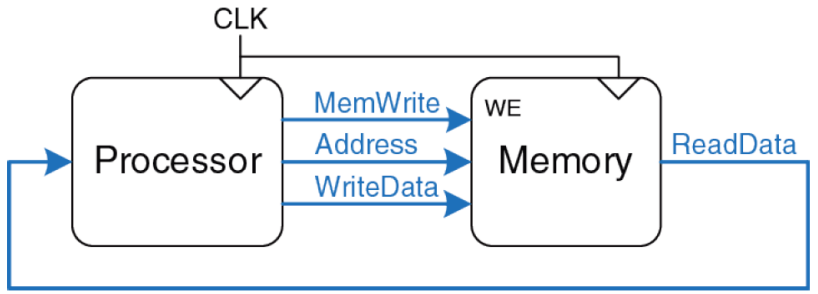
Архитектура компьютера

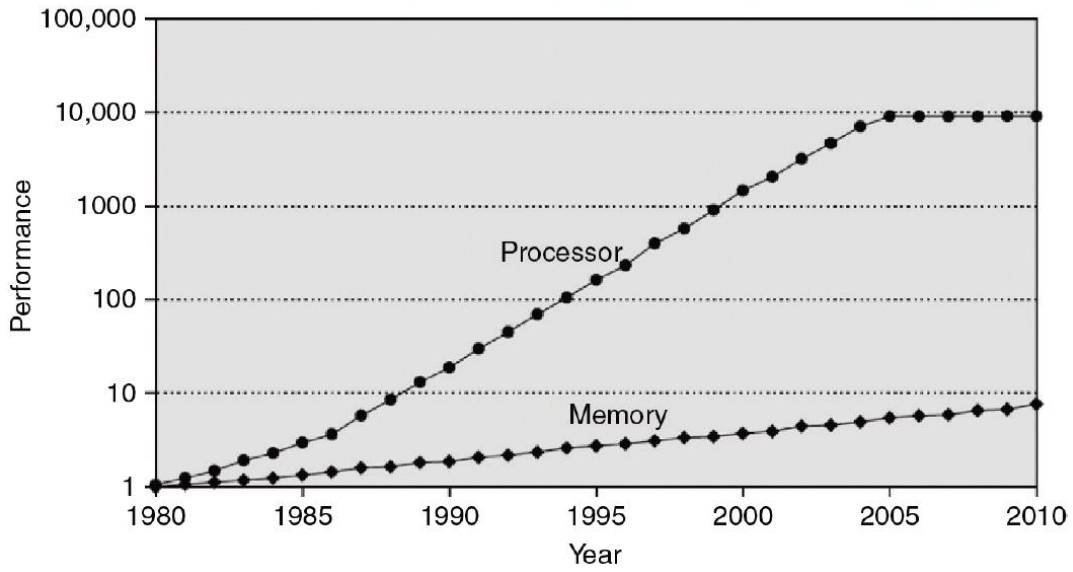
Кэш-память

План лекции

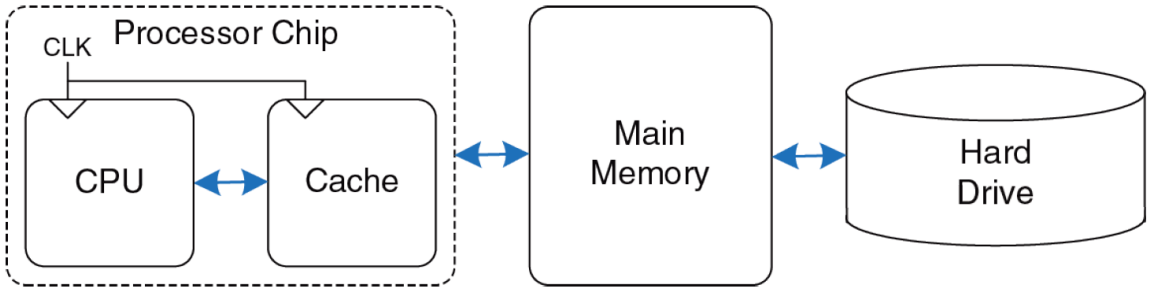
- Характеристики кэш-памяти
- Кэш прямого отображения
- Множественно-ассоциативный кэш
- Полностью ассоциативный кэш
- Алгоритмы замещения данных
- Основные оптимизации кэш-памяти

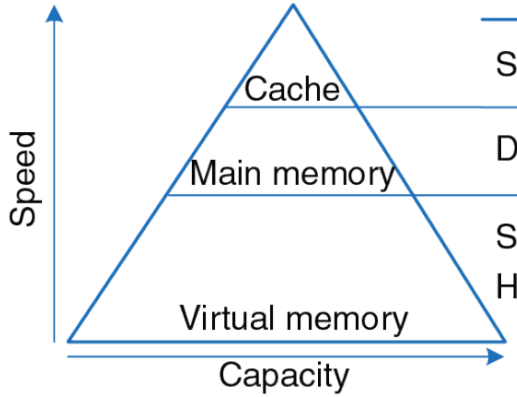
Процессор–память





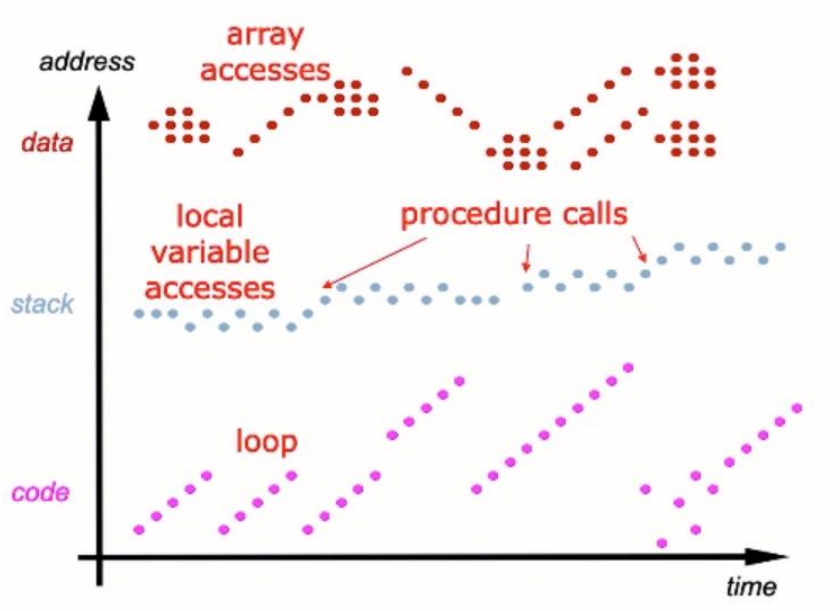
Иерархия памяти





Technology	Price / GB	Access Time (ns)	Bandwidth (GB/s)
SRAM	$10,000	1	25+
DRAM	$10	10–50	10
SSD	$1	100,000	0.5
HDD	$0.1	10,000,000	0.1

