

Московский физико-технический институт
(государственный университет)
Факультет радиотехники и кибернетики
Кафедра информатики и вычислительной техники

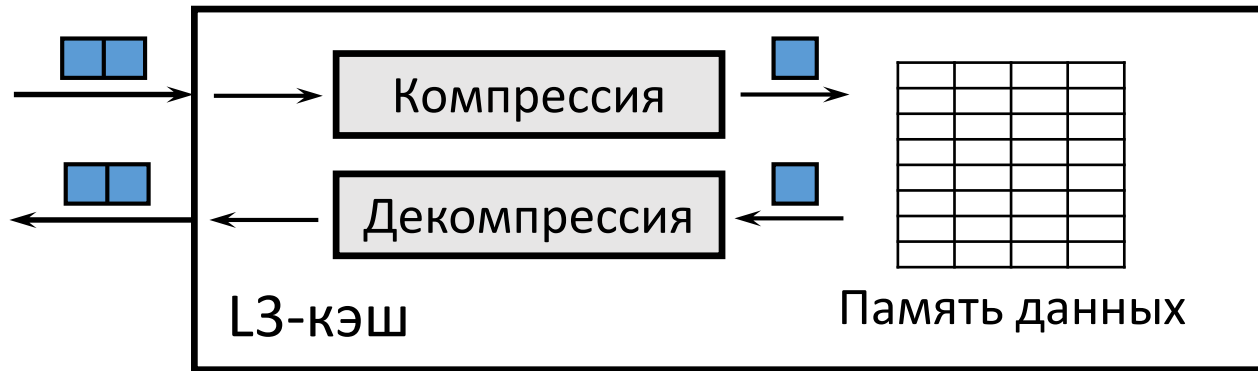
ИССЛЕДОВАНИЕ ПРИМЕНИМОСТИ КОМПРЕССИИ ДАННЫХ В КЭШ-ПАМЯТИ ПРОЦЕССОРОВ С АРХИТЕКТУРОЙ «ЭЛЬБРУС»

Выпускная квалификационная работа
(бакалаврская работа)

Студент: Сурченко А.В.
Научный руководитель: Кожин А.С.

Москва, 2018

Компрессия данных в кэш-памяти



- Применение:
Увеличение эффективного объема кэш-памяти при незначительном изменении площади и времени доступа
- Основные требования:
 - **Скорость** (задержка декомпрессии ~1-2 такта)
 - **Простота** (сохранение основной структуры кэша)
 - **Эффективность** (хорошая степень сжатия данных)
- До сих пор не имеет широкого применения в серийных процессорах
- Целесообразно использование в L3-кэше

Постановка задачи

Цель:

Исследование применимости различных алгоритмов компрессии данных в кэш-памяти третьего уровня процессоров с архитектурой «Эльбрус»

Задачи:

- Разработка RTL-описания алгоритмов компрессии на языке Verilog
- Исследование эффективности работы алгоритмов на FPGA-прототипе процессора «Эльбрус-8С2»
- Проведение пробного синтеза модулей в САПР Synopsys Design Compiler для технологии 16нм с целевой частотой 2ГГц

Алгоритмы компрессии

Название алгоритма	Задержка	Сложность	Степень сжатия
FPC, C-Pack	✗	✗	✓
ZVC, RVC	✓	✓	✗
B+ Δ , B Δ I	✓	✓	✓

- FPC – Frequent Pattern Compression
- C-Pack – Cache Packer
- ZVC – Zero values compression

- RVC – Repeated values compression
- B+ Δ – Base+Delta
- B Δ I – Base-Delta-Immediate

Распространенные формы данных

Нулевые значения: нулевые указатели, разреженные матрицы

0x00000000	0x00000000	0x00000000	0x00000000	...
------------	------------	------------	------------	-----

Повторяющиеся значения: соседние пиксели

0x0200A3FF	0x0200A3FF	0x0200A3FF	0x0200A3FF	...
------------	------------	------------	------------	-----

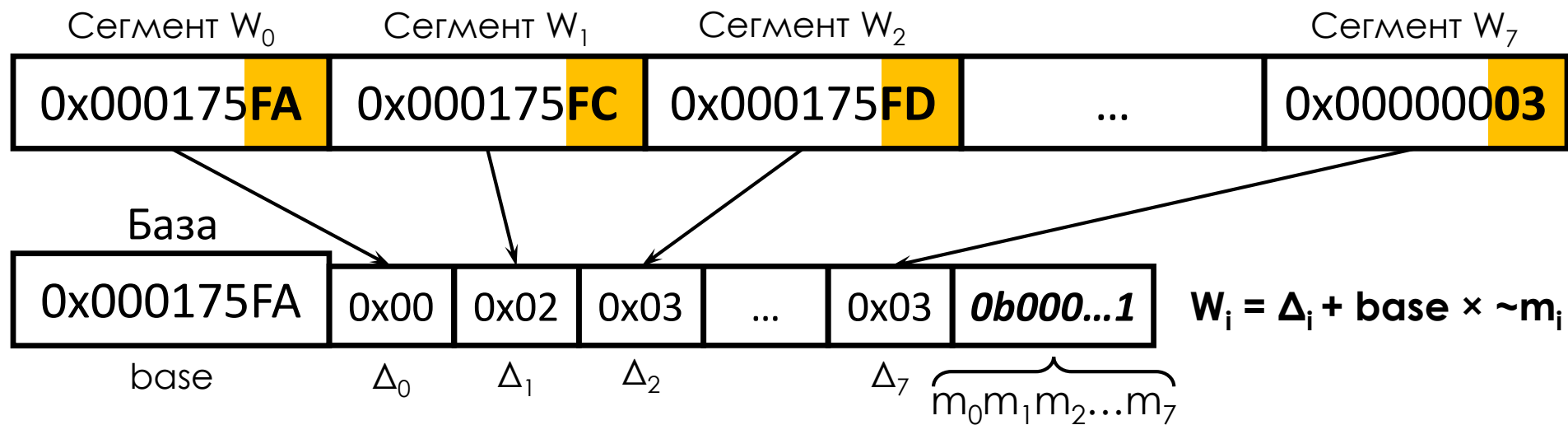
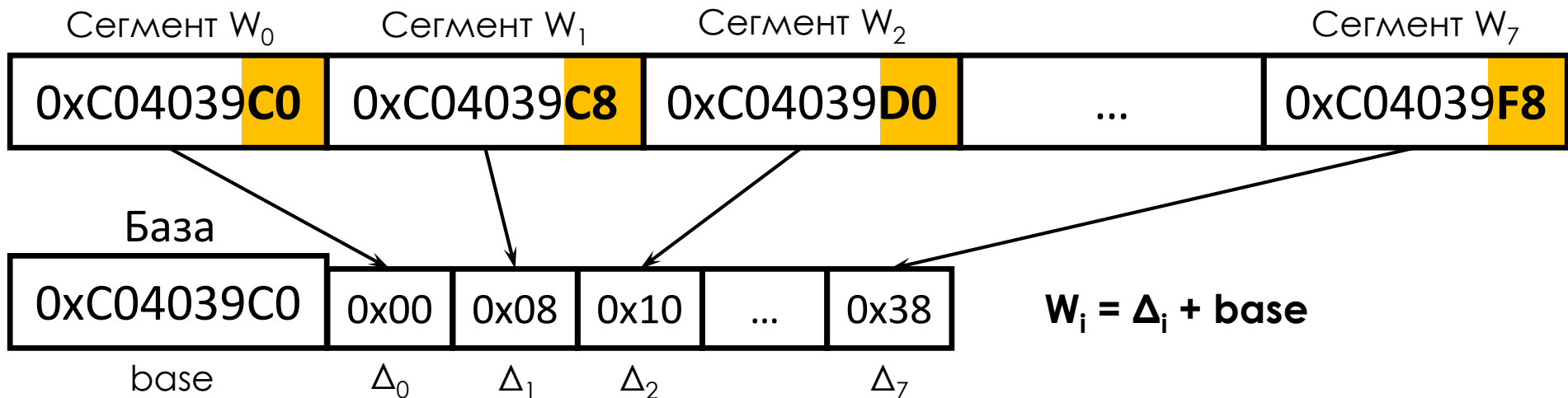
Малые значения: малые величины с типом большого размера

0x00000000 00	0x00000000 0B	0x00000000 03	0x00000000 04	...
----------------------	----------------------	----------------------	----------------------	-----

