

Prof. Dr. Rolf Drechsler, drechsler@informatik.uni-bremen.de, MZH 4330  
Christina Plump, cplump@uni-bremen.de, MZH 4206

## 4. Übungsblatt zur Vorlesung

# Technische Informatik 1

**Aufgabe 1**

(4 + 2 + 1 Punkte)

In einer CPU wird ein voll-assoziativer Cache mit vier Cacheblöcken verwendet. Der Cache ist initial leer. Innerhalb eines laufenden Programms wird nacheinander von den folgenden Speicheradressen gelesen:

0xB43, 0x427, 0x67C, 0xC6A, 0xB43, 0x406, 0xDA7, 0xE59,  
0x406, 0xB43, 0x5E4, 0xE59, 0x427, 0xC6A, 0x406, 0xE59

(Anmerkung: Das Präfix 0x kennzeichnet eine Zahl in Hexadezimaldarstellung, also zur Basis 16. Die zugehörige Dezimalzahl kann wie folgt ermittelt werden  $0x23A = 2 \cdot 16^2 + 3 \cdot 16^1 + 10 \cdot 16^0 = 570$ , jedoch sind die genauen Speicheradressen für die Aufgabe unerheblich und müssen nicht berechnet werden.)

- a) Es ergibt sich folgende Belegungsfolge der Cacheblöcke, wobei nur die Änderungen an den Cacheblöcken markiert wurden:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	0xB43	0x427	0x67C	0xC6A	0xB43	0x406	0xDA7	0xE59	0x406	0xB43	0x5E4	0xE59	0x427	0xC6A	0x406	0xE59
1	0xB43				hit					hit				0xC6A		
2		0x427				0x406			hit				0x427			
3			0x67C				0xDA7				0x5E4				0x406	
4				0xC6A				0xE59				hit				hit

Welche der Verdrängungsstrategien *FIFO*, *LRU*, *LFU* wurde im gegebenen Cache verwendet? Begründet Eure Aussage anhand der Belegungsfolge. Weist dabei für jede Verdrängungsart nach, ob die Belegung dazu passen kann oder nicht (wenn nicht, gebt den Schritt (die Schritte) Zu wie vielen Cache-Misses kommt es?)

- b) Wie wäre dagegen das Verhalten, wenn die *FIFO-Verdrängungsstrategie (first-in-first-out)* verwendet würde? Zu wie vielen Cache-Misses käme es?
- c) Ist es möglich, dass bei der Verdrängungsstrategie aus a) mehr Cache-Misses auftreten als bei *FIFO*? Begründet Eure Aussage und gebt ggf. ein Beispiel an.

**Aufgabe 2**

(1 + 1 + 1 Punkte)

Beim Entwurf eines Rechners wird als Cache-Speicher ein direct-mapped Cache vorgesehen.

- a) Beim endgültigen Entwurf werden versehentlich statt der  $i$  niedrigstwertigen die  $i$  höchstwertigen Bits mit dem Dekoder verdrahtet, während die restlichen Bits als Adress-Tag verwendet werden. Bei nachfolgenden Messungen stellt man fest, dass die Trefferrate nicht wie erwartet bei 90,3 %, sondern deutlich unter 60 % liegt. Wie ist dies zu erklären?

- b) Vor der Entscheidung zugunsten eines direct-mapped Caches wurde eine neue Schaltung für die Realisierung der *FIFO*-Verdrängungsstrategie entwickelt, durch die gegenüber herkömmlichen Schaltungen 12 % der Gatter eingespart werden können. Überlegt, welchen Einfluss diese Entwicklung auf die Entscheidung für oder gegen einen direct-mapped Cache gehabt haben könnte.
- c) Es wurde eine neue Art von direct-mapped Cache entwickelt, in der nicht mehr die  $i$  niedrigstwertigen Bits mit dem Dekoder verdrahtet werden, sondern stattdessen eine Hash-Funktion für die Adresszuordnung innerhalb des Caches verwendet wird. Diese Hash-Funktion erzeugt pseudozufällige Werte für gegebene Eingangsadressen, was bedeutet, dass für die gleiche Adresse immer dasselbe Ergebnis berechnet wird, sich aber die Ergebnisse für ähnliche Adressen stark unterscheiden können.  
Bei welchem dieser Verfahren (konventionell bzw. Hash-Funktion) kann eine höhere Trefferrate erwartet werden? Welches benötigt vermutlich mehr Berechnungszeit bzw. Chipfläche?

### Aufgabe 3

(2 + 2 Punkte)

Um die Abwägung zwischen Kosten und Größe bzw. Geschwindigkeit von Caches zu erleichtern, werden oft mehrere Level von Caches verwendet, sodass mit zunehmendem Cachelevel die Zugriffsgeschwindigkeit und Kosten ab und die Cachegröße zunehmen. Die Suche nach einem Eintrag in einem Cache läuft die Hierarchie entlang; es wird zuerst im untersten Cachelevel (L1) gesucht, dann in dem Level darüber (L2) und nach diesem Verfahren weiter, bis der Eintrag gefunden wurde oder ein Cache-Miss festgestellt wird. Bei diesen mehrstufigen Caches wird zwischen zwei grundlegenden Varianten unterschieden, den *inklusiven* und *exklusiven* Caches. Bei einem *inklusiven Cache* sind Einträge in allen Cachelevels gleichzeitig gespeichert, d. h. ein Eintrag im L1-Cache ist gleichzeitig auch im L2- und L3-Cache; wird ein Eintrag aus einem niedrigen Cachelevel verdrängt, ist er in den höheren noch vorhanden. Bei einem *exklusiven Cache* hingegen ist eine Adresse immer nur in einem Cachelevel vorhanden und kann, falls sie aus diesem verdrängt wird, entweder verworfen oder in einen höheren Cachelevel eingetragen werden.

- Stellt in einer Tabelle die Vor- und Nachteile von mehrstufigen inklusiven und mehrstufigen exklusiven Caches gegenüber.
- Ist eine Mischung der Verdrängungstrategie oder der Cache-Art (vollassoziativ oder direct-mapped) für verschiedene Cachelevel sinnvoll? Begründet Eure Antwort.

### Aufgabe 4

(3 Punkte)

Zeigt algebraisch, dass die durchschnittliche Anzahl von Sektoren einer Festplatte, die bei der Positionierung des Schreib-/Lesekopfes auf einem beliebigen Sektor einer Spur überschritten wird,  $\frac{(n-1)}{2}$  beträgt. Annahmen:

- Der Schreib-/Lesekopf ist bereits auf der richtigen Spur positioniert.
- Die Spur enthält  $n$  Sektoren.
- Der Zugriff auf jeden Sektor der Spur ist gleichwahrscheinlich.

In welchem Zusammenhang steht dieses Ergebnis mit der Aussage aus der Vorlesung, dass die Latenzzeit 0,5 Umdrehungen beträgt?

**Abgabe:** bis **Donnerstag, den 18.05.2023, 08:15 Uhr per e-Mail an den Tutor und Christina Plump mit folgendem Betreff:** [TI-Abgabe] <Tutorium> - <Gruppe>: Blatt <Blatt>