

Aufgabe 1: Speicherhierarchie

a)

Der Einsatz von Speicherhierarchien ist für eine schnelle Ausführung Befehlen notwendig, da sich folgender Sachverhalt feststellen lässt. Der (Daten- bzw. Adress-) Bus überträgt die angeforderten Daten langsamer als die CPU arbeitet. Dadurch entsteht ein Engpass, der sogenannte Von-Neumann-Flaschenhals, zwischen Prozessor und (Haupt-) speicher. Diese Übertragungsgeschwindigkeit bestimmt die Gesamtarbeitsgeschwindigkeit. Diesen Effekt kann man durch einen möglichst geringen Datentransport zwischen den beiden Komponenten abgemildert werden. Dies geschieht durch die Nutzung von Speicherhierarchien.

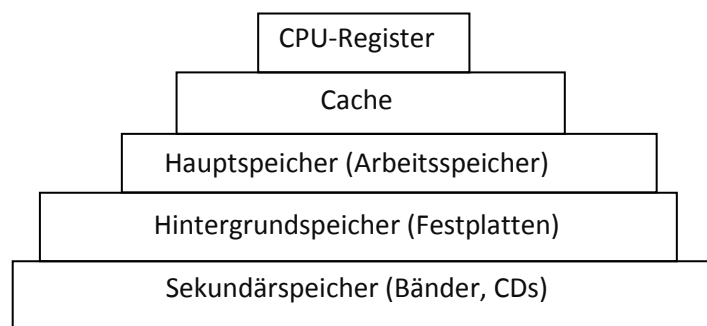
Die Speicherhierarchie ist eine Konsequenz aus dem Lokalitätsverhalten der Programme und der zur Verfügung stehenden Speichertechnologie. Das Prinzip der Lokalität wird auch durch die 90:10-Regel wiedergegeben: „Bei 90% aller Zugriffe werden nur 10% aller Daten, mit denen ein Programm insgesamt arbeitet, benötigt“. Die Lokalität hat 2 Dimensionen:

- Zeitliche Lokalität: Auf dasselbe Wort wird mit großer Wahrscheinlichkeit in kurzer Zeit wieder zugegriffen.
- Räumliche Lokalität: Nachdem ein Wort gelesen wurde, wird mit großer Wahrscheinlichkeit auf das Nachbarwort zugegriffen.

b)

Speicherhierarchien basieren auf dem Prinzip der Anordnung von Arbeits- und Datenspeichern nach Zugriffshäufigkeit, bzw. Wahrscheinlichkeit. Die Daten können mit steigender Zugriffszeit bei gleichzeitiger steigender Speicherkapazität in folgenden (Zwischen-)Speichern abgelegt werden: Register, Cache, Hauptspeicher, Hintergrundspeicher. Je näher ein Speicher an der CPU liegt, desto kürzer ist die Zugriffszeit. Um die Zugriffszeit zu optimieren werden häufig verwendete Daten bzw. deren Kopien möglichst nahe an der CPU zwischengespeichert (Register, Cache). Werden sie länger nicht benötigt so werden sie von den aktuell häufig verwendeten Daten verdrängt und weiter unten in der Hierarchie abgelegt. Daten, die aktuell nicht verwendet werden befinden sich im Hintergrundspeicher (Festplatten, Laufwerke...ect).

Idee: Mit der Geschwindigkeit des schnellen und dem Preis und der Größe des großen und billigen Speichers arbeiten.



c)

Prinzipien bei der Entwicklung von Speicherhierarchien:

- Blockplatzierung: Wo kann ein Block in der oberen Ebene platziert werden?
- Blockidentifizierung: Wie wird ein Block in der oberen Ebene gefunden?
- Blockersetzung: Welcher Block soll bei einem Fehlzugriff ersetzt werden?
- Schreibstrategie: Was geschieht beim Schreiben?

### Aufgabe 2: Auflösung von Digitalkameras

a)

8 Bit = 1 Byte

1024 Byte = 1 KByte

1024 Kbyte = 1 MByte

i)  $(1600 \times 1280) \times 1 \text{ Bit} = 2048000 \text{ Bits} = 256000 \text{ Bytes} = 250 \text{ KB} \approx 0,2441 \text{ MB}$  ✓

ii)  $(1600 \times 1280) \times 8 \text{ Bit} = 16384000 \text{ Bits} = 2048000 \text{ Bytes} = 2000 \text{ KB} \approx 1,953125 \text{ MB}$  ✓

iii)  $(1600 \times 1280) \times 24 \text{ Bit} = 49152000 \text{ Bits} = 6144000 \text{ Bytes} = 6000 \text{ KB} \approx 5,85938 \text{ MB}$  ✓

Achtung: "K" vs. "k"

1 kg = 1000g

1 KB = 1024 Byte

"k" = 1000      „K“ = 1024

b)

Speicherplatzbedarf für ein Video

1 Minute Video: 25 Bilder/s => 1500 Bilder / min

i)  $0,24414062 \times 1500 = 366,212 \text{ MBytes}$

ii)  $1,953125 \times 1500 = 2,86 \text{ GBytes}$

iii)  $5,859375 \times 1500 = 8,58 \text{ GB}$

Der Speicherplatz, den ein einminütiges (Farb-) Video benötigen würde liegt bei 8,5 GB. Dies übersteigt die Speicherkapazität von für die Hobbyfotographie gebräuchlichen Speicherkarten. Auch der Speicherplatz, den Videos mit einer Schwarz-Weiß bzw. Graustufenskala benötigen würden, würde die Arbeit mit derartigen Videos stark erschweren. → Notwendigkeit der Kompression

