

## АРХИТЕКТУРА АППАРАТНЫХ СРЕДСТВ · УЧЕБНЫЙ МОДУЛЬ 1/3

# Виды памяти и принцип работы. Основные характеристики

Разберём, как компьютер хранит биты, почему ОЗУ требует питания и что означает скорость памяти. Сначала изучите всю теорию, разберите интерактивные модели, затем пройдите тест в конце страницы. Ответы сохраняются в этом браузере автоматически.

Теория → тест → PDF

15 проверяемых вопросов

3 развёрнутых ответа

Работает без интернета

ФИО студента

Розыков Самандар

Группа

РУПО 26/11

Дата

2026-10-14

01

## Зачем компьютеру память

Процессор умеет **выполнять инструкции**, но ему нужно где-то брать эти инструкции, получать исходные данные и оставлять результаты. Для этого существует память — устройства и структуры, которые хранят информацию в виде битов. Один бит принимает значение **0** или **1**; восемь бит образуют байт.

Представьте повара. **Процессор** — это повар, **регистры** — продукты прямо в руках, **кэш** — маленькая полка рядом с плитой, **оперативная память** — рабочий стол, а **SSD** — кладовая. Повар может хранить многое в кладовой, но непрерывно бегать в неё за каждой ложкой неудобно.

### Путь данных в типичном ПК

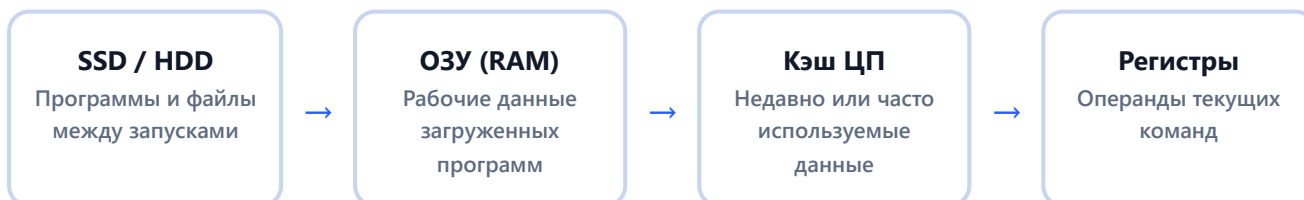


Схема упрощена: процессор и контроллеры взаимодействуют с памятью сложнее; здесь показана идея приближения данных к вычислениям.

Когда вы запускаете игру, ОС обычно читает исполняемые файлы и необходимые ресурсы с SSD и отображает или загружает их в память. Во время работы процессор обращается к данным через кэш и ОЗУ, а изменения в файлах по мере необходимости записываются в постоянное хранилище. **Место на диске и объём ОЗУ — разные ресурсы.**

02

По каким признакам различают память

Нет единственного признака «быстрая / медленная». Память классифицируют сразу по нескольким независимым свойствам.

| Признак                | Категории                                                                     | Пример                                                                                        |
|------------------------|-------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| Сохранение без питания | Энергозависимая / энергонезависимая                                           | RAM теряет содержимое; SSD сохраняет                                                          |
| Назначение             | Регистры, кэш, основная память, долговременное хранение                       | Кэш L1 ускоряет доступ ядра к данным                                                          |
| Способ доступа         | Произвольный, последовательный и др.                                          | DRAM обеспечивает доступ к адресуемым ячейкам; лента ориентирована на последовательное чтение |
| Возможность записи     | Чтение-запись, однократная или ограниченная перезапись, постоянное содержимое | DRAM перезаписывается многократно; масочное ROM записано при изготовлении                     |
| Физический принцип     | Полупроводники, магнитный или оптический носитель                             | NAND Flash, HDD, оптический диск                                                              |

**Важное различие:** RAM означает «память с произвольным доступом», а не «любая энергозависимая память». Flash тоже обеспечивает адресуемый доступ, но использует иные правила стирания и записи. В повседневной речи «RAM» обычно означает оперативную DRAM.

Основные виды по назначению

**Регистры и кэш**

Находятся в процессоре или рядом с вычислительными блоками. Очень быстрые, ограниченного объёма. Кэш обычно использует SRAM.