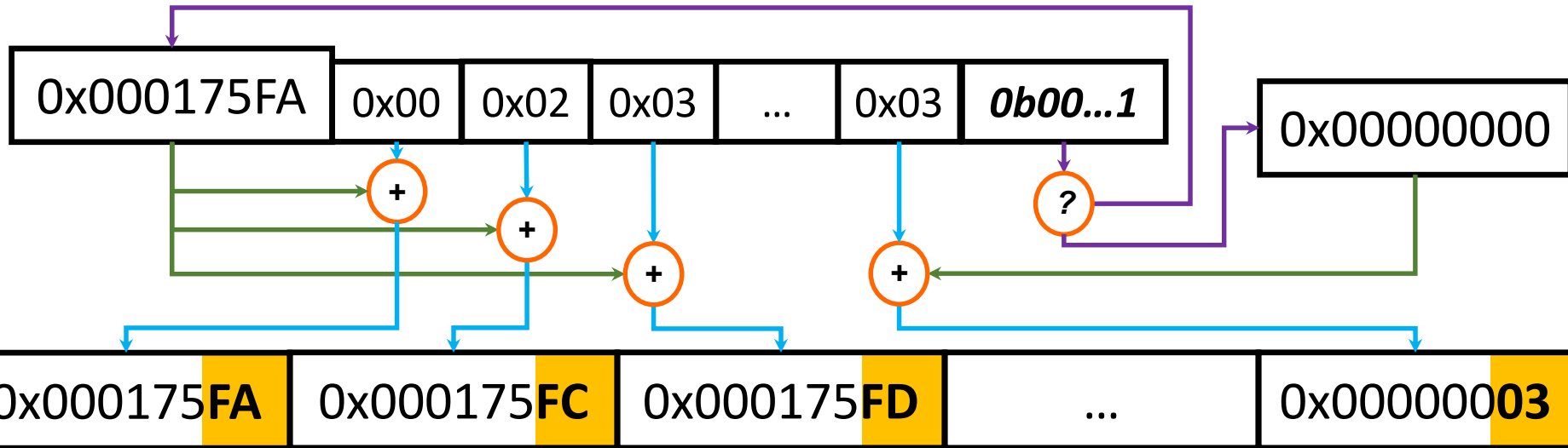
Другие формы: указатели на ту же область в памяти

0xC04039 C0	0xC04039 C8	0xC04039 D0	0xC04039 D8	...
--------------------	--------------------	--------------------	--------------------	-----

Алгоритмы В+Δ и ВΔI



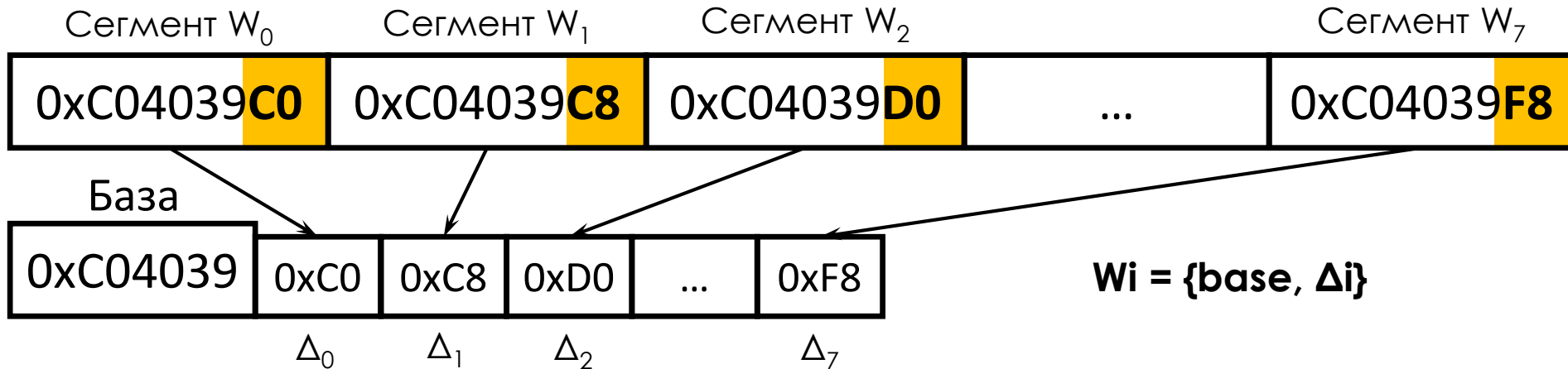
Декомпрессия ВДІ



- Используются простые логические операции, выполнение которых не занимает много времени
- Восстановление исходного значения каждого из сегментов происходит параллельно

Алгоритмы $V+\Delta^*$ и $V\Delta I^*$

- Модификация $V+\Delta$ и $V\Delta I$ – используют конкатенацию вместо сложения



- Конкатенация является более простой операцией, поэтому для реализации алгоритмов $V+\Delta^*$ и $V\Delta I^*$ требуется значительно меньшее число логических элементов, чем для сложения
- Использование конкатенации позволяет сократить размер базы на размер одного смещения и тем самым уменьшить размер сжатой кэш-строки

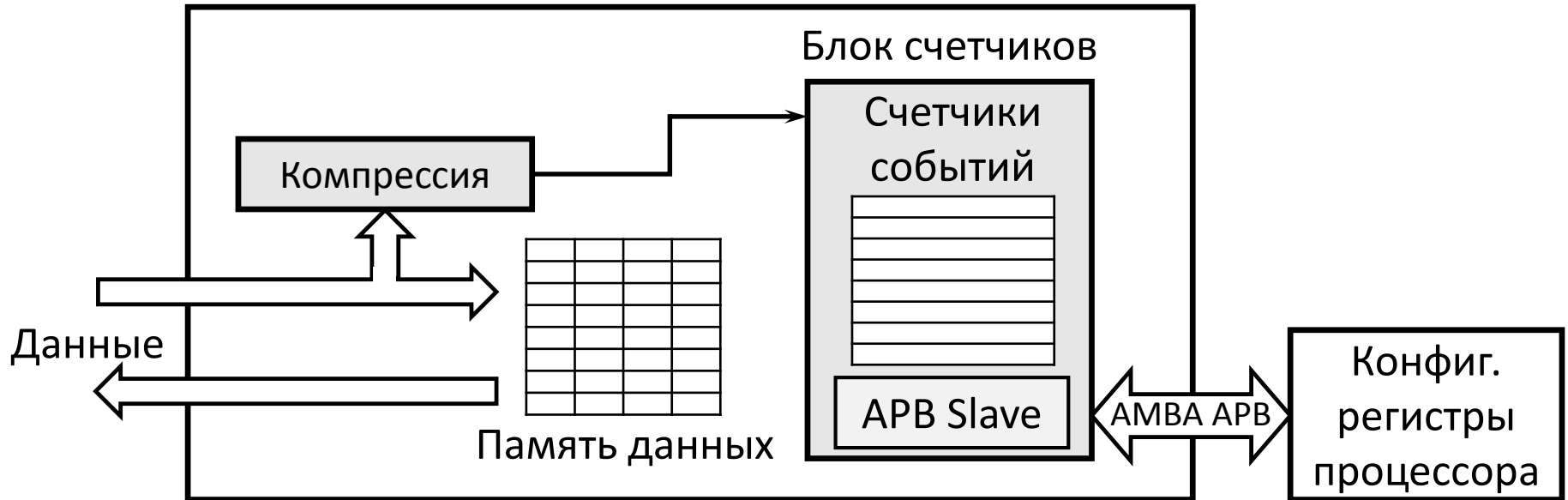
Реализованные схемы компрессии алгоритмов В+Δ, ВΔI, В+Δ* и ВΔI* для кэш-строк размером 64 Б

Название схемы	Размер сегмента, Б	Размер смещения, Б	Размер сжатой строки для В+Δ / ВΔI, Б	Размер сжатой строки для В+Δ* / ВΔI*, Б
Zero values	0	0	0 / 0	0 / 0
Repeated values	8	0	8 / 8	8 / 8
Base2-Δ1	2	1	34 / 38	33 / 37
Base4-Δ1	4	1	20 / 22	19 / 20
Base4-Δ2	4	2	36 / 38	34 / 36
Base8-Δ1	8	1	16 / 17	15 / 16
Base8-Δ2	8	2	24 / 25	22 / 23
Base8-Δ4	8	4	40 / 41	36 / 37
Нет компрессии	—	—	64 / 64	64 / 64

При срабатывании нескольких схем компрессии выбирается та, которая дает наилучшее сжатие