Пространственная и временная локальность данных



Анализ производительности

- Доля попаданий (hit rate – HR)
- Доля промахов (miss rate – MR)
- $MR = \frac{\text{Число промахов}}{\text{Общее число доступов к памяти}} = 1 - HR$
- $HR = \frac{\text{Число попаданий}}{\text{Общее число доступов к памяти}} = 1 - MR$
- AMAT – average memory access time
- $AMAT = t_{cache} + MR_{cache} \cdot (t_{MM} + MR_{MM} \cdot t_{VM})$
- Пример

Уровень памяти	Время доступа в тактах	Процент промахов
Кэш-память	1	10%
Оперативная память	100	0%

- $AMAT = 1 + 0.1 \cdot (100) = 11$
- Какой должен быть MR, чтобы снизить AMAT до 1.5 тактов?
- $1 + m \cdot (100) = 1.5 \quad -> \quad m = 0.5\%$

Четыре вопроса по иерархии памяти

1. Где могут быть размещены данные в кэш-памяти? (Размещение строки)
2. Как найти данные в кэш-памяти (Идентификация строки)
3. Какие данные нужно заместить, при заполненной кэш-памяти? (Замещение строки)
4. Что происходит при записи в кэш-память? (Стратегия записи)

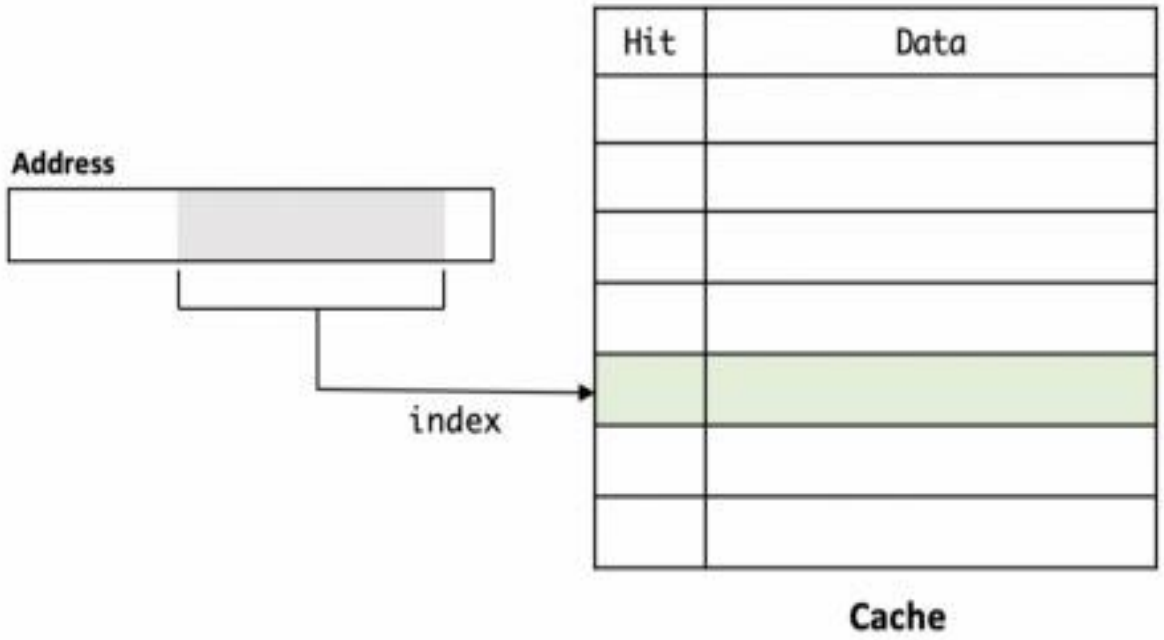
Характеристики кэш-памяти

- Ёмкость – C (capacity)
- Число наборов – S (set)
- Длинна строки (блока) – b (block)
- Количество строк (блоков) – B = C/b
- Степень ассоциативности – N

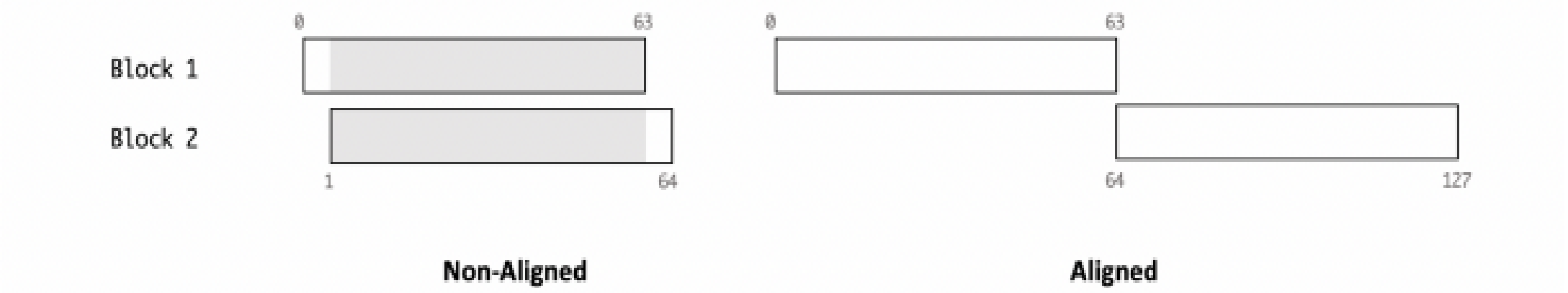
- Кэш состоит из S наборов, каждый из которых содержит одну или несколько строк
- Взаимосвязь между адресом в памяти и расположением в кэш называется отображением
- Каждый адрес в памяти отображается в один и тот же набор кэша