**Оперативная память**

Основное рабочее пространство ОС и программ. Обычно построено на DRAM; содержимое теряется после выключения питания.

**ROM и Flash**

Хранят прошивки и другие данные после отключения питания. Flash можно перепрограммировать, но правила записи отличаются от DRAM.

## SSD / HDD

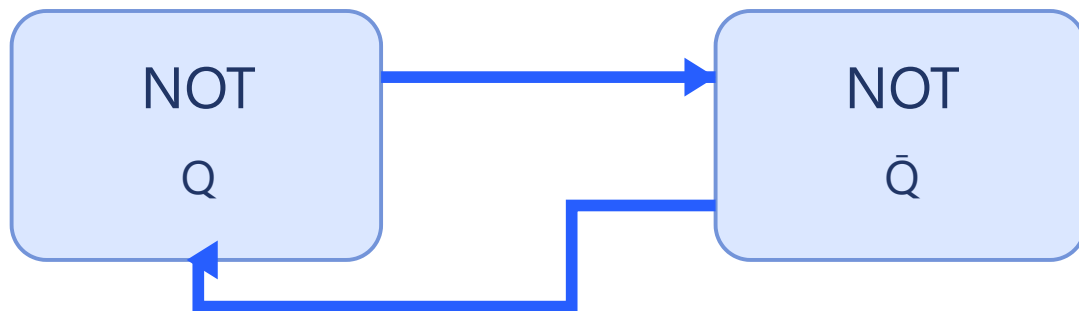
Долговременное хранение файлов. SSD обычно использует NAND Flash, HDD — магнитные пластины.

03

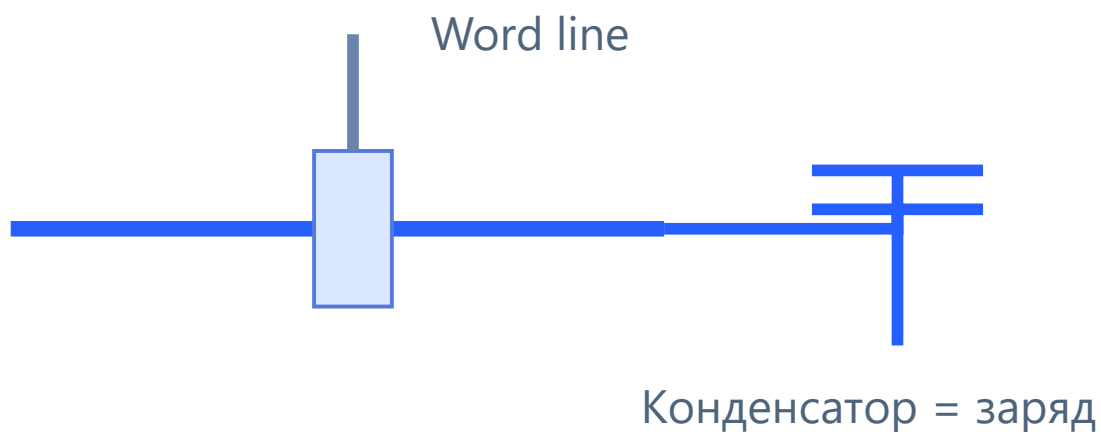
## Как хранится бит: SRAM и DRAM

**SRAM (Static RAM)** удерживает логическое состояние с помощью электронной схемы — обычно нескольких транзисторов, образующих ячейку-защёлку. Пока подаётся питание, состояние сохраняется без периодического обновления. За это приходится платить площадью кристалла и стоимостью: SRAM обычно применяют для относительно небольших быстрых кэшей.

**DRAM (Dynamic RAM)** хранит информацию как электрический заряд в миниатюрном конденсаторе. Для обращения к заряду используется транзистор. Заряд постепенно утекает, поэтому данные нужно **регулярно регенерировать (refresh)**. Чтение DRAM тоже связано с восстановлением состояния ячейки. Более компактная ячейка позволяет создать много гигабайт относительно недорогой памяти.

**SRAM: логическая защёлка**

Состояние поддерживается взаимной обратной связью двух инвертирующих узлов (упрощённое изображение).

