

Лекция. Техническое обеспечение проектирования. Вычислители.

Техническое обеспечение (ТО) САПР включает в себя различные технические средства (hardware), используемые для выполнения автоматизированного проектирования, а именно **вычислительные системы, ЭВМ (компьютеры), периферийные устройства, сетевое оборудование**, а также оборудование некоторых вспомогательных систем (например, измерительных), поддерживающих проектирование.

Используемые в САПР технические средства должны обеспечивать:

- выполнение всех необходимых проектных процедур, для которых имеется соответствующее ПО (при наличии ЭВМ и их систем)
- взаимодействие между проектировщиками и ЭВМ (предполагает наличие интерактивного режима работы)
- взаимодействие между членами коллектива, выполняющими работу над общим проектом (объединение САПР в вычислительную сеть).



В результате общая **структура ТО САПР** представляет собой **сеть узлов**, связанных между собой **средой передачи данных**.

Узлами (станциями данных) являются **компьютеры**, контроллеры встроенного оборудования, рабочие места проектировщиков (АРМ) или рабочие станции (WS — Workstation), ими могут быть также большие ЭВМ (мейнфреймы), отдельные периферийные и измерительные устройства и аппаратура сопряжения со средой передачи данных - сетевые контроллеры.

Среда передачи данных представлена **каналами** передачи данных, состоящими из **линий связи** и коммутационного оборудования.
Рис. 1. Среда передачи данных

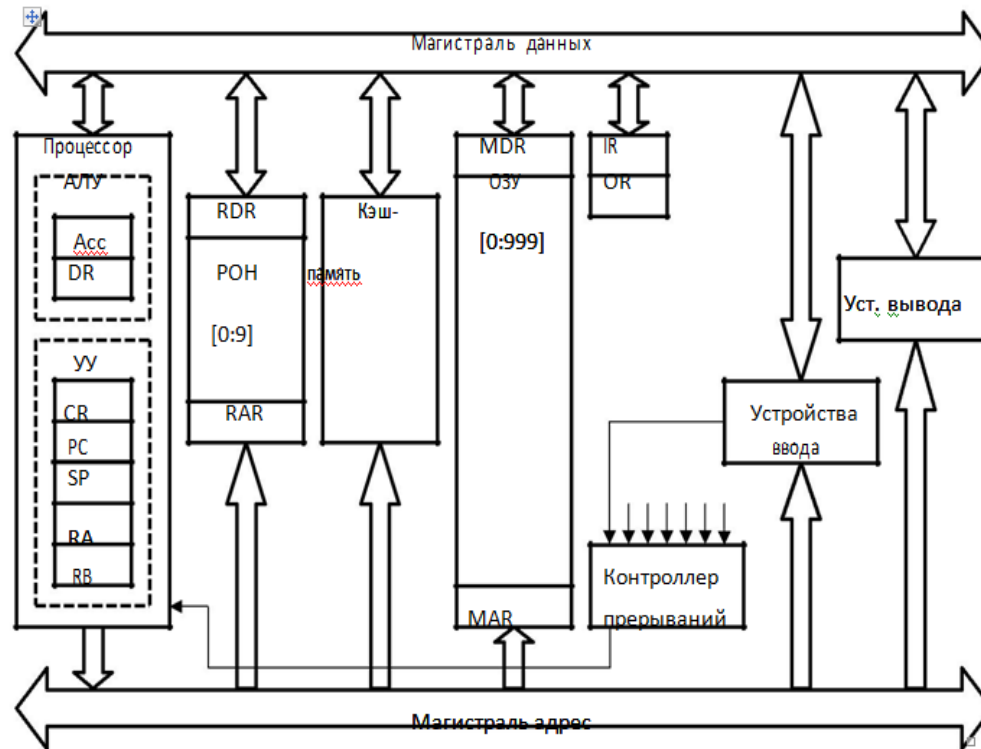
Компьютеры. Архитектура.

Структура ЭВМ включает в себя процессор, оперативную память, кэш-память, регистры общего назначения (РОН), память данных и память команд, контроллер прерываний, регистры ввода-вывода, устройства ввода-вывода, шины. *фон Неймана распределил функции между различными устройствами так, что каждое устройство отвечает за выполнение только своих функций, что существенно упростило схему машины.*

Процессор состоит из **АЛУ** и **устройства управления**, включающего **регистры** — специальные ячейки памяти, расположенные непосредственно в процессоре. Работа с регистрами выполняется намного быстрее, чем с ячейками оперативной памяти, поэтому регистры активно используются как в программах на языке ассемблера, так и компиляторами языков высокого уровня. Названия регистров происходят от их назначения: **аккумулятор (Acc)**, **регистр данных (DR)**, **счётчик команд (PC)**, **командный регистр (CR)**, **регистр адреса (RA)**, **регистр базового адреса (RB)**, **указатель стека (SP)**.

АЛУ осуществляет выполнение одной из допустимых операций, определяемой кодом операции (**COP**), над содержимым **аккумулятора (Acc)** и **регистра данных — DR** (в простейшем МК). Результат операции всегда помещается в **Acc**. При завершении операции АЛУ вырабатывает сигналы признаков результата: - Z (равен 1, если результат равен 0 в Acc); - S (равен 1, если результат в асс отрицателен); - OV (равен 1, если при выполнении операции в асс произошло переполнение разрядной сетки). **Любое действие (команда) делится на совокупность элементарных действий (микроопераций)**. Набор микроопераций хранятся в специальном управляющем запоминающем устройстве и определяется микропрограммой. Формальные шаги выполнения одной команды в машине фон Неймана: **CR** (регистр команд):=**RA**>; - считать из памяти очередную команду на регистр команд;

выполнить; **RA:=RA+1**; увеличить счётчик адреса на единицу и выполнить очередную команду и т.д. Последовательность команд (из системы команд) задается исполнителем и записывается в память.



Устройство управления (УУ) осуществляет:

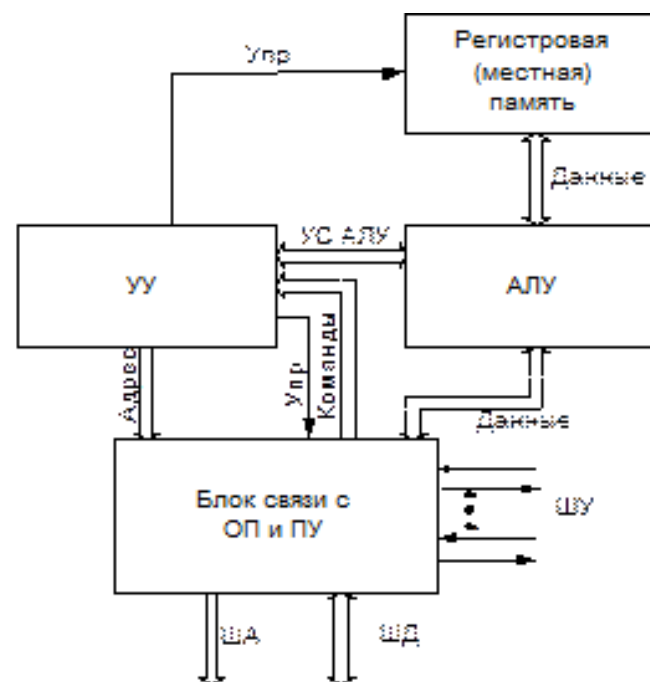
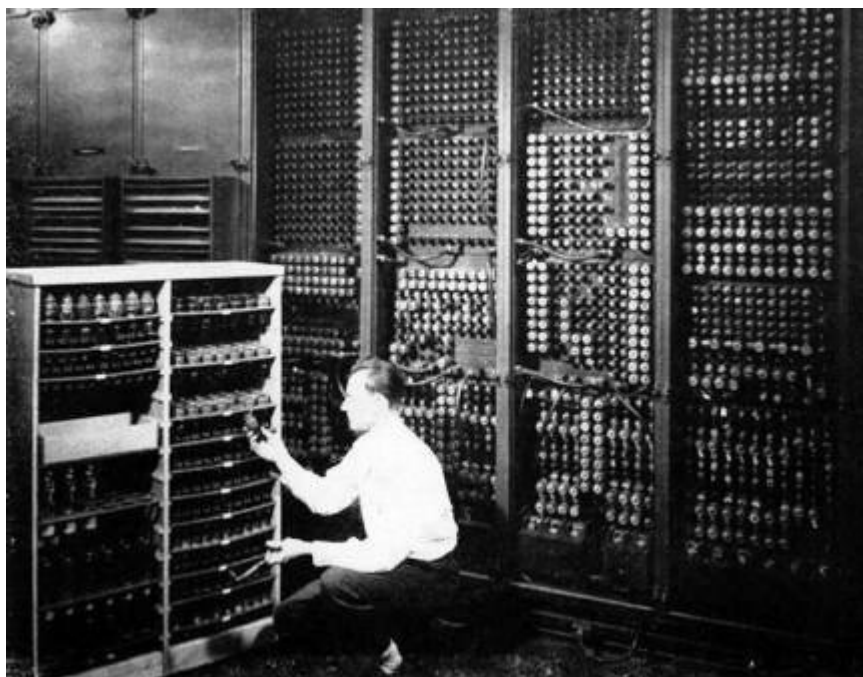
- **выборку команд из памяти** в нужном порядке
- **выборку из памяти данных**, задаваемых адресами команды;
- **выполнение микроопераций**, предписанных командой;
- **останов или переход** к выполнению следующей команды.

В состав УУ входят следующие регистры:

- CR(command reg.)** — регистр, содержащий код команды;
- PC(program counter)** — регистр (счётчик) адреса текущей команды;
- SP(stack pointer)** — указатель стека, содержащий адрес верхушки стека;
- RA** — **регистр (счетчик) адреса**, при выполнении текущей команды в него записывается адрес *следующей* команды.
- RB** — **регистр базового адреса**;
- I(Interrupt, InterruptEnable)** - флаг разрешения прерываний

ENIAC – первый компьютер, 1946 год. Вес – 30 т, занимал целое помещение в 85 кубических метров пространства. Большое тепловыделение, энергопотребление, постоянные неполадки из-за разьёмов электронных ламп. Окислы приводили к исчезновению контактов и лампы теряли связь с платой, что требовало постоянного обслуживания.