- Кэш прямого отображения – Набор S содержит только одну строку – $S = B$
- Множественно-ассоциативный кэш – Каждый набор S состоит из N строк – $S = B/N$
- Полностью ассоциативный кэш – Имеет только один набор $S = 1$

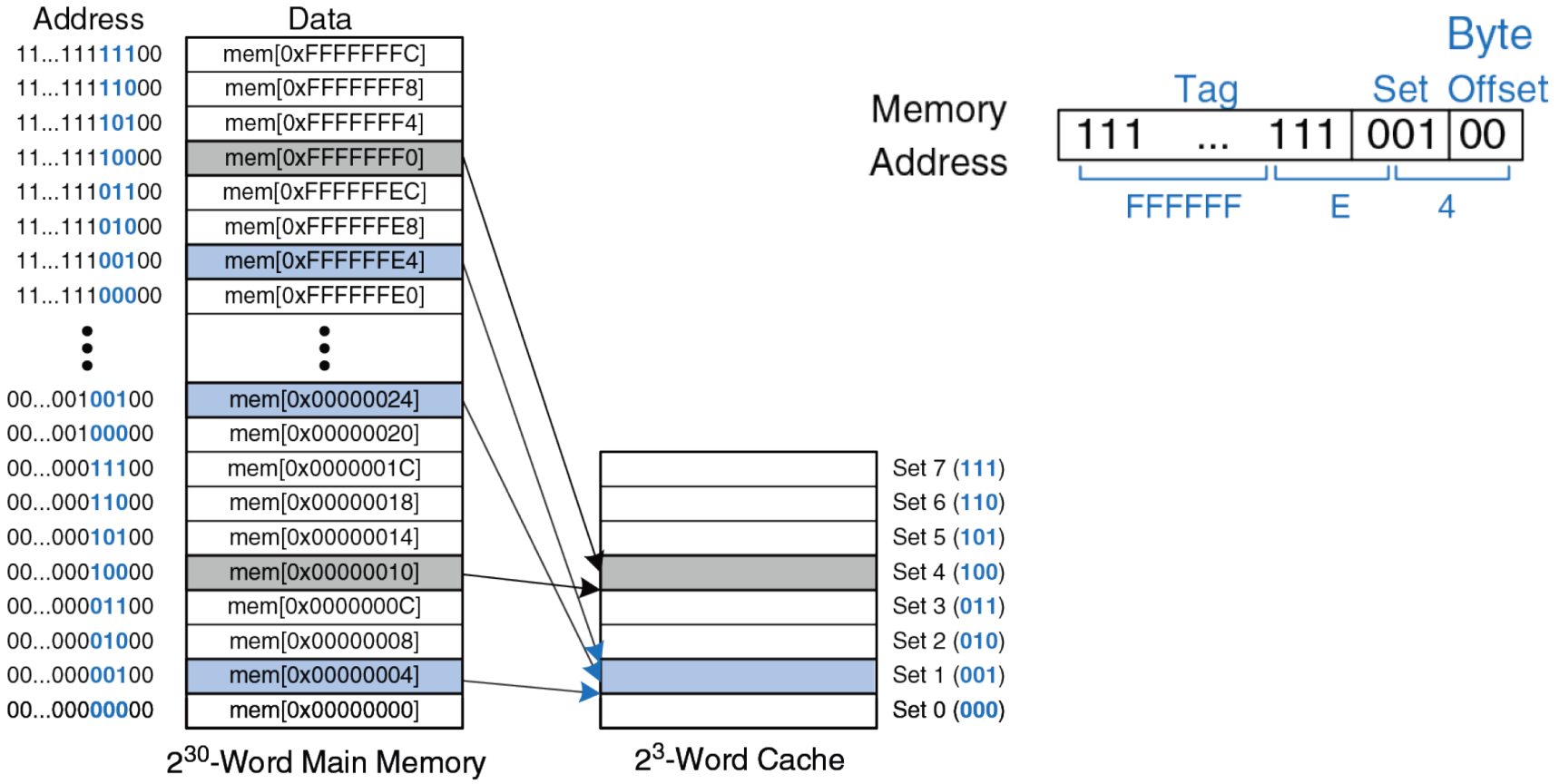
Организация кэш



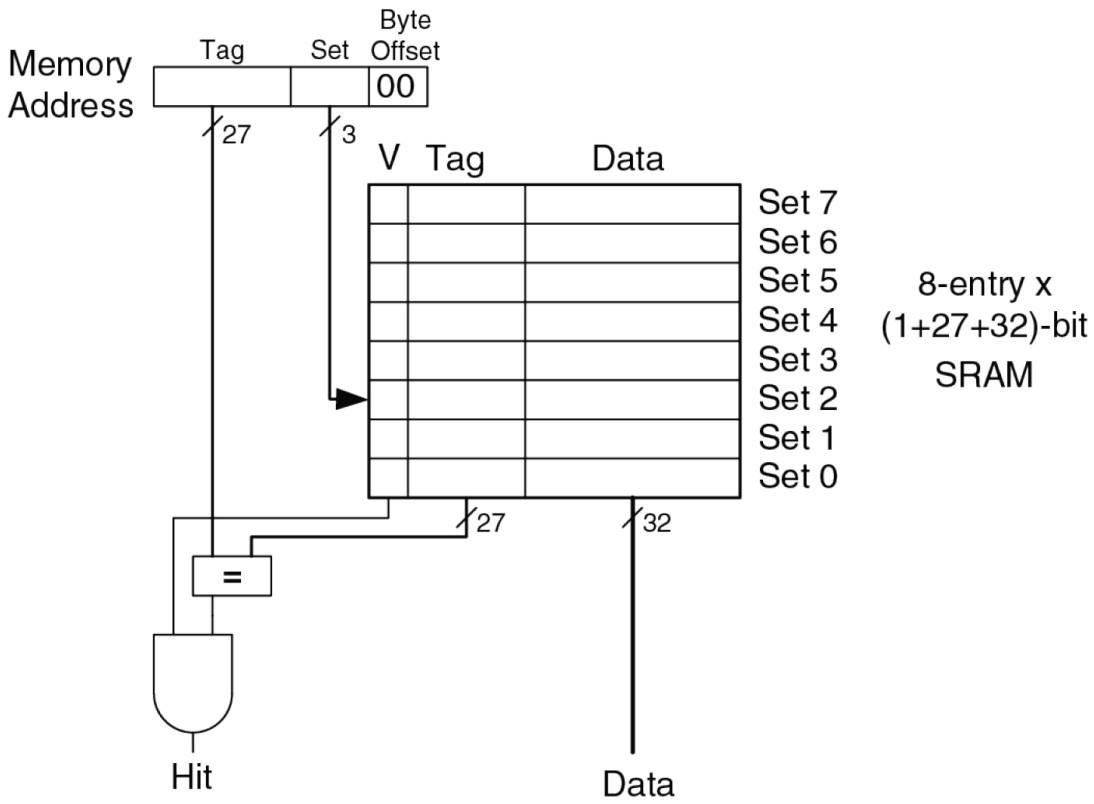
Выровненный начальный адрес блока



Кэш прямого отображения