Анализ эффективности

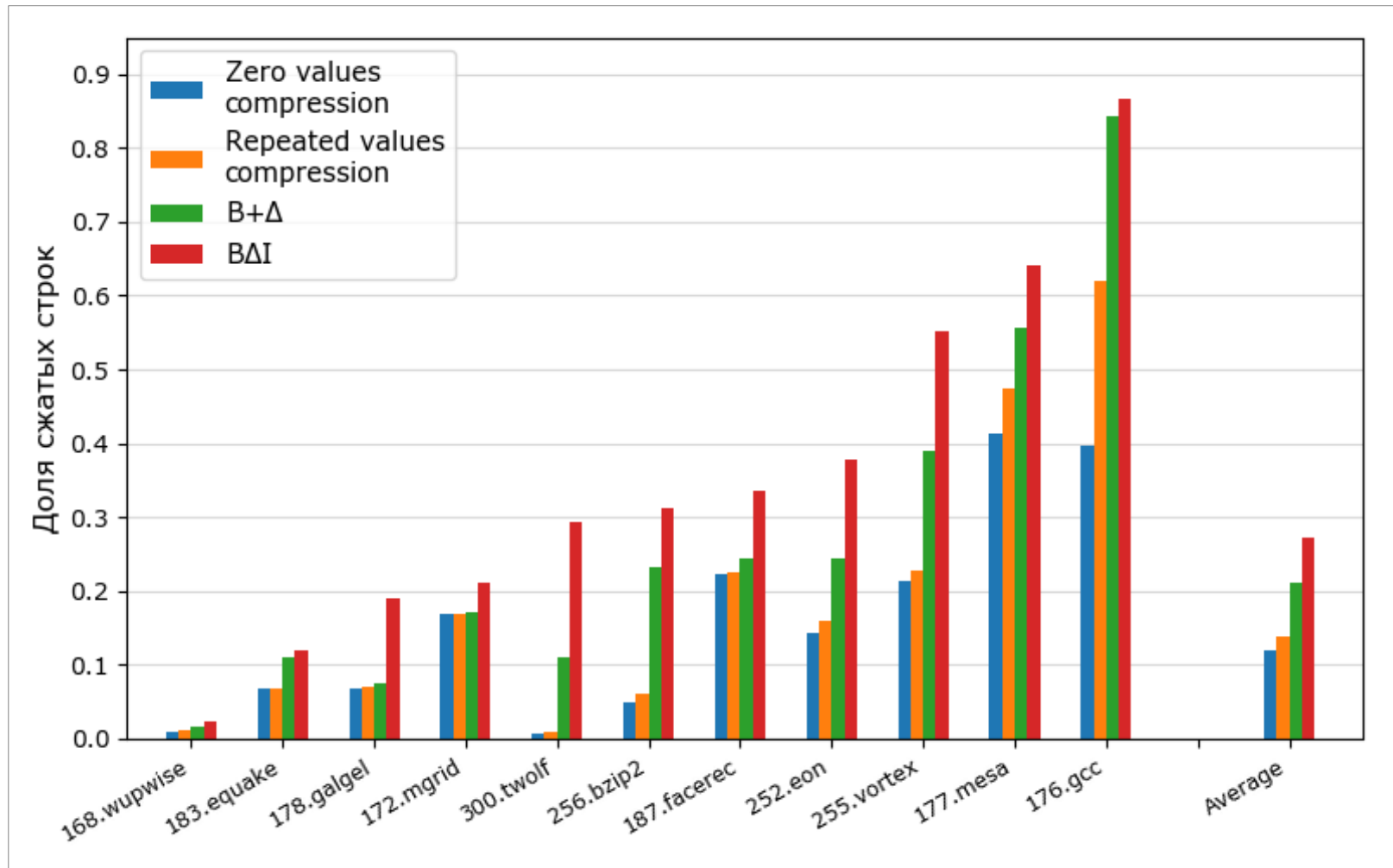
Банк L3-кэша в составе FPGA-прототипа



- Алгоритмы компрессии были интегрированы в L3-кэш FPGA-прототипа процессора «Эльбрус-8С2» без занесения сжатых данных в память
- Была разработана система мониторинга, позволяющая собирать данные о работе схем компрессий через программно-доступные счетчики событий
- Использовались задачи пакета SPEC CPU2000, запускаемые в однопоточном режиме

Результаты измерений

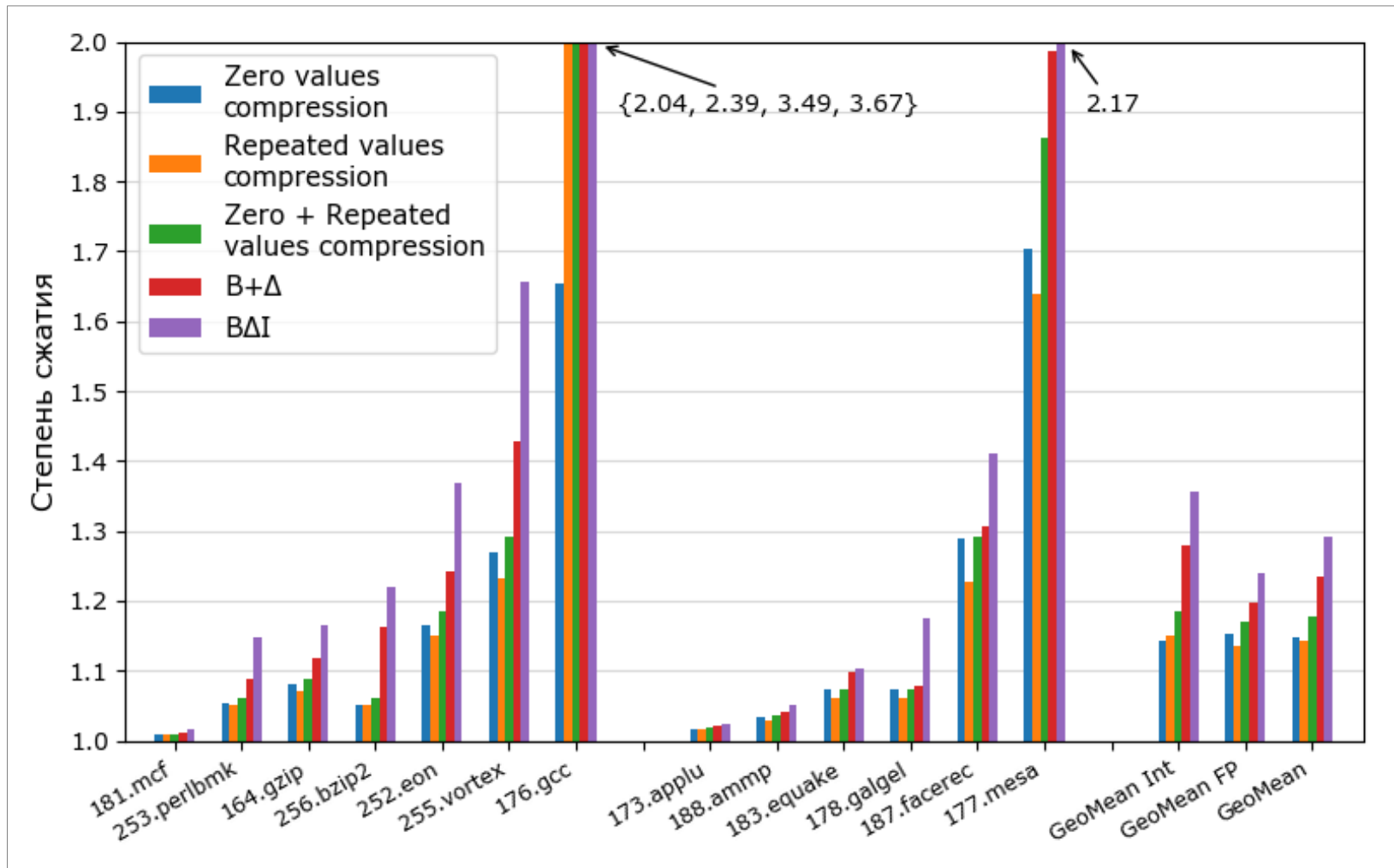
Доля сжатых кэш-строк для разных алгоритмов компрессии



Алгоритмы B+Δ и BΔI дают больший, чем ZVC и RVC, процент сжатых строк (21,1% и 27,3% против 12,0% и 13,9%)

Результаты измерений

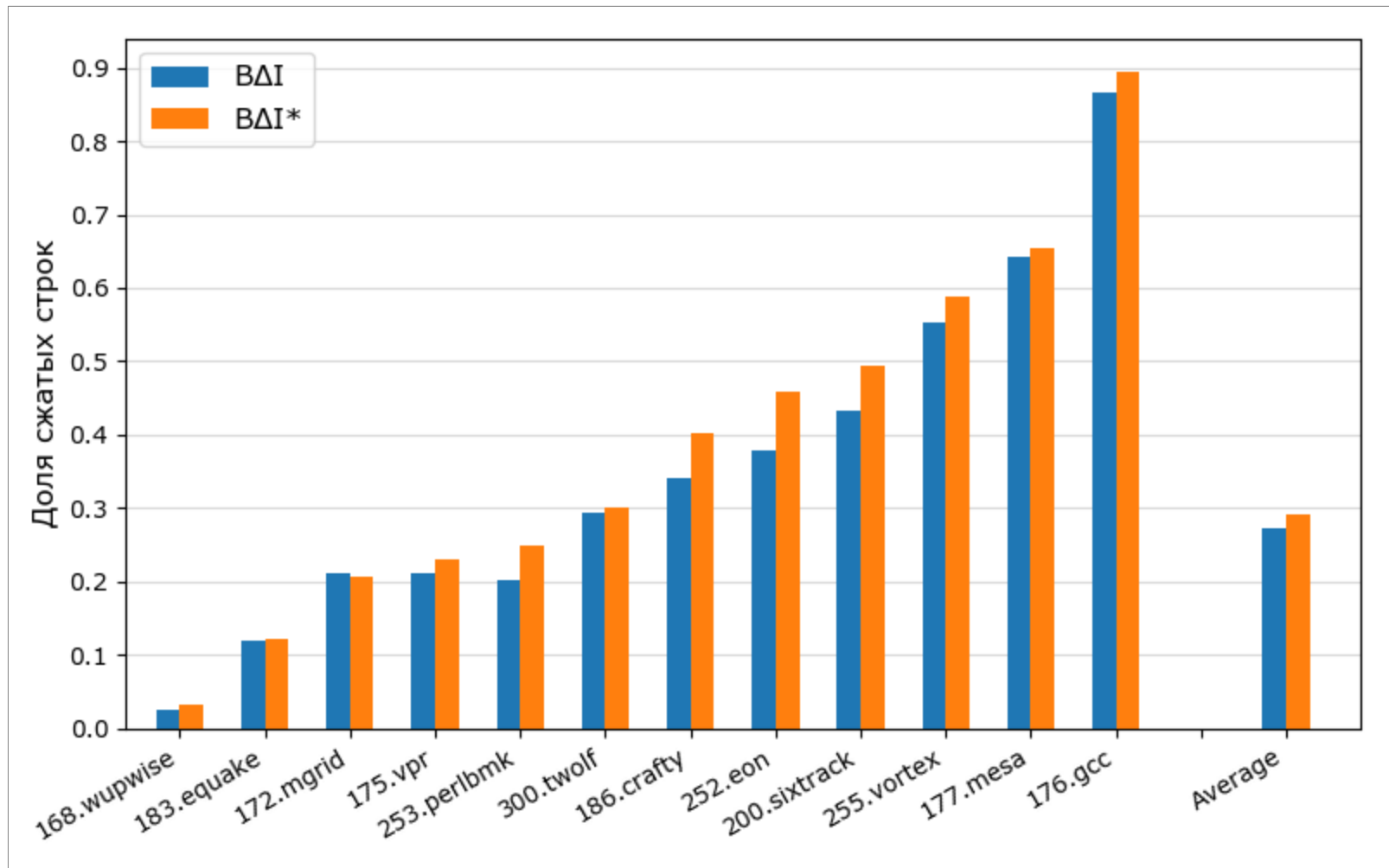
Степень сжатия для разных алгоритмов компрессии



