

**Департамент образования Ивановской области**

Областное государственное бюджетное профессиональное

образовательное учреждение

**«Ивановский промышленно-экономический колледж»**

ЦИКЛОВАЯ МЕТОДИЧЕСКАЯ КОМИССИЯ

ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

**Лекционный курс**

**ОП.02 Архитектура аппаратных средств**

для студентов специальности

09.02.07 Информационные системы и программирование

Иваново 2021 г.

ВВК

Г

Автор[[1]](#footnote-1): Куличихина А.О.

Лекционный курс ОП.02 Архитектура аппаратных средств для обучающихся по специальности 09.02.07 Информационные системы и программирование / авт. А.О. Куличихина – Иваново: ОГБПОУ ИВПЭК, 2021. – 61 с.

Данный лекционный курс предназначен для ознакомления с ОП.02 Архитектура аппаратных средств, для обучающихся по специальности 09.02.07 Информационные системы и программирование.

Одобрено на заседании ЦМК Информационно-коммуникационных технологий.

Протокол № 4 от «9» декабря 2021г.

Председатель ЦМК: Т.А. Буркова

© Областное государственное бюджетное профессиональное

образовательное учреждение «Ивановский промышленно-

экономический колледж», 2021г.

© А.О. Куличихина, 2021г.

**Содержание**

[Лекция 1 4](#_Toc89946049)

[Лекция 2 7](#_Toc89946050)

[Лекция 3 11](#_Toc89946051)

[Лекция 4 17](#_Toc89946052)

[Лекция 5 23](#_Toc89946053)

[Лекция 6 29](#_Toc89946054)

[Лекция 7 34](#_Toc89946055)

[Лекция 8 51](#_Toc89946056)

[Лекция 9 55](#_Toc89946057)

[Лекция 10 60](#_Toc89946058)

Лекция 1

**Понятия аппаратных средств ЭВМ, архитектуры аппаратных средств**

**Понятия аппаратных средств ЭВМ**

**Компьютер** – модульный прибор, состоящий из различных устройств, каждое из которых выполняет определенные задачи.

**Компьютер** – это электронное устройство, которое выполняет операции ввода информации, хранения и обработки ее по определенной программе, вывод полученных результатов в форме, пригодной для восприятия человеком. За любую из названных операций отвечают специальные блоки компьютера.

За все эти операции отвечают соответствующие блоки компьютера:

– устройства ввода информации;

– центральный процессор;

– запоминающие устройства;

– устройства вывода информации.

Все эти блоки, в свою очередь, состоят из отдельных устройств. Так, в центральный процессор могут входить арифметико-логическое устройство, внутреннее запоминающее устройство в виде регистров процессора и кэш-памяти, управляющее устройство. Устройство ввода, как правило, тоже не является одной конструктивной единицей. Поскольку виды входной информации разнообразны, источников ввода данных может быть несколько: клавиатура служит для ввода буквенно-цифровой информации, сканер – для ввода графической информации, а мышь –для ввода сигналов управления прикладными программами.

**Запоминающее устройство** – это блок ЭВМ, предназначенный для временного (оперативная память) и продолжительного (постоянная память) хранения программ, входных и результирующих данных, а также промежуточных результатов. Информация в оперативной памяти сохраняется временно лишь при включенном питании, но оперативная память имеет большее быстродействие. В постоянной памяти данные могут сохраняться даже при отключенном компьютере, но скорость обмена данными между постоянной памятью и центральным процессором, в подавляющем большинстве случаев, значительно меньше.

**Арифметико-логическое устройство (АЛУ)** – это блок ЭВМ, в котором происходит преобразование данных по командам программы: арифметические действия над числами, преобразование кодов и др.

**Управляющее устройство (УУ)** координирует работу всех блоков компьютера. В определенной последовательности он выбирает из оперативной памяти команду за командой. Каждая команда декодируется, по потребности элементы данных из указанных в команде ячеек оперативной памяти передаются в АЛУ; АЛУ настраивается на выполнение действия, указанной текущей командой (в этом действии могут принимать участие также устройства ввода-вывода); дается команда на выполнение этого действия. Этот процесс будет продолжаться до тех пор, пока не возникнет одна из следующих ситуаций: исчерпаны входные данные, от одного из устройств поступила команда на прекращение работы, выключено питание компьютера.

Описанный принцип построения ЭВМ носит название фон нейманской архитектуры - по имени американского ученого венгерского происхождения Джона фон Неймана, который предложил ее еще в 40-е годы прошлого столетия.

**Классическая архитектура (архитектура фон Неймана)**– одно арифметико-логическое устройство (АЛУ), через которое проходит поток данных, и одно устройство управления (УУ), через которое проходит поток команд – программа. Это однопроцессорный компьютер. К этому типу архитектуры относится и архитектура персонального компьютера с общей шиной. Все функциональные блоки здесь связаны между собой общей шиной, называемой также системной магистралью.

Основными блоками фон Неймана являются устройства управления (УУ) и АЛУ, память (ОЗУ), внешняя память, устройства ввода и вывода.

Современную архитектуру компьютера определяют следующие принципы:

**Принцип программного управления**. Обеспечивает автоматизацию процесса вычислений на ЭВМ. Согласно этому принципу, для решения каждой задачи составляется программа, которая определяет последовательность действий компьютера. Эффективность программного управления будет выше при решении задачи этой же программой много раз (хотя и с разными начальными данными).

**Принцип программы, сохраняемой в памяти.** Согласно этому принципу, команды программы подаются, как и данные, в виде чисел и обрабатываются так же, как и числа, а сама программа перед выполнением загружается в оперативную память, что ускоряет процесс ее выполнения.

**Принцип произвольного доступа к памяти.** В соответствии с этим принципом, элементы программ и данных могут записываться в произвольное место оперативной памяти, что позволяет обратиться по любому заданному адресу (к конкретному участку памяти) без просмотра предыдущих.

На основании этих принципов можно утверждать, что современный компьютер – техническое устройство, которое после ввода в память начальных данных в виде цифровых кодов и программы их обработки, выраженной тоже цифровыми кодами, способно автоматически осуществить вычислительный процесс, заданный программой, и выдать готовые результаты решения задачи в форме, пригодной для восприятия человеком.

Реальная структура компьютера значительно сложнее, чем рассмотренная выше (ее можно назвать логической структурой). В современных компьютерах, в частности персональных, все чаще происходит отход от традиционной архитектуры фон Неймана, обусловленный стремлением разработчиков и пользователей к повышению качества и производительности компьютеров. Качество ЭВМ характеризуется многими показателями. Это и набор команд, которые компьютер способный понимать, и скорость работы (быстродействие) центрального процессора, количество периферийных устройств ввода-вывода, присоединяемых к компьютеру одновременно и т.д. Главным показателем является быстродействие – количество операций, какую процессор способен выполнить за единицу времени. На практике пользователя больше интересует производительность компьютера – показатель его эффективного быстродействия, то есть способности не просто быстро функционировать, а быстро решать конкретные поставленные задачи.

Как результат, все эти и прочие факторы способствуют принципиальному и конструктивному усовершенствованию элементной базы компьютеров, то есть созданию новых, более быстрых, надежных и удобных в работе процессоров, запоминающих устройств, устройств ввода-вывода и т.д. Тем не менее, следует учитывать, что скорость работы элементов невозможно увеличивать беспредельно (существуют современные технологические ограничения и ограничения, обусловленные физическими законами). Поэтому разработчики компьютерной техники ищут решения этой проблемы усовершенствованием архитектуры ЭВМ.

Так, появились компьютеры с многопроцессорной архитектурой, в которой несколько процессоров работают одновременно, а это означает, что производительность такого компьютера равняется сумме производительностей процессоров. В мощных компьютерах, предназначенных для сложных инженерных расчетов и систем автоматизированного проектирования (САПР), часто устанавливают два или четыре процессора. В сверхмощных ЭВМ (такие машины могут, например, моделировать ядерные реакции в режиме реального времени, прогнозировать погоду в глобальном масштабе) количество процессоров достигает нескольких десятков.

Скорость работы компьютера существенным образом зависит от быстродействия оперативной памяти. Поэтому, постоянно ведутся поиски элементов для оперативной памяти, затрачивающих меньше времени на операции чтения-записи. Но вместе с быстродействием возрастает стоимость элементов памяти, поэтому наращивание быстродействующей оперативной памяти нужной емкости не всегда приемлемо экономически.

Проблема решается построением многоуровневой памяти. Оперативная память состоит из двух-трех частей: основная часть большей емкости строится на относительно медленных (более дешевых) элементах, а дополнительная (так называемая кэш-память) состоит из быстродействующих элементов. Данные, к которым чаще всего обращается процессор находятся в кэш-памяти, а больший объем оперативной информации хранится в основной памяти.

Раньше работой устройств ввода-вывода руководил центральный процессор, что занимало немало времени. Архитектура современных компьютеров предусматривает наличие каналов прямого доступа к оперативной памяти для обмена данными с устройствами ввода-вывода без участия центрального процессора, а также передачу большинства функций управления периферийными устройствами специализированным процессорам, разгружающим центральный процессор и повышающим его производительность.

Лекция 2

**История развития вычислительных устройств и приборов. Классификация ЭВМ по принципу действия, по поколениям, назначению, по размерам и функциональным возможностям.**

История компьютера тесным образом связана с попытками человека облегчить, автоматизировать большие объёмы вычислений. Даже простые арифметические операции с большими числами затруднительны для человеческого мозга. Поэтому уже в древности появилось устройство – абак. Абак (греч. αβαξ, abákion, лат. abacus − доска) − это счётная доска, простейшее счётное устройство, применявшееся для арифметических вычислений приблизительно с IV века до н. э. в Древней Греции, Древнем Риме. В Европе абак применялся до XVIII века.

В *России* ещё в средние века (16-17 вв.) на основе абака было разработано другое приспособление – *русские счёты.*

Механические приспособления

Механизация вычислительных операций началась в XVII веке. На первом этапе для создания механических вычислительных устройств использовались механизмы, аналогичные часовым.

В 1623 год − немецкий ученый Вильгельм Шиккард разработал первое в мире механическое устройство («суммирующие часы») для выполнения *операций* *сложения и вычитания* шестиразрядных десятичных чисел. Было ли устройство реализовано при жизни изобретателя, достоверно неизвестно, но в 1960 году оно было воссоздано по чертежам и подтвердило свою работоспособность.

В 1642 году французский механик Блез Паскаль сконструировал первое в мире механическое цифровое вычислительное устройство («Паскалин»), построенное на основе зубчатых колес. Оно могло суммировать и вычитать пятиразрядные десятичные числа, а последние модели оперировали числами с восемью десятичными разрядами.

В 1673 г. немецкий философ и математик создал механический калькулятор, который при помощи двоичной системы счисления выполнял умножение, деление, сложение и вычитание. Операции умножения и деления выполнялись путём многократного повторения операций сложения и вычитания.

Однако широкое распространение вычислительные аппараты получили только в 1820 году, когда француз Чарльз Калмар изобрёл машину, которая могла производить *четыре основных арифметических действия*. Машину Калмара назвали арифмометр. Благодаря своей универсальности арифмометры использовались довольно длительное время до 60-х годов ХХ века.

Автоматизация вычислений

Идея автоматизации вычислительных операций пришла из часовой промышленности. Старинные монастырские башенные часы были построены так, чтобы в заданное время включать механизм, связанный с системой колоколов.

В 1833 году английский ученый, профессор Кембриджского университета Чарльз Беббидж разработал проект аналитической машины, которая имела черты современного компьютера. Это был гигантский арифмометр с программным управлением, арифметическим и запоминающим устройствами. Оно имело устройство для ввода информации, блок управления, запоминающее устройство и устройство вывода результатов.

Сотрудницей и помощницей Ч. Беббиджа во многих его научных изысканиях была леди Ада Лавлейс (урожденная Байрон).

Она разработала первые программы для машины и предвосхитила основы современного программирования для цифровых вычислительных машин с программным управлением. Заложила многие идеи и ввела ряд понятий и терминов, сохранившихся до настоящего времени.

Она предсказала появление современных компьютеров как многофункциональных машин не только для вычислений, но и для работы с графикой, звуком. В середине 70-х годов двадцатого столетия министерство обороны США официально утвердило название единого языка программирования американских вооруженных сил. Язык носит название Ada. День программиста отмечается в день рождения Ады Лавлейс [10 декабря](https://pandia.ru/text/category/10_dekabrya/).

Особенностью Аналитической машины стало то, что здесь впервые был реализован принцип разделения информации на *команды и данные*. Для ввода и вывода данных Бэббидж предлагал использовать перфокарты-листы из плотной бумаги с информацией, наносимой с помощью отверстий.

В 1888 году американский инженер Герман Холлерит сконструировал первую электромеханическую счётную машину. Эта машина, названная табулятором, могла считывать и сортировать статистические записи, закодированные на перфокартах. Для работы этой машины использовалось электричество. В 1890 изобретение Холлерита было использовано в 11-ой американской переписи населения. Работа, которую 500 сотрудников выполняли в течение семи лет, Холлерит с 43 помощниками на 43 табуляторах выполнил за один месяц.

Дальнейшее развитие науки и техники позволили в 1940-х годах построить первые вычислительные машины. В 1944 г. американский инженер Говард Эйкен при поддержке фирмы Ай-Би-Эм (IBM) сконструировал компьютер для выполнения баллистических расчетов. Этот компьютер, названный «Марк 1», по площади занимал примерно половину футбольного поля и включал более 800 километров проводов, около 750 тыс. деталей, 3304 реле. «Марк-1» был основан на использовании электромеханических реле и оперировал десятичными числами, закодированными на перфоленте. Машина могла манипулировать числами длиной до 23 разрядов. Для перемножения двух 23-разрядных чисел ей было необходимо 4 секунды.

Но электромеханические реле работали недостаточно быстро. В 1946 г. По заказу Армии США был создан первый широкомасштабный электронный цифровой компьютер ЭНИАК (ENIAC - электронный числовой интегратор и вычислитель), который можно было перепрограммировать для решения полного диапазона задач. Разработали его американские ученые Джон Уильям Мокли и Джон Преспер Экерт. В ЭНИАКе в качестве основы компонентной базы электромеханические реле были заменены [вакуумными](https://pandia.ru/text/category/vakuum/) лампами. Всего комплекс включал 17468 ламп, 7200 кремниевых диодов, 1500 реле, 70000 резисторов и 10000 конденсаторов. Потребляемая мощность – 150 кВт по тем временам было достаточно для освещения большого города. Вычислительная мощность – 300 операций умножения или 5000 операций сложения в секунду. Вес – 27 тонн, более 30 метров. Вычисления проводились в десятичной системе. ЭНИАК использовался для расчета баллистических таблиц, предсказания погоды, расчетов в области [атомной энергетики](https://pandia.ru/text/category/atomnaya_yenergetika/), аэродинамики, изучения космоса.

В СССР вычислительная машина МЭСМ (малая электронная счётная машина) была создана в 1951 году под руководством академика Сергея Алексеевича Лебедева. Машина вычисляла факториалы натуральных чисел и решала уравнения параболы. Одновременно Лебедев работал над созданием БЭСМ - быстродействующей электронной счётной машины, разработка которой была завершена в 1953 году.

В 1971 году фирмой Intel (США) был создан первый микропроцессор - программируемое логическое устройство, изготовленное по технологии СБИС (сверхбольших интегральных схем).

В 1964г. сотрудник Стэнфордского исследовательского центра Дуглас Энгельбарт продемонстрировал работу первой мыши-манипулятора, но только четыре года спустя мышка была показана на компьютерной конференции в Сан-Франциско.

Первый персональный компьютер (ПК) в 1976г. выпустила фирма Apple; в СССР ПК появились в 1985г.

**Классификация ЭВМ.**

**По принципу действия.** Электронная вычислительная машина, компьютер -- комплекс технических средств, предназначенных для автоматической обработки информации в процессе решения вычислительных и информационных задач. По принципу действия вычислительные машины делятся на три больших класса: аналоговые (АВМ), цифровые (ЦВМ) и гибридные (ГВМ).

**Цифровые вычислительные машины** (ЦВМ) -- вычислительные машины дискретного действия, работают с информацией, представленной в дискретной, а точнее, в цифровой форме. **Аналоговые вычислительные машины** (АВМ) -- вычислительные машины непрерывного действия. Работают с информацией, представленной в непрерывной (аналоговой) форме, т.е. в виде непрерывного ряда значений какой-либо физической величины (чаще всего электрического напряжения).

**Гибридные вычислительные машины** (ГВМ) -- вычислительные машины комбинированного действия. Работают с информацией, представленной и в цифровой, и в аналоговой форме; они совмещают в себе достоинства АВМ и ЦВМ. ГВМ целесообразно использовать для решения задач управления сложными быстродействующими техническими комплексами.

**По назначению.** По назначению ЭВМ можно разделить на три группы: универсальные (общего назначения), проблемно-ориентированные и специализированнные.

**Универсальные ЭВМ** предназначены для решения самых различных инженерно-технических задач: экономических, математических, информационных и других задач, отличающихся сложностью алгоритмов и большим объемом обрабатываемых данных. Они широко используются в вычислительных центрах коллективного пользования и в других мощных вычислительных комплексах.

Характерными чертами универсальных ЭВМ являются:

* высокая производительность;
* разнообразие форм обрабатываемых данных: двоичных, десятичных, символьных, при большом диапазоне их изменения и высокой точности их представления;
* обширная номенклатура выполняемых операций, как арифметических, логических так и специальных;
* большая емкость оперативной памяти;
* развитая организация системы ввода-вывода информации, обеспечивающая подключение разнообразных видов внешних устройств.

**Проблемно-ориентированные ЭВМ** служат для решения более узкого круга задач, связанных, как правило, с управлением технологическими объектами; регистрацией, накоплением и обработкой относительно небольших объемов данных; они обладают ограниченными по сравнению с универсальными ЭВМ аппаратными и программными ресурсами (управляющие вычислительные комплексы).

**Специализированные ЭВМ** используются для решения узкого круга задач или реализации строго определенной группы функций. Такая узкая ориентация ЭВМ позволяет четко специализировать их структуру, существенно снизить их сложность и стоимость при сохранении высокой производительности и надежности их работы (программируемые микропроцессоры специального назначения; адаптеры и контроллеры, выполняющие логические функции управления отдельными несложными техническими устройствами, агрегатами и процессами).

**По функциональным возможностям** По функциональным возможностям ЭВМ можно разделить на сверхбольшие (суперЭВМ), большие, малые, сверхмалые (микроЭВМ).

Функциональные возможности ЭВМ обусловливают важнейшие технико-эксплуатационные характеристики:

* быстродействие, измеряемое усредненным количеством операций, за единицу времени;
* разрядность и формы представления чисел, с которыми оперирует ЭВМ;
* номенклатура, емкость и быстродействие всех запоминающих устройств;
* способность ЭВМ одновременно работать с несколькими пользователями и выполнять одновременно несколько программ (многопрограммность);
* типы и технико-эксплуатационные хар-ки операционных систем программ. обеспеч.;
* программная совместимость с другими типами ЭВМ;
* система и структура машинных команд;
* возможность подключения к каналам связи и к вычислительной сети;
* эксплуатационная надежность ЭВМ;
* коэффициент полезного использования ЭВМ во времени, определяемый соотношением времени полезной работы и времени профилактики.

