

Aufgabe 1: Cache-Adressierung

Gegeben sei ein Prozessor mit einer Adresslänge von 12 Bit und einem 2 KiB großem Cache. Die Blockgröße des Caches beträgt 16 Byte. Die Adressierung findet auf Datenwortebene statt, wobei ein Datenwort 32 Bit lang ist.

- a) Geben Sie die Anzahl der Bits an, die für den Index, Tag und Offset benötigt werden, wenn der Cache ein *direct mapped cache* ist.

- b) Geben Sie die Anzahl der Bits an, die für den Index, Tag und Offset benötigt werden, wenn der Cache ein *16-way set associative cache* ist.

- c) Angenommen, die beiden Caches aus den Unteraufgaben a) und b) werden jeweils beim Bearbeiten (Lesen und Schreiben) von vier 1 KiB Daten genutzt, und angenommen, Sie verwenden beide die gleiche Ersetzungsstrategie. Wo könnten Effizienzunterschiede zwischen den zwei Cache-Architekturen liegen wenn die vier Dateien gleichzeitig oder nach einander bearbeitet werden? Begründen Sie Ihre Antwort zumindest mit der Suchzeit und der Hit-Rate.

- d) Warum werden bei der Adressierung des *fully associative caches* keine Index-Bits benötigt? Würden Sie einen *fully associative cache* wählen, wenn Ihnen die Suchzeit oder die Hit-Rate wichtiger ist? Begründen Sie Ihre Antwort.

Aufgabe 2: Cache-Storage

Gegeben ist ein Prozessor mit einer Adresslänge von 8 Bit. Der integrierte *2-way set associative cache* besitzt 4 Sets und eine Blockgröße von 4 Byte. Die kleinste adressierbare Einheit beträgt 1 Byte. Bei einem schreibenden Zugriff wird bei einem Hit das Verfahren *copy-back* verwendet, bei einem Miss das Verfahren *fetch-on-write*.

- a) Wie groß ist der Cache? Geben Sie Ihre Antwort in Byte an.
- b) Nehmen Sie an, dass im Cache bereits folgende Einträge vorhanden sind:

	Block 0							Block 1						
	<i>V</i>	<i>D</i>	<i>Tag</i>	<i>Data</i>				<i>V</i>	<i>D</i>	<i>Tag</i>	<i>Data</i>			
Set 0	1	0	0x1	0x1	0x2	0x3	0x4							
Set 1														
Set 2	1	0	0x8	0xA	0xB	0xC	0xD							
Set 3	1	0	0xF	0x8	0x9	0xE	0xF							

Cache (V:= valid bit, D:= dirty bit)

Auf dem Prozessor werden nun sequenziell folgende Speicherzugriffe ausgeführt ($w := write, r := read$):

r(0x10) - r(0x9B) - r(0x9A) - w(0x94) \leftarrow 0x2 - r(0x97) - w(0x99) \leftarrow 0xF

- (i) Tragen Sie in die obige Cache-Tabelle die resultierenden Verwaltungsinformationen (*V*, *D*, *Tag*) sowie die Daten ein, wie sie **nach** der Zugriffssequenz im Cache vorliegen. Beachten Sie dabei die oben angegebene Strategie (policy) für schreibende Zugriffe.

Data	Address
0x0	0x94
0xF	0x95
0xE	0x96
0xD	0x97
0xC	0x98
0xB	0x99
0xA	0x9A
0x9	0x9B
0x8	0x9C
0x7	0x9D
0x6	0x9E
0x5	0x9F

Auszug aus dem Speicher (vor der obigen Zugriffssequenz)

- (ii) Wurde während der Zugriffssequenz jemals auf den Hauptspeicher geschrieben? Falls ja, wann und an welche Adresse(n)?
- (iii) Welches Datum wurde mit dem Befehl r(0x97) aus dem Cache gelesen?

Aufgabe 3: Cache-Replacement Policy

Gegeben ist ein *fully associative cache*, der aus 4 Blöcken besteht und in einem Block 8 Datenwörter halten kann, wobei ein Datenwort 1 Byte groß ist. Die Adressierung erfolgt auf Datenwortebene. Die Adresslänge beträgt 12 Bit.