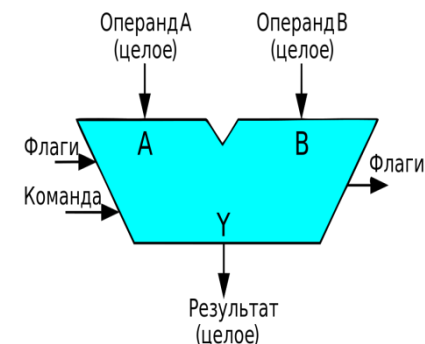
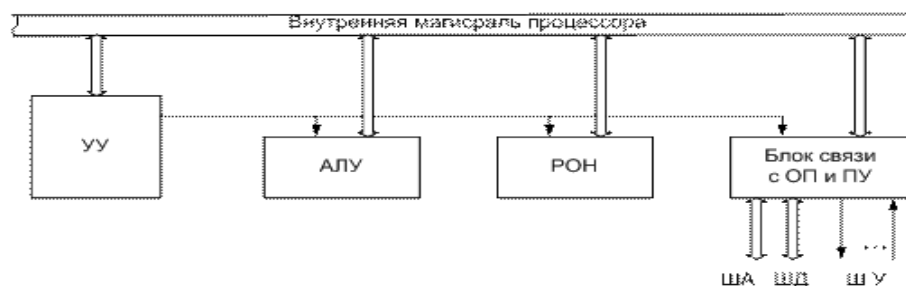
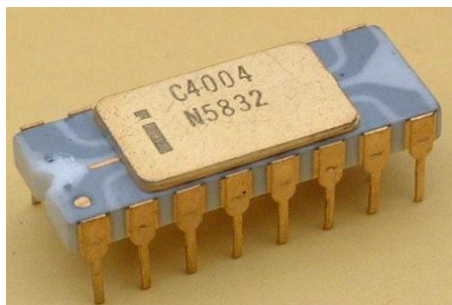
Процессоры с непосредственными связями имеют потенциально высокое быстродействие, поскольку пропускная способность множества шин велика. Однако большое число шин ограничивают возможность интеграции аппаратуры процессора в кристалл, поэтому они находят применение только в больших ЭВМ.



Параллельно развивались полупроводниковые технологии - в 1906 работы по детекторам, в 30-х - 40-х годов появились первые германиевые диоды, в 1947г. – изобретены транзисторы. В 1959 году **Роберт Нойс** изобрел интегральную микросхему (*ИМС*).

Микропроцессоры. Фирма **Intel** основана Робертом **Нойсом**, Гордоном **Муром** и Эндрю **Гроувом** в 1968 году для реализации возможностей интеграции электронных компонентов. Сначала производила п/п запоминающие устройства. Первым была МС «3101» - 64 разряда, биполярная статическая ОЗУ. Следующим было изобретение «4004» (1971) - **микропроцессор** с 2300 п/п транзисторов в своём составе, по производительности не хуже, чем ENIAC, а размером меньше фаланги пальца по сравнению с 85 м³ последнего. Изначально проект (заказ Busicom) калькулятора предусматривал не менее 12 чипов разной архитектуры. Инженеры Intel предложили создать универсальное устройство, которое использует для работы **команды из ОЗУ** (реально состоит из 4 чипов: **ПЗУ, контроллер ввода-вывода, ОЗУ и процессор 4004**).

Отличием от большого компьютера является единственная шина (сеть) внутри кристалла, которая связывает все узлы, посредством которой эти узлы «общаются».



4004 – 4-разрядная, р-МОП микросхема. **Арифметическое логическое устройство (АЛУ)** - блок процессора, который под управлением устройства управления (УУ) служит для выполнения арифметических и логических преобразований над данными, называемыми в этом случае операндами. Действия процессора выполняются по тактам (частота 100 кГц).

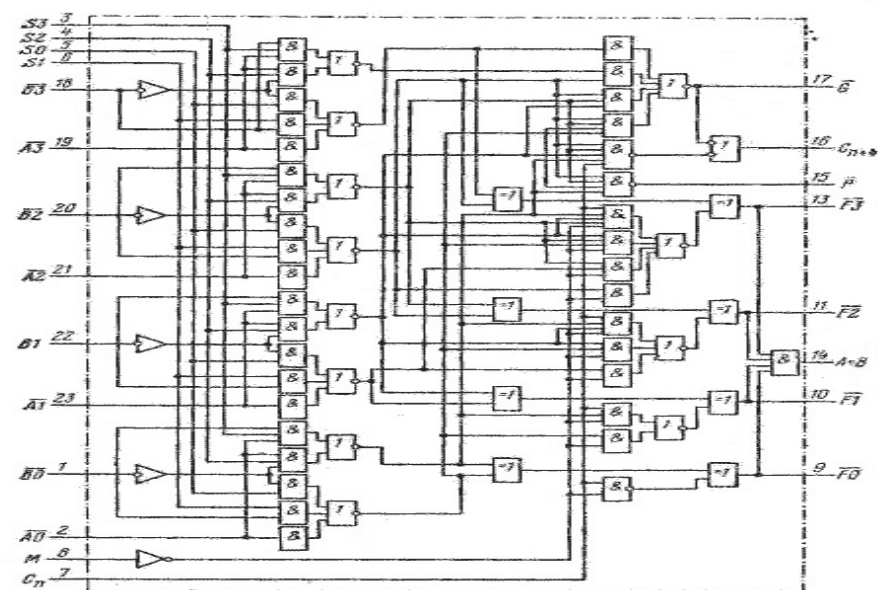
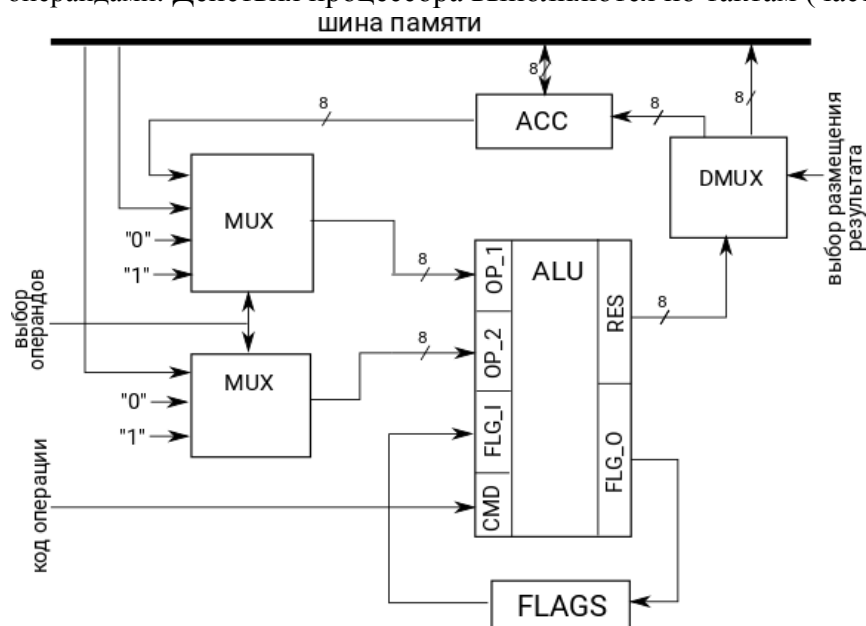


Рис. а) Управление АЛУ б) принципиальная схема 4-х битового АЛУ.

Следующим этапом стал выпуск в 1972 году **процессора «8008»**. В отличие от предыдущей модели он уже больше похож на современные процессоры. 8008 – 8 разрядный, имеет аккумулятор, 6 регистров общего назначения, указатель стека, 8 регистров адреса, команды ввода-вывода.

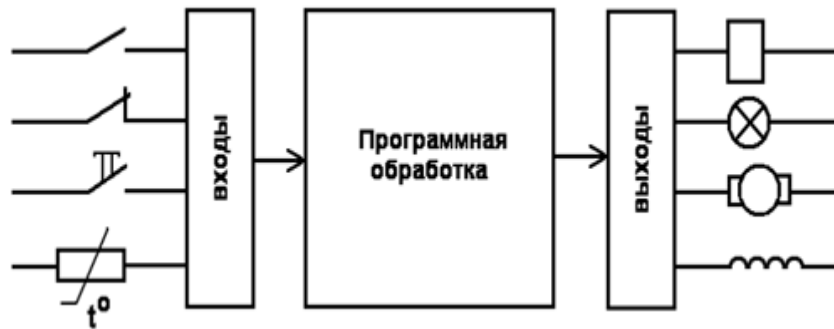
Процессор – это самая главная часть, сердце всей системы. Он предназначен для выполнения различных действий с **числами в памяти** и с **данными в портах ввода/вывода** по определенной программе. Все три части вычислительной системы связаны между собой **шинами данных**, по которым цифровые сигналы передаются от процессора к модулю памяти и портам ввода/вывода и обратно к процессору. **Микропроцессор** — устройство, отвечающее за выполнение **арифметических, логических операций и операций управления**, записанных в машинном коде, реализованное в виде одной микросхемы



Связь при помощи шин.

Микроконтроллер (МК, PLC) это **маленький компьютер, размещенный в одной интегральной микросхеме**. МК представляет собой микропроцессорное устройство, предназначенное для сбора, преобразования, обработки, хранения информации и выработки команд управления, имеет конечное количество входов и выходов, подключенных к ним датчиков, ключей, исполнительных механизмов для объекта управления, и предназначен для работы в режимах реального времени. У него есть АЦП, flash-память, EEPROM-память, множество регистров, порты ввода-вывода, а также таймеры, счетчики, компараторы, АЦП, USARTы (Universal Synchronous/Asynchronous Receiver/Transmitter - периферийное устройства, преобразующее входящие и исходящие байты в последовательный поток данных) и т. п. Микроконтроллер, в отличие от микропроцессора, обычно имеет небольшую разрядность (8 – 16 бит) и богатый **набор команд манипулирования отдельными битами**. Битовые команды дают возможность управлять дискретным оборудованием (поднять/опустить шлагбаум, включить/выключить лампу, нагреватель, запустить/остановить двигатель, открыть/закрыть клапан, и проч.)

Архитектура микроконтроллера устоявшаяся, облегчающая процесс программирования, связи между блоками постоянные (а не перепрошиваемые). При программировании (прошивке МК) изменяется только постоянная память, на которую опирается вся работа МК.

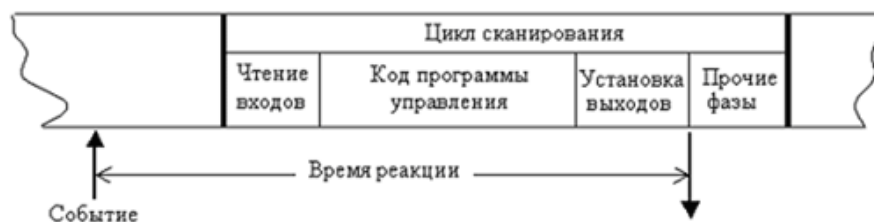


Микроконтроллер загружается после подачи питания и начинает выполнять программу, записанную в его flash-памяти. При этом он может через порты ввода/вывода управлять самыми разнообразными внешними устройствами. МК – микросхема, внутреннее содержимое которой мы создаем сами. Если захочется внести какие-либо изменения в работу электронного устройства, достаточно будет лишь **перепрограммировать МК**. Микроконтроллеры ликвидируют **разрыв между аппаратным и программным обеспечением** – они выполняют программу как обычный компьютер, являясь в тоже время дискретными элементами, которые могут взаимодействовать с другими компонентами схемы.

