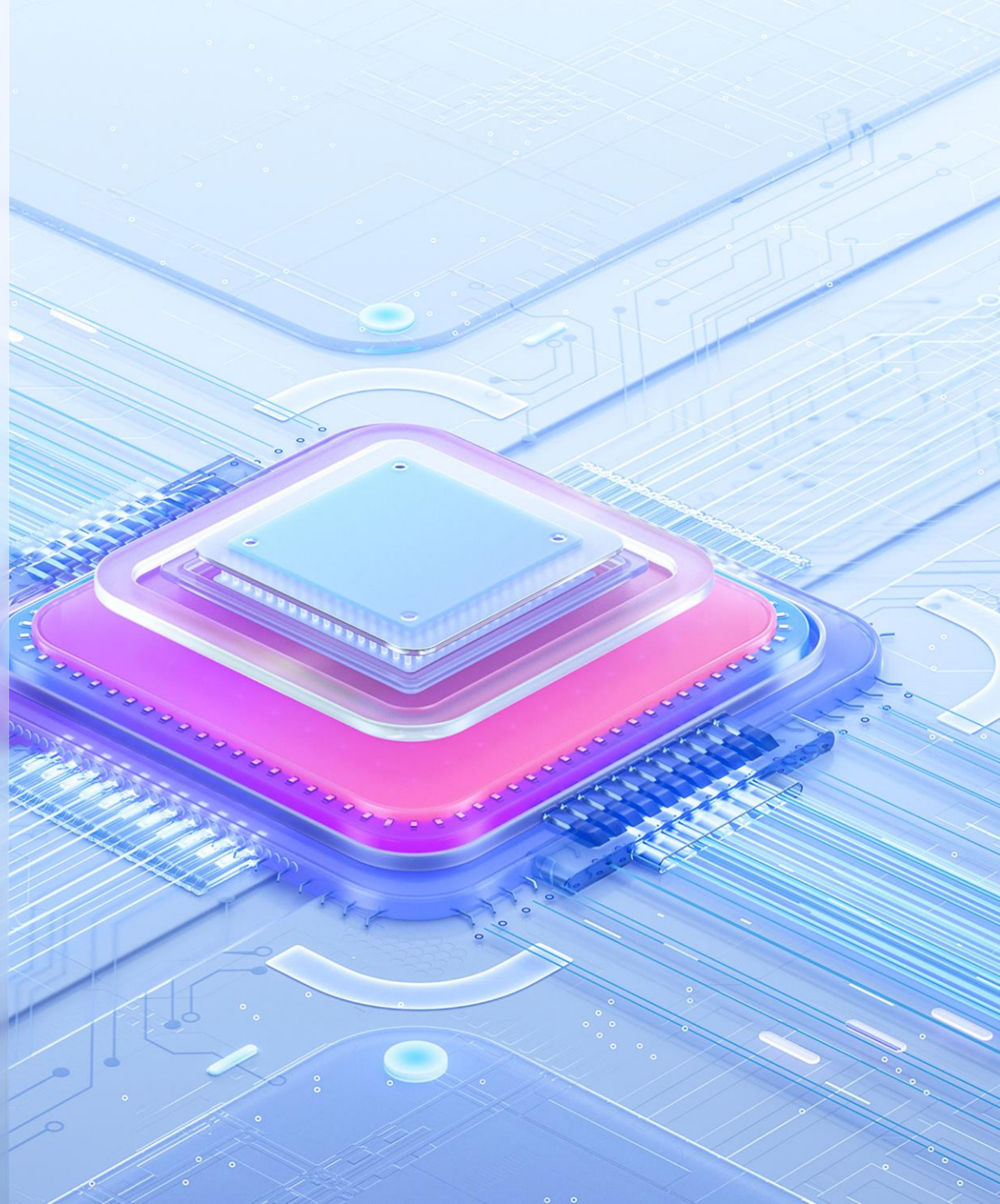
Наблюдения и выводы из анализа производительности доступных на рынке RISC-V серверов

Митап RISC-V альянса

Дмитрий Петроченко, Сбер

Апрель, 2024



Объект исследования

Современные аппаратные решения разных поколений и архитектур на типовых enterprise нагрузках