Лучшая сжимаемость данных наблюдается для целочисленных задач (для B+Δ: 1,279 для Int и 1,197 для FP; для BΔI: 1,356 для Int и 1,239 для FP)

Результаты измерений

Сравнение работы алгоритмов ВΔI и ВΔI*



Процент сжимаемых строк для алгоритма ВΔI* по сравнению с ВΔI выше на 1,8%

Лишь для 4 задач из 26 алгоритм ВΔI сжимает больше строк, чем ВΔI*

Результаты пробного синтеза

Технология 16нм

	Компрессия		Декомпрессия	
	ВΔI	ВΔI*	ВΔI	ВΔI*
Задержка крит. пути:	580ps	350ps	240ps	130ps
Число логических элементов:	34600	8900	30100	3700
Площадь отн. банка L3:	0,22%	0,08%	0,21%	0,03%

Выводы:

- Алгоритм ВΔI* по сравнению с ВΔI требует меньше ресурсов и имеет меньшую задержку критического пути
- Все операции за исключением компрессии по алгоритму ВΔI укладываются в один такт целевой частоты (2 ГГц)

Результаты работы

- Предложены модифицированные алгоритмы $V+\Delta^*$ и $V\Delta I^*$
- Разработано RTL-описание алгоритмов компрессии и декомпрессии ZVC, RVC, $V+\Delta$, $V\Delta I$, $V+\Delta^*$ и $V\Delta I^*$
- Разработана аппаратура для измерения эффективности различных алгоритмов компрессии на прототипе процессора «Эльбрус-8С2»
- Проведен пробный синтез алгоритмов $V\Delta I$ и $V\Delta I^*$ по технологии 16нм
- Анализ результатов измерений и синтеза показал, что алгоритм $V\Delta I^*$ является наиболее подходящим для применения с целью компрессии данных в L3-кэше процессоров с архитектурой «Эльбрус»

Дополнительные материалы

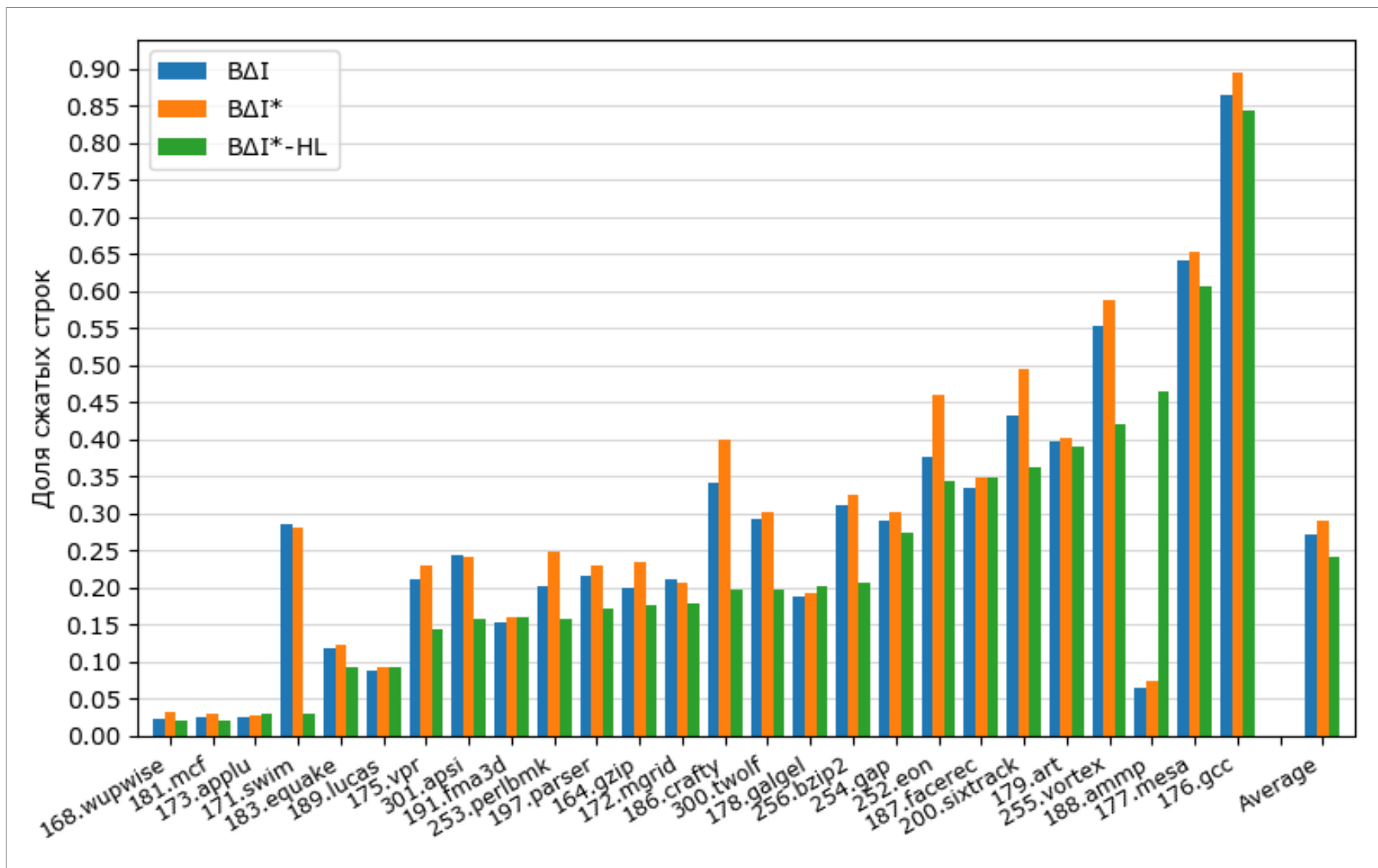
Алгоритм VDI*-HL

Введено требование сжатие кэш-строки более, чем до половинного размера

Название схемы	Размер сегмента, Б	Размер смещения, Б	Размер маски, Б	Размер сжатой строки для VDI*, Б	Код
Zero values	0	0	0	0	000
Base4-Δ1	4	1	2	21	001
Base16-Δ4	16	4	1	29	010
Base8-Δ3	8	3	1	30	011
Base2-Δ1	2	7 бит	4	33	100
Нет компрессии	–	–	–	68	111

С учетом того, что из-за тэгов защищенного режима фактический размер кэш-строки составляет 68 байт, требуется сжатие до <34Б

