

Лекция 3. Память

Основная память (ОП). Она предназначена для хранения и оперативного обмена информацией с прочими блоками компьютера. ОП содержит два вида запоминающих устройств: постоянное запоминающее устройство (ПЗУ) и оперативное запоминающее устройство (ОЗУ).

1 Оперативная память

Оперативная память, или *оперативное запоминающее устройство (ОЗУ)*, предназначено для приема, хранения и выдачи информации и представляет собой самую быстродействующую запоминающую систему компьютера. Оперативная память обозначается RAM (Random Access Memory — память с Произвольным доступом). Процессор имеет возможность выполнять программы только после того, как они загружены в оперативную рабочую память, т.е. и память, доступную для программ пользователя. CPU имеет непосредственный доступ к данным, находящимся в оперативной памяти, а к внешней памяти (на гибких или жестких дисках) — через буфер, являющийся также разновидностью оперативной памяти. Работа программ, загруженных с внешнего носителя, возможна только после того, как она будет скопирована в RAM.

Однако оперативная память имеет существенный недостаток, заключающийся в том, что она временная, т.е. при отключении питания оперативная память полностью очищается. При этом данные, не записанные на внешний носитель, будут утеряны. Основная задача RAM — предоставлять необходимую информацию в виде двоичных кодов по запросам CPU, т.е. данные в любой момент должны быть доступны для обработки.

Существует два типа ОЗУ: статическое и динамическое. Статическое ОЗУ (Static RAM, SRAM) конструируется с использованием D-триггеров. Информация в ОЗУ сохраняется на протяжении всего времени, пока к нему подается питание: секунды, минуты, часы и даже дни. Статическое ОЗУ работает очень быстро. Обычно время доступа составляет несколько наносекунд. По этой причине статическое ОЗУ часто используется в качестве кэш-памяти второго уровня.

В динамическом ОЗУ (Dynamic RAM, DRAM), напротив, триггеры не используются. Динамическое ОЗУ представляет собой массив ячеек, каждая из которых содержит транзистор и крошечный конденсатор. Конденсаторы могут быть заряженными и разряженными, что позволяет хранить нули и единицы. Поскольку электрический заряд имеет тенденцию исчезать, каждый бит в динамическом ОЗУ должен обновляться (перезаряжаться) каждые несколько миллисекунд, чтобы предотвратить утечку данных. Поскольку об обновлении должна заботиться внешняя логика, динамическое ОЗУ требует более сложного сопряжения, чем статическое, хотя этот недостаток компенсируется большим объемом.

Поскольку динамическому ОЗУ нужен только 1 транзистор и 1 конденсатор на бит (статическому ОЗУ требуется в лучшем случае транзисторов на бит), динамическое ОЗУ имеет очень высокую плотность записи (много битов на одну микросхему). По этой причине основная память почти всегда строится на основе динамических ОЗУ. Однако динамические ОЗУ работают очень медленно (время доступа занимает десятки наносекунд). Таким образом, сочетание кэш-памяти на основе статического ОЗУ и основной памяти на основе динамического ОЗУ соединяет в себе преимущества обоих устройств.

Конструктивно оперативная память выполняется и виде модулей микросхем, что позволяет дополнять объем оперативной памяти, которая используется не только в ПК, но и в самых разных периферийных устройствах— от видеокарт до лазерных принтеров. Микросхемы оперативной

памяти и этом случае могут принадлежать к разным модификациям, но все они относятся к типу динамической оперативной памяти (DRAM).

Синхронные динамические ОЗУ (Synchronous DRAM, SDRAM). Синхронное динамическое ОЗУ управляется одним синхронизирующим сигналом. Данное устройство представляет собой гибрид статического и динамического ОЗУ. Основное преимущество синхронного динамического ОЗУ состоит в том, что оно исключает зависимость микросхемы памяти от управляющих сигналов. ЦП сообщает памяти, сколько циклов следует выполнить, а затем запускает эти циклы. Каждый цикл на выходе дает 4, 8 или 16 бит в зависимости от количества выходных строк. Устранение зависимости от управляющих сигналов приводит к увеличению скорости передачи данных между ЦП и памятью.

Следующим этапом в развитии памяти SDRAM стала память DDR (Double Data Rate — передача данных с двойной скоростью).

DDR SDRAM (от англ. Double Data Rate Synchronous Dynamic Random Access Memory — синхронная динамическая память с произвольным доступом и удвоенной скоростью передачи данных) — тип компьютерной памяти используемой в вычислительной технике в качестве оперативной и видеопамати. При использовании DDR SDRAM достигается удвоенная скорость работы, нежели в SDRAM, за счёт считывания команд и данных не только по фронту, как в SDRAM, но и по спаду тактового сигнала. За счёт этого удваивается скорость передачи данных без увеличения частоты тактового сигнала шины памяти. Таким образом, при работе DDR на частоте 100 МГц мы получим эффективную частоту 200 МГц (при сравнении с аналогом SDR SDRAM).

