PROFESSOR ERKLÄRBÄR

Serie Informatik · Folge 5 · Advanced

INFORMATIK

LEHRERINFO ZUM VIDEO

Rechnerarchitektur

FACH	STUFE	DAUER	VIDEO
Informatik	Gymnasium · Sek II	13 min	youtube.com/@Erklaerbaer

01 Lernziele

Nach dem Video können die Schülerinnen und Schüler …

- die fünf Bestandteile der Von-Neumann-Architektur nennen und erklären, warum «Programm und Daten im selben Speicher» den Computer zur Universalmaschine macht,
- ALU, Steuerwerk und die Register Akku, Befehlszähler, Befehlsregister und Statusflags unterscheiden,
- Daten-, Adress- und Steuerbus unterscheiden und aus der Breite des Adressbusses den adressierbaren Speicher berechnen,
- den Befehlszyklus (holen, decodieren, ausführen) an einem kleinen Maschinenprogramm Schritt für Schritt durchspielen,
- die Speicherhierarchie mit Zugriffszeiten und dem Lokalitätsprinzip begründen,
- Von-Neumann-Flaschenhals, Harvard-Architektur, Pipeline, Mehrkern sowie RISC und CISC einordnen.

02 Voraussetzungen

Bauteile eines Computers, Prozessor, Arbeitsspeicher und Mainboard (Folge 5 · Grundlagen); Addierer und Flipflops (Folge 4 · Advanced); Hexadezimalzahlen und Zweierpotenzen (Folge 2 · Advanced).

03 Ablauf des Videos

ZEIT	KAPITEL	INHALT UND BEISPIEL
0:00	Einstieg	Bei 3 GHz kommt Licht pro Takt nur ≈ 10 cm weit · Ident · Fahrplan
0:39	Was du brauchst	Prozessor, Arbeitsspeicher, Busse · Addierer, Flipflops, Hex · Hinweis auf 05G
1:06	Die Von-Neumann-Architektur	Fünf Teile · Programm und Daten im selben Speicher · Universalmaschine · auch das Handy

1:56	Im Prozessor	ALU, Akku, Befehlszähler, Befehlsregister, Statusflags · Vergleich über das Null-Flag · Steuerwerk als Stellwerk
3:04	Bussysteme	Daten-, Adress-, Steuerbus · 16 Leitungen → 64 KiB, 32 → 4 GiB, 64 → 16 EiB · in der Praxis 48 Bit → 256 TiB
4:16	Der Befehlszyklus	Holen, decodieren, ausführen · Mini-Programm 7 + 5 = 12 · Sprungbefehl · Befehl oder Zahl? PAUSE-FRAGE Akku und Befehlszähler
6:06	Speicherhierarchie	Register bis Festplatte · Cache 1 s → RAM 1,5 min → SSD 1 Tag → HDD 4 Monate · Lokalität · Cache-Grössen im Datenblatt
7:08	Schneller werden	Von-Neumann-Flaschenhals · Harvard (Arduino) · Pipeline 400 gegen 103 Takte · Sprungvorhersage · Kerne, RISC/CISC, GPU · CPU- und GPU-Limit
8:48	Projekt-Baustein	Mini-CPU: Befehlszähler, Befehlsregister, Akku, 16 Byte RAM · erstes Programm 3 · 4 = 4 + 4 + 4
9:20	Prüfungsfallen	fünf Fallen (siehe 04)
10:11	Zusammenfassung	Kernaussagen · zurück zu den 10 cm Licht pro Takt
10:47	Quiz	4 Fragen · Endkarte Folge 6 Advanced

04 Prüfungsfallen

- Der Befehlszähler zählt die ausgeführten Befehle – er enthält die Adresse des nächsten Befehls; ein Sprung setzt ihn neu.
- Die Breite des Datenbusses bestimmt die Speichergrösse – das tut der Adressbus (n Leitungen → 2^n Adressen).
- Der Cache ist ein Teil des Arbeitsspeichers – er ist ein kleiner, schneller Zwischenspeicher im Prozessor.
- Doppelt so viele Kerne machen alles doppelt so schnell – nur, wenn sich die Aufgabe aufteilen lässt.
- Bei Von Neumann sind Programm und Daten getrennt – gerade nicht; getrennt sind sie bei der Harvard-Architektur.