Als Ersetzungsstrategie gilt *least recently used* (LRU). Bei schreibenden Zugriffen wird bei einem *Hit* das Verfahren *write-through* (WT) verwendet und bei einem *Miss* das Verfahren *write-around* (WA).

Nehmen Sie an, dass die folgenden Speicherzugriffe t_0 bis t_{11} sequenziell unter der Verwendung des Caches stattfinden:

t_0	r(0xAAA)	t_4	w(0xBA7)	t_8	w(0xABA)
t_1	r(0x123)	t_5	r(0xBA1)	t_9	r(0x50A)
t_2	r(0xA10)	t_6	r(0x051)	t_{10}	w(0x055)
t_3	w(0xAAF)	t_7	r(0xA17)	t_{11}	r(0x50F)

- a) Tragen Sie in die Tabelle die jeweils aktuellen Verwaltungsdaten des Caches zum Zeitpunkt t ein. Geben Sie weiters an, ob es sich um einen lesenden oder einen schreibenden Zugriff handelt (r/w) und ob ein *Hit* oder ein *Miss* vorliegt (h/m). In die Spalte „pol“ tragen Sie die relevante Ersetzungsstrategie ein (LRU, WA, WT).

					Block 0		Block 1		Block 2		Block 3	
t	Adresse	r/w	h/m	pol	Tag	V	Tag	V	Tag	V	Tag	V
0												
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												

r := read, w :=write, h :=hit, m :=miss, V :=valid bit, pol := policy (LRU, WA, WT)

- b) Geben Sie die Hit-Rate des Caches nach der Zugriffssequenz an.

Aufgabe 4: Multi-Core Theorie

Bearbeiten Sie folgende Aufgabenstellungen zum Thema Multi-Core.

- a) Wieso sind spezielle Maschinenbefehle für eine Synchronisierung in einem Multi-Core-System notwendig?

- b) Erklären Sie in Ihren eigenen Worten das Beispiel *Summe von einem Array mit 100000 Zahlen* aus der Vorlesung.

Require: c Cores, CoreID $P_n \in \{0, \dots, c-1\}$, Array A mit zu summierenden Zahlen, Anzahl s der zu summierenden Zahlen

Ensure: None

```
1:  $Sum[P_n] = 0$ ;  
2: for ( $i = s/c * P_n$ ;  $i < s/c * (P_n + 1)$ ;  $i++$ ) do  
3:    $Sum[P_n] = Sum[P_n] + A[i]$ ;  
4: end for  
5:  $half \leftarrow c$   
6: do  
7:    $synch()$ ;  
8:   if ( $mod2(half) \neq 0$ )  $\wedge$  ( $P_n == 0$ ) then  
9:      $Sum[0] = Sum[0] + Sum[half - 1]$ ;  
10:  end if  
11:   $half = half/2$ ;  
12:  if ( $P_n < half$ ) then  
13:     $Sum[P_n] = Sum[P_n] + Sum[P_n + half]$ ;  
14:  end if  
15: while ( $half > 1$ )
```

Aufgabe 5: Multi-core Prozessoren - Ausnutzung

Sie wollen ein bereits existierendes Programm P beschleunigen, indem Sie dieses teilweise parallel ausführen. P hat folgende Eigenschaften:

- 10% von P sind nicht parallelisierbar.
 - 20% von P können auf maximal zwei Cores (Prozessorkerne) parallel ausgeführt werden.
 - Der restliche Teil kann ohne Einschränkungen auf beliebig vielen Cores ausgeführt werden.
- a) Wie viele Cores (Prozessorkerne) sind mindestens notwendig, um einen Speed-Up (Geschwindigkeitszuwachs) von 3 zu erreichen? Die Synchronisierung und Kommunikation zwischen den Cores soll vernachlässigt werden.

b) Wie hoch liegt der theoretisch mögliche Speed-Up?

c) Wie hoch ist die Ausnutzung in Prozent bei Unteraufgabe a)? Runden Sie auf eine ganze Zahl.