Программное обеспечение (ПО) контроллеров состоит из двух частей. **Первая часть** - это **системное программное обеспечение**. Проводя аналогию с компьютером можно сказать, что это некая **операционная система**, она управляет работой узлов контроллера, внутренней диагностикой, осуществляет взаимосвязи составляющих частей. Системное программное обеспечение расположено в постоянной памяти центрального процессора и всегда готово к работе. После включения питания МК готов взять на себя управление системой уже через несколько миллисекунд.

Вторая часть – **пользовательское программное обеспечение** (написанная вами программа). МК работают циклически по методу периодического опроса входных данных.

Рабочий цикл МК включает 4 фазы: 1. Опрос входов; 2. Выполнение пользовательской программы; 3. Установку значений выходов; 4. Вспомогательные операции (диагностика, подготовка данных для отладчика, визуализации и т. д.).



Выполнение 1 фазы обеспечивается системным программным обеспечением. После чего управление передается той программе, которую вы сами записали в память, а по ее завершению управление опять передается системному уровню. За счет этого обеспечивается максимальная простота построения прикладной программы – ее создатель не обязан знать, как производится управление аппаратными ресурсами. Необходимо знать, с какого входа приходит сигнал и как на него надо реагировать на выходах.

Время реакции на событие будет зависеть от времени выполнения одного цикла прикладной пользовательской программы. Определение времени реакции – времени от момента события до момента выдачи соответствующего управляющего сигнала – поясняется на рисунке:

Обладая памятью, МК, в зависимости от предыстории событий, способен реагировать по-разному на текущие события. Возможности перепрограммирования, управления по времени, развитые вычислительные способности, включая цифровую обработку сигналов, поднимают МК на более высокий уровень в отличие от простых комбинационных автоматов.

В 80 году Intel выпускает **Intel 8051** — первый однокристальный **микроконтроллер (МК)** гарвардской архитектуры, который был произведен Intel для использования во встраиваемых системах (торговый автомат, светофор, говорящий плюшевый медведь, автомобильные и авиационные системы, аудио и TV техника). **К 1990 г. выпущено 100 млн. таких МК.** Для своего времени он имел немислимые 128 000 транзисторов. Для программирования использовались языки высокого уровня: Форт, Бейсик, Паскаль, PL/M, Modula-2, однако они не получили такого широкого распространения, как **C** и **ассемблер**.

Различия микроконтроллеров. Микроконтроллеры можно разделить по таким критериям:

Разрядность - это длина одного слова, обрабатываемого контроллером или процессором за такт, чем она больше, тем быстрее микроконтроллер может обработать большие массивы данных.

Система команд. **RISC**-архитектура, или сокращенная система команд. Ориентирована на быстрое выполнение базовых команд за 1, реже 2 машинных цикла, а также имеет большое количество универсальных регистров, и более длинный способ доступа к постоянной памяти. **CISC**-

архитектура, или полная система команд, характерна прямая работа с памятью, большее число команд, малое число регистров (ориентирована на работу с памятью), длительность команд от 1 до 4 машинных циклов. Пример – процессоры Intel, AMD.

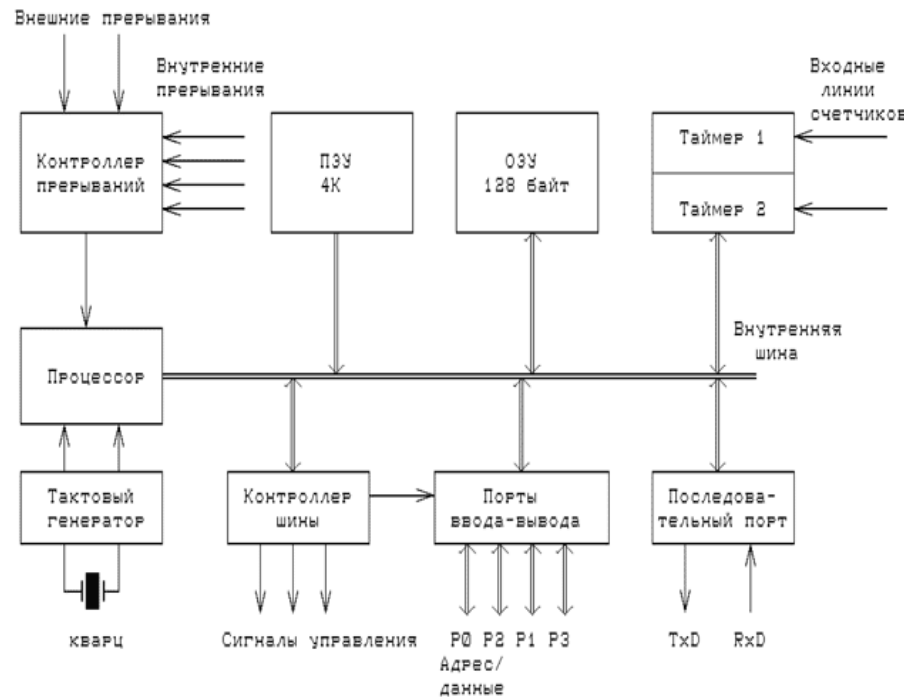


Рис. Обобщенная структурная схема микроконтроллера Intel 8051

Например, $+365_{(10)} = +101101101000000_{(2)}$. Процессоры с фиксированной запятой (точкой) стоят во всех телефонах.

2. Процессоры с плавающей запятой (ПЗ) имеют операнды, разделенные на мантиссу (дробь) и экспоненту (степень), под которые отводятся конкретные биты в 32, 64-разрядном операнде. Основание известно, но не записывается. Например, в десятичном варианте число $1.528535047E-25$ имеет мантиссу = 1.528535047, а экспоненту = - 25. В двоичном варианте в 32-разрядном числе первые два разряда – это знаки, 2-8 разряды – запись экспоненты в виде суммы степеней двойки, с 9 по 31 разряд – запись двоичной мантиссы в виде суммы обратных степеней двойки.

Процессоры с плавающей запятой значительно более сложные и наиболее дорогие. Выбор архитектуры, разрядности, системы команд, структуры памяти влияет на конечную стоимость устройства, при единичном производстве разница в цене может быть незначительна, но при тиражировании – более чем ощутимой.

Архитектура памяти. Архитектура фон-Неймана разработана в 1945 году. Основная черта - общая область памяти для команд и данных, пересылка данных и выборка команды не может осуществляться одновременно. **Гарвардская архитектура** – раздельная память данных и память программ.

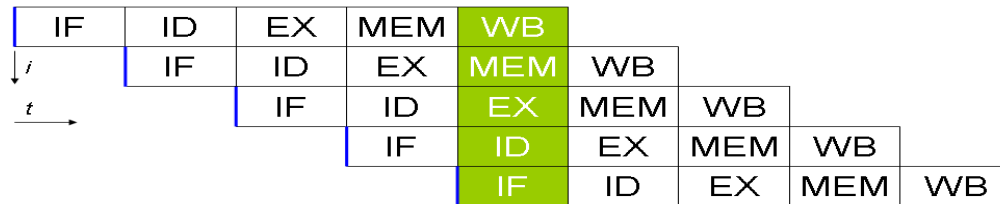
Способ представления данных. Число с фиксированной точкой или запятой (ФТ, ФЗ), как и число с плавающей точкой (ПТ, ПЗ) является приблизительным представлением вещественного числа. Например, одна десятая (0.1) или сотая (0.01) может быть представлена только приблизительно двоичным числом ФТ или ПТ (точно в десятичной ФТ, ФЗ). В десятичной системе представления чисел с плавающей запятой (3 разряда) операцию умножения $0,12 \times 0,12 = 0,0144$ в нормальной форме представляется в виде $(1,20 \cdot 10^{-1}) \times (1,20 \cdot 10^{-1}) = (1,44 \cdot 10^{-2})$. В формате с фиксированной запятой получим вынужденное округление $0,120 \times 0,120 = 0,014$. (потеряли правый разряд числа).

По этому показателю все процессоры делятся на:

1. Процессоры с фиксированной запятой или точкой (ФЗ, ФТ).

Операнд в 16-разрядном процессоре имеет целочисленный вид, а о наличии точки знает только программист. Целые числа без знака в двухбайтовом формате (2 байта = 16 бит) могут принимать значения со знаком - от -2^{15} до $+2^{15-1}$, т.е. от -32768 до 32767.

Конвейеры. Упрощенно, это несколько параллельных действий за один такт. За счет усложнения архитектуры позволяют поднять производительность. Например, конвейер позволяет прочитать инструкцию, исполнить предыдущую и записать в шину данных одновременно.



IF — получение инструкции,
ID — расшифровка инструкции,
EX — выполнение,
MEM — доступ к памяти,
WB — запись в регистр.

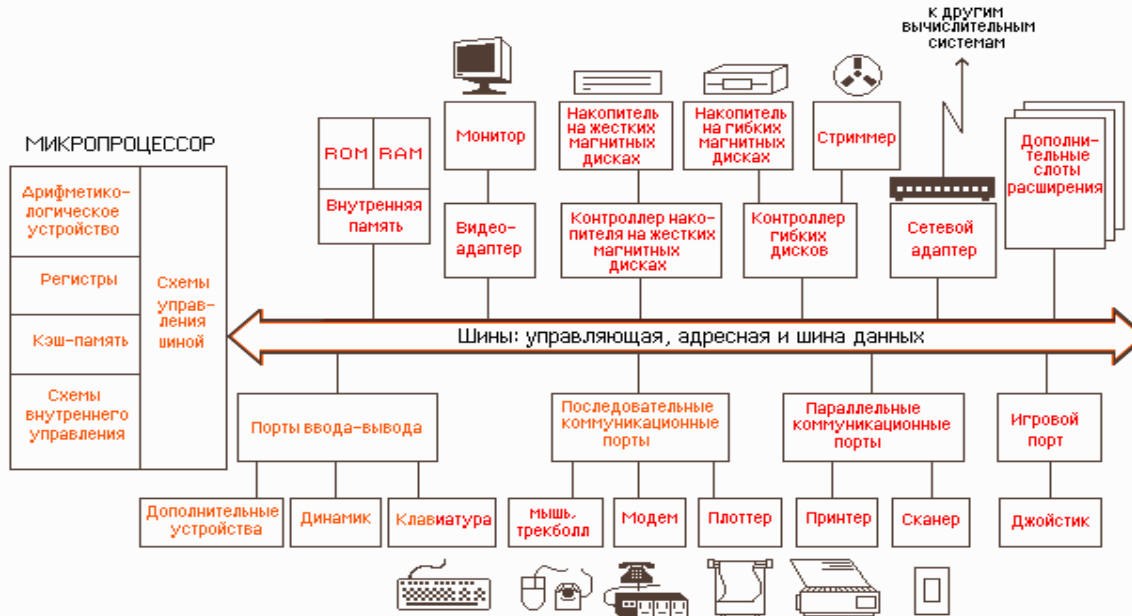
The Network Is The Computer. Sun Microsystems

В 1981 году фирма IBM выпустила на рынок свой **персональный компьютер** - IBM PC, положив начало новому витку компьютерной революции. В марте того же года появилась еще одна, не менее важная новинка. Фирма **3Com** представила миру первую **Ethernet-карту**. Этим двум технологиям было суждено изменить мир. Обе имели открытые спецификации, что позволило независимым производителям немедленно начать производство клонов. И в основе обеих технологий лежала **шина**.