05 Einsatz im Unterricht

Einstieg: «Warum hat ein Prozessor eigenen Speicher, wenn es doch den Arbeitsspeicher gibt?» – Vermutungen sammeln, dann Einstieg schauen.

Rollenspiel: Schülerinnen und Schüler spielen Befehlszähler, Steuerwerk, ALU und Speicher und führen das Programm aus Aufgabe 3 «von Hand» aus.

Simulator: Das Mini-Programm aus Kapitel 5 im Johnny-Simulator oder im Little Man Computer nachbauen und Takt für Takt beobachten (kostenlos im Browser).

Prüfungsvorbereitung: Aufgaben 2, 3 und 4 sind im Stil schriftlicher Prüfungen gestellt.

Hinweis: Die Zeitvergleiche in Kapitel 6 skalieren einen Cache-Zugriff von ≈ 1 ns auf 1 s; reale Werte schwanken je nach System (Arbeitsspeicher ≈ 80–100 ns, SSD ≈ 100 µs, Festplatte ≈ 10 ms). Die Cache-Grössen sind ein Beispiel eines aktuellen Desktop-Prozessors. Die Mini-CPU des Videos ist ein vereinfachtes Akkumulator-Modell; heutige Prozessoren haben viele Register und deutlich mehr Pipeline-Stufen.

06 Quellen

- John von Neumann: „First Draft of a Report on the EDVAC“ (1945)
- David A. Patterson, John L. Hennessy: „Computer Organization and Design – The Hardware/Software Interface“, RISC-V Edition, 2. Aufl., Morgan Kaufmann (2020)
- Andrew S. Tanenbaum, Todd Austin: „Structured Computer Organization“, 6. Aufl., Pearson (2013)
- Intel 64 and IA-32 Architectures Software Developer's Manual, Vol. 1 und 3A (Adressierung mit 48 Bit, 4-Level-Paging)
- Jeff Dean / Peter Norvig: „Latency Numbers Every Programmer Should Know“ (Grössenordnungen der Zugriffszeiten)
- Gesellschaft für Informatik: Bildungsstandards Informatik für die Sekundarstufe II (2016)

NAME _____ KLASSE _____ DATUM _____

Rechnerarchitektur

1 Von-Neumann-Architektur ●○○

Nenne die fünf Bestandteile der Von-Neumann-Architektur und **erkläre**, warum man für ein neues Programm keine Kabel umstecken muss.

2 Busbreiten ●●○

a) **Berechne**, wie viele Bytes ein Prozessor mit 20 Adressleitungen erreichen kann (eine Adresse pro Byte). b) **Bestimme**, wie viele Adressleitungen für 4 GiB nötig sind. c) Wie viele Takte braucht ein Datenbus mit 8 Bit bzw. mit 32 Bit, um 16 Byte zu übertragen?

3 Befehlszyklus von Hand ●●○

Im Speicher steht: Zelle 0: LADE 8 · Zelle 1: ADDIERE 9 · Zelle 2: SPEICHERE 10 · Zelle 3: HALT · Zelle 8: 3 · Zelle 9: 4. **Gib** nach jedem Befehl den Inhalt von Akku, Befehlszähler und Zelle 10 **an**. **Beschreibe**, was passiert, wenn in Zelle 3 stattdessen SPRINGE 1 steht.

NACH BEFEHL IN ZELLE	AKKU	BEFEHLSZÄHLER	ZELLE 10
0			
1			
2			

4 Pipeline ●●○

Ein Prozessor hat eine Pipeline mit 5 Stufen, jede Stufe dauert einen Takt. **Berechne** die Takte für 200 Befehle ohne und mit Pipeline und den Faktor, um den die Pipeline schneller ist. **Erkläre**, warum Sprünge die Pipeline ausbremsen.

