

АРБИТРЫ И РАЗДЕЛЕНИЕ ПАМЯТИ МЕЖДУ НЕСКОЛЬКИМИ ПРОЦЕССОРНЫМИ ЯДРАМИ КОГЕРЕНТНОСТЬ



ШКОЛА СИНТЕЗА
ЦИФРОВЫХ СХЕМ

ПРИ ПАРТНЕРСТВЕ

Занятие №19

29 февраля 2025





**ШКОЛА СИНТЕЗА
ЦИФРОВЫХ СХЕМ**



Никита Поляков

YADRO, ведущий инженер

Школа синтеза, инженер-ведущий

2005 - 2011: МФТИ. ФРТК.

2008 - 2020: МЦСТ. Эльбрус

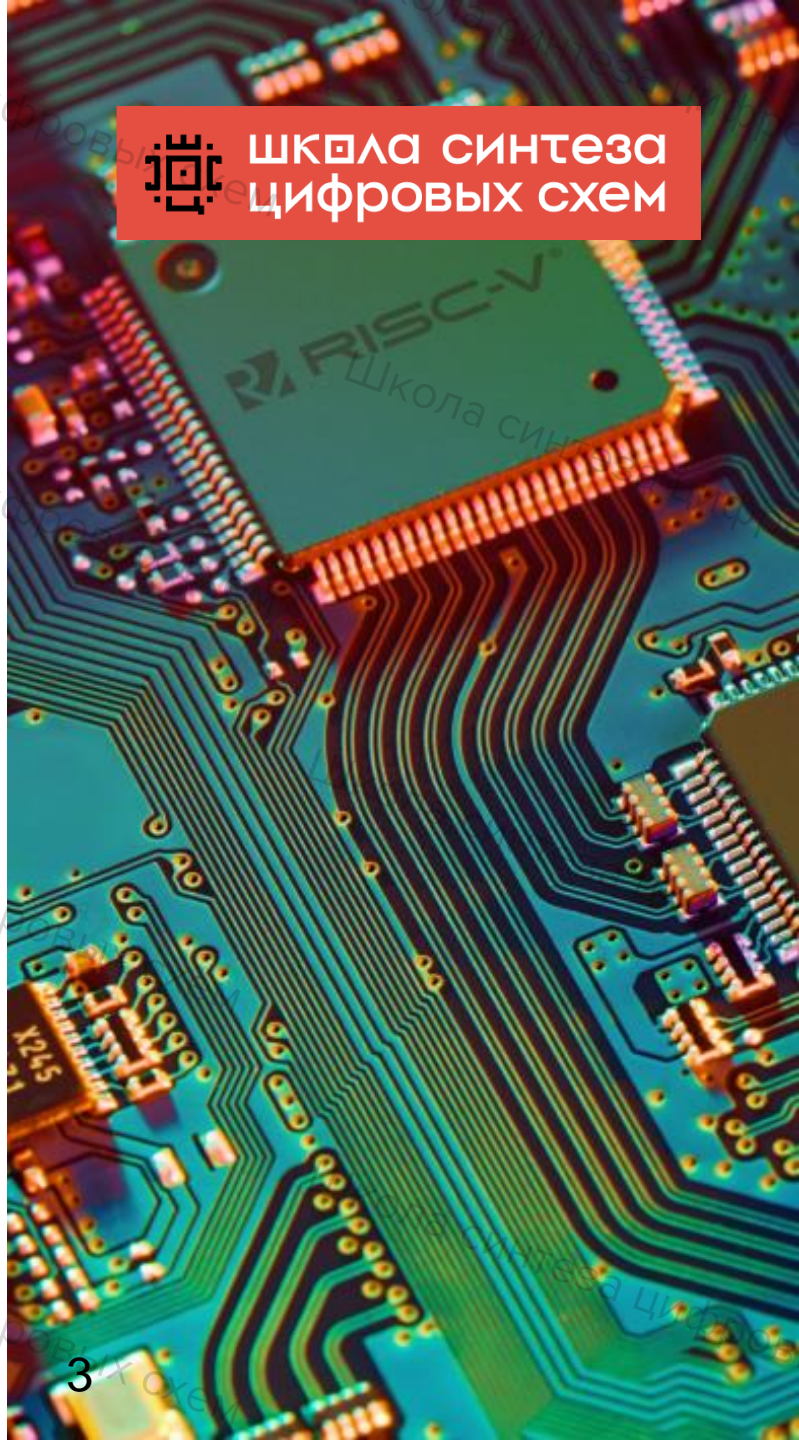
2020 - н.в.: YADRO. RISC-V

ТЕМЫ ЗАНЯТИЯ

- Многопроцессорные системы
- Арбитры
- Многопортовая память
- Когерентность



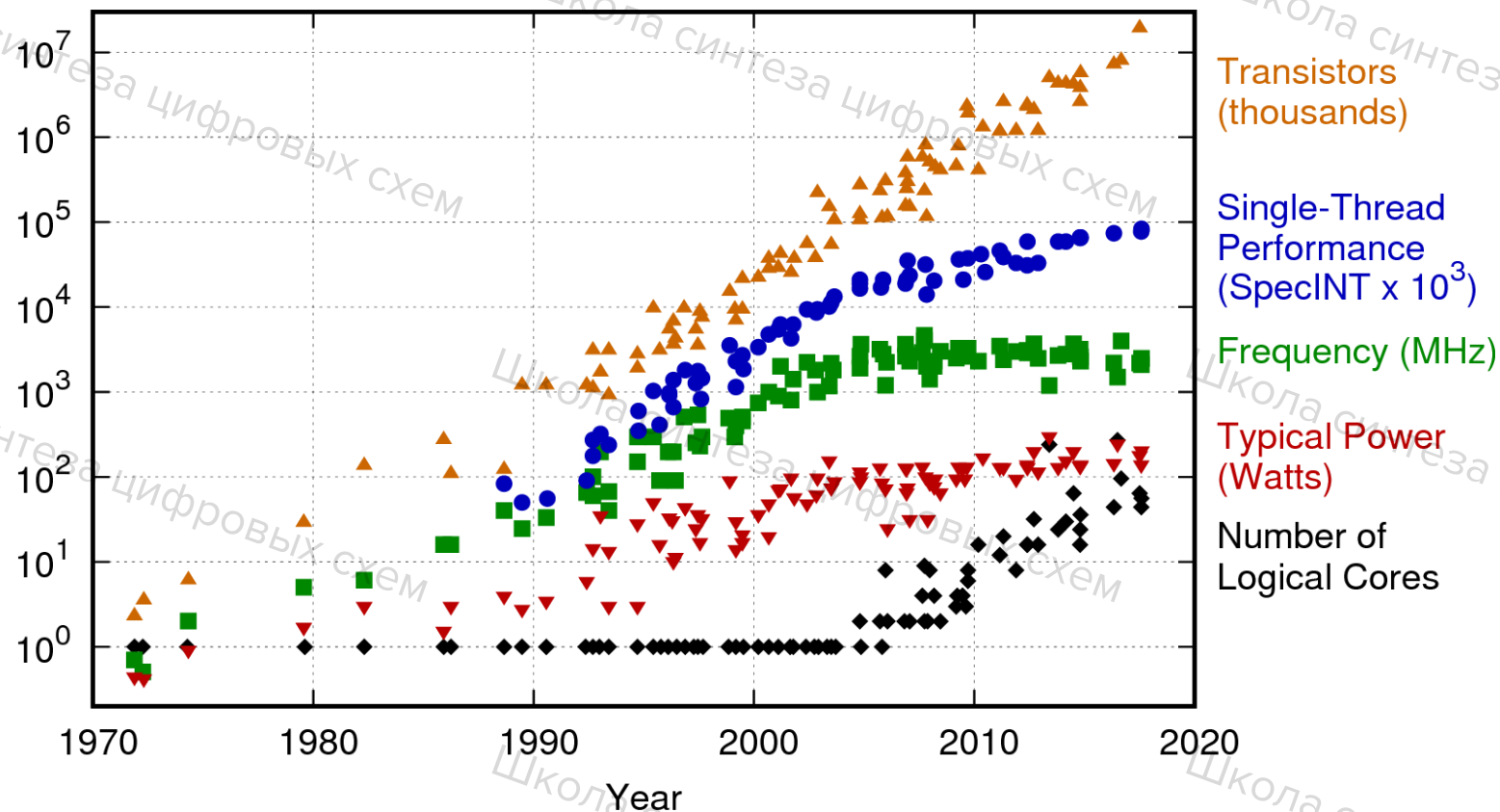
ШКОЛА СИНТЕЗА
ЦИФРОВЫХ СХЕМ



МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ И ЗАКОН МУРА



42 Years of Microprocessor Trend Data



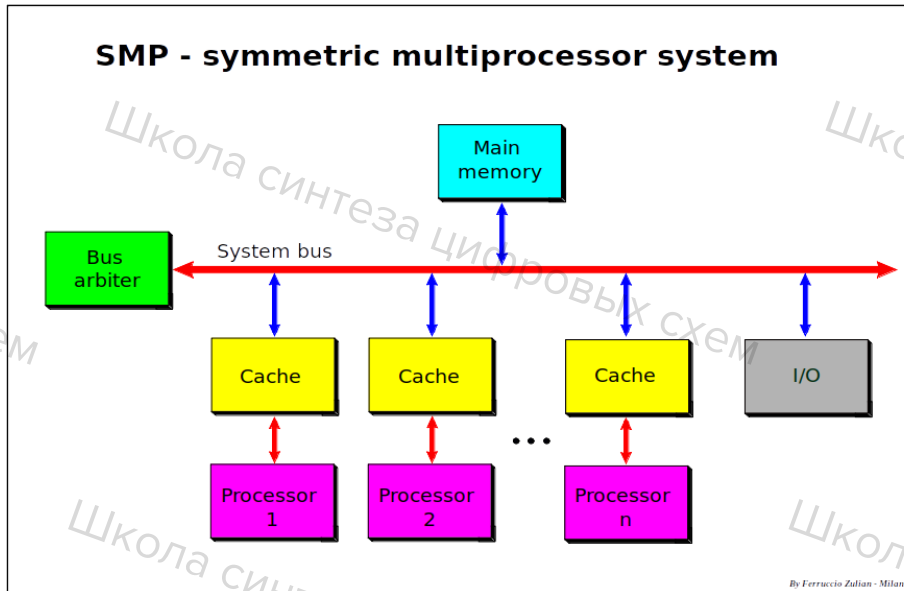
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

Источник: Karl Rupp “42 Years of Microprocessor Trend Data”.
<https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/>

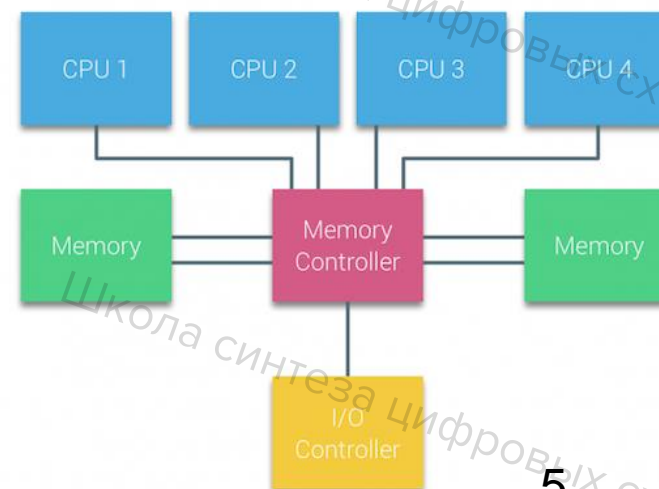
МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ. SMP/UMA



- **UMA** — *Uniform memory access* — архитектура многопроцессорных компьютеров с общей памятью, в которой время запроса к данным из памяти не зависит ни от того, какой именно процессор обращается к памяти, ни от того, какой именно чип памяти содержит нужные данные.
- bus-based symmetric multiprocessing (SMP);
- crossbar switches;