Термин «шина» означает, что все разъемы соединены параллельно. В процессоре это питание, тактовые синхронизирующие импульсы, сигналы запроса прерываний, адреса и данные. Все устройства, подключенные к шине, не могут передавать данные одновременно, иначе те перепутаются между собой. Поэтому информацию приходится передавать поочередно по вызову процессора, который тоже включен в шину. Каждый компонент PC, каждый регистр ввода/вывода и ячейка RAM имеют свой адрес и входят в общее адресное пространство PC. Для адресации к какому-либо устройству PC служит шина адреса, по которой передается уникальный идентификационный код. **Системная шина** осуществляет обмен информацией между всеми устройствами компьютера и включает в себя **шины адреса, данных и управления**. Пропускная способность шины

определяется умножением тактовой частоты на ее разрядность.

Параллельная шина означает, что все 8 бит (или 1 байт) пересылаются одновременно каждый по своему проводу (например, кабель принтера - он значительно толще, чем последовательный кабель мыши). Чем выше разрядность шины (количество проводников, входящих в нее), тем больше данных может быть передано за определенный промежуток времени и выше производительность шины. Компьютеры с процессором 80286 имели 16-разрядную шину данных, с CPU 80386 и 80486 — 32-разрядную, а компьютеры с CPU семейства Pentium имеют уже 64-разрядную шину данных.



Скорости работы с памятью: За такт передаётся 64 бита, то есть 8 байт. Соответственно, если рабочая частота шины равна 100 МГц (100'000'000 тактов в секунду) — то скорость передачи данных будет равна 8 байт * 100'000'000 герц \approx 800 мегабайт в секунду. **DDR-шина** способна передавать за один такт удвоенный объём данных, а **Quad Data Rate** шина — то она и вовсе «400-мегагерцевая», так как за один такт передаёт четыре пакета данных. Хотя реальная частота работы у всех трёх вышеописанных шин одинаковая — 100 мегагерц.

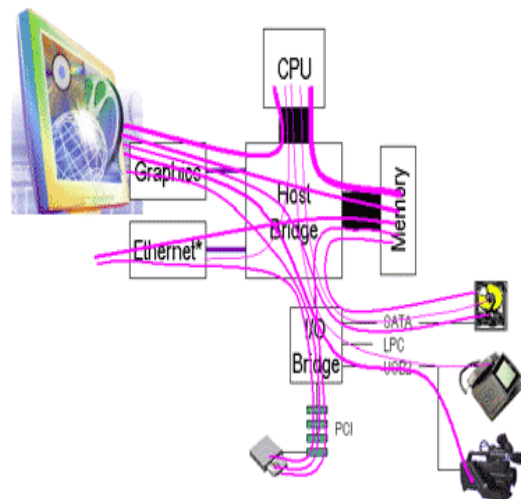
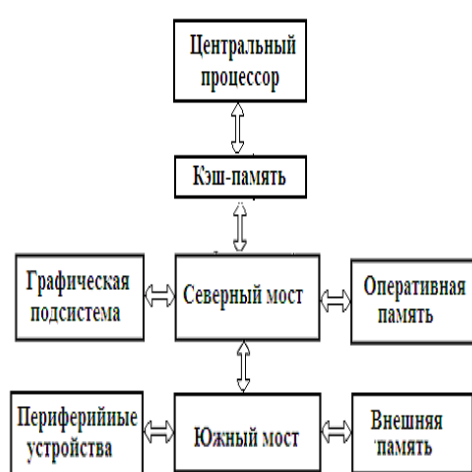
В сети Ethernet нет ни процессора, ни прерываний, но там тоже есть **шина**. В первой версии Ethernet это был коаксиальный кабель, к которому адаптеры сети подключались специальными разъемами (DB-15, его еще можно увидеть на старых сетевых картах). Поскольку все адаптеры подключены к шине параллельно, в любой момент времени может работать на передачу только один из них, иначе они перепутаются и будут потеряны. Это явление называется **коллизией** (столкновением). Никаких особых способов для предотвращения таких ситуаций не существует, и, если в сети передается много информации разными машинами, коллизии происходят то и дело. Раздельный способ использования общей шины так и называется «**множественный доступ с контролем несущей и разрешением коллизий**». Хаб (мост) – средство борьбы с коллизиями.

Шинная архитектура неэффективна и медленна, но эти недостатки окупаются ее простотой и дешевизной. По мере роста требований к скорости передачи начались поиски путей увеличения пропускной способности шины.

Чтобы повысить пропускную способность шины, нужно увеличить ее разрядность. Так 8-битная шина ISA, которой были оборудованы первые IBM PC, получила дополнительную секцию и стала 16-битной.

Другой способ - **уменьшение количества устройств в домене коллизий**. Если разделить сеть на несколько подсетей, в каждой из них станет свободнее - конкурирующих за доступ к шине устройств станет меньше. Но эти небольшие куски сети придется объединить специальными устройствами - **мостами**. Ethernet-мост пропускает информационный пакет из одной сети в другую, основываясь на адресе назначения пакета (MAC адрес). Третий способ повышения пропускной способности шины - **увеличение ее тактовой частоты**. Повышение частоты дается с трудом. Если частота превышена, переходные процессы при переключениях транзисторов не успевают завершиться за один такт, и это вызывает сбой, 1 и 0 перестают четко различаться по уровню. Также возрастают потери (разогрев) за счет конечного времени переключения транзистора.

То же ограничение есть и в **Ethernet-сетях**. Даже при наличии усилителей и повторителей **домен коллизий** не должен превышать 2,5 км, а для 100 Мбит/с Fast Ethernet - всего 210 м. Иначе не удастся вовремя распознать коллизию и сетевые карты не смогут согласовать очередность передачи. Поэтому чем быстрее шина, тем меньше должен быть ее размер.

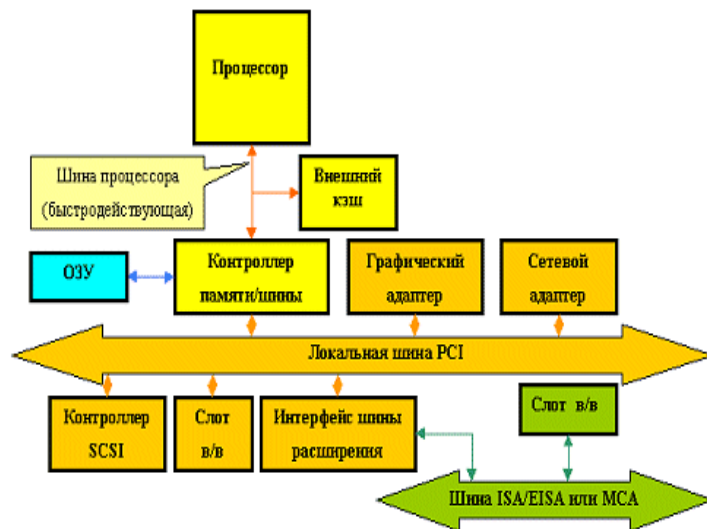


В архитектуре вычислителя тоже появились **две сети**, соединенные мостом: высокоскоростная системная шина, на которой присутствовали всего два устройства - процессор и память, и шина ISA, работающая на низкой частоте. Их по традиции называют **северным** и **южным** мостами. В наиболее быструю сеть - северный мост современных чипсетов обычно включены процессор, память, AGP-видео и южный мост, то есть всего четыре

потребителя и **коммутатор**, который пропускает к каждому устройству лишь те пакеты, которые ему адресуются. Южный мост более медленный, отвечает за шины расширения PCI и ISA и интегрированные в материнскую плату устройства - контроллеры дисков, разного рода внешние порты, сетевой адаптер и пр..

PCI-E - локальная сеть в пределах компьютера. Современные системы должны одновременно обрабатывать несколько конкурирующих потоков данных от разных источников. Данные не одинаковы (рис.) и должны быть «помечены» таким образом, чтобы система ввода-вывода могла назначить им правильные приоритеты обработки. *) Сеть **InfiniBand** для сложных вычислительных задач с большим количеством коммуникаций

В то время как **процессоры** уже не первый год успешно движутся в направлении **параллельных архитектур** (SIMD-расширения, суперскалярность, конвейеризация, Hyper-Treading и многоядерность), **шины передачи данных не менее успешно переходят на последовательные решения.** Причины обеих тенденций схожи и довольно просты - необходимо сбалансированно наращивать производительность всех компонентов компьютеров, однако не всякие существующие архитектурные решения способны эффективно масштабироваться. Множественные соединения «точка-точка» влекут за собой появление в топологии ввода-вывода нового элемента – **коммутатора (switch).** Коммутатор обеспечивает обмен данными между конечными потребителями-источниками. Например, если их трафик не затрагивает оперативную память, то и нет необходимости передавать его через Host Bridge (или Северный мост).



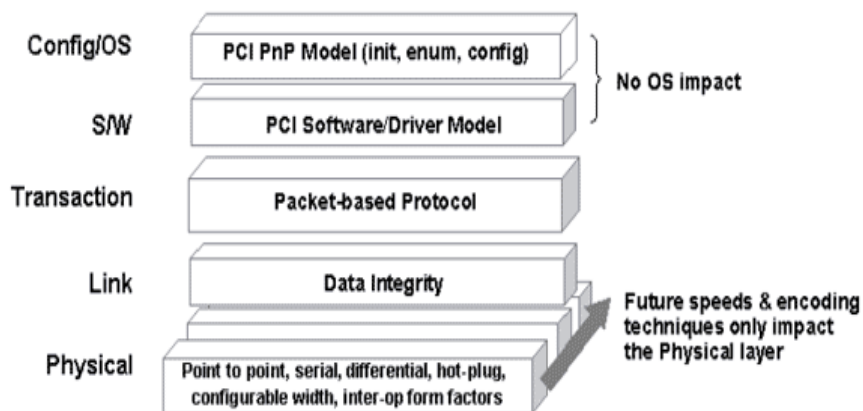
Архитектура компьютера с шиной PCI

Шина PCI (**Peripheral Component Interconnect bus**) - шина для подсоединения периферийных устройств. Стала массово применяться для Pentium-систем. Частота шины от 20 до 33 МГц (66МГц), теоретически максимальная скорость 132/264 Мбайт/с для 64 бит. В современных материнских платах частота на шине PCI задается как 1/2 входной частоты процессора. Может работать параллельно с шиной процессора, не обращаясь к ней за запросами. Например, процессор работает себе с кэшем или системной памятью, а в это время по сети на винчестер пишется информация.