Алгоритм $\text{ВДI}^*\text{-HL}$

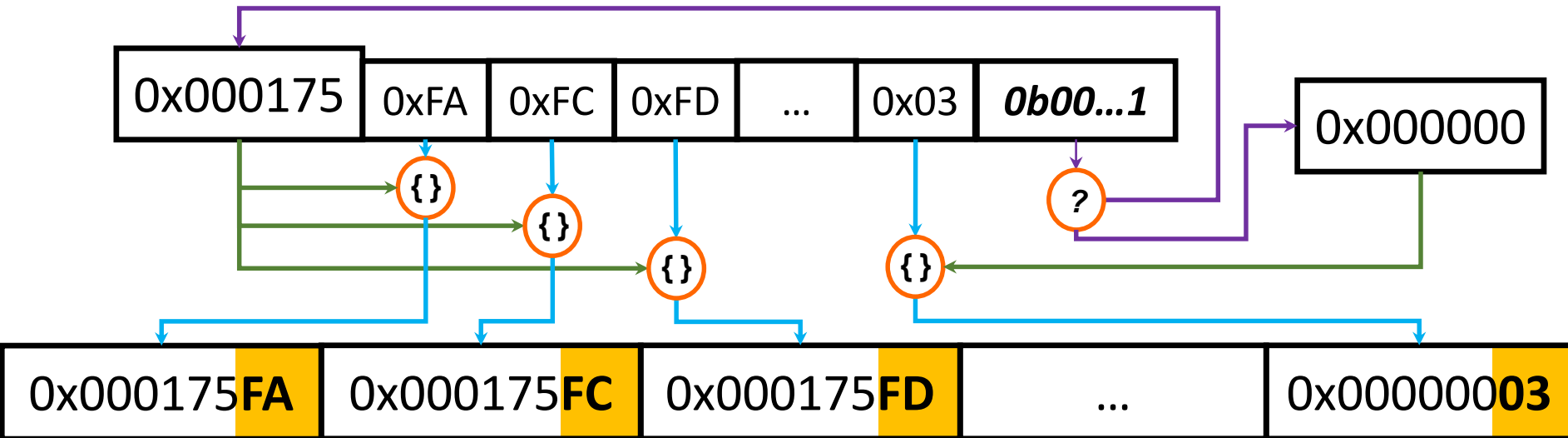


Падение доли сжатия для алгоритма $\text{ВДI}^*\text{-HL}$ по сравнению с ВДI^* равно 4,9%

Теги защищенного режима

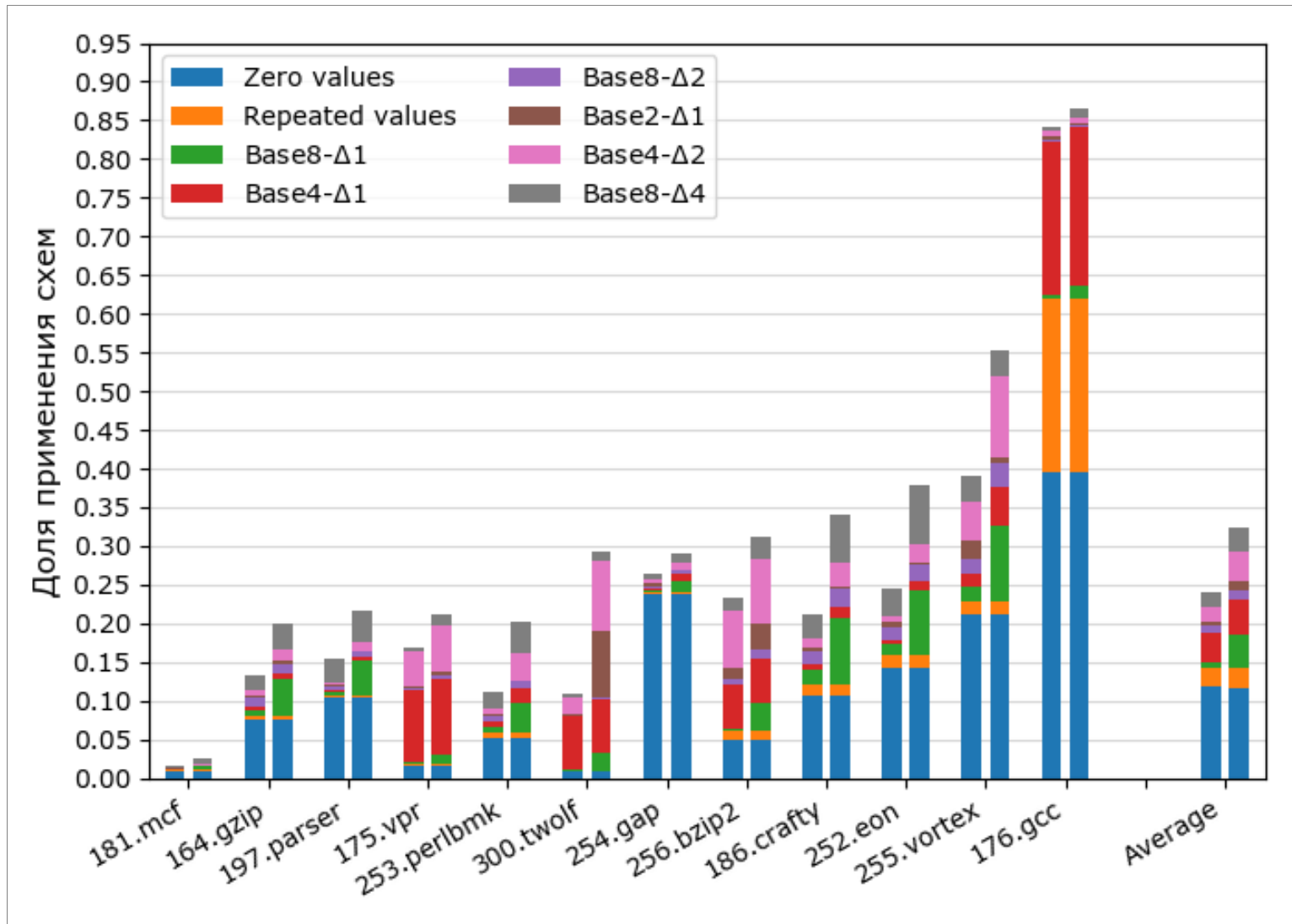
- Каждое четырехбайтовое слово расширяется двухразрядным тегом
- Физический размер кэш-строки:
68B = 64B данных и 4B тегов
- Теги сжимаются отдельно от остальных данных и только по схеме Zero values compression