### Aufgabe 3: Bits und Bytes

a)

i) a) 720 KB

$$= 737280 \text{ Bytes} = 5898240 \text{ Bits} \checkmark$$

$$= 0,703125 \text{ MB} \approx 6,866 * 10^{-4} \text{ GB} \checkmark \approx 6,548 * 10^{-10} \text{ TB} \quad 0,0000006706 \text{ TB}$$

b) 1,44 MB

$$= 1474,58 \text{ MB} = 1509949,44 \text{ Bytes} = 12079595,52 \text{ Bits}$$

$$= 1,4063 * 10^{-3} \text{ GB} = 1,3733 * 10^{-6} \text{ TB} ?$$

ii)a) 700 MB

$$= 716800 \text{ KB} = 734003200 \text{ Bytes} = 5872025600 \text{ Bits}$$

$$= 0,68359375 \text{ GB} = 6,6757 * 10^{-4} \text{ TB} ?$$

ii)b) 4,7 GB

$$= 4812,8 \text{ MB} = 4928307,2 \text{ KB} = 5046586573 \text{ Bytes} = 4,03726 * 10^{10} \text{ Bits}$$

$$= 4,5898 * 10^4 \text{ TB} ?$$

iii)a) 80 GB

$$= 81920 \text{ MB} = 83886080 \text{ KB} \approx 8,5899 * 10^{10} \text{ Bytes} \approx 8,7961 * 10^{13} \text{ Bits}$$

$$= 0,078125 \text{ TB} ?$$

b)

Eine Umrechnung ist aus folgenden Gründen nicht immer sinnvoll:

- Übersichtlichkeit
- Rundungsfehler können u.U. entstehen

Solche Umrechnungen sind in der Regel nicht sinnvoll, da es sich meistens um gerundete Angaben handelt, und evtl. auftretende „gebrochene“ Bits existieren nicht.

### Aufgabe 4: Von-Neumann-Rechnerarchitektur

CPU		Hauptspeicher	Ein- und Ausgabegeräte z.B. Tastatur, Maus, Monitor, Drucker ect...
Rechenwerk	Steuerwerk		

Bus									

1) CPU: Die CPU besteht aus dem Rechenwerk (ALU), das Berechnungen und logische Verknüpfungen durchführt und dem Steuerwerk, das die Ausführung von Befehlen steuert und die Adressen der dazu benötigten Daten verwaltet.

2) Hauptspeicher: Der Hauptspeicher, auch Speicherwerk oder Arbeitsspeicher genannt speichert Daten und Programme (flüchtig!). Das Rechenwerk kann darauf zugreifen. Er besteht i.d.R. aus ROM und RAM.

3) Ein- und Ausgabegeräte: Aufgabe der E/A Geräte ist es eine Verbindung zum Anwender (z.B. über Tastatur, Monitor) herzustellen und Daten zu anderen Systemen zu übermitteln (Modem, Netzwerk)

4) Bus: Die Aufgabe der Busse besteht darin, Daten, Adressen und Steuersignale zwischen CPU und Speicher zu transportieren oder eine Verbindung zwischen Ein- und Ausgabegeräten und Prozessor herzustellen.

Bild (siehe Skript)

CPU	
Datenprozessor	Befehlsprozessor
MR L A	Decodierer + Steuerwerk
Rechenwerk ALU	IR
MBR	MAR
	PC

Arbeitsspeicher:  $2^n$  Zellen m Bits

Adressbus

Datenbus

Datenprozessor: klassisches Verarbeiten von Daten, Ausführen von Berechnungen

- Rechenwerk (ALU)
- Mind. 3 Register zur Aufnahme von Operanden
  - Akkumulator A
  - Multiplikator MR
  - Link-Register L
- MBR (Memory Buffer Register) zur Kommunikation mit dem (Haupt-)speicher
- 

Befehlsprozessor: entschlüsselt Befehle und steuert die Ausführung

- IR (Instruction Register): aktueller Befehl im Befehlsregister
- MAR (Memory Adress Register): Adresse des Speicherplatzes des nächsten Befehls

- PC (Program Counter): nächster Befehl im Befehlszähler
- Entschlüsselung durch den Befehlsdecodierer
- Ausführung durch das Steuerwerk gesteuert

b)

Der Von-Neumann-Flaschenhals entsteht zwischen CPU und Arbeitsspeicher: Da die Busse langsamer arbeiten als die CPU kommt es hier zu einem Engpass und dadurch zu einer Verringerung der Arbeitsgeschwindigkeit.

Früher: CPU hat Daten so langsam verarbeitet, wie Busse geliefert haben. Heute: CPUs schneller, Datentransport über die Busse kommt nicht mehr hinterher.

Lösung: Man versucht das Problem über Caches zu reduzieren.

#### Aufgabe 5: Klassifikation nach Flynn

a)

	# Daten	
# Anweisungen	SISD	SIMD
	MISD	MIMD

SISD: single instruction, single data = klassischer Von-Neumann-Rechner

SIMD: single instruction, multiple data = z.B. Vektorrechner oder Arrayprozessoren

MISD: multiple instruction, single data = Daten/ Befehle werden über mehrere Busse bereitgestellt, aber nur ein Prozessor zur Verarbeitung => umstritten, ob sinnvoll

MIMD: multiple instruction, multiple data = echt parallele Verarbeitung von versch. Daten auf mehreren Prozessoren

b)

SISD: Serieller Rechner, der nach der Von-Neumann-Struktur, oder der Harvard-Struktur aufgebaut sind.

SIMD: Großrechner zur Video, oder Audioverarbeitung, Spracherkennung...ect

MISD:

MIMD: Multiprozessorsysteme

c)

Weitere Klassifikation nach

- Preis und Leistungsfähigkeit

- Maschinenbefehlssatz: RISC – CISC
- Anzahl der Prozessorkerne
- Busbreite