

Rechnerstrukturen

Wintersemester 2005/2006

9. Übungsblatt

Abgabetermin: 01.02.06 und 02.02.06 in der Übung

Aufgabe 1:

16 Punkte

Gegeben sei ein Hauptspeicher der Größe 1 KB sowie ein Cache mit 32 Bytes. Es werde auf Byte-Ebene adressiert, eine Adresse habe also eine Länge von 10 Bits. Der Cache verwende zudem die LRU-Ersetzungsstrategie.

Die CPU greife auf den Speicher aufgrund des folgenden Referenzstrings zu (Adreßangaben in Hexadezimaldarstellung):

0x2A, 0x2B, 0x2D, 0x2C, 0x2A, 0x2D, 0x2C, 0x18C, 0x18A, 0x18C, 0x2A, 0x18A, 0x2B, 0x2C, 0x2A

Zu Beginn sei der Cache leer. Skizzieren Sie den Ablauf der Speicherzugriffe, indem Sie jeweils angeben

- ob der Zugriff zu einem hit oder einem miss führt
- inwiefern sich der Inhalt des Caches verändert

Betrachten Sie dabei folgende Cachetypen und diskutieren Sie jeweils kurz die ermittelten Hitraten.

- direct mapped, mit Bit 0-4 einer Adresse als Index und Bit 5-9 als Tag. Jede der 32 Cache-Zeilen speichere hier also ein 1 Byte Datum.
- direct mapped, mit Bit 5-9 einer Adresse als Index und Bit 0-4 als Tag. Jede der 32 Cache-Zeilen speichere hier also wiederum ein 1 Byte Datum.
- direct mapped, mit Bit 2-4 einer Adresse als Index und Bit 5-9 als Tag. Zusätzlich werden Bit 0 und 1 als Offset genutzt. Dabei speichert eine Cache-Zeile mehrere Daten, wobei bei zwei Adressen mit ansonsten identischem Tag und Index der Offset ein Datum innerhalb der Zeile auswählt. Der Cache hat in diesem Fall also 8 Zeilen, wobei jede Zeile vier 1 Byte Daten speichert.
- 2-way set associative mit Bit 2-3 als Index und Bit 4-9 als Tag.
- fully associative. Es gebe hier also 32 Cache-Zeilen, von denen jede ein 1 Byte Datum speichert.

Aufgabe 2:

2+4+6 Punkte

Betrachten Sie den Cache-Typ direct mapped mit Offset wie oben beschrieben. In einem Hauptspeicher (byteweise Adressierung) mit s Bytes betrage die Breite eines Datenworts n Bits, als Index werden i Bits einer Adresse verwendet, wobei Bit k das erste Bit des Index von LSB (LSB = Bit 0) aus betrachtet sei. Als Offset werden entsprechend die Bits 0 bis $k - 1$ genutzt. Die restlichen Bits entfallen auf das Tag.

- Wieviele Zeilen umfasst ein solcher Cache?
- Wie ist eine Zeile eines solchen Caches aufgebaut und wieviele Bits werden für jeden Teil sowie für eine ganze Cache-Zeile benötigt.
- Nennen Sie Gründe, warum die Offset-Erweiterung des direct-mapped Caches sinnvoll ist?

- bitte wenden -

Aufgabe 3:

6 Punkte

Was können Sie über die Cache Hitrate für folgende Folge von Speicherzugriffen sagen? Welche Eigenschaft fehlt hier, die reale Speicherzugriffsfolgen in der Regel aber aufweisen?

0x0, 0x4, 0x8, 0x10, 0x14, 0x18, 0x1C, 0x24, 0x28, 0x2C, 0x30, 0x34, 0x38, 0x3C, 0x40, 0x44, 0x48,
0x4C, 0x50, 0x54, 0x58, 0x5C

Aufgabe 4:

3+3+2 Punkte

Ein 2-way set associative Cache eines System mit 32-Bit Adressen speichere je ein 4-Byte Wort je Zeile und habe eine Kapazität von 128 K Bytes. Adressiert wird auf Byte-Ebene.

- a. Wieviele Bits werden für Index und Tag benötigt?
- b. Welchen Index haben die hexadezimalen Adressen 0x0284A482, 0x01148C89, 0x0038CF00 und 0x0038CF01?
- c. Welche der Adressen aus b können parallel im Cache gehalten werden?

Erinnerung: Zweimaliges Vorrechnen in der Übung ist Schein-Voraussetzung!