2 Постоянная память

ПЗУ (постоянное запоминающее устройство), или ROM (Read-Only Memory — постоянная память). ПЗУ служит для хранения неизменяемой (постоянной) программной и справочной информации, позволяет оперативно только считывать хранящуюся в нем информацию. Содержание памяти специальным образом «зашивается» в устройстве при его изготовлении для постоянного хранения.

Прежде всего в постоянную память записывают программу управления работой самого процессора. В ПЗУ находятся программы управления дисплеем, клавиатурой, принтером, внешней памятью, программы запуска и остановки компьютера, тестирования устройств. Данные записываются в ПЗУ в процессе производства. Для этого изготавливается трафарет с определенным набором битов, который накладывается на фоточувствительный материал, а затем открытые (или закрытые) части поверхности вытравливаются. Единственный способ изменить программу в ПЗУ — поменять всю микросхему.

Однако чтобы компаниям было проще разрабатывать новые устройства, основанные на ПЗУ, были выпущены **программируемые ПЗУ (Programmable ROM, PROM)**. В отличие от обычных ПЗУ, их можно программировать в условиях эксплуатации, что позволяет сократить время выполнения заказа.

Следующая разработка этой линии — **стираемое программируемое ПЗУ (Erasable PROM, EPROM)**, которое можно программировать в условиях эксплуатации, а также стирать с него информацию. Если кварцевое окно в данном ПЗУ подвергать воздействию сильного ультрафиолетового света в течение 15 минут, все биты установятся в 1. Если нужно сделать много изменений во время одного этапа проектирования, стираемые ПЗУ гораздо экономичнее, чем обычные программируемые ПЗУ, поскольку их можно использовать многократно.

Следующий этап — **электронно-перепрограммируемое ПЗУ** (Electronically EPROM, EEPROM), с которого можно стирать информацию, не подвергая воздействию ультрафиолетовых лучей. Кроме того, чтобы перепрограммировать данное устройство, его не нужно вставлять в специальный аппарат для программирования, в отличие от стираемого программируемого ПЗУ.

Более современный тип электронно-перепрограммируемого ПЗУ — **флэш-память**. **Перепрограммируемая постоянная память (Flash Memory)** — энергонезависимая память. Важнейшая микросхема Flash-памяти — модуль BIOS. Роль BIOS двоякая: с одной стороны это неотъемлемый элемент аппаратуры, а с другой стороны — важный модуль любой операционной системы. BIOS (Basic Input/Output System — базовая система ввода-вывода) — совокупность программ, предназначенных для автоматического тестирования устройств после включения питания компьютера и загрузки операционной системы в оперативную память.

Наиболее широко среди пользователей компьютеров известна BIOS материнской платы, но BIOS присутствуют почти у всех компонентов компьютера: у видеоадаптеров, сетевых адаптеров, модемов, дисковых контроллеров, принтеров. По своей сути BIOS является посредником между аппаратным и программным обеспечением компьютера.

Главная функция BIOS материнской платы — инициализация устройств, подключённых к материнской плате, сразу после включения питания компьютера. BIOS проверяет работоспособность устройств (т. н. самотестирование, англ. POST — Power-On Self Test), задаёт низкоуровневые параметры их работы (например, частоту шины центрального микропроцессора), и после этого ищет загрузчик операционной системы (англ. Boot Loader) на доступных носителях информации и передаёт управление операционной системе. Операционная система по ходу работы может изменять большинство настроек, изначально заданных в BIOS. В некоторых реализациях BIOS позволяет производить загрузку операционной системы через интерфейсы, изначально для этого не предназначенные, в том числе USB и IEEE 1394. Также возможна загрузка по сети (применяется, например, в т. н. «тонких клиентах»).

Также BIOS содержит минимальный набор сервисных функций (например, для вывода сообщений на экран или приёма символов с клавиатуры), что и обуславливает расшифровку её названия: Базовая система ввода-вывода.

В некоторых BIOS'ах реализуется дополнительная функциональность (например, воспроизведение аудио-CD или DVD-дисков), поддержка встроенной рабочей среды (например, интерпретатор языка Basic) и др.

В отличие от стираемого ПЗУ, которое стирается под воздействием ультрафиолетовых лучей, и от электронно-перепрограммируемого ПЗУ, которое стирается по байтам, флэш-память стирается и записывается блоками.

3 Кэш-память

Процессоры всегда работали быстрее, чем память. Так как процессоры и память совершенствуются параллельно, это несоответствие сохраняется. На практике такое несоответствие в скорости работы приводит к тому, что, когда процессор обращается к памяти, проходит несколько машинных циклов, прежде чем он получит запрошенное слово. Чем медленнее работает память, тем дольше процессору приходится ждать, тем больше циклов проходит.

