

Источник: Таненбаум Э., Остин Т. Архитектура компьютера. 6-е изд. — СПб.: Питер, 2013. — 816 с.: ил.
ISBN 978-5-496-00337-7
Стр. 20-71

Тема 1 Введение

Цифровой компьютер – это машина, которая может решать задачи, выполняя данные ей команды. Последовательность команд, описывающих решение определенной задачи, называется программой. Электронные схемы каждого компьютера могут распознавать и выполнять ограниченный набор простых команд. Поэтому все программы перед выполнением должны быть превращены в последовательность таких команд, которые обычно не сложнее чем

- Сложить два числа;
- Проверить, не является ли число нулем;
- Скопировать кусок данных из одной части памяти в другую.

Это примитивные команды и являются тем языком, которым человек общается с компьютером. Такой язык называется машинным. Обычно разработчик старается сделать машинные команды как можно проще, что бы избежать дополнительных сложностей при конструировании компьютера, снизить затраты на электронику. Поэтому большинство машинных языков очень примитивны и их использование трудно и утомительно.

Это привело к тому, что с течением времени появились ряд уровней абстракций, каждая из которых настраивается над абстракцией более низкого уровня. Такой поход называется многоуровневой компьютерной организацией.

1.1 Многоуровневая компьютерная организация

Существует разница между тем, что удобно людям и тем, что удобно для компьютера (или что удобно программисту и что удобно разработчику электроники). В литературе это получило название семантического разрыва. Семантический разрыв определяет различие принципов, лежащих в основе языков программирования высокого уровня и тех принципов, которые определяют архитектуру ЭВМ. В этом курсе мы рассмотрим, каким образом можно решить эти проблемы.

1.1.1 Языки, уровни и виртуальные машины

Обозначим через Я1 – язык программирования, удобный для человека. Машинные команды также образуют некоторый язык, понятный

компьютеру, который обозначим как Я0. Для того, чтобы перевести программы с языка Я1 на понятный компьютеру язык Я0 можно использовать следующие подходы.

Первый способ – замена каждой команды, написанной на языке Я1 эквивалентным набором команд, написанных на языке Я0. В этом случае компьютер выполняет новую программу, написанную на языке Я0 вместо старой, записанной на языке Я1. Такая технология называется трансляцией.

Второй способ – написание программы на языке Я0, которая берет команды, написанные на языке Я1 в качестве исходных данных, рассматривает каждую команду по очереди и сразу выполняет эквивалентный набор команд языка Я0. Эта технология не требует составления новой программы на языке Я0, Эта технология называется интерпретацией, а программа, осуществляющая интерпретацию называется интерпретатором.

Различия между трансляцией и интерпретацией заключается в том, что при трансляции все программа Я1 переделывается в программу Я0 и программа Я1 может быть отброшена, а в память компьютера загружается программа Я0. При интерпретации каждая команда перекодируется и тут же выполняется.

На практике проще представить себе существование некоторого гипотетического компьютера, или виртуальной машины, для которой машинным языком является удобный для нас язык Я1, чем думать о трансляции и интерпретации. Назовем такую машину М1, а машину, понимающую язык Я0, машиной М0.

Чтобы трансляция и интерпретация были бы целесообразны, необходимо, чтобы языки Я0 и Я1 не очень отличались друг от друга. Казалось бы, это требование находится в некотором противоречии с целью создания языка Я1, удобного для человека. Однако это противоречие разрешается следующим очевидным образом.

Создается следующий набор команд Я2, который в большей степени ориентирован на человека. Этому языку соответствует виртуальная машина М2, соответствующие трансляторы или интерпретаторы Я2-Я1 и т.д. Изобретение ряда языков, каждый из которых более удобен для человека, чем предыдущий может продолжаться до тех пор, пока мы не дойдем до подходящего. Каждый последующий язык рассматривает предыдущий как основу и поэтому можно представить компьютер в виде уровней (см. рис. В1).

Человеку, пишущему программы на языке Яп не обязательно знать об интерпретаторах и трансляторах более низких уровней. Большинство пользователей, использующих машину уровня n интересуются только этим уровнем. Однако для изучения работы компьютера необходимо изучить все уровни.

Деление на уровни является в некотором роде условным. Можно говорить о шести уровнях представления современного компьютера. (рис. В2).

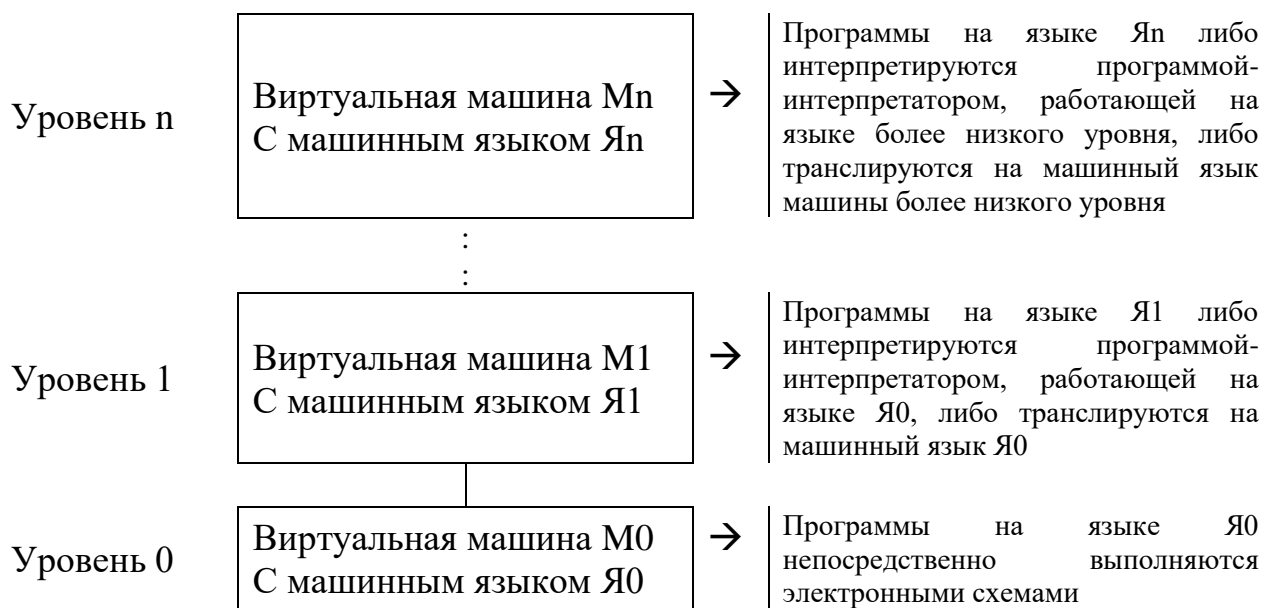
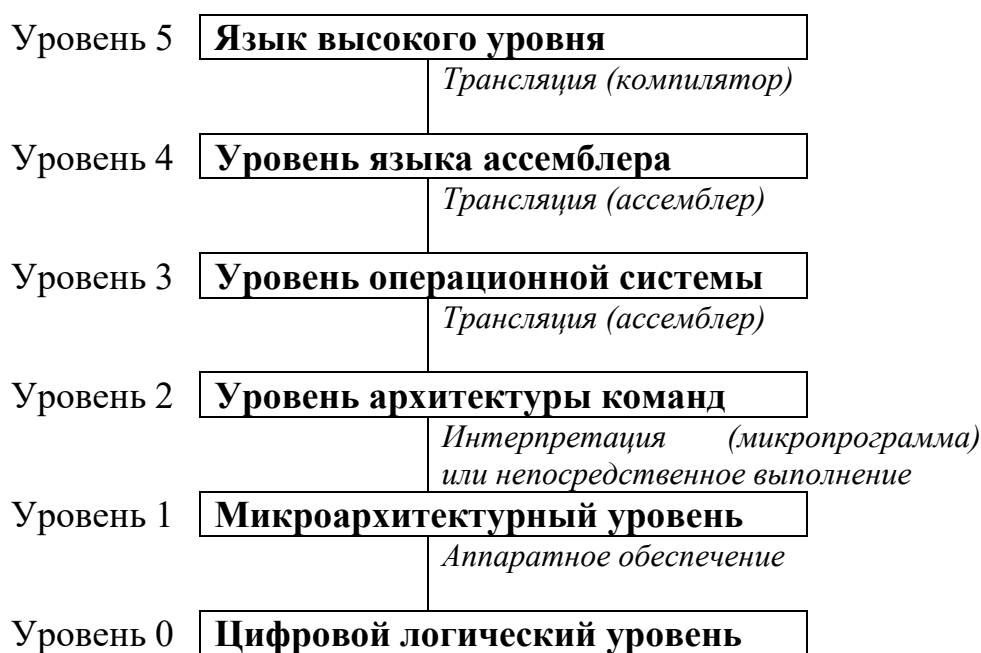