Лекция 3

**Базовые логические операции и схемы. Таблицы истинности. Схемные логические элементы. Принцип работы, таблица истинности, логические выражения, схемы.**

|  |
| --- |
| ***Логический элемент компьютера* — это часть электронной логичеcкой схемы, которая реализует элементарную логическую функцию.** |

**Логическими элементами компьютеров являются электронные схемы И, ИЛИ, НЕ, И—НЕ, ИЛИ—НЕ** и другие (называемые также **вентилями**), а также **триггер.**

С помощью этих схем можно реализовать любую логическую функцию, описывающую работу устройств компьютера. Обычно у вентилей бывает от двух до восьми входов и один или два выхода.

Чтобы представить два логических состояния — “1” и “0” в вентилях, соответствующие им входные и выходные сигналы имеют один из двух установленных уровней напряжения. Например, +5 вольт и 0 вольт.

Высокий уровень обычно соответствует значению “истина” (“1”), а низкий — значению “ложь” (“0”).

**Каждый логический элемент имеет свое условное обозначение, которое выражает его логическую функцию, но не указывает на то, какая именно электронная схема в нем реализована.** Это упрощает запись и понимание сложных логических схем.

Работу логических элементов описывают с помощью таблиц истинности.

|  |
| --- |
| ***Таблица истинности* это табличное представление логической схемы (операции), в котором перечислены все возможные сочетания значений истинности входных сигналов (операндов) вместе со значением истинности выходного сигнала (результата операции) для каждого из этих сочетаний.** |

Глоссарий, определения логики

Высказывание - это повествовательное предложение, про которое можно определенно сказать истинно оно или ложно (истина (логическая 1), ложь (логический 0)).

Логические операции - мыслительные действия, результатом которых является изменение содержания или объема понятий, а также образование новых понятий.

Логическое выражение - устное утверждение или запись, в которое, наряду с постоянными величинами, обязательно входят переменные величины (объекты). В зависимости от значений этих переменных величин (объектов) логическое выражение может принимать одно из двух возможных значений: истина (логическая 1) или ложь (логический 0).

Сложное логическое выражение - логическое выражение, состоящее из одного или нескольких простых логических выражений (или сложных логических выражений), соединенных с помощью логических операций.

Логические операции и таблицы истинности

1) Логическое умножение или конъюнкция:

Конъюнкция - это сложное логическое выражение, которое считается истинным в том и только том случае, когда оба простых выражения являются истинными, во всех остальных случаях данное сложенное выражение ложно.

Обозначение: F = A & B.

Таблица истинности для конъюнкции

A B F

1 1 1

1 0 0

0 1 0

0 0 0

2) Логическое сложение или дизъюнкция:

Дизъюнкция - это сложное логическое выражение, которое истинно, если хотя бы одно из простых логических выражений истинно и ложно тогда и только тогда, когда оба простых логических выраженbя ложны.

Обозначение: F = A v B.

Таблица истинности для дизъюнкции

A B F

1 1 1

1 0 1

0 1 1

0 0 0

3) Логическое отрицание или инверсия:

Инверсия - это сложное логическое выражение, если исходное логическое выражение истинно, то результат отрицания будет ложным, и наоборот, если исходное логическое выражение ложно, то результат отрицания будет истинным. Другими простыми слова, данная операция означает, что к исходному логическому выражению добавляется частица НЕ или слова НЕВЕРНО, ЧТО.

Обозначение: F = ¬A.

Таблица истинности для инверсии

A ¬А

1 0

0 1

4) Логическое следование или импликация:

Импликация - это сложное логическое выражение, которое истинно во всех случаях, кроме как из истины следует ложь. То есть данная логическая операция связывает два простых логических выражения, из которых первое является условием (А), а второе (В) является следствием.

«A → B» истинно, если из А может следовать B.

Обозначение: F = A → B.

Таблица истинности для импликации

A B F

1 1 1

1 0 0

0 1 1

0 0 1

5) Логическая равнозначность или эквивалентность:

Эквивалентность - это сложное логическое выражение, которое является истинным тогда и только тогда, когда оба простых логических выражения имеют одинаковую истинность.

«A ↔ B» истинно тогда и только тогда, когда А и B равны.

Обозначение: F = A ↔ B.

Таблица истинности для эквивалентности

A B F

1 1 1

1 0 0

0 1 0

0 0 1

6) Операция XOR (исключающие или)

«A ⊕ B» истинно тогда, когда истинно А или B, но не оба одновременно.

Эту операцию также называют "сложение по модулю два".

Обозначение: F = A ⊕ B.

A B F

1 1 0

1 0 1

0 1 1

0 0 0

Порядок выполнения логических операций в сложном логическом выражении

1. Инверсия;

2. Конъюнкция;

3. Дизъюнкция;

4. Импликация;

5. Эквивалентность.

Для изменения указанного порядка выполнения логических операций используются скобки.

**Логические элементы и их схемная реализация**

Для выполнения логических операций и решать логические задачи с помощью средств электроники были изобретены логические элементы. Их создают с помощью диодов, транзисторов и комбинированных элементов (диодно-транзисторные). Такая логика получила название диодной логики (ДЛ), транзисторной (ТЛ) и диодно–транзисторной (ДТЛ). Используют как [полевые](https://elenergi.ru/princip-raboty-i-xarakteristiki-polevyx-tranzistorov.html), так и [биполярные транзисторы](https://elenergi.ru/tranzistory-bipolyarnye-osnovnye-xarakteristiki-i-sxemy-soedinenij.html). В последнем случае предпочтение отдается устройствам типа n-p-n, так как они обладают большим быстродействием.

1. **Логический элемент «ИЛИ»**

Схема логического элемента «ИЛИ» представлена на рисунке 1. На каждый из входов может подаваться сигнал в виде какого-то напряжения (единица) или его отсутствия (ноль). На резисторе R появиться напряжение даже при его появлении на каком – либо из диодов.

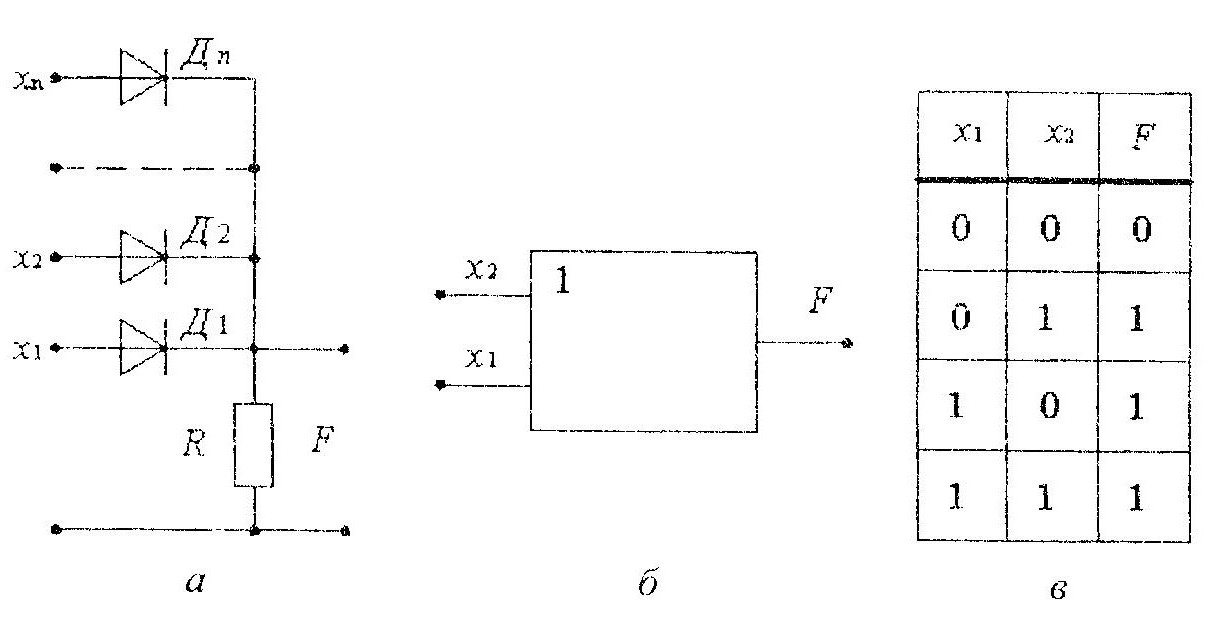
[](https://elenergi.ru/wp-content/uploads/2015/06/elektricheskaya-principialnaya-sxema-logicheskogo-elementa-ili.jpg)

Рисунок 1 – Схема логического элемента «ИЛИ»

Элементы или могут иметь несколько логических входов. Если используются не все входы, то те входы, которые не используются следует соединять с землей (заземлять), чтобы избежать появления посторонних сигналов.

На рисунке 1б показано обозначение на электрической схеме элемента, а на 1в таблица истинности.

1. **Логический элемент «И»**

Схема элемента приведена на рисунке 2. Если хотя – бы к одному из входов будет сигнал равный нулю, то через диод будет протекать ток. Падение напряжения на диоде стремится к нулю, соответственно на выходе тоже будет ноль. На выходе сможет появится сигнал только при условии, что все диоды будут закрыты, то есть на всех входах будет сигнал. Рассчитаем уровень сигнала на выходе устройства:[Напряжение на виходе элемента И](https://elenergi.ru/wp-content/uploads/2015/06/%D0%9D%D0%B0%D0%BF%D1%80%D1%8F%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0-%D0%B2%D0%B8%D1%85%D0%BE%D0%B4%D0%B5-%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%B0-%D0%98.jpg)

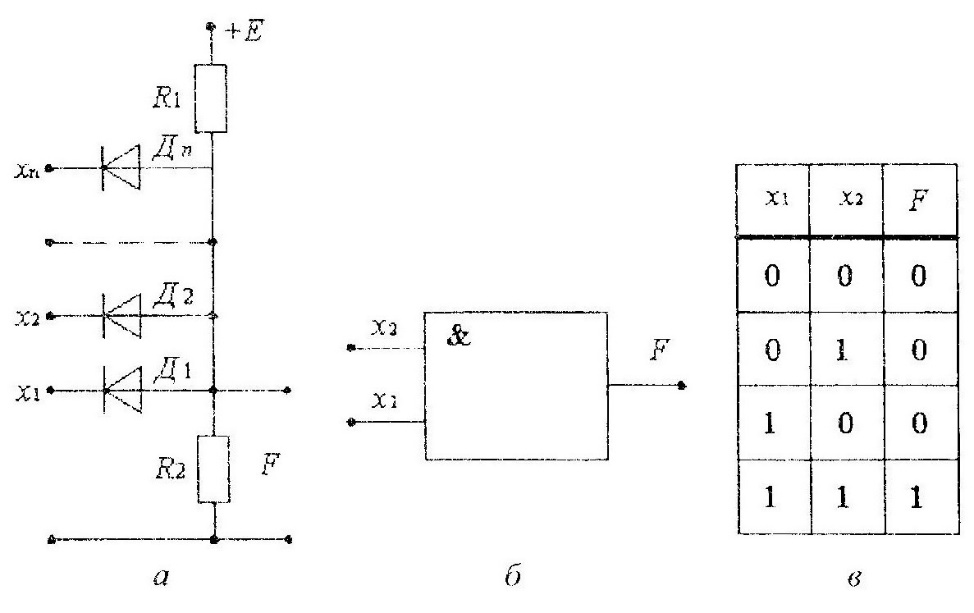
[](https://elenergi.ru/wp-content/uploads/2015/06/elektricheskaya-principialnaya-sxema-logicheskogo-elementa-i.jpg)

Рисунок 2 – Схема логического элемента «И

1. **Логический элемент «НЕ»**

В логическом элементе «НЕ» используют транзистор (Рисунок 3). при наличии положительного напряжения на входе х=1 транзистор открывается и напряжение его коллектора стремится к нулю. Если х=0 то положительного сигнала на базе нет, транзистор закрыт, ток не проходит через коллектор и на резисторе R нет падения напряжения, соответственно на коллекторе появится сигнал Е. условное обозначение и таблица истинности приведены на рисунке 3

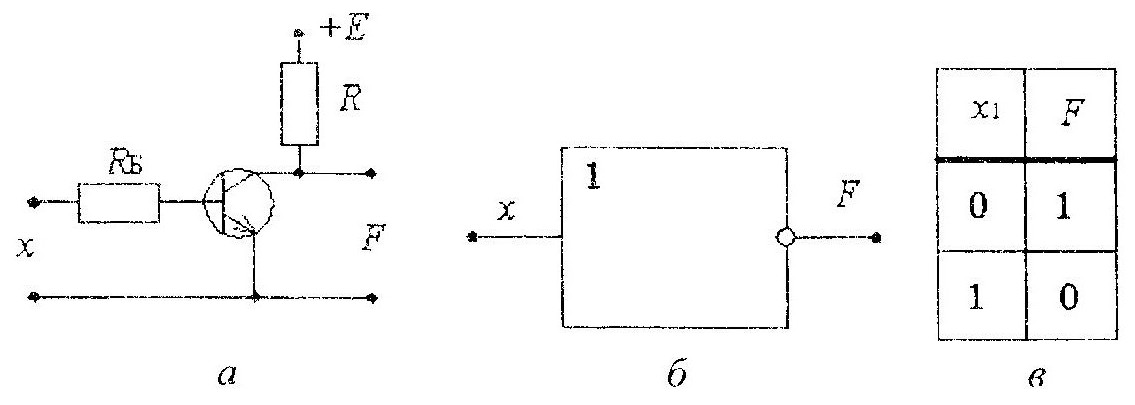
[](https://elenergi.ru/wp-content/uploads/2015/06/elektricheskaya-principialnaya-sxema-logicheskogo-elementa-ne.jpg)

Рисунок 3 – Логический элемент «НЕ»

1. **Логический элемент «ИЛИ-НЕ»**

При создании различных схем на логических элементах часто применяют элементы комбинированные. В таких элементах совмещены несколько функций. Принципиальная схема показана на рисунке 4

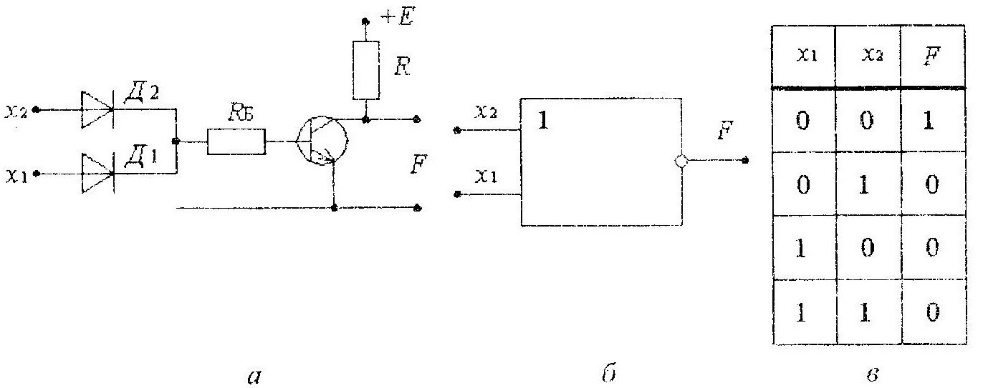
[](https://elenergi.ru/wp-content/uploads/2015/06/%D0%9B%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5-%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D1%8B-%D0%98%D0%9B%D0%98-%D0%9D%D0%95.jpeg)

Рисунок 4 – Логический элемент «ИЛИ-НЕ»

Здесь диоды Д1 и Д2 выполняют роль элемента «ИЛИ», а транзистор играет роль инвертора. Обозначение элемента на схеме и его таблица истинности (Рисунок 4) и в соответственно.

1. **Логический элемент «И-НЕ»**

Показана схема на рисунке 5. Здесь диод Д3 выполняет роль так сказать фильтра во избежание искажения сигнала. Если на вход х1 или х2 не подан сигнал (х1=0 или х2=0), то через диод Д1 или Д2 будет протекать ток. Падение на нем не равно нулю и может оказаться достаточным для открытия транзистора. Последствием чего может стать ложное срабатывание и на выходе вместо единицы мы получим ноль. А если в цепь включить Д3, то на нем упадет значительная часть напряжения открытого на входе диода, и на базу транзистора практически ничего не приходит. Поэтому он будет закрыт, а на выходе будет единица, что и требуется при наличии нуля на каком-либо из входов. На рисунке 5 и в показаны таблица истинности и схемное обозначение данного устройства.

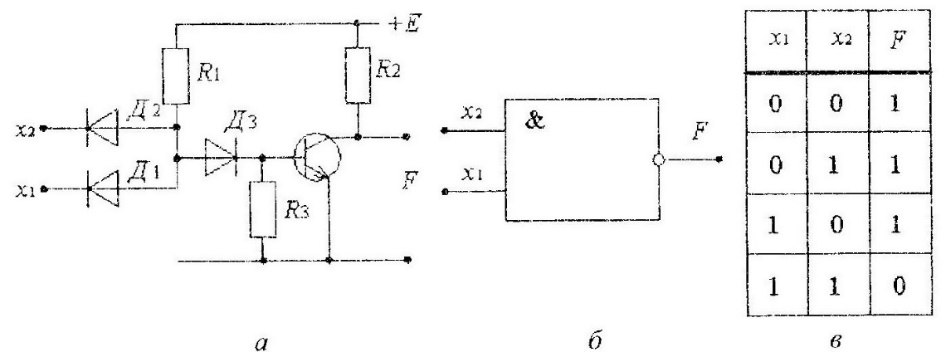
[](https://elenergi.ru/wp-content/uploads/2015/06/elektricheskaya-principialnaya-sxema-logicheskogo-elementa-i-ne.jpg)

Рисунок 5 – Логический элемент «И-НЕ»

Логические элементы получили широчайшее применение в электронике и микропроцессорной технике. Многие системы управления строятся с использованием именно этих устройств.

Лекция 4

**Базовые представления об архитектуре ЭВМ. Принципы фон Неймана. Простейшие типы архитектур. Принцип открытой архитектуры. Магистрально-модульный принцип организации ЭВМ. Классификация параллельных компьютеров. Классификация архитектур вычислительных систем.**

Архитектурой компьютера считается его представление на неко­тором общем уровне, включающее описание пользовательских воз­можностей программирования, системы команд, системы адреса­ции, организации памяти и т. д. Архитектура определяет принципы действия, информационные связи и взаимное соединение основных устройств компьютера: процессора, основной памяти (ОП), внешних запоминающих устройств (ВЗУ) и периферийных устройств (ПУ). Совокупность процессора и основной памяти называют центральным устройством (ЦУ) компьютера. Общность архитектуры разных компьютеров обеспечивает их со­вместимость с точки зрения пользователя.

Структура компьютера — это совокупность его функциональ­ных элементов и связей между ними. Элементами могут быть самые различные устройства — от основных устройств компьютера до простейших схем. Структура компьютера графически представля­ется в виде структурных схем, с помощью которых можно дать опи­сание компьютера на любом уровне детализации.

В настоящее время в зависимости от организации ЦУ различают два вида архитектур ЭВМ:

1 Архитектура фон Неймана.

2 Гарвардская архитектура.

Принципы (архитектура) фон Неймана

В основу построения большинства компьютеров положены сле­дующие общие принципы, сформулированные в 1945 г. американ­ским ученым Джоном фон Нейманом.

1. Принцип программного управления. Из него следует, что про­грамма состоит из набора команд, которые выполняются процессо­ром автоматически друг за другом в определенной последователь­ности.