Разработчики шины **PCI Express** не стали изобретать велосипед и взяли за основу наработки в области сетевого оборудования. Получилось что-то очень напоминающее **Gigabit Ethernet** - и на физическом уровне, и на уровне протоколов передачи данных. И самое главное отличие новой шины: PCI Express стала **последовательной**, а значит, четко **разнесены уровни представления данных и уровень их передачи.** Информация, которую необходимо передать, **упаковывается в пакеты**, куда заносятся данные о получателе и коды обнаружения/исправления ошибок. Получившийся сплошной поток (где идут вперемешку данные, приложения и вспомогательная информация) уже передается - абсолютно неважно каким способом - через физическую среду.

Стандарт PCI Express предусматривает **традиционную многоуровневую модель, аналогичную сетевой – ЭМВОС (Рис.).** На самом верхнем уровне располагаются прикладные приложения, их драйверы, использующие PCI-

устройство. Для них в новой схеме не меняется ровным счетом ничего - как старые операционки как работали с PCI, так и будут работать с PCI Express.



Добавлено два новых уровня (**Transaction Layer** и **Link Layer**), которые иначе, как **TCP** и **IP** не назовешь - выполняемые функции абсолютно те же, что и у "сетевых" аналогов. **Transaction Layer** заведует **первоначальной упаковкой данных**, передачей их конкретному получателю и гарантиями корректной доставки сообщения. **Link Layer** указывает физический адрес назначения пакета, по которому контроллеры шины принимают решение о направлении пакета в конкретную физическую линию, здесь же располагается код обнаружения и исправления ошибок в принятом пакете (CRC), номер пакета, позволяющий отличить один пакет от другого, и др.

Внизу этой пирамиды размещается собственно физическая реализация шины передачи данных - это две дифференциальные пары проводников с импедансом 50 Ом (первая пара работает на прием, вторая - на передачу), данные передаются с использованием **избыточного кодирования** по

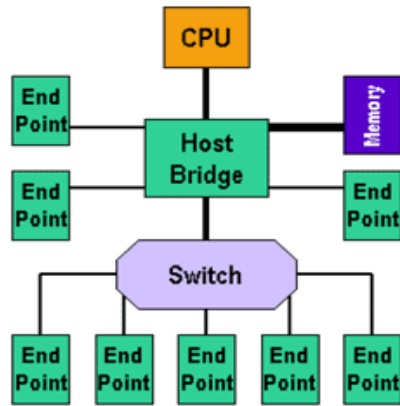
схеме "8/10" с исправлением ошибок. Это позволяет исправлять многие простые ошибки, неизбежные на столь высоких частотах, без лишних повторных передач и привлечения протоколов вышележащих уровней. Логический код 8B/10B заменяет исходные символы длиной в 8 бит на символы длиной в 10 бит. Так, в коде 8B/10B результирующие символы могут содержать 1024 битовых комбинации, в то время как исходные символы - только 256. В результирующем коде можно отобрать 256 комбинаций, которые не содержат большого количества нулей, а остальные считать **запрещенными кодами**. Если запрещенный код принят, при помощи специального итеративного алгоритма (поиск по решетке) находится "наиболее подходящая" группа, исправляя сбойные биты на "правильные". Наглядная аналогия - системы OCR со словами: они вполне в состоянии отловить порожденную при распознавании текста "ошибку" и заменить ее на "ошибку".

Кроме того, применяется **скремблирование**, что бы уменьшить долю "постоянных" составляющих в сигнале (не более четырех нулей или единиц подряд). Это позволяет приемнику уверенно синхронизироваться по фронтам поступающего сигнала, поскольку никакого дополнительного ("внешнего") синхронизирующего сигнала от тактового генератора в PCI Express не используется.

Как и в любой сети, передаваемые данные дополнительно нарезаются **небольшими кусочками - кадрами**. При тактовой частоте шины 2,5 ГГц мы получим скорость 2,5 Гбит/с. С учетом выбранной схемы "8/10" выходит 250 Мбайт/с, однако многоуровневая сетевая иерархия не может не сказаться на скорости работы, и реальная производительность шины оказывается значительно ниже - всего лишь около 200 Мбайт/с в каждую сторону. Впрочем, даже это на 50% больше, чем теоретическая пропускная способность шины PCI.

Но это далеко не предел: PCI Express позволяет объединять в шину нескольких независимых линий передачи данных. Стандартом **предусмотрено использование 1, 2, 4, 8, 16 и 32 пар проводников** - передаваемые данные поровну распределяются между ними по схеме "первый байт на первую линию, второй - на вторую, ..., n-й байт на n-ю линию, n+1-й снова на первую, n+2 снова на вторую" и так далее. Это не параллельная передача данных и даже не увеличение разрядности шины (поскольку все передающиеся по линиям данные передаются абсолютно независимо и асинхронно) - это **именно объединение** нескольких независимых линий. Причем передача по нескольким линиям никак не влияет на

работу остальных слоев "пирамиды" и реализуется на "нижнем", физическом уровне. Именно этим достигается прекрасная масштабируемость PCI - Express, позволяющая организовывать шины с максимальной пропускной способностью до $32 \times 200 = 6,4$ Гбайт/с в одном направлении, под стать лучшим параллельным шинам сегодняшнего дня.



PCI Express относится к шинам класса "точка-точка", то есть одна шина может соединять только два устройства (в отличие от PCI, где на общую шину "вешались" все PCI-слоты компьютера), поэтому для организации подключения более чем одного устройства в топологию PCI Express, **включаются хабы и свитчи (коммутаторы)**, распределяющие сигнал по нескольким маршрутам. Это тоже одно из главных отличий PCI Express от прежних параллельных шин.

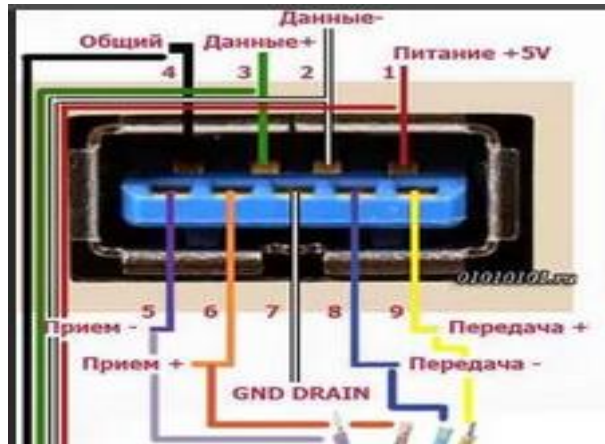
По мере развития электроники можно нарастить тактовую частоту шины (до 10 ГГц, что уже вплотную приближается к теоретическому пределу передачи сигнала по медным соединениям, оцениваемому в 15–20 ГГц, без ущерба для совместимости с приложениями и драйверами. В плане практической реализации шина PCI-Express представляет собой целый аппаратный комплекс, затрагивающий северный и южный мосты чипсета, коммутатор и оконечные устройства. PCI Express предполагается использовать для связи всех ключевых компонентов системы, кроме "внешних" (USB) и накопителей на магнитных дисках (SATA как более подходящий для работы именно с HDD).

Третья сеть компьютера - USB «универсальная последовательная шина» (Universal Serial Bus) является промышленным стандартом расширения архитектуры PC, ориентированным на интеграцию с телефонией и устройствами бытовой электроники. Его разработка поддерживалась такими гигантами, как **Microsoft, Intel и HP**, которые хорошо понимали необходимость создания нового универсального интерфейса, способного заменить собой существующее на тот момент разнообразие внешних интерфейсов (параллельный порт, последовательный, порт для подключения джойстика, внешний SCSI — и в итоге они действительно исчезли с материнских плат.



Во главе интерфейса USB стоит **хост - контроллер** — аппаратный компонент, действующий под управлением команд - драйверов. К нему подключаются «**хабы**» (пассивные, либо активные разветвители), и конечные устройства, напрямую или через хабы. Хабы осуществляют маршрутизацию, посылая пакеты только в те порты, которые необходимы для достижения конкретного устройства. Общее количество устройств в данном дереве может достигать до 128. Хост с определенной периодичностью по очереди опрашивает оконечные устройства, выделяя им временные промежутки, в течение которых те могут передавать данные. Существуют несколько типов логических каналов связи (мышкам и клавиатурам — канал прерываний, накопителям — изохронный, канал управляющих и поточных данных). Равноправия в USB нет: какие бы устройства друг с другом ни соединялись, одно из них должно играть роль хоста, в то время как остальные —

На физическом уровне (USB-2,0) четыре провода: два для передачи данных, «земля» и +5 В (900мА) для цепей питания. Скорость - *LowSpeed* (скорость до 1,5 Мбит/с), *FullSpeed* (скорость до 12 Мбит/с), USB 2.0 - *HiSpeed* (скорость до 480 Мбит/с).



Стандарт USB 3.0 позволяет одновременно передавать и получать данные. Максимальная длина не сможет превысить 3 метра. USB использует также линейное избыточное кодирование 8/10 бит, превращающее восьмибитные данные в десятибитные, хотя это уменьшает скорость передачи полезной информации на 20%. Помимо роста скорости в новом стандарте увеличилась сила тока, которую может запросить устройство — теперь верхняя граница 0.9 А, что позволило

[illegible]

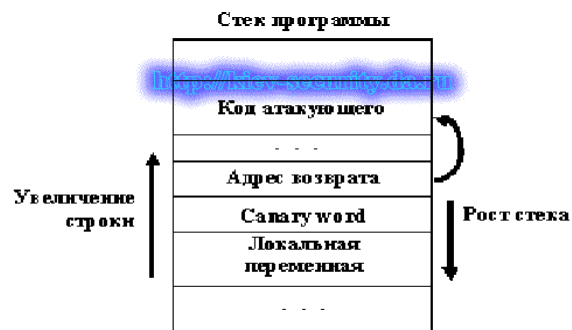
Производительность. FLOPS — число операций двойной точности, выполняемых за одну секунду. Под **производительностью процессора** принято понимать скорость выполнения им той или иной задачи на наборе приложений. Тест производительности, программа тестирования - бенчмарк (*Benchmark*).

На производительность процессора оказывают непосредственное влияние его **микроархитектура, размер кэша, тактовая частота и количество ядер процессора**. Помимо одноядерных в настоящее время существует широкий спектр двухъядерных и четырех и т.д. процессоров для ПК. Переход от одноядерных процессоров к многоядерным — это современная тенденция развития процессоров. При этом не просто меняется архитектура процессоров, но и требуется изменение всей инфраструктуры, включая программное обеспечение. Многоядерные процессоры могут дать выигрыш по производительности, только если используется **оптимизированное под многоядерность, хорошо распараллеливаемое программное обеспечение**.