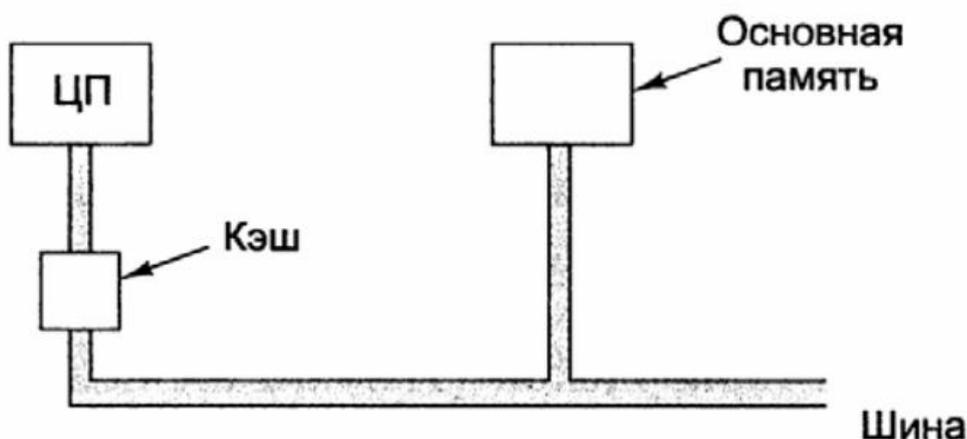
КЭШ-память — это буферная, не доступная для пользователя быстродействующая память, автоматически используемая компьютером для ускорения операций с информацией, хранящейся в более медленно действующих запоминающих устройствах. Например, для ускорения операций с

основной памятью организуется регистровая КЭШ-память внутри микропроцессора (КЭШ-память первого уровня) или вне микропроцессора на материнской плате (КЭШ-память второго уровня); для ускорения операций с дисковой памятью организуется КЭШ-память на ячейках электронной памяти.

Принцип его действия основан на том, что простой (не работа) более быстрого устройства сильно влияет на общую производительность (**во-первых**) и что скорее всего запрашиваются данные, сохраненные недавно (**во-вторых**). Поэтому между устройствами помещают буфер быстрой памяти (небольшой по сравнению со всеми хранимыми данными), что позволяет снизить потери быстрого устройства, как на записи, так и на чтении.

Основная идея кэш-памяти проста: в ней находятся слова, которые чаще всего используются. Если процессору нужно какое-нибудь слово, сначала он обращается к кэш-памяти. Только в том случае, если слова там нет, он обращается к основной памяти. Если значительная часть слов находится в кэш-памяти, среднее время доступа значительно сокращается. Таким образом, успех или неудача зависит от того, какая часть слов находится в кэш-памяти. Давно известно, что программы не обращаются к памяти наугад. Если программе нужен доступ к адресу А, то скорее всего после этого ей понадобится доступ к адресу, расположенному поблизости от А. Практически все команды обычной программы (за исключением команд перехода и вызова процедур) вызываются из последовательных областей памяти. Кроме того, большую часть времени программа тратит на циклы, когда ограниченный набор команд выполняется снова и снова. Точно так же при манипулировании матрицами программа скорее всего будет обращаться много раз к одной и той же матрице, прежде чем перейдет к чему-либо другому.

Ситуация, когда при последовательных обращениях к памяти в течение некоторого промежутка времени используется только небольшая ее область, называется **принципом локальности**. Этот принцип составляет основу всех систем кэш-памяти. Идея состоит в том, что когда определенное слово вызывается из памяти, оно вместе с соседними словами переносится в кэш-память, что позволяет при очередном запросе быстро обращаться к следующим словам. Общее устройство процессора, кэш-памяти и основной памяти иллюстрирует рисунок 1.



Если слово считывается или записывается k раз, компьютеру требуется сделать 1 обращение к медленной основной памяти и $k - 1$ обращений к быстрой кэш-памяти. Чем больше k , тем выше общая производительность.

Мы можем сделать и более строгие вычисления. Пусть s — время доступа к кэш-памяти, m — время доступа к основной памяти и h — коэффициент кэш-попаданий, который показывает соотношение числа обращений к кэш-памяти и общего числа всех обращений к памяти. В нашем примере $h = (k - 1)/k$.

Некоторые авторы выделяют **коэффициент кэш-промахов**, равный $1-h$. Таким образом, мы можем вычислить среднее время доступа:

$$t_{\text{дост}} = c + (1 - h) \cdot m$$

Если $h \rightarrow 1$, то есть все обращения делаются только к кэш-памяти, то время доступа стремится к c . С другой стороны, если $h \rightarrow 0$, то есть каждый раз нужно обращаться к основной памяти, то время доступа стремится к $c+m$: сначала требуется время c для проверки кэш-памяти (в данном случае безуспешной), а затем — время m для обращения к основной памяти. В некоторых системах обращение к основной памяти может начинаться параллельно с использованием кэш-памяти, чтобы в случае кэш-промаха цикл обращения к основной памяти уже начался. Однако эта стратегия требует способности останавливать процесс обращения к основной памяти в случае кэш-попадания, что усложняет разработку подобного компьютера.