- multistage interconnection networks.
- **SMP** — *Symmetric multiprocessing* или *shared-memory multiprocessing* — архитектура многопроцессорных компьютеров, в которой два или более одинаковых процессора сравнимой производительности подключаются единообразно к общей памяти (и периферийным устройствам) и выполняют одни и те же функции.



Шина
(system bus)



Коммутатор
(crossbar /
switch)

МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ. NUMA

NUMA (*Non-Uniform Memory Access* или *Non-Uniform Memory Architecture*) — архитектура организации компьютерной памяти, используемая в мультипроцессорных системах. Процессор имеет быстрый доступ к локальной памяти через свой контроллер, а также более медленный канал до памяти, подключенной к контроллерам (слотам) других процессоров, реализуемый через компьютерную шину обмена данными.



АРБИТРЫ

Арбитр — устройство, контролирующее доступ нескольких абонентов/запросчиков к общему ресурсу.

Виды:

- Централизованный;
- Централизованный сериализованный (daisy chain);
- Распределенный с обнаружением коллизий.



Vector of requests

0
1
1
0
1
0
0
1



Arbiter
[policy]

Vector of grants

0
0
0
0
0
0
0
1



АРБИТРЫ. ЦЕНТРАЛИЗОВАННЫЙ АРБИТРАЖ



Политики/алгоритмы:

- Фиксированный/статический приоритет a.k.a. Find first;
- Random (pseudo);
- Round robin;
- Весовой;
- Временной.