Выборка команд программы из памяти осуществляется с помощью счетчика команд. Этот узел процессора последовательно увели­чивает хранимый в нем адрес очередной команды на длину коман­ды. Так как команды программы расположены в памяти друг за дру­гом, то тем самым организуется выборка цепочки команд из после­довательно расположенных ячеек памяти.

Если после выполнения команды следует перейти не к следую­щей, а к какой-то другой, используются команды условного или без­условного переходов (ветвления), которые заносят в счетчик команд номер ячейки памяти, содержащей следующую команду. Выборка команд программы из памяти прекращается после достижения и выполнения команды «стоп».

Таким образом, процессор исполняет программу автоматически, без вмешательства человека.

2. Принцип однородности памяти. Программы и данные хранятся в одной и той же памяти. Поэтому компьютер не различает, что хранится в данной ячейке памяти — число, текст или команда. Над командами можно выполнять такие же действия, как и над данны­ми. Это открывает целый ряд возможностей. Например, программа в процессе своего выполнения также может подвергаться переработ­ке, что позволяет задавать в самой программе правила получения некоторых ее частей. Более того, команды одной программы мо­гут быть получены как результаты исполнения другой программы. На этом принципе основаны методы трансляции —перевода текста программы с языка программирования высокого уровня на язык конкретной машины.

3.Принцип адресности. Структурно основная память состоит из перенумерованных ячеек; процессору в произвольный момент вре­мени доступна любая ячейка. Отсюда следует возможность давать имена областям памяти, так, чтобы к запомненным в них значени­ям можно было впоследствии обращаться в процессе выполнения программ с использованием присвоенных имен.

**Принцип открытой архитектуры заключается в следующем:**

1. Регламентируются и стандартизируются только описание принципа действия компьютера и его конфигурация (определённая совокупность аппаратных средств и соединений между ними). Таким образом, компьютер можно собирать из отдельных узлов и деталей, разработанных и изготовленных независимыми фирмами-производителями.
2. Компьютер легко расширяется и модернизируется за счёт наличия разъемов, в которые пользователь может вставлять разнообразные устройства, и, тем самым устанавливать конфигурацию своей машины в соответствии со своими личными предпочтениями.

**Выгоды открытой архитектуры:**

* Конкуренция между производителями привела к удешевлению компьютерных комплектующих, а значит и самих компьютеров
* Появление большого количества компьютерного оборудования позволило покупателям расширить свой выбор, что также способствовало снижению цен на комплектующие и повышению их качества
* Модульная структура компьютера и простота сборки позволила пользователям самостоятельно выбирать необходимые им устройства и с легкостью производить их установку, также стало возможным без особых сложностей в домашних условиях собирать и модернизировать свой компьютер
* Возможность модернизации привела к тому, что пользователи смогли выбирать компьютер исходя из своих настоящих потребностей и толщины кармана, что опять-таки способствовало все большей популярности персональных компьютеров.

**Недостатки:**могут возникнуть проблемы совместимости

Преимущества открытой архитектуры заключаются в том, что пользователь получает возможность:

1) выбрать конфигурацию компьютера. Действительно, если вам не нужен принтер, или не хватает средств на его приобретение, никто не заставляет вас его покупать вместе с новым компьютером. Раньше было не так,- все устройства продавались единым комплектом, причем какого-то определенного типа, так, что выбрать или заменить что-то было невозможно.

2) расширить систему, подключив к ней новые устройства. Например, накопив денег и купив принтер, вы легко сможете подключить его к вашему компьютеру.

3) модернизировать систему, заменив любое из устройств более новым. Действительно, не нужно для этого выбрасывать весь компьютер! Достаточно вместо одного устройства подключить другое. В частности, можно заменить материнскую плату, чтобы из компьютера на базе процессора старого типа получить компьютер на базе процессора нового типа.

#### Магистрально-модульный принцип построения компьютера

Под *архитектурой компьютера* понимается его логическая организация, структура, ресурсы, т. е. средства вычислительной системы, которые могут быть выделены *процессу обработки данных* на определенный интервал времени. Архитектура современных ПК основана **на магистрально-модульном принципе**.

Модульный принцип позволяет потребителю самому подобрать нужную ему конфигурацию компьютера и производить при необходимости его модернизацию. Модульная организация системы опирается на магистральный (шинный) принцип обмена информации. Магистраль или **системная шина** - это набор электронных линий, связывающих воедино по адресации памяти, передачи данных и служебных сигналов процессор, память и периферийные устройства.

Обмен информацией между отдельными устройствами ЭВМ производится по трем многоразрядным шинам, соединяющим все модули, - **шине данных, шине адресов и шине управления**.

Подключение отдельных модулей компьютера к магистрали на физическом уровне осуществляется с помощью контроллеров, а на программном обеспечивается драйверами. Контроллер принимает сигнал от процессора и дешифрует его, чтобы соответствующее устройство смогло принять этот сигнал и отреагировать на него. За реакцию устройства процессор не отвечает - это функция контроллера. Поэтому внешние устройства ЭВМ заменяемы, и набор таких модулей произволен.

Разрядность **шины данных** задается разрядностью процессора, т. е. количеством двоичных разрядов, которые процессор обрабатывает за один такт.

Данные по шине данных могут передаваться как от процессора к какому-либо устройству, так и в обратную сторону, т. е. шина данных является двунаправленной. К основным режимам работы процессора с использованием шины передачи данных можно отнести следующие: запись/чтение данных из оперативной памяти и из *внешних запоминающих устройств*, чтение данных с устройств ввода, пересылка данных на устройства вывода.

Выбор абонента по обмену данными производит процессор, который формирует код адреса данного устройства, а для ОЗУ - код адреса ячейки памяти. Код адреса передается по **адресной шине**, причем сигналы передаются в одном направлении, от процессора к устройствам, т. е. эта шина является однонаправленной.

По **шине управления** передаются сигналы, определяющие характер обмена информацией, и сигналы, синхронизирующие взаимодействие устройств, участвующих в обмене информацией.

Внешние устройства к шинам подключаются посредством **интерфейса**. Под интерфейсом понимают совокупность различных характеристик какого-либо переферийного устройства ПК, определяющих организацию обмена информацией между ним и центральным процессором. В случае несовместимости интерфейсов (например, интерфейс системной шины и интерфейс *винчестера*) используют **контроллеры**.

Чтобы устройства, входящие в состав компьютера, могли взаимодействовать с центральным процессором, в IBM-совместимых компьютерах предусмотрена **система прерываний (Interrupts)**. Система прерываний позволяет компьютеру приостановить текущее действие и переключиться на другие в ответ на поступивший запрос, например, на нажатие клавиши на клавиатуре. Ведь с одной стороны, желательно, чтобы компьютер был занят возложенной на него работой, а с другой - необходима его мгновенная реакция на любой требующий внимания *запрос. Прерывания* обеспечивают немедленную реакцию системы.

**Классификация параллельных компьютеров**

Основным параметром классификации паралелльных компьютеров является наличие общей ([SMP](https://parallel.ru/computers/classes.html#smp)) или распределенной памяти ([MPP](https://parallel.ru/computers/classes.html#mpp)). Нечто среднее между SMP и MPP представляют собой [NUMA](https://parallel.ru/computers/classes.html#numa)-архитектуры, где память физически распределена, но логически общедоступна. [Кластерные](https://parallel.ru/computers/classes.html#cluster) системы являются более дешевым вариантом MPP. При поддержке команд обработки векторных данных говорят о векторно-конвейерных процессорах, которые, в свою очередь могут объединяться в [PVP](https://parallel.ru/computers/classes.html#vector)-системы с использованием общей или распределенной памяти. Все большую популярность приобретают идеи комбинирования различных архитектур в одной системе и построения неоднородных систем.

При организациях распределенных вычислений в глобальных сетях (Интернет) говорят о [мета-компьютерах](https://parallel.ru/computers/reviews/meta-computing.html), которые, строго говоря, не представляют из себя параллельных архитектур.

Более подробно особенности всех перечисленных архитектур будут рассмотрены далее на этой странице, а также в описаниях конкретных компьютеров - представителей этих классов. Для каждого класса приводится следующая информация:

* краткое описание особенностей архитектуры,
* примеры конкретных компьютеров,
* перспективы масштабируемости,
* типичные особенности построения операционных систем,
* наиболее характерная модель программирования (хотя возможны и другие).

Таблица 1. Массивно-параллельные системы (MPP)

|  |  |
| --- | --- |
| **Архитектура** | Система состоит из однородных вычислительных узлов, включающих:   * один или несколько центральных процессоров (обычно RISC), * локальную память (прямой доступ к памяти других узлов невозможен), * коммуникационный процессор или сетевой адаптер * иногда - жесткие диски (как в SP) и/или другие устройства В/В   К системе могут быть добавлены специальные узлы ввода-вывода и управляющие узлы. Узлы связаны через некоторую коммуникационную среду (высокоскоростная сеть, коммутатор и т.п.) |
| **Примеры** | IBM RS/6000 [SP2](https://parallel.ru/computers/computers.html#sp2), Intel PARAGON/ASCI Red, CRAY [T3E](https://parallel.ru/computers/computers.html#crayt3e), Hitachi [SR8000](https://parallel.ru/news/hitachi_sr8000f1.html), транспьютерные системы [Parsytec](https://parallel.ru/computers/vendors.html" \l "parsytec). |
| **Масштабируемость** | Общее число процессоров в реальных системах достигает нескольких тысяч (ASCI Red, Blue Mountain). |
| **Операционная система** | Существуют два основных варианта:   1. Полноценная ОС работает только на управляющей машине (front-end), на каждом узле работает сильно урезанный вариант ОС, обеспечивающие только работу расположенной в нем ветви параллельного приложения. Пример: Cray T3E. 2. На каждом узле работает полноценная UNIX-подобная ОС (вариант, близкий к [кластерному](https://parallel.ru/computers/classes.html#cluster) подходу). Пример: IBM RS/6000 SP + ОС AIX, устанавливаемая отдельно на каждом узле. |
| **Модель программирования** | Программирование в рамках модели **передачи сообщений** ( [MPI](https://parallel.ru/tech/tech_dev/mpi.html), [PVM](https://parallel.ru/tech/tech_dev/ifaces.html#pvm), [BSPlib](https://parallel.ru/tech/tech_dev/ifaces.html" \l "bsplib)) |

### Таблица 2 . Симметричные мультипроцессорные системы (SMP)

|  |  |
| --- | --- |
| **Архитектура** | Система состоит из нескольких однородных процессоров и массива общей памяти (обычно из нескольких независимых блоков). Все процессоры имеют доступ к любой точке памяти с одинаковой скоростью. Процессоры подключены к памяти либо с помощью общей шины (базовые 2-4 процессорные SMP-сервера), либо с помощью crossbar-коммутатора (HP 9000). Аппаратно поддерживается когерентность кэшей. |
| **Примеры** | [HP 9000 V-class](https://parallel.ru/computers/computers.html#exemplar), N-class; SMP-cервера и рабочие станции на базе процессоров Intel (IBM, HP, Compaq, Dell, ALR, Unisys, DG, Fujitsu и др.). |
| **Масштабируемость** | Наличие общей памяти сильно упрощает взаимодействие процессоров между собой, однако накладывает сильные ограничения на их число - не более 32 в реальных системах. Для построения масштабируемых систем на базе SMP используются [кластерные](https://parallel.ru/computers/classes.html#clusters) или [NUMA](https://parallel.ru/computers/classes.html#numa)-архитектуры. |
| **Операционная система** | Вся система работает под управлением единой ОС (обычно UNIX-подобной, но для Intel-платформ поддерживается Windows NT). ОС автоматически (в процессе работы) распределяет процессы/нити по процессорам (scheduling), но иногда возможна и явная привязка. |
| **Модель программирования** | Программирование в модели **общей памяти**. (POSIX threads, [OpenMP](https://parallel.ru/tech/tech_dev/openmp.html)). Для SMP-систем существуют сравнительно эффективные средства [автоматического распараллеливания](https://parallel.ru/tech/tech_dev/auto_par.html). |

### Таблица 3.Системы с неоднородным доступом к памяти (NUMA)

|  |  |
| --- | --- |
| **Архитектура** | Система состоит из однородных базовых модулей (плат), состоящих из небольшого числа процессоров и блока памяти. Модули объединены с помощью высокоскоростного коммутатора. Поддерживается единое адресное пространство, аппаратно поддерживается доступ к удаленной памяти, т.е. к памяти других модулей. При этом доступ к локальной памяти в несколько раз быстрее, чем к удаленной.  В случае, если аппаратно поддерживается когерентность кэшей во всей системе (обычно это так), говорят об архитектуре **cc-NUMA** (cache-coherent NUMA) |
| **Примеры** | HP [HP 9000 V-class](https://parallel.ru/computers/computers.html" \l "exemplar) в SCA-конфигурациях, SGI [Origin2000](https://parallel.ru/computers/computers.html#origin2000), Sun [HPC 10000](https://parallel.ru/computers/computers.html#starfire), IBM/Sequent [NUMA-Q 2000](https://parallel.ru/computers/computers.html#numa-q), SNI [RM600](https://parallel.ru/computers/computers.html#rm600). |
| **Масштабируемость** | Масштабируемость NUMA-систем ограничивается объемом адресного пространства, возможностями аппаратуры поддежки когерентности кэшей и возможностями операционной системы по управлению большим числом процессоров. На настоящий момент, максимальное число процессоров в NUMA-системах составляет 256 (Origin2000). |
| **Операционная система** | Обычно вся система работает под управлением единой ОС, как в [SMP](https://parallel.ru/computers/classes.html#smp). Но возможны также варианты динамического "подразделения" системы, когда отдельные "разделы" системы работают под управлением разных ОС (например, Windows NT и UNIX в NUMA-Q 2000). |
| **Модель программирования** | Аналогично [SMP](https://parallel.ru/computers/classes.html#smp). |

### 

### Таблица 4. Параллельные векторные системы (PVP)

|  |  |
| --- | --- |
| **Архитектура** | Основным признаком PVP-систем является наличие специальных векторно-конвейерных процессоров, в которых предусмотрены команды однотипной обработки векторов независимых данных, эффективно выполняющиеся на конвейерных функциональных устройствах.  Как правило, несколько таких процессоров (1-16) работают одновременно над общей памятью (аналогично [SMP](https://parallel.ru/computers/classes.html#smp)) в рамках многопроцессорных конфигураций. Несколько таких узлов могут быть объединены с помощью коммутатора (аналогично [MPP](https://parallel.ru/computers/classes.html#mpp)). |
| **Примеры** | NEC SX-4/[SX-5](https://parallel.ru/computers/computers.html#sx-5), линия векторно-конвейерных компьютеров CRAY: от CRAY-1, CRAY J90/[T90](https://parallel.ru/computers/computers.html#crayt90), [CRAY SV1](https://parallel.ru/computers/computers.html#sv1), [CRAY X1](https://parallel.ru/computers/computers.html#X1), серия Fujitsu [VPP](https://parallel.ru/computers/computers.html#vpp700). |
| **Модель программирования** | Эффективное программирование подразумевает векторизацию циклов (для достижения разумной производительности одного процессора) и их распараллеливание (для одновременной загрузки нескольких процессоров одним приложением). |

### Таблица 5. Кластерные системы

|  |  |
| --- | --- |
| **Архитектура** | Набор рабочих станций (или даже ПК) общего назначения, используется в качестве дешевого варианта [массивно-параллельного](https://parallel.ru/computers/classes.html#mpp) компьютера. Для связи узлов используется одна из стандартных сетевых технологий (Fast/Gigabit Ethernet, Myrinet) на базе шинной архитектуры или коммутатора.  При объединении в кластер компьютеров разной мощности или разной архитектуры, говорят о **гетерогенных** (неоднородных) кластерах.  Узлы кластера могут одновременно использоваться в качестве пользовательских рабочих станций. В случае, когда это не нужно, узлы могут быть существенно облегчены и/или установлены в стойку. |
| **Примеры** | [NT-кластер](https://parallel.ru/news/ncsa_cluster.html) в NCSA, [Beowulf](https://parallel.ru/computers/reviews/beowulf.html)-кластеры. |
| **Операционная система** | Используются стандартные для рабочих станций ОС, чаще всего, свободно распространяемые - Linux/FreeBSD, вместе со специальными средствами поддержки параллельного программирования и распределения нагрузки. |
| **Модель программирования** | Программирование, как правило, в рамках модели передачи сообщений (чаще всего - [MPI](https://parallel.ru/tech/tech_dev/mpi.html)). Дешевизна подобных систем оборачивается большими накладными расходами на взаимодействие параллельных процессов между собой, что сильно сужает потенциальный класс решаемых задач. |

Лекция 5

**Организация работы и функционирование процессора. Микропроцессоры типа CISC,RISC, MISC.Характеристики и структура микропроцессора. Устройство управления, АЛУ, микропроцессорная память: назначение, упрощенные функциональные схемы.**

**Организация работы процессора:**

**Последовательная обработка.**

Во время процесса процессор считывает последовательность команд, содержащихся в памяти, и исполняет их. Очерёдность считывания команд изменяется в случае, если процессор считывает команду перехода — тогда адрес следующей команды может оказаться другим. Другим примером изменения процесса может служить случай получения команды останова или переключение в режим обработки аппаратного прерывания.

Скорость перехода от одного этапа цикла к другому определяется тактовым генератором. Тактовый генератор вырабатывает импульсы, служащие ритмом для центрального процессора.

**Параллельная обработка.**

В основу было положено понятие потока, под которым понимается последовательность элементов, команд или данных, обрабатываемая процессором. Соответствующая система классификации основана на рассмотрении числа потоков инструкций и потоков данных и описывает четыре архитектурных класса:

·      SISD (single instruction stream / single data stream) - одиночный поток команд и одиночный поток данных

·      MISD (multiple instruction stream / single data stream) - множественный поток команд и одиночный поток данных.

·      SIMD (single instruction stream / multiple data stream) - одиночный поток команд и множественный поток данных.

·      MIMD (multiple instruction stream / multiple data stream) - множественный поток команд и множественный поток данных

В основе параллельного компьютера лежит идея использования для решения одной задачи нескольких процессоров, работающих сообща, причем процессоры могут быть как скалярными, так и векторными.

**Конвейерная обработка.**

Конвейерная архитектура (pipelining) была введена в центральный процессор с целью повышения быстродействия. Обычно для выполнения каждой команды требуется осуществить некоторое количество однотипных операций, например: выборка команды из ОЗУ, дешифрация команды, адресация операнда в ОЗУ, выборка операнда из ОЗУ, выполнение команды, запись результата в ОЗУ. Каждую из этих операций сопоставляют одной ступени конвейера. Например, конвейер микропроцессора с архитектурой MIPS-I содержит четыре стадии:

·      получение и декодирование инструкции (Fetch)

·      адресация и выборка операнда из ОЗУ (Memory access)

·      выполнение арифметических операций (Arithmetic Operation)

·      сохранение результата операции (Store)

После освобождения k-й ступени конвейера она сразу приступает к работе над следующей командой. Если предположить, что каждая ступень конвейера тратит единицу времени на свою работу, то выполнение команды на конвейере длиной в n ступеней займёт n единиц времени, однако в самом оптимистичном случае результат выполнения каждой следующей команды будет получаться через каждую единицу времени

**Суперскалярная обработка.**

Способность выполнения нескольких машинных инструкций за один такт процессора. Появление этой технологии привело к существенному увеличению производительности.

Основными параметрами процессоров являются:

· тактовая частота,

· разрядность,

· рабочее напряжение,

· коэффициент внутреннего умножения тактовой частоты,

· размер кэш памяти.