Основная память и кэш-память делятся на блоки фиксированного размера с учетом *принципа локальности*. Блоки внутри кэш-памяти обычно называют **строками КЭШа (cache lines)**. При кэш-промахе из основной памяти в кэш-память загружается вся строка, а не только необходимое слово. Например, если строка состоит из 64 байт, обращение к адресу 260 влечет за собой загрузку в кэш-память всей строки (байты с 256 по 319) на случай, если через некоторое время понадобятся другие слова из этой строки. Такой путь обращения к памяти более эффективен, чем вызов каждого слова по отдельности, потому что однократный вызов k слов происходит гораздо быстрее, чем вызов одного слова k раз.

Кэш-память очень важна для высокопроизводительных процессоров. Однако здесь возникает ряд вопросов.

Первый вопрос — **объем кэш-памяти**. Чем больше объем, тем лучше работает память, но тем дороже она стоит.

Второй вопрос — **размер строки кэша**. Кэш-память объемом 16 Кбайт можно разделить на 1024 строки по 16 байт, 2048 строк по 8 байт и т. д.

Третий вопрос — **механизм организации кэш-памяти**, то есть то, как она определяет, какие именно слова находятся в ней в данный момент.

Четвертый вопрос — **должны ли команды и данные находиться вместе в общей кэш-памяти**. Проще всего разработать объединенную кэш-память (unified cache), в которой будут храниться и данные и команды. В этом случае вызов команд и данных автоматически уравнивается.

Однако в настоящее время существует тенденция к использованию разделенной кэш-памяти (split cache), когда команды хранятся в одной кэш-памяти, а данные — в другой. Такая архитектура также называется **гарвардской (Harvard architecture)**, поскольку идея использования отдельной памяти для команд и отдельной памяти для данных впервые воплотилась в компьютере Маге III, который был создан Говардом Айкеном (Howard Aiken) в Гарварде. Современные разработчики пошли по этому пути, поскольку сейчас широко распространены конвейерные архитектуры, а при конвейерной организации должна быть возможность одновременного доступа и к командам, и к данным (операндам). Разделенная кэш-память позволяет осуществлять параллельный доступ, а общая — нет. К тому же, поскольку команды обычно не меняются во время выполнения программы, содержание кэша команд не приходится записывать обратно в основную память.

Наконец, *пятый вопрос* — **количество блоков кэш-памяти**. В настоящее время очень часто кэш-память первого уровня располагается прямо на микросхеме процессора, кэш-память второго

уровня — не на самой микросхеме, но в корпусе процессора, а кэш-память третьего уровня — еще дальше от процессора.

Кэш-памятью управляет специальное устройство — контроллер, который, анализируя выполняемую программу, пытается предвидеть, какие данные и команды вероятнее всего понадобятся в ближайшее время процессору, и подкачивает их в кэш-память.

4 Внешняя память

Внешняя память относится к внешним устройствам ПК и используется для долговременного хранения любой информации, которая может когда-либо потребоваться для решения задач. В частности, во внешней памяти хранится все программное обеспечение компьютера. Внешняя память содержит разнообразные виды запоминающих устройств, но наиболее распространенными, имеющимися практически на любом компьютере, являются накопители на жестких (НЖМД).

Магнитные диски.

Магнитный диск состоит из одной или нескольких алюминиевых поверхностей, в настоящее время компания IBM делает их из стекла, покрытых магнитным слоем. Изначально их диаметр составлял 50 см, сейчас — от 3 до 12 см, у портативных компьютеров — меньше 3 см, причем это значение продолжает уменьшаться. Головка диска, содержащая индукционную катушку, движется над поверхностью диска, опираясь на воздушную подушку. Когда головка проходит над намагниченной областью, в ней (в головке) возникает положительный или отрицательный ток, что дает возможность считывать записанные ранее биты. Поскольку диск вращается под головкой, поток битов может записываться, а потом считываться.

Конфигурация дорожки диска показана на рисунке 2.



Дорожкой называется круговая последовательность битов, записанных на диск за его полный оборот. Каждая дорожка делится на секторы фиксированной длины.

