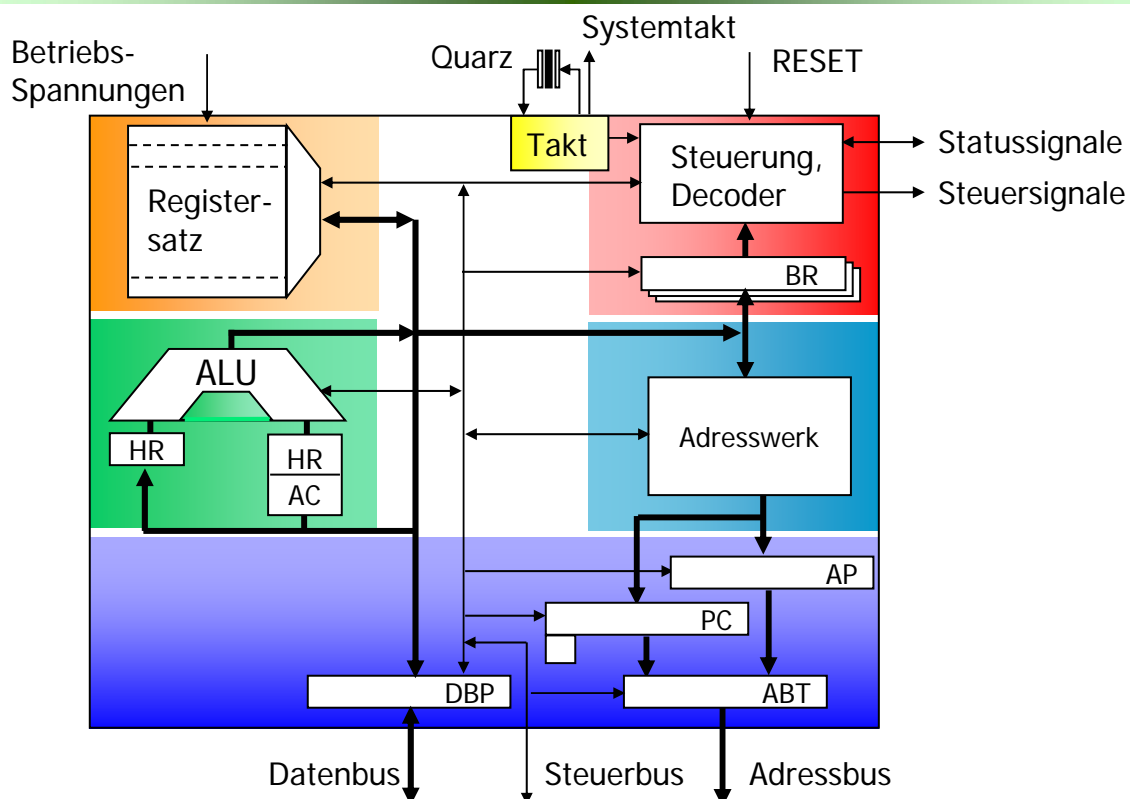
Komponenten des von Neumann Rechners

□ Ein-/Ausgabesystem (Peripheriegeräte)

- Geräte zur Eingabe von Daten und Programmen und zur Ausgabe der verarbeiteten Daten (Bildschirme, Drucker, Terminals, ...)
- Diese Geräte sind über Ein-/Ausgabe-Schnittstellen mit dem Rechner verbunden.
- Die Verbindung der Schnittstellen mit dem Prozessor (und zu den Peripheriegeräten) geschieht durch Adreß-, Daten- und Steuerleitungen.



3.2 Aufbau eines einfachen μP



Aufbau eines einfachen μ P

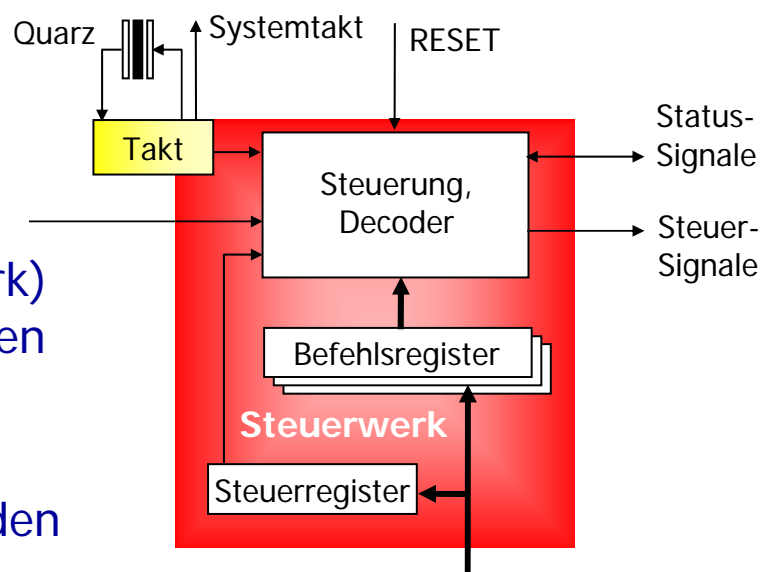
- ❑ **Steuerwerk**
- ❑ **Rechenwerk**
- ❑ **Registersatz**
- ❑ **Adresswerk**
- ❑ **Systembusschnittstelle**
- ❑ **Interne Busse**



