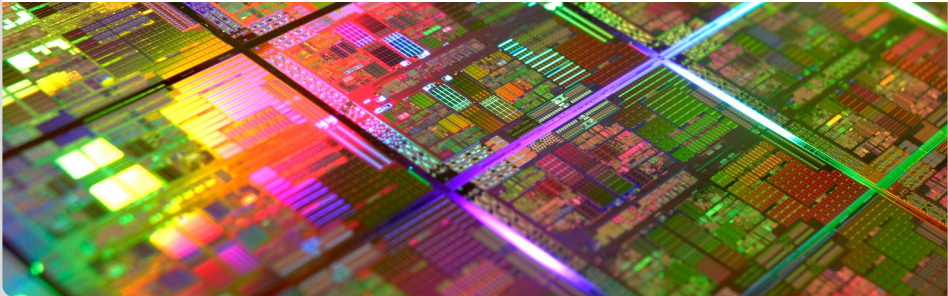


Tutorium Rechnerorganisation

Woche 10

Tutorien 3 und 4 zur Vorlesung Rechnerorganisation



- Einführung in Caches
- Übungsaufgaben

Wozu Caches?

Wo braucht man einen (CPU-)Cache?

Wozu Caches?

Wo braucht man einen (CPU-)Cache?

- Zugriff auf Speicher ist zeitintensiv, CPU muss unnötig warten

Wo braucht man einen (CPU-)Cache?

- Zugriff auf Speicher ist zeitintensiv, CPU muss unnötig warten
- Cache speichert häufig oder wahrscheinlich verwendete Speicherzellen zwischen
 - Zeitliche Lokalität: Wenn eine Speicherzelle referenziert wurde, ist es wahrscheinlich, dass sie in naher Zukunft wieder referenziert wird.
 - Örtliche Lokalität: Referenzierte Speicherzellen liegen oft in der Nähe der zuvor- referenzierten Speicherzellen.

Wo braucht man einen (CPU-)Cache?

- Zugriff auf Speicher ist zeitintensiv, CPU muss unnötig warten
- Cache speichert häufig oder wahrscheinlich verwendete Speicherzellen zwischen
 - Zeitliche Lokalität: Wenn eine Speicherzelle referenziert wurde, ist es wahrscheinlich, dass sie in naher Zukunft wieder referenziert wird.
 - Örtliche Lokalität: Referenzierte Speicherzellen liegen oft in der Nähe der zuvor- referenzierten Speicherzellen.
- Oft Hierarchie von Cache-Speichern: L1-Cache, L2-Cache, ...

- In VL Rechnerorganisation: Beschränkung auf CPU-Caches
- **Cache-Hit:** Benötigtes Datenwort ist im Cache gespeichert und kann von dort geladen werden
- **Cache-Miss:** Benötigtes Datenwort ist nicht im Cache gespeichert und muss aus dem Speicher geladen werden
- **Hit-Rate:** Anteil der Cache-Hits an den Gesamtzugriffen (gibt Effizienz an)

- In VL Rechnerorganisation: Beschränkung auf CPU-Caches
- **Cache-Hit:** Benötigtes Datenwort ist im Cache gespeichert und kann von dort geladen werden
- **Cache-Miss:** Benötigtes Datenwort ist nicht im Cache gespeichert und muss aus dem Speicher geladen werden
- **Hit-Rate:** Anteil der Cache-Hits an den Gesamtzugriffen (gibt Effizienz an)

- In VL Rechnerorganisation: Beschränkung auf CPU-Caches
- **Cache-Hit:** Benötigtes Datenwort ist im Cache gespeichert und kann von dort geladen werden
- **Cache-Miss:** Benötigtes Datenwort ist nicht im Cache gespeichert und muss aus dem Speicher geladen werden
- **Hit-Rate:** Anteil der Cache-Hits an den Gesamtzugriffen (gibt Effizienz an)

- In VL Rechnerorganisation: Beschränkung auf CPU-Caches
- **Cache-Hit:** Benötigtes Datenwort ist im Cache gespeichert und kann von dort geladen werden
- **Cache-Miss:** Benötigtes Datenwort ist nicht im Cache gespeichert und muss aus dem Speicher geladen werden
- **Hit-Rate:** Anteil der Cache-Hits an den Gesamtzugriffen (gibt Effizienz an)

