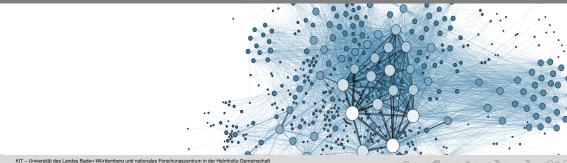




Grundbegriffe der Informatik **Tutorium 33**

Maximilian Staab, maximilian.staab@fsmi.uni-karlsruhe.de Lukas Bach, lukas.bach@student.kit.edu | 1.12.2016



Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

Zum Übungsblatt

MIMA

Maschinenbefehle

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

Zum Übungsblatt

■ N

MIMA

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

Zum Übungsblatt

N = Menge der natürlichen Zahlen (1, 2, 3, ...)

MIMA

Maschinenbefehle

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

lacksquare \mathbb{N}_0

MIMA

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

 $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$

MIMA

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas bach@student kit edu

Zum Übungsblatt

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

 $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$

MIMA

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

lacktriangle = Menge der Reellen Zahlen

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

 $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$

lacktriangle = Menge der Reellen Zahlen

R⁺

Zum Übungsblatt

MIMA

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

■ \mathbb{N} = Menge der natürlichen Zahlen (1, 2, 3, ...)

lacktriangle = Menge der Reellen Zahlen

Arr = Menge der positiven reellen Zahlen

Zum Übungsblatt

MIMA

Maschinenbefehl

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

■ \mathbb{N} = Menge der natürlichen Zahlen (1, 2, 3, ...)

lacktriangle = Menge der Reellen Zahlen

Arr = Menge der positiven reellen Zahlen

lacksquare

Zum Übungsblatt

IVIIIVIA

Maschinenbefeh

Anmerkungen zum letzten Übungsblatt



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

 \blacksquare R = Menge der Reellen Zahlen

Arr = Menge der positiven reellen Zahlen

 \blacksquare \mathbb{R}_0 gibt es nicht! 0 ist auch so schon in \mathbb{R}

Zum Übungsblatt

IVIIIVIA

Maschinenbeteni

Anmerkungen zum letzten Übungsblatt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Was ist sind die folgenden Mengen?

■ \mathbb{N} = Menge der natürlichen Zahlen (1, 2, 3, ...)

 $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$

 \blacksquare R = Menge der Reellen Zahlen

Arr = Menge der positiven reellen Zahlen

lacksquare \mathbb{R}_0 gibt es nicht! 0 ist auch so schon in \mathbb{R}

 $ightharpoonup \mathbb{R}_0^+$ genauso nicht!

Zum Übungsblatt

IVIIIVIA

Maschinenbeien

Anmerkungen zum letzten Übungsblatt



Maximilian Staab.

Aufgaben

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

Was ist sind die folgenden Mengen?

■ N = Menge der natürlichen Zahlen (1, 2, 3, ...)

 \blacksquare R = Menge der Reellen Zahlen

Arr = Menge der positiven reellen Zahlen

 \blacksquare \mathbb{R}_0 gibt es nicht! 0 ist auch so schon in \mathbb{R}

 \blacksquare \mathbb{R}_0^+ genauso nicht!

Aufgabe: R : A* → A*

• $R(\varepsilon) = \varepsilon$

 $\forall x \in A : R(x) = x$

 $\forall w \in A^* \forall x \in A \forall y \in A : R(xwy) = yR(w)x$

■ Zeige: $\forall n \in \mathbb{N}_0 : \forall w \in A^n : |R(w)| = |w|$

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

■ Theoretischer, idealisierter Prozessor

Zum Übungsblatt

MIMA

Maschinenbefehle

Was ist die MIMA?



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Maschinenbefehle

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Nah an Technischer Informatik

Maschinenbetehl

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Nah an Technischer Informatik

Maschinenbeten

Grundaufbau:

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Nah an Technischer Informatik

iviaschinenbeieni

Grundaufbau:

Aufgaben

Adressen als 20bit Datenwort

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Nah an Technischer Informatik

Maschinenbeten

Grundaufbau:

- Adressen als 20bit Datenwort
- Speicherworte als 24bit Datenwort

Was ist die MIMA?



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Theoretischer, idealisierter Prozessor

Zum Übungsblatt

Funktioniert wie ein echter Prozessor, ist aber simpler

MIMA

Nah an Technischer Informatik

iviaschinenbeten

Grundaufbau:

- Adressen als 20bit Datenwort
- Speicherworte als 24bit Datenwort
- Maschinenbefehle als...
 - 4bit Befehl und 20bit Adresse
 - oder 8bit Befehl und unwichtigem Rest

Aufbau der MIMA: Steuerwerk



Maximilian Staab,

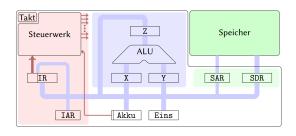
maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle



Aufbau der MIMA: Steuerwerk



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

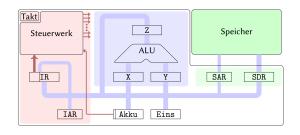
Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

Steuerwerk



Aufbau der MIMA: Steuerwerk



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

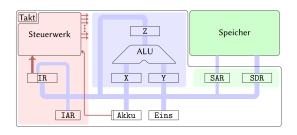
MIMA

Maschinenbefehl

Aufgaben

Steuerwerk

 Instruction Register (IR) enthält den nächsten auszuführenden Befehl



Aufbau der MIMA: Steuerwerk



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

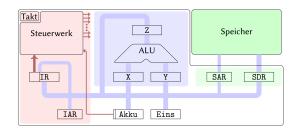
MIMA

Maschinenbefehl

Aufgaben

Steuerwerk

- Instruction Register (IR) enthält den nächsten auszuführenden Befehl
- Instruction Adress Register (IAR) enthält die Adresse des nächsten Befehls



Aufbau der MIMA: Steuerwerk

Steuerwerk

TR.

