

УДК 004.855.5

РАЗРАБОТКА АЛГОРИТМА ИСПОЛНЕНИЯ ПРОГРАММ В МОДЕЛИ NESTING-SAFE RECOVERABLE LINEARIZABILITY

Кокорин И. В. (федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)

Научный руководитель – научный сотрудник факультета ИТиП, Аксенов В.Е.

(федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»)

Аннотация. Предлагаемый проект нацелен на разработку алгоритма исполнения конкурентных программ, поддерживающих восстановление после сбоев, при наличии энергонезависимой, побайтово адресуемой памяти (NVRAM).

Введение. Традиционно, память в компьютерных системах делится на два вида: 1) RAM, адресуемая побайтово, обеспечивающая быстрый случайный доступ и являющаяся энергозависимой; и 2) дисковая память, адресуемая поблочко, обеспечивающая очень медленный случайный доступ и чуть более быстрый последовательный, являющаяся энергонезависимой. Энергонезависимая память со случайным доступом (NVRAM) объединяет достоинства этих двух видов памяти.

Рассмотрим модель, в которой вся разделяемая между процессами память является энергонезависимой со случайным доступом, при этом кэши и регистры процессора являются энергозависимыми.

Цель работы заключается в разработке и реализации инфраструктуры, позволяющей запускать в описанной модели программы, которые будут самостоятельно восстанавливаться после сбоев (как после сбоя одного процесса, так и всей системы в целом).

Основная часть. Изучаемая модель представлена в статье [1]. В этой модели с каждой операцией Op предоставляется операция $Op.Recover$, которая вызывается при сбое системы и завершает операцию Op . В случае повторных сбоев $Op.Recover$ вызывается после каждого, до тех пор, пока не удастся завершить исполнение Op .

В рамках работы была изучена архитектура энергонезависимой байтоадресуемой памяти и был разработан алгоритм для восстановления системы после сбоя с учётом особенностей NVRAM.

Алгоритм поддерживает для каждого потока в NVRAM персистентный стек, к которому имеет доступ только один процесс. Персистентный стек состоит из персистентных фреймов, которые хранятся в стеке для каждой выполняющейся в данный момент функции, выполняющейся в NVRAM. Персистентный фрейм содержит адрес функции Op , адрес $Op.Recover$ и аргументы, с которыми была вызвана функция Op .

Доступ к NVRAM имеют только рабочие потоки (и, соответственно, только они имеют персистентный стек), в то время координацией и восстановлением после сбоев занимается главный поток. В случае падения одного рабочего потока, главный поток порождает поток-восстановитель, который проходит по персистентному стеку упавшего потока с конца и вызывает $Op.Recover$ для каждой функции Op , которая не была завершена в момент сбоя потока. Если $Op.Recover$ выполнена успешно, то поток продолжает восстановление с функции, вызвавшей Op (или завершается, если Op была самой внешней функцией). В случае повторного сбоя восстановление продолжается с той функции, на которой произошёл повторный сбой. При этом нужно учитывать возможность порчи персистентного стека, так как запись персистентного стекового фрейма может быть неатомарна (он может не влезть в одну кеш-линию).

Выводы. Результаты, полученные в работе, позволят на основе алгоритма реализовать систему, способную восстанавливаться после сбоев с использованием NVRAM.

Литература.

1. Attiya H., Ben-Baruch O., Hendler D. Nesting-safe recoverable linearizability: Modular constructions for non-volatile memory // Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, 2018, pp. 7-16

Кокорин И.В. (автор)

Подпись

Аксенов В.Е. (научный руководитель)

Подпись