

CPU Simulator

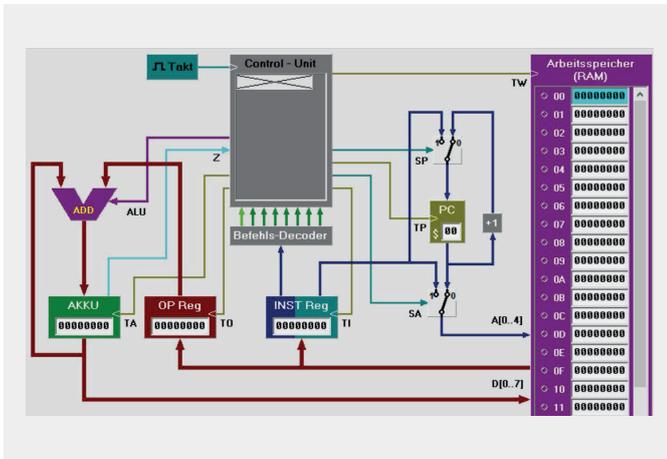


Abb. 1: Simulationssoftware MikroSim32, basierend auf ein 8-Bit Mikroprozessor

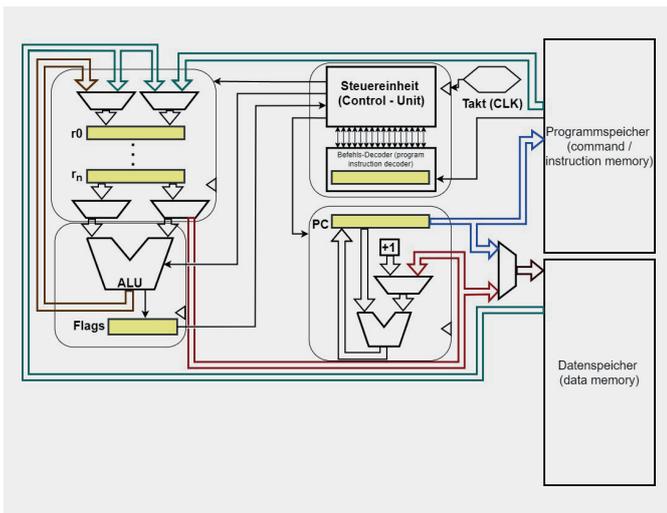


Abb. 2: Blockdiagramm des neuen RISC-Simulators

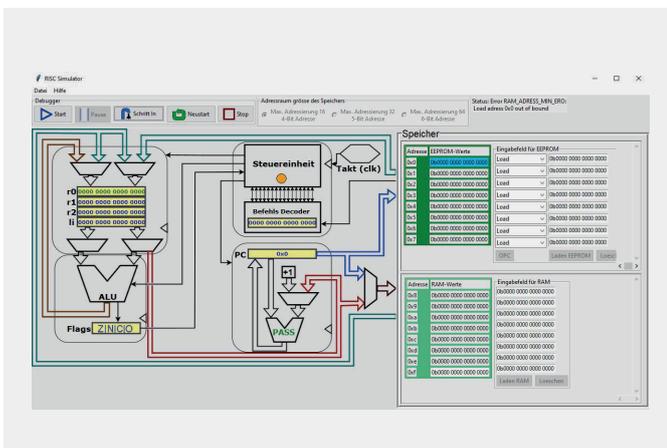


Abb. 3: RISC-SIM Version 1.0 Windows

Problemstellung

Im MCFUN-Modul an der Hochschule Luzern wird die Simulationssoftware MikroSim32 (siehe Abbildung 1) verwendet, um den Studierenden die Struktur und Funktionsweise einer Central Processing Unit (CPU) sowie den Ablauf in einer Micro Controller Unit (MCU) zu vermitteln. Der aktuelle MikroSim32-Simulator an der HSLU ist veraltet und spiegelt den Aufbau von Mikrocontrollern nicht mehr zeitgemäss wider. Zudem treten vermehrt Probleme bei der Ausführung auf modernen Betriebssystemen auf. Daher wird in dieser Bachelorarbeit das bestehende MikroSim32-Programm überarbeitet und auf den neusten Stand gehoben.

Lösungskonzept

Die gegenwärtige Konzeption zielt darauf ab, einen Simulator gemäss dem RISC-Prinzip zu entwerfen, wobei besonderes Augenmerk auf eine zeitgemässe, benutzerfreundliche grafische Benutzeroberfläche gelegt wird. Diese Initiative basiert auf dem Leitbild des RISC-V. Die konzeptionelle Umsetzung wird durch das Blockdiagramm, welches in Abbildung 2 veranschaulicht ist, dargestellt.

Realisierung

Es wird auf der Grundlage der Programmiersprache Python, der Harvard-Architektur und dem RISC-Prinzip eine neue 16-Bit Central Processing Unit Simulation Programm aufgebaut. Dieser Simulator trägt den Namen RISC-SIM und steht für RISC-Simulationsprogramm. Der Prozessablauf der CPU wird durch eine Animation visualisiert. Diese läuft in einem Thread und wird mit einer FSM realisiert. Ausserdem wird für die grafische Oberfläche die Library Tkinter verwendet. Diese bietet die Möglichkeit, zusammen mit Python-Code, eine grafische Oberfläche zu erstellen. Die Wahl auf Tkinter ist darauf zurückzuführen, dass sie auf den Betriebssystemen Windows, Mac und Linux ein lauffähiges Grafisches Benutzeroberfläche anbietet.

Ergebnisse

Es zeigt sich, dass der RISC-SIM Version 1.0 (siehe Abbildung 3) ohne eine Entwicklungsumgebung auf den Betriebssystemen Windows und Mac lauffähig ist und seinen Zweck erfüllt. Es gibt zwei Bugfixes, einen optischen und einen beim Laden von .txt-Dateien, die jedoch keinen Einfluss auf die Lauffähigkeit des Programms haben. Ein weiterer Punkt ist, dass der RISC-SIM Version 1.0 für Mac auf einem Apple M2 (arm64) Architektur basierenden Hardware erzeugt wurde. Alle auf Intel-Architektur basierenden Macs laufen mit der RISC-SIM Version 1.0 nicht. Damit ist der RISC-SIM Version 1.0 Mac nicht rückwärts kompatibel, was auch der Philosophie von Apple entspricht. Ausserdem setzt Apple für ihre neuen Mac-Notebooks auf den M2-Chip. Somit stellt dies kein Problem dar, da die kommenden Studenten zwangsläufig mit der M2-Hardwarearchitektur arbeiten werden. Mitunter steht der Code und das Programm unter der BSD-3 Lizenz.



Diplomand
Sigrist Markus

Dozent
Prof. Dr. M. Thalmann

Themengebiet
Technische Informatik & Embedded Systems

Projektpartner
Intern