3.2.1 Steuerwerk

Steuert die Systemkomponenten

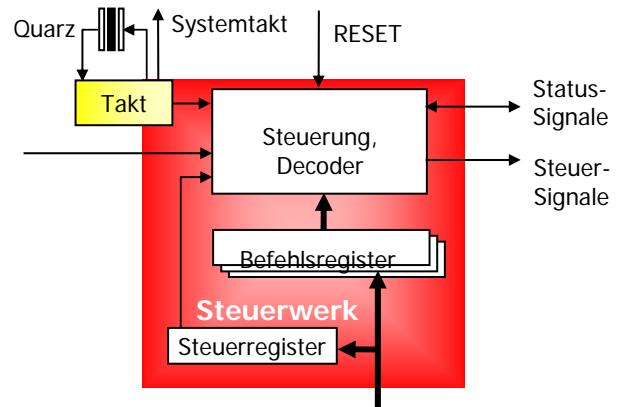
- **Befehlsregister**
enthält den gerade ausgeführten Befehl
- **Dekoder** (mikro-programmiertes Schaltwerk)
wird von den Statussignalen beeinflusst und erzeugt die Steuersignale
- **Taktgenerator** erzeugt den vom externen Quarz festgelegten Systemtakt



3.2.1 Steuerwerk

Synchrones Schaltwerk

Meist liegt ein sog. **dynamisches Schaltwerk** vor, d. h. die Zustandsinformation ist nicht in Flipflops, sondern in Kondensatoren gespeichert



→ Mindesttaktfrequenz ist erforderlich

Unterhalb dieser Taktfrequenz gehen die Inhalte durch Leckströme bereits vor dem nächsten Taktzyklus verloren.

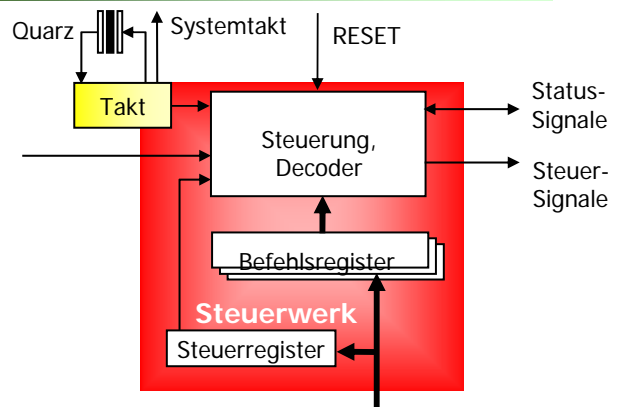
Taktgenerator ist bei modernen Mikroprozessoren on Chip (mit externem Quarz verbunden)



Taktgenerator

Aufgaben des Taktgenerators

- Taktfrequenz herstellen
- Erzeugung eines mit dem Prozessortakt synchronisierten Rücksetzsignals



Beim Rücksetzen durchläuft das Steuerwerk eine Initialisierungsroutine

Diese wird bei vielen Mikroprozessoren ausgeführt, während das Rücksetzsignal aktiv ist

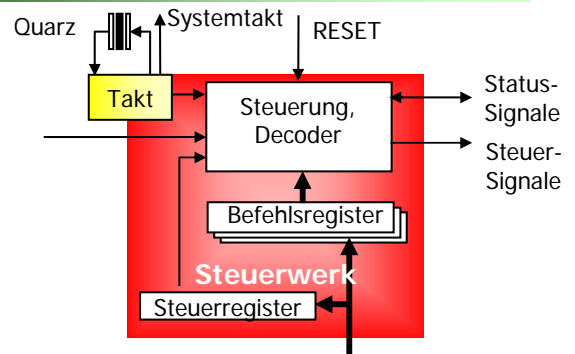
Deshalb muss das Rücksetzsignal genauen zeitlichen Spezifikationen genügen.