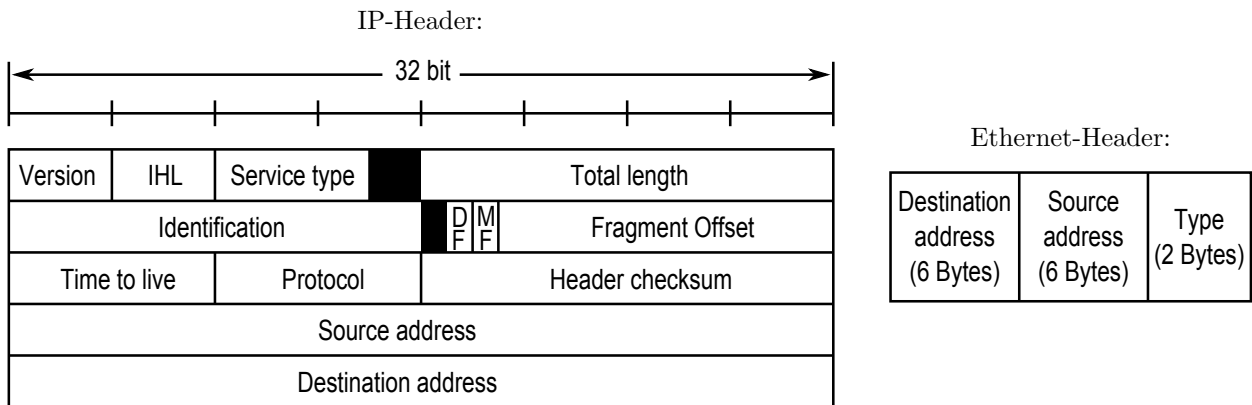
d) Wie können Sie eine Ausnutzung von 100% erreichen?

Aufgabe 6: Ethernet Protokoll

Die nachfolgend angeführte Bitfolge in hexadezimaler Notation stellt ein IP-Paket dar, das über das Ethernet Protokoll transportiert wird:

0057A65C7698 0A0EC667670C 0800 4 5 00 003F 35F0 0000 7F 01 8DF2 80820C6A 80821F01 4E 4749...

Start der Folge ist links. Den genauen Aufbau der Protokoll-Header können Sie den folgenden Grafiken entnehmen:



- Markieren Sie in obiger Bitfolge den Beginn des IP-Headers klar ersichtlich.
- Geben Sie die *time to live* (*TTL*) des IP-Pakets in **hexadezimaler** Notation an.
- Geben Sie die beiden IP-Adressen in **dezimaler** Notation an.
- Wie lautet das erste Byte des mittels IP transportierten *Payloads* in **binärer** Notation?

Aufgabe 7: Netzwerke – Datenkapselung

In einem Netzwerk sollen mittels SCP 10000 Byte Nutzdaten übertragen werden. Sie verwenden folgende Protokolle:

$$TCP - IPv4 - Ethernet - SCP$$

Es gelten folgende Header-Größen: IPv4 20 Byte, TCP 20 Byte, Ethernet 24 Byte, SCP 8 Byte. Die Transportschicht beschränkt die Größe eines übertragenen Pakets (Nutzdaten + Header) auf maximal 1024 Byte. Alle darunterliegenden Schichten unterliegen keiner weiteren Beschränkung.

- a) Ordnen Sie die angegebenen Protokolle den Layern im TCP/IP-Referenzmodell zu. In Klammer sind die jeweils entsprechenden Schichten des OSI-Modells angegeben.

Layer	Protokoll
Application (5-7)	
Transport (4)	
Internet (3)	
Network (1-2)	

- b) In wie viele Einzelpakete muss die Transportschicht das ursprüngliche SCP-Paket aufteilen?

- c) Wie viel Prozent Overhead $\left(= \sum \frac{\text{Größe Header}}{\text{übermittelte Daten}} \right)$ fallen bei der Übertragung an?

