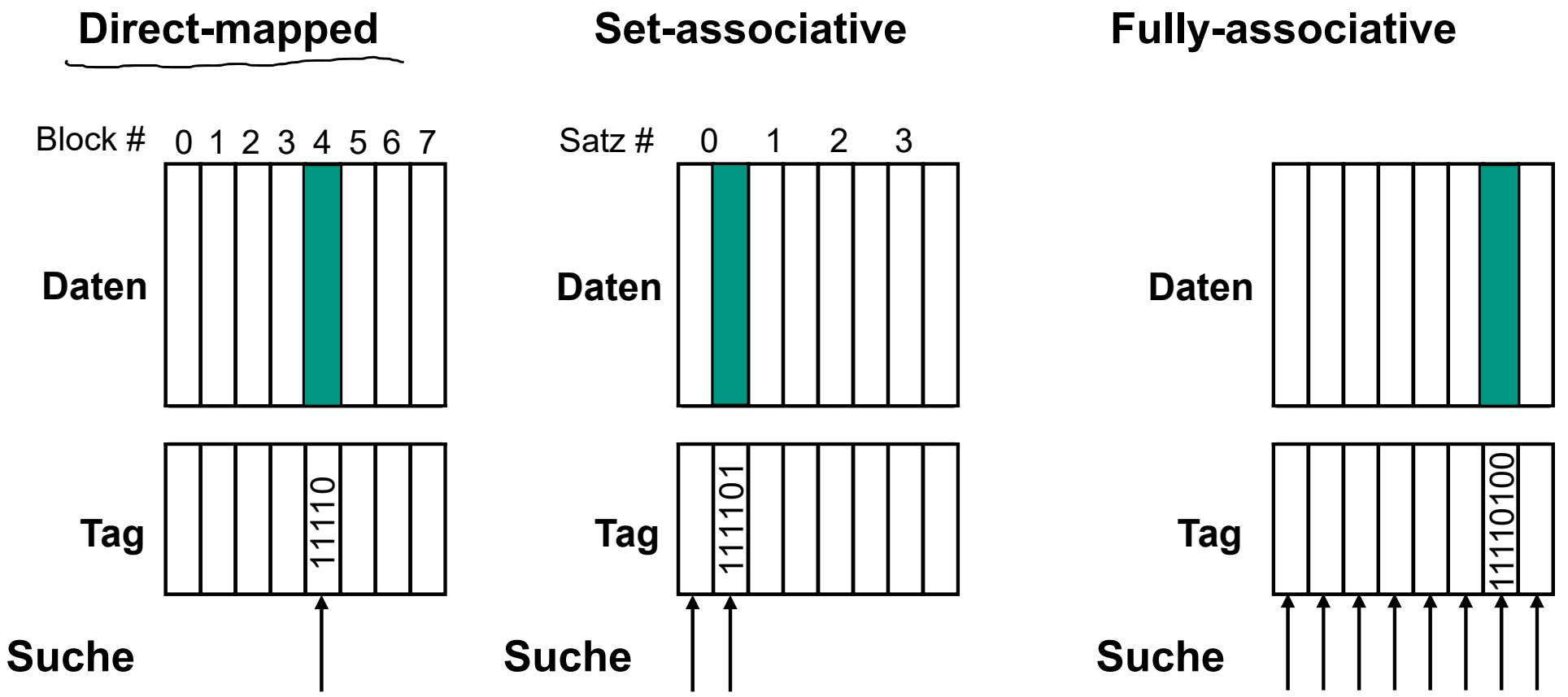
Kapitel 9
Cache-Speicher

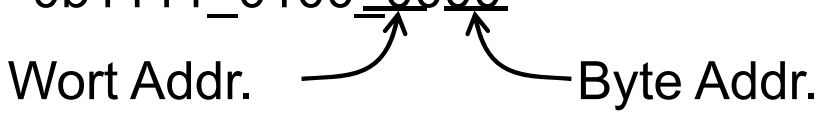
9.2 Cache-Speicher

- **Beispiel: Organisation eines Caches mit 8 Speicherplätzen**
  - Zugriff auf Adresse 0xF40 = 0b1111_0100_0000



9.2 Cache-Speicher

- **Beispiel: Organisation eines Caches mit 8 Speicherplätzen**
  - Zugriff auf Adresse 0xF40 = 0b1111_0100_0000