Декомпрессия ВДІ*



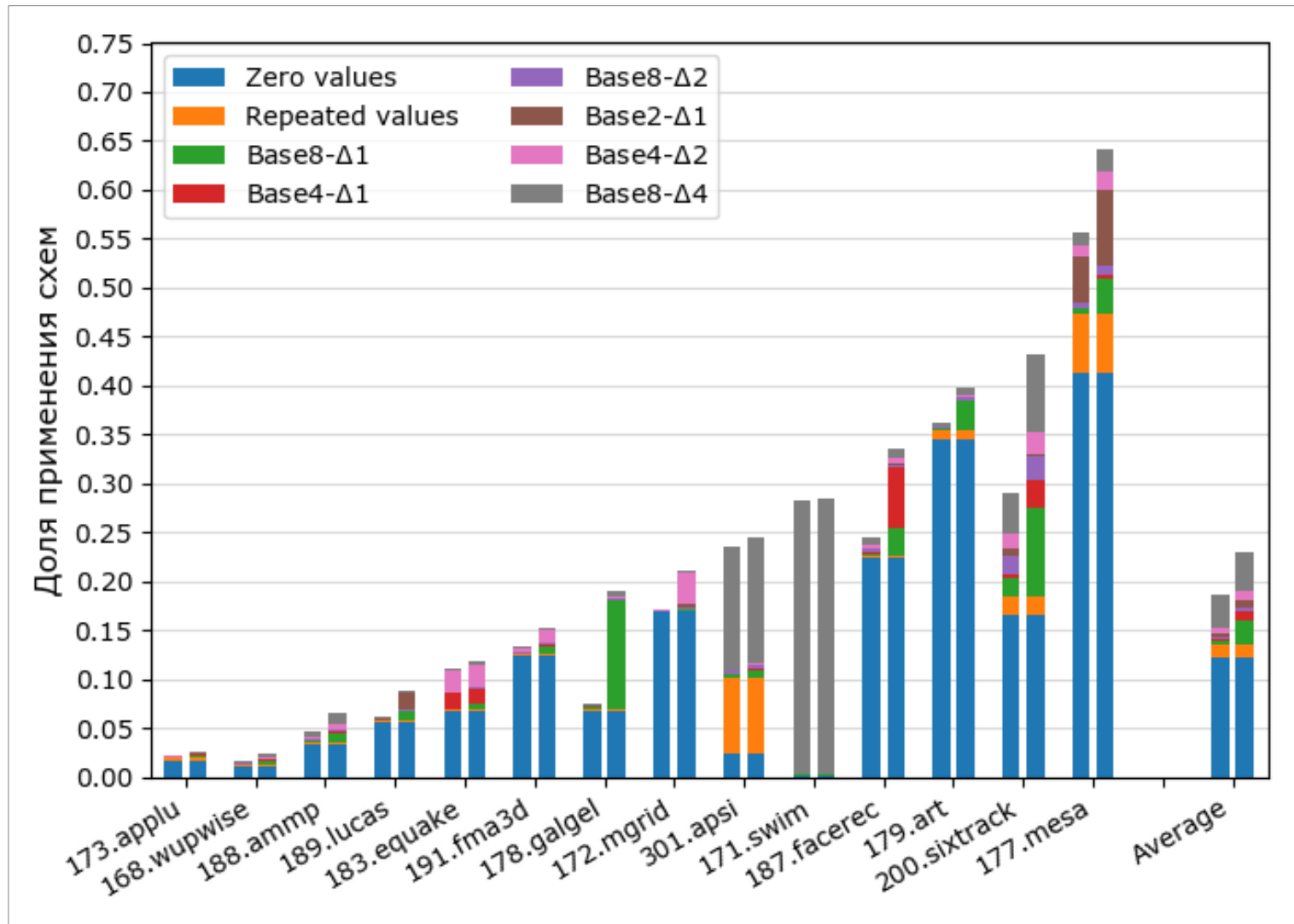
Результаты измерений

Вклад разных схем компрессии алгоритмов В+Δ и ВΔ1 для Int-задач



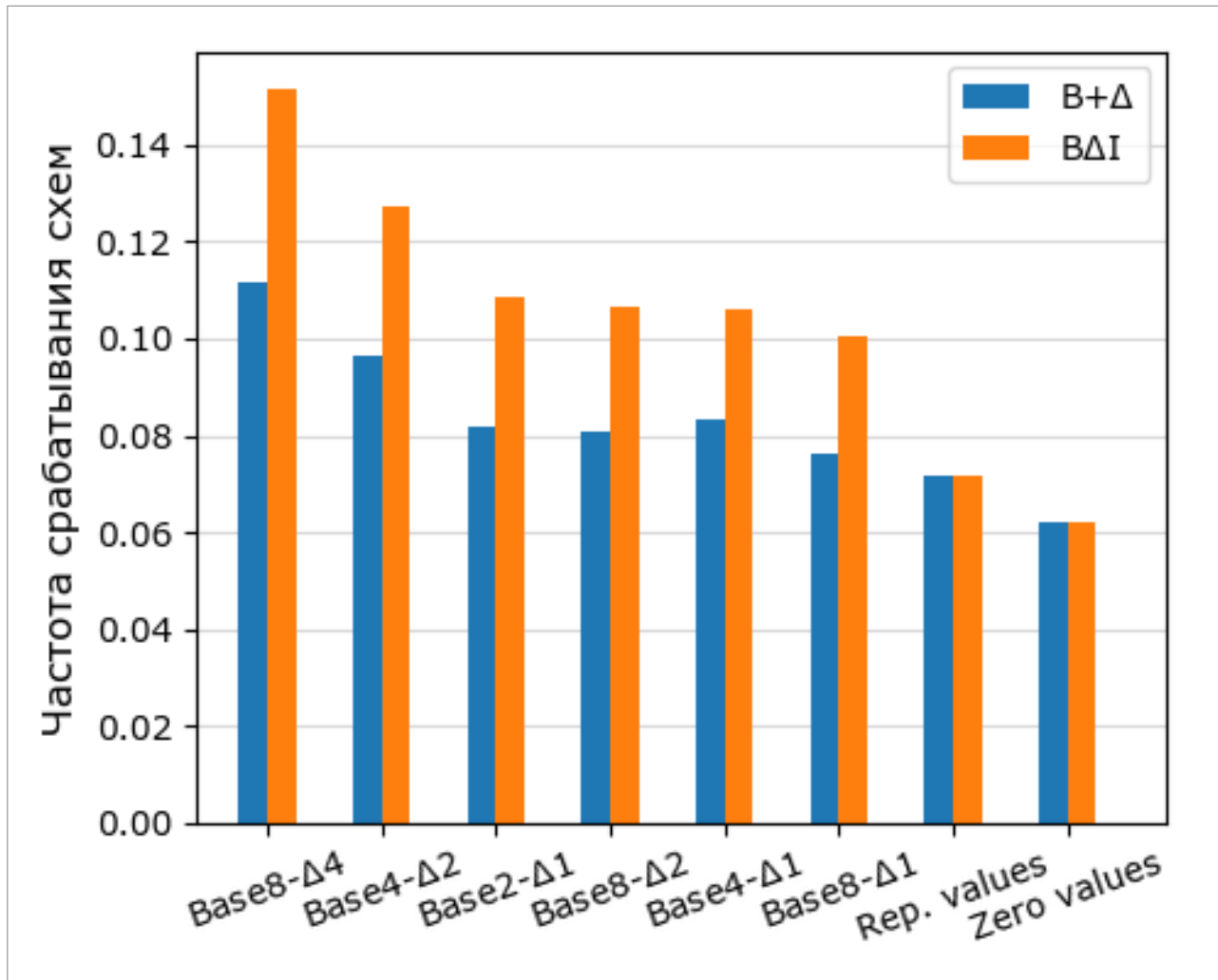
Результаты измерений

Вклад разных схем компрессии алгоритмов В+Δ и ВΔI для FP-задач



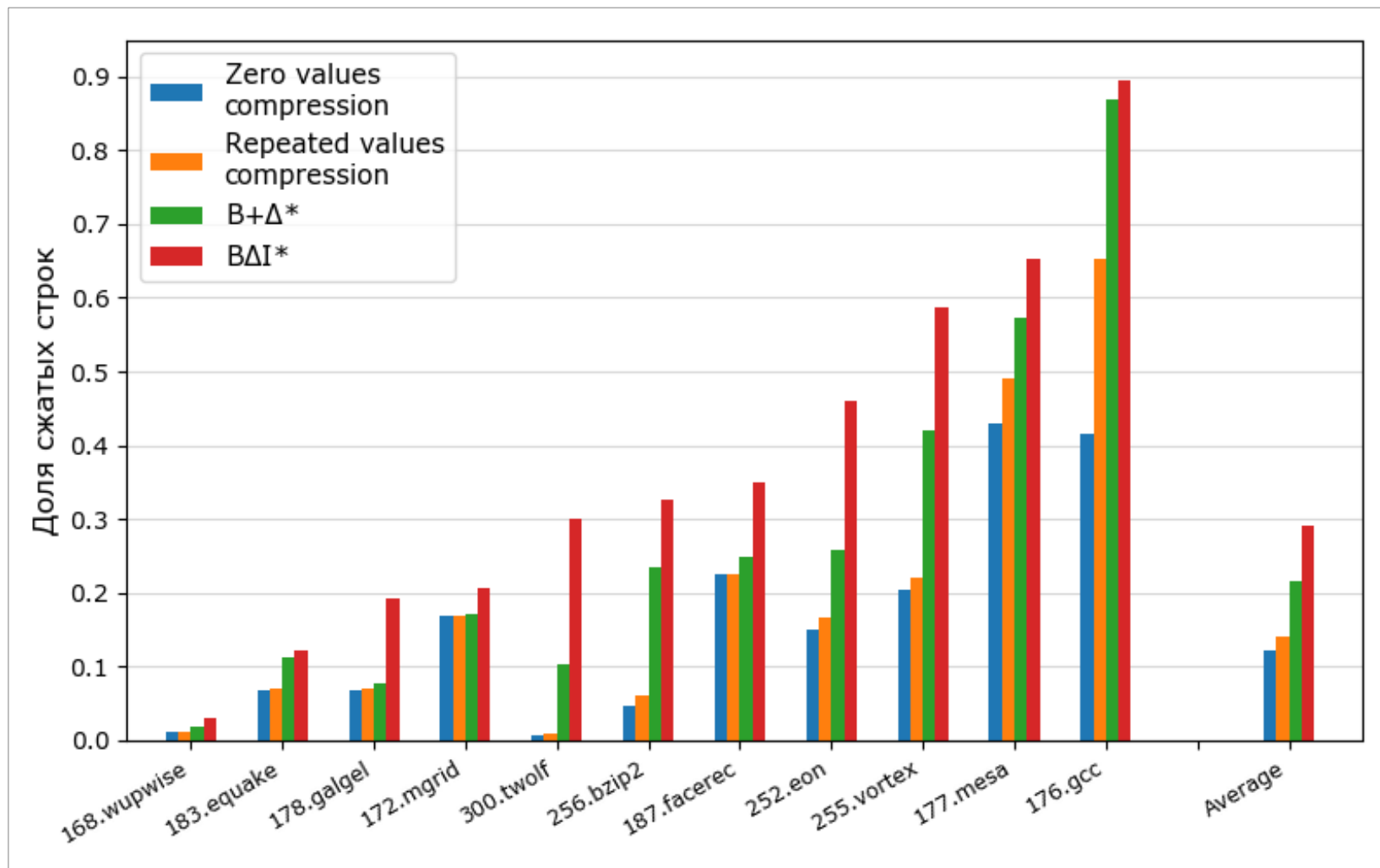
Результаты измерений