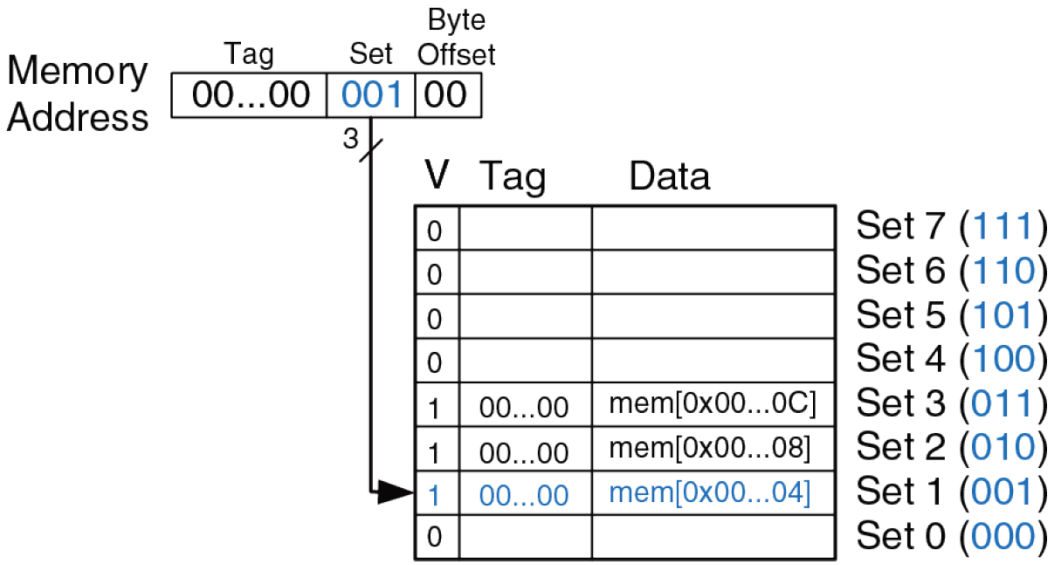


Кэш прямого отображения



Пример

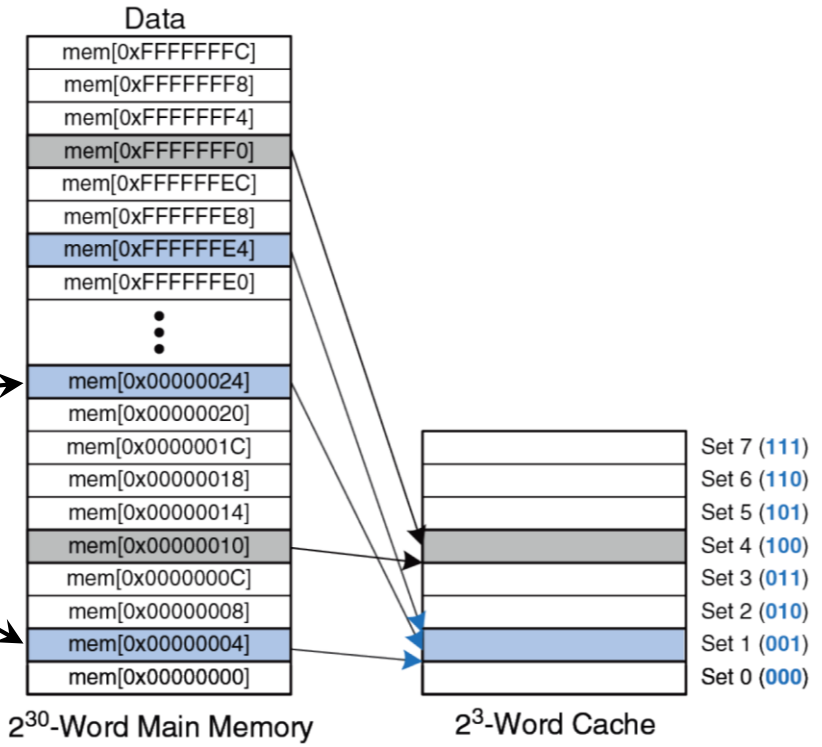
```
        addi $t0, $0, 5
loop:   beq  $t0, $0, done
        lw   $t1, 0x4($0)
        lw   $t2, 0xC($0)
        lw   $t3, 0x8($0)
        addi $t0, $t0, -1
        j    loop
done:
```



$$MR = \frac{\text{Число промахов}}{\text{Общее число доступов к памяти}} = \frac{3}{15} = 20\%$$