Рис. В1.



На рисунке отсутствует уровень физических устройств, расположенный ниже уровня 0. Это электронные схемы и не являются предметом рассмотрения курса.

На самом нижнем уровне – цифровом логическом – объекты состоят из вентилях и хотя вентили являются электронными схемами, их работа может быть описана как цифровые средства. Вентили образуют бит памяти, которые в свою очередь, образуют регистры.

Следующий уровень – микроархитектурный. На этом уровне рассматриваются совокупности регистров, которые образуют локальную (или регистровую) память и арифметическо-логическое устройство (АЛУ), предназначенное для выполнения простых операций. Регистры вместе с

АЛУ образуют тракт данных. Основная операция состоит в следующем: выбирается один или два регистра, АЛУ производит над ними некоторую операцию (например, сложение), результат помещается в один из регистров. В некоторых машинах тракт данных контролируется микропрограммой. На других контроль осуществляется аппаратными средствами.

Второй уровень – уровень архитектуры системы команд. Руководства по машинному языку, выпускаемые фирмами-производителями содержат информацию именно об этом уровне. Когда там описывается система команд, то описываются команды, выполняемые программой-интерпретатором или аппаратными средствами.

Следующий третий уровень – уровень операционной системы-- обычно гибридный. Большинство команд этого уровня имеется также и на уровне архитектуры системы команд (команды, имеющиеся на одном уровне, также могут иметься и на других). Особенности уровня: новые команды, иная организация памяти, способность выполнять две и более программ одновременно и некоторые другие. Команды третьего уровня, идентичные командам второго, выполняются микропрограммой или аппаратными средствами, а не операционной системой. Часть команд (не имеющих на втором уровне), выполняются операционной системой.

Нижние уровни конструируются не для того, что бы с ними работал обычный программист. Они предназначены для работы интерпретаторов и трансляторов и поддерживаются системными программистами. Уровни с четвертого и выше предназначены для работы прикладных программистов. Следующая особенность: уровни 2 и 3 обычно интерпретируются, а уровни 4 и выше как правило, поддерживаются транслятором. Уровни 1, 2 и 3 – цифровые, т.е. программы, написанные на языках этих уровней, состоят из наборов цифр. Начиная с четвертого уровня, языки содержат слова и сокращения, понятные человеку.

Четвертый уровень представляет собой символьную запись языка более низкого уровня. Программа, которая выполняет трансляцию называется ассемблером.

Пятый уровень обычно состоит из языков, разработанных для прикладных программистов. Такие языки называются языками высокого уровня.

Выводы: компьютер проектируется как иерархическая структура уровней, каждый из которых надстраивается над предыдущим. Каждый уровень представляет собой абстракцию с различными объектами и операциями.

Набор типов данных, операций и особенностей каждого уровня называется архитектурой. Архитектура связана с аспектами, которые видны программисту. Аспекты разработки, технологии и т.д. не являются частью архитектуры. Термины компьютерная архитектура и компьютерная организация в сущности означают одно и то же.

1.1.2 Развитие многоуровневых машин

Аппаратное обеспечение состоит из электронных схем, памяти, устройств ввода-вывода, т.е. из осязаемых объектов.

Программное обеспечение состоит из алгоритмов и программ. В первых вычислительных машинах разница между программными и аппаратными средствами была очевидной. Со временем эта грань стала размываться. Сейчас можно говорить о том, что аппаратное и программное обеспечения логически эквивалентны. Разделение функций программного и аппаратного обеспечения определяется такими факторами как стоимость, скорость, надежность, а также частота ожидаемых изменений.

1.1.2.1. Изобретение микропрограммирования

У первых компьютеров было только два уровня архитектуры набора команд и цифровой логический уровень.

В 1951 году Морис Уилкс (Кембриджский университет) предложил идею трехуровневого компьютера. Такой компьютер должен иметь встроенный неизменяемый интерпретатор (микропрограмму), функция которого заключалась в выполнении программ посредством интерпретатора. Таким образом, аппаратное обеспечение должно было выполнять только микропрограммы с ограниченным набором команд. Электронные схемы существенно упростились, цена уменьшилась, а надежность возросла. К 70-м годам идея микропрограммирования стала преобладающей.

