

Zusammenfassung: FPM, EDO

- ❑ Sie arbeiten asynchrone zum Systembus
- ❑ Für eine Datenübertragung ist ein Handshaking-Verfahren notwendig. Ein Lesevorgang läuft wie folgt:
 - Prozessor signalisiert der Speichersteuerung, dass eine Adresse anliegt.
 - Wenn die Daten am Ausgang des DRAMs bereitliegen, teilt die Speichersteuerung dem Prozessor dies mit (BRDY-Signal). Erst dann liest der Prozessor die Daten ein.
 - Dazwischen ist die CPU im Leerlauf und führt Wartezyklen aus.
- ❑ Varianten von EDO-DRAM (BEDO-DRAM) können die Daten ohne Wartezyklen liefern, aber nur bis zu einem Bustakt von 66 MHz.

SDRAM

SDRAM-Technologie hat sich (durch intensive Unterstützung von Intel) schnell durchgesetzt und beherrscht heute den Speichermarkt.

- Alle Ein- und Ausgangssignale sind synchron zum Systemtakt.
- Prozessor, Chipsatz und Speicher kommunizieren über ein Bussystem, das synchron mit der gleichen Frequenz getaktet ist.

Intern sind SDRAMs aus zwei unabhängigen Speicherbänken aufgebaut (auch bis zu 4 Speicherbänke).

Nach dem Anlegen der Zeilen- und Spaltenadresse, generiert die Speichersteuerung die nachfolgenden Adressen und führt einen alternierenden und überlappenden Zugriff auf die beiden Speicherbänke selbstständig aus

SDRAM

SDRAMs (Synchrone Dynamische RAMs):

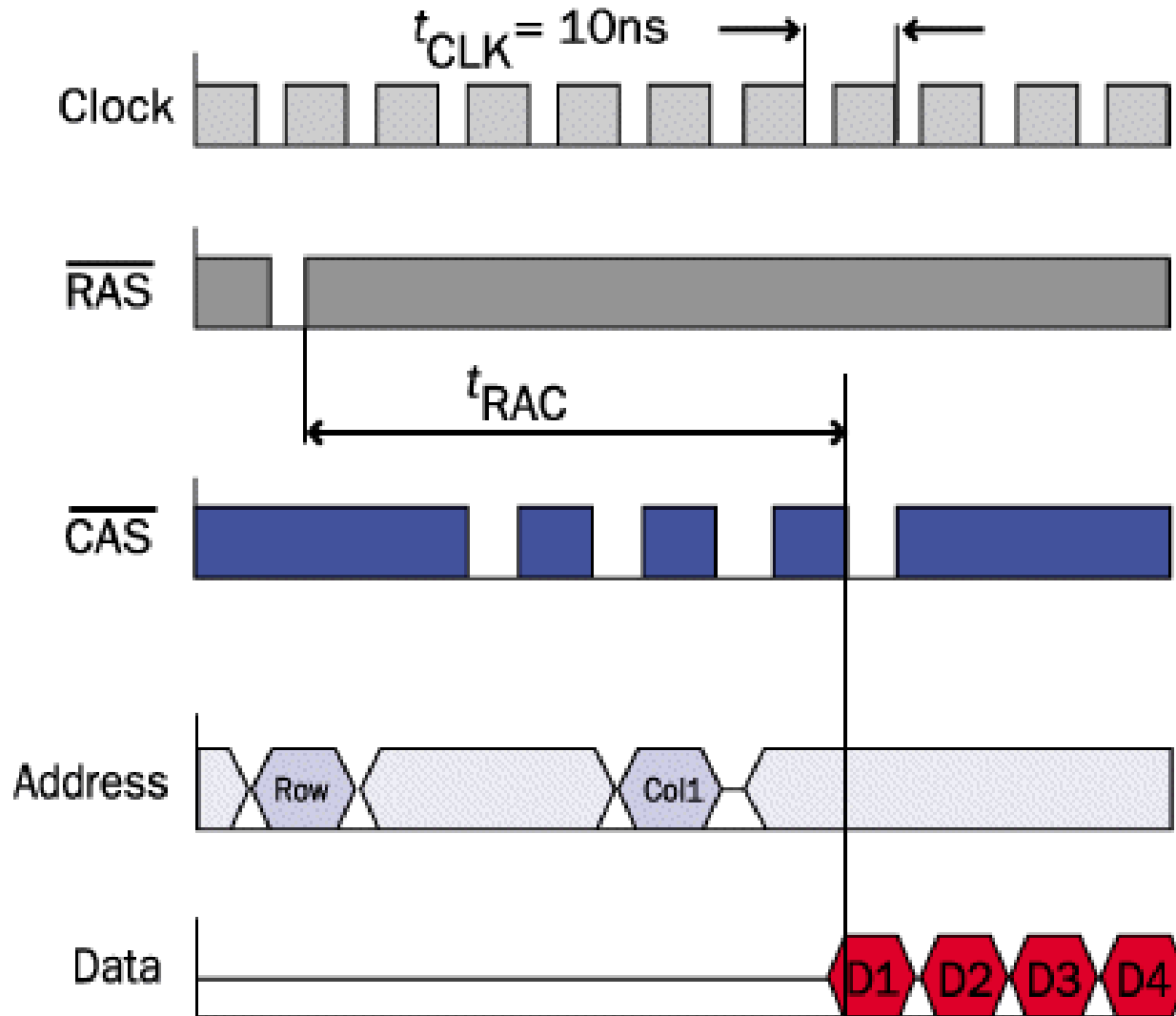
SDRAMs arbeiten synchron mit dem Systemtakt

Intern bestehen SDRAM-Bausteine aus zwei unabhängigen Speicherbänken. Eine Bank kann vorgeladen werden (Precharge), während die andere Bank einen Lese- oder Schreibzugriff durchführt.

Aktuelle SDRAMs besitzen je nach Kapazität sogar vier interne Speicherbänke.

SDRAMs können Busgeschwindigkeiten von bis zu 100 MHz bearbeiten.

Timinig-Diagramm eines SDRAM



100 MHz-SDRAM
können alle 10 ns
Daten liefern.

Datentransferrate:

- 500 Mbyte/sec (66 MHz)
- 800 Mbyte/s (PC100-Module).
- 1,06 Gbyte/s (PC133-Module)
- Praxis:
ca. 12 %
Leistungssteigerung bei
gleichem Prozessortakt
und 100 MHz statt 66
MHz

DDRAM

Nächste Stufe der SDRAM-Entwicklung (SDRAM II).

Bestehen intern aus vier unabhängigen Speicherbänken, die parallel „Instruktionen“ bearbeiten können.

Prinzip der DDR-DRAMs:

Erweiterung der Bandbreite durch Nutzung beider Taktflanken. Daten werden bei steigender und fallender Taktflanke übertragen → doppelter Datendurchsatz

Laufzeitverzögerungen sind sehr kritisch, deshalb wird zur Synchronisation nicht nur der Systemtakt, sondern auch ein bidirektionales Strobe-Signal (DQS) benutzt.

