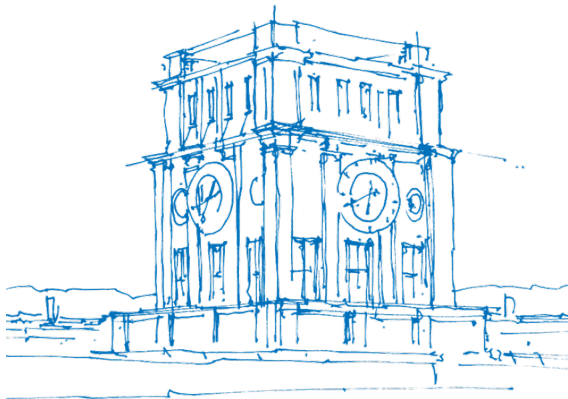


Einführung in die Rechnerarchitektur

Lukas Hertel

Lehrstuhl für Rechnerarchitektur und Parallele Systeme
Fakultät für Informatik
Technische Universität München

15. November 2021



TUM Uhrenturm

Adressierung bei RISC und CISC

- Wie kann man auf Adressen zugreifen bei x86?
- Ist das nötig? Wie machen das andere Plattformen?
- Beispiele für RISC?

Adressierung

Zusammenfassung

- $[BASE + (INDEX * SCALE) + DISPLACEMENT]$
 - ☐ SCALE kann 1, 2 oder 4 sein
 - ☐ DISPLACEMENT kann 1 byte oder 4 byte gross sein
 - ☐ BASE und INDEX Register werden nicht verändert

Wiederholung zu Adressierungsformen

[MOV EBX, [EAX + ECX * 4 + 12]]

- Darstellung in RISC (Nur BASE)
- Kein MUL (da ineffizient)
- Gleiche Register dürfen sich verändern wie im Original
- Vorteile/Nachteile

```
class Struktur1
public:
    int wert;
    int feld1[16];
;

class Struktur2
public:
    Struktur1* feld2[8];
    int info;
;
```


Unterprogrammaufruf

- Unterschied Sprung & Unterprogramm-Aufruf
- Stack bei Unterprogrammaufruf

Möglichkeiten für einen Unterprogrammaufruf mit Parametern



- Register
- Speicherstellen
- Stack
- Vorteile/Nachteile?