

CNA: 1. Testatprüfung

Was muss ein Rechner können?

Ein Rechner muss einen Algorithmus ausführen können, dazu braucht er:

- Steuerwerk: Befehle eines Programms der Reihe nach ausführen
- Speicher: Zahlen speichern
- Rechenwerk: Speicherinhalt als Zahl interpretieren und manipulieren (rechnen)
- Ablaufsteuerung: auf Inhalt des Speichers reagieren

Was besagt das Mooresche Gesetz?

- Die Anzahl der Transistoren pro Fläche verdoppelt sich alle 18 Monate.
- Die Anzahl der Transistoren pro Fläche steigt um 60% pro Jahr.

Welche speziellen Arten von Prozessoren gibt es?

- Mikrocontroller: Mikroprozessor, Peripheriefunktionen und Speicher auf einem Chip (SoC: System on a Chip)
- DSP: Digitaler Signalprozessor, bearbeitet digitale Signale, z.B. Audio- oder Videosignale
- GPU: Graphics Processing Unit, für rechenintensive 2D- und 3D-Aufgaben
- Krypto-Prozessoren: ver- und entschlüsselt Daten, liegt zwischen CPU und Memory
- Mathematischer Koprozessor, z.B. FPU (Floating Point Unit); heute auf der CPU

Was ist der Unterschied zwischen SRAM und DRAM?

- SRAM: statisches RAM, benötigt 6 Transistoren pro Speicherzelle (Flip-Flop), behält seinen Wert
- DRAM: dynamisches RAM, benötigt 1 Transistor pro Speicherzelle, muss aufgefrischt werden
- PSRAM: DRAM mit eingebauter Auffrischung

Wie sieht die Speicherhierarchie aus?

1. Register (SRAM)
2. Cache (SRAM)
3. Arbeitsspeicher (DRAM)

4. Dateisystem (SSD, HD)
5. Archiv (HD, optische Medien, Magnetbänder)

Wie funktioniert der Fetch/Decode/Execute-Cycle?

1. Fetch: den nächsten Befehl ins Befehlsregister laden; Programmzähler erhöhen
2. Decode: den Befehl dekodieren; ermitteln, welcher Befehl auszuführen ist; zusätzliche Datenwörter aus dem Speicher laden, falls der Befehl diese benötigt
3. Execute: den Befehl ausführen; das Ergebnis im Speicher abspeichern; weiter bei 1.

Woraus besteht eine Von-Neumann-Maschine?

1. Rechenwerk
2. Steuerwerk
3. Speicher
4. Ein- und Ausgabe

Wie können negative Zahlen dargestellt werden?

1. Vorzeichenbehafteter Wert: erstes Bit 0 für positive, 1 für negative Zahlen
2. 1er-Komplement: alle Bits umkehren
3. 2er-Komplement: alle Bits umkehren, eins hinzuaddieren
4. Exzesscode: Versatz um $+n$

Welche Gleitkommazahlen gibt es nach IEEE 754?

- einfache Genauigkeit: 32 Bits (1 Vorzeichen, 8 Exponent, 23 Mantisse)
- doppelte Genauigkeit: 64 Bits (1 Vorzeichen, 11 Exponent, 52 Mantisse)

Wie wird eine Fließkommazahl in IEEE 754 dargestellt?

1. Vorzeichen ermitteln, 1 für negative, 0 für positive Zahl
2. die Zahl durch Multiplikation bzw. Division mit 2^n in das Intervall $[1;2[$ bringen (normalisieren)
3. den (positiven oder negativen!) Exponenten n mit Excess127 normalisieren (127 addieren)
4. von der normalisierten Zahl 1 abziehen (redundant, da immer eine 1 vorne steht)

5. die Mantisse aus der Summe von $1/2 + 1/4 + \dots + 1/2^n$ darstellen und bei den entsprechenden Stellen die Bits auf 1 setzen
6. Vorzeichen, Exponent und Mantisse binär auflisten
7. die Binärzahlen zu je 4 Bits gruppieren
8. die einzelnen Gruppen als hexadezimale Zahl darstellen

Wie ermittelt man eine Fließkommazahl anhand der IEEE-754-Darstellung?