1.1.2.2 Изобретение операционной системы

Первые операционные системы появились в 60-е годы. Придумана она была для того, чтобы автоматизировать работу оператора. Однако создание операционной системы было первым шагом на пути в развитии новой виртуальной машины. К уровню архитектуры команд добавлялись новые команды и в итоге сформировался новый уровень. Некоторые команды нового уровня были идентичны командам предыдущего, но появились и новые команды, которые полностью отличались. Эти команды тогда назывались макросами ОС или вызовами супервизора. Сейчас используют термин системный вызов. Первые операционные системы были ориентированы для работы в пакетном режиме. В начале 60-х годов в МТИ разработали операционную систему, которая позволяла одновременно работать нескольким пользователям. В такой системе ресурсы центрального процессора разделялись между несколькими пользователями и такие системы назывались и сейчас называются системами с разделением времени.

1.1.2.3 Перемещение функциональности на уровень микрокоманд

С 1970 г. микропрограммирование стало обычным, производители вводили новые команды путем расширения микропрограммы, т.е. программными методами. Многие новые команды не представляли особой ценности, т.к. эти же действия можно было сделать уже имеющимися средствами. Однако новые команды могли выполнять операции быстрее или предоставляли дополнительные удобства пользователю. Например:

- Ускорение работы с массивами (индексная и косвенная адресация)
- Системы прерывания, которые дают команду процессору, как только закончилась операция ввода или вывода;
- Способность приостановить одну программу и начать другую, используя небольшое количество команд (переключение процесса);
- Специальные команды для обработки изображений, мультимедийных данных.

1.1.2.4 Устранение микропрограммирования

В 60-х – 70-х годах количество микропрограмм увеличивалось, но они работали все медленнее и медленнее, т.к. требовали значительного объема памяти. Пришло понимание того, что с устранением микропрограмм резко сократиться количество команд и компьютеры станут работать быстрее. Таким образом, компьютеры вернулись к тому состоянию, в котором они были до изобретения микропрограммирования.

Вывод: граница между аппаратным и программным обеспечением постоянно перемещается. Так же обстоит дело с уровнями – между ними нет четких границ.

1.2 Развитие компьютерной архитектуры

В ходе эволюции компьютерных технологий были разработаны сотни разных компьютеров. Многие из них давно забыты, в то время как влияние других на современные идеи оказалось весьма значительным. В этом разделе мы дадим краткий обзор некоторых ключевых исторических моментов, чтобы лучше понять, каким образом разработчики дошли до концепции современных компьютеров. Разумеется, мы рассмотрим только основные моменты развития, оставив многие подробности за скобками. Некоторые «исторические вехи» этого пути — компьютеры, которые мы будем рассматривать — представлены в табл. 1.1.

Таблица 1.1. Основные этапы развития компьютеров

Год выпуска	Название компьютера	Создатель	Примечания
1834	Аналитическая машина	Бэббидж	Первая попытка построить цифровой компьютер
1936	Z1	Зус	Первая релейная вычислительная машина
1943	COLOSSUS	Британское правительство	Первый электронный компьютер
1944	Mark I	Айкен	Первый американский компьютер общего назначения
1946	ENIAC I	Экерт/ Маушл	С этой машины начинается история современных компьютеров
1949	EDSAC	Уилкс	Первый компьютер с программами, хранящимися в памяти
1951	Whirlwind I	MTI	Первый компьютер реального времени
1952	IAS	Фон Нейман	Эта архитектура используется в большинстве современных компьютеров
1960	PDP-1	DEC	Первый мини-компьютер (продано 50 экземпляров)
1961	1401	IBM	Очень популярный компьютер для малого бизнеса
1962	7094	IBM	Лидер в области научных расчетов начала 1960-х годов
1963	B5000	Burroughs	Первая машина, разработанная для языка высокого уровня
1964	360	IBM	Первое семейство компьютеров
1964	6600	CDC	Первый суперкомпьютер для научных расчетов
1965	PDP-8	DEC	Первый мини-компьютер массового потребления (продано 50 000 экземпляров)
1970	PDP-11	DEC	Эти мини-компьютеры доминировали на компьютерном рынке в 70-е годы
1974	8080	Intel	Первый универсальный 8-разрядный компьютер на микросхеме

1974	CRAY-1	Cray	Первый векторный суперкомпьютер
1978	VAX	DEC	Первый 32-разрядный супермини-компьютер
1981	IBM PC	IBM	Началась эра современных персональных компьютеров
1981	Osborne-1	Osborne	Первый портативный компьютер
1983	Lisa	Apple	Первый ПК с графическим пользовательским интерфейсом
1985	386	Intel	Первый 32-разрядный предшественник линейки Pentium
1985	MIPS	MIPS	Первый компьютер RISC
1985	XC2064	Xilinx	Первая программируемая вентильная матрица (FPGA)
1987	SPARC	Sun	Первая рабочая станция RISC на основе процессора SPARC
1989	GridPad	Grid Systems	Первый коммерческий планшетный компьютер
1990	RS6000	IBM	Первый суперскалярный компьютер
1992	Alpha	DEC	Первый 64-разрядный ПК
1992	Simon	IBM	Первый смартфон
1993	Newton	Apple	Первый карманный компьютер
2001	POWER4	IBM	Первая двухъядерная многопроцессорная микросхема

1.2.1. Нулевое поколение – механические компьютеры (1642-1945)

Первую счетную машину создал французский ученый Блез Паскаль в 1642 году. Ему тогда было 19 лет, он создал машину для своего отца, сборщика налогов. Машина могла выполнять только операции сложения и вычитания. Тридцать лет спустя великий немецкий математик Готфрид Вильгельм Лейбниц (1646-1716) построил счетную машину, которая помимо операций сложения и вычитания могла выполнять операции умножения и деления. Еще через 150 лет профессор математики Кембриджского университета Чарльз Бэбидж (1792-1871) (изобретатель спидометра) разработал и сконструировал разностную машину, предназначенную для подсчета таблиц чисел морской навигации. Машина могла выполнять только один алгоритм – метод конечных разностей с использованием полиномов. Машина, выполнявшая только один алгоритм, и Бэбидж начал разрабатывать (и

потратил на это очень много средств) аналитическую машину. У аналитической машины было запоминающее устройство, вычислительное устройство, устройство ввода (для перфокарт), устройство вывода (перфоратор и печатающее устройство). Машина могла выполнять разные задачи. Она считывала команды с перфокарт и выполняла их. Поскольку аналитическая машина программировалась на ассемблере, то ей было необходимо программное обеспечение. Первый программист – племянница поэта Байрона Ада Ловлейс. В ее честь назван язык программирования АДА. Аналитическая машина была механической. Идеи Бэбиджа опередили эпоху в том смысле, что технологическая база не позволяла создавать устройства такой сложности с приемлемой надежностью.