на одной из первых серверных двух-процессорных **RISC-V** платформ

Ориентируемся на **Java приложения** в связи с распространенностью среди задач банка

Наблюдения

На нагрузках банка наблюдаются отставания в производительности по сравнению с типовыми платформами

От сложных приложений перешли на более простые бенчмарки, на которых также наблюдается отставание

пример:

SPECjvm2008, среди задач которой есть чувствительные как к подсистеме памяти, так и к частоте процессора

В процессе анализа выявлены 2 ключевых фактора

1. Проблемы производительности и системы реализации кешей
2. Высокое время отклика от момента запроса данных до момента их получения в подсистеме кешей и памяти

Характеристики рассмотренной RISC-V серверной системы

RISC-V

Два 64-ядерных процессора, работающих на частоте 2Ghz:

- L1: 64KB, L2: 1MB, L3: 64MB
  - L2 общий на каждый кластер, кластер – 4 ядра

NUMA топология: 4 NUMA ноды на сокет (по количеству каналов памяти),

- в каждой NUMA ноде по 4 кластера (16 ядер)

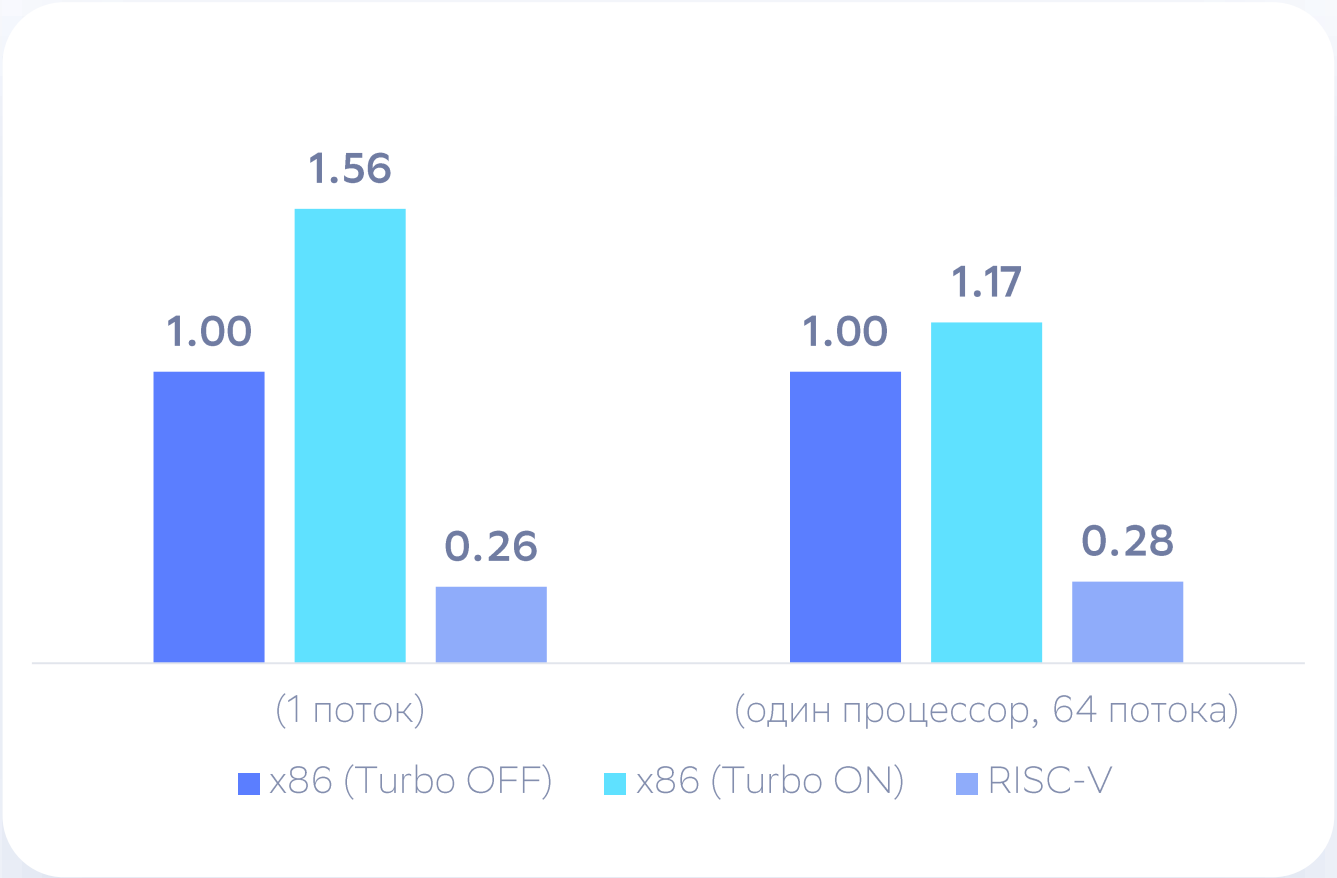
x86 платформа

Два 32-х ядерных процессора (по 64 логических потока, SMT ON), 2Ghz базовая частота (до 3.2GHz Turbo)

- L1: 48KB, L2: 1.25MB, L3: 48MB

NUMA топология: каждый сокет – отдельная NUMA нода (SubNUMA OFF)

Результаты на SPECjvm2008
(сравнимая частота процессора)



Более высокая частота на x86 – существенный фактор, объясняет заметную часть отличия

Ряд нагрузок чувствителен к производительности подсистемы кешей и памяти (scimark.large), этот фактор рассмотрен более подробно далее

Комментарий:

В дальнейших измерениях частота на x86 системе зафиксирована на уровне, сравнимом с RISC-V системой

Измерения между системами, сравнимыми по количеству логических потоков

Для оценки базовых характеристик памяти разработан синтетический тест

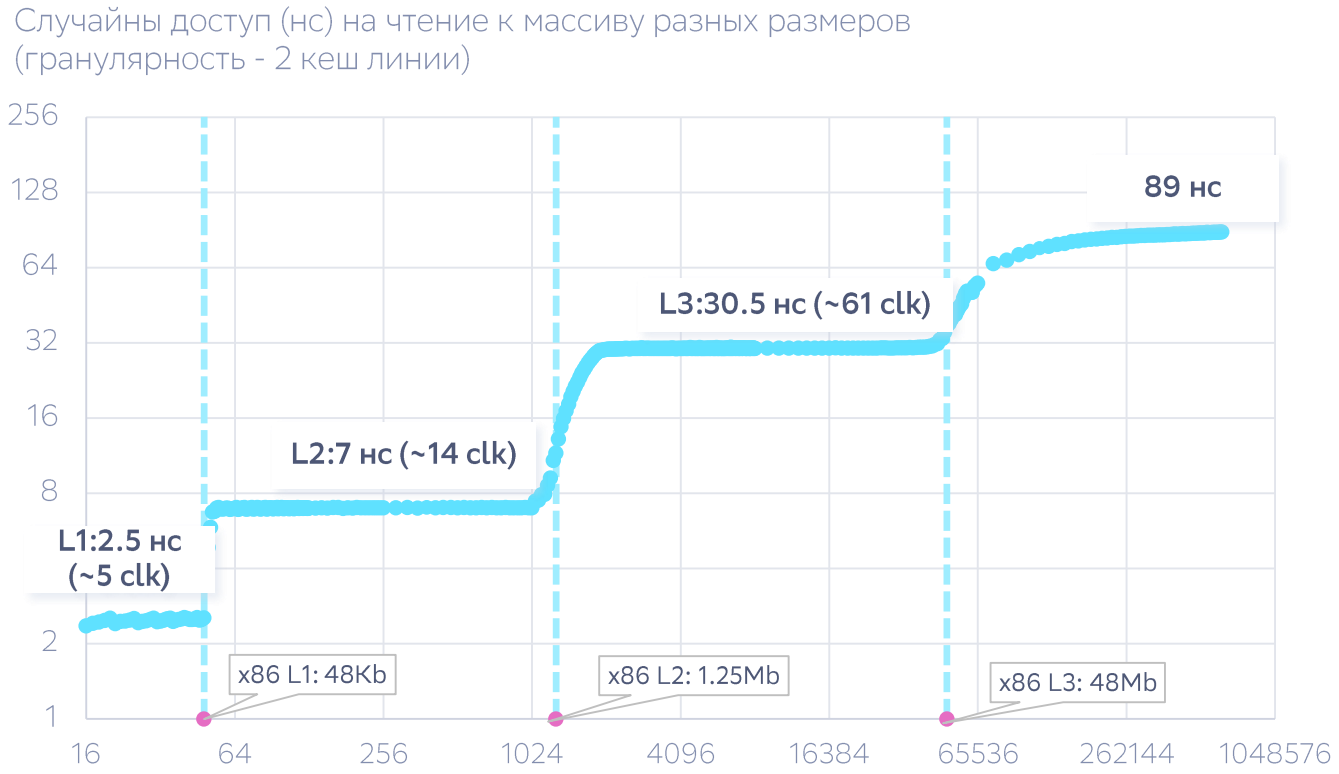
Сценарий:
случайный доступ к массиву заданного размера (pointer chasing)“p_next = *curr”

