

Architektur/Chip/Komponente

Department Informatik 4
Systemsoftware
Friedrich-Alexander-Universität Erlangen-Nürnberg

WS 2026/2027



Viele Komponenten ähneln sich. Z.B.:

- i286, i386, Pentium, AMD64, ... bauen aufeinander auf
- alte Legacy-Chips (PIT, PIC, RTC, ...) sind Bestandteile der modernen Chip-Sets
- viele moderne Netzwerk-Karten sind NE2000-kompatibel
- ...

Gleiche Chips kommen u.U. auf verschiedenen Platinen vor. Z.B.:

- Speicher-Chips
- ROM-Bausteine
- ...

=> Hierarchische Aufteilung *Architektur, Chip, Komponente, System*



Architektur: Teil eines Chips

Chip: Bauteil, das i.a. auf einer Platine eingelötet ist
(z.B. Intel 80386, Motorola MC146818A, ITE 8661F,
SST 29EE020, ...)

Komponente: Teil, das man als End-Kunde im Laden als fertiges Einzelteil kaufen kann (z.B. Netzteil, Chassis, CPU, Motherboard, Festplatte, Grafik-Karte, ...)

System: Gesamtheit aller zu emulierenden Komponenten und Busse.



Verschiedene Aufteilungen:

Architektur/Chip:■ Chips nicht umkonfigurierbar

- Architekturen und Busse sind zur Compile-Zeit bekannt
- Anzahl der Architekturen und Busse in einem Chip gering

Chip/Komponente:■ Komponenten nicht umkonfigurierbar

- Chips und Busse zur Compile-Zeit bekannt
- u.U. viele Chips und Busse auf einer Komponente

Komponente/System:■ System konfigurierbar

- Komponenten und Busse erst zur Laufzeit bekannt
- System besteht aus vielen, über viele Busse verbundene Komponenten



- Chips nicht umkonfigurierbar
 - Architekturen und Busse sind bekannt
 - Anzahl der Architekturen und Busse in einem Chip gering
- => Architekturen können sich gegenseitig direkt aufrufen.



Chip:

```
#include "arch-a.c" /* Contains "a_func" */  
  
inline b_func(int op) {  
    a_func(op);  
}  
  
#include "arch-b.c" /* Calls "b_func" */
```

Compiler verbindet Aufrufe zur Compile-Zeit

=> sehr gute Performance.



- Komponenten nicht umkonfigurierbar
- Chips und Busse zur Compile-Zeit bekannt
- u.U. viele Chips und Busse auf einer Komponente

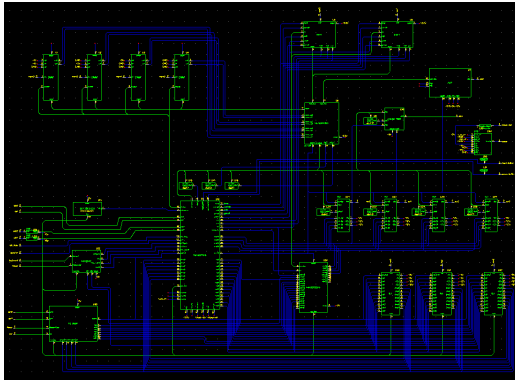
Durch die große Anzahl von Chips und Bussen wird eine Verknüpfung über Inline-Funktionen wie bei „Architektur/Chip“ praktisch unmöglich:

- Die Compile-Zeit für einige FAUmachine-Chips liegt schon im Minuten-Bereich. Ein komplettes Inlining aller Motherboard-Chips würde Compile-Zeiten von Stunden bedeuten.
- in-, out-, load-, store-Aufrufe der CPU gehen im Wesentlichen an externe Steck-Karten. Callback-Funktionen bzw. Speicherbereiche in der CPU gecacht.

=> Verbindung der Chips mit den Bussen innerhalb von Komponenten über Callback-Mechanismus.



Beispiel FAUmachine-Motherboard-Komponente:



Über kleines Tool in C-Datei (ca. 1000 Zeilen Code) übersetzbar.



- System konfigurierbar
- Komponenten und Busse erst zur Laufzeit bekannt
- System besteht aus vielen, über viele Busse verbundene Komponenten

Zur Compile-Zeit ist nichts bekannt.

=> Verbindung der Komponenten im System über
Callback-Mechanismus.



Es gibt viele Möglichkeiten,

- - Architekturen,
 - Chips,
 - Komponenten,
 - das Systemzu modellieren,
- das System zu konfigurieren.