**DRAM: транзистор и конденсатор**

Управляющая линия открывает транзистор, позволяя читать или записывать заряд. Для хранения требуется refresh.

Функциональные схемы, а не подробные схемы микросхем. Оба вида обычно энергозависимы.

**Интерактив: почему DRAM нужна регенерация**

Сымитируйте утечку заряда. Порог 40% и шаг 22% выбраны **только для наглядности**; реальные ячейки и контроллеры намного сложнее.



Заряд: **100%** · Интерпретация: **1 (распознаётся)**

Исходно записана 1. Модель упрощена: время не соответствует реальной DRAM.

| Свойство                | SRAM                          | DRAM                                         |
|-------------------------|-------------------------------|----------------------------------------------|
| Физическая ячейка       | Транзисторная защёлка         | Заряд конденсатора и доступ через транзистор |
| Refresh                 | Не нужен для удержания данных | Необходим периодически                       |
| Плотность / цена за бит | Ниже / выше                   | Выше / ниже                                  |
| Обычное применение      | Кэш CPU                       | Системное ОЗУ                                |

## 04

## Адреса, чтение и запись

Представьте память как длинный ряд пронумерованных ячеек. У каждой области есть **адрес**: по нему процессор запрашивает чтение или запись. В большинстве современных систем память **байтово-адресуемая** — отдельный адрес соответствует байту. При этом физическая DRAM внутри устроена гораздо сложнее: строки, столбцы, банки и группы банков.

1. Процессор инициирует обращение к адресу (обычно сначала проверяется кэш).
2. Контроллер памяти выбирает нужные микросхемы, банки, строки и столбцы в DRAM.
3. При чтении данные передаются к процессору; при записи контроллер изменяет содержимое.
4. При необходимости контроллер выполняет служебные операции, в том числе refresh.



### Интерактив: 32 адресуемых бита (учебная модель)

Щёлкните по ячейке, затем запишите 0 или 1. Это модель **битовых** ячеек для обучения, не размер обычного адресуемого элемента современной ОЗУ.

0

1

2

3

4

5

6

7

Адрес 0 (двоичный 00000): бит 0

**Пример с байтами:** если по адресу 100 расположен байт  $10101010_2$ , это восемь бит. Следующий байт обычно имеет адрес 101. Для чтения числа размером четыре байта процессор может обращаться к диапазону адресов 100–103 (с учётом порядка байтов и правил архитектуры).

## 05

## Главные характеристики памяти

Скорость компьютера нельзя определить только объёмом ОЗУ или одной цифрой «МГц». Важно сразу несколько параметров.

| Характеристика         | Что означает                                  | Пример                                                             |
|------------------------|-----------------------------------------------|--------------------------------------------------------------------|
| Ёмкость                | Сколько данных может храниться                | 16 ГБ RAM, 1 ТБ SSD                                                |
| Задержка (latency)     | Сколько времени проходит до получения ответа  | Измеряется в нс или тактах; зависит от доступа                     |
| Пропускная способность | Сколько данных передаётся за секунду          | Например, 25,6 ГБ/с теоретически для DDR4-3200 на 64-битном канале |
| Скорость передачи      | Число передач в секунду                       | DDR4-3200: 3200 MT/c                                               |
| Тайминги               | Задержки разных операций, выраженные в циклах | CL, tRCD, tRP, tRAS                                                |
| Энергопотребление      | Сколько энергии требуется                     | Существенно для ноутбука и сервера                                 |
| Надёжность             | Обнаружение ошибок, исправление, ресурс       | ECC у поддерживаемых платформ                                      |

Ёмкость: не путайте бит и байт

**1 байт = 8 бит.** В десятичном обозначении 1 GB = 1 000 000 000 байт; в двоичном 1 GiB = 1 073 741 824 байта. В интерфейсах иногда используют «ГБ» для величин, фактически соответствующих GiB. Пример: 16 GiB ОЗУ = 16 × 1024<sup>3</sup> байт.