1. jede Ziffer der hexadezimalen Zahl mit vier Bits im Binärcode darstellen
2. die Bitreihe aufteilen
 1. Erstes Bit: Vorzeichen
 2. die nächsten 8 (single) bzw. 11 (double) Bits: Exponent
 3. die letzten 23 (single) bzw. 52 (double) Bits: Mantisse
3. die Mantisse aufsummieren
 1. Erstes Bit = $1/2$
 2. Zweites Bit = $1/4$
 3. n-tes Bit = $1/2^n$
4. die Mantisse mit 1 addieren (bei der Konvertierung weggelassen, da redundant)
5. den Exponent bestimmen und 127 davon subtrahieren (Excess127)
6. den Wert ausrechnen
 1. Mantisse * 2^{Exponent}
 2. Vorzeichen nicht vergessen

Mit welcher Schaltung lassen sich AND, OR und NOT realisieren?

- AND: zwei serielle Schalter
- OR: zwei parallele Schalter
- NOT: ein Öffner

Wie lauten die DeMorganschen Gesetze?

1. $\neg(A \vee B) \equiv \neg A \wedge \neg B$
2. $\neg(A \wedge B) \equiv \neg A \vee \neg B$

Wodurch zeichnet sich die Harvard-Architektur aus?

- Separate Speicher für Daten und Befehle
- Separate Busse zu den beiden Speichern
- Vorteil gegenüber Von-Neumann-Architektur

1. Befehle und Daten können gleichzeitig gelesen werden: Geschwindigkeit
2. Strikte Trennung von Daten und Programmen: Sicherheit
3. Datenwortbreite und Befehlswortbreite sind unabhängig voneinander
4. Synchrones Laden durch mehrerer Rechenwerke

In der Praxis sind oft Mischformen von Harvard- und Von-Neumann-Rechnern zu finden.

Welche Benchmarkprogramme gibt es?

- Linpack: Lineare Gleichungssysteme
- SPEC: Standard Performance Evaluation
- Whetstone: Floating-Point- und Integer-Berechnungen
- Dhrystone: Integer-Berechnungen
- Weitere für PC: 3DMark, Windows-Leistungsindex, Geekbench

In welchen Einheiten wird Computerperformance gemessen?

- MIPS: Million Instructions per Second (unspezifisch)
- Flops (MegaFlops, GigaFlops): Floating Point Operations per Second
- Laufzeit spezifischer Programme/Rechenaufgaben

Wie lauten die wichtigsten Kennwerte der aktuell leistungstärksten Computer (Stand 2016)?

- ca. 10 Millionen Cores
- ca. 100 Petaflops ($100 \cdot 10^{15}$ Flops)
- ca. 15 Megawatt Leistung

Was bedeutet Endian?

- In welcher Reihenfolge die Ziffern einer Grösse aufgelistet werden
 - Big Endian: grosse zuerst
 - * Datumsangabe 2016/10/19 (19. Oktober 2016)
 - * Zahlen in Englisch: 122 one hundred twenty two
 - Little Endian: kleine zuerst
 - * Datumsangabe 19.10.2016 (auch 19. Oktober 2016)
 - * Zweistellige Zahlen in Deutsch: 22 zweiundzwanzig
- In der Informatik bezeichnet Endian die Byte-Reihenfolge im Arbeitsspeicher.
 - Big Endian
 - * UNIX

- * Java
- * Motorola
- * Freescale
- Little Endian
- * Windows
- * Intel

Welche Levels/Stufen gibt es bei den Rechnerarchitekturen?

- Level 5: Problem-oriented language level
 - Translation (compiler)
- Level 4: Assembly language level
 - Translation (assembler)
- Level 3: Operating system machine level
 - Partial interpretation (operating system)
- Level 2: Instruction set architecture level
 - Interpretation (microprogram) or direct execution
- Level 1: Microarchitecture level
 - Hardware
- Level 0: Digital logic level

Welche Operationsarten gibt es?

1. Datentransfer-Operationen
2. Arithmetische und logische Operationen
3. Programmablaufsteuerung

Welche Informationen enthält ein Befehl?

- durchzuführende Operation
- 0, 1 oder n Operanden: Typ, Länge Adressierungsart und Adressen von:
 - erstem Quellenoperand
 - zweitem Quellenoperand
 - Resultat
- Adresse des nächsten Befehls
 - implizit (durch Befehlslänge)
 - explizit (durch bedingten Sprung)

Welche Adressierungsarten gibt es?