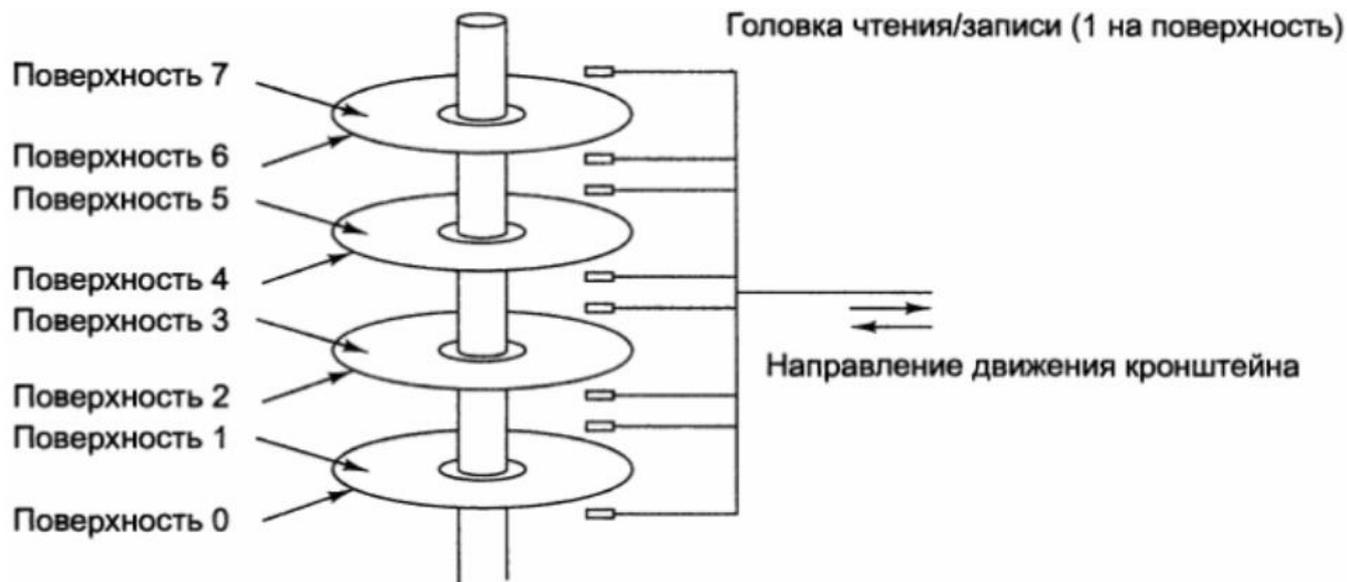
Каждый сектор обычно содержит 512 байт данных. Перед данными располагается преамбула (preamble), которая позволяет головке синхронизироваться перед чтением или записью. После данных идет код исправления ошибок (Error-Correcting Code, ECC), позволяющий исправлять множественные ошибки. Между соседними секторами находится межсекторный интервал. Емкость

форматированного диска обычно на 15% меньше емкости неформатированного, на нем учитываются преамбулы, ЕСС-коды и межсекторные интервалы. У всех дисков есть кронштейны, они могут перемещаться туда и обратно по радиусу на разные расстояния от шпинделя, вокруг которого вращается диск. На разных расстояниях от оси записываются разные дорожки. Таким образом, дорожки представляют собой ряд концентрических кругов, расположенных вокруг шпинделя. Ширина дорожки зависит от величины головки и от точности ее перемещения. На сегодняшний момент диски имеют от 5000 до 10 000 дорожек на см, то есть ширина каждой дорожки составляет от 1 до 2 микрон.

Следует отметить, что дорожка — это не углубление на поверхности диска, а просто кольцо намагниченного материала, которое отделяется от других дорожек небольшими пограничными областями. Плотность записи битов на концентрических дорожках различается в зависимости от расстояния от центра диска и зависит главным образом от качества поверхности диска и чистоты воздуха.

Чтобы достичь высокого качества поверхности и достаточной чистоты воздуха, диски герметично закрываются. Такие диски называются винчестерами.

Большинство магнитных дисков состоит из нескольких пластин, расположенных друг под другом, как показано на рисунке 3.



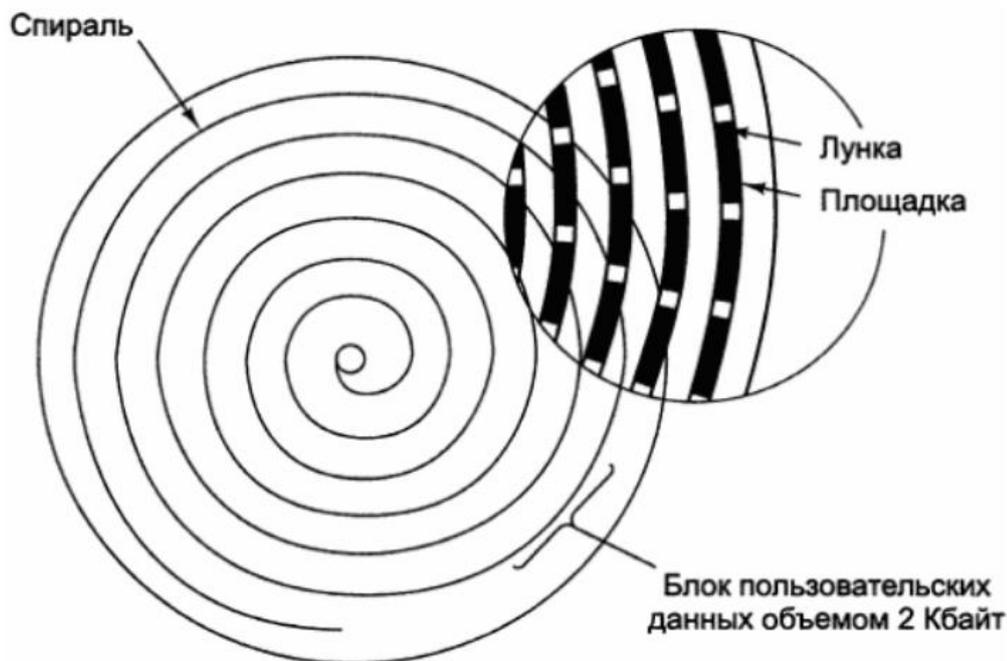
Каждая поверхность снабжена кронштейном и головкой. Кронштейны скреплены таким образом, что одновременно могут перемещаться на разные расстояния от оси. Совокупность дорожек, расположенных на одном расстоянии от центра, называется **цилиндром**.

