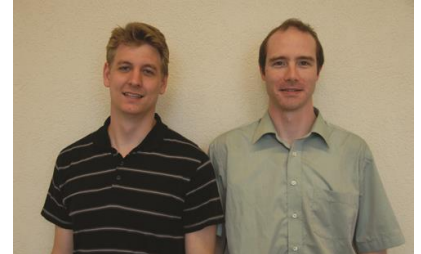


## Multicore Programming unter Linux

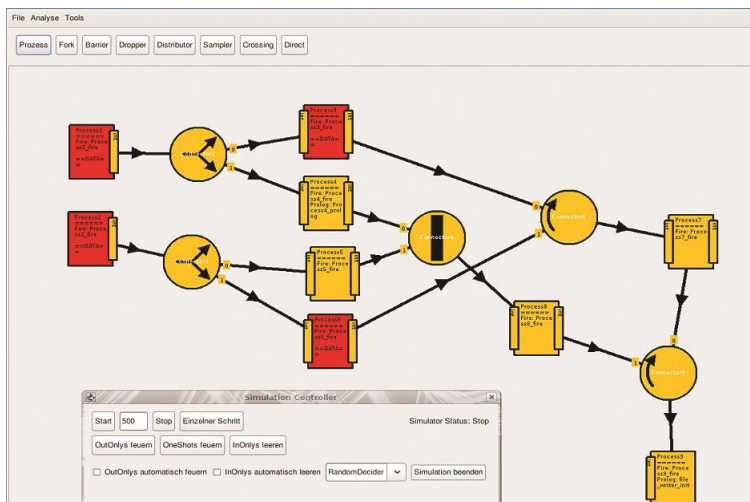
Die zunehmende Verbreitung von Multicore-Prozessoren im Bereich der eingebetteten Systeme verlangt nach neuen Programmiermodellen, welche die Entwicklung von Anwendungen für diese Systeme entsprechend unterstützen. Die momentan meist verbreitete Methodik, um Software für parallele Systeme zu entwickeln, basiert auf Threads. Ein grosser Nachteil ist, dass nicht vorhersehbare Abläufe zu Deadlocks führen können.

Diese Arbeit beschreibt ein aktororientiertes Programmiermodell, das die Entwicklung von Software für Multicore-Systeme erleichtert. Mit einfachen und klar definierten verknüpfbaren Elementen - Prozessen und Konnektoren - ist es möglich, eine Anwendung mit deterministischem Verhalten zu beschreiben. Wir stellen eine Beschreibungssprache zur Verfügung, welche die Formulierung von Anwendungen innerhalb des Programmiermodells ermöglicht. Eine Codegenerierung erlaubt die Umsetzung der entworfenen Anwendung in kompilierbaren Code. Eine Sammlung von Werkzeugen unterstützt den Entwickler bei der Arbeit mit dem Programmiermodell (z.B. Deadlockerkennung). Für die Detektion von Deadlocks werden mathematische Methoden in Kombination mit heuristischen Verfahren eingesetzt. Ein Simulator bietet die Möglichkeit, ein System zu analysieren.



Diplomierende  
Peter Herter  
Tobias Martin Klausner

Dozent  
Markus Thaler



Die visuelle Oberfläche unterstützt den Programmierer in der Erstellung und Manipulation von Anwendungen innerhalb des Programmiermodells. Die Simulation ist bei der Analyse des Modells hilfreich.