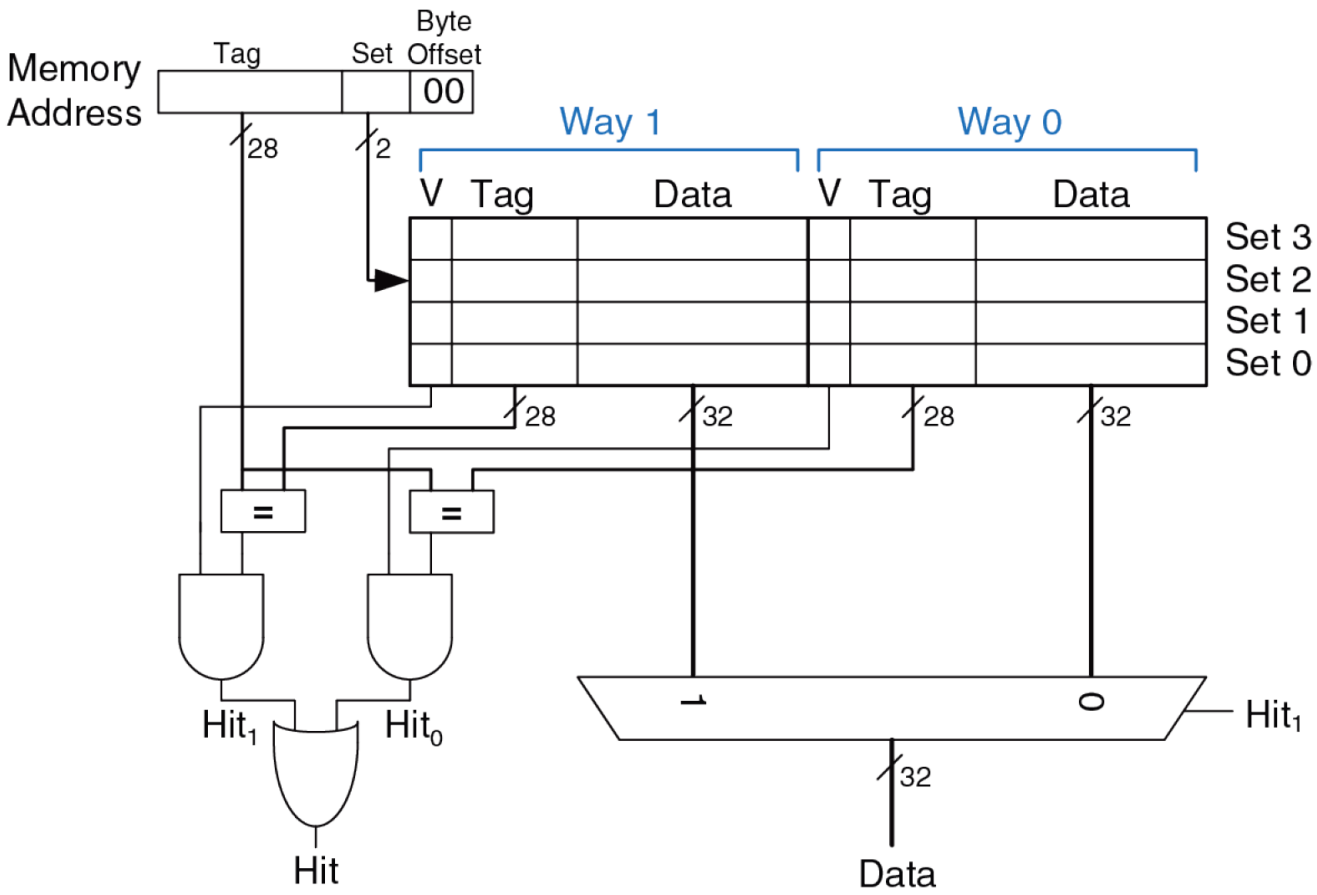
Пример вытеснения (evict)

```
      addi $t0, $0, 5
loop: beq  $t0, $0, done
      lw   $t1, 0x4($0)
      lw   $t2, 0x24($0)
      addi $t0, $t0, -1
      j    loop
done:
```



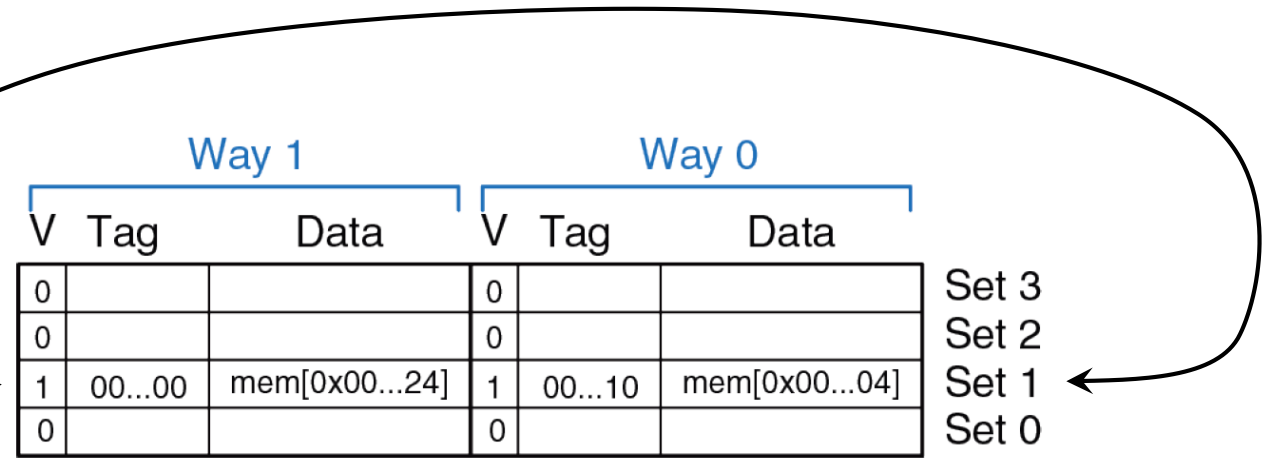
$$MR = \frac{\text{Число промахов}}{\text{Общее число доступов к памяти}} = \frac{10}{10} = 100\%$$