- **Cachezeile/Cacheblock:** Datenmenge, mit der der Cache befüllt wird (meist größer als Wortbreite)
- **Offset:** Niedrigstwertige Adressbits, geben Position innerhalb der Cachezeile an
- **Tag:** Höchstwertige Adressbits, werden zur Identifikation der im Cache vorhandenen Speicherbereiche in jeder Cachezeile mitgespeichert
- **Status Bits:** Zusätzliche Informationen, die pro Cachezeile mitgespeichert werden, z.B.:
 - Valid Bit: Gibt an, ob eine Cachezeile gültig ist
 - Dirty Bit: Gibt an, ob eine Cachezeile gegenüber dem Speicher verändert wurde (für Rückschreibeverfahren)

- **Cachezeile/Cacheblock:** Datenmenge, mit der der Cache befüllt wird (meist größer als Wortbreite)
- **Offset:** Niedrigstwertige Adressbits, geben Position innerhalb der Cachezeile an
- **Tag:** Höchstwertige Adressbits, werden zur Identifikation der im Cache vorhandenen Speicherbereiche in jeder Cachezeile mitgespeichert
- **Status Bits:** Zusätzliche Informationen, die pro Cachezeile mitgespeichert werden, z.B.:
 - Valid Bit: Gibt an, ob eine Cachezeile gültig ist
 - Dirty Bit: Gibt an, ob eine Cachezeile gegenüber dem Speicher verändert wurde (für Rückschreibeverfahren)

- **Cachezeile/Cacheblock:** Datenmenge, mit der der Cache befüllt wird (meist größer als Wortbreite)
- **Offset:** Niedrigstwertige Adressbits, geben Position innerhalb der Cachezeile an
- **Tag:** Höchstwertige Adressbits, werden zur Identifikation der im Cache vorhandenen Speicherbereiche in jeder Cachezeile mitgespeichert
- **Status Bits:** Zusätzliche Informationen, die pro Cachezeile mitgespeichert werden, z.B.:
 - Valid Bit: Gibt an, ob eine Cachezeile gültig ist
 - Dirty Bit: Gibt an, ob eine Cachezeile gegenüber dem Speicher verändert wurde (für Rückschreibeverfahren)

- **Cachezeile/Cacheblock:** Datenmenge, mit der der Cache befüllt wird (meist größer als Wortbreite)
- **Offset:** Niedrigstwertige Adressbits, geben Position innerhalb der Cachezeile an
- **Tag:** Höchstwertige Adressbits, werden zur Identifikation der im Cache vorhandenen Speicherbereiche in jeder Cachezeile mitgespeichert
- **Status Bits:** Zusätzliche Informationen, die pro Cachezeile mitgespeichert werden, z.B.:
 - Valid Bit: Gibt an, ob eine Cachezeile gültig ist
 - Dirty Bit: Gibt an, ob eine Cachezeile gegenüber dem Speicher verändert wurde (für Rückschreibeverfahren)

Gründe für Cache-Misses

- **Compulsary Miss:** Erster Zugriff auf Speicherbereich

- **Compulsary Miss:** Erster Zugriff auf Speicherbereich
- **Capacity Miss:** Speicherdatum wurde aufgrund der beschränkten Cachegröße aus Cache verdrängt

- **Compulsary Miss:** Erster Zugriff auf Speicherbereich
- **Capacity Miss:** Speicherdatum wurde aufgrund der beschränkten Cachegröße aus Cache verdrängt
- **Conflict Miss:** Speicherdatum wurde aufgrund Einschränkungen der Cache-Organisation aus Cache verdrängt

- **Compulsary Miss:** Erster Zugriff auf Speicherbereich
- **Capacity Miss:** Speicherdatum wurde aufgrund der beschränkten Cachegröße aus Cache verdrängt
- **Conflict Miss:** Speicherdatum wurde aufgrund Einschränkungen der Cache-Organisation aus Cache verdrängt
- **Coherency Miss:** Cachekohärenz des Speicherdatums im Cache ist nicht gegeben (nur bei Multiprozessor-Systemen)

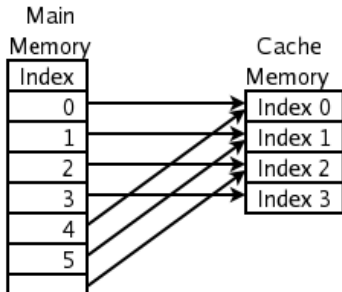
- Festlegung, wo eine bestimmte Adresse im Cache abgelegt und gesucht werden muss
 - Voll-assoziativer Cache (AV)
 - Direkt-abgebildeter Cache (DM)
 - n-fach satzassoziativer Cache (A_n)