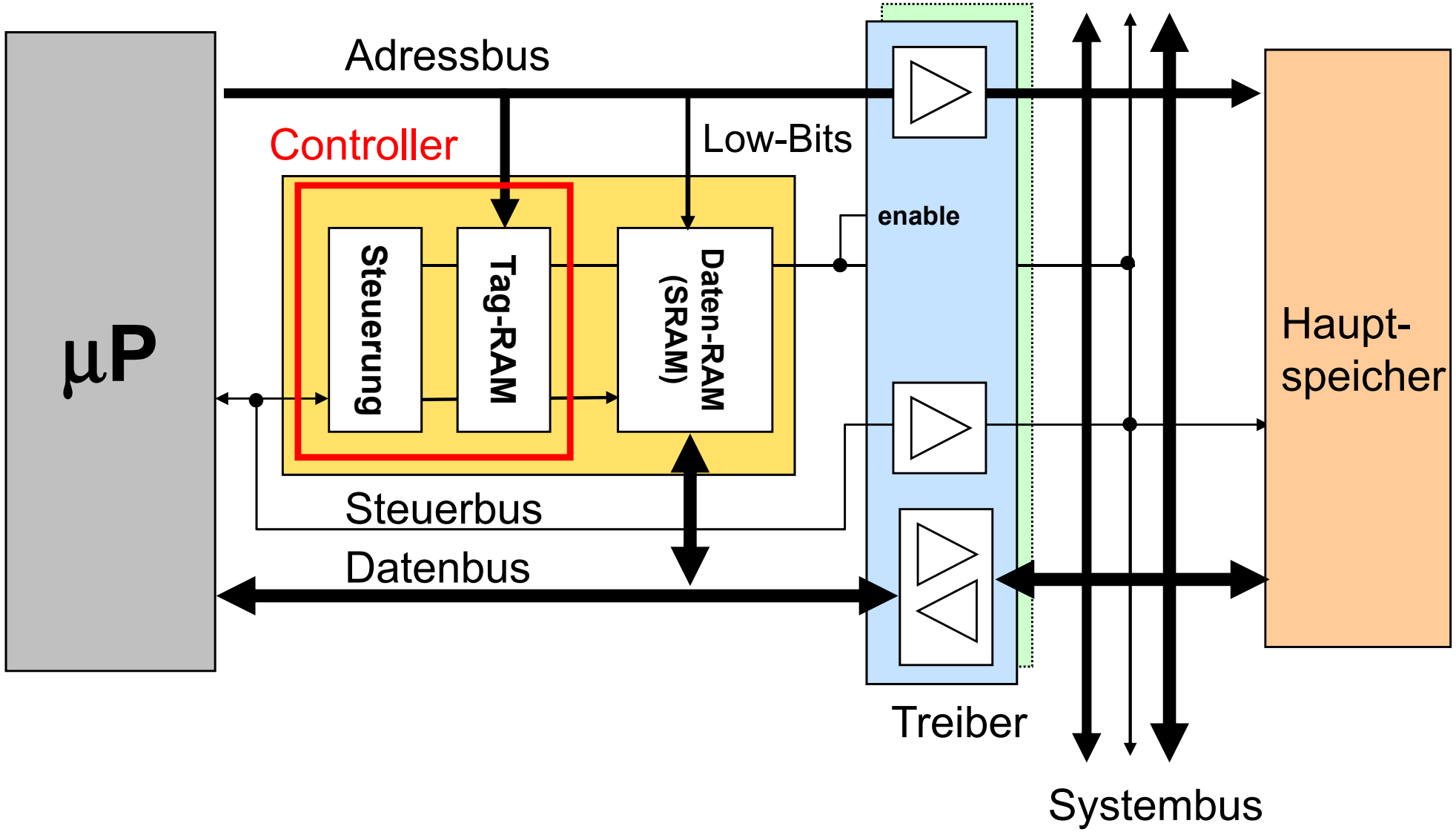
  - Direct Mapped:
    - Speicherblock 0xF40 kann nur an einer Stelle stehen
  - 2-Way Set Associative:
    - Speicherblock 0xF40 kann an zwei Stellen stehen
  - Voll-Assoziativ:
    - Speicherblock 0xF40 kann an allen Stellen stehen

9.2 Cache-Speicher

- **Ersetzungsstrategie**
  - gibt an, welcher Teil des Cachespeichers nach einem Cache-Miss durch eine neu geladene Speicherportion überschrieben wird.
  - Notwendig bei voll- oder n-fach satzassoziativer Cachespeicher-Organisation
  - Meist wird die sehr einfache Strategie gewählt:
    - Wenn ein ungültiger Eintrag vorhanden, ersetze diesen, ansonsten, wähle einen Eintrag aus:
    - Least-Recently-Used (LRU): wähle den Eintrag aus, der am längsten nicht benutzt worden ist.
      - Relativ einfach für 2-fach satzassoziativ, noch machbar für 4-fach satzassoziativ, nicht mehr sinnvoll für höhere Assoziativität
    - Random
      - Annähernd so leistungsfähig wie LRU für hohe Assoziativität