№	Процессор		Быстродействие:	Поле 1:	Поле 2:	Поле 3:
			Общий рейтинг	Частота	Авторазгон	Ядер / потоков
1	AMD Ryzen Threadripper 3970X		35936.71	3700 MHz	4500 MHz	32 / 64
2	AMD Ryzen Threadripper 3960X		31258.71	3800 MHz	4500 MHz	24 / 48
3	AMD EPYC 7742		26198.68	2250 MHz	3400 MHz	64 / 128
4	AMD EPYC 7702P		24762.43	2000 MHz	3350 MHz	64 / 128
5	AMD EPYC 7452		23413.76	2350 MHz	3350 MHz	32 / 64
6	Intel Xeon W-3275M		21821.03	2500 MHz	4400 MHz	28 / 56

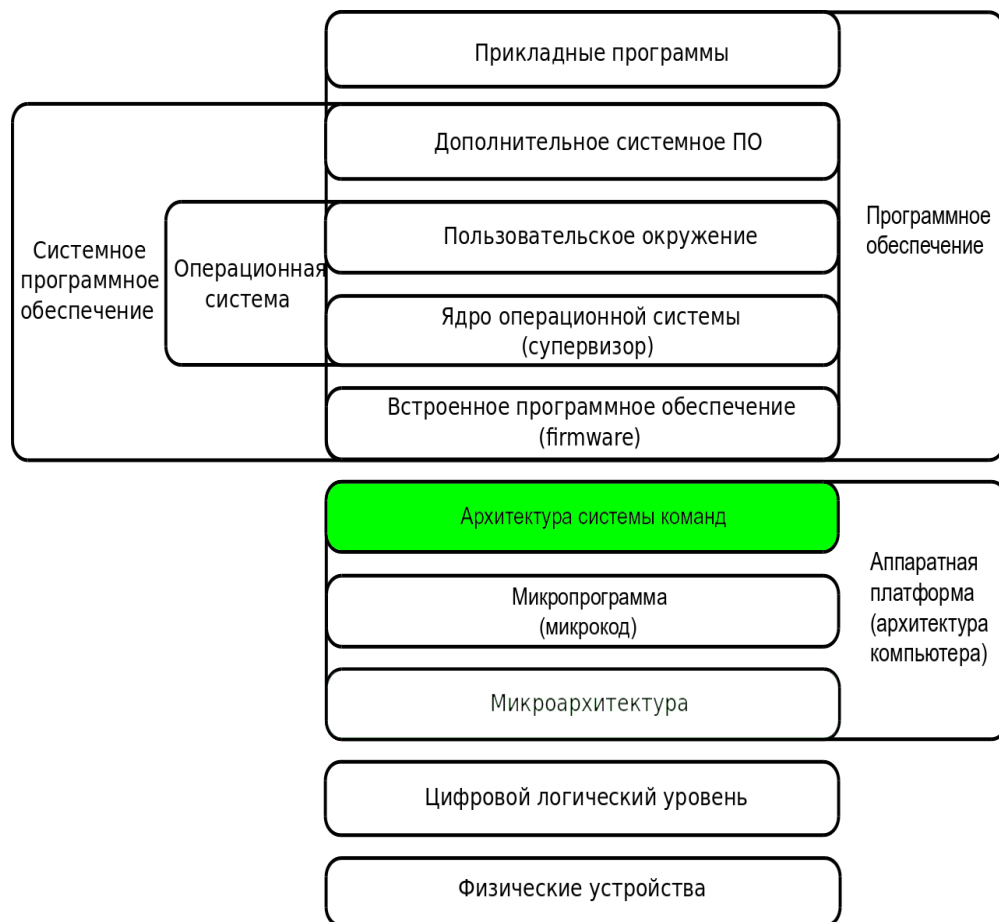
Энергоэффективность, то есть производительностью в расчете на ватт потребляемой им электроэнергии. При достижении потребляемой процессором мощности рубежа в 100 Вт, а тем более его превышении энергоэффективность стала одной из важнейших характеристик процессора. Дело в сложности охлаждения процессора и одна из причин состоит в том, что энергопотребление процессоров сегодня достигло уже той критической отметки, когда дальнейшее увеличение тактовой частоты невозможно, поскольку подобные одноядерные процессоры просто нечем охлаждать. Энергоэффективность зависит от таких характеристик, как микроархитектура процессора, технологический процесс производства, тактовая частота, потребляемая мощность и поддержка процессором функций энергосбережения.

Функциональные возможности. Нынешние процессоры Intel и AMD поддерживают следующие технологии: виртуализации, защиты от вирусов, 64-разрядных вычислений, защиты от перегрева, энергосбережения. В компьютерных технологиях "виртуализация" понимается как абстракция вычислительных ресурсов - пользователь работает с удобным для себя представлением объекта, и для него не имеет значения, как объект устроен в действительности. Различают: - виртуализация платформ, продуктом этого вида виртуализации являются виртуальные машины — некие программные абстракции, запускаемые на платформе реальных аппаратно-программных систем; - виртуализация ресурсов - упрощение представления аппаратных ресурсов для пользователя и получение неких пользовательских абстракций оборудования. Например, виртуальная память делит физическую память на блоки и распределяет их между различными задачами.



Процессорная технология защиты от вирусов реализована во всех новых процессорах Intel и AMD. Суть этой технологии заключается в перехватывании зловредного кода и сведении к минимуму опасности умышленного переполнения буфера (Buffer Overflow).

Многоуровневая компьютерная организация. Как юзеру разобраться в процессах, существующих в компьютере?



Объекты на уровне 0 называются **вентильями**. Вентили состоят из аналоговых компонентов (транзисторы) и смоделированы как цифровые средства, на входе у которых цифровые сигналы (или набор 0 и 1), а на выходе – результат простых функций («И», «ИЛИ», «НЕ»). Каждый вентиль формируется из нескольких транзисторов. Несколько вентилях формируют триггер, который запоминает 0 или 1 - 1 бит памяти. Биты памяти, объединенные в группы, например, по 16, 32 или 64, формируют **регистры**. Каждый регистр может содержать одно двоичное число до определенного предела.

Уровень 1. Микроархитектурный уровень, представляет собой локальную память (РОН) и схему, называемую **АЛУ** (арифметико-логическое устройство). АЛУ выполняет простые арифметические операции. Регистры вместе с АЛУ формируют **тракт данных**, по которому поступают данные.

Основная операция тракта данных состоит в следующем. Выбирается один или два регистра, АЛУ производит над ними какую-либо операцию, например сложения, а результат помещается в один из этих регистров.

Уровень 2. Уровень архитектуры команд (УУ управляет АЛУ) - набор машинных команд, которые выполняются микропрограммой-интерпретатором или аппаратным обеспечением.

Уровень 3. Уровень операционной системы, управляет всеми устройствами (набор команд уровня 2 + новые команды операционной системы (управления), собственная организация памяти, способность выполнять две и более программ одновременно и др.

Нижние три уровня предназначены для работы интерпретаторов и трансляторов, поддерживающих более высокие уровни. Составляются **системными программистами**, которые специализируются на разработке и построении новых виртуальных машин.

Различие между уровнями 1, 2, 3 и уровнями 4, 5 и выше — особенность языка. Машинные языки уровней 1, 2 и 3 — цифровые. Они состоят из длинных рядов цифр, которые удобны для компьютеров, но совершенно неудобны для людей. Начиная с четвертого уровня, языки содержат слова и сокращения, понятные человеку и предназначены для **прикладных программистов**, решающих конкретные задачи.

Уровень 4. Уровень языка **ассемблера**. Представляет собой символическую форму одного из языков более низкого уровня. На этом уровне можно писать программы в приемлемой для человека форме. Эти программы сначала транслируются на язык уровня 1, 2 или 3, а затем интерпретируются соответствующей виртуальной или фактически существующей машиной. Программа, которая выполняет трансляцию, называется ассемблером.

Уровень 5. Язык высокого уровня. Обычно состоит из языков, разработанных для прикладных программистов. Такие языки называются языками высокого уровня. Существуют сотни языков высокого уровня. Наиболее известные среди них — BASIC, C, C++, Java, LISP, Python и Prolog. Программы, написанные на этих языках, обычно транслируются на уровень 3 или 4. Трансляторы, которые обрабатывают эти программы, называются **компиляторами** или иногда используется метод **интерпретации**. Например, программы на языке Java обычно интерпретируются.

Микроконтроллеры. В последние десятилетия наиболее распространены:

8-битные микроконтроллеры PIC фирмы Microchip и AVR фирмы Atmel;

16-битные MSP430 фирмы Texas Instrument (TI);

32-битные микроконтроллеры архитектуры ARM, на базе которой выпускается масса различных продуктов сторонними разработчиками.

Архитектура ARM занимает лидирующие позиции и охватывает 75% рынка 32-64 разрядных встраиваемых RISC-микропроцессоров. МК с архитектурой ARM производят такие компании, как Analog Devices, Atmel, Xilinx, Altera, Cirrus Logic, Intel, Marvell, NXP, STMicroelectronics, Samsung, LG, MediaTek, MStar, Qualcomm, SonyEricsson, Texas Instruments, nVidia, Freescale, Миландр, HiSilicon и другие.

Мир электроники завоёвывают 32 - 64 разрядные микроконтроллеры на основе **процессоров ARM**. Этот прорыв обусловлен их высокой производительностью, совершенной архитектурой, малым потреблением энергии, низкой стоимостью и развитыми средствами программирования. Микроконтроллерное ядро ARM разработано одноименной английской компанией, организованной в 1990 году. Название ARM происходит от "**Advanced RISC Machines**". Компания специализируется сугубо на разработке микропроцессорных ядер и периферийных блоков, при этом не имеет производственных мощностей и поставляет свои разработки в электронной форме, на основе которой клиенты конструируют свои собственные микроконтроллеры.



Любой **МК ARM** включает **ядро, память и периферийные блоки**. Ядро — часть микроконтроллера, осуществляющая выполнение одного потока команд, например, Cortex-M4. Ядро можно сравнить с процессором в ПК. Оно умеет только выполнять команды и передавать данные другим блокам. Производитель МК не разрабатывает ядро. Ядро покупается у компании ARM Limited. Современные ядра Cortex-A53-A57 объединены на одном чипе с применением технологии ARM big.LITTLE и поддерживают набор команд ARMv8 (“версию архитектуры”).

Главное отличие между различными МК — в периферии. Периферийные блоки осуществляют взаимодействие с «внешним миром» или выполняют специфические функции,

недоступные ядру МК. Периферийные блоки предназначены для решения специфических задач. Периферийные блоки не выполняют инструкции. Они лишь выполняют команды ядра.

