



Bachelor-Arbeit / Forschungsarbeit Nr. 1032

Entwicklung eines RISC-V Emulators in Java und Integration in ein bestehendes Framework für Prozessoremulatoren



Methoden

Software-Entwurf
Programmierung in Java

Themengebiete

Rechnerarchitektur
Grafische Benutzeroberfläche

Hintergrund

RISC-V ist eine quelloffene und frei verfügbare Befehlssatzarchitektur. Anders als bisherige Befehlssatzarchitekturen kann RISC-V ohne die Entrichtung von Nutzungs- oder Lizenzgebühren sowohl privat als auch kommerziell verwendet werden. Darüber hinaus kann man die Befehlssatzarchitektur weiterentwickeln, ohne sich zu verpflichten, diese Weiterentwicklungen offen zu legen. Aufgrund dieser Vorteile ist RISC-V sowohl für Forschung und Lehre als auch für Unternehmen sehr attraktiv und gewinnt stetig an Bedeutung. So beschäftigen sich auch bekannte Unternehmen wie Western Digital, Esperanto und Siemens mit RISC-V.

Ein vom IKR entwickeltes Framework für Prozessoremulatoren wird zur Unterstützung in der Lehre verwendet. Dieses in Java programmierte Framework umfasst derzeit die drei in den Lehrveranstaltungen des IKR behandelten Prozessoren "Minimalprozessor", "IKR CISC" und "IKR RISC-II". Dieses Framework soll nun um die RISC-V Architektur erweitert werden.

Problembeschreibung

Ziel dieser Arbeit ist die Entwicklung eines RISC-V Emulators in der Programmiersprache Java und die Integration in das bestehende Framework. Dabei soll der RISC-V Emulator mindestens das RV32I Base Integer Instruction Set implementieren. Die Arbeit gliedert sich in folgende Schritte:

- Einarbeitung in RISC-V und das bestehende Framework für Prozessoremulatoren
- Implementierung einer Klasse riscV.java zur Emulation eines RISC-V Prozessors
- Integration in das Framework und Erweiterung der GUI

Erworbene Kenntnisse und Fähigkeiten

Sie verstehen die Funktionsweise eines RISC-V Prozessors und erweitern Ihre Programmierkenntnisse in Java. Des Weiteren erlernen Sie die Konzeption und Entwicklung von GUI-Applikationen. Schließlich erwerben Sie die Fähigkeit, sich in bestehende Software-Frameworks einzuarbeiten und diese zu erweitern.

Voraussetzungen

Rechnerarchitektur
Programmierkenntnisse in Java

Erwünschte Vorkenntnisse

Scanner- und Parser-Generatoren

Kontakt

Dipl.-Ing. Matthias Meyer

Raum 1.334 (ETI II), Telefon 685-67975, E-Mail matthias.meyer@ikr.uni-stuttgart.de

M.Sc. Christian Koehler

Raum 1.320 (ETI II), Telefon 685-69001, E-Mail christian.koehler@ikr.uni-stuttgart.de