- Speicheradresse kann in jeder Cachezeile stehen
- Alle Cachezeilen sind “gleichberechtigt”
- Effizient: Conflict Misses sind nicht möglich
- Hoher Schaltungsaufwand: Beim Suchen muss der Tag jeder einzelnen Cachezeile verglichen werden

- Speicheradresse wird auf genau eine Cachezeile abgebildet
- Adressbits zwischen Tag und Offset geben die Zeilennummer an
- Weniger effizient: Effizienzverlust durch Conflict Misses
- Geringer Schaltungsaufwand: Ein Vergleich für den gesamten Cache ist ausreichend

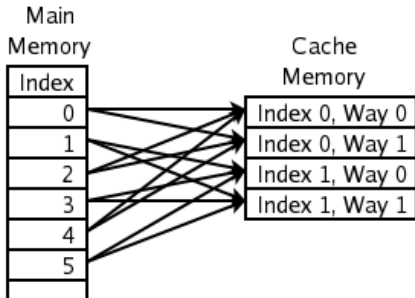
- Kompromiss zwischen voll-assoziativem und direkt-abgebildetem Cache
- Speicheradresse wird auf eine Menge von Zeilen (“Satz”) abgebildet
- Adressbits zwischen Tag und Offset geben die Satznummer an
- Innerhalb des Satzes muss in allen Zeilen gesucht werden
- Kompromiss zwischen voll-assoziativem und direkt-abgebildetem Cache:
 - Conflict Misses können auftreten, sind aber unwahrscheinlicher als beim direkt-abgebildeten Cacheblock
 - Es werden n Vergleiche benötigt

Direct Mapped Cache Fill



Each location in main memory can be cached by just one cache location.

2-Way Associative Cache Fill



Each location in main memory can be cached by one of two cache locations.

(Quelle: Wikipedia)

Wie verhält sich der Cache beim Aktualisieren eines Speicherdatenworts, das im Cache vorhanden ist?

- Write-Through (Durchschreibeverfahren): Neues Datum wird parallel im Speicher und Cache aktualisiert.
- Write-Back (Rückschreibeverfahren): Neues Datum wird nur im Cache gespeichert und dort als “dirty” markiert. Bei Verdrängung muss es vorher in den Speicher zurückgeschrieben werden.

Was sind die Vor- und Nachteile?

Wie verhält sich der Cache beim Aktualisieren eines Speicherdatenworts, das im Cache vorhanden ist?

- Write-Through (Durchschreibeverfahren): Neues Datum wird parallel im Speicher und Cache aktualisiert.
- Write-Back (Rückschreibeverfahren): Neues Datum wird nur im Cache gespeichert und dort als "dirty" markiert. Bei Verdrängung muss es vorher in den Speicher zurückgeschrieben werden.

Was sind die Vor- und Nachteile?

Wie verhält sich der Cache beim Aktualisieren eines Speicherdatenworts, das im Cache vorhanden ist?

- Write-Through (Durchschreibeverfahren): Neues Datum wird parallel im Speicher und Cache aktualisiert.
- Write-Back (Rückschreibeverfahren): Neues Datum wird nur im Cache gespeichert und dort als “dirty” markiert. Bei Verdrängung muss es vorher in den Speicher zurückgeschrieben werden.

Was sind die Vor- und Nachteile?

Wie verhält sich der Cache beim Aktualisieren eines Speicherdatenworts, das **nicht** im Cache vorhanden ist?

- Write-Allocate: Auch in Cache schreiben
- No-Write-Allocate: Keine Cache-Veränderung

Welche Cachezeile wird ersetzt (verdrängt), wenn der Cache voll ist?
(nicht notwendig bei direkt-abgebildeten Caches - [Warum?](#))

- LRU (Least Recently Used)
- LFU (Least Frequently Used)
- FIFO (First In First Out)
- Randomisiert

Theoretisches Optimum: Immer denjenigen Eintrag verdrängen, auf den am längsten nicht zugegriffen wird

Übungsaufgabe 1

Gegeben seien ein **direkt-abgebildeter Cache**, ein **2-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 64 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$12, \$8A, \$9A, \$6C, \$34, \$54, \$68, \$FE, \$17