Множественно-ассоциативный кэш

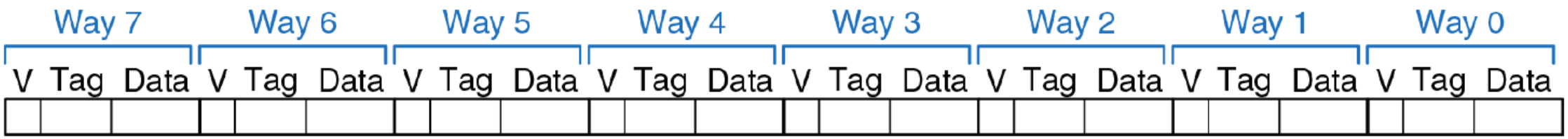


Тот же пример

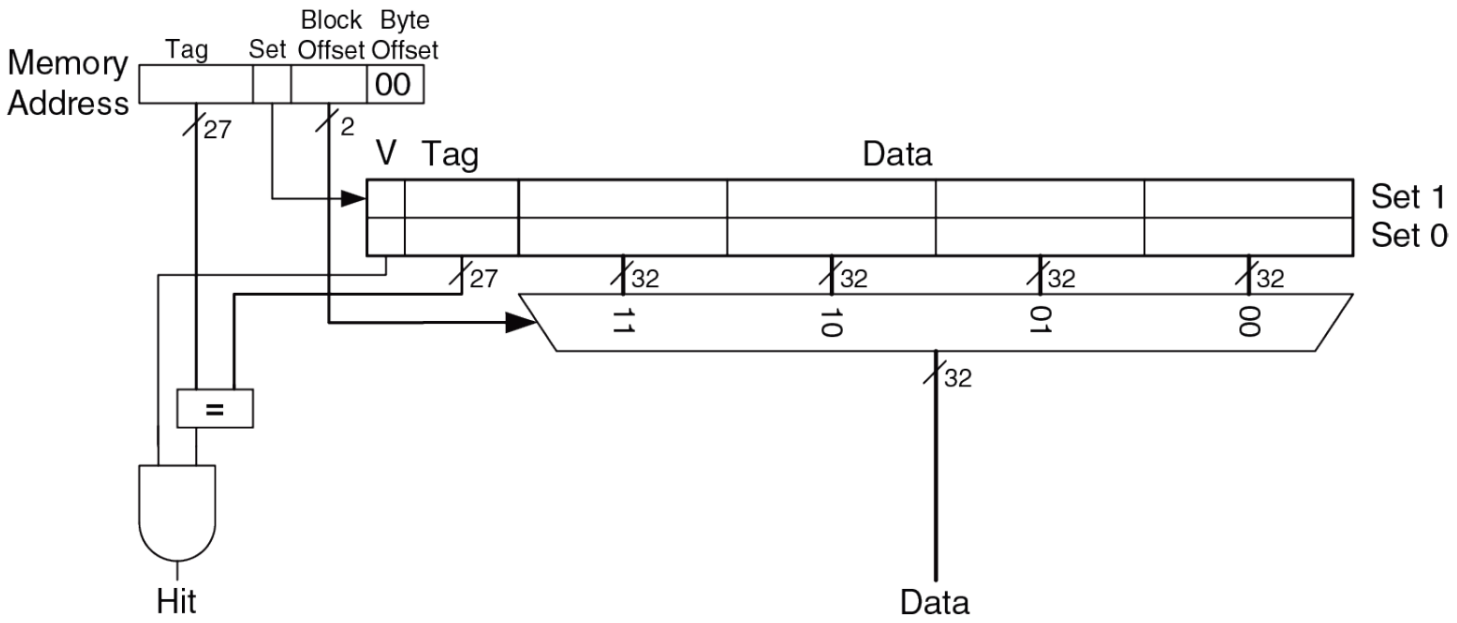
```
        addi $t0, $0, 5
loop:   beq  $t0, $0, done
        lw   $t1, 0x4($0)
        lw   $t2, 0x24($0)
        addi $t0, $t0, -1
        j    loop
done:
```



Полностью ассоциативный кэш

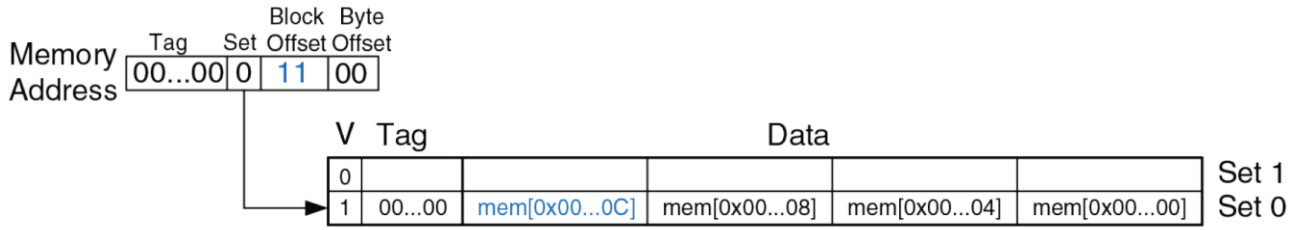


Длина строки



Первый пример

```
        addi $t0, $0, 5
loop:   beq  $t0, $0, done
        lw   $t1, 0x4($0)
        lw   $t2, 0xC($0)
        lw   $t3, 0x8($0)
        addi $t0, $t0, -1
        j    loop
done:
```



$$MR = \frac{\text{Число промахов}}{\text{Общее число доступов к памяти}} = \frac{1}{15} = 6.67\%$$

Способы организации кэш

Способ организации	Количество секций (N)	Количество наборов (S)
Прямого отображения	1	B
Множественно-ассоциативный	1 < N < B	B/N
Полностью ассоциативный	B	1

Четыре вопроса по иерархии памяти

1. ~~Где могут быть размещены данные в кэш-памяти? (Размещение строки)~~
2. ~~Как найти данные в кэш-памяти (Идентификация строки)~~
3. Какие данные нужно заместить, при заполненной кэш-памяти? (Замещение строки)
4. Что происходит при записи в кэш-память? (Стратегия записи)

Алгоритмы замещения данных

- LRU (Least Recently Used) – наиболее давнего использования +
- PLRU (Pseudo-Least Recently Used) – псевдо наиболее давнего использования +/–
- FIFO (First In First Out) – замещение в порядке очереди
- LFU (Least Frequently Used) – наименее частого использования +
- RND (Random Replacement) – замена случайной строки –

	Associativity								
	Two-way			Four-way			Eight-way		
Size	LRU	Random	FIFO	LRU	Random	FIFO	LRU	Random	FIFO
16 KiB	114.1	117.3	115.5	111.7	115.1	113.3	109.0	111.8	110.4
64 KiB	103.4	104.3	103.9	102.4	102.3	103.1	99.7	100.5	100.3
256 KiB	92.2	92.1	92.5	92.1	92.1	92.5	92.1	92.1	92.5