Тактовая частота определяет количество элементарных операций (тактов), выполняемые процессором за единицу времени. Тактовая частота современных процессоров измеряется в МГц (1 Гц соответствует выполнению одной операции за одну секунду, 1 МГц=106 Гц). Чем больше тактовая частота, тем больше команд может выполнить процессор, и тем больше его производительность. Первые процессоры, которые использовались в ПК работали на частоте 4,77 МГц, сегодня рабочие частоты современных процессоров достигают отметки в 2 ГГц (1 ГГц=103 МГц).

Разрядность процессора показывает, сколько бит данных он может принять и обработать в своих регистрах за один такт. Разрядность процессора определяется разрядностью командной шины, то есть количеством проводников в шине, по которой передаются команды. Современные процессоры семейства Intel являются 32-разрядными.

Рабочее напряжение процессора обеспечивается материнской платой, поэтому разным маркам процессоров отвечают разные материнские платы. Рабочее напряжение процессоров не превышает 3 В. Снижение рабочего напряжения разрешает уменьшить размеры процессоров, а также уменьшить тепловыделение в процессоре, что разрешает увеличить его производительность без угрозы перегрева.

Коэффициент внутреннего умножения тактовой частоты - это коэффициент, на который следует умножить тактовую частоту материнской платы, для достижения частоты процессора. Тактовые сигналы процессор получает от материнской платы, которая из чисто физических причин не может работать на таких высоких частотах, как процессор. Для получения более высоких частот в процессоре происходит внутреннее умножение на коэффициент 4, 4.5, 5 и больше.

**Архитектура процессоров (CISC, RISC, MISC)**

[Микропроцессор](https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%80%D0%BE%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80) — это устройство, представляющее собой одну или несколько больших интегральных схем (БИС), выполняющих функции процессора ЭВМ. Классическое вычислительное устройство состоит из арифметического устройства (АУ), устройства управления (УУ), запоминающего устройства (ЗУ) и устройства ввода-вывода (УВВ).

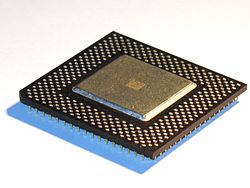
[](https://ru.wikiversity.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:UpSL3A2.JPG)

Рисунок 6 - Intel Celeron 400 Socket 370 в пластиковом корпусе [PPGA](https://ru.wikipedia.org/wiki/PPGA), вид сверху.

Существуют процессоры различной архитектуры.

[CISC](https://ru.wikipedia.org/wiki/CISC) (англ. Complex Instruction Set Computing) — концепция проектирования процессоров, которая характеризуется следующим набором свойств:

* большим числом различных по формату и длине команд;
* введением большого числа различных режимов адресации;
* обладает сложной кодировкой инструкции.

Процессору с архитектурой CISC приходится иметь дело с более сложными инструкциями неодинаковой длины. Выполнение одиночной CISC-инструкции может происходить быстрее, однако обрабатывать несколько таких инструкций параллельно сложнее.

Облегчение отладки программ на ассемблере влечет за собой загромождение узлами микропроцессорного блока. Для повышения быстродействия следует увеличить тактовую частоту и степень интеграции, что вызывает необходимость совершенствования технологии и, как следствие, более дорогого производства.

*Достоинства архитектуры CISC:*

* Компактность наборов инструкций уменьшает размер программ и уменьшает количество обращений к памяти.
* Наборы инструкций включают поддержку конструкций высокоуровневого программирования.

*Недостатки архитектуры CISC*:

* Нерегулярность потока команд.
* Высокая стоимость аппаратной части.
* Сложности с распараллеливанием вычислений.

RISC (Reduced Instruction Set Computing). Процессор с сокращенным набором команд. Система команд имеет упрощенный вид. Все команды одинакового формата с простой кодировкой. Обращение к памяти происходит посредством команд загрузки и записи, остальные команды типа регистр-регистр. Команда, поступающая в CPU, уже разделена по полям и не требует дополнительной дешифрации.

Часть кристалла освобождается для включения дополнительных компонентов. Степень интеграции ниже, чем в предыдущем архитектурном варианте, поэтому при высоком быстродействии допускается более низкая тактовая частота. Команда меньше загромождает ОЗУ, CPU дешевле. Программной совместимостью указанные архитектуры не обладают. Отладка программ на RISC более сложна. Данная технология может быть реализована программно-совместимым с технологией CISC (например, суперскалярная технология).

Поскольку RISC-инструкции просты, для их выполнения нужно меньше логических элементов, что в конечном итоге снижает стоимость процессора. Но большая часть программного обеспечения сегодня написана и откомпилирована специально для CISC-процессоров фирмы Intel. Для использования архитектуры RISC нынешние программы должны быть перекомпилированы, а иногда и переписаны заново.

*Достоинства архитектуры RISC*

* Снижение нерегулярности потока команд
* Обогащение пространственным параллелизмом

*Недостатки архитектуры RISC*

* Каждое действие выполняется в 1 такт

MISC (Multipurpose lnstruction Set Computer). Элементная база состоит из двух частей, которые либо выполнены в отдельных корпусах, либо объединены. Основная часть – RISC CPU, расширяемый подключением второй части – ПЗУ микропрограммного управления. Система приобретает свойства CISC. Основные команды работают на RISC CPU, а команды расширения преобразуются в адрес микропрограммы. RISC CPU выполняет все команды за один такт, а вторая часть эквивалентна CPU со сложным набором команд. Наличие ПЗУ устраняет недостаток RISC, выраженный в том, что при компиляции с языка высокого уровня микрокод генерируется из библиотеки стандартных функций, занимающей много места в ОЗУ. Поскольку микропрограмма уже дешифрована и открыта для программиста, то времени выборки из ОЗУ на дешифрацию не требуется.

**Характеристики и структура микропроцессора.**

Упрощенно структуру МП можно представить в следующем виде (Рисунок 2).

Основными узлами МП являются:

* операционное устройство;
* управляющее устройство;
* счетчик команд;
* регистр команд;
* регистры 1...п;
* регистр признаков.

*Операционное устройство* (ОУ) — это устройство, в котором выполняются операции. Оно включает в себя следующие узлы: регистры, сумматоры, каналы передачи информации, мультиплексоры для коммутации каналов, шифраторы, дешифраторы и т.д. Основу ОУ составляет *арифметико-логическое устройство* (АЛУ), которое предназначено для выполнения предусмотренных МП арифметических и логических операций.

*Управляющее устройство* (УУ) координирует действия узлов ОУ. Оно вырабатывает в некоторой временной последовательности управляющие сигналы, пол действием которых в узлах ОУ выполняются требуемые действия.

*Счетчик команд* (СК) (программный счетчик) — это регистр, который хранит адрес следующей команды, которая должна быть выполнена вслед за предыдущей. В процессе исполнения команды содержимое СК автоматически инкрементируется (увеличивается) после выборки каждого байта команды. Таким образом, в любой момент времени его содержимое представляет собой адрес очередной ячейки памяти, где может храниться либо следующая команда, либо дополнительные данные, привлекаемые МП для исполнения данной команды. СК является программно доступным. На содержимое СК пользователь может повлиять только с помощью команд, изменяющих последовательное выполнение программы (например, команд безусловного перехода), а также с помощью некоторых специальных команд.

*Регистр команд* (РК) — регистр, который воспринимает код текущей команды (первый байт кода команды, код операции) с шины данных и осуществляет его хранение в течение всего времени ее исполнения. РК не доступен программисту. После выполнения очередной команды в РК автоматически записывается код следующей команды из ячейки оперативной памяти, адрес которой содержится в счетчике команд.

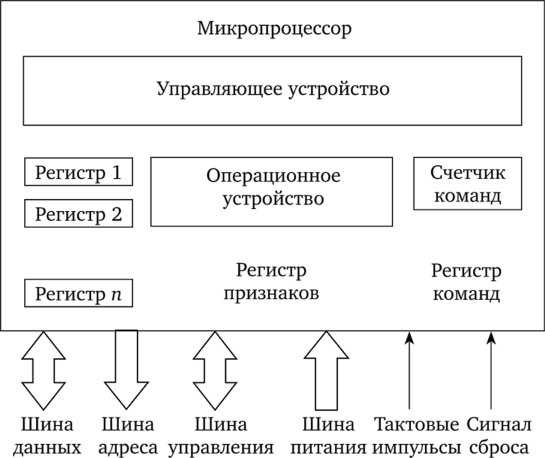


Рисунок 7 – Упрощенная структура МП

*Регистры l...n* — регистры общего назначения, предназначенные для оперативного хранения данных и промежуточных результатов, участвующих в процессе выполнения программы.

*Регистр признаков* (РП) (регистр флагов, регистр состояния) — регистр, представляющий собой набор триггеров (индикаторов, флагов). РП дает информацию об особенностях результата последней операции, выполненной АЛУ. Каждый триггер РП предназначен для хранения одного признака результата выполнения операции.

По *шине данных* передаются двоичные сигналы, соответствующие кодам данных и команд управляющих программ.

*Шина адреса* служит для указания места (адреса) расположения данных. По ней МП передает двоичный код соответствующей ячейки памяти (откуда взять или куда записать двоичный код, передаваемый по шине данных).

По *шине управления* передаются сигналы, соответствующие направлению передачи информации, т.е. откуда нужно считать данные (источник) и куда их нужно записать (приемник).

Причем существуют как внутренние шины МП, так и системные шины МПС. Указанные шины согласованы между собой буферными схемами, расположенными также в МП.

Укрупненная функциональная схема УУ приведена на рис. 11.88.

*Дешифратор команд* (ДК) — логический блок, выбирающий в соответствии с поступающим из регистра команд кодом операции (КОП) один из множества имеющихся у него выходов.

*Постоянное запоминающее устройство микропрограмм* хранит в своих ячейках управляющие сигналы (импульсы), необходимые для выполнения в блоках МП операций обработки информации. Импульс по выбранному ДК в соответствии с кодом операции считывает из ПЗУ микропрограмм необходимую последовательность управляющих сигналов.

**Структура базового микропроцессора**

Функционально микропроцессор можно разделить на две части (Рисунок 3):

-  операционную (устройство управления и устройство обработки данных), содержащую арифметико-логическое устройство, микропроцессорную память (за исключением сегментных регистров), блок микропрограммного управления, объединенных в устройство обработки данных, и устройство управления;

-  интерфейсную (или устройство связи с магистралью), содержащую блок сегментных регистров микропроцессорной памяти, блок регистров команд (регистры памяти для хранения кодов команд, выполняемых в ближайшие такты работы) и сумматор адреса.

Устройство обработки данных предназначено для выполнения команд. Устройство управления обеспечивает синхронизацию работы устройств микропроцессора, выработку управляющих сигналов и сигналов состояния для обмена с другими устройствами, анализ и соответствующую реакцию на сигналы других устройств ЭВМ.

Устройство связи с магистралью обеспечивает формирование физического адреса памяти и адреса внешнего устройства, выбор команд из памяти, обмен данными с запоминающими устройствами, внешними устройствами, другими процессорами по магистрали.

Обе части микропроцессора работают параллельно, причем интерфейсная часть опережает операционную, так что выборка очередной команды из памяти (ее запись в блок регистров команд и предварительный анализ) выполняется во время выполнения операционной частью предыдущей команды. Современные микропроцессоры имеют несколько групп регистров в интерфейсной части, работающих с различной степенью опережения, что позволяет выполнить операции в конвейерном режиме. Такая организация микропроцессора позволяет существенно повысить его эффективное быстродействие.

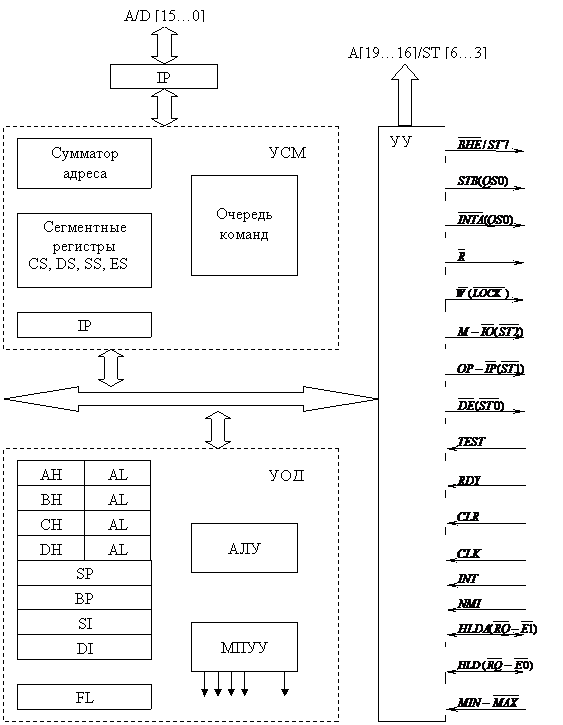


Рисунок 8 – Упрощенная структурная схема микропроцессора (на примере i8086)

На внешних выводах микропроцессора широко используется принцип мультиплексирования сигналов – передача разных сигналов по общим линиям с разделением времени. Кроме того, одни и те же выводы могут использоваться для передачи разных сигналов в зависимости от режима (минимальный или максимальный)

Лекция 6

**Системы команд процессора. Регистры процессора: сущность, назначение, типы. Параллелизм вычислений. Конвейеризация вычислений. Суперскаляризация. Матричные и векторные процессоры.**

Система команд, выполняемых процессором, представляет собой нечто подобное таблице истинности логических элементов или таблице режимов работы более сложных логических микросхем. То есть она определяет логику работы процессора и его реакцию на те или иные комбинации внешних событий.

**Система команд -** это набор допустимых для данного процессора управляющих кодов и способов адресации данных. Система команд жестко связана с конкретным типом процессора, поскольку определяется аппаратной структурой блока дешифрации команд, и обычно не обладает переносимостью на другие типы процессоров.

В общем случае система команд процессора включает в себя следующие четыре основные группы команд:

* команды пересылки данных;
* арифметические команды;
* логические команды;
* команды переходов.

*Команды пересылки данных* не требуют выполнения никаких операций над операндами. Операнды просто пересылаются (точнее, копируются) из источника (Source) в приемник (Destination). Источником и приемником могут быть внутренние регистры процессора, ячейки памяти или устройства ввода/вывода. АЛУ в данном случае не используется.

*Арифметические команды* выполняют операции сложения, вычитания, умножения, деления, увеличения на единицу (инкрементирования), уменьшения на единицу (декрементирования) и т.д. Этим командам требуется один или два входных операнда. Формируют команды один выходной операнд.

*Логические команды* производят над операндами логические операции, например, логическое И, логическое ИЛИ, исключающее ИЛИ, очистку, инверсию, разнообразные сдвиги (вправо, влево, арифметический сдвиг, циклический сдвиг). Этим командам, как и арифметическим, требуется один или два входных операнда, и формируют они один выходной операнд.

*Команды переходов* предназначены для изменения обычного порядка последовательного выполнения команд. С их помощью организуются переходы на подпрограммы и возвраты из них, всевозможные циклы, ветвления программ, пропуски фрагментов программ и т.д. Команды переходов всегда меняют содержимое счетчика команд. Переходы могут быть условными и безусловными. Именно эти команды позволяют строить сложные алгоритмы обработки информации.

В соответствии с результатом каждой выполненной команды устанавливаются или очищаются биты регистра состояния процессора (PSW). Но надо помнить, что не все команды изменяют все имеющиеся в PSW флаги. Это определяется особенностями каждого конкретного процессора.У разных процессоров системы команд существенно различаются, но в основе своей они очень похожи. Количество команд у процессоров также различно. Например, у упоминавшегося уже процессора МС68000 всего 61 команда, а у процессора 8086 — 133 команды. У современных мощных процессоров количество команд достигает нескольких сотен. В то же время существуют процессоры с сокращенным набором команд (так называемые RISC-процессоры), в которых за счет максимального сокращения количества команд достигается увеличение эффективности и скорости их выполнения.

Команды пересылки данных.

Команды пересылки данных занимают очень важное место в системе команд любого процессора.

Они выполняют следующие важнейшие функции:

* загрузка (запись) содержимого во внутренние регистры процессора;
* сохранение в памяти содержимого внутренних регистров процессора;
* копирование содержимого из одной области памяти в другую;
* запись в устройства ввода/вывода и чтение из устройств ввода/вывода.

В некоторых процессорах (например, Т-11) все эти функции выполняются одной единственной командой MOV (для байтовых пересылок — MOVB) но с различными методами адресации операндов.

**Регистр процессора** — сверхбыстрая оперативная память внутри процессора, предназначенная прежде всего для хранения промежуточных результатов вычисления.

Регистр представляет собой цифровую электронную схему, служащую для временного хранения двоичных чисел. В процессоре имеется значительное количество регистров, большая часть которых используется самим процессором и недоступна программисту (например, при выборке из памяти очередной команды она помещается в регистр команд, и программист обратиться к этому регистру не может). Имеются также регистры, которые в принципе программно доступны, но обращение к ним осуществляется из программ операционной системы (например, управляющие регистры и теневые регистры дескрипторов сегментов). Этими регистрами пользуются в основном разработчики операционных систем.

Доступ к значениям, хранящимся в регистрах, как правило, в несколько раз быстрее, чем доступ к ячейкам оперативной памяти (даже если кеш-память содержит нужные данные), но объём оперативной памяти намного превосходит суммарный объём регистров (объём среднего модуля оперативной памяти сегодня составляет 1-4 Гб[4], суммарная «ёмкость» регистров общего назначения/данных для процессора Intel 80x86 16 битов \* 4 = 64 бита (8 байт)).

По типу ***приёма*** и выдачи информации различают 2 типа регистров:

* С последовательным приёмом и выдачей информации — сдвиговые регистры.
* С параллельным приёмом и выдачей информации — параллельные регистры.

По назначению регистры различаются на:

* аккумулятор — используется для хранения промежуточных результатов арифметических и логических операций и инструкций ввода-вывода;
* флаговые — хранят признаки результатов арифметических и логических операций;
* общего назначения — хранят операнды арифметических и логических выражений, индексы и адреса;
* индексные — хранят индексы исходных и целевых элементов массива;
* указательные — хранят указатели на специальные области памяти (указатель текущей операции, указатель базы, указатель стека);
* сегментные — хранят адреса и селекторы сегментов памяти;
* управляющие — хранят информацию, управляющую состоянием процессора, а также адреса системных таблиц.

**Параллелизм и конвейеризация вычислений**

Параллелизм и конвейеризация имеют одинаковые цели — повышение производительности вычислительных систем, оба подхода предполагают достижение этой цели за счет “размножения” аппаратных средств (избыточности аппаратуры вычислительных систем), однако организация вычислительного процесса в них достаточно различается, чтобы оправдать их сравнение. Приведенная таблица отражает наиболее существенные различия между этими подходами

Параллелизм можно определить как возможность одновременного выполнения разных и математических или служебных операций. Распараллеливание нахождение алгоритма решения задачи, использующего параллелизм и реализация этого алгоритма в ВС.

Параллельные ВС. это многопроцессорные ВС., в которых параллелизм используется для повышения производительности при решении задач за счет одновременного выполнения разных операций на разных процессорах или обрабатывающих устройствах одного процессора.

Проблемы распараллеливания:

Некоторые задачи очень хорошо распараллеливаются (например, многие задачи обработки матриц), некоторые - очень плохо.

Решение вопросов взаимодействия параллельных процессоров (например, обмен с памятью, межпроцессорный обмен, синхронизация и др.)