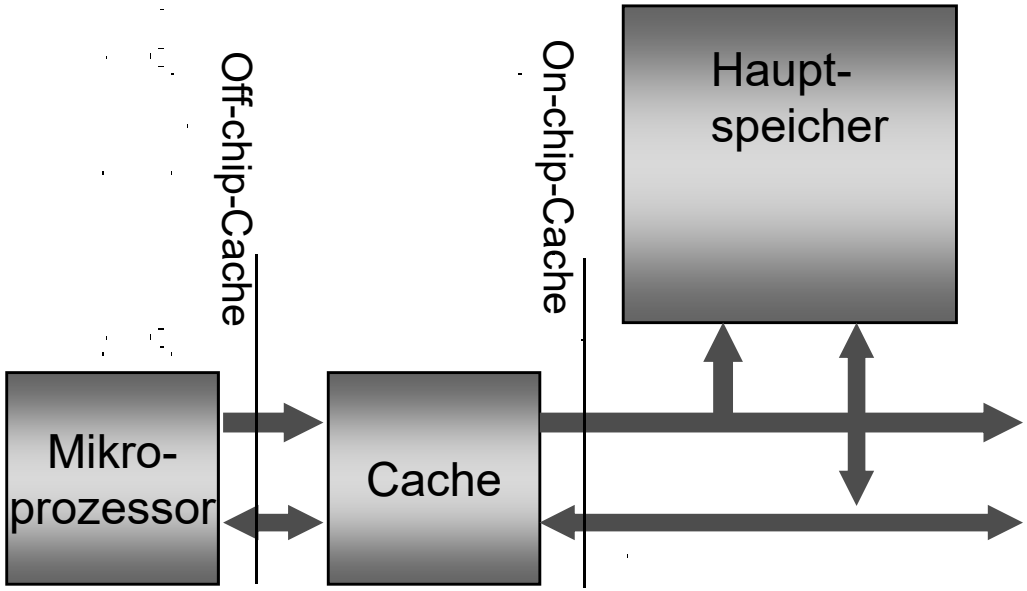
9.2 Cache-Speicher

- **Anbindung des Cache-Speichers (prozessorexterner Cache)**



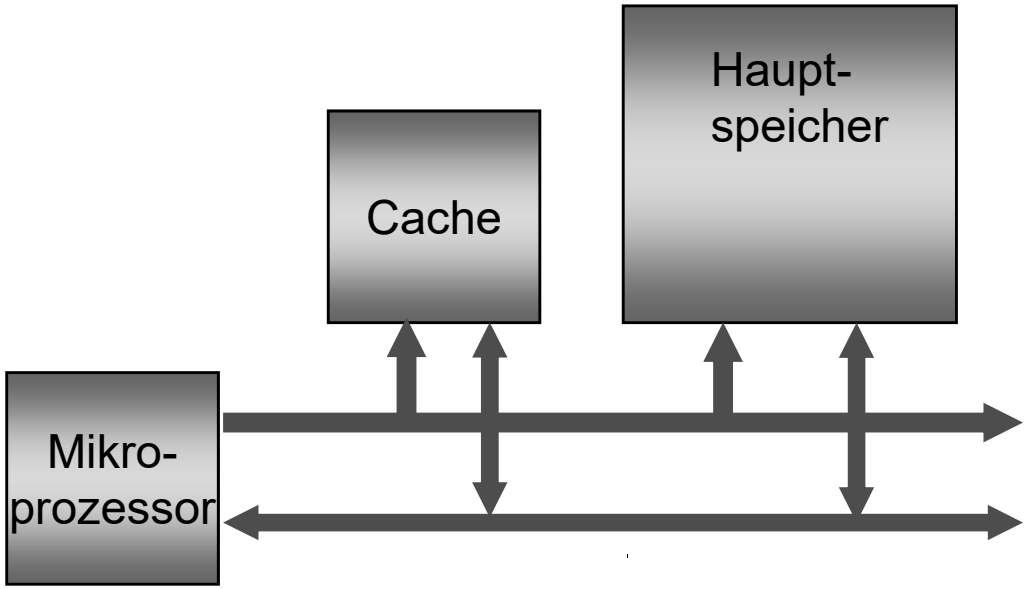
9.2 Cache-Speicher

- **Anbindung des Cache-Speichers**
  - Cache als Pufferspeicher zwischen Mikroprozessor und Hauptspeicher: **Look-through Cache**
  - Prozessor, Cache und Hauptspeicher sind in Reihe angeordnet
    - Typisch für Rechensysteme, bei denen mehrere Prozessor-/Cache-Einheiten auf einen gemeinsamen Speicherbus zugreifen
    - Zugriffsanforderungen der Prozessoren werden von den Caches abgefangen und von der jeweiligen Cache-Steuereinheit nur dann an den Hauptspeicher weitergegeben, wenn diese nicht vom Cache befriedigt werden können



