Tobias Eidelpes
2015WS

Technische Grundlagen der Informatik Übungsblatt 7

Aufgabe 1.

Lösung.

(a)

(1) **Direct Mapped Cache**

Adresslänge $log_2(32\text{ GiB} = 35\text{ bit})$
Tag $log_2(32\text{ GiB}) - log_2(256\text{ KiB}) \approx 17\text{ bit}$
Index $256\text{ KiB} \div 128\text{ byte} = 2048\text{ Blöcke} = 2^{11}$
Offset $log_2(128\text{ byte}) = 7\text{ bit}$

(2) **4-Way Set Associative Cache**

Adresslänge $log_2(32\text{ GiB} = 35\text{ bit})$
Tag $log_2(32\text{ GiB}) - log_2(256\text{ KiB}) \approx 17\text{ bit}$
Index $256\text{ KiB} \div 128\text{ byte} = 2048\text{ Blöcke} = 2^{11}$
Offset $log_2(128\text{ byte}) = 7\text{ bit}$

(3) **Fully Associative Cache**

Adresslänge $log_2(32\text{ GiB} = 35\text{ bit})$
Tag $log_2(32\text{ GiB}) - log_2(128\text{ byte}) \approx 28\text{ bit}$
Index 0 bit
Offset $log_2(128\text{ byte}) = 7\text{ bit}$

(b)

$$0\text{xBADCODED} = (1011\,1010\,1101\,1100\,0000\,1101\,1110\,1101)_2$$
$$0\text{x12345432} = (0001\,0010\,0011\,0100\,0101\,0100\,0011\,0010)_2$$

(1) **Direct Mapped Cache**

(1)

Tag	Index	Offset
00010111010110111	00000011011	1101101
2EB7	1B	6D

(2)

Tag	Index	Offset
10010001101	00010101000	0110010
048D	A8	32

(2) **4-Way Set Associative Cache**

(1)

Tag	Index	Offset
00010111010111011100	000011011	1101101
BADC	1B	6D

(2)

Tag	Index	Offset
00010010001101000	010101000	0110010
2468	A8	32

(3) **Fully Associative Cache**

(1)

Tag	Offset
00010111010110111100000011011	1101101
175B81B	6D

(2)

Tag	Offset
10010001101000101010000	0110010
2468A8	32

(c)

Direct Mapped Cache

$$17 \text{ bit Tag} + 1 \text{ Valid} + 1 \text{ Dirty} = 19 \text{ bit} \cdot 2048 = 38912 \text{ bit} = 4864 \text{ byte}$$

4-Way Set Associative Cache

$$17 \text{ bit Tag} + 1 \text{ Valid} + 1 \text{ Dirty} + 3 \text{ bit} = 22 \text{ bit} \cdot 2048 = 45056 \text{ bit} = 5632 \text{ byte}$$

Fully Associative Cache

$$28 \text{ bit Tag} + 1 \text{ Valid} + 1 \text{ Dirty} + 10 \text{ bit} = 40 \text{ bit} = 5 \text{ byte}$$

Aufgabe 2.

Lösung.

t	Adresse	Tag	Offset	h/m	r/w	Tag	v	a	Tag	v	a
t_0	-	-	-	-	-	000000	0	0	000000	0	0
t_1	-	-	-	-	-	000000	0	0	000000	0	0
t_2	-	-	-	-	-	000000	0	0	000000	0	0
t_3	0x23	001000	11	m	w	001000	0	0	000000	0	0
t_4	0x11	000100	01	m	w	001000	1	1	000100	0	0
t_5	0x11	000100	01	h	r	001000	1	0	000100	1	1
t_6	0x33	001100	11	m	w	001100	1	0	000100	1	1
t_7	0x11	000100	01	h	r	001100	1	1	000100	1	0
t_8	0x23	001000	11	m	r	001000	1	0	000100	1	1
t_9	0x11	000100	01	h	w	001000	1	1	000100	1	0

Die *miss rate* ist gleich $4/7 \approx 57\%$

Aufgabe 3.

Lösung.

(a) Es können maximal 2 Blöcke · 4 Sets · 16 Datenwörter = 128 byte an Nutzdaten gespeichert werden.

(b)

	7	6	5	4	3	2	1	0	
MSB	Tag	Tag	Index	Index	Offset	Offset	Offset	Offset	LSB

(c)

Adresse (dezimal)	Adresse (binär)	Hit/Miss	Set	Block	Inhalt	Anzahl Zugriffe
65	0100 0001	Miss	0	0	mem[64-79]	1
77	0100 1101	Hit	0	0	mem[64-79]	2
111	0110 1111	Miss	2	0	mem[96-111]	1
222	1101 1110	Miss	1	0	mem[208-223]	1
42	0010 1010	Miss	2	1	mem[32-47]	1
121	0111 1001	Miss	3	0	mem[112-127]	1
110	0110 1110	Hit	2	0	mem[96-111]	2
48	0011 0000	Miss	3	1	mem[48-63]	1
163	1010 0011	Miss	2	1	mem[160-175]	1
208	1101 0000	Hit	1	0	mem[208-223]	2
242	1111 0010	Miss	3	0	mem[240-255]	1
220	1101 1100	Hit	1	0	mem[208-223]	3
78	0100 1110	Hit	0	0	mem[64-79]	3
120	0111 1000	Miss	3	0	mem[112-127]	1
51	0011 0011	Hit	3	1	mem[48-63]	2