Возможность реализации выбранного распараллеленного алгоритма на данной параллельной структуре.

Конвейеризация вычислений - разбиение вычислений на последовательные этапы с целью реализации этих этапов на отдельных ступенях конвейера для повышения производительности.

Конвейер - устройство, состоящее из N последовательно соединенных частей

(ступеней конвейера), каждая из которых выполняет очередной шаг вычислений за время t (такт конвейера).

Часто конвейерный режим используют в векторных ВС. Векторные ЭВМ и ВС выполняют не только скалярные операции (над числами), но и операции над векторами (массивами чисел), при этом векторные операции выполняются параллельно для всех элементов вектора, либо - на конвейере.

Выигрыш в производительности достигается за счет сокращения расходов на управление, а также за счет снижения универсальности устройства.

Основные проблемы организации вычислений в системах с перестраиваемой структурой связаны с обеспеченном параллелизма вычислений и распределенного децентрализованного управлении процессами и ресурсами. Эти проблемы разработаны только в первом приближении, и известные способы организации параллельных вычислений в распределенных системах с децентрализованным управлением еще не достигли необходимого уровня универсальности и формализации.

Параллельная обработка задач, т. е. мультипрограммный режим функционирования системы, обеспечиваемся достаточно простыми средствами. После ввода задания в систему модуль, принявший задание, посылает через коммутационное поле запрос на поиск свободного обрабатывающего модуля. Когда свободный модуль найден, ему посылается задание, определяющее имена наборов данных, в которых размещается программа, исходные данные и в которые должны быть помещены результаты вычислений. Из задания и программы модуль получает сведения о ресурсах, необходимых для выполнения задания: емкости операционной памяти, числе процессоров и неразделяемых наборах данных. Модуль закрепляет за собой необходимые ресурсы, и после обеспечения задания требуемыми ресурсами инициируется процесс выполнения задачи. По завершении обработки ресурсы освобождаются и в дальнейшем предоставляются очередным заданиям. Число процессов, реализуемых параллельно, определяется числом модулей, входящих в состав системы, и при наличии очереди заданий производительность системы пропорциональна числу модулей.

Параллельные программы строятся традиционными способами: выделением подзадач и ветвей программы, операций над векторами и матрицами и организацией конвейерной обработки данных. Наиболее просто реализуются вычисления с выделением подзадач н параллельных ветвей. При возникновении ветви в ведущей программе модуль посылает запрос на поиск свободного модуля, в который загружается программа и данные ветви, и ветвь выполняется как самостоятельная задача, по завершении которой в ведущий модуль отсылаются результаты обработки. Параллельные вычисления по конвейерной и матричной схемам организуются за счет создания соответствующих конфигураций связей между модулями – линейных (кольцевых) и матричных структур. Построение таких структур в многомодульных системах, в которых часть модулей занята выполнением ранее созданных задач, является пока нерешенной проблемой. Обычно для матричных вычислений в систему встраивает в качестве специального модуля матричный процессор, обеспечивающий высокопроизводительную обработку блоков данных.

В вычислительной системе с перестраиваемой структурой должно быть реализовано распределенное (децентрализованное) управление ресурсами. Это означает, что в системе не должно быть выделенного модуля (даже многократно зарезервированного), на который возложена задача централизованною управления функционированием системы. Распределенное управление основано на согласованной работе всех модулей системы, каждый из которых реализует одинаковый набор правил управления, обеспечивающий эффективное использование всех ресурсов системы. Распределенное управление повышает надежность системы, поскольку каждый модуль способен реализовать управление ресурсами и процессами, и одновременно повышает производительность системы, так как управляющие решения формируются без затрат времени на сбор информации о состоянии всех элементов системы (а за это время ситуация в системе может существенно измениться).

Конвейеризация вычислений - разбиение вычислений на последовательные этапы с целью реализации этих этапов на отдельных ступенях конвейера для повышения производительности.

Конвейер - устройство, состоящее из N последовательно соединенных частей (ступеней конвейера), каждая из которых выполняет очередной шаг вычислений за время t (такт конвейера). Таким образом, для решения задачи, требующей N шагов, потребуется время, равное t N, однако производительность конвейера может быть достаточно велика, поскольку освобождающиеся ступени могут заполняться новыми данными. В результате на каждом такте конвейер может выдавать очередной результат.

**Процессоры с несколькими линиями конвейера получили название суперскалярных** . Pentium — первый суперскалярный процессор Intel. Здесь две линии, что позволяет ему при одинаковых частотах быть вдвое производительней i80486, выполняя сразу две инструкции за такт.

Во многих вычислительных системах, наряду с конвейером команд, используются конвейеры данных. Сочетание этих двух конвейеров дает возможность достичь очень высокой производительности на определенных классах задач, особенно если используется несколько различных конвейерных процессоров, способных работать одновременно и независимо друг от друга.

*Операции над вещественными числами (с плавающей запятой)*

*Сопроцессоры.* Для расширения вычислительных возможностей центрального процессора — выполнения арифметических операций, вычисления основных математических функции (тригонометрических, показательных, логарифмических) и т. д. — в состав ЭВМ добавляется математический сопроцессор. Применение сопроцессора повышает производительность вычислений в сотни раз. В разных поколениях процессоров он назывался по-разному — FPU (Floating Point Unit — блок чисел/операций с плавающей точкой — БПЗ) или NPX (Numeric Processor extension — числовое расширение процессора).

Для процессоров 386 и ниже сопроцессор был отдельной микросхемой, подключаемой к локальной нише основного процессора. В любом случае сопроцессор исполняет только свои специфические команды, а всю работу по декодированию инструкций и доставке данных осуществляет ЦП.

Блоки операций с плавающей запятой. С программной точки зрения сопроцессор и процессор выглядят как единое целое. В современных (486+) процессорах БПЗ располагается на одном кристалле с центральным процессором.

**Матричные и векторные процессоры**

Матричные и векторные процессоры предназначены для обработки изображений, матриц, массивов данных.

Матричный процессор – имеет архитектуру, рассчитанную на обработку числовых массивов. Архитектура включает в себя матрицу процессорных элементов, например 64x64, работающих одновременно. Постпроцессор предназначен для реализации специальных функций, например управление базой данных.

Векторный процессор – обеспечивает параллельное выполнение операций над массивами данных, векторами. Он характеризуется специальной архитектурой, выполненной на группе параллельно работающих процессорных элементах.

Векторная обработка увеличивает производительность за счет обработки целого набора данных (вектора) одной командой, в различие скалярных, которые обрабатывают отдельные элементы этого массива. При работе в векторном режиме векторные процессоры обрабатывают данные практически параллельно, что делает их в несколько раз быстрыми, чем в скалярном режиме. Максимальная скорость передачи данных в векторном формате составляет 64 Гбайт \ с, что почти в 2 раза больше, чем в скалярных машинах. В настоящее время созданы однокристальные векторно-конвейерные процессоры, такие, как SX-6. Основными компонентами микропроцессора являются скалярный процессор и 8 векторных, суммарная производительность которых составляет 64 Гфлопс. Процессоры NEC, HITACHI.

Лекция 7

**Системные платы. Виды, характеристики, форм-факторы. Корпуса ПК. Блоки питания. Основные шины расширения. Прямой доступ к памяти. Прерывания. Драйверы.**

Системная плата – это микросхема содержащая  набор компонент электроники, с помощью которых  осуществляется взаимодействие узлов(устройств) компьютера. Системная  плата покрыта сетью медных проводников-дорожек, по которым подается электропитание и осуществляется передача данных между узлами(устройствами) компьютера.

Системная плата (англ. motherboard, MB, матери́нская пла́та, также используется название англ. mainboard — главная плата; на компьютерном жаргоне — мама, мать, материнка) — сложная многослойная печатная плата, на которой устанавливаются основные компоненты персонального компьютера либо сервера начального уровня (центральный процессор, контроллер ОЗУ и собственно ОЗУ, загрузочное ПЗУ, контроллеры базовых интерфейсов ввода-вывода).(Wikipedia)

Системная плата является основной компонентой компьютера. Приоритет системной платы не случаен, так как она:  
•  во-первых, обеспечивает связь между компонентами;  
•  во-вторых, отвечает за функционирование устройств, осуществляя передачу питания для всех  элементов;  
•  в-третьих, она контролирует состояния важнейших узлов.

Таким образом, материнская плата является своеобразным фундаментом для будущей системы, задавая ей основные характеристики, возможности наращивания мощности и даже сроки морального износа.   Основными вопросами при выборе материнской платы является:  
•   архитектура системной платы;  
•   основные разъемы и поддерживаемые интерфейсы;  
•    технологии, которыми она «напичкана».

**1. Архитектура материнских плат**

Архитектура системной платы определяется разработчиком (производителем) материнской платы. При её конструировании используется понятие форм-фактора (form factor).

Форм-фактор представляет собой физические параметры платы и определяет тип корпуса, в котором она может быть установлена. Форм-факторы системных плат могут быть стандартными (т.е. взаимозаменяемыми) или нестандартными. Нестандартные форм-факторы, к сожалению, являются препятствием для модернизации компьютера, поэтому от их использования лучше отказаться.

Форм-фактор, или типоразмер системной платы, определяет ее габариты, параметры электропитания, расположение монтажных элементов (отверстий, клипсов), размещение разъемов различных интерфейсов и т. д

Форм-фактор (техника) — стандарт технического изделия, описывающий некоторую совокупность его технических параметров.

Существуют множество форм-факторов, среди них, наиболее известными являются:  
*устаревшие:*  
•   AT(полноразмерная, Baby-AT);  
•    LPX(mini-);  
*современные:*  
•    NLX  
•    ATX (micro-, flex-, mini-);  
•    BTX (micro-, pico)  
•    WTX  
•    CEB  
•    ITX(mini-, nano-, pico)  
*прочие:*  
•    независимые конструкции (разработки компаний Compaq, Packard Bell, HewlettPackard, портативные/мобильные системы и т.д.).

**AT (полноразмерный, Baby-AT)**

Форм фактор AT (англ. Advanced Technology) — первый широко использовавшийся форм-фактор в персональных компьютерах. Форм-фактор АТ делится на две, отличающиеся по размеру модификации – AT(полноразмерная) и Baby AT.

Форм-фактор AT(полноразмерный) был создан IBM в 1984 году и пришёл на смену ранее существовавшим форм-факторам PC и XT. Размер полноразмерной AT платы достигает до 12″ в ширину. При монтаже такой платы, в большинстве корпусов мешал отсек для дисководов, жестких дисков и блок питания. Кроме того, расположение компонентов платы на большом расстоянии друг от друга вызывало проблемы при работе на больших тактовых частотах.

В 1985 IBM представила уменьшенную версию форм-фактора — Baby AT (аналогичные уменьшенные версии форм-факторов более новых стандартов выходили с префиксом micro-). Размер платы Baby AT 8.5″ в ширину и 13″ в длину, некоторые производители уменьшали размеры платы для экономии материала или по каким-то другим причинам.  Материнские платы размера Baby AT могли быть установлены практически в любой корпус, за исключением корпусов уменьшенной высоты и Slimline. Именно поэтому и получили наибольшее распространение. Стандарт был очень популярен вплоть до 1997, когда на смену Baby AT пришёл форм-фактор ATX  
Все AT платы имеют общие черты. Для крепления платы в корпусе в плате сделаны три ряда отверстий, которые должны соответствовать отверстиям в корпусе при монтаже. Почти все имеют последовательные и параллельные порты, присоединяемые к материнской плате через соединительные планки. Они также имеют один 5-контактный(DIN) разъём клавиатуры, впаянный на плату в задней части. Гнездо под процессор устанавливается на передней стороне платы. Слоты SIMM и DIMM находятся в различных местах, хотя почти всегда они расположены в верхней части материнской платы.

Недостатками форм-фактора AT и Baby AT:  
•    чаще всего на Baby AT платах все коннекторы собраны в одном месте, в результате чего либо кабели от коммуникационных портов тянутся практически через всю материнскую плату к задней части корпуса, либо от портов IDE и FDD — к передней.  
•    гнезда для модулей памяти, заезжающие чуть ли не под блок питания. При ограниченности свободы действий внутри весьма небольшого пространства корпуса MiniTower.  
•    неудачно решен вопрос с охлаждением — воздух не поступает напрямую к самой нуждающейся в охлаждении части системы — процессору.

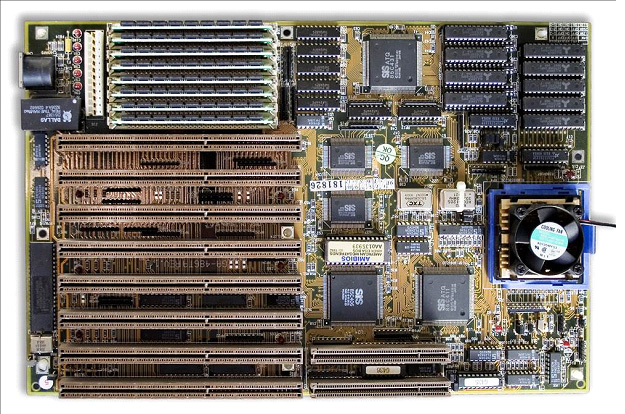
[](https://advteach.files.wordpress.com/2012/02/baby-at.jpg)

Рисунок 9 – Системная плата форм фактора Baby AT

**LPX**

LPX – форм фактор материнской платы с размером 9х11-13″ (229х279-330 мм), разработан Western Digital в 1987 году. Предназначался для использования в корпусах Slimline или Low-profile. Платы расширения устанавливались параллельно системной плате, посредством переходника с повернутыми на 90° разъемами. Это позволило заметно уменьшить высоту корпуса, поскольку обычно именно высота карт расширения влияет на этот параметр. Расплатой за компактность стало максимальное количество подключаемых карт  не более 3 штук. Ещё одно нововведением был интегрированный в материнскую плату видеочип. Платформы форм-фактора

Размеры плат форм-фактора ITX:  
•    LPX – 22,9 × 27,9-33,0см  
•    Mini-ITX – 20,3-22,9 × 25,4-27,9 см

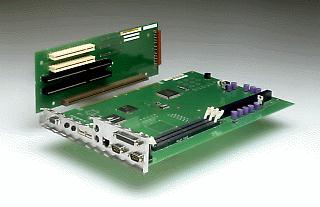
[](https://advteach.files.wordpress.com/2012/02/lpx1.jpg)

Рисунок 10 – Системная плата форм фактора LPX

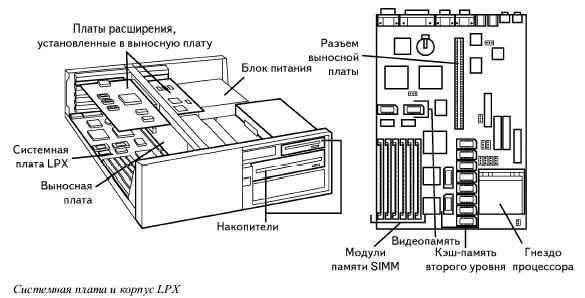
[](https://advteach.files.wordpress.com/2012/02/lpx2.jpg)

Рисунок 11 - Схема системной платы форм фактора LPX

     LPX широкого распространения не получили и после появления NLX, LPX начал вытесняться этим форм-фактором.

**NLX**

Конструкция NLX предложена в ноябре 1996 года компанией Intel и разработана совместно с IBM, DEC и другими производителями системных плат с низким профилем. Версия 1.2 была выполнена в марте 1997 году. В апреле 1998 появилась версия 1.8. Форм фактор NLX стал использоваться в корпоративных системах Slimline таких компаний, как Compaq, HP, Toshiba и др. Корпорация Intel продвигала форм-фактор NLX как базу для построения компактных компьютеров.

Системная плата согласно этой спецификации разделена на две части. В специальный разъем (получивший название NLX Riser Connector), непосредственно примыкающий к блоку питания, вставляется процессорная плата (содержит процессор, BIOS, слоты для модулей оперативной памяти). Кроме контактов питания разъем имеет информационную (системную) шину. Другая плата (названная Riser card) установлена в корпусе компьютера стационарно (то есть является частью компьютерной системы) и может иметь слоты интерфейсов PCI, USB, IEEE1394 и любых других имеющихся и перспективных стандартов. Таким образом, после установки процессорная плата автоматически оказывается подключенной к питанию и к шинам интерфейсов.

Многочисленные усовершенствования, отличающие форм фактор NLX от конструкции LPX, позволяют в полной мере использовать самые последние технологии в области системных плат. NLX — это улучшенная и, что самое главное, полностью стандартизированная версия независимой конструкции LPX, т.е. одну плату NLX можно заменить платой другого поставщика, что было невозможным для плат форм-фактора LPX. Применение системных плат LPX ограничено физическими размерами современных процессоров и соответствующих им теплоотводов, а также новыми типами шин (например, AGP).

Эти проблемы были учтены при разработке форм-фактора NLX (Конструкция системной платы NLX также позволял разместить сдвоенный процессор Pentium III, установленный в разъемы Slot 1.

[](https://advteach.files.wordpress.com/2012/02/nlx1.jpg)

Рисунок 12 – Системная плата форм фактора LPX

Начиная с 2000 года в большинстве систем Slimline применяются различные модели системных плат ATX.

**ATX**

Форм фактор ATX разработан и предложен производителям компьютерных систем в компанией Intel и предназначался для замены использовавшегося долгое время стандарта AT. Официально спецификация ATX была опубликована фирмой Intel в июле 1995 года. Такой открытой публикацией Intel создала новый промышленный стандарт, который позволял производство системных плат другим производителям материнских плат (HP и т. д.).

В ATX сочетаются наилучшие черты стандартов Baby-AT и LPX и заложены многие дополнительные усовершенствования. По существу, ATX – это “лежащая на боку” плата Baby-AT с измененным разъемом и местоположением источника питания. Конструкция ATX физически несовместима ни с Baby-AT, ни с LPX и поэтому для системной платы ATX нужен особый корпус и источник питания.

Конструкция ATX позволила усовершенствовать стандарты Baby-AT и LPX:  
•    Наличие встроенной двойной панели разъемов ввода-вывода. На тыльной стороне системной платы есть область с разъемами ввода-вывода шириной 6,25 и высотой 1,75 дюйма. Это позволяет расположить внешние разъемы непосредственно на плате и исключает необходимость использования кабелей, соединяющих внутренние разъемы и заднюю панель корпуса, как в конструкции Baby-AT.  
•    Наличие одноключевого внутреннего разъема источника питания. Это упрощает замену разъемов на источнике питания типа Baby-AT. Спецификация ATX содержит одноключевой разъем источника питания, который легко вставляется и который невозможно установить неправильно. Этот разъем имеет контакты для подвода к системной плате напряжения 3,3 В, а это означает, что для системной платы ATX не нужны встроенные преобразователи напряжения, которые часто выходят из строя.  
•    Перемещение процессора и модулей памяти. Изменены места расположения этих устройств: теперь они не мешают платам расширения, и их легко заменить новыми, не вынимая при этом ни одного из установленных адаптеров. Процессор и модули памяти расположены рядом с источником питания и обдуваются одним вентилятором, что позволяет обойтись без специального вентилятора для процессора, который не всегда эффективен и часто склонен к поломкам. Есть также место и для большого пассивного теплоотвода.  
•    Более удачное расположение внутренних разъемов ввода-вывода. Эти разъемы для накопителей на гибких и жестких дисках смещены и находятся не под разъемами расширения или самими накопителями, а рядом с ними. Поэтому можно уменьшить длину внутренних кабелей к накопителям, а для доступа к разъемам не нужно убирать одну из плат или накопитель.  
•    Улучшение охлаждения. Процессор и модули памяти охлаждаются тем же вентилятором, что и источник питания. Кроме того, в конструкции ATX вентилятор источника питания направляет поток воздуха внутрь корпуса, увеличивая в нем давление и препятствуя проникновению пыли и грязи. Вы можете установить фильтр и сделать компьютер еще более защищенным.  
•    Снижение стоимости. Конструкция ATX не требует наличия гнезд кабелей к разъемам внешних портов, встречающихся на системных платах Baby-AT, дополнительного вентилятора для процессора и 3,3-вольтного стабилизатора на системной плате. В этой конструкции используется один-единственный разъем питания. Кроме того, вы можете укоротить внутренние кабели дисковых накопителей. Все это существенно уменьшает стоимость не только системной платы, но и всего компьютера, включая корпус и источник питания.