DDR-SDRAM

□ **DDR-SDRAM** (*Double-Data-Rate-SDRAM*)

Die DDR-SDRAMs entsprechen in Bauform und Funktionsweise den "normalen" SDRAM-Modulen, jedoch werden im Gegensatz zu diesen die Speicherzellen zweimal pro Takt ausgelesen bzw. geschrieben. Dadurch erreichen die DDR-SDRAM Module den doppelten Datendurchsatz.

□ **SLDRAM** (*Sync Link RDRAM*)

Weiterentwicklung der SDRAM Technologie, die höhere Busfrequenzen erlaubt und damit eine höhere Leistung ermöglicht.

RDRAM/ Concurrent - RDRAM / Direct RDRAM

□ **RDRAM/ Concurrent - RDRAM / Direct RDRAM**

Die RDRAM- Technologie gibt es seit 1995 und hat sich in vielen Workstations und Spielekonsolen bewährt

Der spezielle Bus (Rambus Channel) setzt jedoch ein entsprechendes Design der Hauptplatine voraus

Die Varianten **Concurrent RDRAM** und **Direct RDRAM** unterscheiden sich hinsichtlich der maximalen Taktrate und der eingesetzten Protokolle

6.6 Organisation des Hauptspeichers

Hauptsspeicher:

- ❑ lineare Liste von Speicherworten
- ❑ Aufgebaut aus Speicherbausteinen
- ❑ Zugriffszeit hängt allein von der Art der verwendeten Speicherbausteine ab
- ❑ Die Breite des Arbeitsspeichers entspricht i. A. der Breite des Datenbus (8, 16, 32, 64 Bit). Dies entspricht der maximalen Informationsmenge, auf die in einem Buszyklus zugegriffen werden kann.

6.6 Organisation des Arbeitsspeichers

Bei Prozessoren mit einer Datenbusbreite > 8 Bit kann meist immer noch auf einzelne Bytes zugegriffen werden

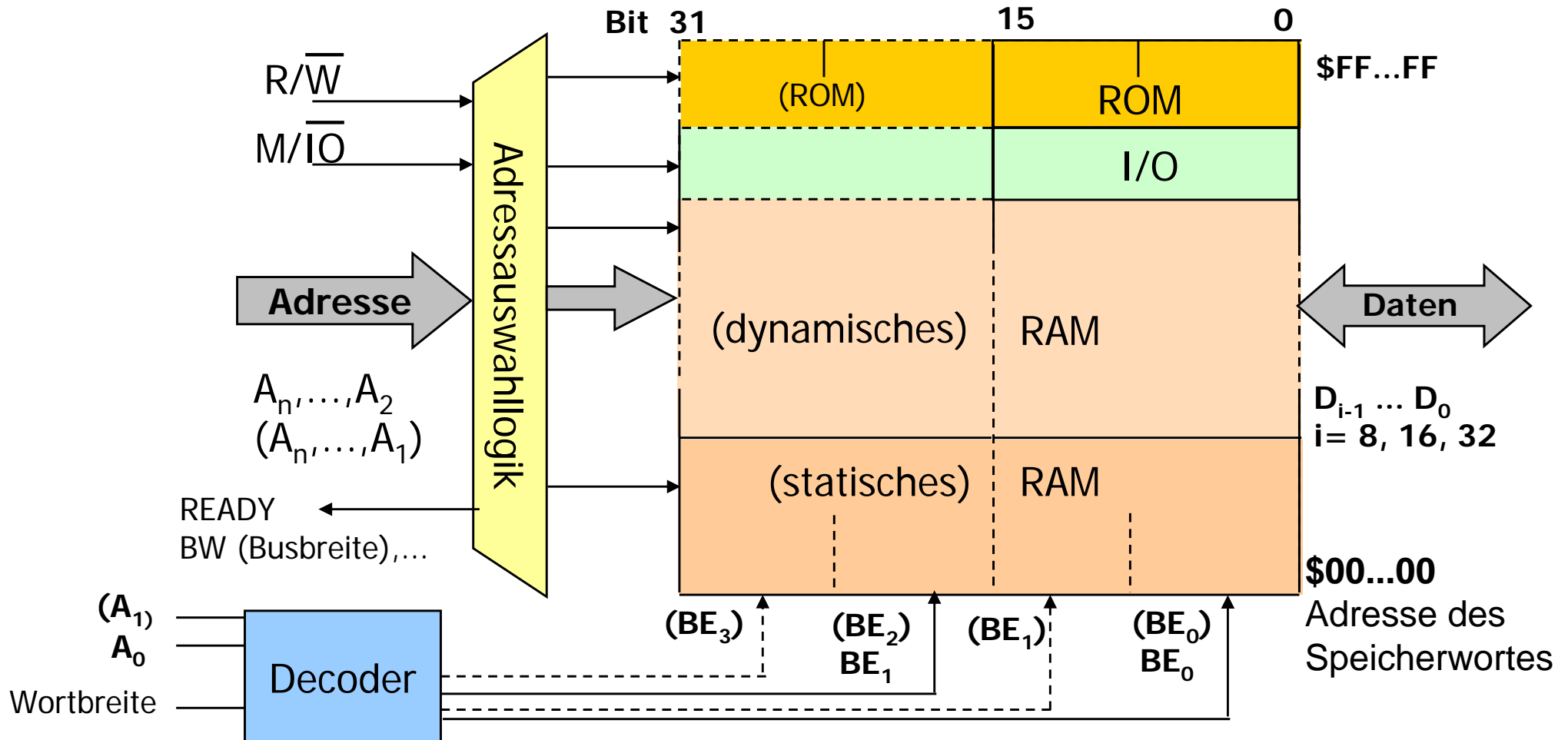
→ Speicherkapazität wird auch bei breiteren Organisationsformen in Bytes angegeben

Die maximale Kapazität des Hauptspeichers ist durch die Breite des Adressbusses gegeben

- 8-Bit Prozessoren mit 16-Bit Adressbus: 64 kByte
- 16-Bit Prozessoren mit 24-Bit Adressbus: 16 Mbyte
- 32-Bit Prozessoren mit 32-Bit Adressbus: 4 GByte

Speicher-Belegungsplan (*memory map*)

Gibt an, welche Speicherbausteine für welche Bereiche des Hauptspeichers verwendet wurden



Speicher-Belegungsplan (*memory map*)

Im Beispiel:

obere Adressen: ROM, z.B. für nicht-flüchtige Teile des Betriebssystems (Bootstrap, BIOS)

dann: IO-Bereich (Prozessor mit memory-mapped IO)

Rest: RAM

meist dynamische RAM-Bausteine, da diese große Kapazität besitzen und billig sind

Nachteil: Sie sind auch langsam

Aus diesem Grund werden manchmal in kleinen Speicherbereichen auch statische RAMs eingesetzt, auf die ohne Wartezyklen zugegriffen werden kann

Speicher-Belegungsplan (*memory map*)

Die Speicherbreite kann über den Speicherbereich variieren

Die Byte- und Breiten-Auswahl eines Speicherzugriffs erfolgt in der Regel über die niederwertigsten Adressbits (z.B. A_0 , A_1 bei 32 Bit Speicherbreite) sowie spezieller Wortbreite-Signale vom Mikroprozessor

Speicherbelegungspläne werden häufiger noch feiner untergliedert, indem z. B. die genaue Lage von Betriebssystemtabellen im ROM-Bereich oder von Geräten im IO-Bereich angegeben wird.