Перед запуском измерения обеспечиваются следующие требования к заполнению массива:

- Случайный выбор при переходе к следующему элементу
  - Гранулярность доступа (stride) 128 байт
- Обход всех кеш линий, принадлежащих массиву, в соответствии с шагом (stride)
- После полного обхода возвращаемся к первому элементу массива

Комментарий: *C++ бенчмарк, компилятор: GCC 11.3, ОС: FC38*

Измерения на x86



x86 система (Turbo off):

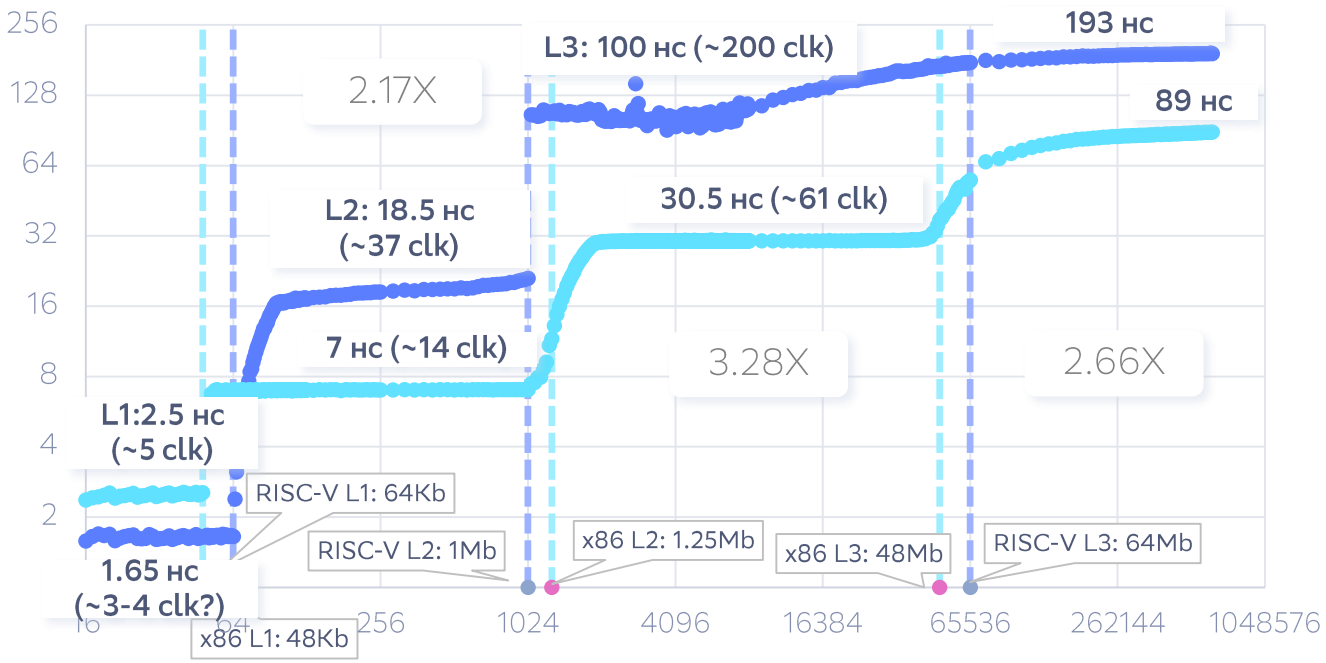
Ось X: размер массива (Kb)
Ось Y: оценка времени единичного доступа к массиву (нс)