IAR

Takt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben

Steuerwerk

- Instruction Register (IR) enthält den nächsten auszuführenden Befehl
- Instruction Adress Register (IAR) enthält die Adresse des nächsten Befehls

 Takt bestimmt die "Tickrate", also die Geschwindigkeit

SDR

Speicher

SAR

ALU

Eins

Akku



Aufbau der MIMA: Steuerwerk

Steuerwerk

TR.

IAR

Takt



Maximilian Staab

maximilian.staab@fsmi.uni-karlsruhe.de. Lukas Bach.

lukas bach@student kit edu

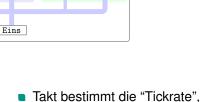
Zum Übungsblatt

MIMA

Aufgaben

Steuerwerk

- Instruction Register (IR) enthält den nächsten auszuführenden Befehl
- Instruction Adress Register (IAR) enthält die Adresse des nächsten Befehls



SDR

Speicher

SAR

ALU

Akku

Steuerwerk interpretiert alle Befehle und führt sie aus

also die Geschwindigkeit

Aufbau der MIMA: Steuerwerk

Steuerwerk

TR.

IAR

Z

ALU

Akku

Takt



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

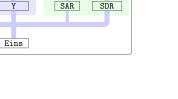
MIMA

Maschinenbefehl

Aufgaben

Steuerwerk

- Instruction Register (IR) enthält den nächsten auszuführenden Befehl
- Instruction Adress Register (IAR) enthält die Adresse des nächsten Befehls



Speicher

- Takt bestimmt die "Tickrate", also die Geschwindigkeit
- Steuerwerk interpretiert alle Befehle und führt sie aus
- Welche Befehle es gibt: Siehe später

Aufbau der MIMA: Akku und Eins



Maximilian Staab,

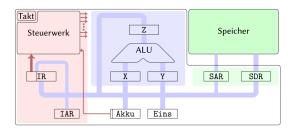
maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle



Aufbau der MIMA: Akku und Eins



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

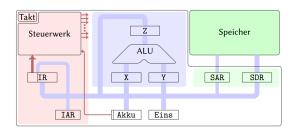
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Akku und Eins

Aufbau der MIMA: Akku und Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

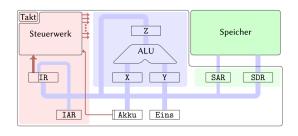
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben



Akku und Eins

 Akku dient als Zwischenspeicher für Datenworte

Aufbau der MIMA: Akku und Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

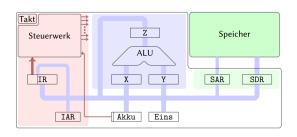
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Akku und Eins

- Akku dient als Zwischenspeicher für Datenworte
- Hält maximal ein Wort

Aufbau der MIMA: Akku und Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

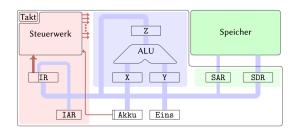
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben



Akku und Eins

- Akku dient als Zwischenspeicher für Datenworte
- Hält maximal ein Wort

 Eins liefert die Konstante 1, hält also Strom

Aufbau der MIMA: Akku und Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

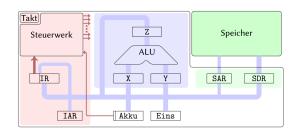
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben



Akku und Eins

- Akku dient als Zwischenspeicher für Datenworte
- Hält maximal ein Wort

- Eins liefert die Konstante 1, hält also Strom
- z.B. erhöhen des IAR

Aufbau der MIMA: ALU



Maximilian Staab,

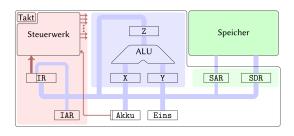
maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle



Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

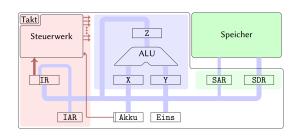
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Arithmetic Logic Unit (ALU) / Rechenwerk

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

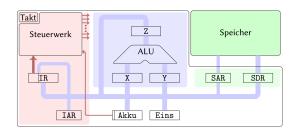
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Arithmetic Logic Unit (ALU) / Rechenwerk

Durchführt arithmetische Operationen

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

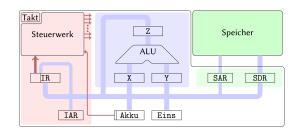
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Arithmetic Logic Unit (ALU) / Rechenwerk

- Durchführt arithmetische Operationen
- **mod** , **div** ,+,-,..., bitweises OR/AND/...

Aufbau der MIMA: ALU



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

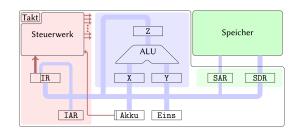
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Arithmetic Logic Unit (ALU) / Rechenwerk

 Durchführt arithmetische Operationen

- X und Y sind Eingaberegister
- **mod** , **div** ,+,-,..., bitweises OR/AND/...