- Absolute oder direkte Adressierung: absolute Adresse

- LDA \$0832 (lade den Wert aus Speicherzelle 832)
- Registeradressierung: Name des Registers
 - LDA R1 (lade den Wert aus dem Register 1)
- Unmittelbare Adressierung: Wertangabe
 - LDA #13 (lade den Wert 13)
- Indirekte Adressierung
 - LDA (IX): lade den Wert aus dem Register, dessen Adresse unter “IX” zu finden ist
- Indizierte Adressierung: absolute Adressierung mit Versatz
 - LDA \$0832, 5 (lade den Wert fünf Speicherzellen nach der Adresse 832)

Welche Arten von Befehlen gibt es?

- Einadressbefehle
 - INC \$001: Erhöhe den Wert auf der Speicherzelle 1 um 1
- Zweiadressbefehle
 - ADD \$001, \$002: Addiere die Werte auf den Speicherzellen 1 und 2 und schreibe das Ergebnis auf die Speicherzelle 1 (oder 2)
- Dreiadressbefehle
 - ADD \$001, \$002, \$003: Addiere die Werte auf den Speicherzellen 1 und 2 und schreibe das Ergebnis auf die Speicherzelle 3

Was macht und woraus besteht ein Steuerwerk?

Das Steuerwerk steuert den Ablauf der Befehlsabarbeitung. Es verfügt über:

- einen Program Counter, der auf die nächste Instruktion zeigt
- einen Instruktionsregister
- einen Adressregister
- einen Stackpointer

Wie funktioniert ein Stack?

- Der Stack ist ein Stapelspeicher, die Daten werden darauf “gestapelt”.
- Es kann nur immer auf das zuoberst gespeicherte Datenelement zugegriffen werden.
- FIFO: first in, first out
- LIFO: last in, last out
- Der Stack Pointer (Stapelzeiger) zeigt immer auf den obersten Eintrag
- Befehle
 - push: Daten auf den Stack schreiben (obendrauf legen)
 - pop: Daten vom Stack auslesen (wegnehmen)

Wozu wird ein Stack gebraucht?

- Zur Ausführung von Unterprogrammen
 - Parameterübergabe
 - Speicherung der Rücksprungadresse
 - Ablage des Rückgabewertes
- Als Zwischenspeicher
- Zur Interrupt-Behandlung

Was versteht man unter dem Semantic Gap?

- Die Kluft zwischen verschiedenen Sprachen (der Unterschied ihrer Ausdrucksstärken)
 - natürliche Sprache: “die Zahl x um drei Erhöhen und um zwei reduzieren”
 - mathematische Notation: $x + 3 - 2$
 - Programmiersprache (Java): $x = x + 3 - 2$;
 - Maschinensprache
 - * ADD &x, 3
 - * SUB &x, 2
- Hochsprachen wie Java versuchen den Semantic Gap zu schliessen.

Wie lauten die in der Informatik gebräuchlichsten SI-Vorsätze?

- kleiner als 1:
 - 10^{-3} : milli, m
 - 10^{-6} : micro, μ
 - 10^{-9} : nano, n
- grösser als 1:
 - 10^3 : kilo, k
 - 10^6 : mega, M
 - 10^9 : giga, G
 - 10^{12} : tera, T
 - 10^{15} : peta, P
 - 10^{18} : exa, E

Was ist ein PC-Chipsatz?

- Er unterstützt den Prozessor bei seinen Aufgaben.
- Er realisiert die elektrischen Anschlüsse (Pins, Schnittstellen)
- Er besteht aus:

- North-Bridge/MCH (Memory Controller Hub): Steuert Datenfluss zwischen CPU, Speicher und South-Bridge
- South-Bridge/ICH (I/O Controller Hub): Steuert Datenfluss zwischen Peripherie, PCI-Bus, Festplatten und externen Schnittstellen und der North-Bridge

Welche RAM-Busse gibt es?

- SDRAM: synchrones DRAM
- DDR RAM: double data rate
 - DDR II RAM: vierfach Fetch
 - DDR III RAM: achtfach Fetch

Welche RAM-Modul-Bauformen gibt es?

- SIMM: Single Inline Memory Module (ältere RAM-Bausteine)
- DIMM: Dual Inline Memory Module (modernere RAM-Bausteine für PC)
- SO-DIMM: Small Outline DIMM (für Laptops)

Was sind die Vorteile von RAID?

- Redundante Abspeicherung: ermöglicht Datenrettung im Falle kaputter Festplatten
- Ausfallsicherheit: das System läuft weiter im Falle einer kaputten Festplatte
- Realisierung extrem grosser virtueller Laufwerke aus mehreren Festplatten
- Fehlererkennung