Задержка и пропускная способность — это разное

Две аналогии: задержка — сколько вы ждёте первый автобус, пропускная способность — сколько пассажиров все автобусы перевозят за час. Высокая пропускная способность не означает, что первое обращение обязательно быстрее.

Условное сравнение требований к скорости и ёмкости



Схема показывает относительную ёмкость в обычном ПК; ширины столбцов условные, не в единой шкале.

## DDR, MT/с и тайминги: как читать характеристики

**DDR (Double Data Rate)** — семейство технологий памяти, передающих данные дважды за период базового тактового сигнала. Поэтому для скорости модулей лучше использовать **MT/с (миллионы передач в секунду)**, а не путать её с частотой внутренних тактов. DDR4 и DDR5 — разные поколения, механически и электрически несовместимые.

Для обычного 64-битного канала без учёта служебных накладных расходов теоретическая пропускная способность считается так:

Ширина: 64 бит = 8 байт

Пропускная способность (ГБ/с) =  $\text{MT/с} \times 8 / 1000$

Пример: DDR4-3200  $\rightarrow 3200 \times 8 / 1000 = 25,6$  ГБ/с

Тайминг **CL** — задержка CAS в тактах памяти после определённой команды чтения. Значение CL само по себе недостаточно для сравнения разных скоростей. Для DDR-3200 (физический такт передачи 1600 МГц) период составляет 0,625 нс; CL16 соответствует примерно **10 нс только для этой CAS-составляющей**. Полная задержка чтения больше и зависит от состояния строки, контроллера и иных факторов.

Период такта (нс)  $\approx 2000 / \text{скорость DDR (MT/с)}$

Часть задержки CAS (нс)  $\approx \text{CL} \times 2000 / \text{MT/с}$

DDR4-3200 CL16  $\rightarrow 16 \times 2000 / 3200 = 10$  нс

Нельзя заключить, что комплект DDR5-5600 CL46 обязательно быстрее во всех задачах, чем DDR4-3200 CL16. У систем разные контроллеры, платформы и пропускная способность; требуется сравнение реальных сценариев.

### 07

## Почему компьютеру не хватает памяти и что такое кэш

Процессор может выполнять операции быстрее, чем большая DRAM успевает выдавать данные. Чтобы не ждать ОЗУ на каждом шаге, используют **кэш L1, L2 и L3**. При **cache hit** нужные данные уже находятся в кэше; при **cache miss** приходится обращаться к следующему уровню, а иногда и к RAM.

Кэш использует принцип **локальности**: если программа обратилась к данным, то нередко скоро обратится к ним же (*временная локальность*) или к соседним адресам (*пространственная локальность*). Поэтому кэш обычно переносит не один случайный байт, а целую **кэш-линию** (например, 64 байта в типичных x86-системах).



Уровни показаны последовательно для понимания идеи. Реальные схемы могут быть инклюзивными, неинклюзивными и с несколькими путями обращения.

### Что студент должен уметь объяснить

RAM нужна не вместо SSD, а между SSD и процессором: она достаточно объёмная для текущей работы и быстрее долговременного хранилища. Кэш нужен не вместо RAM, а чтобы чаще не обращаться к ней. Регистры необходимы для данных, непосредственно участвующих в исполнении команд.

08

## Практическая ситуация: выбираем память для задачи

| Ситуация                                                   | Что важнее и почему                                                                         |
|------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| Открыты десятки вкладок, редактор и виртуальная машина     | Достаточный <b>объём RAM</b> : меньше риска, что ОС начнёт часто выгружать страницы на диск |
| Вычисления над большими массивами данных                   | Важны объём, задержка, кэш-локальность и <b>пропускная способность</b>                      |
| Хранение фотографий после выключения                       | Нужна <b>энергонезависимая</b> память, например SSD                                         |
| Ускорить повторяющееся вычисление на небольшом массиве     | Важны <b>кэш</b> , локальность доступа, алгоритм и процессор                                |
| Сервер должен обнаруживать типичные одиночные ошибки битов | Платформа с соответствующей поддержкой <b>ECC RAM</b> и мониторингом ошибок                 |

Итоговая цепочка: **физический способ хранения** → **свойства памяти** → **роль в системе** → **показатели производительности**. Если различаете эти четыре уровня, вы сможете осмысленно читать спецификации железа.