Статический приоритет

```
module fixed_prio_arb(  
    input wire [3:0] req,  
    output wire [3:0] gnt  
);  
    assign gnt[0] = req[0];  
    assign gnt[1] = ~req[0] & req[1];  
    assign gnt[2] = ~req[0] & ~req[1] & req[2];  
    assign gnt[3] = ~req[0] & ~req[1] & ~req[2] & req[3];  
endmodule
```

$$T_{\text{delay}} \sim \log N$$
$$S \sim N$$

АРБИТРЫ. ROUND ROBIN

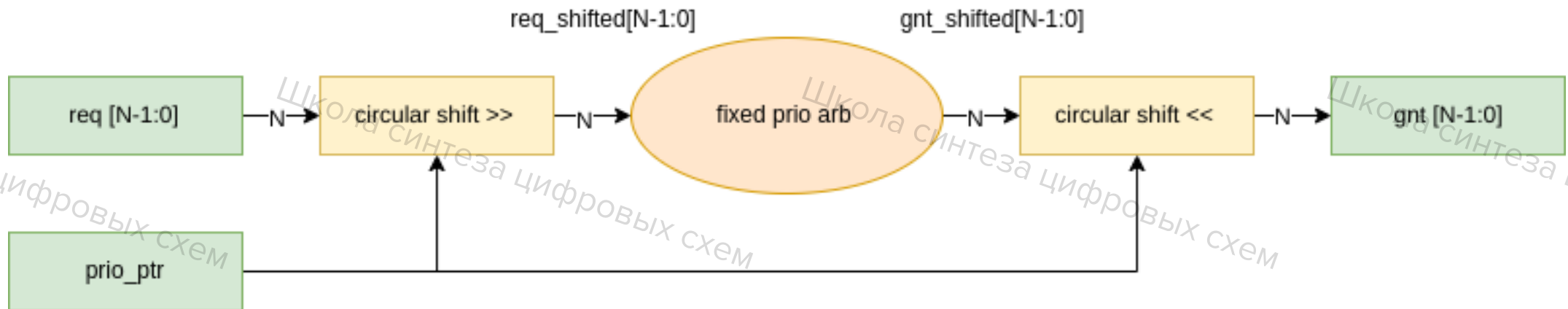
Round Robin – алгоритм равноправного предоставления доступа к ресурсу путем циклического изменения приоритета



gnt

0 0 1 1 0 0 1 0 0 0 1 0 0

prio



$$T_{\text{delay}} \sim 3 * \log N$$

$$S \sim N * \log N$$

АРБИТРЫ. ROUND ROBIN



Варианты изменения приоритета:

А) Последовательное: $\text{prio_ptr}++$

В) Следующий за получившим грант: $\text{prio_ptr} = \text{gnt_ptr} + 1$

Опции имплементации:

- Подтверждение приема гранта (ack);
- Заморозка гранта до приема (lock);
- Спекулятивный арбитраж (pending);
- Несколько грантов (multigrant).

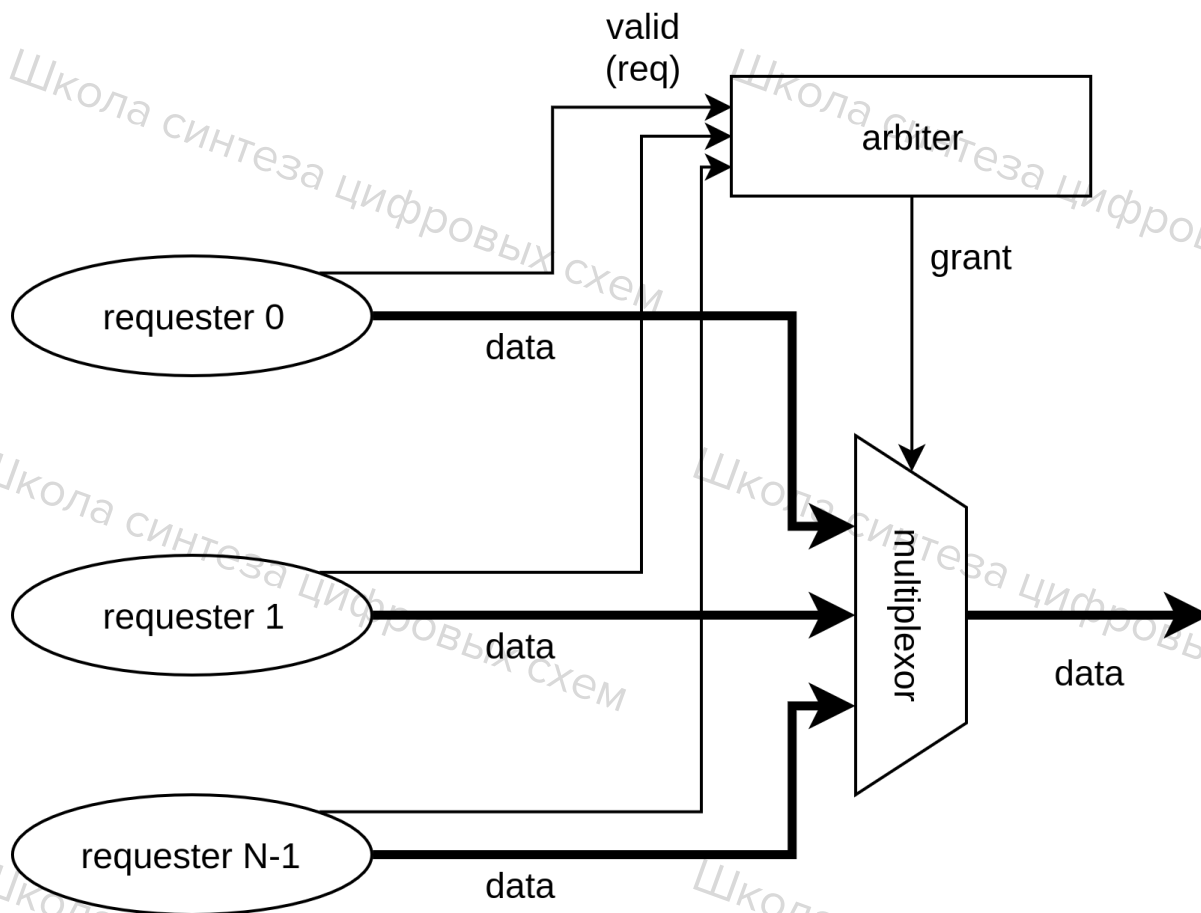


next_prio B

next_prio A

Термин Round-robin обозначает письмо к одному адресату, подписанное несколькими корреспондентами, к примеру, общественное обращение к должностному лицу. Название произошло, видимо, в XVII столетии, от французского выражения «ruban rond» (round ribbon, переплетенные в обруч ленты). Таким образом определялся порядок подписания коллективных петиций и ходатайств без соблюдения иерархии подписавшихся, при котором подписи располагались кружком, с тем, чтобы было невозможно найти автора или зачинщика, подписавшегося первым, своеобразная круговая порука.