Отличительные черты ATX:  
•    все разъемы плат расширения подключены непосредственно к системной плате (нет никаких выносных плат, как у LPX или NLX).  
•    разъемы перпендикулярны к плоскости системной платы.  
•    имеют уникальную платформу удвоенной высоты для всех встроенных разъемов на системной плате.

ATX определяет следующие характеристики:  
•    геометрические размеры материнских плат,  
•    общие требования по положению разъёмов и отверстий на корпусе,  
•    положение блока питания в корпусе,  
•    геометрические размеры блока питания,  
•    электрические характеристики блока питания,  
•    форму и положение ряда разъёмов (преимущественно питания).

Сегодня стандарт ATX выпускается в нескольких вариациях, которые отличаются друг от друга не только габаритами, но и содержимым.

Размеры плат форм-фактора:  
•    АТХ(полноразмерная) — 30,5 × 24,4 см  
•    MicroATX(декабрь 1997)- 24,4 × 24,4 см  
•    FlexATX(март 1999г.) – 22,9 x 19,1см



Рисунок 13 - Системная плата форм фактора ATX

**BTX**

Официальное представление спецификации The Balanced Technology Extended (ВТХ) 1.0 Public Release состоялось в июле 2004 г. Предполагалось, что BTX придёт на смену форм-фактору ATX.  Форм фактор BTX предназначался для «сверх горячих» процессоров, с тепловой мощностью более 100 Вт.

Назначение ВТХ официально сформулировано следующим образом: спецификации разработаны с целью стандартизации интерфейсов и определения форм-факторов для настольных вычислительных систем в области их электрических, механических и термических свойств. Спецификации описывают механические и электрические интерфейсы для разработки системных плат, шасси, блоков питания и других системных компонентов.

Главные преимущества форм-фактора ВТХ перед АТХ:  
•    возможность применения низкопрофильных компонентов для сборки миниатюрных систем;  
•    продуманное размещение элементов системы внутри корпуса с учетом путей прохождения потоков воздуха и термобаланса;  
•    масштабируемость в рамках доступных модификаций — полно-форматного ВТХ, microBTX, picoBTX;  
•    возможность использования небольших блоков питания;  
•    оптимизированная конструкция крепления системной платы, качественные механические элементы для установки массивных компонентов.

Размеры плат форм-фактора BTX:  
•    BТХ(2004г.) – 32,5 × 26,7 см  
•    MicroBTX(2004г.) – 26,4 × 26,7 см  
•    PicoBTX (2004г.) – 20,3 × 2,67 см

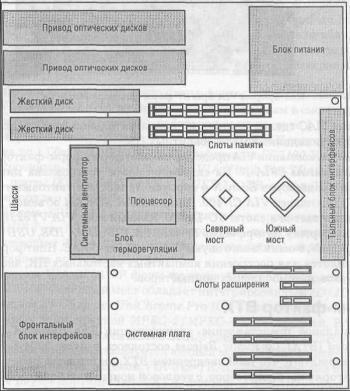
[](https://advteach.files.wordpress.com/2012/02/btx2.jpg)

Рисунок 14 – Схема конструкции системной платы форм фактора BTX

**ITX**

Подразделение Platform Solutions компании VIA Technologies поставило задачу создать системную плату с минимальными размерами (насколько возможно), причем не придумывая для этого нового, не совместимого с уже существующими форм-фактора.

В 2001 году эта задача была выполнена. Новая плата приобрела название ITX, однако уменьшение размеров всего на 6% оказалось недостаточным для промышленного производства, поэтому платы форм-фактора ITX так и не увидели свет. Материнские платы разрабатывались для процессора Cyrix и были существенно меньше по размеру наименьшего форм фактора flexATX на тот период.

Системные платы mini-ITX обладают большинством необходимых портов  вода/вывода. Тем не менее между платами miniITX и другими моделями ATX существует ряд различий:  
•    Процессор в плате mini-ITX обычно припаян к гнезду, что делает невозможным обновление или замену процессора.  
•    В большинстве корпусов mini\_ITX установлены блоки питания TFX, которые поставляются лишь несколькими компаниями, а значит, замена такого блока питания обойдется недешево.  
•    Доступные на рынке блоки питания TFX имеют небольшую выходную мощность, какправило до 240 Ватт.  
•    Встроенный графический адаптер нельзя заменить платой AGP.

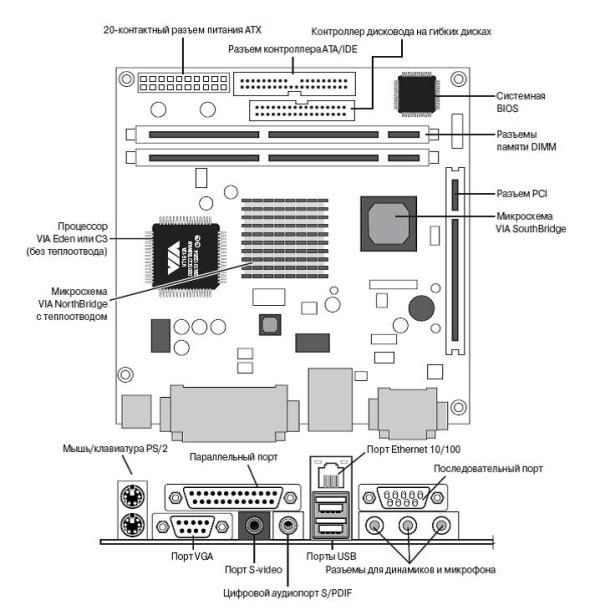
[](https://advteach.files.wordpress.com/2012/02/itx1.jpg)

Рисунок 15 – Схема конструкции  форм фактора ITX

В апреле 2005 года компания VIA представила плату c еще меньшими габаритами, которая характеризовалась минимальными глубиной и шириной, допустимыми в рамках стандарта flex-ATX. Новый форм-фактор назывался nano-ITX.

В январе 2007 г. компания VIA представила первую материнскую плату стандарта Pico-ITX. Эти платы изготавливаются для сверхкомпактных компьютерных систем невысокой производительности. На ней устанавливается процессор VIA C7 с тактовой частотой в 1 ГГц, чипсеты CX700M/VX700 с интегрированным графическим ядром, память DDR2 SO-DIMM. Плата обладает четырьмя разъёмами USB, одним контроллером Ethernet иконтроллером SATA-2. Кроме того, производителям удалось разместить на плате звуковую карту VIA VT1708A (7.1 HDA и S/PDIF) и универсальный картридер.

В настоящее время материнские платы Pico-ITX разработаны также и для процессора Intel Atom.  
Размеры плат форм-фактора ITX:  
•    Mini-ITX – 17,0 × 17,0 см  
•    Nano-ITX – 12,0 × 12,0 см  
•    Pico-ITX – 10,0 × 7,2 см  
В настоящее время этот форм фактор претерпевает сложные времена, в связи с появлением аналогичных габаритов от BTX и ATX

[](https://advteach.files.wordpress.com/2012/02/itx2.jpg)

Рисунок 16 –  Системная плата форм фактора ITX

Поскольку платы и корпуса mini-ITX предоставляются небольшим количеством компаний, возможности модернизации или замены системных компонентов существенно ограниченны. Тем не менее, так как платы mini-ITX соответствуют стандарту flex-ATX, их можно устанавливать в любых корпусах форм-факторов flex-ATX, micro-ATX или полноразмерных ATX и применять вмонтированные в корпуса блоки питания. В свою очередь, в большинство корпусов mini-ITX нельзя установить платы flex-ATX, micro-ATX или ATX; кроме того, в таких корпусах, как правило, имеется блок питания TFX. Остановив свой выбор на системе mini-ITX, необходимо подобрать подходящий  тип процессора, обладающий достаточным быстро действием, ведь замена или модернизация процессора практически всегда будет означать замену системной платы.

**Конструкция корпусов ПК**

Компьютерный корпус - служит для монтажа компонентов компьютерной системы, их питания, условий охлаждения, снижение уровня радиоволн. Так вот, корпуса существуют двух видов:

горизонтальные (desktop), они в свою очередь подразделяются на большие, низкопрофильные и маленькие

вертикальные (tower) – большие, средние и маленькие

Основным параметром, который определяет качество корпуса, является толщина метала, крепления системной платы, а также стенок. Известные фирмы, которые специализируются на выпуске корпусов, делают из металла толщиной 1 мм. Дешевые корпуса таким похвастаться не могут и всего толщина 0,5-0,6 мм. Их стенки легко прогибаются. Компьютеры в небольших корпусах тесно упакованы, облегчены и ограничены в расширении оборудования. Низкопрофильные спроектированы, так что бы занимать как можно меньше места. Большое количество людей предпочитают вертикальные корпуса (один из таких корпусов представлен на рисунке), потому что в таких корпусах больше места для установки дополнительного оборудования: дополнительных жестких дисков, приводов и другое. Вертикальные корпуса, как правило, устанавливают на полу, что является одним из преимуществ, сохраняя больше места на рабочем столе. Компьютерный корпус не должен пропускать радиоволн, мешающие нормальной работе бытовой электронике.

Распространение электромагнитных волн внутри корпуса не причинит вреда компьютеру и не сможет повлиять на его работу, но, например, открытый корпус может мешать работе телевизора, радиоприемников, радио- и мобильных телефонов и других устройств. Эти волны не опасны человеку, по крайней мере, в том уровне, на каком они исходят из компьютера. Самое правильное - это оставаться на расстоянии вытянутой руки от компьютера и монитора, закрывать все заглушки и сам корпус. Достаточно дорогие корпуса покрыты пермаллоем. Этот материал не пропускает электромагнитные излучения (радиоволны). Но для домашнего ПК это неоправданная роскошь.

**Какие моменты следует учесть при покупке корпуса**

Для начала необходимо определиться, какой корпус и для каких целей. Чем мощнее система, тем больше она выделяет тепла и тем быстрее его нужно отводить из корпуса. Определившись с системой, можно начинать приобретение и подбор системного блока. Надо сказать, что чем мощнее ваш компьютер, тем дороже в итоге будет корпус. Как только нашли модель, которая по душе, отправляйтесь в магазин и приступайте к осмотру. Обратите внимание на то, что металлические края конструкции корпуса должны быть завальцованы – порезы от острых краёв корпуса чрезвычайно серьезны. Блок питания, в идеальной ситуации должен быть закреплен в самой верхней точке корпуса, с таким расчётом, чтобы процессор и край материнской платы не был закрыт БП. Т.е. размещение компонентов внутри системного блока должно быть таким, при котором ни одни из компонентов «не нависает» над другим. Жёсткие диски, приводы компакт-дисков – всё это должно быть разведено и не должно препятствовать нормальному потоку воздуха.

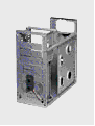
Блоки питания в дорогие корпуса ставят, как правило, достаточно надёжные. Однако рекомендуется докупить отдельный БП, если корпус вам нравится, а вот блок питания вызывает вопросы. Это не способ экономии, а всего лишь попытка сохранить ваше драгоценное «железо» в том виде, в котором вы его покупали. Поверьте, вид палёного процессора стоимостью более двух сотен долларов – зрелище не для слабых духом.

**Компоненты**

Корпус состоит из двух основных компонентов:

**Шасси**: идеальная толщина металла - 0,7 и 0,8 миллиметров, большое количество посадочных мест, дополнительные ребра жесткости, края завальцованы и исключают возможность порезов, крепеж удобен, панель для материнской платы съемная, блок питания расположен горизонтально над материнской платой и не ограничивает доступ к элементам системного блока. Во всех моделях предусмотрены места для установки дополнительных вентиляторов и других устройств:

**Линейка**

**774**

**881**

**885**



Рисунок 17 – Линейка

**Блок питания** - одна из важных частей корпуса. Он обеспечивает надежную работу всех внутренних систем и устройств компьютера. При выборе блока питания необходимоучитывать некоторые критерии:

MTFB (mean time before failure - примерное время до первой неполадки) или MTTF (mean time to failure - тоже самое, что и предыдущее), обычно это минимум 100 тысяч часов.

Диапазон изменения входного напряжения при сохранении стабильной работы блока питания. Для 110В хороший блок питания должен выдержать от 90 до 130, для 220В - 180 до 270.

Пиковый ток при включении. Это значение тока, проходящего по системе в момент инициализации блока питания. Чем меньше, тем лучше, т.к. блок питания не несёт такой большой тепловой удар.

Время (в мс - миллисекундах) удержания выходного напряжения в пределах точно заданных значений после отключения входного (20 мс - хорошее, 10-15 мс - зашибись) :)

У блока питания есть один недостаток: он подстраивается под поглощаемый ток, например система поглощает практически постоянное кол-во энергии, но есть момент, когда SCSI 10000 rpm диск (поглощающий много) выключает двигатель для перехода в режим "засыпания" и блок питания, должен успеть снизить частоты "наполнения" конденсатора. До того как он это сделает, БП делает выброс выработанной энергии. Время на "раздумье" данного параметра измеряется в микросекундах. Последнее время эта проблема почти не существует, т.к. технология контроля поглощение/генерация довольно продвинулась.

В хороших БП есть схема защиты выходных напряжений (в основном вешается на клей к радиаторам, т.к. не является частью платы БП). Просто-напросто наличие данной схемы - это уже хорошо, а если она ещё и точная и рабочая, так это вообще идеально :). Значения её должны быть "отключение при превышении 1/5 напряжения", т.е. для 5В - 6В это критическое напряжение. При зашкаливании, линия 5В принудительно отключается.

Мощность на выходах БП на каждом канале. Параметр означает максимальную сумму Ампер которую способен сгенерировать БП без угрозы повреждения.

Стабилизация напряжения при изменении нагрузки от "мин" до "мах" - похожее с пунктом 5.

Отношение поглощение от сети/вырабатывание на выходе (КПД). Значение, показывающее кол-во энергии которая преобразовывается в тепло во время преобразования тока. Измеряется в %. Чем больше значение эффективности, тем лучше (точнее выработка блока питания и меньше тепла в корпусе).

Ripple, или реакция на шум. Практически одно и тоже что и 5, только реакция на скачки на входе блока питания.

**Основные требования**

Основными требованиями, которые предъявляется к корпуса, являются:

**Совместимость** с предполагаемым форм-фактором материнской платы и блоком питания.

**Соответствие размеров**. Корпус должен быть достаточно велик для размещения всех требуемых устройств — но в то же время достаточно мал, чтобы поместиться в отведенном для него месте.

**Оптимальность конструкции**. Сборка-разборка должна происходить просто, конструкция корпуса должна предусматривать свободный доступ ко всем компонентам.

**Продуманность вентиляции**. Схема вентиляции должна обеспечивать наиболее оптимальное охлаждение компонентов:

применение задних вытяжных вентиляторов существенно улучшает температурный режим;

все задние вентиляторы должны ориентировать воздушный поток в одну сторону, лучше всего — наружу;

применение втяжных фронтальных вентиляторов для вертикальных корпусов практически бесполезно (за редкими исключениями);

использовать вентиляторы диаметром менее 80…100 мм нежелательно;

передние и задние отверстия для циркуляции воздуха не должны перекрываться фальш-панелями и заглушками;

провода и кабели внутри корпуса не должны болтаться, как попало, лучше всего скрутить их в жгуты, перевязать и аккуратно уложить так, чтобы они не мешали потоку воздуха.

**Качество исполнения**. На корпусе не должно быть острых кромок; кроме того, должна обеспечиваться необходимая жесткость креплений.

Остальные параметры — цвет, дизайн и габариты — дело вкуса и личных предпочтений.

**Основные шины расширения.**

Устройства, подключаемые к шине, разделяются на два основных типа: busmasters и busslaves. Busmasters - это устройства, способные управлять работой шины, то есть инициировать запись/чтение и т. п. Busslaves - соответственно, устройства, которые могут только отвечать на запросы.

Важнейшей характеристикой шины является ее разрядность, которая определяет количество данных, передаваемых по шине одновременно (за один такт). Понятно, что чем больше разрядность шины, тем больше ее производительность, хотя, правда, это и не всегда так, так как количество передаваемой в секунду информации зависит еще и от собственно ее частоты. По назначению шины можно разделить на три категории:

·        Шина данных;

·        Адресная шина;

·        Шина управления.

**Шина данных**

По этой шине происходит обмен данными между процессором, картами расширения и памятью. Особую роль здесь играет так называемый DMA-контроллер (DirectMemoyAccess), через который происходит управление транспортировкой данных, минуя процессор. Такой способ хорош тем, что освобождает ресурсы CPU для других нужд. Разрядность шины данных может составлять 8 бит, 16 бит, 32 бит и так далее.

**Адресная шина**

Данные, которые в большом количестве кочуют по шине через материнскую плату, должны, в конце концов, сделать где-нибудь помежкточную остановку. Местом для этой остановки являются отдельные ячейки памяти. Каждая ячейка должна иметь свой адрес. Следовательно, объем памяти, который может адресовать процессор, зависит от разрядности адресной шины. Его можно вычислить по формуле:

Объем адресуемой памяти = 2n, где n - число линий в адресной шине.

Процессор 8088, например, имел в своем распоряжении 20 адресных линий и, таким образом, мог адресовать всего 1 Mb памяти (220=1048576). В компьютерах на базе процессора 80286 адресная шина была уже 24-разрядной, а процессоры 80486 имеют уже 32-разрядную шину, которая позволяет адресовать 4 им гигабайта памяти.

**Шина управления**

Конечно же, незачем просто транспортировать данные по шине и располагать их в памяти, если непонятно, куда их нужно переслать и какое устройство в них нуждается. Разрешение этой проблемы на себя шина контроллера, называемая также системной шиной, или шиной управления.

В качестве конечных пунктов системной шины можно рассматривать слоты расширения, интегрированные на материнскую плату контроллеры и прочее. Все эти устройства соединены между собой шиной управления. Логично предположить, что от ее производительности во многом зависит производительность всей системы, и чем больше тактовая частота и разрядность этой шины, тем лучше. Внешний вид слотов расширения, которые установлены на материнской плате, зависит именно от типа шины управления. Понятно, что, например, разъемы 32-разрядной системной шины будут отличаться от разъемов 16-разрядной шины.

**Шины ISA и EISA**

Шина ISA была первой стандартизированной системной шиной (ISA означает IndustryStandartArchitecture) и долгие годы являлась стандартом в области РС. И даже сегодня разъемы этой шины можно встретить на некоторых системных платах.