Частота применения разных схем компрессии алгоритмов $B+\Delta$ и $B\Delta I$



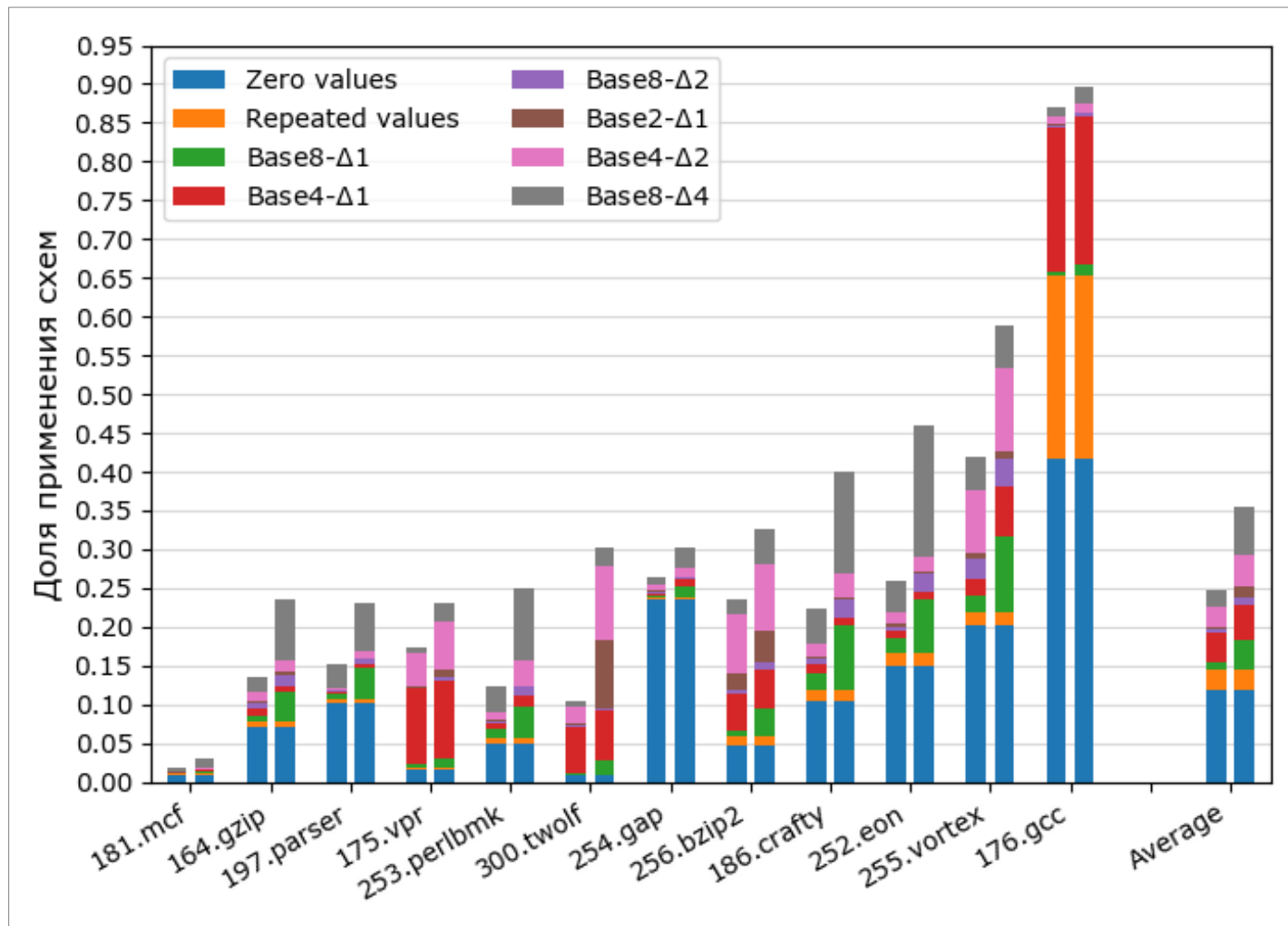
Результаты измерений

Доля сжатых кэш-строк для разных алгоритмов компрессии



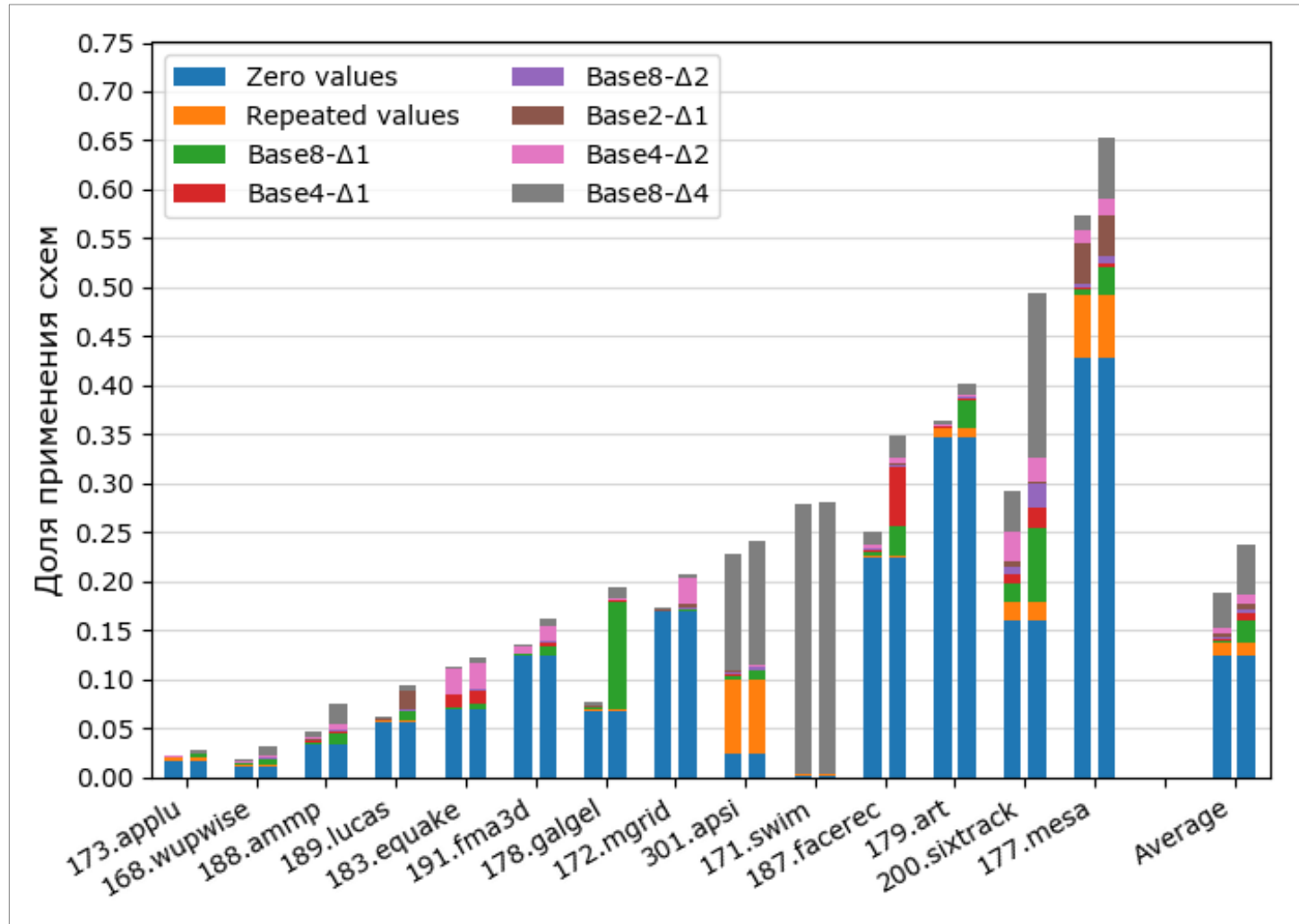
Результаты измерений

Вклад разных схем компрессии алгоритмов $V+\Delta^*$ и $V\Delta I^*$ для Int-задач