Qualcomm - Snapdragon 835 – это интеграция четырех ядер ARM Cortex-A73 (ARMv8-A) и столько же Кryo 280 (видоизмененное ядро Cortex-A73)

Huawei Kirin - 970, 960, 955 обладал набором ядер A72(73)+A53.

Samsung Galaxy A7: 8 ядер Cortex A53 с частотой 1,9 ГГц

[STM32F103RET6](#) – 32-разрядный высокопроизводительный RISC-микроконтроллер с ядром ARM Cortex-M3 компании **STMicroelectronics** работает на частоте 72МГц и имеет встроенную память Flash – 512 Кбайт, SRAM – 64 Кбайт. В микроконтроллер интегрированы три 12-битных АЦП, четыре 16-битных таймера общего назначения, два ШИМ-таймера, а также интерфейсы – I2C, SPI, I2S, SDIO, USART, USB, CAN. **Области применения:** системы управления двигателями, медицинское оборудование, навигационные системы, промышленное оборудование.

К 2015 г. разработано шесть основных семейств: **ARM7™ - ARM11™** и SecurCore™. Совместно с компанией Intel разработаны семейства XScale™ и StrongARM®. В дополнение к ARM-архитектуре интегрируются несколько расширений: - Thumb® - 16-разр. набор инструкций, улучшающий эффективность использования памяти программ; - DSP - набор инструкций для цифровой обработки сигналов; - Jazelle™ - расширение для аппаратного непосредственного исполнения Java-инструкций; - Media - расширение для 2-4-кратного увеличения скорости обработки аудио и видео сигналов.

ARM7	ARM9	ARM9E	ARM10E	ARM11
Ключевые особенности: <ul style="list-style-type: none">• 32-разр. RISC ЦПУ для недорогих и маломощных приложений;• Набор инструкций Thumb для оптимизации размера кода;• быстроедействие до 133 МГц. Области применения: <ul style="list-style-type: none">• Мобильные телефоны (процессор управления радиосвязью)• MP3-плееры• Цифровые фотокамеры• Струйные принтеры• Автомобильное управление	Ключевые особенности: <ul style="list-style-type: none">• 32-разр. RISC ЦПУ средней и высокой производительности;• Расширение архитектуры набором инструкций Thumb; Области применения: <ul style="list-style-type: none">• Устройства на основе ОС;• Смартфоны (прикладной процессор);• Персональные цифровые помощники;• ТВ-приставки;• Струйные и лазерные принтеры.	Ключевые особенности: <ul style="list-style-type: none">• 32-разр. RISC ЦПУ средней и высокой производительности;• Конфигурируемая система памяти;• Наборы инструкций Thumb и ARM ЦСП; Области применения: <ul style="list-style-type: none">• Опциональный сопроцессор с плавающей запятой VFP9-S.• Устройства на основе ОС;• Беспроводные устройства, персональные цифровые помощники и смартфоны;• ТВ-приставки;• Сетевое оборудование;• Струйные и лазерные принтеры;• Запоминающие устройства.	Ключевые особенности: <ul style="list-style-type: none">• 32-разр. RISC ЦПУ с максимальным отношением производительность/тактовая частота;• Наборы инструкций Thumb и ARM ЦСП;• Высокая производительность при малом энергопотреблении;• Опциональный сопроцессор VFP10. Области применения: <ul style="list-style-type: none">• Устройства на основе ОС;• Сетевое оборудование;• Высококачественные лазерные принтеры;• Телевизионное оборудование повышенной четкости (HDTV).	Ключевые особенности: <ul style="list-style-type: none">• 32-разр. RISC ЦПУ с поддержкой всех архитектурных расширений (Thumb, ЦСП, Java и мультимедиа);• Опциональный сопроцессор с плавающей запятой. Области применения: <ul style="list-style-type: none">• Устройства на основе ОС;• Сетевое оборудование;• Беспроводные устройства, персональные цифровые помощники, видеофоны нового поколения;

Устройства цифровой обработки сигналов (ЦОС).

Разработчики могут также опираться на следующие варианты цифровой обработки: **процессор (CPU)** или **графический процессор (GPU)** или **DSP** (двоичный сигнальный процессор), **программируемая пользователем вентильная матрица (FPGA)** и специально разработанная **специализированная интегральная схема (ASIC)**.

DSP (цифровые сигнальные процессоры) хороши, когда необходимо реализовывать сложные алгоритмы для потока данных, следующих на не очень высокой скорости — алгоритмы, которые удобно описывать с помощью **последовательных программ**. К сожалению, их производительности хватает далеко не во всех случаях. Несколько миллиардов умножений в секунду — это совсем не много при скоростях потоков данных порядка 100 Мбт/с и более.

ASIC (*application-specific integrated circuit*), *интегральная схема, специализированная для решения конкретной задачи, выполняющая строго ограниченные функции, характерные только для данного устройства (управление мобильным телефоном, кодирование/декодирование аудио- и видеосигналов (сигнальные процессоры) и пр..*

*ASIC ориентирована на максимальную производительность с возможностью задания заранее установленных производителем настроек. Содержат процессоры, блоки памяти, интеллектуальные блоки (IP). ASIC лучше всего подходит для приложений с большим объемом вычислений. При разработке цифровых ASIC для описания их функциональности используют языки описания аппаратных устройств (HDL), такие как **Verilog** и **VHDL**.*

Графический процессор (графический ускоритель) был впервые представлен в 1980-х годах для разгрузки графических операций с процессором. ГП включают в себя тысячи ядер, разработанных для эффективного выполнения одинаковых математических функций. *Nvidia Tesla V100 содержит 5120 ядер CUDA для операций многократного накопления за один цикл и 640 тензорных ядер для умножения матрицы за один цикл. Многие из самых быстрых суперкомпьютеров в мире включают в себя тысячи как графических процессоров, так и центральных процессоров.*

FPGA состоит из внутренних аппаратных блоков с программируемыми пользователем межсоединениями для настройки операций для конкретного приложения. *В отличие от других упомянутых устройств, соединения между блоками могут быть легко перепрограммированы, изменяя внутреннюю работу аппаратного обеспечения и позволяя FPGA приспосабливаться к изменениям проекта или даже поддерживать новое приложение. Такая гибкость делает FPGA отличным выбором для приложений, таких как цифровое телевидение, бытовая электроника, системы кибербезопасности и беспроводная связь. Описывается языками **VHDL** и **Verilog**.*

Наблюдается сближение всех категорий, поскольку поставщики ищут оптимальный набор функций для новых приложений. SoC FPGA поставляются с жесткими или программными IP-процессорами, графическими процессорами и блоками DSP. Процессоры включают в себя аппаратные ускорители и ASIC для криптографических функций, а графический процессор NVIDIA Tesla T4 включает встроенные элементы FPGA для приложений логического вывода ИИ

В чем можно писать программы? Для создания программы необходимы: - тестовый редактор; - компилятор; - редактор связей; - библиотеки функций. Кроме программатора для написания и загрузки программы нужна **IDE – среда для разработки**. Можно, конечно, писать код в блокноте, пропускать через компиляторы и т.д. Зачем это нужно, когда есть отличные готовые варианты.

Для AVR созданы интегрированные среды разработки (**IDE**, Integrated Development Environment) - это система программных средств, используемая для разработки программного обеспечения (ПО). Наиболее распространенные из них **AVRStudio, ATmelStudio, WINAVR, CodeVision, IAR Embedded Workbench**, которые позволяют создавать проекты, и писать программы как в ассемблере, так и на СИ. Кроме этого, аппаратные вычислительные платформы типа **Arduino** требуют минимума знаний языка C++.

Для **разработки ПО** под ARM архитектуру применяются инструментарии от компаний **Keil** и **IAR Systems** (под C). Компания Keil принадлежит ARM, и при пользовании услугами этих двух компаний вы получаете очень хорошую техническую поддержку. Существует и вариант полностью бесплатного инструментария, например, **среда разработки Eclipse** и **компилятор GCC**.

Программатор нужен для загрузки прошивки в память МК, самым дешевым и популярным считается **USBASP**. В некоторых случаях используют микроконтроллеры AVR с аппаратной поддержкой USB - ATmega8U2; □ □ ATmega16U2; □ ATmega32U2; □ В память МК предварительно нужно загрузить UART бутлоадер - для этого все равно нужен программатор для AVR-микроконтроллеров.

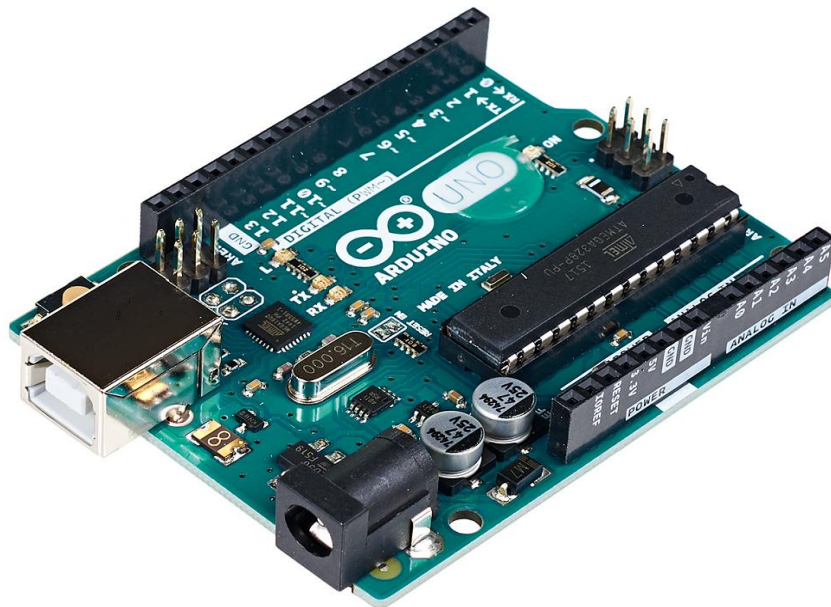
Визуальное программирование - это графические среды, позволяющих не писать программы, а рисовать их. Scratch, ArduBlock и FLProg — три попытки сделать так, чтобы программирование стало доступно даже дошкольникам. Пример профессионального использования – пакет **Matlab/Simulink**, позволяющий выполнять симуляцию работы моделируемых систем и устройств, осуществляя их имитационное моделирование, и автоматическую генерацию кодов C/C++, Verilog и VHDL на основе отлаженной модели. При этом коды C/C++ возможно применить для прошивки МК, а коды Verilog и VHDL – для конфигурации ПЛИС и разработки ASICs.

Проект **Arduino** разработан специально для учебных целей. Он насчитывает десятки плат различных форм и с разным количеством контактов. Главное в ардуино – это то, что вы покупаете не просто микроконтроллер, а **полноценную отладочную плату**, распаянную на качественной текстолитовой печатной плате, покрытой маской и смонтированными SMD компонентами. Самые распространенные – это Arduino Nano и Arduino UNO, они по сути своей идентичны, разве что «Нано» меньше примерно в 3 раза чем «Уно».