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

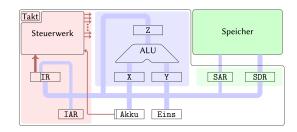
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Arithmetic Logic Unit (ALU) / Rechenwerk

- Durchführt arithmetische Operationen
- **mod** , **div** ,+,-,..., bitweises OR/AND/...
- X und Y sind Eingaberegister
- Z ist Ausgaberegister

Aufbau der MIMA: ALU



Maximilian Staab,

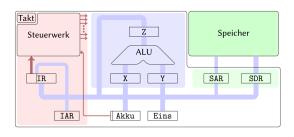
maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle



Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben

Takt Steuerwerk Z Speicher ALU IR X Y SAR SDR

Speicher(werk)

Speicher selbst speichert Befehle und Daten.

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

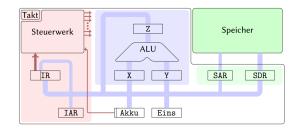
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Speicher(werk)

Speicher selbst speichert Befehle und Daten. Speicherwerk besteht aus:

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

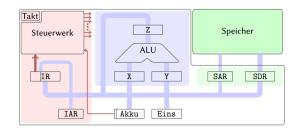
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Speicher(werk)

Speicher selbst speichert Befehle und Daten. Speicherwerk besteht aus:

 Speicheradressregister (SAR) ist die Adresse, bei der im Speicher gespeichert/gelesen werden soll

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

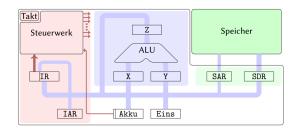
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben



Speicher(werk)

Speicher selbst speichert Befehle und Daten. Speicherwerk besteht aus:

 Speicheradressregister (SAR) ist die Adresse, bei der im Speicher gespeichert/gelesen werden soll Speicherdatenregister (SDR) Datum, das bei der Adresse gespeichert werden soll/ gelesen wurde.

Aufbau der MIMA: ALU



Maximilian Staab,

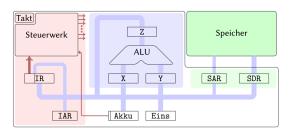
maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle



Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

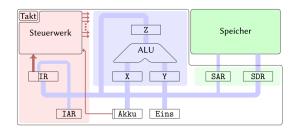
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben



Busse

Aufbau der MIMA: ALU



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

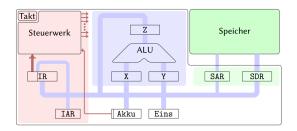
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Busse

"Kabel" zwischen den Verbindungen

Aufbau der MIMA: ALU



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

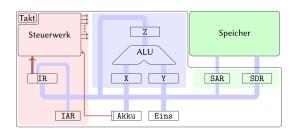
lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Busse

- "Kabel" zwischen den Verbindungen
- Ein kompletter Bus überträgt entweder 1, 0, oder nichts

Aufbau der MIMA: ALU



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

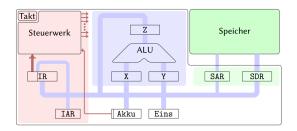
lukas bach@student kit edu

Zum Übungsblatt

MIMA

Maschinenbefehl

Aufgaben



Busse

- "Kabel" zwischen den Verbindungen
- Ein kompletter Bus überträgt entweder 1, 0, oder nichts

 Kann nur eine einzige Information auf einmal übertragen

Konventionen zu MIMA Programmen



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas bach@student kit edu

Zum Übungsblatt

MIMA

Um MIMA Programme und dazugehörige Definitionen verständlicher zu machen, vereinbaren wir folgende Konventionen:

Maschinenbetehl

Konventionen zu MIMA Programmen



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben

Um MIMA Programme und dazugehörige Definitionen verständlicher zu machen, vereinbaren wir folgende Konventionen:

 Befehle (eigentlich Bitfolge) schreiben wir als Befehlname und Adresse

Konventionen zu MIMA Programmen



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben

Um MIMA Programme und dazugehörige Definitionen verständlicher zu machen, vereinbaren wir folgende Konventionen:

- Befehle (eigentlich Bitfolge) schreiben wir als Befehlname und Adresse
 - lacktriangledown 0010000000000000000101010 \equiv STV 42

Konventionen zu MIMA Programmen



Maximilian Staab,
maximilian.staab@fsmi.uni-karlsruhe.de,
Lukas Bach,
lukas bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefeh

Aufgaben

Um MIMA Programme und dazugehörige Definitionen verständlicher zu machen, vereinbaren wir folgende Konventionen:

- Befehle (eigentlich Bitfolge) schreiben wir als Befehlname und Adresse
 - lacktriangle 0010000000000000000101010 \equiv STV 42
- $X \leftarrow Y \equiv$ "Der Variable X wird der Wert Y zugewiesen"

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu Befehlssyntax

Formel

Bedeutung

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Maxi	milian	Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

lukas.bach@student.kit.edu Befehlssyntax	Formel	Bedeutung
Zum Übungsblatt	Akku ← const	Lade eine Konstate <i>const</i> in den Akku

MIMA

Maschinenbefehle

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de.

Lukas Bach, 1ukas.bach@student.kit.edu Befehlssyntax Formel Bedeutung	
	state <i>const</i> in den
Zum Übungsblatt Akku	
MIMA LDV adr Akku \leftarrow M(adr) Lade einen We	ert vom Speicher
Maschinenbefehle bei Adresse au	dr in den Akku

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de.