- **Наблюдаются ожидаемые уровни, коррелирующие с размерами L1/L2/L3**
- **Результаты измерения совпадают с характеристиками кешей из открытых источников**

Настройки системы:
- *Используются huge pages*
- *Память выделяется на локальном сокете*

Измерения на RISC-V

Случайны доступ (нс) на чтение к массиву разных размеров (гранулярность - 2 кеш линии)



x86 система (Turbo off):

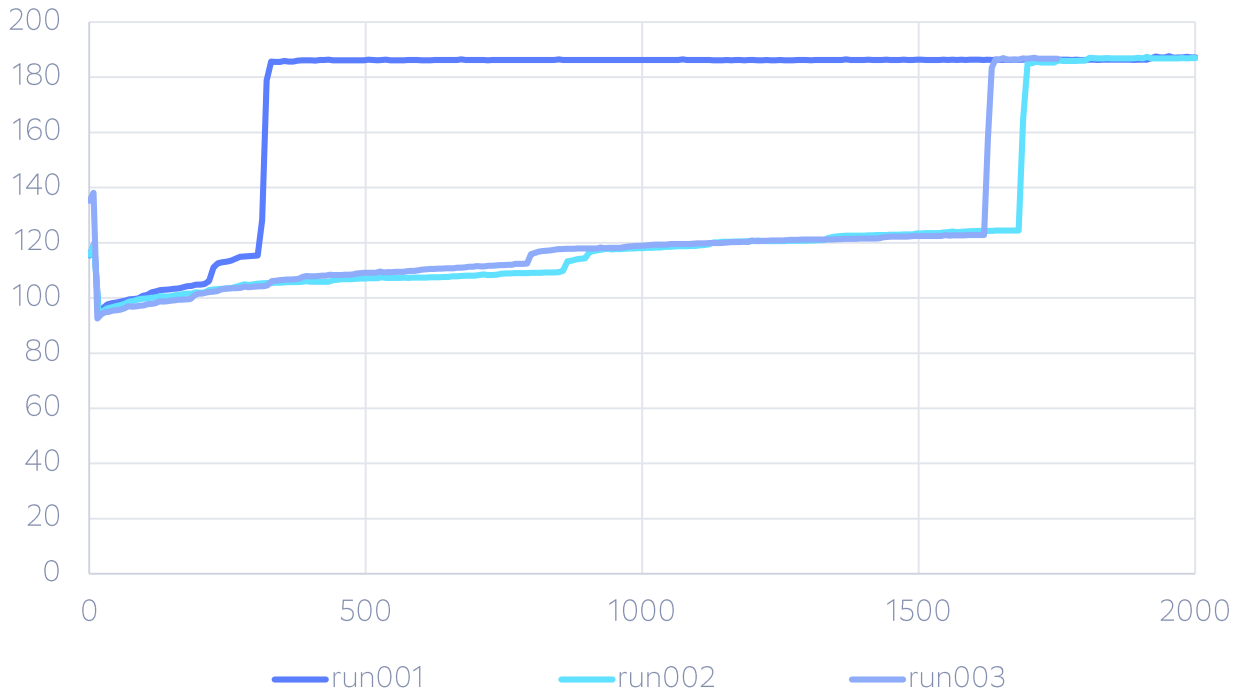
RISC-V система:

Ось X: размер массива (Kb)
Ось Y: оценка времени единичного доступа к массиву (нс)

- **Измерения на RISC-V показывают большее время доступа к кешам L2 и L3, а также к памяти**
- **В L3 наблюдается неожиданный рост времени доступа ранее, чем достигнут размер L3**