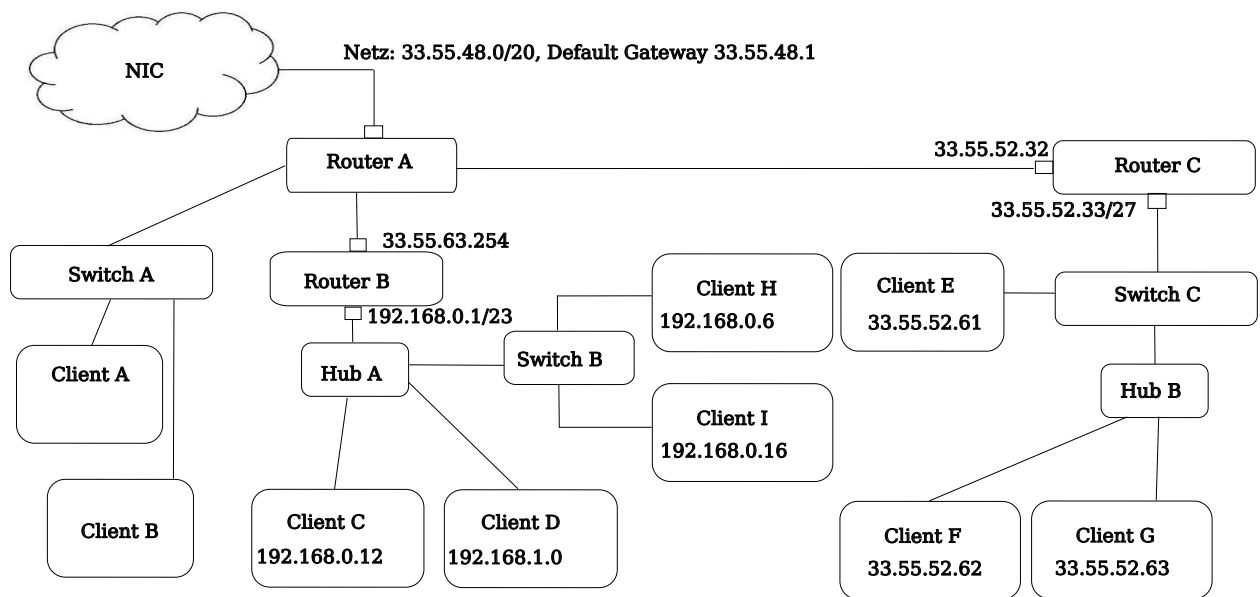
- d) Nehmen Sie an, Sie können die Pakete im Netzwerk mit 128 kb/s übertragen. Wie lange dauert die Übertragung? Wie hoch ist die *Nutzdatenrate*?

Hinweis: Die Nutzdatenrate gibt den Teil der gesamten Datenrate an, der nach Abzug des Overheads noch an tatsächlich nutzbarer Kapazität des Mediums für die eigentlichen Nutzdaten übrig bleibt.

- e) Sie verbinden sich über eine mittels VPN (Virtual Private Network) gesicherte WLAN-Verbindung zum Netzwerk und können die Pakete mit 128 kb/s übertragen. Welche Konsequenzen ergeben sich durch Ihre Verbindung in Bezug auf die Übertragungszeit?

Aufgabe 8: Netzwerke – Analyse

Sie bekommen das folgende Netzwerk vom *Network Information Center* (NIC) zugewiesen:



Beantworten Sie folgende Aufgabenstellungen unter Zuhilfenahme obiger Grafik:

- Geben Sie die Netzadresse und die Broadcast-Adresse Ihres Netzwerks an.
- Geben Sie die Netzadresse und die Broadcast-Adresse des von Router B verwalteten Subnetzes an.
- Von Client F wird ein Paket zu Client B geschickt. An welche Netzwerk-Elemente wird dieses Paket übermittelt?
- Von Client E wird ein Paket zu Client I geschickt. An welche Netzwerk-Elemente wird dieses Paket übermittelt?
- Client H schickt ein Paket an die IP-Adresse 212.164.65.5. An welche Netzwerk-Elemente wird dieses Paket übermittelt?
- Wie viele IP-Adressen stehen noch für weitere Hosts im Teilnetz von Router C zur Verfügung?
- Finden Sie den/die Fehler in der Netzwerkkonfiguration.