



Grundlagen der Rechnerarchitektur und Betriebssysteme

Instruction Set Architecture

Florian Knoch · Lehrstuhl für Praktische Informatik,
insbes. Systemnahe Programmierung, Universität Bamberg

Instruction Set Architectures

auf einen Blick



AUFGABE 1

Strukturieren Sie folgende Begriffe in einer Konzeptwolke:

- Intel
 - ARM
 - RISC
 - CISC
 - AMD64
 - VLIW
 - RISC-V
 - ISA
 - x64
 - AMD
 - EPIC
 - AArch64
 - x86-64
 - RISC-V
 - PowerPC
- Internat.

Instruction Set Architectures

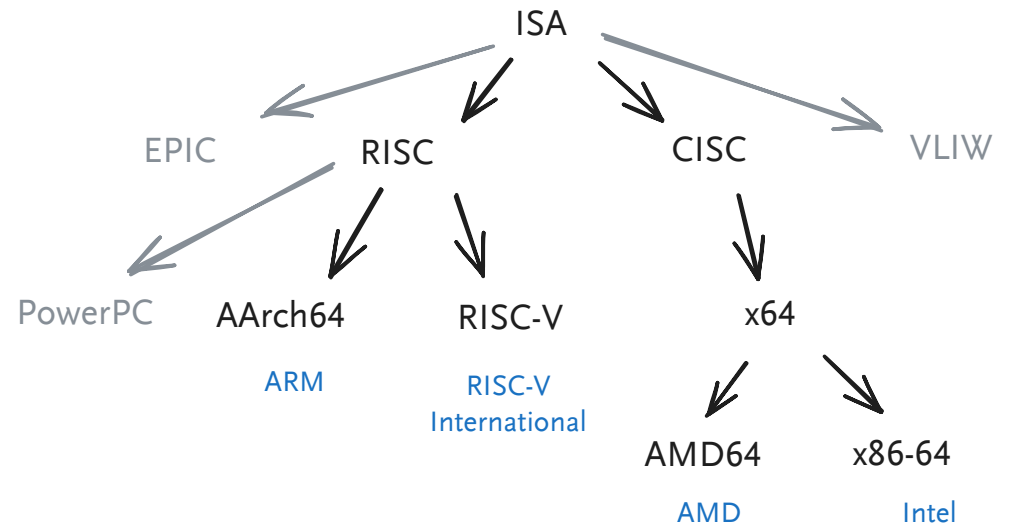


auf einen Blick

AUFGABE 1

Strukturieren Sie folgende Begriffe in einer Konzeptwolke:

- Intel
- ARM
- RISC
- CISC
- AMD64
- VLIW
- RISC-V
- ISA
- x64
- AMD
- EPIC
- AArch64
- x86-64
- RISC-V Internat.
- PowerPC



Instruction Set Architectures

Mehr als nur Intel



Abbildung 1: Installationsseite des Betriebssystems FreeBSD

© FreeBSD Community (<https://www.freebsd.org/>)