В конце 30-х -- начале 40-х годов XX века счетные машины были сконструированы в Германии и Америке, в которых были использованы электромагнитные реле.

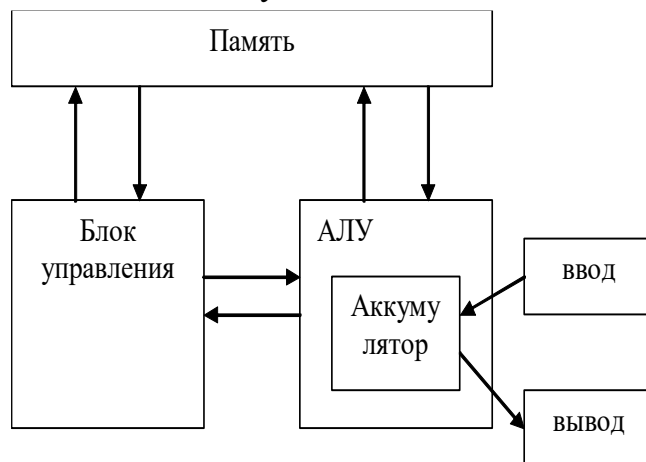
Машина Джона Атанасова была очень развита для своего времени. В ней использовалась бинарная арифметика, информационные емкости, которые периодически обновлялись. К сожалению, эта машина так и не заработала.

Говард Айкен, опираясь на исследования Бэбиджа, решил создать такой же компьютер, но на основе реле. Работа над первым компьютером была закончена в 1944 году. Называлась машина «MARK 1». Затем началась эра электроники.

4.2.2 Первое поколение – электронные лампы (1945-1955)

Стимулом для разработки электронного компьютера стала Вторая мировая война. Машина создавалась для шифровки и дешифровки в Великобритании. Одним из создателей этой машины был Алан Тьюринг.

В Америке Джон Моушли и Дж. Преспер Экерт начали конструировать компьютер, предназначенный в первую очередь для составления таблиц для нацеливания тяжелой артиллерии. К моменту завершения разработки война закончилась, машина стала не нужна для военных целей, и разработчикам было разрешено организовать школу, где они рассказывали о своей работе. Эта машина – ENIAC. Патент на ЦВМ они не получили, т.к. приоритет был отдан Атанасову.



В это же время в Институте специальных исследований в Принстоне один из участников проекта ENIAC Джон фон Нейман, сконструировал свою версию компьютера.

Он предложил размещать программу вместе с данными в оперативной памяти и использовать бинарную арифметику. Основной проект известен теперь как фон-

неймановская вычислительная машина. Он был использован в машине EDSAC. Практически все современные компьютеры являются фон-неймановскими машинами. Схема архитектуры этой машины приведена на рисунке. Машина не имела операций с плавающей точкой. Нейман полагал, что любой сведущий математик способен держать плавающую точку в голове.

Приблизительно в это же время в МТИ был создан компьютер Whirlwind-1. Особенности компьютера: слова небольшой длины (16 бит) и работа в PMB. Он является прототипом мини-компьютера.

В 1953 году IBM создала свой первый компьютер IBM-704.

4.2.3. Второе поколение – транзисторы (1955-1965)

Транзисторы были изобретены в лаборатории Bell Джоном Бардином, Уолтером Браттейном и Уильямом Шокли, за что в 1956 году им была присуждена Нобелевская премия. Первый компьютер на транзисторах был построен в МТИ и назывался TX-1, а затем TX-2. Практического значения эти компьютеры не имели, но один из разработчиков, Кеннет Ольсен в 1957 году основал фирму DEC и произвели первую серийную машину на транзисторах PDP-1 (1961 г.). Эта была самая быстродействующая машина того времени. Время цикла – 5 микросекунд. Это в два раза меньше, чем у IBM-7090 (транзисторного аналога IBM-709). Стоил PDP-1 \$120 000, а IBM – миллионы. Компания DEC продала десятки компьютеров PDP и так возникла компьютерная промышленность. Один из компьютеров был отдан в МТИ, где был создан первый графический дисплей, а студенты написали первую компьютерную игру – «Война миров».

Затем была создана машина PDP-8, которая была 12-разрядной, стоила \$16 000, а главное нововведение – одна шина. Шина – это набор параллельно соединенных проводов для связи компонентов компьютера. Структура компьютера с общей шиной приведена на рисунке. Такая структура с тех пор используется во всех компьютерах.



В 1964 году компания CDC выпустила машину CDC-6600, которая имела производительность на порядок выше, чем IBM-7090 и ее более дешевый аналог IBM-1401. Высокая производительность обеспечивалась за счет того, что внутри центрального процессора находилась машина с высокой степенью параллелизма. Разработчиком этого компьютера был Сеймур Крей. Он посвятил свою жизнь созданию мощных компьютеров, которые сейчас

называются суперкомпьютерами. Это компьютеры CDC-6600, CDC-7600, Crey-1.

Разработчики упомянутых выше компьютеров занимались в первую очередь аппаратным обеспечением, стремясь повысить его надежность, быстродействие и снизить стоимость.

Следует отметить еще один проект – Burroughs B50000. Разработчики создавали компьютер с намерением программировать ее на языке Algol 60 (предшественник языка Pascal), сконструировав аппаратное обеспечение так, что бы упростить работу компилятору. Так появилась идея, что программное обеспечение тоже надо учитывать при разработке компьютера.

4.2.4 Третье поколение – интегральные схемы (1965-1980)

В 1958 году была изобретена кремниевая технология (изобретатель – Роберт Нойс). Компьютеры на интегральных схемах были меньшего размера, работали быстрее, стоили дешевле. Наиболее значительные следующие.

К 1964 г. Фирма IBM лидировала на рынке, но выпускаемые ей компьютеры были программно несовместимы. Компания сделала решительный шаг. Она выпустила серию компьютеров на транзисторах System 360, которые были предназначены как для научных, так и для коммерческих расчетов. System 360 содержала много нововведений. Это было семейство компьютеров с одним и тем же ассемблером. Каждая новая модель была больше и мощнее предыдущей. Идея создания семейств компьютеров вскоре стала популярной и в течение нескольких лет большинство компьютерных компаний выпустило целые серии сходных машин.

Еще одно нововведение – мультипрограммирование. В памяти располагалось несколько программ и пока одна программа ждала окончания ввода-вывода, другая выполнялась.

Мир микрокомпьютеров сделал также большой шаг вперед вместе с производством компьютеров PDP-11. Во многих отношениях PDP-11 была младшим братом IBM 360 по организации компьютера и наличию в семействе машин разной стоимости и производительности.