Adressauswahl

- Der höherwertige Teil der Adresse dient über eine Adressauswahllogik zur Speicherbaustein-Auswahl

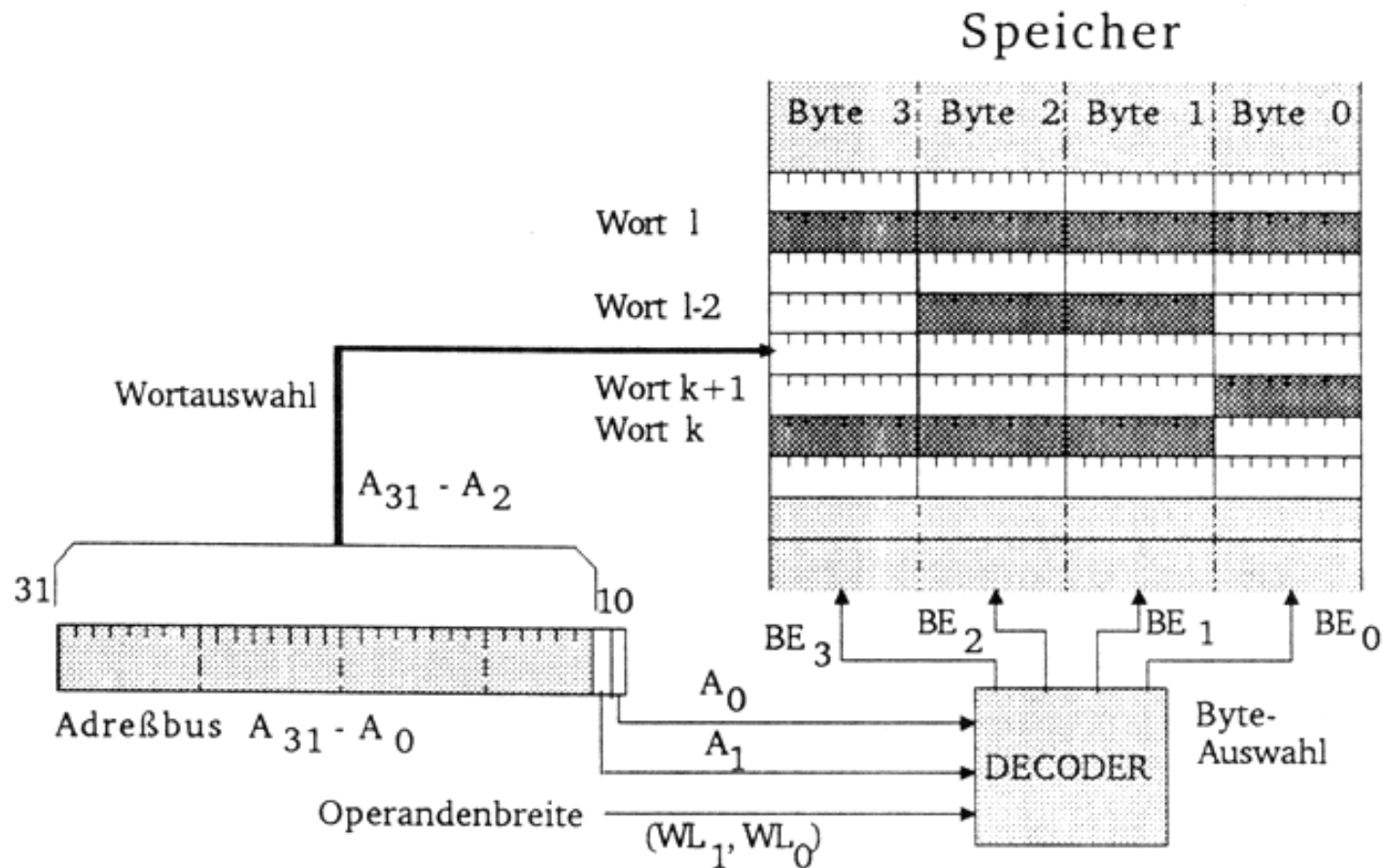
→ \overline{CS} -Signale der Speicherbausteine

Hier wird i. A. auch die Zugriffsrichtung, Speicher- bzw. I/O-Zugriffe sowie eventuelle Wartezyklen ermittelt

- Der mittlere Teil der Adresse geht direkt an die Adresseingänge der Bausteine
- Der niederwertige Teil der Adresse dient zusammen mit Wortbreiten-Signalen zur Wortauswahl innerhalb der Speicherbreite

Adressauswahl

Auswahl eines Bytes, Worts oder Doppelworts in einem Speicherwort



Adressauswahl

zum Beispiel:

oberes Wort: 32-Bit Wort, aligned

$A_1A_0 = 00$ [Startbyte 0], $WL_1WL_0 = 00$ [Wortbreite 32 Bit]

mittleres Wort: 16-Bit Wort, unaligned

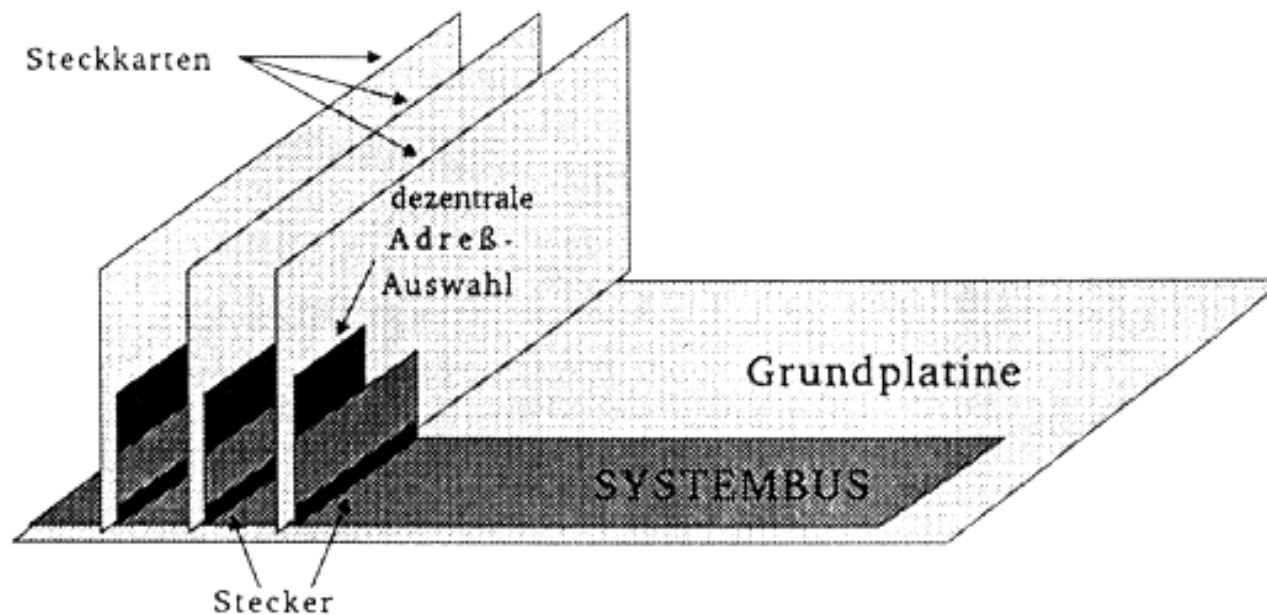
$A_1A_0 = 01$ [Startbyte 1], $WL_1WL_0 = 01$ [Wortbreite 16 Bit]

unteres Wort: 32-Bit Wort, unaligned (2 Speicherzugriffe nötig)