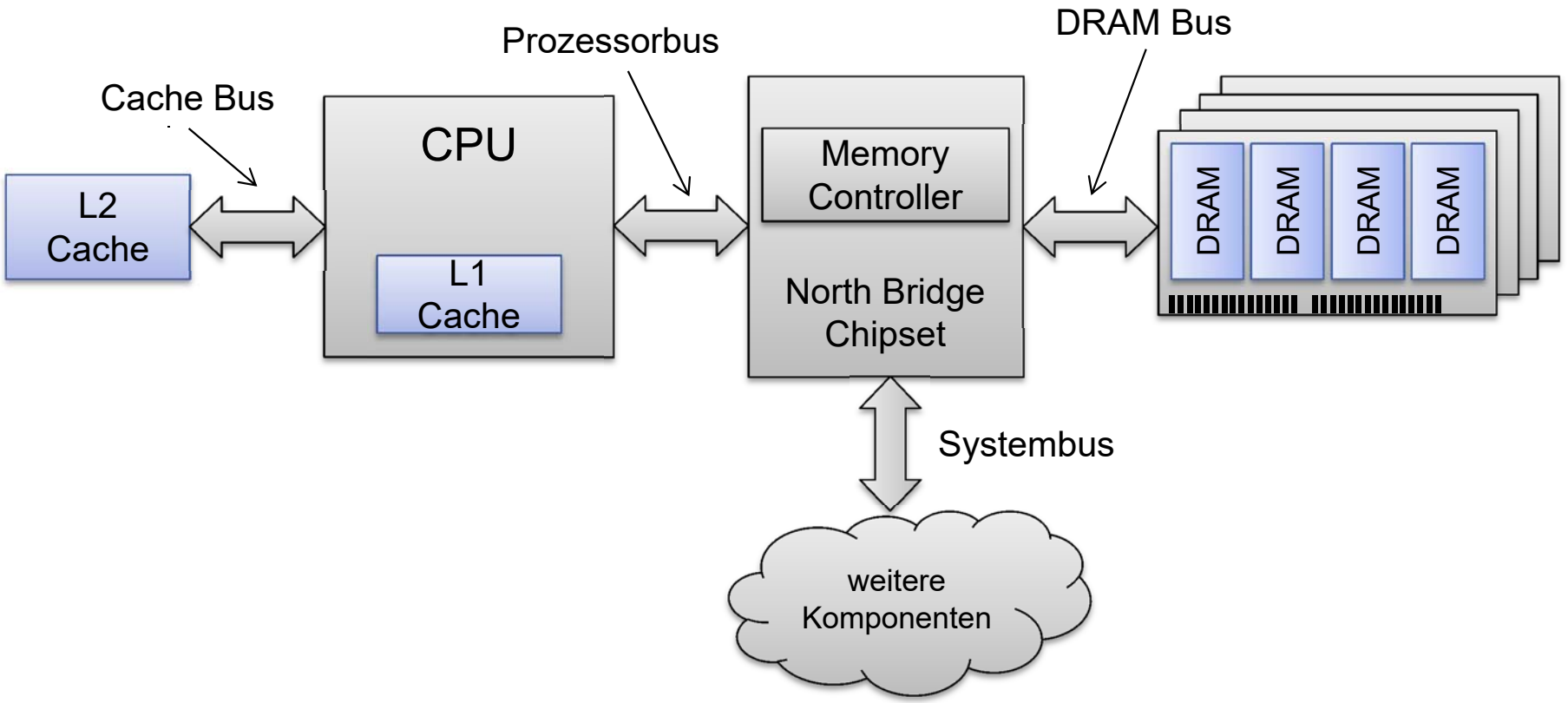
9.2 Cache-Speicher

- **Anbindung des Cache-Speichers**
  - Cache als Pufferspeicher zwischen Mikroprozessor und Hauptspeicher: **Look-aside Cache**
    - Cache und Hauptspeicher werden parallel am Speicherbus berteiben
    - Zugriffsanforderungen des Prozessors geht gleichzeitig an Cache und an Hauptspeicher
      - Kann die Anforderung vom Cache befriedigt werden, wird der Hauptspeicherzugriff gestoppt, wenn nicht, hat Hauptspeicherzugriff schon begonnen



9.2 Cache-Speicher

- **Anbindung des Cache-Speichers**
  - Cache als Pufferspeicher zwischen Mikroprozessor und Hauptspeicher: **Backside-Cache**
    - Neben dem Prozessor-/Speicherbus zusätzlicher Cache-Anschluss des Prozessors



9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Entwurfsparameter**
    - Kapazität
    - Blockgröße
    - Assoziativität
    - Ersetzungstrategie
    - Aktualisierungsstrategie
  - **Cache-Hierarchie**
    - Cache-Anbindung
  - **Leistung:**
    - **Mittlere Zugriffszeit:**
    - t_{Access} = (Hit-Rate) * t_{Hit} + (1 - Hit-Rate) * t_{Miss}