Lukas Bach, lukas .bach@student.kit.edu Befehlssyntax		Formel	Bedeutung	
Zum Übungsblatt	LDC const	Akku ← const	Lade eine Konstate <i>const</i> in den	
Zuiii Obuligsbiatt			Akku	
MIMA	LDV adr	$Akku \leftarrow M(adr)$	Lade einen Wert vom Speicher	
Maschinenbefehle —		bei Adresse <i>adr</i> in den Akk		
iviascrimenbelenie –	STV adr	$M(adr) \leftarrow Akku$	Lade Speichere den Wert aus	
Aufgaben	Aufgaben dem Akku im Speic		dem Akku im Speicher bei	
			Adresse adr	

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Maximilian Staab,		
mavimilian etaah@femi	uni-karleruha	de

Lukas Bach,		Formel	Bedeutung	
Lukas Bach, lukas.bach@student.kit.edu Befehlssyntax		Former	Bedeutung	
	LDC const	Akku ← const	Lade eine Konstate <i>const</i> in den	
Zum Übungsblatt			Akku	
MIMA	LDV adr	$Akku \leftarrow M(adr)$	Lade einen Wert vom Speicher	
Maschinenbefehle -			bei Adresse <i>adr</i> in den Akku	
Waschinenbelenie	STV adr	$M(adr) \leftarrow Akku$	Lade Speichere den Wert aus	
Aufgaben			dem Akku im Speicher bei	
			Adresse adr	
_	LDIV adr	$Akku \leftarrow M(M(adr))$	Lade einen Wert vom Speicher	
			bei der Adresse, die bei adr ge-	
			speichert ist, und lade den Wert	
			in den Akku	
=				

MIMA Befehle



Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach, lukas.bach@student.kit.edu Befehlssyntax		Formel	Bedeutung
Zum Übungsblatt	LDC const	Akku ← const	Lade eine Konstate <i>const</i> in den Akku
MIMA Maschinenbefehle	LDV adr	$Akku \leftarrow M(adr)$	Lade einen Wert vom Speicher bei Adresse <i>adr</i> in den Akku
Aufgaben	STV adr	M(adr) ← Akku	Lade Speichere den Wert aus dem Akku im Speicher bei Adresse <i>adr</i>
LDIV adr Akku ← M($Akku \leftarrow M(M(adr))$	Lade einen Wert vom Speicher bei der Adresse, die bei <i>adr</i> ge- speichert ist, und lade den Wert in den Akku
	STIV adr	$M(M(adr)) \leftarrow Akku$	Speichere den Wert im Akku bei der Adresse, die in <i>adr</i> gespeichert ist.

MIMA Befehle (2)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

 ${\overset{\tt lukas.bach@student.kit.edu}{\texttt{E}}} ine \ MIMA-Maschine \ beherrscht \ folgende \ Maschinenbefehle:$

Zum Übungsblatt

Befehlssyntax

Formel

Bedeutung

MIMA

Maschinenbefehle

MIMA Befehle (2)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu. Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

7mm	Übung	chlat
Zuiii	Obuilig	Sulai

	Befehlssyntax	Formel	Bedeutung
MIMA	ADD adr	$Akku \leftarrow Akku + M(adr)$	Addiere den Wert
Maschinenbefehle			bei <i>adr</i> zum Akku
Aufgaben			dazu.

MIMA Befehle (2)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu. Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Zum	Hhu	nac	hlat
Zuiii	Obu	Hys	Diai

	Befehlssyntax	Formel	Bedeutung
MIMA	ADD adr	$Akku \leftarrow Akku + M(adr)$	Addiere den Wert
Maschinenbefehle			bei <i>adr</i> zum Akku
Aufgaben			dazu.
	"OP" adr	Akku"OP"M(adr)	Wende bitweise
			Operation auf
			Akku mit Wert
			bei adr an. $Op \in$
			$\{AND, OR, XOR\}.$

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edm Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Zum Übungsblatt

Befehlssyntax Bedeutung

MIMA

Maschinenbefehle

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edm Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Zum Übungsblatt

	Befehlssyntax	Bedeutung					
MIMA	NOT	Bitweise	Invertierung	aller	Bits	des	Akku-
Maschinenbefehle		Datenwortes					

Aufgaben

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.ed_Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Zum	ш	hu	na	ch	Jot
Zuiii	U	IJЦ	Hy	OL	лац

	Befehlssyntax	Bedeutung					
MIMA	NOT	Bitweise	Invertierung	aller	Bits	des	Akku-
Maschinenbefehle		Datenwortes					
Aufgaben	RAR	Rotiere alle Akku-Bits eins nach rechts					

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.ed_Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

7mm	Ubung	rchlat
Zuiii	Opulif	Jouran

	Befehlssyntax	Bedeutung					
MIMA	NOT	Bitweise	Invertierung	aller	Bits	des	Akku-
Maschinenbefehle		Datenwortes					
Aufgaben	RAR	RAR Rotiere alle Akku-Bits eins nach rechts					
3	EQL adr	Setze Akk	ku auf 11···1	1, falls	Wert k	oei <i>adı</i>	gleich
		Akku-Wert, setze Akku auf 00 · · · 00 sonst.					