Диски CD-ROM.

Компакт-диск изготавливается с использованием очень мощного инфракрасного лазера, который выжигает отверстия диаметром 0,8 микрон в специальном стеклянном мастер-диске. По этому мастер-диску делается шаблон с выступами в тех местах, где лазер прожег отверстия. В шаблон вводится жидкая смола (поликарбонат), и, таким образом, получается компакт-диск с тем же набором отверстий, что и в стеклянном диске. На смолу наносится очень тонкий слой алюминия, который, в свою очередь, покрывается защитным лаком. После этого наклеивается этикетка. Углубления в нижнем слое смолы называются **лунками** (pits), а ровные пространства между лунками — **площадками** (lands).

Лунки и площадки записываются по спирали. Запись начинается на некотором расстоянии от отверстия в центре диска и продвигается к краю, занимая 32 мм диска. Спираль проходит 22 188 оборотов вокруг диска (примерно 600 на 1 мм).

Если спираль распрямить, ее длина составит 5,6 км. Спираль изображена на рисунке 4.



DVD диски.

Изначально аббревиатура DVD расшифровывалась как Digital Video Disk (цифровой видеодиск), сейчас она официально превратилась в Digital Versatile Disk (цифровой многоцелевой диск). DVD-диски в целом похожи на компакт-диски. Как и обычные компакт-диски, они имеют 120 мм в диаметре, создаются на основе поликарбоната и содержат лунки и площадки, которые освещаются лазерным диодом и считываются фотодетектором. Однако существует несколько различий:

- ◆ меньший размер лунок (0,4 микрона вместо 0,8 микрона, как у обычного компакт-диска);
- ◆ более плотная спираль (0,74 микрона между дорожками вместо 1,6 микрона);
- ◆ красный лазер (с длиной волны 0,65 микрона вместо 0,78 микрона).

Диски Blu-Ray.

Преемником DVD можно считать технологию Blu-Ray, предусматривающую применение синего лазера вместо красного. Синий лазер отличается более короткой длиной волны, а значит, повышенной точностью; за счет этого обстоятельства он позволяет уменьшать размеры лунок и площадок. На односторонних дисках BluRay умещается около 25 Гбайт данных; на двухсторонних— 50 Гбайт. Скорость передачи данных составляет 4,5 Мбит/с.

«Флэшки».

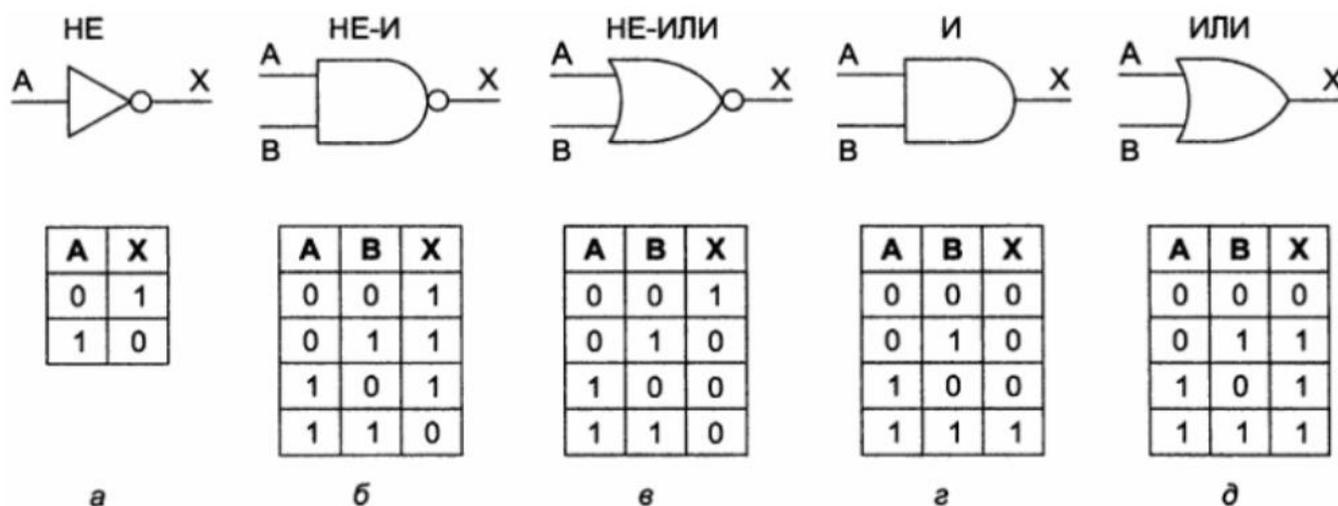
У классической flash карты (как usb flash, так и карты памяти Compact Flash, SD, MMC и т.п.) нет механических частей, она не нуждается ни в батарейках, ни в аккумуляторах, флешка это набор микросхем, в чипах которых способна храниться цифровая информация. Это устройство компактное, быстрое, дешёвое и не самое надёжное.