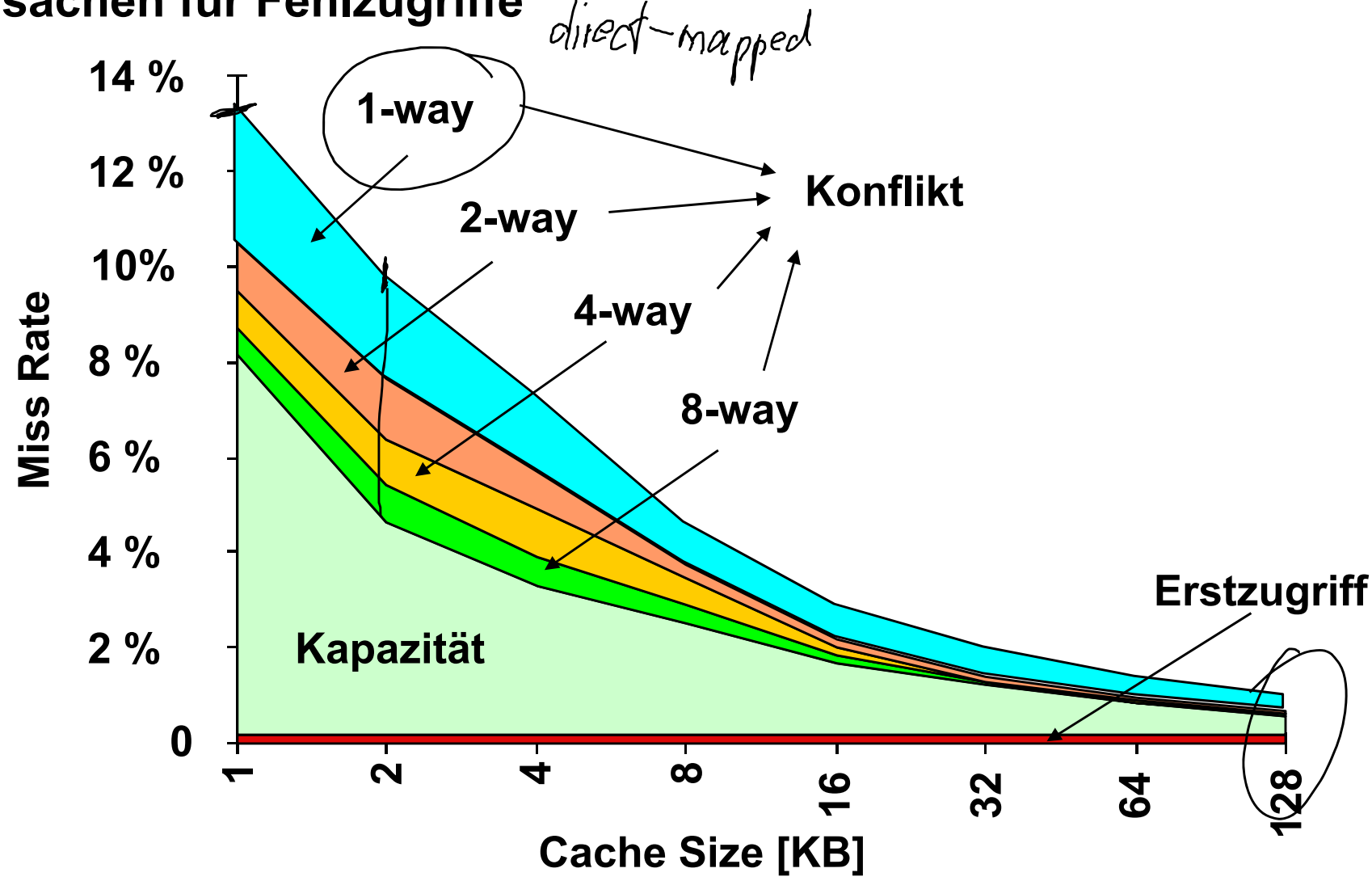
      Fehlzugriffsrate
      - Beeinflusst durch Treffer- / Fehlzugriffsrate und
      - Busbandbreite

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
- **Ursachen für Fehlzugriffe**
  - **Erstzugriff (compulsory - obligatorisch):**
    - Beim ersten Zugriff auf einen Cache-Block befindet sich dieser noch nicht im Cache-Speicher und muss erstmals geladen werden
    - Kaltstartfehlzugriffe (cold start misses) oder Erstbelegungsfehlzugriffe (first reference misses).
  - **Kapazität (capacity):**
    - Falls der Cache-Speicher nicht alle benötigten Cache-Blöcke aufnehmen kann, müssen Cache-Blöcke verdrängt und eventuell später wieder geladen werden.
  - **Konflikt (conflict):**
    - ein Cache-Block wird verdrängt und später wieder geladen, falls zu viele Cache-Blöcke auf denselben Satz abgebildet werden
      - treten bei direkt abgebildeten oder satzassoziativen Cache-Speichern beschränkter Größe auf.