3.2.1 Steuerwerk

Mikroprogrammsteuerwerk

Im Festwertspeicher liegt für jeden Befehl ein Mikroprogramm



Mikroprogramm \equiv Folge von Mikrobefehlen

Aufbau eines Mikrobefehls:

Folge-Adresse	Register	Rechenwerk		Systembus Schnittstelle	Adresswerk	externe Steuersignale
		ALU	AC,HR			

Einzelne Bits eines Mikrobefehls \equiv Mikrooperationen \equiv Auswahl- und Freigabesignale für die benötigten Komponenten



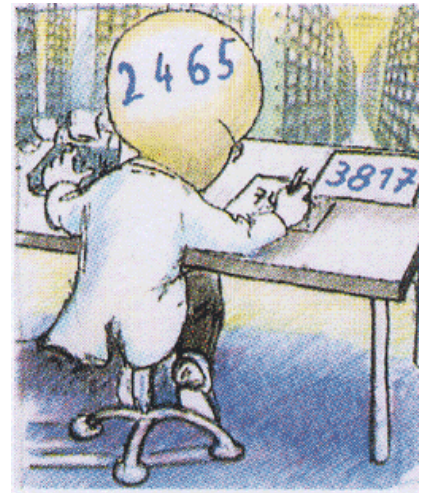
Phasen der Befehlsausführung



Holphase



Decodierphase



Ausführungsphase



Phasen der Befehlsausführung

Holphase:

der nächste Befehl in das Befehlsregister laden

Decodierphase:

der Befehlsdecoder ermittelt die Startadresse des Mikroprogramms, welches den Befehl ausführt

Ausführungsphase:

das Mikroprogramm steuert die Befehlsausführung, indem es entsprechende Signalfolgen an die anderen Prozessorkomponenten übermittelt und Meldesignale auswertet



3.2.1 Steuerwerk

Das Befehlsregister besteht aus mehreren Registern:

Gründe:

- unterschiedlich lange Befehlsformate:
verschiedene Befehle sind unterschiedlich lang
(1-Wort-Befehle, 2-Wort-Befehle, 3-Wort-Befehle, ...)
- Vorabladen von Befehlen (*Opcode-Prefetching*):
zur Steigerung der Verarbeitungsgeschwindigkeit werden bereits mehrere folgende Befehle in das Befehlsregister geladen, während der aktuelle Befehl gerade dekodiert wird

Opcode prefetch queue, Warteschlange, Pipelining



3.2.1 Steuerwerk

- Für jeden Befehl liegt ein Mikroprogramm im Festwertspeicher des Steuerwerks vor
- Mikroprogramme können vom Benutzer nicht verändert werden
 - ➔ Das Steuerwerk ist mikroprogrammiert

Andere Variante:

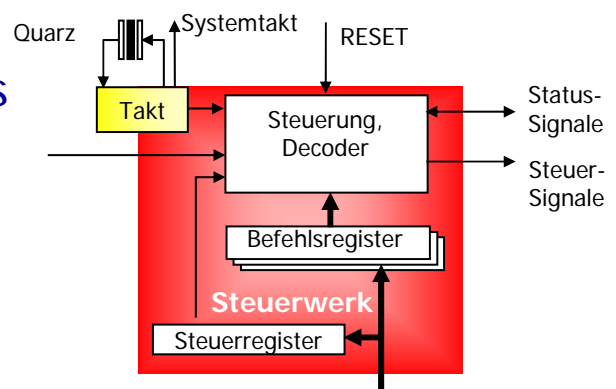
Steuerwerk als festverdrahtetes Schaltwerk (RISC-Prozessoren)



Das Steuerregister

Mit Hilfe des **Steuerregisters** kann die aktuelle Arbeitsweise des Steuerwerks beeinflusst werden

Die Bedeutung der Bits des Steuerregisters hängen vom jeweiligen Prozessor ab



Beispiele für die Bedeutung von Bits des Steuerregisters:

- **Interrupt enable Bit**
bestimmt, ob auf eine Unterbrechungsanforderung am INT-Eingang reagiert wird