### ЧАСТЬ II • ПРОВЕРКА ЗНАНИЙ

## Тест по изученному материалу

Теоретическая часть завершена. Выберите один ответ в каждом вопросе, затем нажмите «Проверить тест». Результат покажет число правильных ответов и объяснит ошибки. В конце ответьте на развёрнутые вопросы — их проверяет преподаватель.

17 из 18 ответов

Проверка без штрафа за повторную попытку



## Часть А. Выберите правильный ответ

1. Какая память обычно используется в качестве основной оперативной памяти ПК?

☐ NAND Flash

☒ **DRAM**

☐ Магнитная лента

☐ Оптический диск

Правильный ответ: DRAM. ОЗУ компьютера обычно организовано на DRAM.

2. Что происходит с обычной DRAM после длительного отключения питания?

☐ Данные сохраняются навсегда

☒ **Данные теряются**

☐ Память превращается в ROM

☐ Данные записываются на SSD автоматически

Правильный ответ: Данные теряются. DRAM энергозависима; для сохранения содержимого ей нужно питание и refresh.

3. Зачем DRAM выполняет refresh?

☐ Для дефрагментации

☒ **Для восстановления постепенно исчезающего заряда**

☐ Для увеличения количества ячеек

☐ Для проверки интернета

Правильный ответ: Для восстановления постепенно исчезающего заряда. Заряд конденсаторов DRAM утекает, поэтому его надо регенерировать.

#### 4. Где обычно применяется SRAM?

- ☒ В кэше процессора

☐ В магнитных пластинах HDD

☐ В оптических DVD

☐ Только в SSD

Правильный ответ: В кэше процессора. SRAM быстрее, но дороже за бит и занимает больше площади, поэтому подходит для кэшей.

#### 5. Чем задержка отличается от пропускной способности?

☐ Ничем

- ☒ Задержка — ожидание ответа; пропускная способность — объём данных за время

☐ Задержка измеряется только в гигабайтах

☐ Пропускная способность означает сохранность данных

Правильный ответ: Задержка — ожидание ответа; пропускная способность — объём данных за время. Это два независимых свойства: время одного обращения и скорость потока данных.

#### 6. Что означает 3200 MT/c?

- ☒ 3200 миллионов передач в секунду

☐ 3200 мегабайт объёма

☐ 3200 наносекунд задержки

☐ Число ядер процессора

Правильный ответ: 3200 миллионов передач в секунду. MT/c — миллионы передач данных в секунду.

### 7. Какова теоретическая пропускная способность DDR4-3200 по одному 64-битному каналу?

- ☐ 3,2 ГБ/с
- ☐ 6,4 ГБ/с
- ☒ 25,6 ГБ/с
- ☐ 3200 ГБ/с

Правильный ответ: 25,6 ГБ/с.  $3200 \times 8 \text{ байт} \div 1000 = 25,6 \text{ ГБ/с}$ .

### 8. Сколько бит в одном байте?

- ☐ 2
- ☐ 4
- ☒ 8
- ☐ 16

Правильный ответ: 8. 1 байт = 8 бит.

### 9. Что такое попадание в кэш (cache hit)?

- ☒ Искомые данные обнаружены в кэше
- ☐ Все данные удалены
- ☐ Запущен антивирус
- ☐ Завершена запись на диск

Правильный ответ: Искомые данные обнаружены в кэше. Если данные уже в кэше, обращение к следующему медленному уровню не нужно.

**10. Как называется повторное использование недавно запрошенных данных?**☐ Износ☒ **Временная локальность**☐ Шифрование☐ Сегментация

Правильный ответ: Временная локальность. Временная локальность — высокая вероятность скорого повторного обращения.

**11. Какая память подходит для хранения документов после выключения?**☐ Регистры CPU☐ Кэш L1☐ DRAM☒ **SSD**

Правильный ответ: SSD. SSD обеспечивает энергонезависимое хранение файлов.