АРБИТРЫ. МУЛЬТИПЛЕКСОРЫ



МНОГОПОРТОВАЯ ПАМЯТЬ. АППАРАТНЫЕ БЛОКИ SRAM



Статическая память с произвольным доступом

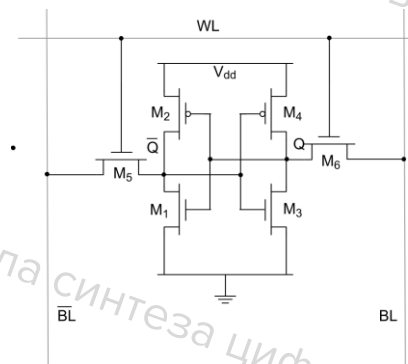
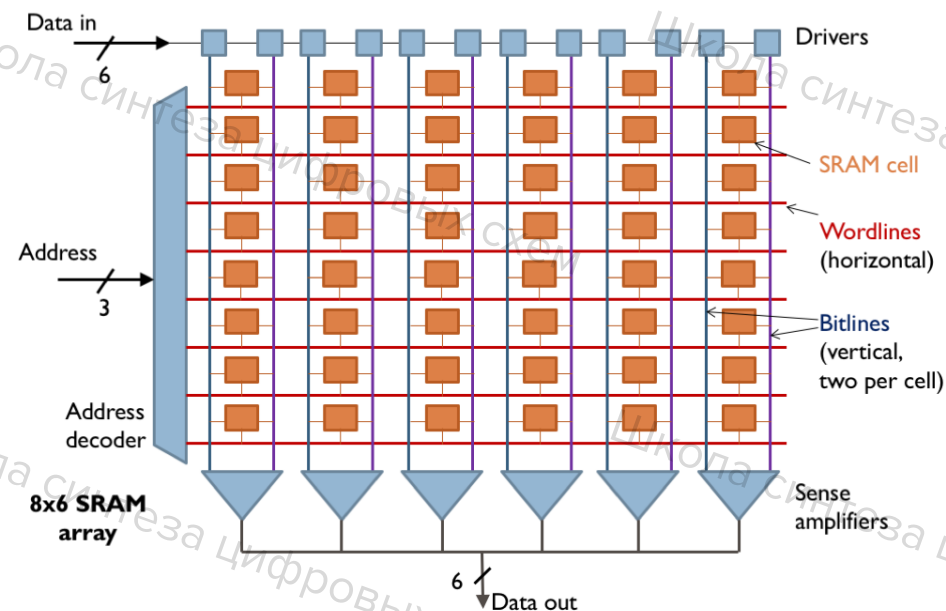
(SRAM, *static random access memory*) — полупроводниковая оперативная память, в которой каждый двоичный разряд хранится в схеме с положительной обратной связью, позволяющей поддерживать состояние без регенерации, необходимой в динамической памяти (DRAM). Тем не менее сохранять данные без перезаписи SRAM может, только пока есть питание, то есть SRAM остается энергозависимым типом памяти.

Произвольный доступ (*RAM — random access memory*) — возможность выбирать для записи/чтения любой из битов.

По сравнению с массивами DFF (DFF array):

- **Преимущества:** Меньшая площадь и энергопотребление
- **Недостатки:** задержка данных чтения от 1 до нескольких тактов

Static RAM (SRAM)



МНОГОПОРТОВАЯ ПАМЯТЬ. ПОРТЫ SRAM



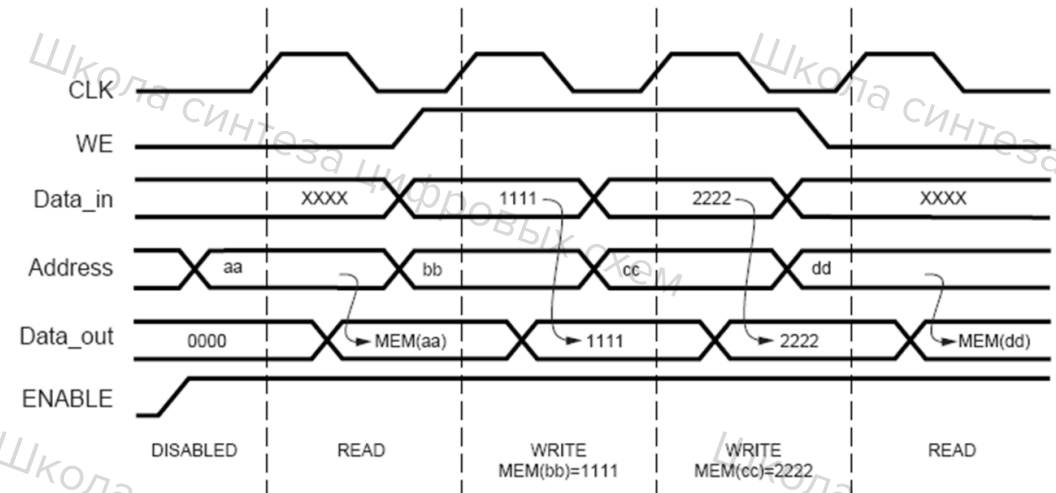
Широко применяются следующие типы блоков по количеству портов:

- 1 read-write port (single port)
- 1 read + 1 write port (two port, pseudo dual port)
- 2 read-write port (dual port, true dual port)

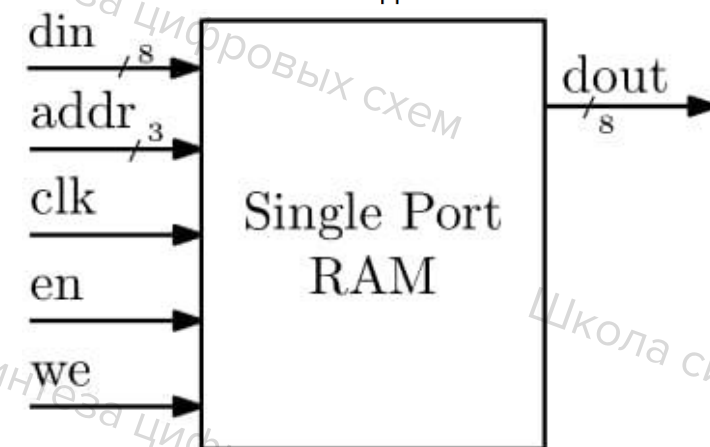
Пример интерфейса single port RAM:

- Din – данные записи
- Addr – адрес обращения
- Clk – синхросигнал
- En – разрешение работы
- We – признак записи (0 – чтение)
- Dout – данные чтения