Стратегии чтения и записи в кэш

- Стратегии чтения
  - Чтение с параллельной выборкой (look-aside)
  - Чтение со сквозным просмотром (look-through)

- Стратегии записи
  - Сквозная запись (write-through)
  - Буферизированная сквозная запись
  - Отложенная запись (write-back)
    - В среднем на 10% эффективнее сквозной записи. Чаще используется

Многоуровневый кэш

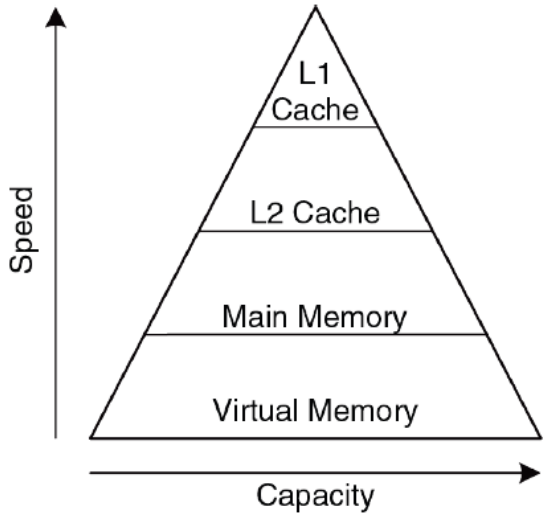
$t_{L1} = 1$

$t_{L2} = 10$

$t_{MM} = 100$

$MR_{L1} = 5\%$

$MR_{L2} = 20\%$



- Инклюзивный кэш
- Эксклюзивный кэш

$$AMAT = t_{L1} + MR_{L1} \cdot (t_{L2} + MR_{L2} \cdot t_{MM})$$

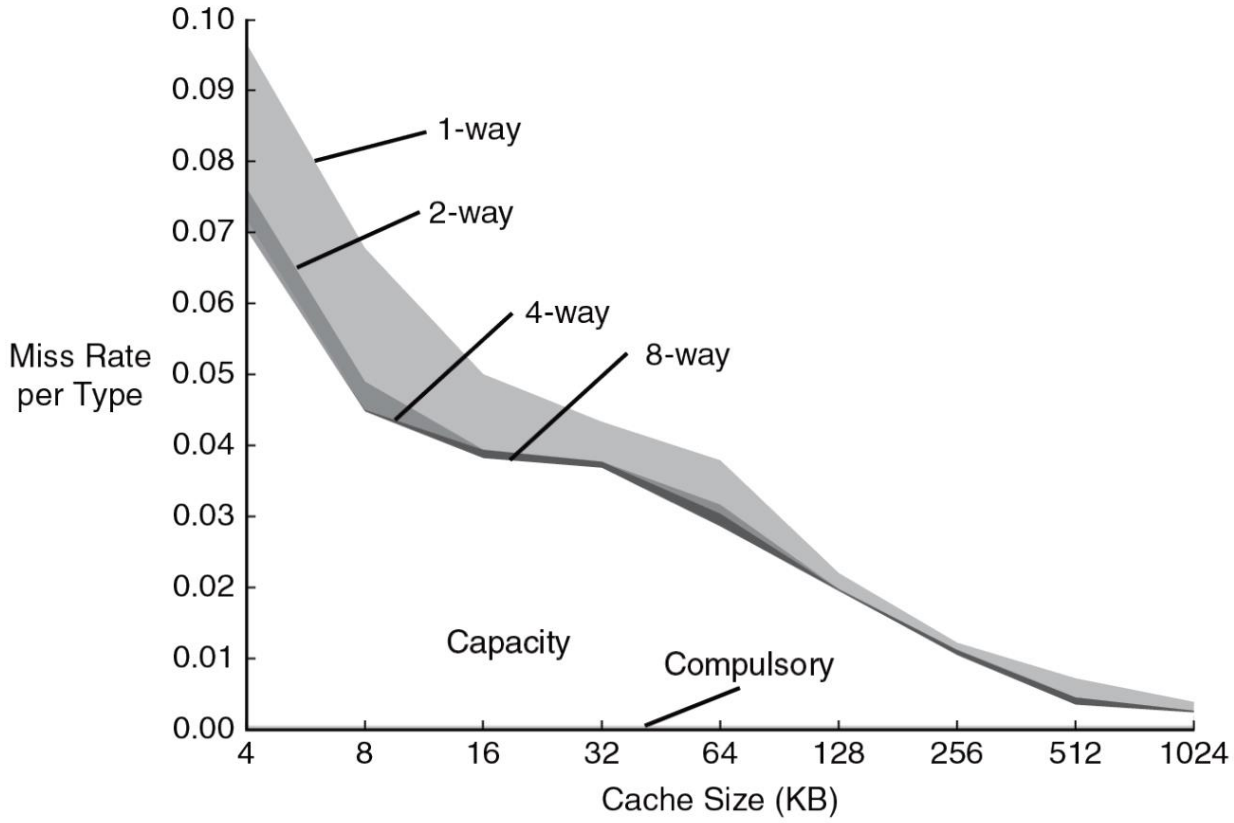
$$AMAT = 1 + 0.05 \cdot (10 + 0.2 \cdot 100) = 2.5 \text{ такта}$$

