

Lernzettel

Speicherhierarchie und Caches:
Cache-Organisation, Assoziativität,
Replacement-Strategien, virtueller Speicher

Universität:	Technische Universität Berlin
Kurs/Modul:	Rechnerorganisation
Erstellungsdatum:	September 20, 2025



Zielorientierte Lerninhalte, kostenlos!
Entdecke zugeschnittene Materialien für deine Kurse:

<https://study.AllWeCanLearn.com>

Rechnerorganisation

Lernzettel: Speicherhierarchie und Caches: Cache-Organisation, Assoziativität, Replacement-Strategien, virtueller Speicher

(1) Überblick und Ziel der Speicherhierarchie. Die Speicherhierarchie dient dazu, hohe Speicherkapazität mit möglichst niedrigen Zugriffskosten zu kombinieren. Durch lokale Adressualität wird versucht, häufig genutzte Daten näher am Prozessor zu halten.

(2) Speicherhierarchie – Aufbau und Größenordnungen. - Register, L1-Cache (D/I), L2-Cache, L3-Cache, Hauptspeicher (DRAM). - Zentrale Komponente: virtueller Speicher, Sekundärspeicher. - Typische Zugriffsgeschwindigkeiten: Register 0,01 ns, L1 1–4 ns, L2 10–40 ns, Hauptspeicher 100 ns bis μ s, Festplatte ms.

(3) Cache-Organisation. - Direktabbildung (Direct-Mapped) - Assoziativität: Set-Associativ (mehrere Zeilen pro Satz) oder vollständig assoziativ.

(4) Adressaufteilung im Cache. Für einen Cache mit Zeilenbreite B , Anzahl Sets S , Assoziativität A gilt allgemein: - Offsetbits: $b = \log_2 B$ - Indexbits: $s = \log_2 S$ - Tagbits: $t = \text{Adressbreite} - b - s$

(5) Miss-Typen und Leistungskennzahlen. - Hit/Miss-Unterscheidung. - Miss-Typen: - Compulsory Misses (erste Referenz), - Conflict Misses (direkte Abbildung), - Capacity Misses (Cache zu klein).

(6) Replacement-Strategien. - LRU (Least Recently Used) - FIFO (First-In-First-Out) - Random Replacement - LFU (Least Frequently Used) - Pseudo-LRU und adaptive Strategien

(7) Schreibpolitik und Konsistenz. - Write-Back vs Write-Through - Write-Allocate vs No-Write-Allocate - Inclusive vs Exclusive Caches; Cache-Coherence-Grundlagen (bei mehreren Caches)

(8) Virtueller Speicher. - Grundidee: logische Adressen vs physische Adressen - Paging: Seitenrahmen, Seitentabellen; hierarchische Seitentabellen - TLB: Translation Lookaside Buffer für schnelle Adressübersetzung

(9) TLB, Seitenabbildung und Page-Faults. - TLB-Misses führen zu Page-Walks in Seitentabellen - Page-Faults: Laden von Seiten aus dem Sekundärspeicher

(10) Leistungskennzahlen und Design-Überlegungen. - Missrate, Zugriffskosten, Bandbreite, Energie - Trade-offs zwischen Größe, Geschwindigkeit, Komplexität

(11) Kurzformeln.

$$\text{Cache-Größe} = \text{Zeilen} \times \text{Zeilenlänge}$$

$$\text{Indexbits} = \log_2(\#\text{Sets}), \quad \text{Offsetbits} = \log_2(\text{Zeilenlänge})$$

$$t = \text{Adressbreite} - \text{Offsetbits} - \text{Indexbits}$$

$$MR = \frac{\#\text{Misses}}{\#\text{Accesses}}$$