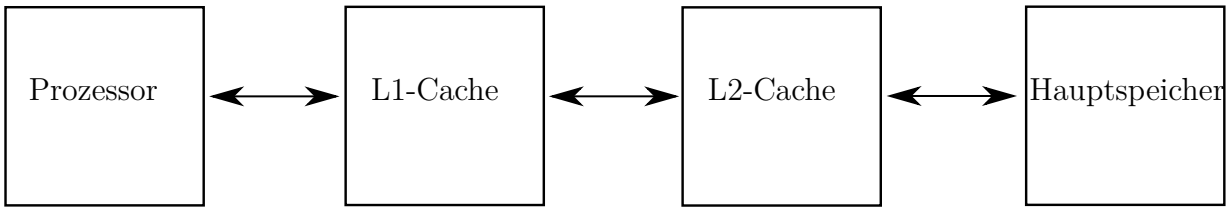
(d) Die *Hit-Rate* ist $6/15 = 40\%$

(e) Es wurden 25% nicht beschrieben.

Aufgabe 5.

Lösung.

(a)



(b)

$$\text{L1-Cache: } 100000 \cdot 0.1 = 10000 \text{ Misses}$$
$$\text{L2-Cache: } 100000 \cdot 0.04 = 4000 \text{ Misses}$$

(c)

$$3 \cdot 10^9/s = 3 \cdot 10^6/ms = 3 \cdot 10^3/\mu s = 3/ns$$

$$t_{L1} = 3/3 = 1ns$$
$$t_{L2} = 45/3 = 15ns$$
$$t_{main} = 150/3 = 50ns$$

(d)

$$t_{eff\_L1} = 0.9 \cdot 3 + 0.1 \cdot 150 = 17.7 \text{ Taktzyklen}$$
$$t_{eff\_L2} = \underbrace{0.9 \cdot 3}_{\text{90\% L1}} + \underbrace{0.1 \cdot 0.96 \cdot 45}_{\text{10\% L2}} + \underbrace{(1 - 0.9 - 0.1 \cdot 0.96) \cdot 150}_{\text{0.04\% Hauptspeicher}} = 7.62 \text{ Taktzyklen}$$
$$17.7 \div 7.62 \approx 2.32$$

Die effektive Speicherzugriffszeit verbessert sich durch den L2-Cache ca. um das 2.32-fache.

Aufgabe 6.

Lösung.

Page-Nr	Frame-Nr	Present-Bit	Zeitpunkt
000	00	0	
001	10	1	t_3
010	00	1	t_1, t_5
011	00	0	
100	11	1	t_4
101	00	0	
110	00	0	
111	01	1	t_2

(a) Virtuelle Adressen sind 20 Bit, physische Adressen 17 bit lang.

(b) Es sind 2^{17} byte physischer Speicher und 2^{20} byte virtueller Speicher adressierbar.

(c)

Page-Nr	Offset
00100	110101000111011

Page-Nummer: 0x04
Offset: 0x6A3B
Frame-Nummer: 0x0
physische Adresse: 0x06A3B

(d)

Page-Nr	Frame-Nr	Present-Bit	Zeitpunkt
000	00	$0 \rightarrow 1$	t_7
001	10	1	t_3
010	00	$1 \rightarrow 0$	t_1, t_5
011	00	0	
100	11	1	t_4
101	00	0	
110	00	$0 \rightarrow 1$	t_6
111	01	$1 \rightarrow 0$	t_2

Aufgabe 7.

Lösung.

(a) Virtuelle Adresse: 3 bit Page-Nummer, 12 bit Offset.
Physische Adresse: 2 bit Frame-Nummer, 12 bit Offset.

Page-Nr	Frame-Nr	Present-Bit	#Zugriffe
000	XX	0	000
001	XX	0	000
010	XX	0	000
011	XX	0	000
100	XX	0	000
101	XX	0	000
110	XX	0	000
111	XX	0	000

(b)

Frame	Frame-Nr	phys. Adressbereich
0	00	0000...0FFF
1	01	1000...1FFF
2	10	2000...2FFF
3	11	3000...3FFF

(c)

Adresse (hex)	Adresse (binär)	Page-Nr
0x4CAD	010|0...	2
0x178A	000|1...	0
0x2431	001|0...	1
0x2B0B	001|0...	1
0x4000	010|0...	2
0x7DEA	011|1...	3
0x6BAC	011|0...	3
0x4FB1	010|0...	2

Reihenfolge der Zugriffe (Page-Nummern): $2 \rightarrow 0 \rightarrow 1 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 2$

- Bei den ersten drei Zugriffen treten *Page-Faults* auf, die Pages werden in die Frames 1, 2 und 3 geladen (Frame 0 durch OS blockiert).
- Page 1 ist bereits geladen, erhöhe Zugriffe auf 2.
- Page 2 ist bereits geladen, erhöhe Zugriffe auf 2.
- Page 3 $\rightarrow$ *Page-Fault* $\rightarrow$ kommt in Frame 2, weil Page 0 niedrigste Zugriffe.
- Page 3 bereits geladen $\rightarrow$ Zugriffe auf 2.
- Page 2 bereits geladen $\rightarrow$ Zugriffe auf 3.

Die Page-Table schaut daher zu Ende so aus:

Page-Nr	Frame-Nr	Present-Bit	#Zugriffe
000	10	$1 \rightarrow 0$	001
001	11	1	$001 \rightarrow 010$
010	01	1	$001 \rightarrow 010 \rightarrow 011$
011	10	$0 \rightarrow 1$	$001 \rightarrow 010$
100	XX	0	000
101	XX	0	000
110	XX	0	000
111	XX	0	000