

**СИСТЕМА СИНТЕЗА РАСПИСАНИЯ ДЛЯ САПР КРУПНОГРАНУЛЯРНЫХ
РЕКОНФИГУРИРУЕМЫХ ПРОЦЕССОРОВ С ПОДДЕРЖКОЙ ВЕТВЛЕНИЯ**

Новоселов Б.С. (ИТМО)

**Научный руководитель – кандидат технических наук, доцент Пенской А.В.
(ИТМО)**

Тема крупногранулярных реконфигурируемых процессоров (CGRA) становится все более актуальной благодаря их высокой производительности, гибкости и энергоэффективности. Эти качества делают их лучшей альтернативой существующим методам для выполнения ряда вычислительных задач. Однако возникает вопрос: можно ли доверять архитектурам CGRA в системах реального времени и обеспечивать их прозрачную интерпретацию с языка высокого уровня? Проект NITTA стремится решить эту задачу. Это конфигурируемая САПР для CGRA-архитектур, которая фиксирует граф вычислений на этапе компиляции (Static Dataflow) и исключает его изменение во время выполнения программы. Такой подход позволяет точно планировать поведение программы с точностью до такта.

Целью данной работы является создание эффективного механизма ветвления потока управления в среде ациклического Static Dataflow, а именно для проекта NITTA. Существуют несколько различных способов организации планировщика, первый из них наивный, он заключается в использовании мультиплексора и обладает очевидными недостатками, учитывая, что узким местом является шина, по которой происходят пересылки данных, в данном варианте придётся выделять столько данных на пересылки, сколько существует вариантов условий; второй же способ порождает различные вариации пересылок и составляет общий из них, тем самым, уменьшая временные затраты на неиспользуемые пересылки. Основная сложность второго подхода, который и был выбран, заключается в оптимизации множественных вложенных ветвлений.

Результатом данной работы стал проверенный с помощью Property Based и юнит тестов эффективный механизм ветвления, в контексте Static Dataflow архитектуры. Разработанное дополнение помогает системам, построенным с использованием NITTA, быть Тьюринг-полными, что положительно отображается на будущем проекта.

Список использованных источников:

1. T. M. Parks, J. L. Pino and E. A. Lee, "A comparison of synchronous and cycle-static dataflow," Conference Record of The Twenty-Ninth Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 1995, pp. 204-210 vol.1, doi: 10.1109/ACSSC.1995.540541. keywords: {Processor scheduling;Streaming media;Computer networks;Computational modeling;Communication channels;Signal processing;Concurrent computing;Parallel processing;Video signal processing;Fires}
2. Lee, E. A., Neuendorffer, S., & Zhou, G. (2014). Dataflow. In C. Ptolemaeus (Ed.), System design, modeling, and simulation using Ptolemy II. Ptolemy.org. Retrieved from
3. Gao, G.R. (1991). The Static Data Flow Model. In: A Code Mapping Scheme for Dataflow Software Pipelining. The Kluwer International Series in Engineering and Computer Science, vol 125. Springer, Boston, MA. https://doi.org/10.1007/978-1-4615-3988-9_2