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edm Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

Zum	ш	h		n		0	h	10
Zuiii	\cup	U	u	П	y	0	U	ıa

	Befehlssyntax	Bedeutung				
MIMA	NOT	Bitweise Invertierung aller Bits des Akku-				
Maschinenbefehle		Datenwortes				
Aufgaben	RAR	Rotiere alle Akku-Bits eins nach rechts				
EQL adr Setze Akku auf 11 ··· 11, falls Wert bei a						
		Akku-Wert, setze Akku auf 00 · · · 00 sonst.				
	JMP adr	Springe zu Befehlsadresse adr				

MIMA Befehle (3)



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.ed_Eine MIMA-Maschine beherrscht folgende Maschinenbefehle:

7mm	Ubung	chlat
Zuiii	Obully:	Sulai

	Betehlssyntax	Bedeutung				
MIMA	NOT	Bitweise Invertierung aller Bits des Akku-				
Maschinenbefehle		Datenwortes				
Aufgaben	RAR	Rotiere alle Akku-Bits eins nach rechts				
3	EQL adr	Setze Akku auf 11 · · · 11, falls Wert bei adr gleich				
		Akku-Wert, setze Akku auf 00 · · · 00 sonst.				
	JMP adr	Springe zu Befehlsadresse adr				
	JMN adr	Springe zu Befehlsadresse adr, falls Akku negativ				
		(also erstes Bit $=$ 1), sonst fahre normal fort.				

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsriche.de Lukas Bach, Befehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Sichern und Laden



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsribe.de Lukas Bach, Befehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsrBefehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

 LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

MIMA

LDC um eine Konstante zu laden

Maschinenbefehle

MIMA Befehle: Sichern und Laden



Maximilian Staab,

 $\substack{\text{maximilian.staab} \\ \text{Lukas Bach,}} \text{Befehle zum laden und Speichern in den Speicher}$

lukas.bach@student.kit.edu

LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

LDC um eine Konstante zu laden

IVIIIVIA

 Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält

Maschinenbefehle

MIMA Befehle: Sichern und Laden



Maximilian Staab,

 $\substack{\text{maximilian.staab} \\ \text{Lukas Bach,}} \text{Befehle zum laden und Speichern in den Speicher}$

lukas.bach@student.kit.edu

 LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

■ LDC um eine Konstante zu laden

MIMA

Maschinenbefehle

Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält: Akku.

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsrBefehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

■ LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

LDC um eine Konstante zu laden

MIMA

Data a series de la casa 7 de des

Maschinenbefehle

Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält: Akku.

Aufgaben

Beispiele:

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsrphs.ds Lukas Bach, Befehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

Maschinenbefehle

LDC um eine Konstante zu laden

MIMA

- LDG diff ciric Ronstante za laden

Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält: Akku.

Aufgaben

Beispiele:

LDV 9 lädt das Datum, das im Speicher bei Adresse 9 liegt, in den Akku.

MIMA Befehle: Sichern und Laden



Maximilian Staab,

 $\substack{\text{maximilian.staab} \\ \text{Lukas Bach,}} \text{Befehle zum laden und Speichern in den Speicher}$

lukas.bach@student.kit.edu

LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

Maschinenbefehle

■ LDC um eine Konstante zu laden

MIIMA

Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält: Akku.

Aufgaben

Beispiele:

- LDV 9 lädt das Datum, das im Speicher bei Adresse 9 liegt, in den Akku.
- STV 9 speichert das Datum, das im Akku liegt, in den Speicher an Adresse 9.

MIMA Befehle: Sichern und Laden



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsrBefehle zum laden und Speichern in den Speicher

lukas.bach@student.kit.edu

 LDV um Daten vom Speicher zu laden, STV um Daten in den Speicher zu schreiben

Zum Übungsblatt

Maschinenbefehle

LDC um eine Konstante zu laden

MIMA

LDC uni eme Nonstante zu lauer

Aufgaben

Daten werden in einem Zwischenspeicher gelagert, der nur ein Datenwort hält: Akku.

Beispiele:

- LDV 9 lädt das Datum, das im Speicher bei Adresse 9 liegt, in den Akku.
- STV 9 speichert das Datum, das im Akku liegt, in den Speicher an Adresse 9.
- LDC 4 lädt die Zahl 4 in den Akku (also kein Speicherzugriff).

MIMA Befehle: Sichern und Laden



Maximilian Staab,	Befehlssyntax	Formel	Bedeutung			
Lukas Bach, lukas.bach@student.kit.ed	LDC const	Akku ← const	Lade eine Konstate const in den			
			Akku			
Zum Übungsblatt	LDV adr	$Akku \leftarrow M(adr)$	Lade einen Wert vom Speicher			
MIMA -			bei Adresse adr in den Akku			
IVIIIVIA	STV adr	$M(adr) \leftarrow Akku$	Lade Speichere den Wert aus			
Maschinenbefehle			dem Akku im Speicher bei			
Aufgaben			Adresse adr			

MIMA Befehle: Sichern und Laden



Maximilian Staab, maximilian.staab@fsmi.uni-	Befehlssyntax	Formel	Bedeutung			
Lukas Bach, lukas.bach@student.kit.ed	LDC const	$Akku \leftarrow const$	Lade eine Konstate const in den			
			Akku			
Zum Übungsblatt	LDV adr	$Akku \leftarrow M(adr)$	Lade einen Wert vom Speicher			
			bei Adresse adr in den Akku			
MIMA -	STV adr	$M(adr) \leftarrow Akku$	Lade Speichere den Wert aus			
Maschinenbefehle			dem Akku im Speicher bei			
Aufgaben			Adresse adr			

