



Bachelor-Arbeit / Forschungsarbeit Nr. 1093

**Implementierung einer gdb-kompatiblen Debugging-Schnittstelle für ein
experimentelles objektbasiertes Rechnersystem**



Methoden

Programmierung in Assembler

Themengebiete

Rechnerarchitektur

Hintergrund

Am Institut entsteht ein FPGA-basierter Prototyp eines neuartigen objektbasierten Prozessors, der sich u. a. durch eine erhöhte Robustheit auf Ebene der Instruction Set Architecture auszeichnet. Für diese Prozessorarchitektur sollen in naher Zukunft ein vollständiges Betriebssystem sowie Anwendungs-Software entwickelt werden.

Die derzeit verfügbaren Werkzeuge zum Debugging von Software auf dem Prozessor stellen lediglich einige rudimentäre Grundfunktionen bereit. Dazu gehören beispielsweise ein einzelner Breakpoint, ein bisher kaum genutzter Single-Step-Modus oder die Ausgabe von Registerinhalten im Fehlerfall.

Die Software gdb ("GNU Debugger") ist ein Entwicklerwerkzeug, das umfangreiche Funktionen zum Debugging von Software bereitstellt. Unter anderem ist ein sogenannter "Remote Mode" vorhanden, in dem gdb z. B. über eine serielle Schnittstelle ein Programm von einem anderen Rechnersystem aus beobachten und steuern kann.

Aufgabenstellung

Ziel der Arbeit ist die Implementierung einer gdb-kompatiblen Debugging-Schnittstelle für das experimentelle Rechnersystem, um das Debugging von Assembler-Programmen im "Remote Mode" zu ermöglichen. Die Implementierung soll in der Assemblersprache des objektbasierten Prozessors erfolgen und umfasst die nachfolgenden Schritte.

- Einarbeitung in das paketbasierte Protokoll der gdb-Remote-Schnittstelle und Implementierung dieses Protokolls.
- Implementierung von gdb-Debugging-Befehlen, wie z. B. "read registers", "set breakpoint" oder "single step".
- Anpassung der bereits vorhandenen Systemsoftware (insbesondere Interrupt- und Fault-Handler), um eine Interaktion mit dem Debugger zu ermöglichen.
- Optional: Integration in die Entwicklungsumgebung "VSCodium" zur Steuerung des Debuggers über eine grafische Benutzeroberfläche.

Erworbene Kenntnisse und Fähigkeiten

Sie erhalten ein tiefes Verständnis der Funktionsweise moderner Entwicklungs- und Debugging-Werkzeuge und vertiefen Ihr Wissen im Bereich der hardwarenahen Software-Entwicklung. Sie lernen außerdem eine neuartige, experimentelle Prozessorarchitektur kennen.

Voraussetzungen

Technische Informatik I

Erwünschte Vorkenntnisse

Erfahrung mit Hardware-naher Programmierung

Kontakt

M.Sc. Timo Madeheim

Raum 1.336 (ETI II), Telefon 685-69012, E-Mail timo.madeheim@ikr.uni-stuttgart.de