Каждая из микросхем флешки состоит из своего рода гнёзд (NAND Flash). При перезаписи информация записывается в одно и то же гнездо, и через некоторое время сектор может начать сбоить, что зачастую и приводит к потере данных — это самая большая проблема флешек, обратная сторона удобства и дешевизны.

Чем больше объем flash памяти, тем больше внутри микросхем. Все вместе они выстроены в рейд-массив, но алгоритм их сборки и взаимодействия значительно сложнее, чем классический рейд-массив из жестких дисков в компьютере. Более того, на каждой из микросхем для ускорения работы и повышения надёжности применяются алгоритмы схожие с принципами действия рейд-массивов. Вдобавок к этим сложностям во флешке существует самая главная микросхема — контроллер, управляющий всеми данными на flash, запоминающий какие секторы флешки уже начали сбоить, считающий контрольные суммы. При загрузке обычной фотографии на flash информация не выкладывается единым куском, а распространяется по многим «хранилищам» довольно причудливым образом. Контроллер «знает» о том, куда и какие части файла были положены.

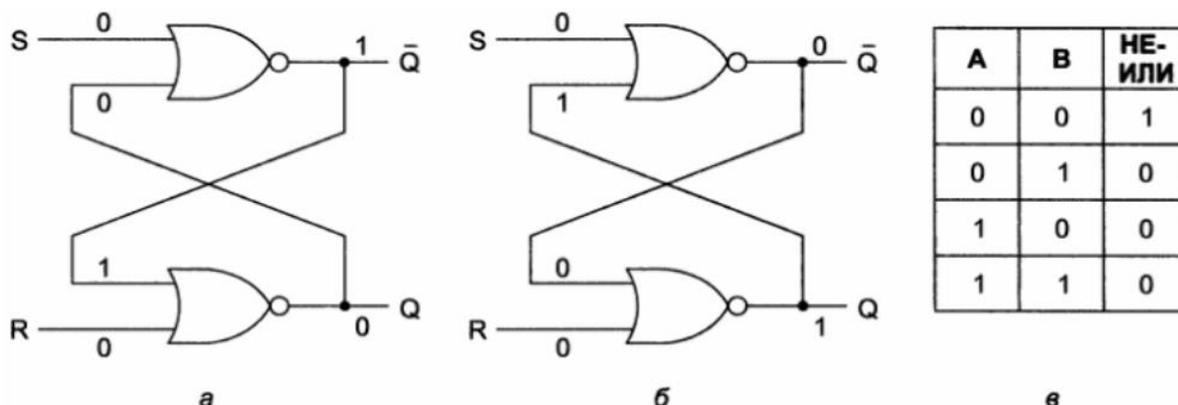
5 Основные компоненты памяти

Значки, которые используются для изображения трех основных типов вентиляей, показаны на рисунке 5 а-в. Там же показаны режимы работы функции для каждой схемы.



Здесь А и В — входные сигналы, X — выходной сигнал. Каждая строка таблицы определяет выходной сигнал для различных комбинаций входных сигналов.

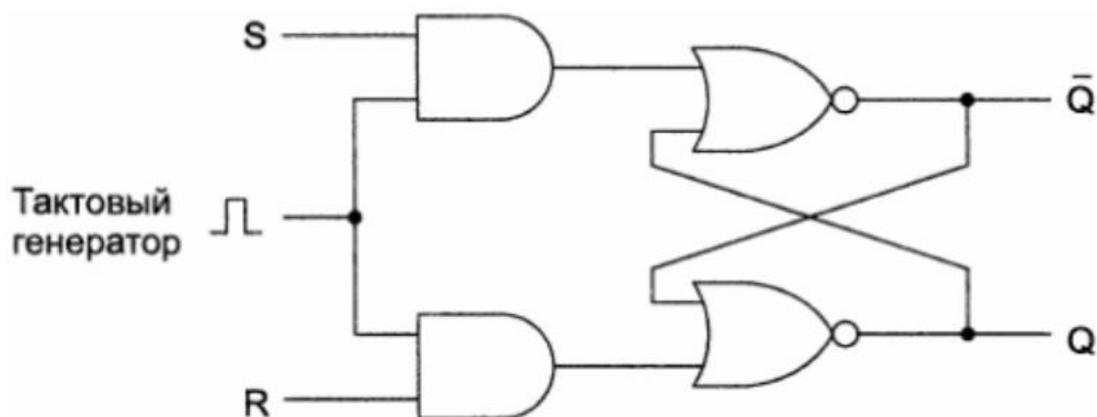
Защелки. Чтобы создать один бит памяти, нужна схема, которая каким-то образом «запоминает» предыдущие входные значения. Такую схему можно сконструировать из двух вентиляей НЕ-ИЛИ, как показано на рисунке 6 (а).