$A_1A_0 = 01$ [Startbyte 1], $WL_1WL_0 = 00$ [Wortbreite 32 Bit]

Modularer Speicheraufbau

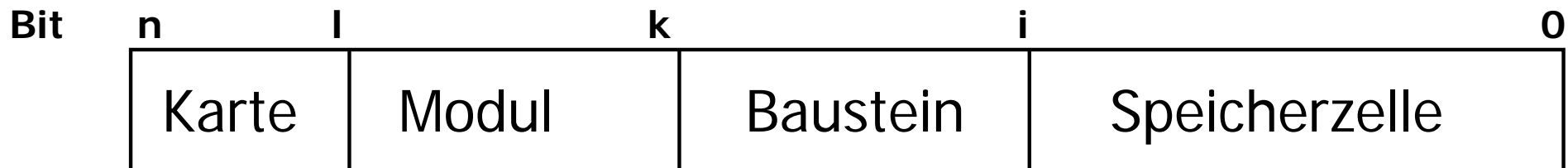
Arbeitsspeicher wird oft auf mehrere Steckkarten verteilt, die über eine Grundplatine mit dem Systembus verbunden sind → Erweiterbarkeit



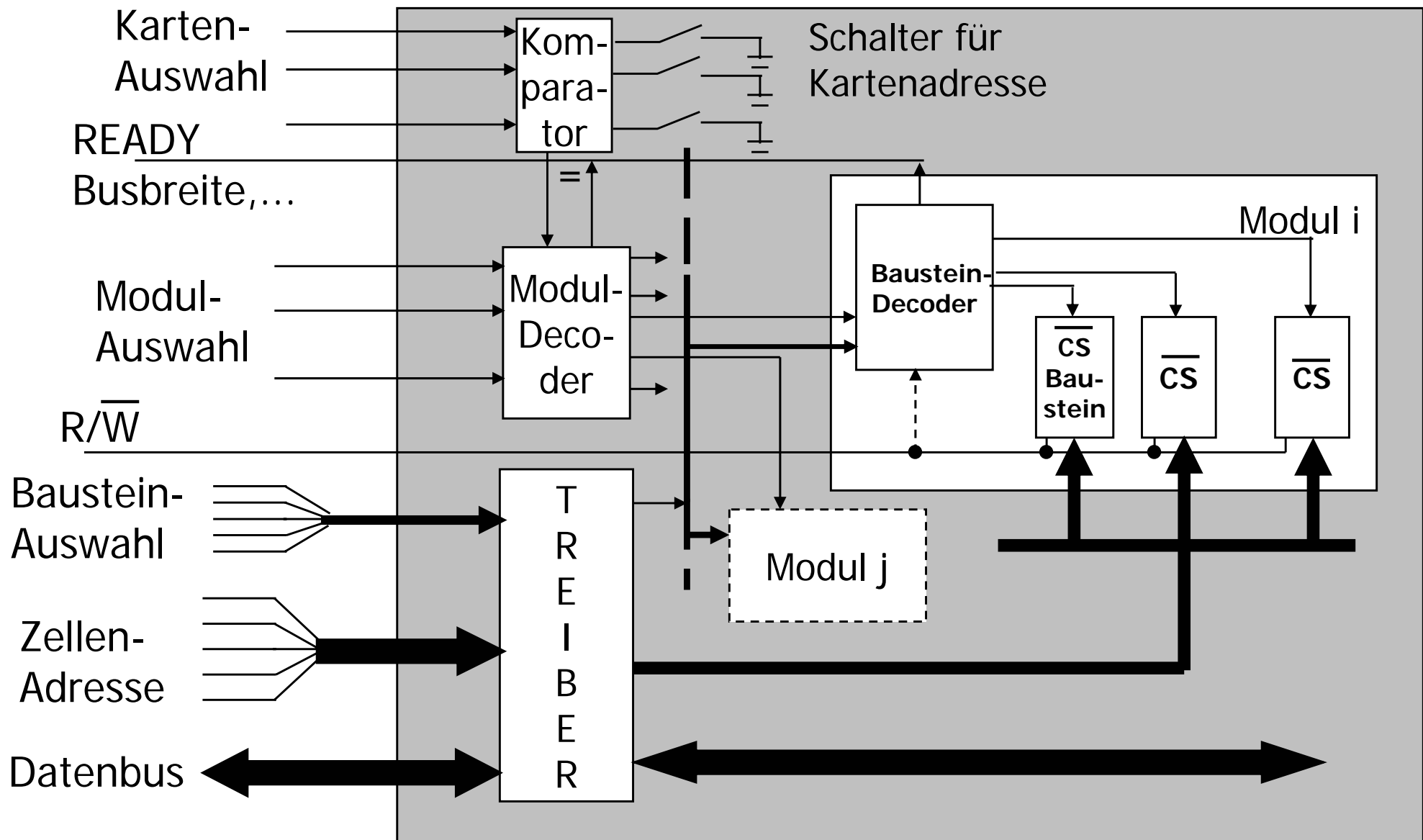
→ anstelle einer zentralen Adressauswahllogik ist eine dezentrale Adressauswahllogik erforderlich

Modularer Speicheraufbau

Die Unterteilung der Bits einer Speicheradresse zur Auswahl einer Speicherzelle ergibt sich dann wie folgt:



Typischer Aufbau einer Steckkarte



Typischer Aufbau einer Steckkarte

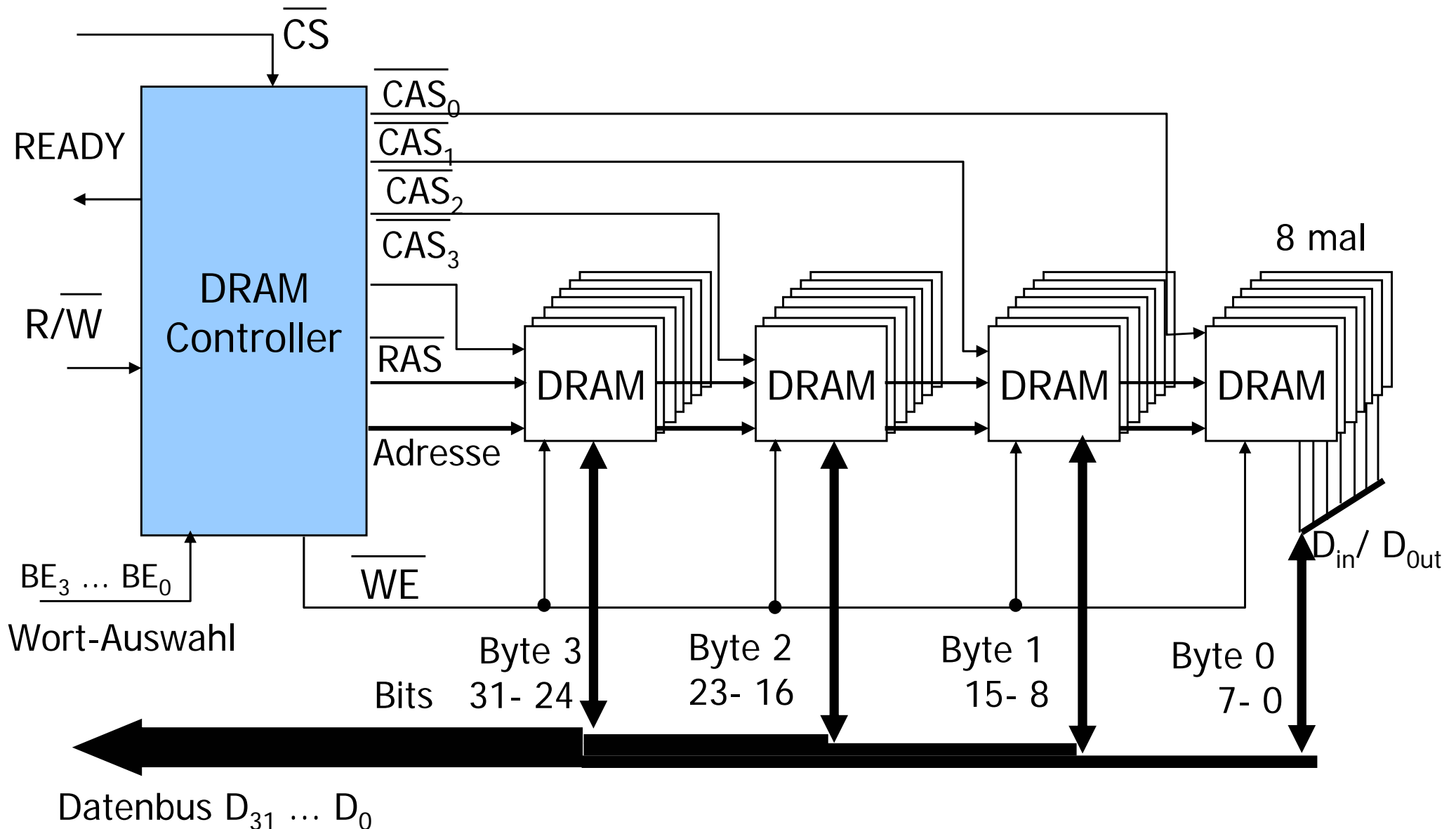
Kartenadresse meist über Schalter (DIP-Schalter) einstellbar

Der Vergleich der Adressbits erfolgt über einen Komparator

Modulaswahl (z. B. SIMMs (*single inline memory module*)) über einen Moduldekoder

Baustein Auswahl über einen Bausteindekoder auf dem Speichermodul

Beispiel eines Speichermoduls



Beispiel eines Speichermoduls

32 Bit breites Speichermodul, Speicher in 4 Bänke zu je acht $n \times 1$ dynamischen Speicherbausteinen organisiert

DRAM-Controller übernimmt Byte- und Bausteinauswahl, Read/Write-Steuerung, Refresh sowie ggf. Wartezyklen (READY-Signal)

Speichermodule

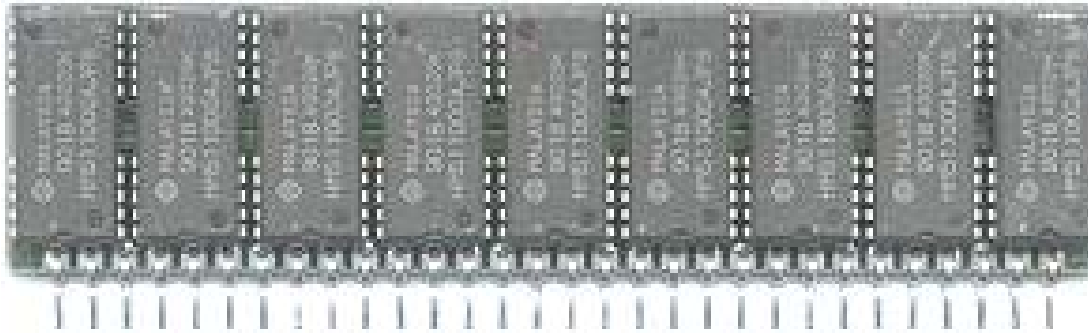
Arbeitsspeicher von modernen Computer werden schon lange nicht mehr wie zu Urzeiten der ersten IBM-PCs mit einzelnen DRAM-ICs bestückt. Seit längerem hat sich schon die Zusammenfassung der einzelnen ICs auf Speichermodulen durchgesetzt.

Vorteile:

- Einfache Realisierung von großen Datenbreiten und Kapazitäten durch Zusammenschaltung der einzelnen ICs.
- Flexibilität beim Handling des Speichers durch einfaches Aufrüsten, Wechseln oder Weiterverwenden der Module.

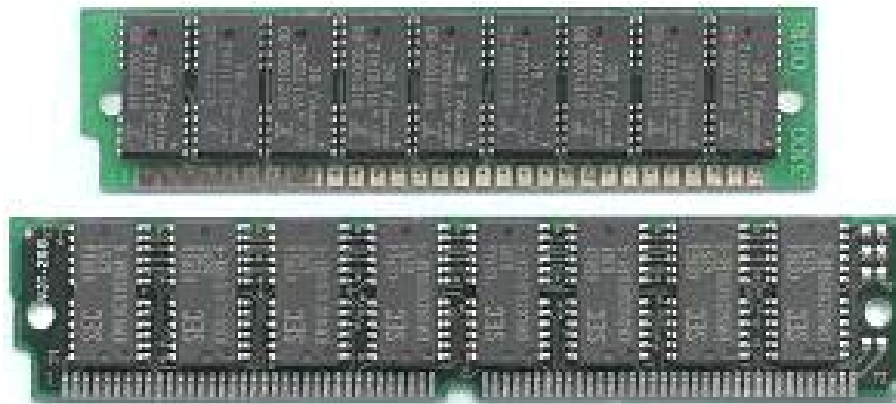
Speichermodule-Typen

- ❑ Single Inline Pin Package (SIPP): veraltet



SIPP-Modul mit seinen 30 pin-förmigen Anschlüssen und einer Datenbreite von 8 Bit.

