

ERA-Tutorium 7

Thomas Kilian

Organisatorisches

- Gibt es Fragen zum letzten Tutorium?
- Wünsche und Anregungen?

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur



Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Bauelemente-Schicht
physikalische Schicht

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Gatter-Schicht
Bauelemente-Schicht
physikalische Schicht

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Mikroarchitektur-Schicht
Gatter-Schicht
Bauelemente-Schicht
physikalische Schicht

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

von-Neumann-Schicht
Mikroarchitektur-Schicht
Gatter-Schicht
Bauelemente-Schicht
physikalische Schicht

Frage zur Vorlesung

- Nennen Sie die 6 funktionalen Schichten der von-Neumann-Architektur

Benutzerprogramm-Schicht
von-Neumann-Schicht
Mikroarchitektur-Schicht
Gatter-Schicht
Bauelemente-Schicht
physikalische Schicht

Diskussion

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk
 - ➔ Mikroprozessor, Coprozessoren, Grafikkarte

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk
 - ➔ Mikroprozessor, Coprozessoren, Grafikkarte
 - Leitwerk

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk
 - ➔ Mikroprozessor, Coprozessoren, Grafikkarte
 - Leitwerk
 - ➔ Mikroprozessor

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk
 - ➔ Mikroprozessor, Coprozessoren, Grafikkarte
 - Leitwerk
 - ➔ Mikroprozessor
 - E/A-Werk

Diskussion

- Welche Komponenten eines heutigen PCs gehören zum
 - Speicherwerk
 - ➔ Hauptspeicher, Hintergrundspeicher, Caches
 - Rechenwerk
 - ➔ Mikroprozessor, Coprozessoren, Grafikkarte
 - Leitwerk
 - ➔ Mikroprozessor
 - E/A-Werk
 - ➔ Busse, IO Devices etc.

Diskussion

Diskussion

- Welche Komponenten lassen sich nicht einordnen?

Diskussion

- Welche Komponenten lassen sich nicht einordnen?
 - Mehrprozessorsysteme

Diskussion

- Welche Komponenten lassen sich nicht einordnen?
 - Mehrprozessorsysteme
 - ➔ von-Neumann-Architektur sieht das nicht vor

Diskussion

Diskussion

- Warum stehen Programm und Daten bei einer von-Neumann-Maschine im selben Speicher?

Diskussion

- Warum stehen Programm und Daten bei einer von-Neumann-Maschine im selben Speicher?
 - ➔ effiziente Speichernutzung

Diskussion

- Warum stehen Programm und Daten bei einer von-Neumann-Maschine im selben Speicher?
 - ➔ effiziente Speichernutzung
 - ➔ ein Programm als Datum für anderes Programm (Übersetzer)

Diskussion

- Warum stehen Programm und Daten bei einer von-Neumann-Maschine im selben Speicher?
 - ➔ effiziente Speichernutzung
 - ➔ ein Programm als Datum für anderes Programm (Übersetzer)
 - ➔ selbstmodifizierende Programme (JIT-Compiler)

Aufgabe 2

Aufgabe 2

- Beschreiben Sie eine geeignete Ausgabefunktion $\lambda!$

Aufgabe 2

- Beschreiben Sie eine geeignete Ausgabefunktion λ !
- $\lambda(z_1) = B_1$

Aufgabe 2

- Beschreiben Sie eine geeignete Ausgabefunktion λ !
 - $\lambda(z1) = B1$
 - $\lambda(z2) = B2$

Aufgabe 2

- Beschreiben Sie eine geeignete Ausgabefunktion λ !
 - $\lambda(z1) = B1$
 - $\lambda(z2) = B2$
 - $\lambda(z3) = \text{nichts}$

Aufgabe 2

Aufgabe 2

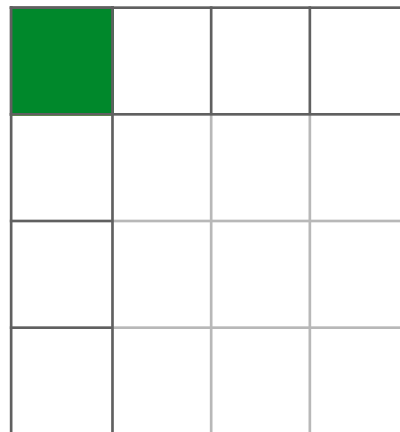
- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .



Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K			

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2		

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G			

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1		

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	z1

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	z1
!G			

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	z1
!G	z1		

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	z1
!G	z1	z1	

Aufgabe 2

- Beschreiben Sie, z.B. in Form einer Tabelle, eine geeignete Zustandsübergangsfunktion δ .

	z1	z2	z3
K	z2	z2	z1
G	z1	z3	z1
!G	z1	z1	z1

MI-Maschine

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel `ADD EAX, 0x1234`

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel ADD EAX, 0x1234
 - IFETCH (lade Instruktion aus Speicher)

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel ADD EAX, 0x1234
 - IFETCH (lade Instruktion aus Speicher)
 - Hole *imm* aus Speicher

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel ADD EAX, 0x1234
 - IFETCH (lade Instruktion aus Speicher)
 - Hole *imm* aus Speicher
 - Addiere