Аналогичные схемы можно построить из вентилях НЕ-И, которые, по существу, идентичны схемам с вентилями НЕ-ИЛИ.

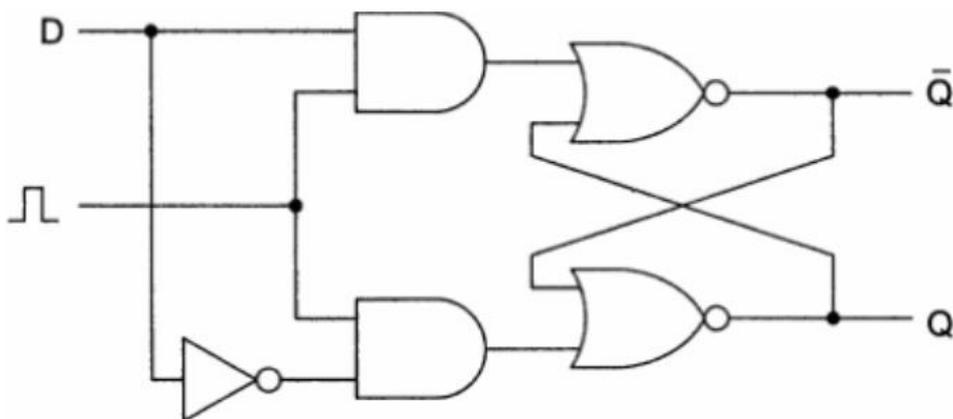
Схема, изображенная на рисунке, называется **SR-защелкой**. У нее есть два входа: S (Setting — установка) и R (Resetting — сброс). У нее также есть два комплиментарных выхода: Q и \bar{Q} .

Часто удобно, чтобы защелка меняла состояние только в определенные моменты. Чтобы достичь этой цели, немного изменим основную схему и получим **синхронную SR-защелку** (рисунок 7).



Эта схема имеет дополнительный синхронизирующий вход, который по большей части равен 0. Если этот вход равен 0, то оба выхода вентилях И равны 0, и независимо от значений S и R защелка не меняет свое состояние. Когда значение синхронизирующего входа равно 1, действие вентилях И прекращается, и состояние защелки становится зависимым от S и R. Для обозначения факта появления единицы на синхронизирующем входе часто используются термины **включение** и **стробирование**.

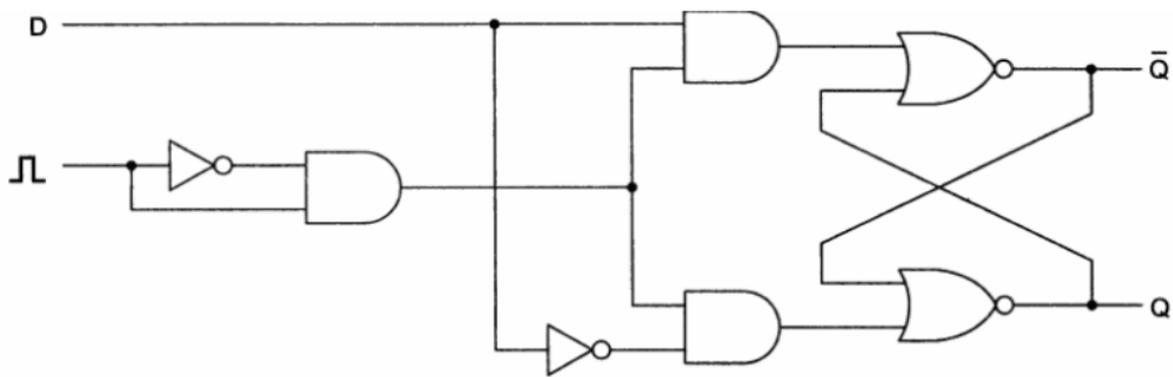
В случае, если $S = R = 1$ возникает неопределенность защёлки. Чтобы не дать ей возникнуть организуется схема только с одним входом. Такая схема (рисунок 8), которая называется **синхронной D-защелкой**, представляет собой память объемом 1 бит.



Триггеры. Многие схемы при необходимости выбирают значение на определенной линии в заданный момент времени и запоминают его. В такой схеме, которая называется **триггером** (flip-flop), смена состояния происходит не тогда, когда синхронизирующий сигнал равен 1, а при переходе синхронизирующего сигнала с 0 на 1 (фронт) или с 1 на 0 (спад). Следовательно, длина синхронизирующего импульса не имеет значения, поскольку переходы происходят быстро.

Различие между триггером и защелкой определяется тем, что триггер запускается перепадом сигнала, а защелка запускается уровнем сигнала.

Простейшая схема триггера (рисунок 9):



Регистры. Регистр представляет собой цифровую электронную схему (функциональный узел), служащую для временного хранения двоичных чисел, состоящую из триггеров и защёлки.

По существу, это наборы триггеров с независимыми информационными входами и (обычно) общим тактовым входом. В качестве регистров подобного рода могут быть использованы без дополнительных элементов многие типы синхронных триггеров.