1. Geben Sie die Längen des Tag-Feldes und die Anzahl der erforderlichen Vergleiche für jede der drei Cache-Architekturen an.
2. Nehmen Sie an, die Caches seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 1

Gegeben seien ein **direkt-abgebildeter Cache**, ein **2-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 64 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$12, \$8A, \$9A, \$6C, \$34, \$54, \$68, \$FE, \$17

1. Geben Sie die Längen des Tag-Feldes und die Anzahl der erforderlichen Vergleiche für jede der drei Cache-Architekturen an.
2. Nehmen Sie an, die Caches seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 1

Gegeben seien ein **direkt-abgebildeter Cache**, ein **2-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 64 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$12, \$8A, \$9A, \$6C, \$34, \$54, \$68, \$FE, \$17

1. Geben Sie die Längen des Tag-Feldes und die Anzahl der erforderlichen Vergleiche für jede der drei Cache-Architekturen an.
2. Nehmen Sie an, die Caches seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 2

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 16 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 2

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 16 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 2

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 16 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 2

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 16 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 3

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 3

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 3

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Übungsaufgabe 3

Gegeben seien ein **direkt-abgebildeter Cache**, ein **4-fach satzassoziativer Cache** und ein **vollassoziativer Cache**.

Die drei Cachespeicher haben jeweils eine **Speicherkapazität von 128 Bytes** und werden in **Blöcken von je 8 Bytes** geladen. Die Hauptspeicheradresse umfasst **32 Bits**.

Falls notwendig, wird die **LRU-Ersetzungsstrategie** verwendet.

Betrachten Sie die Folge der Lesezugriffe auf die folgenden, in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

\$2D, \$38, \$9E, \$D4, \$19, \$29, \$3E, \$9D, \$CA

1. Welche Bits der 32-Bit-Adresse bilden (jeweils) Offset, Tag und Index?
2. Der Zustand eines Cacheblocks wird durch zwei Statusbits (Valid-Bit und Dirty-Bit) gekennzeichnet. Wieviel Speicherplatz wird insgesamt für die Realisierung des Tag-Speichers der einzelnen Cache-Speicher benötigt?
3. Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Ermitteln Sie, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Hit oder Cache-Miss handelt.

Gegeben sei ein **direkt-abgebildeter Cache** mit einer **Speicherkapazität von 128 Byte** und einer **Blockgröße von 16 Bytes**. Die Hauptspeicheradresse umfasst **32 Bits**.

Als Aktualisierungsstrategie wird das **Rückschreib-Verfahren (write back)** verwendet.

1. Skizzieren Sie die Unterteilung der Hauptspeicheradresse.
2. Zur Verwaltung eines Cacheblocks werden zwei Statusbits verwendet: Ein Valid-Bit V und ein Dirty-Bit D. Berechnen Sie den insgesamt erforderlichen Speicherbedarf zur Realisierung dieses Cache-Speichers.

Übungsaufgabe 4

Der Zustand des Cache-Speichers sei durch die Tabelle angegeben. V = 1 kennzeichnen einen gültigen Eintrag im Cache. D = 1 kennzeichnet einen Eintrag im Cache, der gegenüber seiner Originalkopie verändert wurde.

Zeile	D-Bit	V-Bit	Tag
0	0	1	1
1	0	1	1
2	0	0	4
3	0	1	5
4	1	1	0
5	0	1	3
6	1	1	0
7	0	0	1

Betrachten Sie die folgenden Lese- und Schreibzugriffe auf die in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

Adresse	0x44	0xA0	0xC3	0x9E	0x66	0x2D	0x6B	0x49
Read/Write	w	r	w	r	r	w	r	w

- Geben Sie an, ob es sich beim Zugriff auf die jeweiligen Adressen um einen Cache-Miss oder einen Cache-Hit handelt und ob der entsprechende Cacheblock in den Hauptspeicher zurückkopiert werden muss.

Übungsaufgabe 4

4. Skizzieren Sie den schematischen Aufbau des Cache-Speichers. Aus Ihrer Zeichnung sollten die Unterteilung der Hauptspeicheradresse, die Daten- und Adress-Speicher, die Statusbits, alle Vergleicher, Multiplexer/dekoder und das Hit-Signal sowie die Breiten der Daten- und Adressleitungen deutlich sein.

Fertig!