9.2 Cache-Speicher

- Grundlegende Fragen beim Entwurf
- Ursachen für Fehlzugriffe

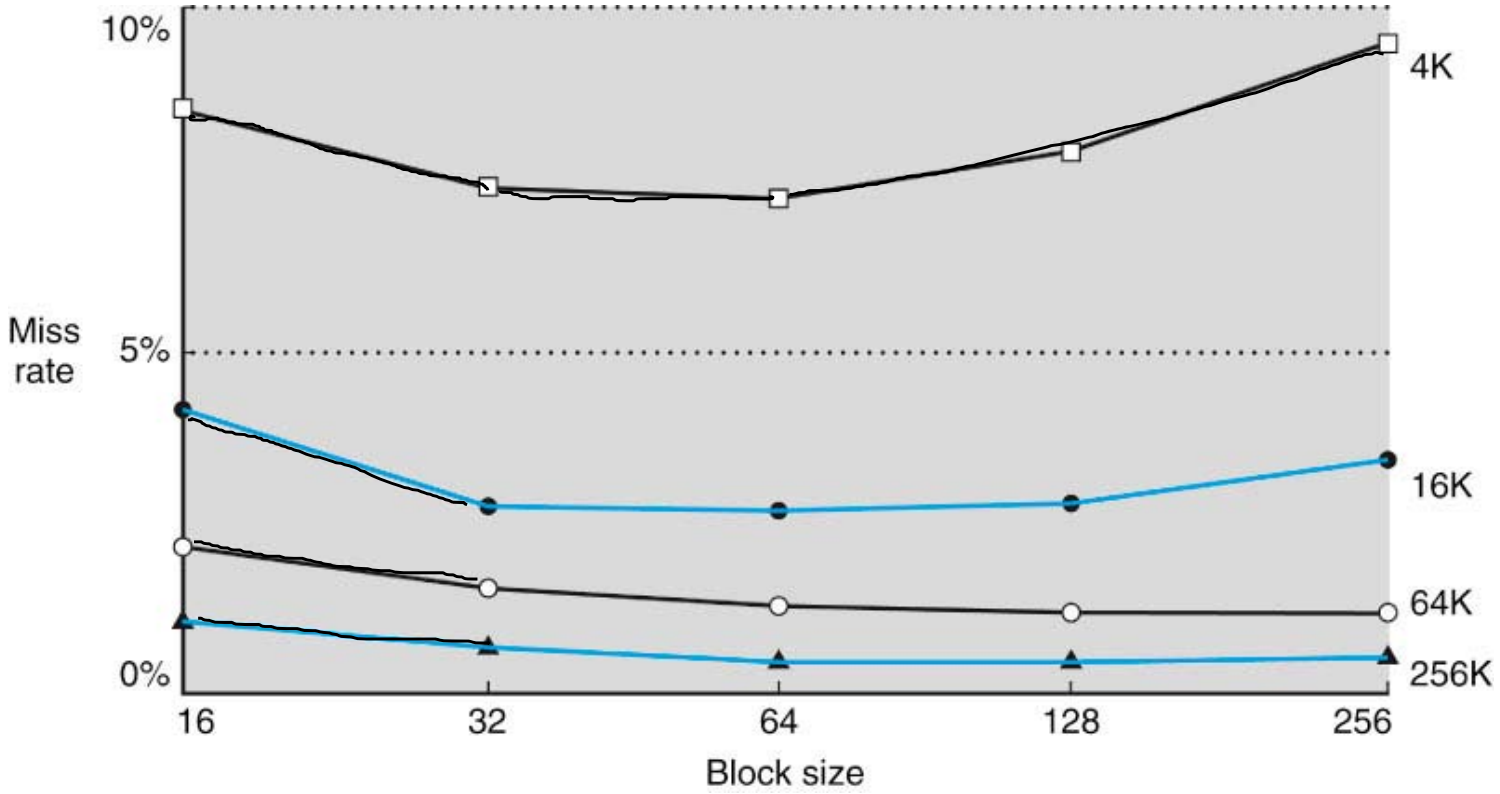


9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Optimierungsparameter Kapazität des Caches:**
    - **Größere Cache-Speicher reduzieren die Fehlzugriffsrate**
    - **Aber:**
      - Es erhöht sich die **Zugriffszeit bei einem Treffer**
        - Zeit für den Zugriff auf die obere Ebene der Speicherhierarchie, worin auch die Zeit enthalten ist, die nötig ist, festzustellen ob ein Zugriff ein Treffer oder ein Fehlzugriff ist
      - Es erhöht sich die Verlustleistung

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Blockgröße auf die Fehlzugriffsrate**



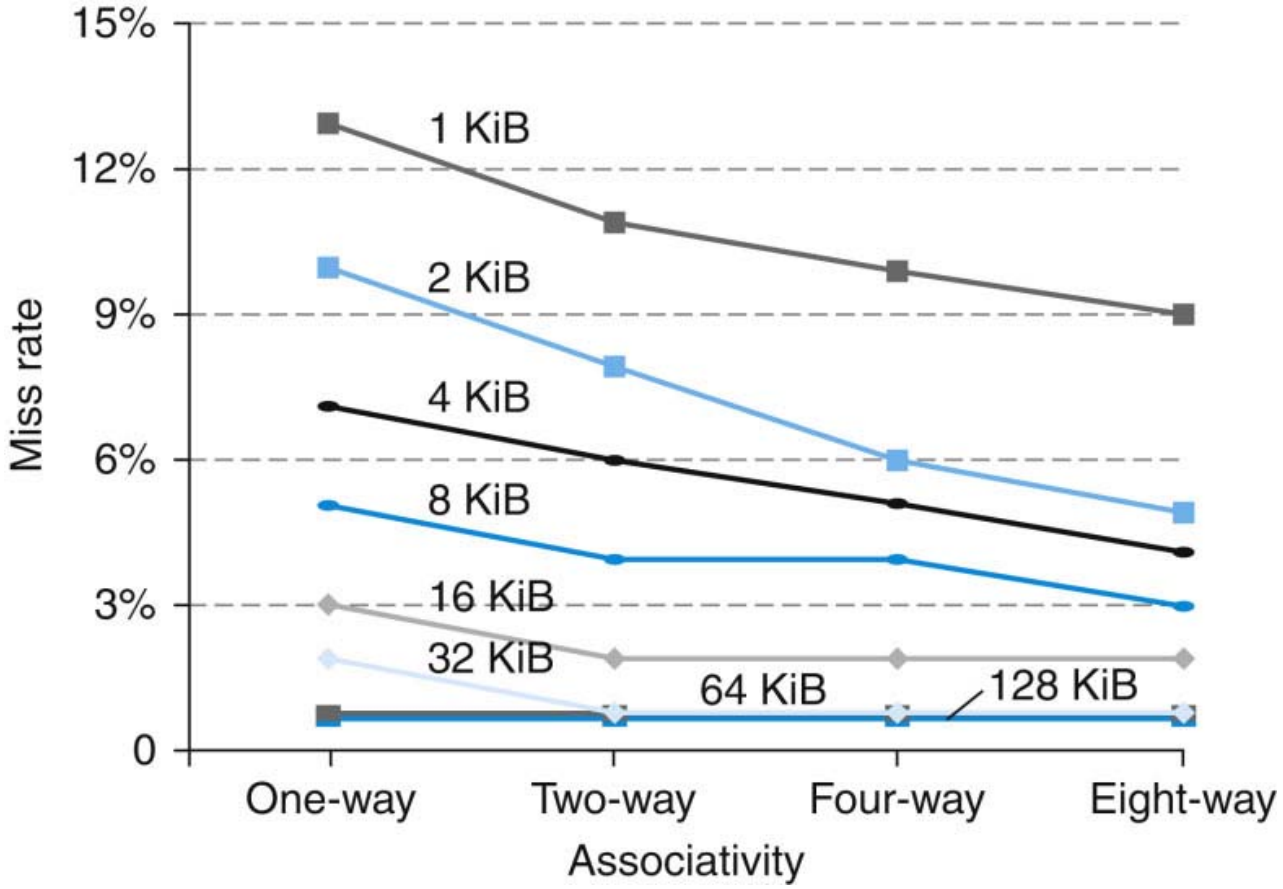
Quelle: J. Hennessy; D. Patterson: Computer Organization and Design.