Beispielprogramm mit initialem Speicherabbild

LDC 5 STV a ₁	:	Adresse Wert	
LDC 7	LDV <i>a</i> ₁	a ₁ 0	_
STV a ₂	STV a ₃	<i>a</i> ₂ 0	
:	HALT	$a_3 \mid 0$	

MIMA Befehle: Indirektes Sichern und Laden



Maximilian Staab, maximilian.staab@fsmi.uni_k	efehlssyntax	Formel	Bedeutung
Lukas Bach, lukas.bach@student.kit.edu	LDIV adr	$Akku \leftarrow M(M(adr))$	Lade einen Wert vom Speicher
Zum Übungsblatt			bei der Adresse, die bei <i>adr</i> ge- speichert ist, und lade den Wert in den Akku
Maschinenbefehle	STIV adr	$M(M(adr)) \leftarrow Akku$	Speichere den Wert im Akku bei der Adresse, die in <i>adr</i> gespei-
Aufgaben			chert ist.

MIMA Befehle: Indirektes Sichern und Laden



Maximilian Staab, maximilian.staab@fsmi.uni-k	efehlssyntax	Formel	Bedeutung
Lukas Bach, lukas.bach@student.kit.edu	LDIV adr	$Akku \leftarrow M(M(adr))$	Lade einen Wert vom Speicher
Zum Übungsblatt			bei der Adresse, die bei <i>adr</i> ge- speichert ist, und lade den Wert in den Akku
Maschinenbefehle Aufgaben	STIV adr	$M(M(adr)) \leftarrow Akku$	Speichere den Wert im Akku bei der Adresse, die in <i>adr</i> gespei- chert ist.

Beispielprogramm mit initialem Speicherabbild

LDIV 4	Adresse	Wert
STV 5	4	6
LDIV 5	5	0
STIV 4	6	7
HALT	7	2

MIMA Befehle: Eins plus Eins



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Eins plus Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

Befehle zu arithmetischen Operationen

MIMA

Maschinenbefehle

MIMA Befehle: Eins plus Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

Befehle zu arithmetischen Operationen

 Eine ALU-Operation, angewandt auf dem Wert des Akkus und dem Wert an gegebener Adresse

MIMA

Maschinenbefehle

MIMA Befehle: Eins plus Eins



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas bach@student kit edu

Zum Übungsblatt

Befehle zu arithmetischen Operationen

Maschinenbefehle

Eine ALU-Operation, angewandt auf dem Wert des Akkus und dem Wert an gegebener Adresse

Beispiele:

MIMA Befehle: Eins plus Eins



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

IVIIIVI

Maschinenbefehle

Aufgaben

Befehle zu arithmetischen Operationen

- Eine ALU-Operation, angewandt auf dem Wert des Akkus und dem Wert an gegebener Adresse
- Beispiele:
 - ADD 4 addiert den Wert im Akku mit dem Wert aus dem Speicher an Adresse 4 und legt das Resultat im Akku ab

MIMA Befehle: Eins plus Eins



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

Befehle zu arithmetischen Operationen

- Eine ALU-Operation, angewandt auf dem Wert des Akkus und dem Wert an gegebener Adresse
 - Beispiele:
 - ADD 4 addiert den Wert im Akku mit dem Wert aus dem Speicher an Adresse 4 und legt das Resultat im Akku ab. Achtung: Addition nicht mit dem Wert 4!

MIMA Befehle: Eins plus Eins



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

- Befehle zu arithmetischen Operationen
- Eine ALU-Operation, angewandt auf dem Wert des Akkus und dem Wert an gegebener Adresse
 - Beispiele:
 - ADD 4 addiert den Wert im Akku mit dem Wert aus dem Speicher an Adresse 4 und legt das Resultat im Akku ab. Achtung: Addition nicht mit dem Wert 4!
 - AND 3 führt bitweise Verundung zwischen dem Wert im Akku und dem Wert aus dem Speicher an Adresse 4 durch und legt das Resultat im Akku ab.

MIMA Befehle: Eins plus Eins



Maximilian Staab, maximilian.staab@fsmi.uni-kar	efehlssyntax	Formel	Bedeutung
Lukas Bach, lukas.bach@student.kit.edu	ADD adr	$Akku \leftarrow Akku + M(adr)$	Addiere den Wert bei adr zum
			Akku dazu.
Zum Übungsblatt	"OP" adr	Akku"OP"M(adr)	Wende bitweise Operation auf
MIMA			Akku mit Wert bei adr an. $Op \in \{AND, OR, XOR\}$.
Maschinenbefehle			(AND, OH, AOH).

MIMA Befehle: Eins plus Eins



Maximilian Staab, maximilian.staab@fsmi.uni-l	Befehlssyntax	Formel	Bedeutung
Lukas Bach, — lukas.bach@student.kit.edu	ADD adr	$Akku \leftarrow Akku + M(adr)$	Addiere den Wert bei adr zum
			Akku dazu.
Zum Übungsblatt	"OP" adr	Akku"OP"M(adr)	Wende bitweise Operation auf
MIMA			Akku mit Wert bei adr an. $Op \in$
Maschinenbefehle			$\{AND, OR, XOR\}.$
Mascrimeribelenie			
Aufgaben	Beispielprogramr	n mit initialem Speicherabb	ild

Aufgaben

Beispielprogramm mit initialem Speicherabbild

LDC 5 ADD 3 AND 4 STV 5 LDC 12

XOR 5 HALT

Adresse	Wert
3	3
4	8
5	17

MIMA Befehle: Bits und Bytes



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Bits und Bytes



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

NOT invertiert alle Bits des Datums im Akku.