- Ардуино может программироваться стандартным языком – «C»; своим собственным – wiring. Стандартная среда для разработки – Arduino IDE; **Программирование** ведется целиком через собственную программную оболочку (IDE), бесплатно доступную на сайте Arduino. В этой оболочке имеется **текстовый редактор, менеджер проектов, препроцессор, компилятор и инструменты для загрузки программы в микроконтроллер**. Оболочка написана на Java на основе проекта Processing, работает под Windows, Mac OS X и Linux. Для программирования Arduino используется упрощенная версия языка C. Для упрощения разработки прошивок существует множество функций, классов, методов и библиотек. Благодаря этому работать с этими микроконтроллерами очень удобно и легко. <https://all-arduino.ru/programmirovanie-arduino/>

Для соединения с компьютером достаточно лишь подключить USB шнур к гнезду USB на плате ардуино, установить драйвера (скорее всего это произойдет автоматически). Программатор установлен на плате.

«**Скетчи**» – это название программ для ардуино;



Характеристики микроконтроллера ATmega328;

Тактовая частота: 16 МГц, Напряжение логических уровней: 5 В

Входное напряжение питания: 7–12 В

Портов ввода-вывода общего назначения: 20

Максимальный ток с пина ввода-вывода: 40 мА

Максимальный выходной ток пина 3.3V: 50 мА

Максимальный выходной ток пина 5V: 800 мА

Портов с поддержкой ШИМ: 6

Портов, подключённых к АЦП: 6

Разрядность АЦП: 10 бит

Flash-память: 32 КБ, EEPROM-память: 1 КБ, Оперативная память: 2 КБ

Габариты: 69×53 мм

Сердцем платформы Arduino Uno является 8-битный микроконтроллер семейства AVR — ATmega328P. Микроконтроллер ATmega16U2 обеспечивает связь микроконтроллера ATmega328P с USB-портом компьютера. При подключении к ПК Arduino Uno определяется как виртуальный COM-порт.

Структура AVR МК изображена на схеме:

АЛУ – арифметико-логическое устройство для выполнения вычислений.

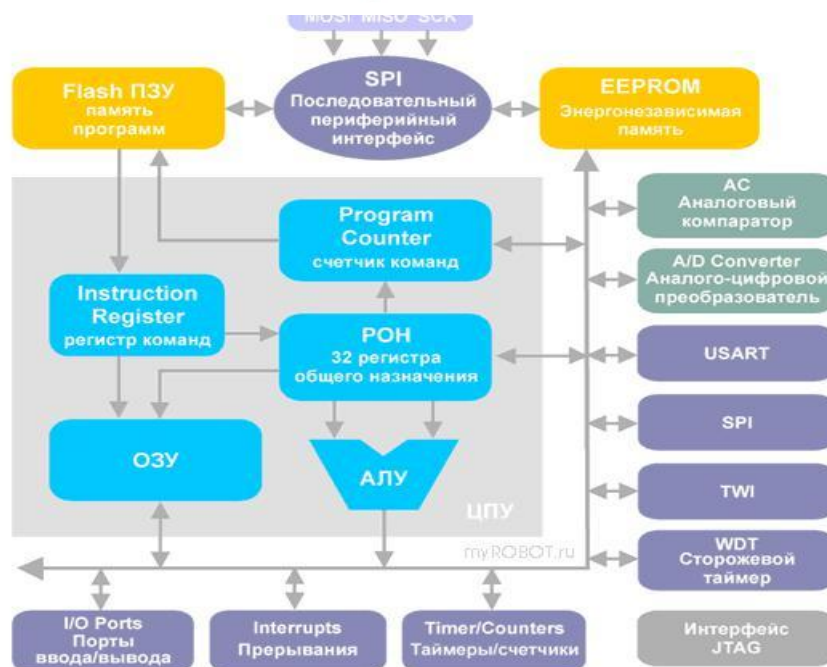
Регистры общего назначения (РОН) – регистры, которые служат как временные ячейки для операций с данными, могут принимать данные и хранить их, после перезагрузки стираются.

Прерывания – события, которые возникают по внутренним или внешним воздействиям и требуют немедленного отвлечения МК.

JTAG – интерфейс для внутрисхемного программирования без снятия МК с платы.

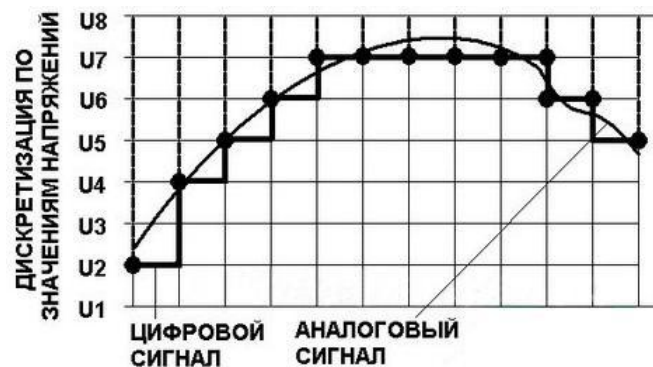
Flash, ОЗУ, EEPROM – виды памяти – программ, временных рабочих данных, долгосрочного хранения. Последние независимы от подачи питания к МК.

Таймеры и счетчики – важнейшие узлы в МК, в некоторых моделях их количество может быть до десятка. Их работа зависит от



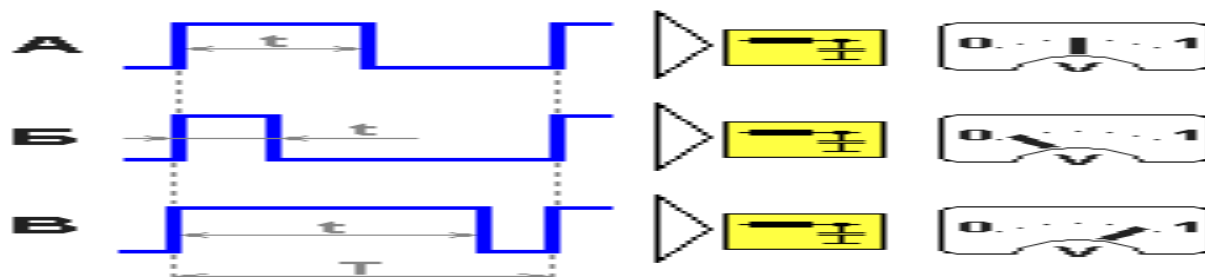
программы, однако выполняются эти действия аппаратно, параллельно основному тексту программы, могут вызвать прерывание (например, по переполнению таймера) на любом этапе выполнения кода. Нужны для того, чтобы отчитывать количество тактов, временные отрезки. Счетчики увеличивают свое значение по какому-либо из событий.

В платах **Arduino (Atmega328P)**, используется 8-канальный АЦП, с разрядностью **10 бит**. К цифровым выводам подключаются цифровые датчики, цифровой сигнал может являться только 1 (единицей) или 0 (нулем), в то время как аналоговый может принимать бесконечное множество значений. **Разрядность АЦП** характеризует качество, точность и чувствительность аналогового входа. Например, 10 битный АЦП (2 в степени 10), распознает плавно изменяющийся цифровой сигнал как числовое значение от 0 до 1024 . 12 битный АЦП видит тот же сигнал, но с более высокой точностью – в виде от 0 до 4096 , а это значит, что измеренные значения входного сигнала будут в 4 раза точнее.



*) Условия, при которых возможно полное восстановление аналогового сигнала по его цифровому эквиваленту с сохранение всей исходно содержащейся в сигнале информации, выражаются теоремами **Найквиста**, **Уиттекера**, **Котельникова**, **Шеннона**. Сущность всех этих теорем практически одинакова: - для дискретизации аналогового сигнала с полным сохранением информации, в его цифровом эквиваленте максимальные частоты в аналоговом сигнале не должны превышать половину частоты дискретизации, т. е. $f_{max} < 1/2 f_d$.

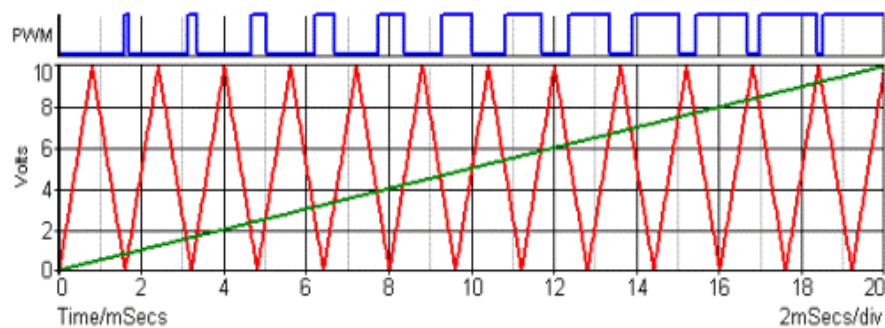
ШИМ – каналы для управления нагрузкой



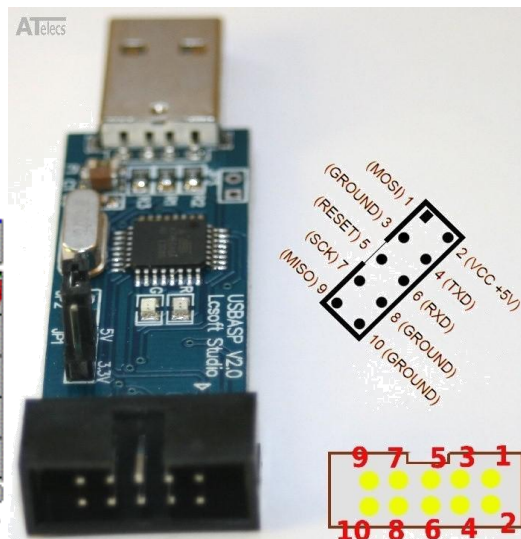
Выходы микроконтроллера довольно «слабые», имеется в виду то, что ток через них обычно составляет $20-40$ миллиампер, что хватит для свечения светодиода и LED-индикаторов. Для более мощной нагрузки необходимы усилители тока или напряжения, например, те же транзисторы или **специальные адаптеры (драйверы)**.

ШИМ-каналы, их можно задействовать, например, для регулировки яркости, температуры, или оборотов двигателя. В том же 328 контроллере их 6 . *) **ШИМ** — это Широтно-Импульсная Модуляция, (по-английски PWM — Pulse Width Modulation), связанная с импульсами

и их шириной. Если изменять ширину (длительность) импульсов постоянной частоты, то можно управлять, например, яркостью источника света, скоростью вращения вала электродвигателя или температурой какого-либо нагревательного элемента.



Принцип получения ШИМ.



Программаторы для МК.