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Blockgröße auf die Fehlzugriffsrate**
    - Größere Blöcke nutzen die örtliche Lokalität, um die Fehlzugriffsraten zu senken
    - Fehlzugriffsrate sinkt bei steigender Blockgröße (im Beispiel bis zu einer Blockgröße von 64 Bytes
    - Größere Blöcke reduzieren die Fehlzugriffe bei Erstzugriffen
    - **Bei fester Cache-Größe**
      - Die Fehlzugriffsrate steigt, wenn die Blockgröße zu einem wesentlichen Teil der Cache-Größe wird
        - Weniger Blöcke können im Cache abgelegt werden
        - Erhöhung der Kapazitäts- und Konflikt-Fehlzugriffe
        - Block wird aus Cache ausgelagert, bevor auf alle Wörter des Blockes zugegriffen worden ist → örtliche Lokalität zwischen Wörtern in einem Block ist kleiner
      - Kosten eines Fehlzugriffs steigen
        - Mit der Blockgröße steigt die Übertragungszeit und damit der Fehlzugriffsaufwand

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Assoziativität auf die Fehlzugriffsrate**



Quelle: J. Hennessy; D. Patterson: Computer Organization and Design.

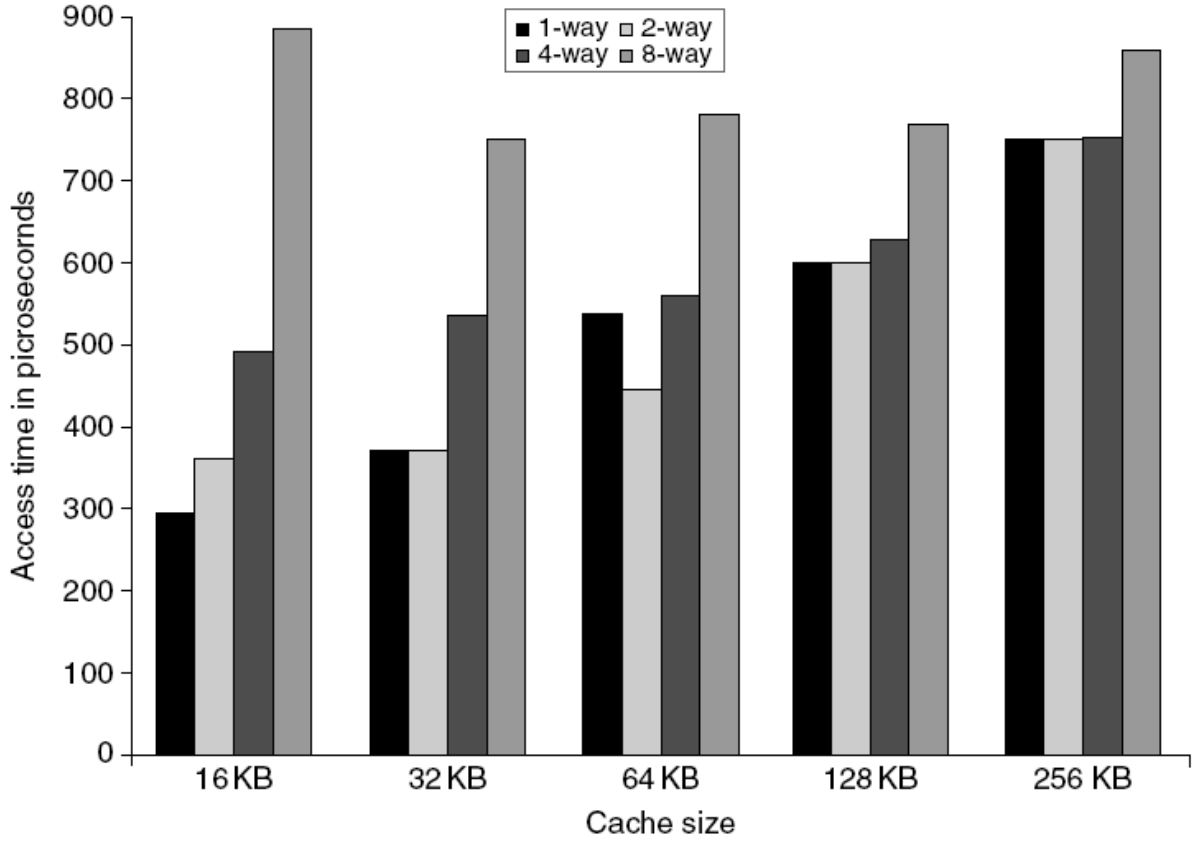
9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Assoziativität auf die Fehlzugriffsrate**
    - Erhöhung der Assoziativität reduziert die Fehlzugriffe aufgrund von Konflikten
    - Beispiel zeigt:
      - Fehlzugriffsrate sinkt für jeden der 8 Cachegrößen bei steigender Assoziativität
      - Die Verbesserung ist signifikant zwischen direct-mapped und 2-fach set-assoziativer Cache-Organisation (20% - 30%)
      - Die Verbesserung ist weniger signifikant mit steigender Assoziativität
      - Kleinere Caches profitieren mehr von höherer Assoziativität, da die grundlegende fehlzugriffsrate von kleineren Caches höher ist