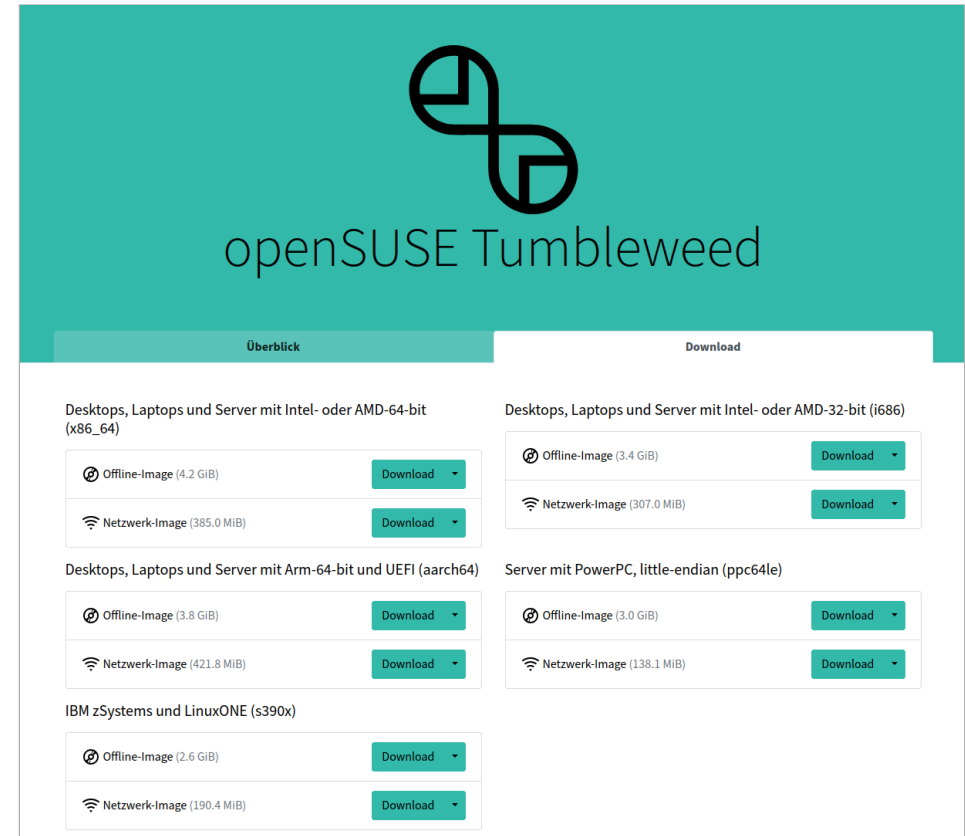


Abbildung 2: Installationsseite des Betriebssystems openSUSE

© openSUSE-Mitwirkende (<https://get.opensuse.org/de/>)

Instruction Set Architectures



CPUs in freier Wildbahn

Smartphones¹

Personal Computer

Supercomputer²

IoT-Geräte

¹ *Smartphone application processor (AP)/system-on-chip (SoC) vendor shipment share worldwide from 2020 to 2025, by quarter.* Counterpoint Research, 2026. <https://www.statista.com/statistics/1226674/smartphone-application-processor-vendor-shipment-share-worldwide/>. Zuletzt aufgerufen am 18.05.2026.

² TOP500-Liste. <https://www.top500.org/lists/top500/2025/11/>. Zuletzt aufgerufen am 18.05.2026.

Instruction Set Architectures



CPUs in freier Wildbahn

Smartphones³

hauptsächlich ARM-basierte Prozessoren (MediaTek, Apple, Qualcomm)

Personal Computer

Supercomputer⁴

IoT-Geräte

³Smartphone application processor (AP)/system-on-chip (SoC) vendor shipment share worldwide from 2020 to 2025, by quarter. Counterpoint Research, 2026. <https://www.statista.com/statistics/1226674/smartphone-application-processor-vendor-shipment-share-worldwide/>. Zuletzt aufgerufen am 18.05.2026.

⁴TOP500-Liste. <https://www.top500.org/lists/top500/2025/11/>. Zuletzt aufgerufen am 18.05.2026.

Instruction Set Architectures



CPUs in freier Wildbahn

Smartphones⁵

hauptsächlich ARM-basierte Prozessoren (MediaTek, Apple, Qualcomm)

Personal Computer

Intel/AMD weiter dominant, Apple mit ARM-basiertem Apple Silicon

Supercomputer⁶

IoT-Geräte

⁵ *Smartphone application processor (AP)/system-on-chip (SoC) vendor shipment share worldwide from 2020 to 2025, by quarter.* Counterpoint Research, 2026. <https://www.statista.com/statistics/1226674/smartphone-application-processor-vendor-shipment-share-worldwide/>. Zuletzt aufgerufen am 18.05.2026.

⁶ TOP500-Liste. <https://www.top500.org/lists/top500/2025/11/>. Zuletzt aufgerufen am 18.05.2026.