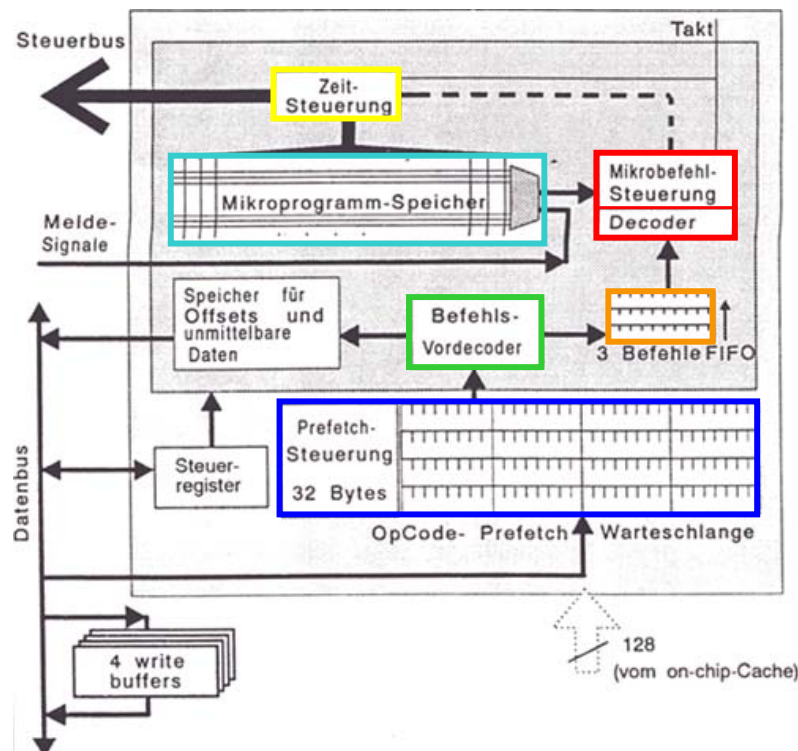


Das Steuerregister

- **User/System Bit**
bestimmt ob der Prozessor im User-Modus (nur beschränkter Teil des Befehlsvorrats nutzbar) oder im Systemmodus (alle Befehle verfügbar, i. A. für das Betriebssystem reserviert) arbeitet
- **Trace Bit**
erlaubt Befehlsabarbeitung im Einzelschritt (Single Step Mode), d. h. nach jeder Befehlsausführung wird eine Unterbrechungsroutine gestartet → Debugging
- **Decimal Bit**
entscheidet, ob Dual oder BCD gerechnet wird



Steuerwerk des Intel 80486



Steuerwerk des Intel 80486

Befehlsregister-Block: Prefetch-Queue

- FIFO-Speicher mit 32 Byte
- Prefetch Steuerung sorgt für weitestmögliche Füllung
- Füllung kann meist aus dem on-chip Cache über einen 128 Bit breiten Datenpfad erfolgen
- Muss die Füllung jedoch aus dem Arbeitsspeicher erfolgen (Cache Miss), haben diese Zugriffe höchste Priorität
- Eventuelle gleichzeitig auf den Bus auszugebende Daten werden in Schreibpuffern (write buffers) zwischengespeichert



Steuerwerk des Intel 80486

Befehls-Vordecoder:

- Aus der Prefetch-Queue gelangen die Befehle in den Befehls-Vordecoder
- Vorbereitung der Befehle
- direkt im Befehl angegebene Operanden (unmittelbare Daten) sowie Adressdistanzen (Offsets) werden abgezweigt und separat gespeichert



Steuerwerk des Intel 80486

Befehls-FIFO:

- vordekodierte Befehle gelangen in den Befehls-FIFO
- Platz für 3 Befehle

Befehls-Decoder:

- entnimmt den obersten vordekodierten Befehl aus dem Befehls-FIFO
- ermittelt die Startadresse des zugehörigen Mikroprogramms



Steuerwerk des Intel 80486

Mikrobefehls-Steuerung, Mikrobefehls-Speicher:

synchrones mikroprogrammiertes Schaltwerk, erzeugt die Steuersignale, interpretiert die Meldesignale

Zeit-Steuerung:

synchronisiert die erzeugten Steuersignale mit dem Systemtakt



Steuerwerk: Zusammenfassung

- ❑ Das Steuerwerk interpretiert die Maschinenbefehle und setzt sie unter Berücksichtigung der Statusinformation in Steuerkommandos für andere Komponenten um.
- ❑ Das Befehlsregister enthält die Speicheradresse des als nächstes auszuführenden Maschinenbefehls, sofern der vergangene Befehl kein Sprungbefehl ist.
- ❑ Das Steuerregister beeinflusst die aktuelle Arbeitsweise des Steuerwerks