8-разрядная шина

Родоначальником в семействе шин ISA была появившаяся в 1981 году 8-разрядная шина (8 bit ISA Bus), которую можно встретить в компьютерах ХТ-генерации. 8-разрядная шина имеет 62 линии, контакты которых можно найти на ее слотах. Они включают 8 линий данных, 20 линий адреса, 6 линий запроса прерываний. Шина функционирует на частоте 4.77 MHz. 8-разрядная шина ISA - самая медленная из всех системных шин (пропускная способность составляет всего 1.2 Mb в секунду), поэтому она уже давным-давно устарела и поэтому сегодня нигде не используется, ну разве что о-о-очень редко (например, некоторые карточки FM-тюнера могут 8-разрядный ISA-интерфейс, так как там шина используется только для управления, а не для передачи собственно данных, и скорость ее работы является некритичной).

16-разрядная шина

Дальнейшим развитием ISA стала 16-разрядная шина, также иногда называемая AT-Bus, которая впервые начала использоваться в 1984 году. Если вы посмотрите на ее слоты (извините, пожалуйста, за плохое качество рисунка), то увидите, что они состоят из двух частей, из которых одна (большая) полностью копирует 8-разрядный слот. Дополнительная же часть содержит 36 контактов (дополнительные 8 линий данных, 4 линии адреса и 5 линий IRQ плюс контакт для нового сигнала SBHE). На этом основании короткие 8-разрядные платы можно устанавливать в разъемы новой шины (сделать это наоборот, конечно же, невозможно).

Передача байта данных по шине ISA происходит следующим образом: сначала на адресной шине выставляется адрес ячейки RAM или порта устройства ввода/вывода, куда следует передать байт, затем на линии данных выставляется байт данных. Производится задержка тактами ожидания и подается сигнал на передачу байта (строб записи), причем неизвестно, успели записаться данные или нет. Поэтому тактова частота шины выбрана 8.33 MHz, чтобы даже самые медленные устройства гарантированно могли произвести по шине обмен даными (командами). Пропускная способность при этом составила 5.3 Mb/s.

**Шина MCA**

В 1987 году компания IBM прекратила выпуск серии РС/АТ и начала производство линии PS/2. Одним из главных отличий нового поколения персональных компьютеров была новая системная шина MCA (MicroChannelArchitecture). Эта шина не обладала обратной совместимостью с ISA, но зато содержала ряд передовых для своего времени решений:

·      8/16/32-разрядная передача данных

·      Пропускная способность составила 20 Mb/s при частоте 10 MHz и максимальной пропускной способности 160 Mb/s (!), то есть больше, чем у 32-разрядной PCI

·      Поддержка нескольких busmaster. Любое устройство, подключенное к шине, может получить право на ее исключительное использование для передачи или приема данных с другого соединенного с ней устройства. Такое устройство, по сути, представляет собой специализированный процессор, который может осуществлять обмен данными по шине независимо от основного процессора. Работу устройств арбитр шины (CACP - CentralArbitrationControlPoint). При распределении функций управления шиной арбитр исходит из уровня приоритета, которым обладает то или иное устройство или операция. Всего таких уровней четыре (в порядке убывания):

·      Регенерация системной памяти

·      Прямой доступ к памяти (DMA)

·      Платы адаптеров

·      Процессор

LocalBus

Все описанные ранее шины (за исключением MCA) имеют общий недостаток - сравнительно низкую пропускную способность. Это связано с тем, что шины разрабатывались в расчете на медленные процессоры. В дальнейшем быстродействие последнего возрастало, а характеристики шин улучшались в основном экстенсивно, за счет добавления новых линий. Препятствием для повышения частоты шины являлось огромное количество выпущенных плат, которые не могли работать на больших скоростях обмена (МСА это касается в меньшей степени, но в силу вышеизложенных причин эта архитектура не играла заметной роли на рынке). В то же время в начале 90-х годов в мире персональных компьютеров произошли изменения, потребовавшие резкого увеличения скорости обмена с устройствами:

·       Создание нового поколения процессоров типа Intel 80486, работающих на внешних частотах до 66 MHz

·      Увеличение емкости жестких дисков и создание более быстрых контроллеров

·      Разработка и активное продвижение на рынок графических интерфейсов пользователя (типа Windows) привели к созданию новых графических адаптеров, поддерживающих более высокое разрешение и большее количество цветов (VGA и SVGA), что привело к нехватке пропускной способности имеющихся шин (MCA, как уже говорилось, не в счет)

Выход из создавшегося положения следующий: осуществлять часть операций обмена данными, требующих высоких скоростей, не через шину ввода/вывода, а через шину процессора, примерно так же, как подключается внешний кэш.

**Шина PCI**

Едва карта VLB успела закрепиться на рынке, как в июне 1992 года фирма Intel изготовила новую шину - шину PCI (PeripheralComponentInterconnect). Именно этот "периферийный соединительный компонент" находится в большинстве современных компьютеров, де-факто стал стандартом для шинной индустрии нашего времени.

Разработчики шины поставили своей целью создать принципиально новый интерфейс, который бы не являлся усовершенствованиями других технологий (как, например EISA), не зависел от платформы (то есть мог работать с будущими поколениями процессоров), имел высокую производительность и был дешев в производстве. Благодаря отказу от использования шины процессора шина PCI оказалась не только процессоронезависимой, но и могла работать самостоятельно, не обращаясь к последней с запросами. Например, процессор может работать с памятью, в то время как по шине PCI передаются данные. Основополагающим принципом шины PCI является применение так называемых мостов (Bridges), которые осуществляют связь шины с другими компонентами системы (например, PCI to ISA Bridge). Другой особенностью является реализация так называемых принципов BusMaster и BusSlave. Например, карта PCI-Masterможет как считывать данные из оперативной памяти, так и записывать их туда без обращения к процессору. Карта PCI-Slave (например, графический контроллер) может только считывать данные.

**Особенности шины PCI:**

·      Синхронный 32-х или 64-х разрядный обмен данными (правда, насколько мне известно, 64-разрядная шина в настоящее время используется только в Alpha-системах и серверах на базе процессоров IntelXeon, но, в принципе, за ней будущее). При этом для уменьшения числа контактов (и стоимости) используется мультиплексирование, то есть адрес и данные передаются по одним и тем же линиям

·      Шина поддерживает метод передачи данных, называемый linearburst (метод линейных пакетов). Этот метод предполагает, что пакет информации считывается (или записывается) одним куском, то есть адрес автоматически увеличивается для следующего байта. Естественным образом при этом увеличивается скорость передачи собственно данных за счет уменьшения числа передаваемых адресов

·      В шине PCI используется совершенно отличный от ISA способ передачи данных. Этот способ, называемый способом рукопожатия (handshake), заключается в том, что в системе определяется два устройства: передающее (Iniciator) и приемное (Target). Когда передающее устройство готово к передаче, оно выставляет данные на линии данных и сопровождает их соответствующим сигналом (IniciatorReady), при этом приемное устройство записывает их (данные) в свои регистры и подает сигнал TargetReady, подтверждая запись данных и готовность к приему следующих. Установка всех сигналов производится строго в соответствии с тактовыми импульсами шины

·      Относительная независимость отдельных компонентов системы. В соответствии с концепцией PCI передачей пакета данных управляет не CPU, а мост, включенный между ним и шиной PCI (HostBridgeCashe/DRAM Controller). Процессор может продолжать работу и тогда, когда происходит обмен данными с RAM. То же происходит и при обмене данными между двумя другими компонентами системы

·      Низкая нагрузка на процессор.

**Драйверы устройств.**

**Драйвер (от английского Driver - водитель) - программа**, отвечающая за работу данного устройства или оборудования, которая обеспечивает связь между компьютером и устройством.

Драйверы еще иногда называют «дрова», а любое устройство называют еще «девайсом».

Например, выражение «дрова для камеры» означает «драйверы для камеры», что в свою очередь означает «программы для совместимости работы операционной системы с камерой».

**Зачем нужны драйверы?**

Появление новых устройств, все более сложных, происходит чрезвычайно быстро и в огромном количестве. Разрабатывают эти [устройства](https://infourok.ru/go.html?href=http%3A%2F%2Fwww.compgramotnost.ru%2Fsostav-computera%2Fdopolnitelnye-ustrojstva-computera) различные компании, придерживаясь своих собственных стандартов и взглядов на жизнь. Соответственно, заранее заложить в операционную систему какие-то программы, обеспечивающие совместимость операционной системы со всеми существующими и еще даже не придуманными устройствами, – невозможно.

Вот и получается, что каждый производитель того или иного устройства должен написать драйверы для своих новых устройств, иначе кто же купит такое несовместимое с компьютером устройство? Причем, программисты той компании, где разработано устройство, пишут несколько драйверов, обеспечивающих совместимость нового устройства с разными операционными системами. Тогда одно и то же устройство можно продавать пользователям разных операционных систем.

**Как узнать, какие на моем компьютере устройства и драйверы? Для виндовс 7, 8:**

**Пуск-Панель управления-Диспетчер устройств**

Для изменения параметров и обновления драйверов используется **Диспетчер устройств**. Чтобы открыть Диспетчер устройств (Device Manager) в Windows 7, щелкните кнопку Пуск, в строке Поиска меню Пуск введите «Диспетчер устройств» и щелкните по имени программы.

Другой способ открыть Диспетчер устройств в Windows 7: пройти по маршруту Пуск — Панель управления — Система и безопасность — вкладка «Оборудование и звук» — Устройства и принтеры — Диспетчер устройств:

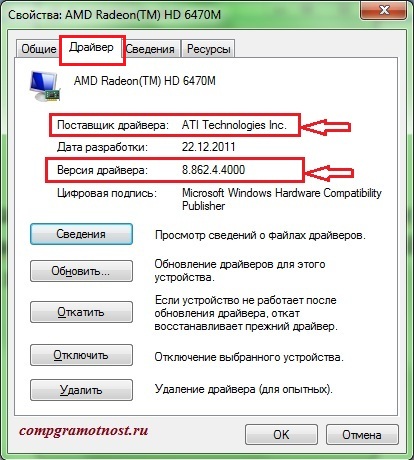
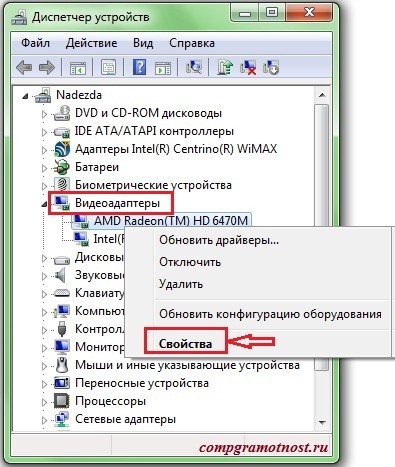


Рисунок 18 – Диспетчер устройств в Windows 7

В Диспетчере устройств выбираем необходимое устройство, кликаем по нему 2 раза левой кнопкой мыши, в выпадающем списке кликаем по нужному устройству правой кнопкой мыши и затем - по кнопке «Свойства».

В результате откроется окно «Свойства», где на вкладке «Драйвер» находим информацию о производителе и драйвере этого устройства

**Нужно ли делать обновление драйверов устройств и каким образом?**

В приведенном выше окне «Свойства» можно сделать обновление драйверов устройств, а также при неудачном обновлении драйвера сделать откат к старому драйверу.

С другой стороны, если не обновлять некоторые драйверы, то могут перестать нормально функционировать соответствующие этим драйверам устройства. Поэтому приходится пользоваться как кнопкой «Обновить» драйвера на приведенном выше скриншоте, так и кнопкой «Откатить».

**Где найти драйверы?**

Если коротко отвечать на этот вопрос, то:

* драйверы иногда находит операционная система Windows 7 самостоятельно в Интернете и сама устанавливает их,
* драйверы можно установить вручную с CD или DVD диска,
* либо установить вручную с официального сайта производителя устройства.

А теперь немного подробнее. Обычно с операционной системой поставляются драйверы для устройств, без которых система не сможет функционировать.

Однако для некоторых устройств могут потребоваться специальные драйверы, обычно предоставляемые на лицензионном CD или DVD диске производителем конкретного устройства.

Например, драйвер для стандартной мышки включен в состав операционной системы. Но если приобрести специальную мышку для компьютерных игр с дополнительными кнопками и функциями, то без установки соответствующего драйвера такая мышка полноценно работать не будет. Драйвер обычно при покупке прилагается  к мышке на диске. CD или DVD диск с этим драйвером желательно сохранять даже после установки с него соответствующей программы на компьютер во избежание впоследствии недоразумений, например, при переустановке операционной системы.

CD или DVD диск с записанными на него драйверами прилагается также к новому принтеру, к новой веб-камере и т.п. Если такого диска нет, а продавец предлагает Вам поискать необходимый драйвер в Интернете (кстати, это реальный случай), то он что-то мудрит и покупать у него устройство не следует. Поэтому при покупке следует выяснять у продавца наличие диска с драйвером для нового устройства.

Если устройство досталось «бывшим в употреблении», а CD или DVD диск с драйверами отсутствует, то драйвер для такого устройства следует поискать на сайте производителя данного устройства. Для этого в поисковой строке Google или Яндекс следует ввести запрос примерно такого содержания: «официальный сайт Название устройства».

Например, для принтера Hewlett-Packard можно ввести запрос «официальный сайт HP».

Драйверы необходимы не только для внешних устройств типа принтера, веб-камеры или мышки. Драйверы необходимы и для внутренних устройств, например, для материнской (системной) платы, для видеокарты, для звуковой карты. Если видеокарта или звуковая карта встроены в материнскую плату, то драйвера для них не нужны, потому что они встроены в драйвера для материнской платы.  Если видеокарта или звуковая карта устанавливаются отдельно, то для каждой карты нужен отдельный диск с драйверами.

**Как устанавливать драйверы?**

Если есть диск, то вставляем его и ждем, когда он автоматически запустится, загрузит программное обеспечение и на этом установка драйвера завершена.

Если этого по каким-то причинам не произошло, то идем в Пуск – Компьютер, кликаем по вставленному диску, находим и запускаем файл setup.exe.

Если драйвер скачен с официального сайта производителя, то точно также находим и запускаем setup.exe.

Чтобы установить драйвер вручную нужно знать Модель и производителя оборудования.

Иногда нужен серийный номер (платы).

Как правило их выделяют самой жирной крупной надписью на оборудовании, либо на коробке от оборудования, либо на системном блоке (если он покупался в сборе).

Также можно найти драйвера по ID оборудованию.

В Диспетчере устройств выбираем: Устройство- Сведения-Свойства-ИД оборудования.

Что такое ИД оборудования: Identificator (идентификатор) - уникальный номер, как правило в нем зашифрована информация о производителе.

Лекция 8

**Виды памяти в технических средствах информатизации. Принципы хранения информации. Накопители на жестких магнитных дисках.**

Устройства хранения данных (память) являются составной частью практически любого современного технического средства информатизации. В памяти хранятся как программы работы технического средства, так и данные, используемые в процессе его работы. Применительно к персональным компьютерам вся память компьютера подразделяется на внутреннюю и внешнюю. Структура памяти компьютера показана на рисунке 19.

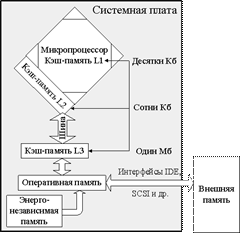


Рисунок 19 – Структура памяти компьютера

**Внутренняя память** предназначена для временного хранения программ и обрабатываемых в текущий момент данных (оперативная память, кэш-память), а также для долговременного хранения информации о конфигурации ПК (энергонезависимая память). Все виды запоминающих устройств, расположенные на системной плате, образуют внутреннюю память ПК, к которой относится:

· сверхоперативная память (кэш-память);

· оперативная память;

· постоянная память;

· энергонезависимая память.

Физической основой внутренней памяти являются электронные схемы (ПЗУ, ОЗУ), отличающиеся высоким быстродействием, но они не позволяют хранить большие объемы данных. Кроме этого, основная внутренняя память – оперативная – является энергозависимой, т.е. при отключении ПК ее содержимое стирается. Вследствие этого возникает необходимость в средствах длительного хранения больших объемов данных.

В персональных компьютерах эта функция возложена на внешнюю память, которая по своим характеристикам в противоположность внутренней памяти, является медленной, энергонезависимой и практически неограниченной.

При изучении носителей важно иметь представление о физических принципах, положенных в основу записи и чтения данных. В современных технических средствах информатизации сочетаются три вида носителей, отличающиеся физическим принципом организации памяти: электрические, магнитные, оптические.

***Накопитель на жестких магнитных дисках*** (**НЖМД**, **жесткий диск**, **хард**, **харддиск**, **HDD**, **HMDD** или **винчестер**, от англ. Hard (Magnetic) Disk Drive, HDD, HMDD)*– энергонезависимое, перезаписываемое запоминающее устройство.* Является основным накопителем данных практически во всех современных компьютерах (Рисунок 20,21).

Информация в HDD записывается на жесткие (алюминиевые или стеклянные) пластины, покрытые слоем ферромагнитного материала, чаще всего двуокиси хрома. В некоторых НЖМД используется одна пластина, в других – несколько на одной оси. Считывающие головки в рабочем режиме не касаются поверхности пластин благодаря прослойке набегающего потока воздуха, образуемого у поверхности при быстром вращении. Расстояние между головкой и диском составляет несколько нанометров (в современных дисках 5-10 нм), а отсутствие механического контакта обеспечивает долгий срок службы устройства. При отсутствии вращения дисков, головки находятся у шпинделя или за пределами диска в безопасной зоне, где исключен их нештатный контакт с поверхностью дисков.

|  |  |
| --- | --- |
| *https://siblec.ru/img/94/094.files/image126.jpg* | *https://siblec.ru/img/94/094.files/image127.jpghttps://siblec.ru/img/94/094.files/image128.jpg* |
| Рисунок 20 – Внешний вид HDD | Рисунок 21– Внутренний вид HDD |

1. **7.5.1. Устройство жесткого диска**

Жесткий диск состоит (Рисунок 22,23) из следующих основных узлов: корпус из прочного сплава, собственно жесткие диски (пластины) с магнитным покрытием, блок головок с устройством позиционирования, электропривод шпинделя и блок электроники.

|  |  |
| --- | --- |
| *https://siblec.ru/img/94/094.files/image129.jpg* | https://siblec.ru/img/94/094.files/image130.jpg |
| Рисунок 22 – Устройство HDD | Рисунок 23 – Функциональная схема HDD |

Устройство позиционирования головок состоит из неподвижной пары сильных, как правило, неодимовых, постоянных магнитов и катушки на подвижном блоке головок.

Вопреки расхожему мнению, жесткие диски не герметичны. Внутренняя полость жесткого диска сообщается с атмосферой через фильтр, способный задерживать очень мелкие (несколько мкм) частицы. Это необходимо для поддержания постоянного давления внутри диска при колебаниях температуры корпуса. Пылинки, оказавшиеся при сборке в жестком диске и попавшие на поверхность диска, при вращении сносятся на еще один фильтр – пылеуловитель.

1. **7.5.2. Характеристики HDD**

*Основные характеристики, определяющие качество функционирование и быстродействие жесткого диска приведены ниже.*

***Интерфейс*** (*англ. interface*)*– набор, состоящий из линий связи, сигналов, посылаемых по этим линиям, технических средств, поддерживающих эти линии, и правил обмена.*

Современные накопители могут использовать интерфейсы:

- ATA (AT Attachment, он же IDE – Integrated Drive Electronic, он же Parallel ATA);

- Serial ATA;

- SCSI (Small Computer System Interface);

- SAS, FireWire, USB и др.