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel ADD EAX, 0x1234
 - IFETCH (lade Instruktion aus Speicher)
 - Hole *imm* aus Speicher
 - Addiere
 - Lade Ergebnis nach EAX

MI-Maschine

- Eine Mikroarchitektur ermöglicht es uns, komplexe Assemblerbefehle in kleinere Mikroprogramme aufzuteilen
- Beispiel ADD EAX, 0x1234
 - IFETCH (lade Instruktion aus Speicher)
 - Hole *imm* aus Speicher
 - Addiere
 - Lade Ergebnis nach EAX
 - Inkrementiere Befehlszähler

MI-Maschine

MI-Maschine

- Befehlszyklus allgemein:

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode
 5. Führe das zum Opcode gehörende Mikroprogramm aus

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode
 5. Führe das zum Opcode gehörende Mikroprogramm aus
 6. Springe zu 1

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode
 5. Führe das zum Opcode gehörende Mikroprogramm aus
 6. Springe zu 1
- 1 bis 3 erledigt IFETCH

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode
 5. Führe das zum Opcode gehörende Mikroprogramm aus
 6. Springe zu 1
- 1 bis 3 erledigt IFETCH
- 4 passiert automatisch (Mapping-PROM)

MI-Maschine

- Befehlszyklus allgemein:
 1. Lege Befehlszähler auf Adressbus
 2. Lade Instruktion (vom Datenbus)
 3. Inkrementiere Befehlszähler
 4. Dekodiere Opcode
 5. Führe das zum Opcode gehörende Mikroprogramm aus
 6. Springe zu 1
- 1 bis 3 erledigt IFETCH
- 4 passiert automatisch (Mapping-PROM)
- 5 ist unsere Aufgabe

MI-Maschine

MI-Maschine

- keine x86-Architektur!

MI-Maschine

- keine x86-Architektur!
- Speicher

MI-Maschine

- keine x86-Architektur!
- Speicher
 - besteht aus linear adressierbaren 16-Bit-Speicherzellen

MI-Maschine

- keine x86-Architektur!
- Speicher
 - besteht aus linear adressierbaren 16-Bit-Speicherzellen
 - Speicheradressen ebenfalls 16 Bit

MI-Maschine

- keine x86-Architektur!
- Speicher
 - besteht aus linear adressierbaren 16-Bit-Speicherzellen
 - Speicheradressen ebenfalls 16 Bit
- 16 Register mit je 16 Bit: r0 bis r15

MI-Maschine

- keine x86-Architektur!
- Speicher
 - besteht aus linear adressierbaren 16-Bit-Speicherzellen
 - Speicheradressen ebenfalls 16 Bit
- 16 Register mit je 16 Bit: r0 bis r15
 - MI-Programmierer hat Zugriff auf alle Register

MI-Maschine

- keine x86-Architektur!
- Speicher
 - besteht aus linear adressierbaren 16-Bit-Speicherzellen
 - Speicheradressen ebenfalls 16 Bit
- 16 Register mit je 16 Bit: r0 bis r15
 - MI-Programmierer hat Zugriff auf alle Register
 - Assembler-Programmierer hat Zugriff auf r0 bis r7

Aufgabe 3

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle
 - ➔ CMP, ADD, DEC

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle
 - ➔ CMP, ADD, DEC
 - Speicherzugriffsbefehle

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle
 - ➔ CMP, ADD, DEC
 - Speicherzugriffsbefehle
 - ➔ beiden MOVs

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle
 - ➔ CMP, ADD, DEC
 - Speicherzugriffsbefehle
 - ➔ beiden MOVs
 - Sprungbefehle

Aufgabe 3

- Ordnen Sie diese Befehle den folgenden Gruppen zu:
 - arithmetische Befehle
 - ➔ CMP, ADD, DEC
 - Speicherzugriffsbefehle
 - ➔ beiden MOVs
 - Sprungbefehle
 - ➔ JZ, JMP

Aufgabe 3

Aufgabe 3

- Warum belegen einige Befehle im Hauptspeicher 16 Bit (eine Speicherzelle), andere dagegen 32-Bit (2 Speicherzellen)?

Aufgabe 3

Adresse	Inhalt
1A35	...
1A36	0E02
1A37	0000
1A38	0100
1A39	A100
1A3A	1A3F
1A3B	0512
1A3C	0600
1A3D	A200
1A3E	1A38
1A3F	...

Aufgabe 3

- Disassemblieren des folgenden Maschinenprogramms

Adresse	Inhalt
1A35	...
1A36	0E02
1A37	0000
1A38	0100
1A39	A100
1A3A	1A3F
1A3B	0512
1A3C	0600
1A3D	A200
1A3E	1A38
1A3F	...

Aufgabe 3

Aufgabe 3

- Welche Wirkung hat dieses Maschinenprogramm?

Aufgabe 3

- Welche Wirkung hat dieses Maschinenprogramm?

$$\Rightarrow r2 = r1 * r0$$

Aufgabe 3

Aufgabe 3

- Welche Teilschritte können gleichzeitig ausgeführt werden?