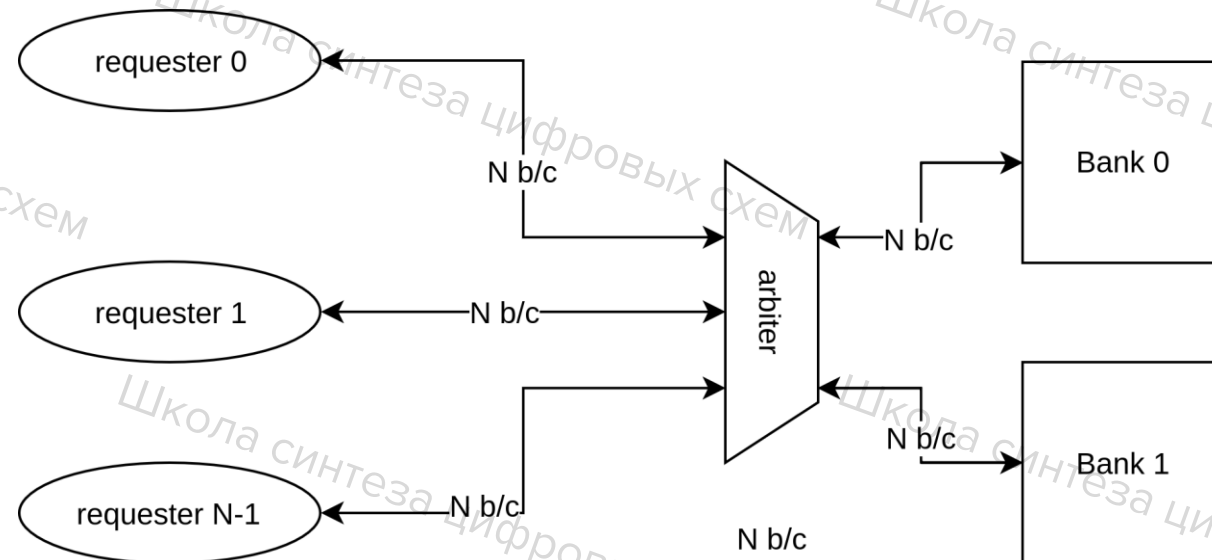
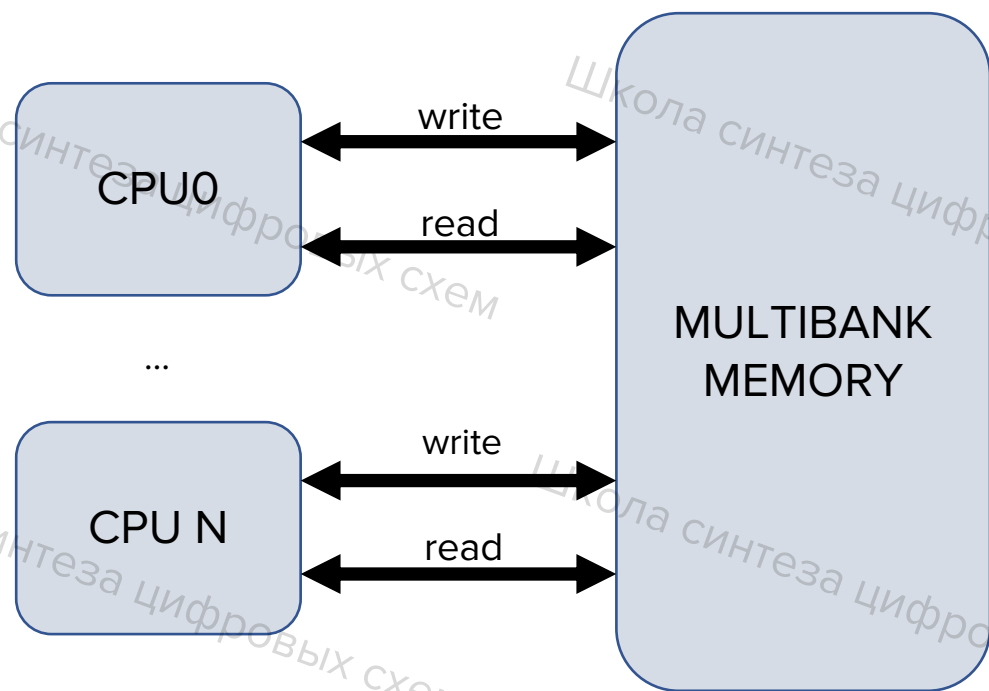
Single port waveform



Source: Xilinx App Note 463



МНОГОПОРТОВАЯ ПАМЯТЬ



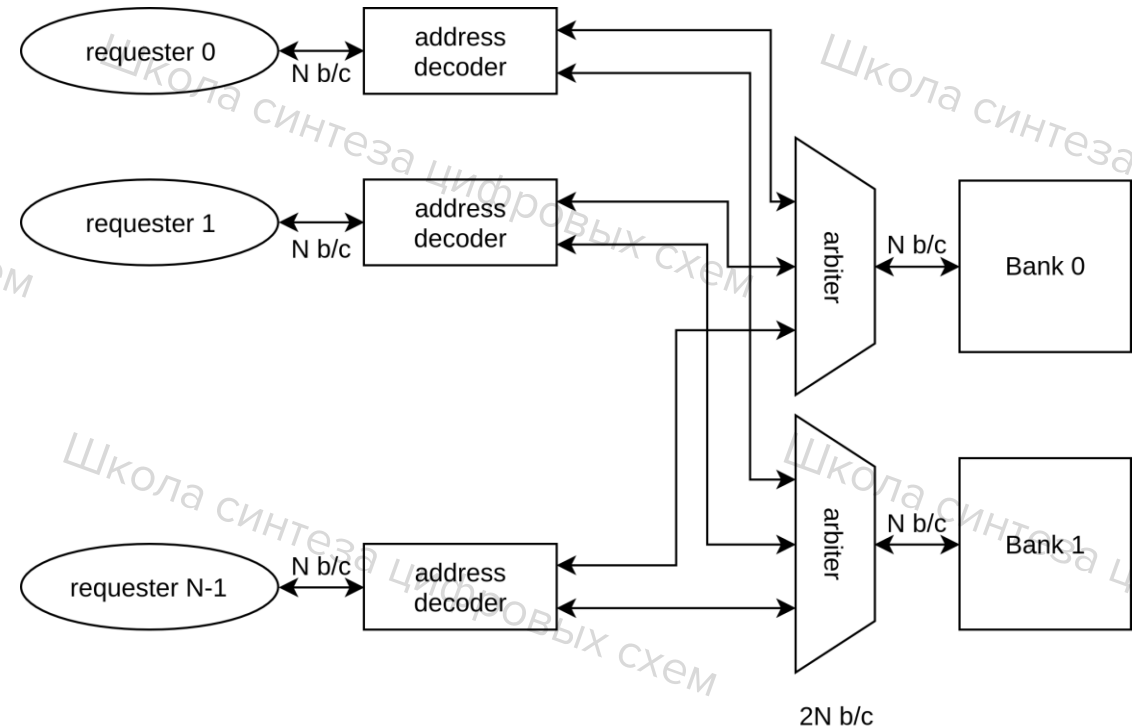
Простой арбитраж дает N байт/такт

МНОГОПОРТОВАЯ ПАМЯТЬ

Для увеличения пропускной способности памяти SRAM разбивают на блоки (bank), каждый из которых содержит свою уникальную часть данных.