MIMA

Maschinenbefehle

MIMA Befehle: Bits und Bytes



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

 NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits

MIMA

Maschinenbefehle

MIMA Befehle: Bits und Bytes



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

MIMA Befehle: Bits und Bytes



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

RAR rotiert alle Bits des Datums im Akku um eine Stelle nach rechts.

MIMA Befehle: Bits und Bytes



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

 NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
 5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

RAR rotiert alle Bits des Datums im Akku um eine Stelle nach rechts. Beispiel mit 5 im Akku: 000001012 wird zu 100000102.

MIMA Befehle: Bits und Bytes



Maximilian Staab,
maximilian.staab@fsmi.uni-karlsruhe.de,
Lukas Bach,
lukas bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

 NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
 5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

- RAR rotiert alle Bits des Datums im Akku um eine Stelle nach rechts. Beispiel mit 5 im Akku: 00000101₂ wird zu 10000010₂.
- EQL adr vergleicht den Wert im Akku mit dem Wert bei addr.

MIMA Befehle: Bits und Bytes



Maximilian Staab,
maximilian.staab@fsmi.uni-karlsruhe.de,
Lukas Bach,
lukas bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

 NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
 5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

- RAR rotiert alle Bits des Datums im Akku um eine Stelle nach rechts. Beispiel mit 5 im Akku: 00000101₂ wird zu 10000010₂.
- EQL adr vergleicht den Wert im Akku mit dem Wert bei addr.
 - Setzt Akku = 11 · · · 11 falls Werte gleich sind.

MIMA Befehle: Bits und Bytes



Maximilian Staab,
maximilian.staab@fsmi.uni-karlsruhe.de,
Lukas Bach,
lukas bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

 NOT invertiert alle Bits des Datums im Akku. Beispiel NOT mit 5 im Akku, angenommen der Akku speichert bis zu 8 bits:
 5₁₀ = 00000101₂, nach der Invertierung: 11111010₂.

- RAR rotiert alle Bits des Datums im Akku um eine Stelle nach rechts. Beispiel mit 5 im Akku: 000001012 wird zu 100000102.
- EQL adr vergleicht den Wert im Akku mit dem Wert bei addr.
 - Setzt Akku = 11 · · · 11 falls Werte gleich sind.
 - Setzt Akku = $00 \cdots 00$ falls Werte nicht gleich sind.

MIMA Befehle: Bits und Bytes



Maximilian Staab, maximilian.staab@fsmi.uni- Lukas Bach,	-⊮ Befeh lssyntax	Bedeutun	g				
lukas.bach@student.kit.edu	NOT	Bitweise	Invertierung	aller	Bits	des	Akku-
Zum Übungsblatt –		Datenwortes					
	RAR	Rotiere alle Akku-Bits eins nach rechts					
MIMA	EQL adr	Setze Akku auf 11 · · · 11, falls Wert bei adr gleich					
Maschinenbefehle		Akku-Wert, setze Akku auf 00 · · · 00 sonst.					

MIMA Befehle: Bits und Bytes



Maximilian Staab, maximilian.staab@fsmi.uni- Lukas Bach,	-⊮ Befeh lssyntax	Bedeutun	g				
lukas.bach@student.kit.edu	NOT	Bitweise	Invertierung	aller	Bits	des	Akku-
Zum Übungsblatt –		Datenwortes					
	RAR	Rotiere alle Akku-Bits eins nach rechts					
MIMA	EQL adr	Setze Akku auf 11 · · · 11, falls Wert bei adr gleich					
Maschinenbefehle		Akku-Wert, setze Akku auf 00 · · · 00 sonst.					

Aufgaben

Beispielprogramm mit initialem Speicherabbild

LDC 5

NOT

RAR RAR NOT EQL 15 RAR EQL 0

: HALT

MIMA Befehle: Springen



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.edu

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

Lukas bach@student.kit.edu Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

MIMA

Maschinenbefehle

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

Lukas bach@student.kit.edu Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

Also Befehle werden von oben nach unten abgearbeitet

MIMA

Maschinenbefehle

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas bach@student kit edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

Maschinenbefehle

Aufgaben

Also Befehle werden von oben nach unten abgearbeitet

Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erh\u00f6ht

Zum Übungsblatt

MIMA

Maschinenbefehle

Aufgaben

Also Befehle werden von oben nach unten abgearbeitet

Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen

JMP adr führt als nächsten Befehl den an Adresse adr aus.

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

MIMA

Maschinenbefehle

- Also Befehle werden von oben nach unten abgearbeitet
- Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen
- JMP adr führt als nächsten Befehl den an Adresse adr aus.
- JMN adr führt als nächsten Befehl den an Adresse adr aus, falls der Akku negativ ist.

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

MIMA

Maschinenbefehle

- Also Befehle werden von oben nach unten abgearbeitet
- Also belefile werden von oben nach unten abgearbeitet
- Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen
- JMP adr führt als nächsten Befehl den an Adresse adr aus.
- JMN adr führt als nächsten Befehl den an Adresse adr aus, falls der Akku negativ ist.
 - Also wenn das erste Bit im Akku negativ ist.