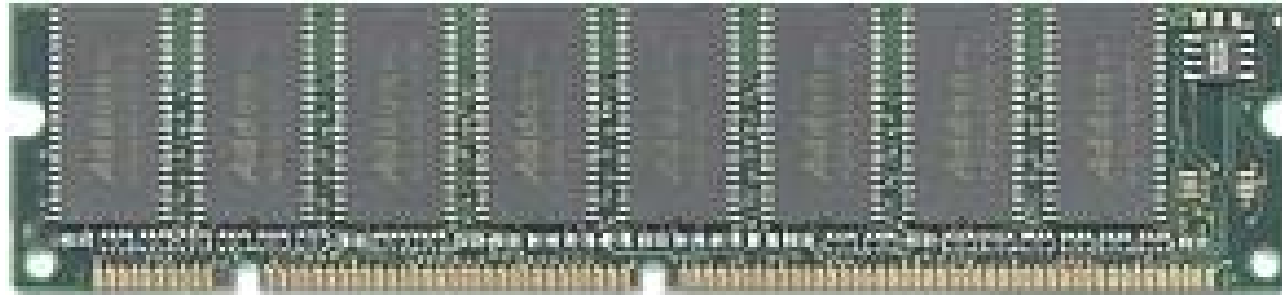
- ❑ Single Inline Memory Module (SIMM): PS/2-Module



SIMM-Module in der 30- und 72-poligen Ausführung mit Datenbreiten von 8 und 32 Bit.

Speichermodule-Typen

□ Dual Inline Memory Module (DIMM):



Die 168-poligen DIMMs besitzen eine Datenbusbreite von 64 Bit.

□ Rambus Inline Memory Modul (RIMM)



Speichertechnologie von Intel. 400 bis 800 MHz sind möglich. Metallabdeckung zur Kühlung der Rambus-DRAMs.

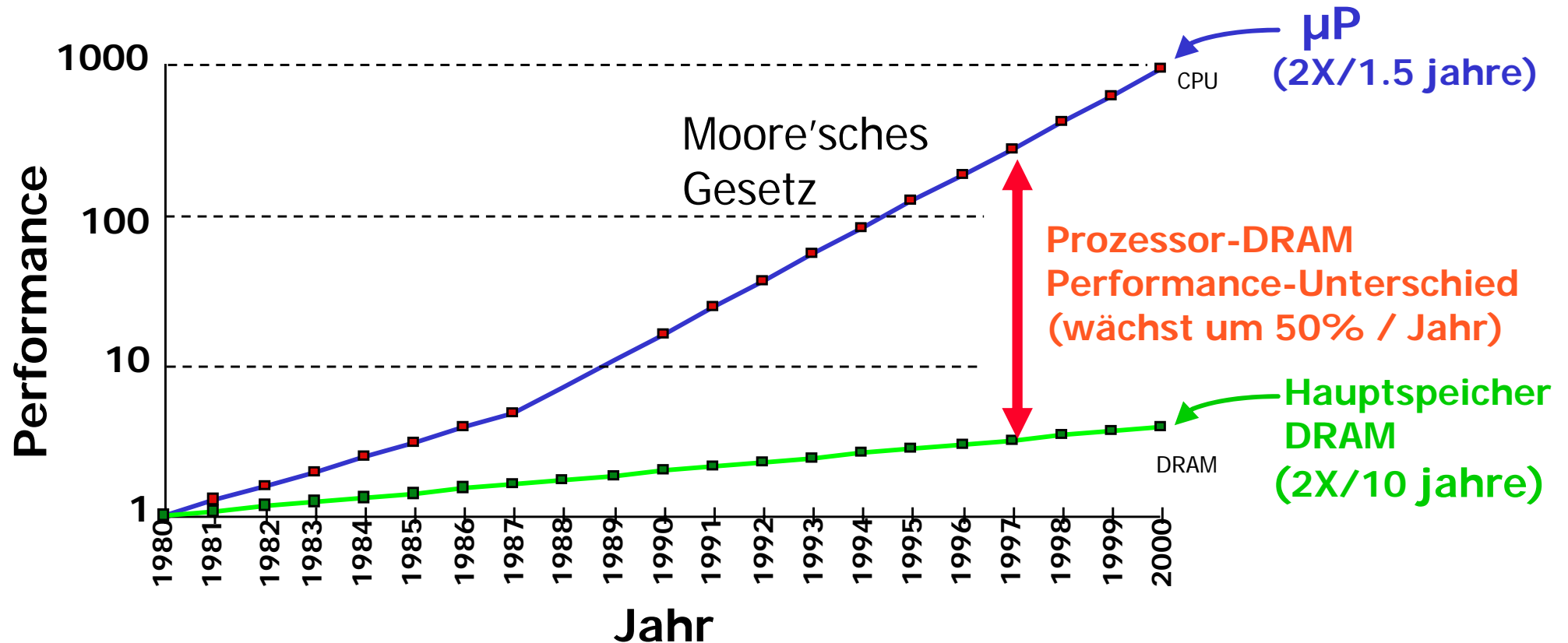
Kapitel 7

Cache-Speicher

- ❑ Speicherhierarchie
- ❑ Funktionsweise
- ❑ Aufbau
- ❑ Organisationsformen



Prozessor-Speicher-Performance-Unterschied



Immer größer werdende Lücke zwischen Verarbeitungsgeschwindigkeit von Prozessoren und Zugriffsgeschwindigkeit der DRAM-Speicherchips des Hauptspeichers

Speicherhierarchie

Ein technologisch einheitlicher Speicher mit ***kurzer Zugriffszeit*** und ***großer Kapazität*** ist aus ***Kostengründen*** i. A. nicht realisierbar

Lösung:

Schichtenweise Anordnung verschiedener Speicher und Verschiebung der Information zwischen den Schichten (Speicherhierarchie)