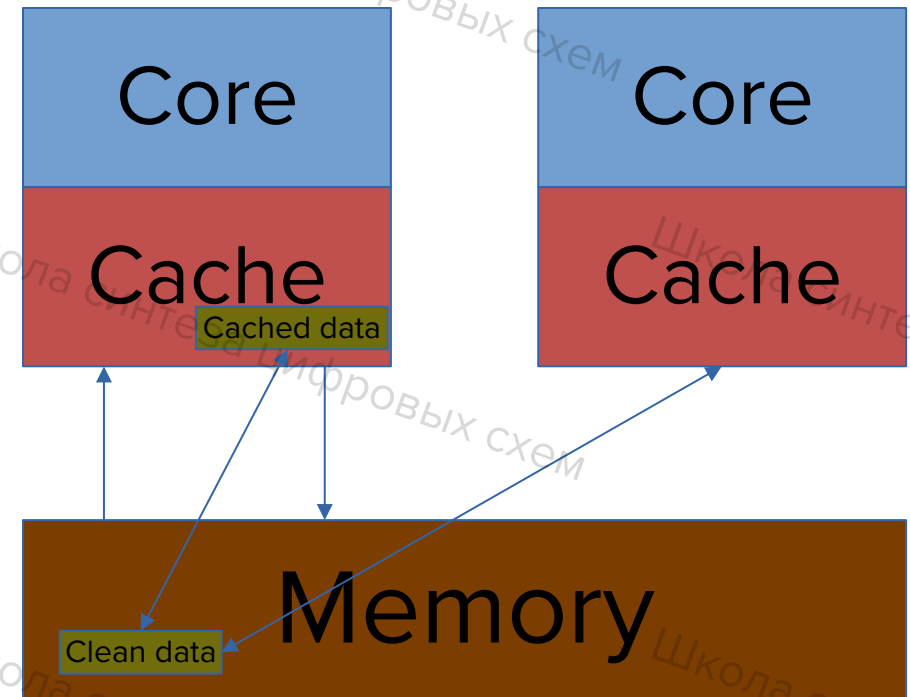
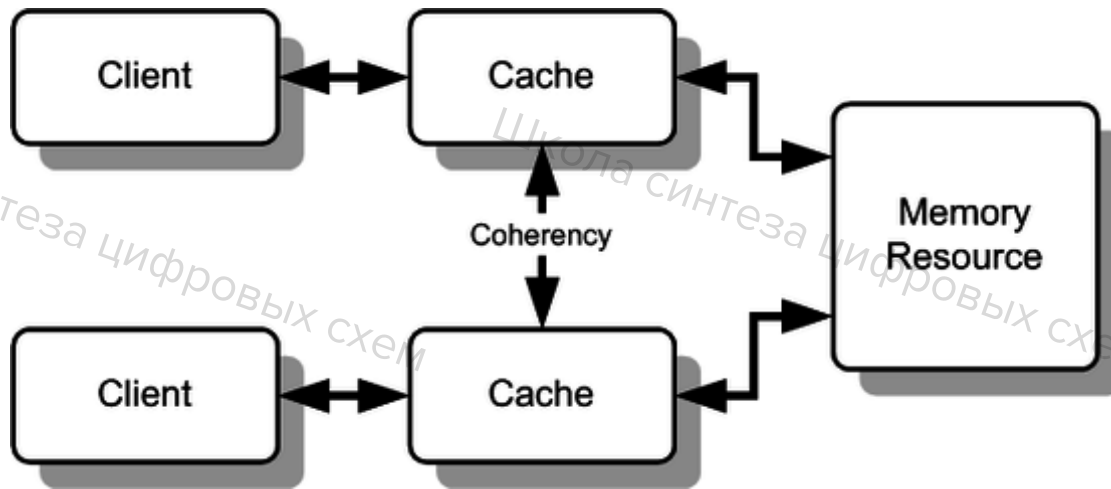
Разделение адресного пространства может быть:

- Крупными непрерывными областями (coarse grain);
- Мелкими чередующимися кусочками (fine grain).