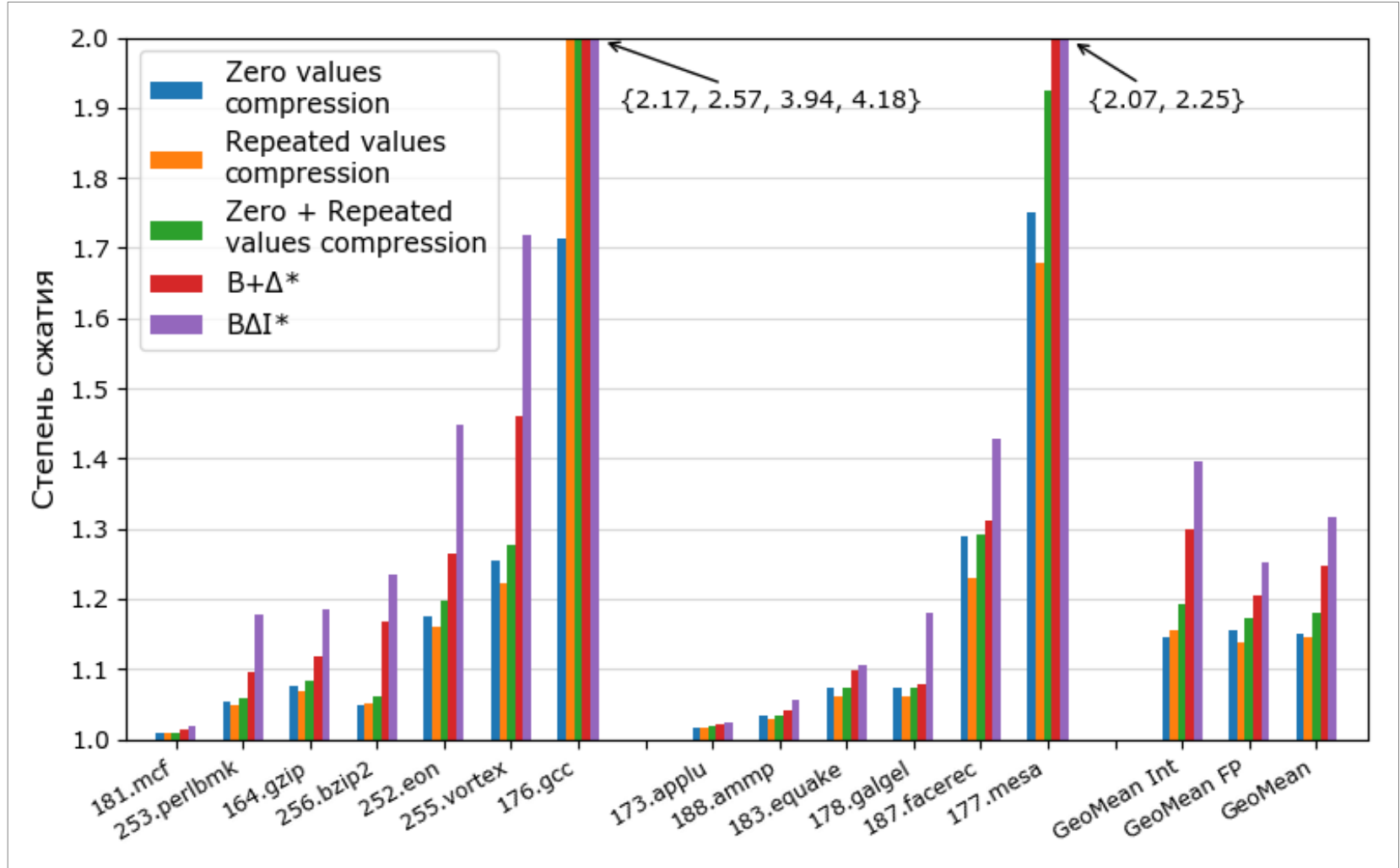
Результаты измерений

Вклад разных схем компрессии алгоритмов $V+\Delta^*$ и $V\Delta I^*$ для FP-задач



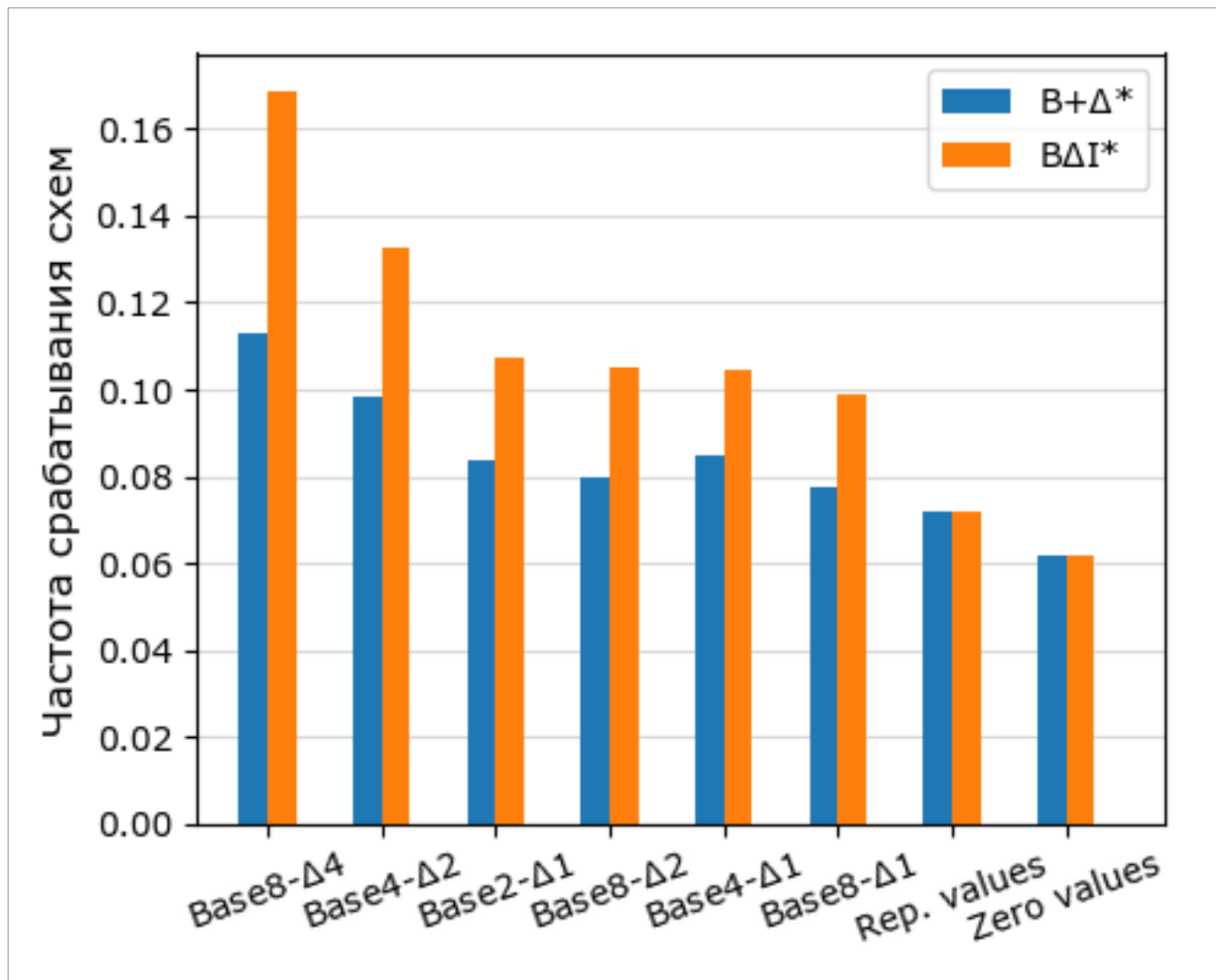
Результаты измерений

Степень сжатия для разных алгоритмов компрессии

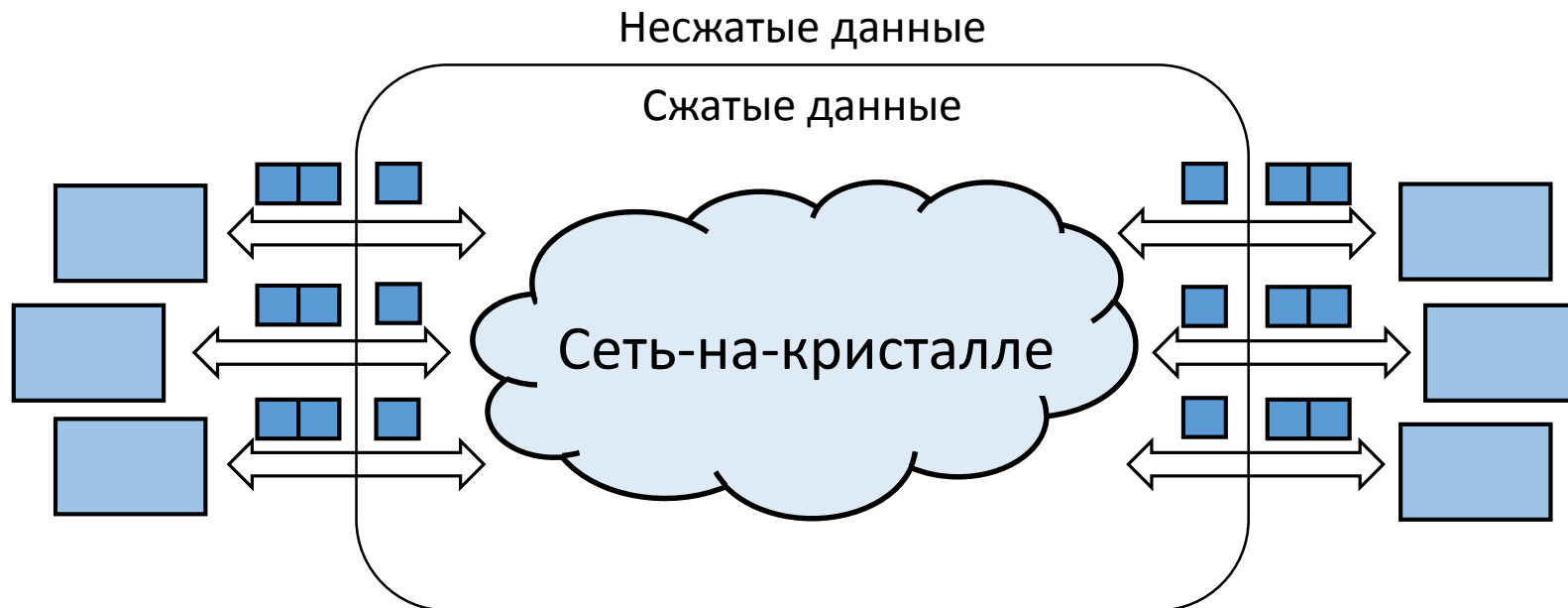


Результаты измерений

Частота применения разных схем компрессии алгоритмов $B+\Delta$ и $B\Delta I$



Возможности применения



- При большом числе подключенных к сети-на-кристалле абонентов ее пропускная способность ограничивает производительность системы
- Компрессия позволит уменьшить размер пакетов данных, передаваемых по сети, повышая тем самым пропускную способность

Реализация алгоритмов