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erhöht

Zum Übungsblatt

MIMA

Maschinenbefehle

- Also Befehle werden von oben nach unten abgearbeitet
- Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen
- JMP adr führt als nächsten Befehl den an Adresse adr aus.
- JMN adr führt als nächsten Befehl den an Adresse adr aus, falls der Akku negativ ist.
 - Also wenn das erste Bit im Akku negativ ist.
 - Wenn vorher ein *EQL* erfolgreich verglichen hat, wird also gesprungen.

MIMA Befehle: Springen



Maximilian Staab.

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach.

lukas.bach@student.kit.edu

 Normalerweise wird die Instruktionsadresse nach jedem Befehl um eins erh\u00f6ht

Zum Übungsblatt

MIMA

Maschinenbefehle

- Also Befehle werden von oben nach unten abgearbeitet
- Mit Sprüngen kann man die MIMA zwingen, zu definiertem Befehl zu springen und damit die Vorgehensreihenfolge zu beeinflussen
- JMP adr führt als nächsten Befehl den an Adresse adr aus.
- JMN adr führt als nächsten Befehl den an Adresse adr aus, falls der Akku negativ ist.
 - Also wenn das erste Bit im Akku negativ ist.
 - Wenn vorher ein *EQL* erfolgreich verglichen hat, wird also gesprungen.
 - Wenn der Akku positiv ist, werden die Befehle nach JMN normal weiter abgearbeitet.

MIMA Befehle: Springen



Maximilian Staab, maximilian.staab@fsmi.uni-	Befehlssyntax	Bedeutung
Lukas Bach, lukas.bach@student.kit.edu	FOL adr	Setze Akku auf 11 · · · 11, falls Wert bei adr gleich
		Akku-Wert, setze Akku auf 00 · · · 00 sonst.
Zum Übungsblatt	JMP adr	Springe zu Befehlsadresse adr
MIMA	JMN adr	Springe zu Befehlsadresse adr, falls Akku negativ
		(also erstes $Bit = 1$), sonst fahre normal fort.
-		Springe zu Befehlsadresse <i>adr</i> Springe zu Befehlsadresse <i>adr</i> , falls Akku negativ

Maschinenbefehle

MIMA Befehle: Springen



Maximilian Staab, maximilian.staab@fsmi.un	Befehlssyntax	Bedeutung
Lukas Bach, lukas.bach@student.kit.e	FOL adr	Setze Akku auf 11 · · · 11, falls Wert bei adr gleich
		Akku-Wert, setze Akku auf 00 · · · 00 sonst.
Zum Übungsblatt	JMP adr	Springe zu Befehlsadresse adr
MIMA	JMN adr	Springe zu Befehlsadresse adr, falls Akku negativ
		(also erstes $Bit = 1$), sonst fahre normal fort.
Maschinenbefehle		
Aufgaben	Beispielprogrami	m mit initialem Speicherabbild
	LDC 5	÷
	a ₁ : JMN a ₂	NOT Adresse Wert

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.

MIMA-Programm schreiben

Zum Übungsblatt Schreibe ein MIMA-Programm:

MIMA

Maschinenbefeh

■ Eingabe: Adresse *a*₁ einer positiven Zahl *x*.

■ Ausgabe: Speichert 3 · x in a₁.

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de, Lukas Bach,

lukas.bach@student.kit.

MIMA-Programm schreiben

Zum Übungsblatt

Schreibe ein MIMA-Programm:

MIMA

Maschinenbefeh

■ Eingabe: Adresse *a*₁ einer positiven Zahl *x*.

Ausgabe: Speichert 3 · x in a₁.

Aufgaben

Lösung:

LDV a₁

ADD a₁

ADD a₁

STV a₁

HALT

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

MIMA-Programm schreiben

Zum Übungsblatt Schreibe ein MIMA-Programm:

■ Eingabe: Adresse a₁ einer positiven Zahl x.

Ausgabe: Speichert x mod 2 in a₁.

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach,

lukas.bach@student.kit.edu

MIMA-Programm schreiben

Zum Übungsblatt

Schreibe ein MIMA-Programm:

MIMA

■ Eingabe: Adresse *a*₁ einer positiven Zahl *x*.

Maschinenbefehl

Ausgabe: Speichert x mod 2 in a₁.

Aufgaben

Lösung:

AND a₁ STV a₁

HALT

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach, lukas.bach@student.kit.e

MIMA-Programm schreiben

Zum Übungsblatt

Schreibe ein MIMA-Programm:

MIMA

■ Eingabe: Adresse *a*₁ einer positiven Zahl *x*.

IVIIIVI

Ausgabe: Speichert x div 2 in a₁.

Maschinenbefeh

Aufgaben



Maximilian Staab,

maximilian.staab@fsmi.uni-karlsruhe.de,

Lukas Bach, lukas.bach@student.kit.e

MIMA-Programm schreiben

Zum Übungsblatt

Schreibe ein MIMA-Programm:

Ι.

■ Eingabe: Adresse *a*₁ einer positiven Zahl *x*.

MIIM

Ausgabe: Speichert x div 2 in a₁.

Maschinenbefehl

Aufgaben

Lösung:

LDC₁

NOT

AND a_1 // Setze "rechtestes" Bit auf 0

RAR

STV a₁

HALT

Maximilian Staab,

maximilian.staab@fsmi.uni Lukas Bach,

lukas.bach@student.kit.ed

Zum Übungsblatt

MIMA

Maschinenbefehle