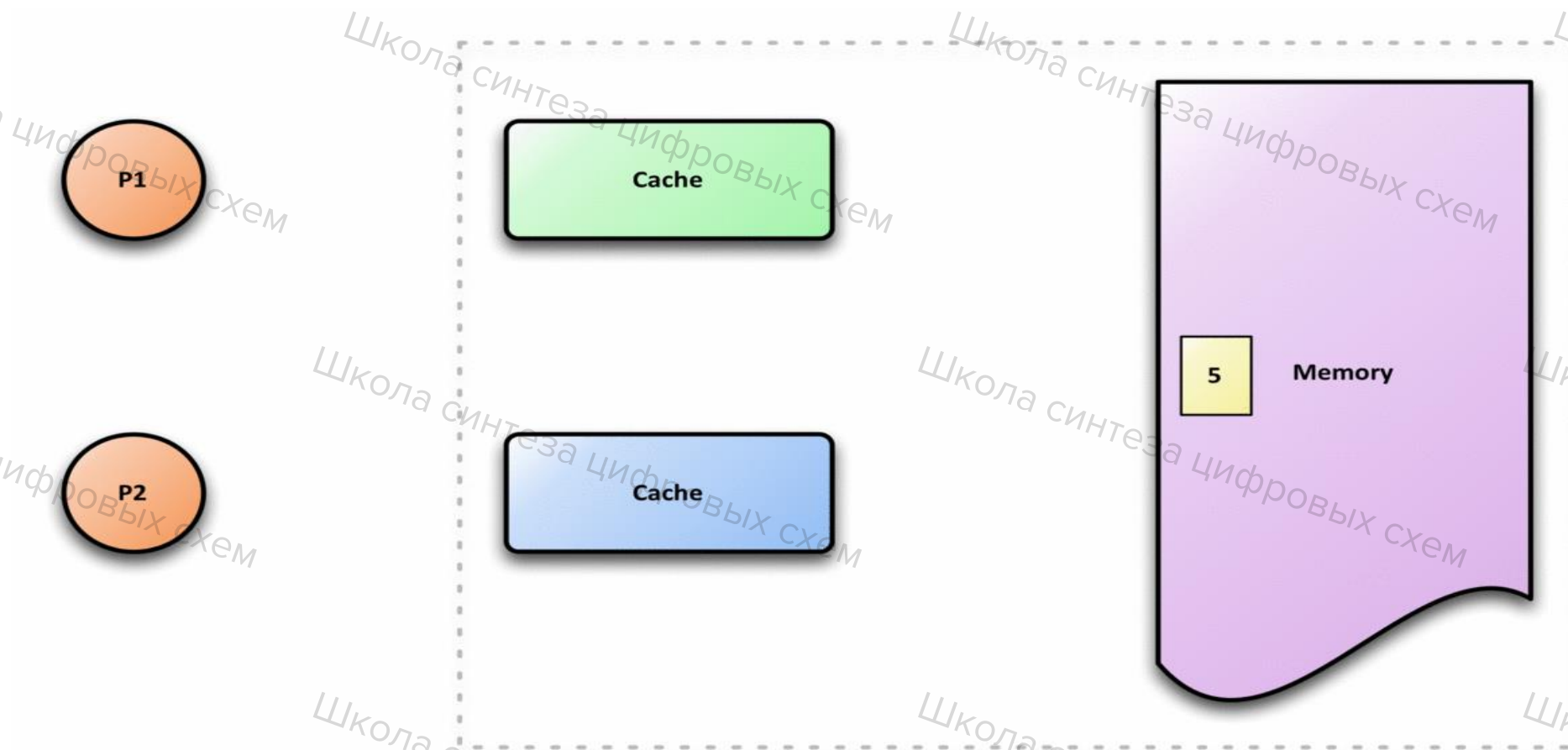


КОГЕРЕНТНОСТЬ

Когерентность кэша (cache coherence) — свойство кэшей, означающее целостность данных, хранящихся в локальных кэшах для разделяемого ресурса.