4.2.5. Четвертое поколение – сверхбольшие интегральные схемы

Появление СБИС в 80-х годах позволило помещать на одну плату сначала десятки тысяч, а затем и миллионы транзисторов. К 80-м годам цены на компьютеры упали на столько, что приобретать компьютеры смогли не только организации, но и отдельные люди. Началась эра персональных компьютеров. Первые персональные компьютеры продавались в виде комплектов, как правило, на базе Intel 8080. Программное обеспечение пользователь писал сам. Затем появилась операционная система CP/M. Эта ОС помещалась на дискету, включала систему управления файлами и

интерпретатор для выполнения пользовательских команд, которые набирались на клавиатуре.

Компания IBM, лидирующая в то время на рынке компьютеров, тоже решила заняться производством персональных компьютеров. Компьютер IBM PC появился в 1981 году и стал самым покупаемым в истории.

Первая версия IBM PC была оснащена операционной системой MS-DOS, которую выпускала крошечная компания Microsoft. Эта компания разработала также собственную ОС Windows, которая работала на базе MS-DOS.

В 90-х годах появились суперскалярные компьютеры. Первый 64-разрядный компьютер был выпущен в 1992 году (Alpha, DEC), но коммерческий успех был скромным – 64-разрядные компьютеры приобрели популярность только в спустя десятилетие и в качестве персональных серверов.

Также в середине 1980-х годов Росс Фримен (Ross Freeman) со своими коллегами из Xilinx разработал метод создания микросхем, который был экономичным и не требовался доступ к производству кристаллического кремния. Новая разновидность микросхем, называемая **программируемой вентиляционной матрицей** (FPGA, Field-Programmable Gate Array), содержала большое количество универсальных логических элементов, которые можно было «запрограммировать» на любую функцию, необходимую устройству. Благодаря своему выдающемуся новому подходу к проектированию оборудование FPGA по своей гибкости не уступает программному обеспечению. Компании-производители кристаллического кремния продолжают миллионами выпускать более быстрые, дешевые и менее энергоемкие микросхемы для массового применения. Однако FPGA остаются популярным средством построения аппаратного обеспечения для узкого круга пользователей — например, построения прототипов, мелкосерийных приложений и образования.

1.2.6 Пятое поколение – невидимые компьютеры

В 1981 году правительство Японии объявило о намерении выделить национальным компаниям 500 миллионов долларов на разработку компьютеров пятого поколения на основе технологий искусственного интеллекта. Однако проект, в целом, оказался несостоятельным.

Пятое поколение компьютеров материализовалось в виде малых по размерам компьютеров – карманных компьютеров и «невидимых компьютеров» -- компьютеров, встраиваемых в бытовую технику, банковские карточки и т.п. Компьютеры пятого поколения ассоциируются не с некоторой архитектурой, а парадигмой использования. В настоящее время этому явлению применяется термин «всепроникающая компьютеризация».

1.3 Типы компьютеров

1.3.1 Технологические аспекты

Закон технологического прогресса, известный как закон Мура (Гордон Мур – один из основателей Intel) утверждает, что количество транзисторов на одной микросхеме удваивается каждые 18 месяцев. По-существу закон Мура не закон, а эмпирическое наблюдение за развитием технологий, и, по мнению специалистов, продержится еще лет 10, может больше. Однако, существует физический предел, определяемый принципом неопределенности Гейзенберга.

Закон Мура связан с так называемым *эффективным циклом*. Т.е. новые возможности порождают новые потребности, для удовлетворения которых требуются новые возможности и т.д.

Еще один фактор развития компьютерных технологий – первый закон программного обеспечения, названный в честь Натана Мирвольда (главный администратор Microsoft). Закон гласит; программное обеспечение это газ, который полностью заполняет резервуар, в котором находится. Т.е. программное обеспечение продолжает развиваться и требовать все больше ресурсов.

4.3.2 Широкий спектр компьютеров

Исследователь из лаборатории Bell Ричард Хамминг заметил, что количественное изменение характеристик на порядок ведет к качественному изменению. В компьютерных технологиях количественные характеристики изменились за 30 лет на 6 порядков. Компьютерные технологии развиваются как по пути увеличения мощности так и снижения цены на единицу мощности. Примерная классификация компьютеров приведена в таблице.

<i>Тип</i>	<i>Цена \$</i>	<i>Сфера применения</i>
«Одноразовые компьютеры»	0,5	Поздравительные открытки
Встроенные компьютеры	5	Часы, машины, бытовые приборы
Игровые компьютеры	50	Домашние компьютерные игры
Персональные компьютеры	500	Настольные и портативные компьютеры
Серверы	5 тыс.	Сетевые серверы
Комплексы рабочих станций	50 – 500 тыс.	Супер-мини компьютеры
Мэйнфреймы	5 млн.	Пакетная обработка данных в банке

1.3.2.1 Одноразовые компьютеры

Наибольшим достижением в этой области можно считать появление микросхем RFID (Radio Frequency Identification – радиочастотная

идентификация). Эти микросхемы без батареек, содержат приемо-передатчик, который по внешнему запросу выдает код. Это может быть использовано для снятия штрих-кодов, идентифицировать можно до конкретной единицы продукции, с увеличением объема памяти таких микросхем можно записывать и другие данные.

Микросхемы могут быть активными и пассивными, работать на разных частотах (чем выше частота, тем выше скорость передачи данных, но меньше радиус действия).

1.3.2.2 Микроконтроллеры

Микроконтроллеры выполняют функции управления устройствами и организации пользовательских интерфейсов. В отличие от RFID-микросхем, которые выполняют минимальный набор функций, микроконтроллеры представляют собой полноценные вычислительные устройства. Содержат процессор, память, устройства ввода-вывода. В ряде случаев программное обеспечение прошивается в памяти производителем.

Микроконтроллеры разделяются на универсальные и специализированные. Особенности микроконтроллеров; низкие цены, работа в реальном масштабе времени, жесткие ограничения на размер и электропотребление.

1.3.2.3 Игровые компьютеры

Это обычные компьютеры, в которых расширенные возможности графических и звуковых контроллеров сочетаются с ограниченным ПО и пониженной расширяемостью. Они оптимизированы на конкретную область применения – выполнение трехмерных игр. Все остальное считается вторичным – отсюда и низкая, по сравнению с ПК, цена.

1.3.2.4 Персональные компьютеры

Некоторые специалисты называют ПК с процессорами Intel, отделяя их от компьютеров, оснащенных высокопроизводительными RISC-микросхемами (Sun UltraSPARC), которые называют рабочими станциями. Но особой разницы между этими компьютерами нет.