5 Speicherhierarchie ●●○

Ordne Festplatte, Register, Arbeitsspeicher, Cache 1, SSD und Cache 3 nach Zugriffszeit. **Begründe** mit dem Lokalitätsprinzip, warum eine Schleife, die eine Liste von vorne nach hinten durchläuft, besonders vom Cache profitiert.

6 Multiplizieren ohne Multiplikationsbefehl ●●●

Die Mini-CPU kennt LADE a, ADDIERE a, SUBTRAHIERE a, SPEICHERE a, SPRINGE a, SPRINGE_WENN_NULL a (springt, wenn der Akku null ist) und HALT. In Zelle 12 steht 0 (Ergebnis), in Zelle 13 die 4, in Zelle 14 die 3 (Zähler), in Zelle 15 die 1. **Schreibe** ein Programm ab Zelle 0, das 3 · 4 berechnet und das Ergebnis in Zelle 12 ablegt.

Lösungen

1 Von-Neumann-Architektur

Rechenwerk (ALU), Steuerwerk, Speicher, Ein- und Ausgabe, Bus. Programm und Daten liegen im selben Speicher, ein Befehl ist auch nur eine Zahl: Für eine neue Aufgabe lädt man ein anderes Programm in den Speicher – die Hardware bleibt gleich (Universalmaschine).

2 Busbreiten

a) 2^{20} = 1 048 576 Byte = 1 MiB. b) 4 GiB = $2^2 \cdot 2^{30} = 2^{32}$ Byte → 32 Leitungen. c) 8 Bit: 16 Takte; 32 Bit = 4 Byte pro Takt: 4 Takte.

3 Befehlszyklus von Hand

Nach Zelle 0: Akku 3, Befehlszähler 1, Zelle 10 leer · nach Zelle 1: Akku 7, Befehlszähler 2 · nach Zelle 2: Akku 7, Befehlszähler 3, Zelle 10 = 7 · HALT stoppt. Mit SPRINGE 1 entsteht eine Endlosschleife: Der Akku wächst bei jedem Durchlauf um 4 (7, 11, 15, …), und Zelle 10 wird jedes Mal überschrieben – bis die 8 Bit überlaufen.

4 Pipeline

Ohne Pipeline: 5 · 200 = 1000 Takte. Mit Pipeline: 5 + 199 = 204 Takte (die ersten 5 Takte füllen die Pipeline, danach wird jeden Takt ein Befehl fertig). Faktor 1000 : 204 ≈ 4,9. Bei einem Sprung stehen schon die falschen Befehle in der Pipeline; sie müssen verworfen werden, die Pipeline läuft neu an. Darum versuchen Prozessoren, Sprünge vorherzusagen.

5 Speicherhierarchie

Register → Cache 1 → Cache 3 → Arbeitsspeicher → SSD → Festplatte (von schnell nach langsam). Eine Schleife benutzt immer wieder dieselben Befehle (zeitliche Lokalität) und liest Daten, die direkt nebeneinander liegen (räumliche Lokalität); der Cache lädt immer einen ganzen Block, darum liegen die nächsten Elemente schon bereit.

6 Multiplizieren ohne Multiplikationsbefehl

0: LADE 14 · 1: SPRINGE_WENN_NULL 8 · 2: SUBTRAHIERE 15 · 3: SPEICHERE 14 · 4: LADE 12 · 5: ADDIERE 13 · 6: SPEICHERE 12 · 7: SPRINGE 0 · 8: HALT. Die Schleife läuft dreimal; in Zelle 12 stehen danach 0 + 4 + 4 + 4 = 12. Das Programm (9 Zellen) und die Daten (4 Zellen) passen zusammen in die 16 Byte der Mini-CPU – Befehle und Daten im selben Speicher.

Q Quiz im Video

1 B (Adresse des nächsten Befehls) · 2 C (Adressbus) · 3 A (holen, decodieren, ausführen) · 4 B (ein gemeinsamer Bus für Befehle und Daten)