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Assoziativität auf die Fehlzugriffsrate**
    - Erhöhung der Zugriffszeit bei Treffer

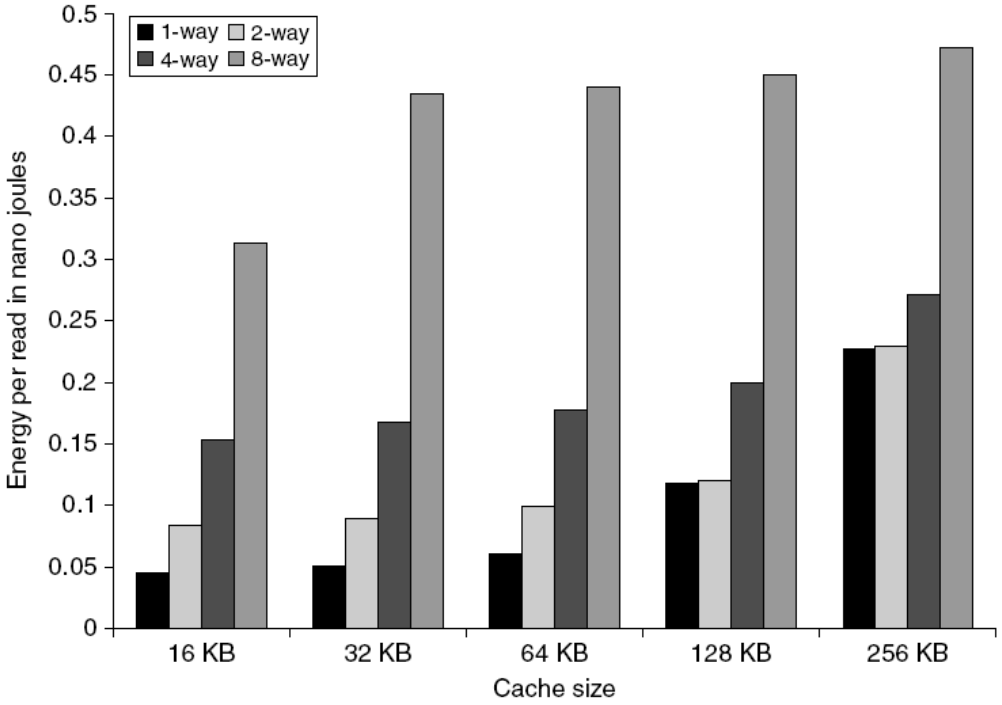
Zugriffszeit vs. Cache-Größe und Assoziativität



9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Einfluss der Assoziativität auf die Fehlzugriffsrate**
    - Erhöhung der elektrischen Verlustleistung (power consumption)
    - Niedrigere Assoziativität reduziert die elektrische Verlustleistung , da auf weniger Cache-Zeilen gleichzeitig zugegriffen wird

Energie vs. Cache-Größe und Assoziativität



J. Hennessy; D. Patterson: Computer Architecture .

9.2 Cache-Speicher

- **Grundlegende Fragen beim Entwurf**
  - **Weitere grundlegende Optimierungsmöglichkeiten**
    - Höhere Anzahl von Cache-Ebenen
      - Reduziert die Speicherzugriffszeit
      - Higher number of cache levels
    - Lese-Zugriffe vor Schreibzugriffen
      - Reduziert den Fehlzugriffsaufwand
    - Eine Vilezahl von Optimierungsmöglichkeiten bestehen

9.2 Cache-Speicher

- **Gültigkeitsproblem**
  - wenn mehrere Verarbeitungskomponenten jeweils unabhängig voneinander auf Speicherwörter des Hauptspeichers zugreifen können.
  - Mehrere Kopien des gleichen Speicherwortes müssen miteinander in Einklang gebracht werden.

  - Eine Cache-Speicherverwaltung heißt **cache-kohärent**, wenn ein Lesezugriff immer den Wert des zeitlich letzten Schreibzugriffs auf das entsprechende Speicherwort liefert.

9.2 Cache-Speicher

- **Mikroprozessorsystem mit DMA-Controller**