$$AMAT_{withoutL2} = 1 + 0.05 \cdot (100) = 6 \text{ тактов}$$

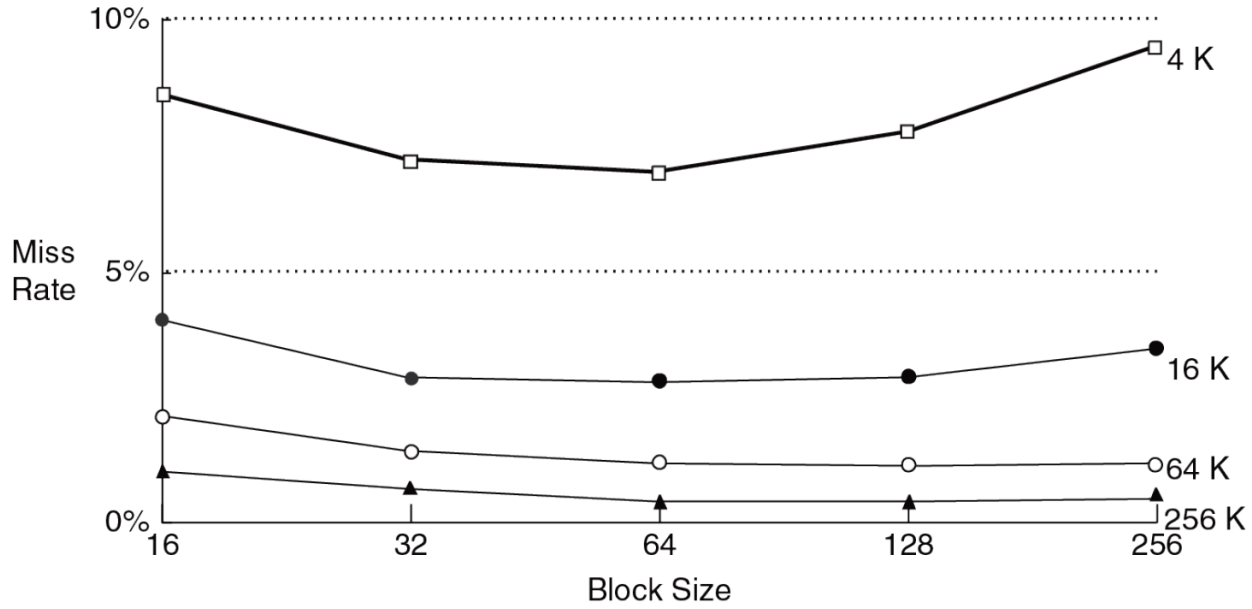
Виды промахов (Three C)

- Неизбежные промахи (Compulsory misses)
- Промахи из-за недостаточной емкости кэш (Capacity misses)
- Промахи из-за конфликтов (Conflict misses)

Частота промахов



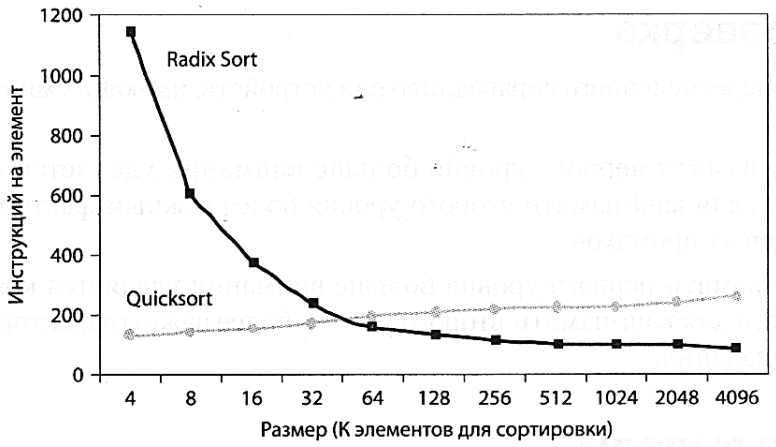
Частота промахов

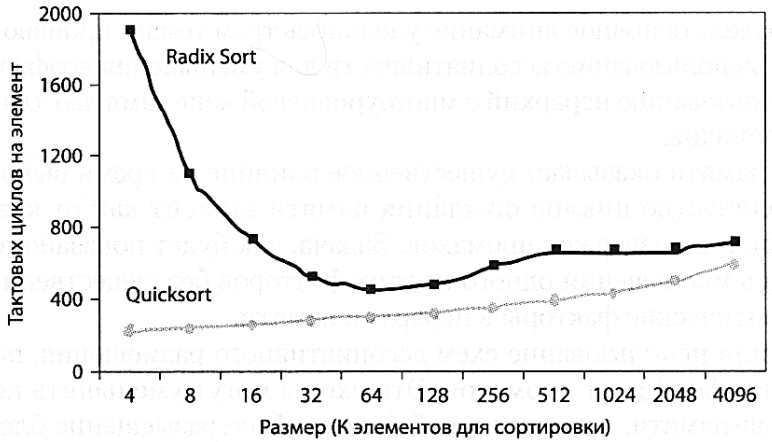


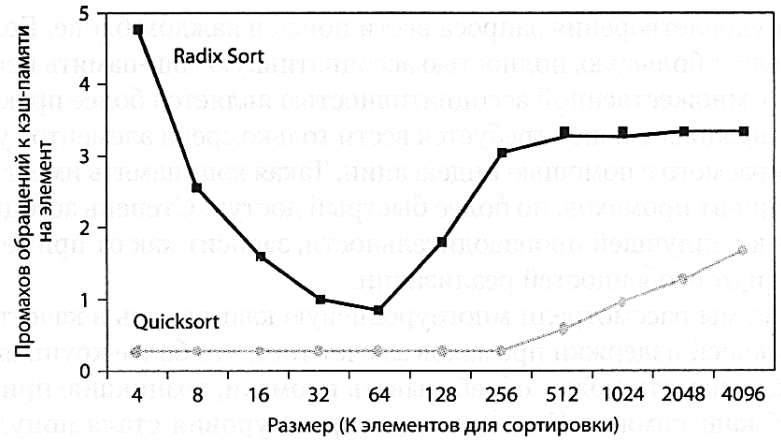
Основные оптимизации кэш

- Больший размер блока для уменьшения доли промахов
- Кэши большего объема для уменьшения доли промахов
- Увеличение ассоциативности для уменьшения доли промахов
- Многоуровневые кэши для уменьшения потерь на промахах
- Предоставление приоритета промахам считывания по отношению к записям для уменьшения потерь на промахи

Пример


Инструкций на элемент
1200
1000
800
600
400
200
0
Radix Sort
Quicksort
4
8
16
32
64
128
256
512
1024
2048
4096
Размер (К элементов для сортировки)


Тактовых циклов на элемент
2000
1600
1200
800
400
0
Radix Sort
Quicksort
4
8
16
32
64
128
256
512
1024
2048
4096
Размер (К элементов для сортировки)


Промахов обращений к кэш-памяти на элемент
5
4
3
2
1
0
Radix Sort
Quicksort
4
8
16
32
64
128
256
512
1024
2048
4096
Размер (К элементов для сортировки)