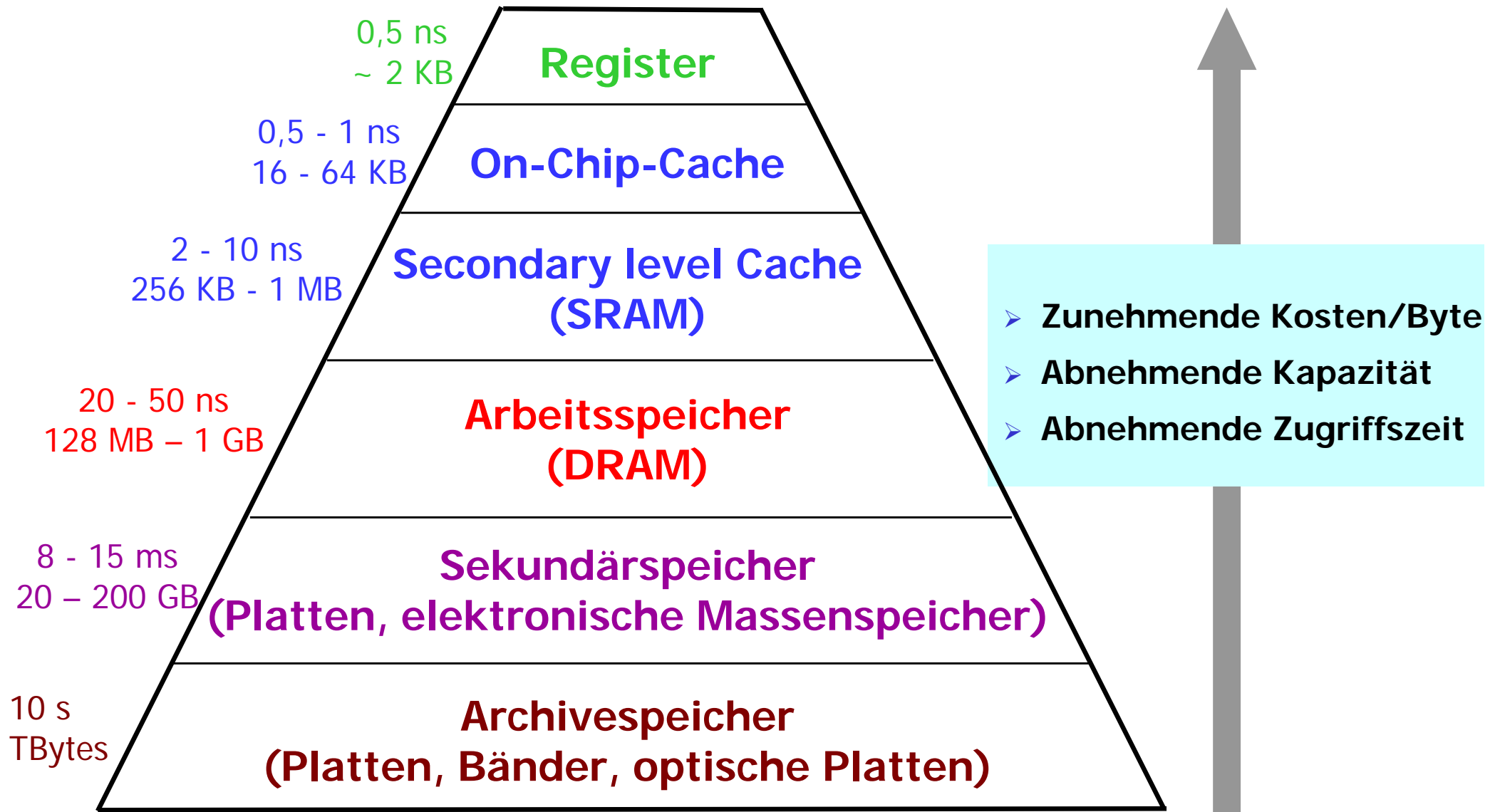
Speicherhierarchie

Speicherhierarchie zum Ausgleich der unterschiedlichen Zugriffszeiten der CPU und des Hauptspeichers.

2 Strategien:

- **Cache-Speicher:**
Kurze Zugriffszeiten → Beschleunigung des Prozessorzugriffs
- **Virtueller Speicher:**
Vergrößerung des tatsächlich vorhandenen Hauptspeichers (z. B. bei gleichzeitiger Bearbeitung mehrerer Prozesse)

Speicherhierarchie



Speicherhierarchie

Wirkung: wie ein großer und schneller Speicher, wenn

- Lokalitätsverhalten der Programmverarbeitung,
- Umlagerung der Information rechtzeitig (Umlagerungsstrategien),
- Inhomogenität des Speichersystems für Benutzer nicht sichtbar ist (Virtueller Speicher)

Leistungsfähigkeit der Hierarchie ist bestimmt durch die Eigenschaften der Speichertechnologien (Zugriffsart, Zugriffszeiten, ...), Adressierung der Speicherplätze und Organisation des Betriebs

Speicherhierarchie

Arbeitsplatz

Schreibtisch-Umgebung

Regale

Magazin

Fernleihe

Register

On-Chip-Cache

Secondary level Cache
(SRAM)

Arbeitsspeicher
(DRAM)

Sekundärspeicher
(Platten, elektronische Massenspeicher)

Archivespeicher
(Platten, Bänder, optische Platten)



Cache-Speicher

Problem:

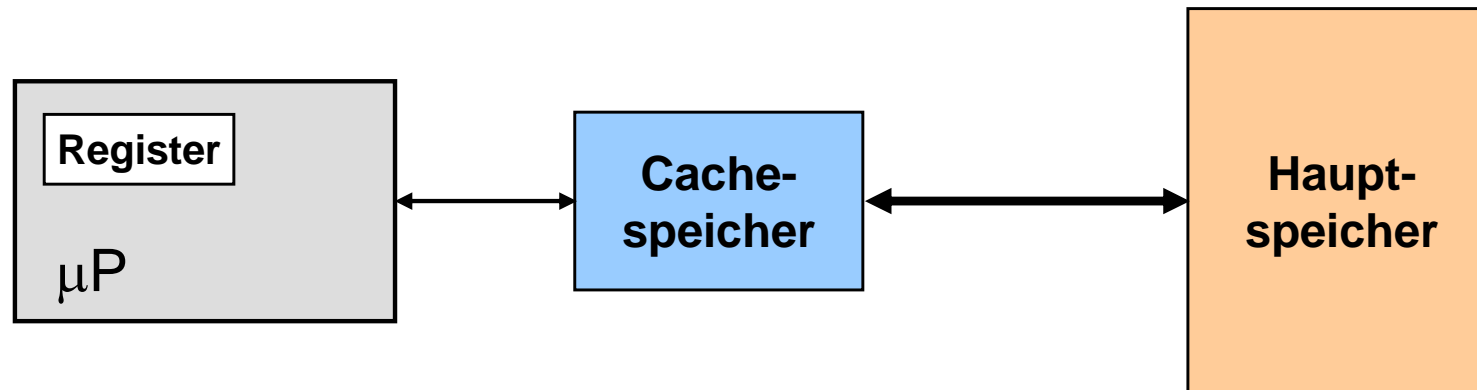
die Buszykluszeit moderner Prozessoren ist erheblich kürzer als die Zykluszeit preiswerter, großer DRAM-Bausteine

- dies zwingt zum Einfügen von Wartezyklen.
SRAM-Bausteine hingegen, die ohne Wartezyklen betrieben werden können, sind jedoch klein, teuer und leistungsintensiv.
- nur kleine Speicher können so aufgebaut werden.