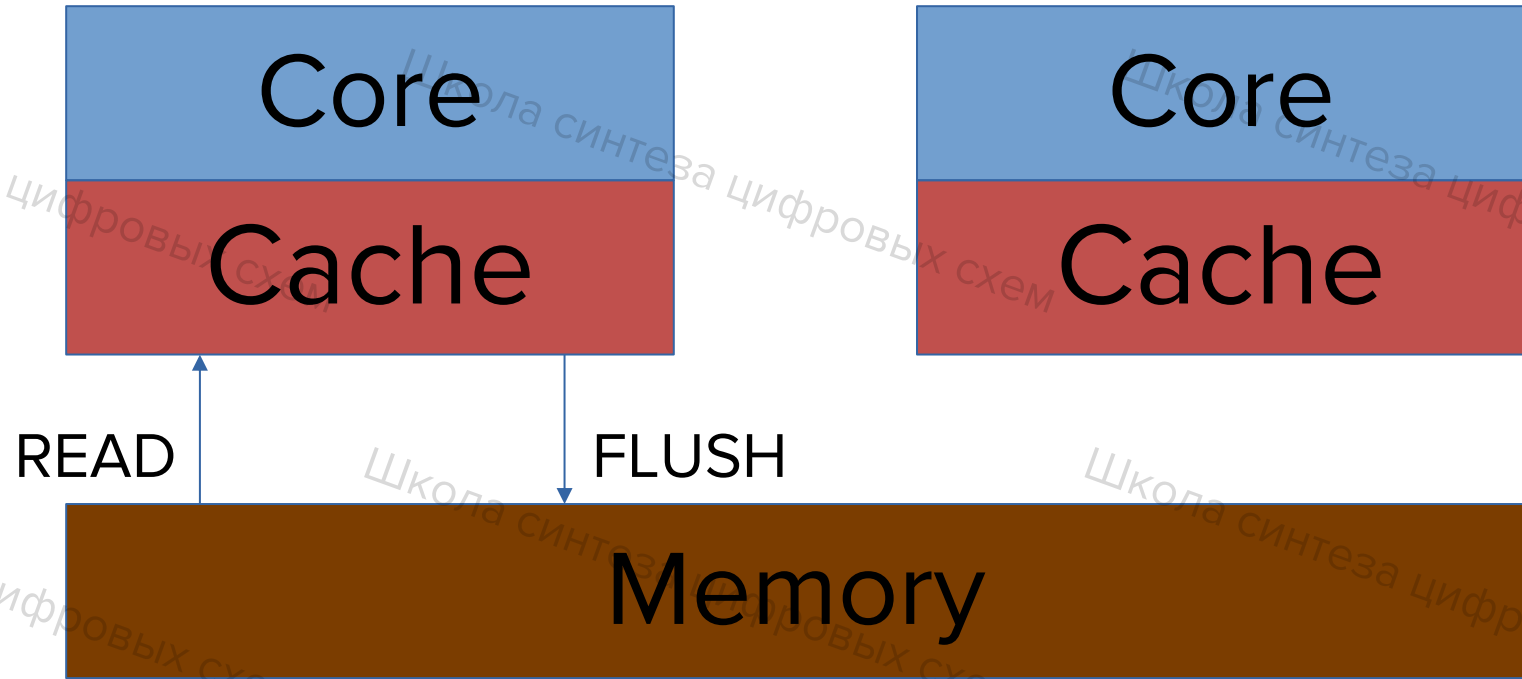
КОГЕРЕНТНОСТЬ. ПРИМЕР НЕКОГЕРЕНТНОЙ СИ



КОГЕРЕНТНОСТЬ. ПРОГРАММНАЯ ПОДДЕРЖКА

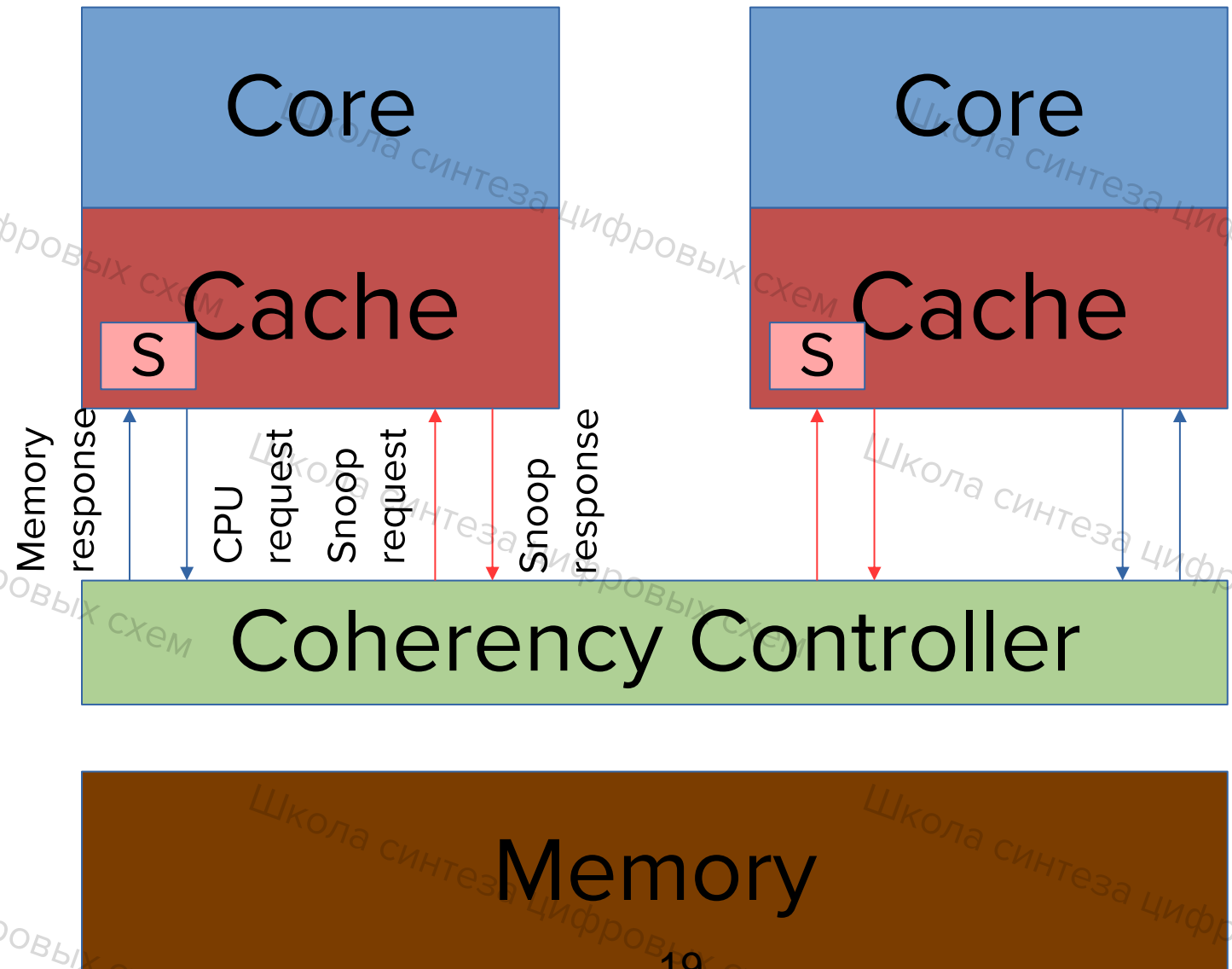


- Используются Cache Management Operation: flush, invalidate
- Модифицированные данные должны быть вытеснены в память чтобы быть видимыми другим CPU (flush)
- Ненужные немодифицированные данные могут быть удалены из кэша без записи в память (invalidate)



КОГЕРЕНТНОСТЬ. АППАРАТНАЯ ПОДДЕРЖКА. MSI

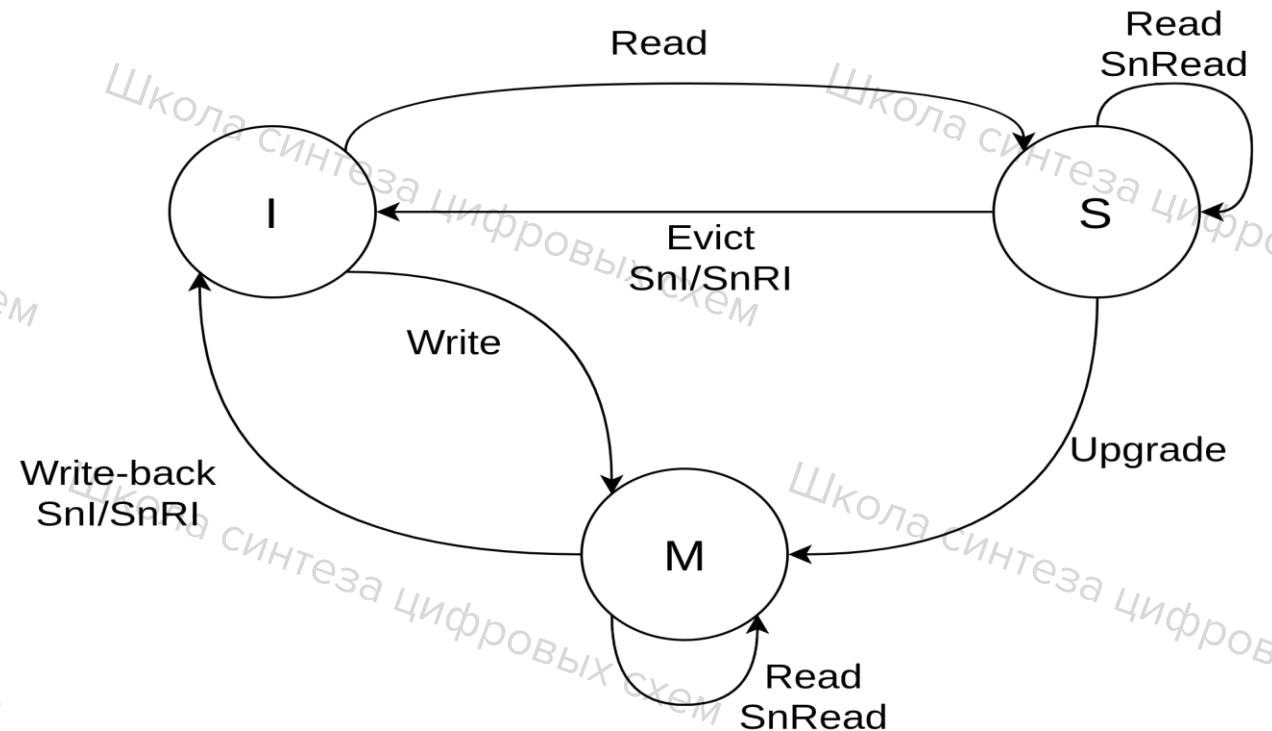
- Когерентность поддерживается за счет Когерентного Контроллера
- Самые популярные протоколы поддержки когерентности основаны на снуп-запросах (snoop)
- Снуп-запрос – служебное сообщение из когерентного контроллера в кэш
- Modified: данные модифицированы в кэше и не соответствуют содержимому памяти. Данные должны быть записаны в память при вытеснении из кэша.
- Shared: данные немодифицированы и могут быть вытеснены без записи в память.
- Invalid: данные либо отсутствуют в кэше, либо были удалены из кэша.



КОГЕРЕНТНОСТЬ. ПРОТОКОЛ MSI



- **SnI** – snoop invalidate
 - **SnRI** – snoop read and invalidate
 - **SnRead** – snoop read
 - **Upgrade** – запись в Shared
 - **Evict** – вытеснение немодифицированных данных
 - **Write-back** – вытеснение модифицированных данных
- Cacheline in M state must be the only (unique) cacheline copy in the system
 - Multiple S state cacheline copies are permitted
 - If a CPU wants to write data it requests write permission (M state transition) from Coherence Controller



	M	S	I
M	✗	✗	✓
S	✗	✓	✓
I	✓	✓	✓

КОГЕРЕНТНОСТЬ. ПРОТОКОЛЫ



MESI: Exclusive (E) состояние имеет немодифицированная строка, расположенная только в одно кэше, запись разрешена в E состоянии.

MOESI: Owned (O) разрешено совместное использование модифицированной строки без обязательного обновления значения в памяти, данные будут записаны кэшем, имеющим состояние O (остальные имеют состояние S)

Snoop-based – снупирование всех кэшей

Directory-based – справочник состояний кэшей