К персональным компьютерам близки карманные компьютеры PDA.

1.3.2.5 Серверы

Мощные персональные компьютеры и рабочие станции часто используются в качестве сетевых серверов как в пределах локальной сети, так и в Интернете. Серверы поставляются в однопроцессорной или

мультипроцессорной конфигурациях. Некоторые серверы способны обрабатывать до миллиона транзакций в секунду.

С точки зрения архитектуры серверы ничем не отличаются от персональных компьютеров, только работают быстрее, имеют больше дискового пространства, устанавливают более быстрые сетевые соединения. Серверы работают под управлением тех же операционных систем, как правило Windows и UNIX.

1.3.2.6 Комплексы рабочих станций

COW-системы (Clusters Of Workstations) – кластеры рабочих станций состоят из нескольких персональных компьютеров или рабочих станций. Эти компьютеры соединены высокоскоростной сетью и снабжены специальным ПО. Предназначены для решения «больших» задач.

В виде кластеров могут быть организованы веб-серверы. Если частота обращения в страницам сайта исчисляется тысячами в секунду, то дешевле организовать кластер из нескольких сотен (или тысяч) серверов и распределить между ними нагрузку. Такие кластеры называются серверными фермами (server farms).

1.3.2.7 Мэйнфреймы

Это большие компьютеры, размером с комнату. В большинстве случаев эти системы – потомки больших компьютеров IBM 360/370. Обычно они работают не намного быстрее, чем мощные серверы, но у них выше скорость процессов ввода/вывода, обладают большим дисковым пространством. Эти системы дорогие. Многие компании считают, что выгоднее вложить деньги в такую систему, чем переписывать все программное обеспечение под персональные компьютеры. Именно из-за этих систем и возникла проблема 2000 года.

В последнее время под влиянием Интернета наблюдается возрождение мэйнфреймов как полноценной категории компьютеров. Они занимают нишу мощных серверов Интернета, способных обрабатывать огромное количество транзакций в секунду.

1.4 Семейства компьютеров

Основное внимание курса уделяется трем популярным архитектурам наборов команд (ISA): x86, ARM и AVR.

Архитектура x86 встречается практически во всех персональных компьютерах (PC с Windows и Linux, а также Mac) и серверных системах.

Архитектура ARM доминирует на мобильном рынке — большинство смартфонов и планшетных компьютеров использует процессоры ARM.

Наконец, архитектура AVR задействована в дешевых микроконтроллерах, встречающихся во многих встроенных системах. Встроенные компьютеры, хотя и незаметны для пользователей, контролируют работу многих агрегатов в автомобилях, телевизорах, микроволновых печах, стиральных машинах, да и вообще практически во всех электронных устройствах стоимостью выше 50 долларов.

1.4.1 Введение в архитектуру x86

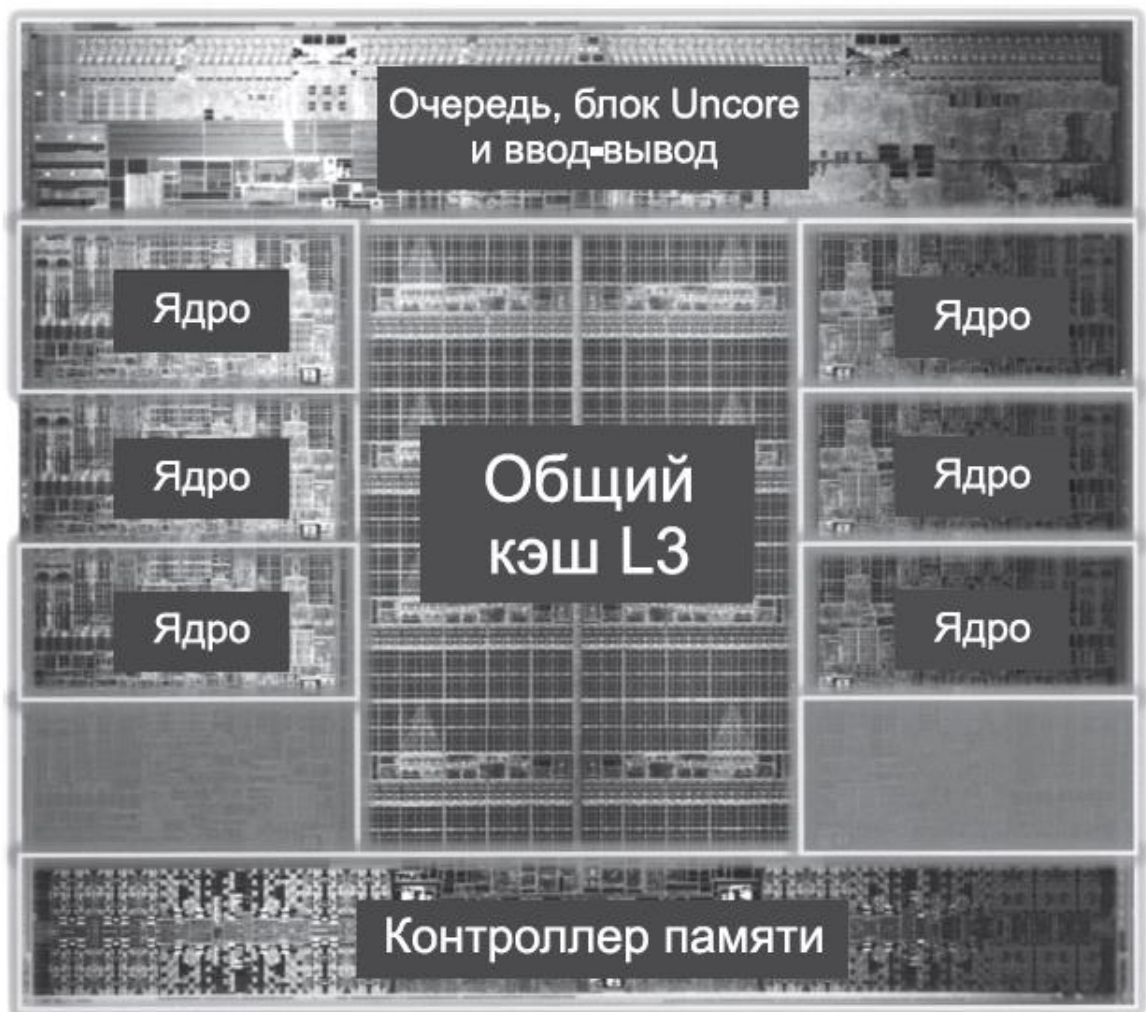
В 1968 году Роберт Нойс, изобретатель кремниевой интегральной схемы, Гордон Мур и Артур Рок, капиталист из Сан-Франциско, основали корпорацию Intel для производства микросхем. В 1969 году японская фирма Buscom обратилась к компании Intel с просьбой выпустить 12 несерийных микросхем. Инженер Тэд Хофф решил, что можно поместить 4-х битный универсальный процессор на одну микросхему. Так в 1970 году появился первый процессор на одной микросхеме, процессор 4004. В 1972 году Intel выпустила 8-битный процессор 8008.