Cache-Speicher

Lösung des Problems:

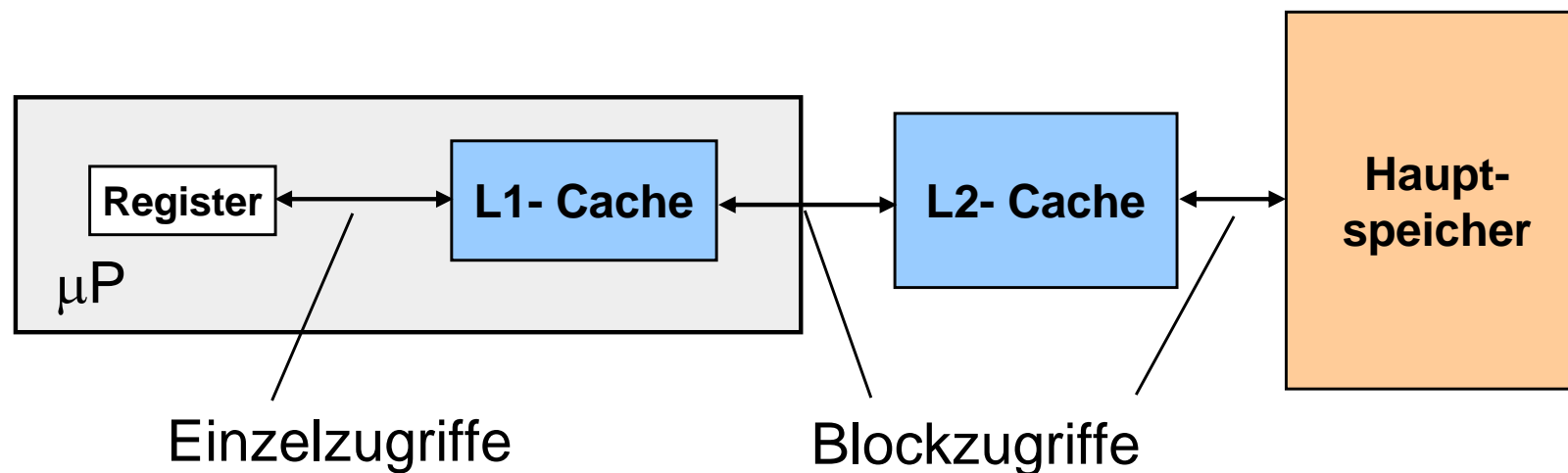
zwischen den Prozessor und den relativ langsamen, aber billigen Hauptspeicher aus DRAM-Bausteinen legt man einen kleinen, schnellen Speicher aus SRAM-Bausteinen, den sogenannten **Cache-Speicher**.



Auf den Cache-Speicher soll der Prozessor fast so schnell wie auf seine Register zugreifen können.

Cache-Speicher

- ❑ **On-Chip-Cache:** integriert auf dem Prozessorchip
 - Sehr kurze Zugriffszeiten (wie die der prozessorinternen Register)
 - Aus technologischen Gründen begrenzte Kapazität
- ❑ **Off-Chip-Cache:** prozessorextern (SRAM-Speicher)



Cache-Speicher

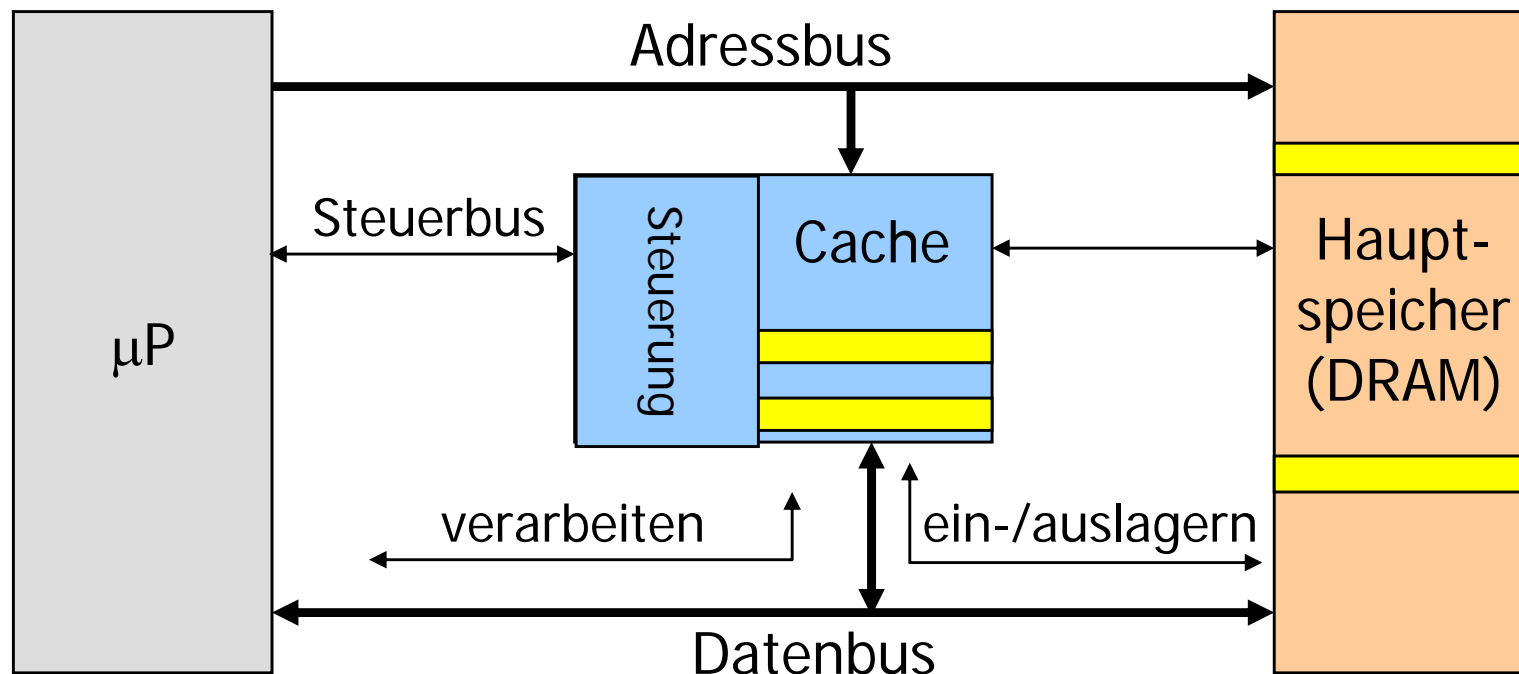
Anwendungsbeispiele:

- Verbesserung der Zugriffszeit des Hauptspeichers eines Prozessors durch einen Cache zur Vermeidung von Wartezyklen (CPU-Cache, Befehls- und Daten-Cache).
- Verbesserung der Zugriffszeit von Plattenspeichern durch einen Cache (Plattencache)

Hier soll im wesentlichen der erste Fall näher betrachtet werden

Cache-Speicher

Unter einem **CPU-Cache-Speicher** versteht man einen kleinen, schnellen Pufferspeicher, in dem Kopien derjenigen Teile des Hauptspeichers bereitgehalten werden, auf die aller Wahrscheinlichkeit nach von der CPU als nächstes zugegriffen wird.



Cache-Speicher

Ein CPU-Cache-Speicher bezieht seine Effizienz im wesentlichen aus der **Lokalitätseigenschaft** von Programmen (*locality of reference*), d. h. es werden bestimmte Speicherzellen bevorzugt und wiederholt angesprochen (z. B. Programmschleifen)

- **Zeitliche Lokalität:** Die Information, die in naher Zukunft angesprochen wird, ist mit großer Wahrscheinlichkeit schon früher einmal angesprochen worden (Schleifen).
- **Örtliche Lokalität:** Ein zukünftiger Zugriff wird mit großer Wahrscheinlichkeit in der Nähe des bisherigen Zugriffs liegen.