Instruction Set Architectures



CPUs in freier Wildbahn

Smartphones⁷

hauptsächlich ARM-basierte Prozessoren (MediaTek, Apple, Qualcomm)

Personal Computer

Intel/AMD weiter dominant, Apple mit ARM-basiertem Apple Silicon

Supercomputer⁸

Intel/AMD üblich, auch einige ARM-basierte Prozessoren von Nvidia dabei

IoT-Geräte

⁷ *Smartphone application processor (AP)/system-on-chip (SoC) vendor shipment share worldwide from 2020 to 2025, by quarter.* Counterpoint Research, 2026. <https://www.statista.com/statistics/1226674/smartphone-application-processor-vendor-shipment-share-worldwide/>. Zuletzt aufgerufen am 18.05.2026.

⁸ TOP500-Liste. <https://www.top500.org/lists/top500/2025/11/>. Zuletzt aufgerufen am 18.05.2026.

Instruction Set Architectures



CPUs in freier Wildbahn

Smartphones⁹

hauptsächlich ARM-basierte Prozessoren (MediaTek, Apple, Qualcomm)

Personal Computer

Intel/AMD weiter dominant, Apple mit ARM-basiertem Apple Silicon

Supercomputer¹⁰

Intel/AMD üblich, auch einige ARM-basierte Prozessoren von Nvidia dabei

IoT-Geräte

ARM üblich, RISC-V aus Lizenzgründen attraktiv

⁹Smartphone application processor (AP)/system-on-chip (SoC) vendor shipment share worldwide from 2020 to 2025, by quarter. Counterpoint Research, 2026. <https://www.statista.com/statistics/1226674/smartphone-application-processor-vendor-shipment-share-worldwide/>. Zuletzt aufgerufen am 18.05.2026.

¹⁰TOP500-Liste. <https://www.top500.org/lists/top500/2025/11/>. Zuletzt aufgerufen am 18.05.2026.

RISC-V



Instruction Set Architecture

Bestandteile nach ARM¹¹ am Beispiel RISC-V

- **Instruction set:** The full list of machine instructions supported by the processor (e.g., arithmetic, control flow, memory [...])
-
-
-

Instruktionen in RV32I

```
// Reg.-Reg.-Befehle  
add a1, t1, a2
```

```
// Reg.-Imm.-Befehle  
blt t1, t2, 0x400
```

```
// Speicherbefehle  
sw a2, 4(t2)
```

```
// Sprungbefehle  
jalr zero, 44(t4)
```

¹¹What is Instruction Set Architecture (ISA)? ARM, 2026. Verfügbar unter <https://www.arm.com/glossary/isa>.
Zuletzt aufgerufen am 19.05.2026.



Instruction Set Architecture

Bestandteile nach ARM¹² am Beispiel RISC-V

- **Instruction set:** The full list of machine instructions supported by the processor (e.g., arithmetic, control flow, memory [...])
- **Instruction formats:** Bit-level layout defining opcodes, operands, and addressing modes
-
-

R-type Instruction in RV32I

| Bits | Bedeutung |
|---------|-----------|
| 31 – 25 | funct7 |
| 24 – 20 | rs2 |
| 19 – 15 | rs1 |
| 14 – 12 | funct3 |
| 11 – 7 | rd |
| 6 – 0 | opcode |

¹²What is Instruction Set Architecture (ISA)? ARM, 2026. Verfügbar unter <https://www.arm.com/glossary/isa>. Zuletzt aufgerufen am 19.05.2026.



Instruction Set Architecture

Bestandteile nach ARM¹³ am Beispiel RISC-V

- **Instruction set:** The full list of machine instructions supported by the processor (e.g., arithmetic, control flow, memory [...])
- **Instruction formats:** Bit-level layout defining opcodes, operands, and addressing modes
- **Registers:** Number, types, and roles (e.g., general purpose, [...] special function)
-

Register in RV32I

| Register | ABI | Beschreibung |
|----------------|--------|-------------------------|
| x0 | zero | Festverdrahtete Null |
| x1 | ra | Rücksprungadresse |
| x2 | sp | Stack Pointer |
| x3 | gp | Global Pointer |
| x4 | tp | Thread Pointer |
| x5–x7, x28–x31 | t0–t6 | Flüchtige Register |
| x8 | s0/fp | Frame Pointer |
| x9, x18–x27 | s1–s11 | Nichtflüchtige Register |
| x10–x17 | a0–a7 | Funktionsargumente |

¹³What is Instruction Set Architecture (ISA)? ARM, 2026. Verfügbar unter <https://www.arm.com/glossary/isa>. Zuletzt aufgerufen am 19.05.2026.