Новая микросхема вызвала большой интерес и Intel начала разработку новой микросхемы, у которой предел обращения к памяти в 16 Кбайт был преодолен. Так появился 8080, выпущенный в 1974 году. Этот процессор произвел революцию.

Таблица 1.4. Семейство процессоров Intel. Тактовая частота измеряется в МГц
(1 МГц = 1 млн циклов/с)

Микро- схема	Дата выпуска	МГц	Количество транзисто- ров	Объем памяти	Примечание
4004	4/1971	0,108	2 300	640 байт	Первый микропроцессор на микросхеме
8008	4/1972	0,08	3 500	16 Кбайт	Первый 8-разрядный микропроцессор
8080	4/1974	2	6 000	64 Кбайт	Первый многоцелевой процессор на микросхеме
8086	6/1978	5—40	29 000	1 Мбайт	Первый 16-разрядный процессор на микросхеме
8088	6/1979	5—8	29 000	1 Мбайт	Использовался в IBM PC
80286	2/1982	8—42	134 000	16 Мбайт	Появилась защита памяти
80386	10/1985	16—33	275 000	4 Гбайт	Первый 32-разрядный процессор
80486	4/1989	25—400	1 200 000	4 Гбайт	Кэш-память на 8 Кбайт
Pentium	3/1993	60—223	3 100 000	4 Гбайт	Два конвейера, у более поздних моделей — MMX
Pentium Pro	3/1995	150—200	5 500 000	4 Гбайт ¹	Два уровня кэш-памяти
Pentium II	5/1997	233—400	7 500 000	4 Гбайт	Pentium Pro плюс MMX
Pentium III	2/1999	650—4400	9 500 000	4 Гбайт	Появились SSE-команды, ускоряющие обработку трехмерной графики
Pentium 4	11/2000	1300—3800	42 000 000	4 Гбайт	Гиперпоточность, дополнительные SSE-команды
Core Duo	1/2006	1600—3200	152 000 000	2 Гбайт	Два ядра на одной подложке
Core	7/2006	1200—3200	410 000 000	64 Гбайт	64-разрядная 4-ядерная архитектура
Core i7	1/2011	1100—3300	1 160 000 000	24 Гбайт	Интегрированный графический процессор

¹ Шина адреса у микропроцессоров Pentium Pro и Pentium II имеет ширину 36 бит, что позволяет непосредственно адресовать 64 Гбайт. – *Примеч. науч. ред.*



Микросхема Intel Core i7-3960X. Подложка имеет размеры 21×21 мм и содержит 2,27 миллиарда транзисторов

1.4.2 Введение в архитектуру ARM

В начале 1980-х годов английская компания Acorn Computer на волне успеха своего 8-разрядного персонального компьютера BBC Micro приступила к работе над второй машиной, надеясь составить конкуренцию недавно выпущенному IBM PC.

Проект был назван Acorn RISC Machine (сокращенно ARM — позднее, после отделения ARM от Acorn, расшифровка была заменена на «Advanced RISC Machine»). Работа была завершена в 1985 году.

Процессор использовал 32-разрядные команды и данные, имел 26-разрядное адресное пространство и производился фирмой VLSI Technology.

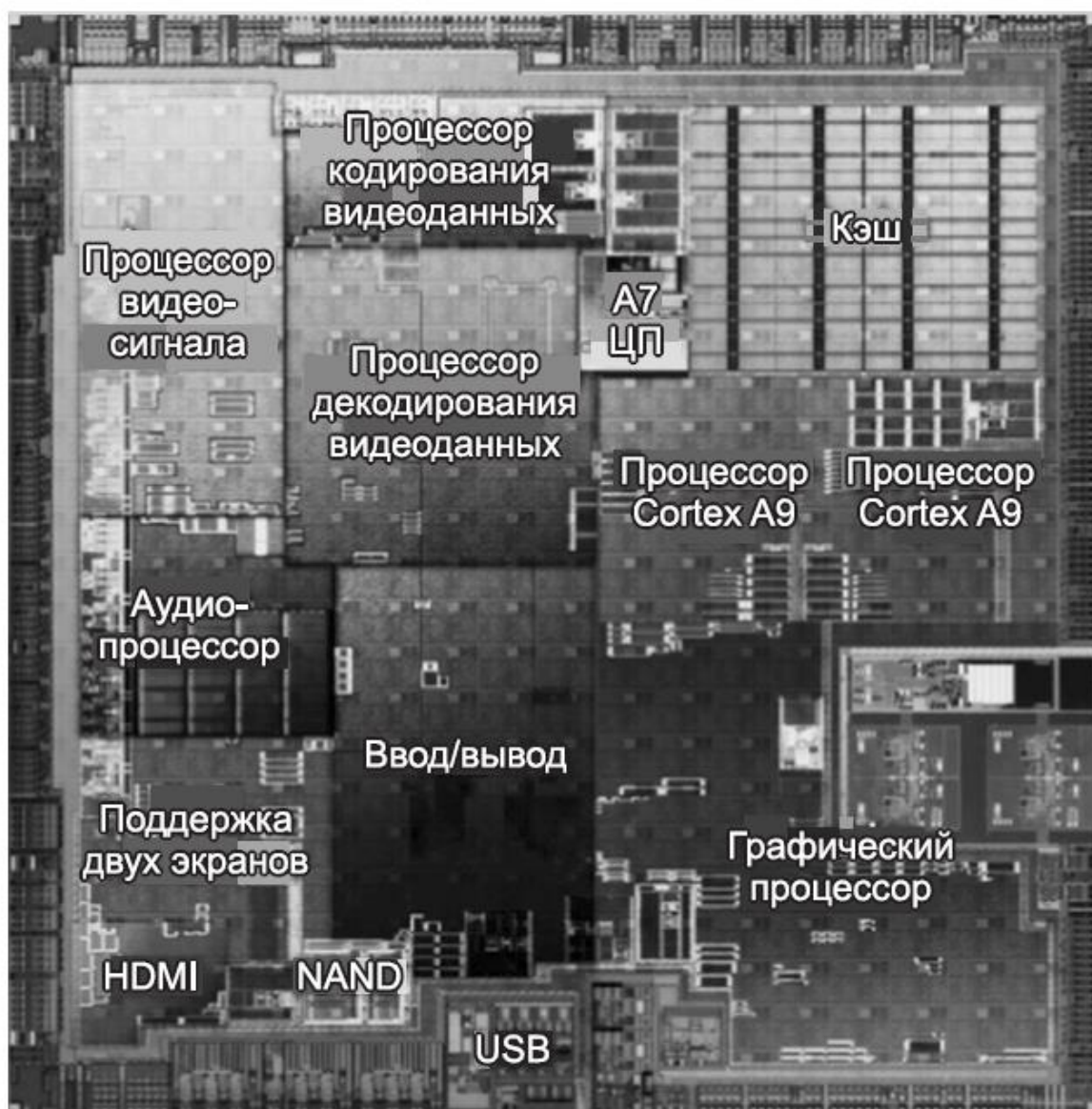
Первая архитектура ARM (названная ARM2) появилась в персональном компьютере Acorn Archimedes. Это была очень быстрая и недорогая для своего времени машина, которая выполняла до 2 MIPS (миллионов команд в секунду) и стоила на момент выпуска всего 899 фунтов. Машина стала весьма популярной в Великобритании, Ирландии, Австралии и Новой Зеландии, особенно в школах.