	RISC-V	x86	Отставание RISC-V
L1 latency	1.65нс (~3-4clk))	2.45нс (~5clk)	0.67X (L1 латенси в 1.5 раза лучше)
L2 latency	18.65нс (37clk)	7нс (~14clk)	2.66X
L3 latency	100нс (~200clk)	30.5нс (~61clk)	3.28X
Memory latency	193нс (~386clk)	89нс (178clk)	2.17X

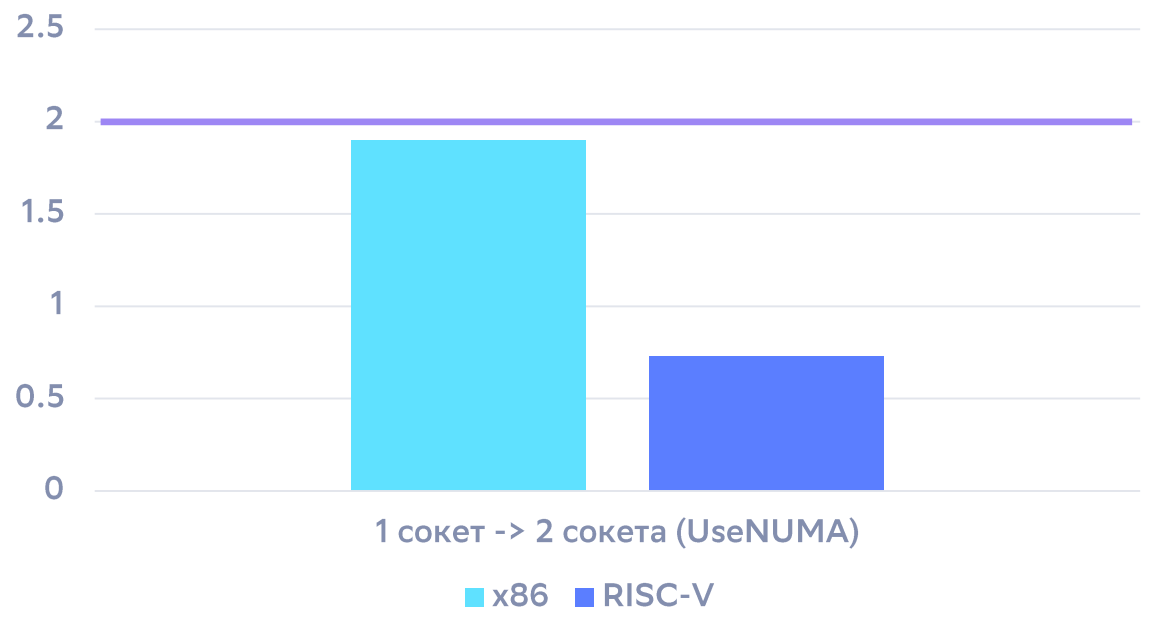
Эффект роста времени доступа к L3 при длительном запуске



Ось X: итерации
Ось Y: оценка времени доступа к одному элементу (нс)

- Существенное увеличение времени отклика для L3 при долгой нагрузке на сценарии со случайным доступом
- Рост температуры ядра не наблюдается
- 100% воспроизводимость на микробенчмарке
- Не наблюдается на x86

Масштабирование 1 сокет -> 2 сокета



RISC-V	x86
407 нс	86 нс

Обмен кеш линией между сокетами

Примечание: оценка стоимости обмена кеш линией между сокетами сделана на модификации сценария:

- Запускается 2 потока на разных сокетах, обеспечивается их последовательная работа.
- Первый поток модифицирует данные
- Второй поток измеряет время потраченное на чтение всего датасета один раз.

Запуск бенчмарка SPECjvm2008 на 2х сокетах (128 потоков) дает меньшую производительность чем на 1м сокете (64 потока) на RISC-V системе

Заключение

Производительность подсистемы кешей на рассмотренной RISC-V системы отличается:

L2: 2.66X
L3: 3.28X
память: 2.17X

Стоимость обмена кеш линией между сокетами отличается в ~4 раза

Неясен эффект роста оценки L3 латенси при длительном запуске синтетического бенчмарка со случайным доступом в память, наблюдаемый на рассмотренной RISC-V системе

Вопросы и ответы