Instruction Set Architecture

Bestandteile nach ARM¹⁴ am Beispiel RISC-V

- **Instruction set:** The full list of machine instructions supported by the processor (e.g., arithmetic, control flow, memory [...])
- **Instruction formats:** Bit-level layout defining opcodes, operands, and addressing modes
- **Registers:** Number, types, and roles (e.g., general purpose, [...] special function)
- **Data types:** Supported types like integers, floating-point numbers, vectors, [...]

Datentypen in RV32I

- **Integer** (RV32I)
- **Single-Precision Floating-Point** („F“ Extension)
- **Double-Precision Floating-Point** („D“ Extension)
- **Vector Operations** („V“ Extension)

(u. v. m., vgl. [ISA-Referenz](#) [1])

¹⁴What is Instruction Set Architecture (ISA)? ARM, 2026. Verfügbar unter <https://www.arm.com/glossary/isa>. Zuletzt aufgerufen am 19.05.2026.

Instruction Set Architecture

Bestandteile nach ARM¹⁵ am Beispiel RISC-V

- **Memory architecture:** Includes addressing modes, endianness, memory protection, and virtual memory support
- **Privilege levels:** Execution modes such as user mode and kernel mode to enable secure OS functionality
- **Interrupts and exceptions:** Mechanisms for handling asynchronous events and fault conditions

Weitere Details:

RISC-V Reader

D. Patterson und A. Waterman [2]



© RISC-V International

¹⁵What is Instruction Set Architecture (ISA)? ARM, 2026. Verfügbar unter <https://www.arm.com/glossary/isa>. Zuletzt aufgerufen am 19.05.2026.

Datenpfad eines einfachen RISC-V-Rechners



AUFGABE 2

Welche Werte haben die Instruktionsteile rd, rs1 und rs2 bzw. imm für die nebenstehenden Instruktionen?

Bestimmen Sie zudem die Konfiguration der Steuersignale A bis G (1, 0 oder X für „Don't care“) bei der Verarbeitung der Instruktionen. B bezeichnet hierbei das RegisterWrite-Signal.

```
addi x5, x5, 1
lw x12, 0(x3)
jalr x10, 8(x11)
beq x5, x11, 0x400
```

Organisatorisches

Kurzer Reminder



- Anmeldung zum Hackspace-Ausflug:
<https://vc.uni-bamberg.de/mod/feedback/view.php?id=2186339>
- Die Sprechstunde ist weiterhin buchbar:
<https://vc.uni-bamberg.de/mod/scheduler/view.php?id=2154592>

Fragen?



- [1] R.-V. International, Hrsg., [Online]. Verfügbar unter: <https://github.com/riscv/riscv-isa-manual/releases/download/20240411/unpriv-isa-asciidoc.pdf>
- [2] D. Patterson und A. Waterman, *The RISC-V Reader: An Open Architecture Atlas*. 2017.



[Kursmaterialien](#) · [Quelltext](#)

Diese Präsentation wurde zuletzt am 25.05.2026 bearbeitet. Sie basiert auf den Folien der zugehörigen Vorlesung von Prof. Dr. Michael Engel. Sofern nicht anders angegeben, sind die Inhalte unter der **Lizenz CC BY-SA 4.0** verfügbar. Das Universitätslogo sowie die Schriftart UB Scala Sans sind Eigentum der Otto-Friedrich-Universität Bamberg.

Folgende **Open-Source-Komponenten** kommen in der Präsentation zum Einsatz:

[circuiteria](#) (Apache 2.0), [finite](#) (MIT), [fontawesome](#) (SIL), [pgf-tikz](#) (GPL 2.0), [typst](#) (Apache 2.0), [typst-ccicons](#) (MIT), [typst-polylux](#) (MIT).