**12. В типичном ПК что быстрее при доступе к данным?**☒ **L1-кэш**☐ HDD☐ SSD SATA☐ Оптический диск

Правильный ответ: L1-кэш. L1-кэш расположен ближе к вычислительным блокам и имеет минимальные задержки.

### 13. Что означает CL16 в маркировке модуля?

- ☐ Ёмкость 16 ГБ
- ☒ Часть задержки CAS в 16 тактах

- ☐ 16 независимых каналов
- ☐ 16 потоков CPU

Правильный ответ: Часть задержки CAS в 16 тактах. CL измеряет один из таймингов DRAM в циклах, а не общую задержку или ёмкость.

### 14. Почему нельзя сравнивать две DDR-памяти только по CL?

- ☐ CL измеряет температуру
- ☒ Нужно учитывать скорость передачи и другие задержки

- ☐ CL относится только к SSD
- ☐ Тайминги не существуют

Правильный ответ: Нужно учитывать скорость передачи и другие задержки. Одинаковое число циклов при разной частоте даёт разное время в нс.

### 15. Какая формулировка верна?

- ☐ ОЗУ и SSD — один и тот же ресурс
- ☐ Большой SSD автоматически увеличивает объём ОЗУ
- ☒ ОЗУ хранит рабочие данные; SSD — долговременные
- ☐ RAM всегда энергонезависима

Правильный ответ: ОЗУ хранит рабочие данные; SSD — долговременные. Это разные уровни хранения данных с разными задачами.

## 15 из 15 (100%)

Вы ответили на 15 из 15 тестовых вопросов. Автоматическая оценка учитывает только вопросы с выбором ответа; развёрнутые ответы проверяет преподаватель.

Вопросы ниже подсвечены: зелёная рамка — верно, красная — ошибка, жёлтая — без ответа.



## Часть Б. Объясните своими словами

Эти вопросы нужны, чтобы проверить понимание, а не угадывание. Автоматическая оценка их не учитывает.

**Развёрнутый вопрос 1. Объясните разницу SRAM и DRAM: как каждая хранит бит, почему одной нужен refresh, а другой нет.**

**Ваш ответ (проверяет преподаватель)**

DRAM хранит бит в конденсаторе. Заряд из него постоянно утекает, поэтому памяти нужен refresh иначе данные просто пропадут.

SRAM хранит бит в системе транзисторов. Конденсаторов там нет, заряд никуда не девается, поэтому refresh не нужен — память держит данные стабильно, пока есть питание.

**Развёрнутый вопрос 2. Компьютер с 8 ГБ ОЗУ и SSD 1 ТБ тормозит при 30 открытых вкладках и двух виртуальных машинах. Объясните вероятную причину и аргументируйте, какое улучшение поможет.**

**Ваш ответ (проверяет преподаватель)**

Причина: Компьютеру не хватает 8 ГБ ОЗУ для вкладок и виртуальных машин. Система начинает использовать файл подкачки на диске, а SSD работает в сотни раз медленнее оперативки, из-за чего всё виснет.

Чтобы решить данную проблему нужно увеличить объем ОЗУ чтобы все процессы помещались в оперативку.

**Развёрнутый вопрос 3. Вычислите теоретическую пропускную способность DDR4-3200 по одному 64-битному каналу и примерную CAS-составляющую задержки для CL16. Укажите формулы.**

**Ваш ответ (проверяет преподаватель)**

Не заполнено

## Источники для дополнительного изучения

Материал адаптирован для обучения. Схемы и модели — упрощённые учебные примеры; значения производительности зависят от конкретного устройства.

1. [OpenStax · Memory Hierarchy](#)
2. [IBM · Hardware hierarchy](#)
3. [IBM · Primary Storage](#)

## Перед сдачей

✓ Указаны ФИО, группа и дата.

✓ Отмечены ответы на вопросы с вариантами и нажата кнопка проверки.

✓ Написаны объяснения в открытых вопросах.

Нажмите «**Отправить работу — сохранить PDF**». Откроется окно печати: в качестве принтера выберите «**Сохранить как PDF**». В отчёте будут ФИО, выбранные варианты, балл и ваши развёрнутые ответы. Отправка на сервер не выполняется.

В диалоге печати выберите «Сохранить как PDF».