***Емкость*** (англ. capacity) – *количество данных, которые могут храниться накопителем.*Емкость современных устройств достигает 2000 Гб. В отличие от принятой в информатике (случайно) системе приставок, обозначающих кратную 1024 величину (кило = 1024, мега = 1 048 576 и т. д.; позже для этого были не очень успешно введены двоичные приставки). Производители указывают неформатированную емкость (вместе со служебной информацией), что делает «зазор» между заявленными «200 Гб» и реальными 160 Гб.

***Физический размер*(*форм-фактор*)**. Почти все современные (2002-2008 гг.) накопители для персональных компьютеров и серверов имеют размер либо 3,5, либо 2,5 дюйма. Последние чаще применяются в ноутбуках. Получили также распространение форматы – 1,8 дюйма, 1,3 дюйма и 0,85 дюйма.

***Время произвольного доступа*** (англ. random access time) – от 3 до 15 мс. Как правило, минимальным временем обладают серверные диски (например, у Hitachi Ultrastar 15K147 – 3,7 мс), самым большим временем из актуальных – диски для портативных устройств (Seagate Momentus 5400.3 – 12,5 мс).

***Скорость вращения шпинделя*** (англ. spindle speed) – количество оборотов шпинделя в минуту (rpm). От этого параметра в значительной степени зависят время доступа и скорость передачи данных. В настоящее время выпускаются винчестеры со следующими стандартными скоростями вращения: 4200, 5400 и 7200 (ноутбуки), 7200 и 10 000 (персональные компьютеры), 10 000 и 15 000 об./мин. (серверы и высокопроизводительные рабочие станции).

***Надежность*** (англ. reliability) – определяется как среднее время наработки на отказ (Mean Time Between Failures, MTBF). При определи надежности в процессе функционирования используется **технология SMART**. (S.M.A.R.T. - англ. Self Monitoring Analysing and Reporting Technology – технология оценки состояния жесткого диска встроенной аппаратурой самодиагностики, а также механизм предсказания времени выхода его из строя).

***Количество операций ввода-вывода в секунду*** – у современных дисков это около 50 оп./сек при произвольном доступе к накопителю и около 100 оп./сек при последовательном доступе.

***Потребление энергии*** – важный фактор для мобильных устройств.

***Уровень шума*** – шум, который производит механика накопителя при его работе. Указывается в децибелах. Тихими накопителями считаются устройства с уровнем шума около 26 дБ и ниже. Шум состоит из шума вращения шпинделя (в том числе аэродинамического) и шума позиционирования.

***Сопротивляемость ударам*** (англ. G-shock rating) – сопротивляемость накопителя резким скачкам давления или ударам, измеряется в единицах допустимой перегрузки во включенном и выключенном состоянии.

***Скорость передачи данных*** (англ. Transfer Rate):

- внутренняя зона диска: от 44,2 до 74,5 Мб/с;

- внешняя зона диска: от 60,0 до 111,4 Мб/с.

***Объем буфера*.** Буфером называется промежуточная память, предназначенная для сглаживания различий скорости чтения/записи и передачи по интерфейсу. В современных HDD он обычно варьируется от 8 до 32 Мб.

Лекция 9

**Мониторы и видеоадаптеры. Устройство, принцип действия, подключение. Проекционные аппараты.**

Монито́р — универсальное устройство визуального отображения всех видов информации, состоящее из дисплея и устройств, предназначенных для вывода текстовой, графической и видео информации на дисплей.

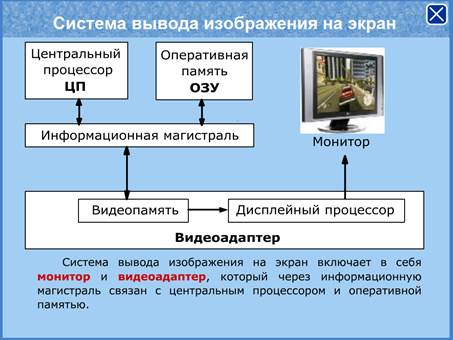


Рисунок 24 – Система ввода изображения на экран

**Классификация мониторов**

**По виду выводимой информации**

Ø алфавитно-цифровые

· дисплеи, отображающие только алфавитно-цифровую информацию

· дисплеи, отображающие псевдографические символы

· интеллектуальные дисплеи, обладающие редакторскими возможностями и осуществляющие предварительную обработку данных

Ø графические

· векторные

· растровые

**По строению**

· ЭЛТ — на основе электронно-лучевой трубки

· ЖК — жидкокристаллические мониторы ( LCD)

· Плазменный — на основе плазменной панели

· Проектор — видеопроектор и экран, размещённые отдельно или объединённые в одном корпусе (как вариант — через зеркало или систему зеркал)

· OLED-монитор — на технологии OLED (органический светоизлучающий диод)

· Виртуальный ретинальный монитор — технология устройств вывода, формирующая изображение непосредственно на сетчатке глаза.

· Лазерный — на основе лазерной панели.

1. **По типу видеоадаптера**

HGC

CGA

EGA

VGA, SVGA

**Физические принципы формирования изображения**

v ЭЛТ - Электронно-лучевой прибор, преобразующий электрические сигналы в световые.

Основные части:

· электронная пушка, предназначена для формирования электронного луча, в цветных кинескопах и многолучевых осциллографических трубках объединяются в электронно-оптический прожектор;

· экран, покрытый люминофором — веществом, светящимся при попадании на него пучка электронов;

· отклоняющая система, управляет лучом таким образом, что он формирует требуемое изображение.

Первым монитором можно по праву считать устройство, разработанное в далеком 1897 году немецким физиком Фердинандом Брауном. Хотя, честно говоря, с нынешними мониторами оно имело более чем отдаленное сходство, поскольку являлось не чем иным, как самым настоящим осциллографом. Тем не менее, начало было положено, и человечество вошло в компьютерную эру, вглядываясь в монохромные катодно-лучевые мониторы первых поколений ЭВМ. Наиболее важным элементом таких дисплеев является кинескоп, называемый также электронно-лучевой трубкой (ЭЛТ). Кинескоп состоит из герметичной стеклянной трубки, внутри которой находится вакуум (то есть, чтобы не мешать свободному полету электронов, весь воздух из нее удален). Один из концов ЭЛ трубки узкий и длинный — это горловина, а другой — широкий и достаточно плоский — это экран. С фронтальной стороны внутренняя часть стекла трубки покрыта люминофором — веществом, которое испускает свет при облучении его заряженными частицами. Для создания изображения в ЭЛТ-мониторе используется электронная пушка, расположенная в той узкой горловине. Из пушки под действием сильного электростатического поля исходит переменный поток электронов (электронный луч), способный отклоняться в вертикальной и горизонтальной плоскости, что обеспечивает его последовательное попадание на все поле экрана. Отклоняющая электроны система состоит из нескольких индукционных катушек, размещенных у горловины кинескопа. С помощью переменного магнитного поля две такие катушки создают отклонение пучка электронов в горизонтальной плоскости, а другие две — в вертикальной. Добравшись до экрана, поток электронов «бомбардирует» люминофорный слой, в результате чего последний начинает светиться. Именно светящиеся точки люминофора и формируют изображение, которое мы видим на экране наших ЭЛТ-мониторов.

**Режимы работы монитора**

*Текстовый режим*.

В текстовом режиме экран монитора условно разбивается на отдельные участки - знакоместа, чаще всего на 25 строк по 80 символов ( знакомест ). В каждое знакоместо может быть введён один из 256 символов, используемых в компьютере. В число этих символов входят большие и малые латинские буквы, цифры, определённые символы, а также псевдографические символы, используемые для вывода на экран таблиц и диаграмм, построения рамок вокруг участков экрана и так далее.

В число символов, изображаемых на экране в текстовом режиме, могут входить и символы кириллицы.

На цветных мониторах каждому знакоместу может соответствовать свой цвет символа и фона, что позволяет выводить красивые цветные надписи на экран. На монохромных мониторах для выделения отдельных частей текста и участков экрана используется повышенная яркость символов, подчёркивание и инверсное изображение.

**Графический режим**.

Графический режим предназначен для вывода на экран графиков, рисунков и так далее. Разумеется в этом режиме можно выводить и текстовую информацию в виде различных надписей, причём эти надписи могут иметь произвольный шрифт, размер и др.

В графическом режиме экран состоит из точек, каждая из которых может быть тёмной или светлой на монохромных мониторах и одного или нескольких цветов - на цветном. Количество точек на экране называется разрешающей способностью монитора в данном режиме. Следует заметить что разрешающая способность не зависит от размеров экрана монитора.

**Как кодируется изображение**

В видеопамяти находится двоичная информация об изображении, выводимом на экран. Эта информация состоит из двоичных кодов каждого видеопикселя.

Код пикселя - это информация о цвете пикселя.

Для получения черно-белого изображения (без полутонов) пиксель может принимать только два состояния: светится — не светится (белый — черный). Тогда для его кодирования достаточно одного бита памяти:

1 — белый,

0 — черный.

Пиксель на цветном дисплее может иметь различную окраску. Поэтому одного бита на пиксель — недостаточно.

Для кодирования 4-цветного изображения требуется два бита на пиксель, поскольку два бита могут принимать 4 различных состояния. Может использоваться, например, такой вариант кодировки цветов:

00 — черный,

01 — красный,

10 — зеленый

11 — коричневый

На RGB-мониторах все разнообразие красок получается из сочетаний трех базовых цветов: красного, зеленого, синего. Из трех цветов можно получить восемь комбинаций:

- - - черный

- - с синий

- з - зеленый

- з с голубой

к - - красный

к - с розовый

к з - коричневый

к з с белый

Здесь каждый базовый цвет обозначается первой буквой, а черточкой — отсутствие цвета.

Следовательно, для кодирования 8-цветного изображения требуется три бита памяти на один видеопиксель. Если наличие базового цвета обозначить единицей, а отсутствие нулем, то получается следующая таблица кодировки восьмицветной палитры:



Двоичный код восьмицветной палитры

Из сказанного, казалось бы, следует вывод: с помощью трех базовых цветов нельзя получить палитру, содержащую больше восьми красок. Однако на экранах современных компьютеров получают цветные изображения, составленные из сотен, тысяч и даже миллионов различных красок и оттенков. Как это достигается?

Оказывается, если иметь возможность управлять интенсивностью (яркостью) свечения базовых цветов, то количество различных вариантов их сочетаний, дающих разные краски и оттенки, увеличивается.

Шестнадцатицветная палитра получается при использовании 4-разрядной кодировки пикселя: к трем битам базовых цветов добавляется один бит интенсивности. Этот бит управляет яркостью всех трех цветов одновременно (интенсивностью трех электронных пучков). Вот таблица кодировки 16-цветной палитры:



Двоичный код шестнадцатицветной палитры. И - бит интенсивности

Большее количество цветов получается при раздельном управлении интенсивностью базовых цветов. Причем интенсивность может иметь более двух уровней, если для кодирования каждого из базовых цветов выделять больше одного бита.

Из сказанного можно вывести правило:

Количество различных цветов - К и количество битов для их кодировки - b связаны между собой формулой: 2b=К

Для получения цветовой гаммы из 256 цветов требуется 8 бит = 1 байт на пиксель, т.к. 28=256.

Размер необходимой видеопамяти определяется размером графической сетки дисплея и количеством цветов. Минимальный размер видеопамяти должен быть таким, чтобы в него помещался один кадр (одна страница) изображения. Например, для сетки 640x200 и черно-белого изображения минимальный размер видеопамяти должен быть таким:

640 х 200 х 1 = 128000 бит = 16000 байт.

Это приблизительно 16 Кбайт.

Для 4-цветной гаммы и той же графической сетки видеопамять должна быть в 2 раза больше — 32 Кбайта; для 8-цветной — 48 Кбайт.

На современных высококачественных дисплеях используется палитра более чем из 16 миллионов цветов. Требуемый размер видеопамяти в этом случае — несколько мегабайт.

Выводы:

1. Информация в видеопамяти — это двоичные коды, обозначающие цвет каждого пикселя на экране.

2. Для кодирования 2 цветов достаточно 1 бита на пиксель; 4 цветов — 2 бита; 8 цветов — 3 бита; 16 цветов — 4 бита и т.д. Количество цветов (К) и размер кода в битах (b) связаны формулой: К=2b.

3. Из трех базовых цветов можно получить 8 различных красок. Большее число красок получается путем управления интенсивностью базовых цветов.

4. Минимально-необходимый объем видеопамяти зависит от размера сетки пикселей и от количества цветов. Обычно в видеопамяти помещается несколько страниц (кадров) изображения одновременно.

Пример 1.. Определить глубину цвета в графическом режиме True Color, в котором палитра состоит из более чем 4 миллиардов (4 294 967 296) цветов.

I = log242 949 67 296 = 32 бит

В современных компьютерах используются различные графические режимы экрана монитора, каждый из которых характеризуется разрешающей способностью и глубиной цвета. Для реализации каждого графического режима требуется определенный объем видеопамяти компьютера.

Пример 2. Определить объем видеопамяти компьютера, который необходим для реализации графического режима монитора High Color с разрешающей способностью 1024´768 точек и палитрой из 65536 цветов.

Глубина цвета составляет:

I = log265 536 = 16 бит

Количество точек изображения равно:

1024´768 = 786 432

Требуемый объем видеопамяти равен:

16 бит ´ 786 432 = 12 582 912 бит » 1,2 Мбайта

Важнейшими характеристиками монитора являются размеры его экрана, которые задаются величиной его диагонали в дюймах (15”, 17”, 21” и т.д.) и размером точки экрана (0,25 мм или 0,28 мм), а разрешающая способность экрана монитора задается количеством точек по вертикали и горизонтали (640´480, 800´600 и т.д.). Следовательно, для каждого монитора существует физически максимально возможная разрешающая способность экрана.

Лекция 10

**Нестандартные периферийные устройства: манипуляторы, дигитайзер, мониторы.**

**Манипуляторы (джойстик)**

Манипуляторы (джойстик)

В геймерских кругах уже давно не утихают споры о том, что же лучше – клавиатура с мышью или джойстик. Эта тема практически настолько же актуальна, как препирания приверженцев консолей и ПК. Однако многие геймеры и не догадываются, что используют неправильное название своего любимого контроллера. Ведь джойстик и геймпад – разные вещи.

Джойстик – это устройство ввода, выглядит как подвижная вертикальная ручка. Он больше всего похож на рычаг коробки передач или на рукоять управления авиационной техникой. Чаще всего на нем присутствуют кнопки, слайдеры и переключатели. Они расположены на верхушке ручки и на устойчивом основании. Некоторые модели крепятся на поверхность с помощью присосок или просто держаться за счёт своего веса.

  Виды контроллеров делится на три категории

Данный вид контроллеров делится на три категории:

* Одномерные. Предназначены для управления по одной оси «x» или «y».
* Двухмерные. Могут получать данные о перемещениях сразу в двух координатных осях.
* Трёхмерные. Помимо осей «x» и «y» также способны считывать координаты оси «z».

Также джойстики делятся на аналоговые и дискретные. Аналоговые предназначены для более тонкого контроля, а потому могут оценивать силу нажатия. Проще говоря, чем сильнее отклонён рычаг – тем больше будет скорость или угол наклона при движении. Дискретные же могут получать сигнал только в виде нуля или единицы, то есть «включён» или «выключен». Из-за своего несовершенства аналоговые контроллеры практически не используются.

**Назначение**

Первые джойстики использовались в стационарных аркадных игровых автоматах или как дополнительный контроллер для домашних приставок. В те времена игры были гораздо проще по своей механике. Управление в них чаще всего требовало лишь движения по одной-двум осям и пары кнопок для простых действий, вроде стрельбы. Поэтому такие контроллеры были достаточно распространены. Со временем игры становились всё сложнее, но потребность в джойстиках не отпала. Несмотря на то, что клавиатуры или пресловутые геймпады подходят для любых игр, в определённых жанрах предпочтительнее использовать именно джойстики. Чаще всего этим жанром являются разнообразные авиасимуляторы или гонки. Рычаги управления используются в них для более реалистичного и удобного управления.

Иногда в связи с джойстиками используются педали и дополнительный рычаг. Определённую популярность имела и связка таких манипуляторов с очками виртуальной реальности. Используя всё это оборудование, геймеры могут достигать максимально реалистичного ощущения от управления техникой в играх.

**Подключения**

Для того, чтобы можно было подключить джойстик к компьютеру, его обычно оснащают USB-разъемом. Есть и такие модели, которые подключаются по беспроводной связи. Вот почему к компьютеру подключают специальный адаптер. Это приемник-передатчик, который обменивается с джойстиком радиосигналами. Отдельные модели сопрягаются с компьютером через канал Bluetooth.

**Трекбол**

Трекбол — ручное указательное устройство ввода информации для компьютера. Аналогично мыши по принципу действия и по функциям: позволяет вводить информацию об относительном перемещении путём вращения рукой закреплённого в корпусе шара и подавать команды нажатиями на кнопки.

**Устройство и принцип работы**

Функционально представляет собой перевёрнутую механическую (шариковую) мышь: сверху или сбоку корпуса находится шар, который пользователь может вращать ладонью или пальцами; также на корпусе находятся кнопки. Как и в случае мыши, сигналы о вращении шара преобразуются в перемещение указателя или поворот объекта на мониторе, а нажатия кнопок приводят к выполнению определённых команд или переключению режимов; разница лишь в том, что у трекбола движется только шар, тогда как у мыши необходимо перемещать по столу корпус.

Конструктивно трекбол также похож на мышь: вращение шарика приводит в движение пару валиков, соединённых с механическими датчиками, либо, в более современном варианте, движения шара сканируют размещённые в корпусе оптические датчики. Протокол обмена данными с компьютером, как правило, также полностью соответствует протоколу для мыши. Поэтому с точки зрения компьютера трекбол является стандартным интерфейсным указательным устройством; в отсутствие специальных драйверов он воспринимаются операционной системой компьютера как стандартная мышь и нормально поддерживаются стандартным универсальным драйвером мыши.

Подключения

Шаг 1. Установка драйвера

Шаг 2. USB подключение к компьютеру

Шаг 3. Настройка ориентации и клавиш

**Дигитайзер**

Дигитайзер (графический планшет) — это устройство, предназначенное для оцифровки изображений, применяемое для создания на компьютере рисунков и набросков. Художник создает изображение на экране, но его рука водит пером по планшету. Как правило, планшет используют профессиональные художники для более точной обработки (создания) изображений.

**Принцип действия**

Дигитайзер, или планшет, как его часто называют, состоит из двух основных элементов: основания и курсора, двигающегося по его поверхности. Принцип действия дигитайзера основан на фиксации местоположения курсора с помощью встроенной в планшет сетки. При нажатии на кнопку курсора его местоположение на поверхности планшета фиксируется, а его координаты передаются в компьютер. Сетка состоит из проволочных или печатных проводников с довольно большим расстоянием между соседними проводниками (от трех до шести мм).

Подключения

Шаг 1. Установка драйвера планшета

Шаг 2. Настройка ориентации и клавиш

Шаг 3. Настройка пера планшета

Шаг 4. Отключите сенсорный ввод

1. **Автор методических указаний** – лицо, создавшее данные методические указания своим трудом.

   **Составитель методических указаний** – лицо, создавшее методические указания, состоящие из других текстов. Составитель не создаёт собственно тексты, он осуществляет подбор материала, его организацию в пределах сборника. [↑](#footnote-ref-1)