Видя успех Archimedes, фирма Apple обратилась к Acorn с предложением разработать процессор ARM для своего нового проекта Apple Newton. Группа ARM покинула Acorn и создала новую компанию, названную Advanced RISC Machines (ARM). В отличие от исходной архитектуры ARM, новый процессор включал 4-килобайтный кэш, существенно повышавший производительность.

В середине 1990-х годов фирма ARM совместно с Digital Equipment Corporation разработала высокоскоростную версию ARM с пониженным энергопотреблением для устройств с ограниченным. Они разработали архитектуру StrongARM, которая имела высокую скорость (233 МГц) и сверхнизкую мощность (1 ватт). Эффективность обеспечивалась простой, четкой структурой, включавшей два 16-килобайтных кэша для команд и данных.

Вероятно, самой известной из архитектур ARM стал процессор ARM7, который был выпущен ARM в 1994 году и продолжает широко использоваться в наши дни. Он содержит отдельные кэши команд и данных, а также реализует 16-разрядный набор команд Thumb — сокращенную версию полного 32-разрядного набора команд ARM, которая позволяет программировать многие стандартные операции в меньших 16-разрядных командах, существенно сокращая объем необходимой памяти.

ARM не производит микропроцессоры. Вместо этого фирма создает архитектуры, средства разработчика и библиотеки и продает лицензии на них разработчикам систем и производителям микросхем. Например, в планшетном компьютере Samsung Galaxy Tab на базе Android использовался процессор ARM.



1.4.3 Введение в архитектуру AVR

Архитектура AVR используется в чрезвычайно низкопроизводительных встроенных системах. История AVR начинается в 1996 году в Норвежском технологическом институте, когда студенты Алф-Эгиль Боген (Alf-Egil Bogen) и Вегард Воллан (Vegard Wollan) спроектировали 8-разрядный RISC-процессор, названный AVR.

Вскоре после завершения проектирования фирма Atmel купила разработку и открыла норвежское отделение, в котором два архитектора продолжали совершенствовать процессор AVR. Первый микроконтроллер AVR — AT90S1200 — был представлен Atmel в 1997 году. Чтобы упростить задачу проектировщиков систем, разводка контактов в точности соответствовала Intel 8051 — одного из самых популярных микроконтроллеров того времени. В наши дни к архитектуре AVR проявляется значительный интерес, потому что она заложена в основу чрезвычайно популярной платформы встроенных контроллеров Arduino.

Архитектура AVR реализована в трех классах микроконтроллеров, перечисленных в таблице.

Модель низшего класса — tinyAVR — спроектирована для большинства приложений с жесткими ограничениями по размерам, мощности и затратам. Она содержит 8-разрядный процессор, простейшую поддержку цифрового ввода-вывода и поддержку аналогового ввода.

В модели megaAVR, используемой в популярной открытой встроенной системе Arduino, также добавлена поддержка последовательного ввода-вывода, внутренние часы и программируемый аналоговый вывод.

Самой производительной моделью в этой категории является микроконтроллер AVR XMEGA, в котором также добавлен ускоритель криптографических операций и встроенная поддержка интерфейса USB.

Микро-схема	Флэш-память	EEPROM	Оперативная память	Контакты	Особенности
tinyAVR	0,5–16 Кбайт	0–512 байт	32–512 байт	6–32	Малые размеры, цифровой ввод-вывод, аналоговый ввод
megaAVR	8–256 Кбайт	0,5–4 Кбайт	0,25–8 Кбайт	28–100	Много периферийных устройств, аналоговый вывод
AVR XMEGA	16–256 Кбайт	1–4 Кбайт	2–16 Кбайт	44–100	Ускорение криптографических операций, ввод-вывод через USB

Микроконтроллеры обычно оснащаются тремя видами памяти: флэш-памятью, перепрограммируемой постоянной памятью (EEPROM, Electrically Erasable Programmable Read Only Memory) и оперативной памятью (RAM, Random Access Memory). Флэш-память программируется через внешний интерфейс с использованием высоких напряжений; в этой памяти хранится код и данные программы. Флэш-память является энергонезависимой, так что даже в случае отключения системы ее содержимое сохраняется. Память EEPROM тоже является энергонезависимой, но в отличие от флэш-памяти она может изменяться программой во время выполнения.

Наконец, в оперативной памяти хранятся переменные во время выполнения программы. Эта память не сохраняет информацию при выключении питания. Список периферийных подсистем для Atmel megaAVR-168:

1. Три таймера (два 8-разрядных и один 16-разрядный).
2. Часы реального времени с тактовым генератором.
3. Шесть каналов широтно-импульсной модуляции, используемых, например, для управления интенсивностью света или скоростью двигателя.
4. Восемь каналов аналогово-цифровых преобразований, используемых для чтения уровней напряжения.
5. Универсальный последовательный приемник/передатчик.
6. Последовательный интерфейс I2C — единый стандарт взаимодействия с датчиками.

7. Программируемый сторожевой таймер, обнаруживающий блокировку системы.
8. Встроенный аналоговый компаратор, сравнивающий два входных напряжения.
9. Детектор снижения напряжения, прерывающий работу системы при сбоях питания.
10. Внутренний программируемый генератор для управления тактовой частотой процессора.

1.4.4 Единицы измерения

Основные метрические приставки				Память	
Порядок	Приставка	порядок	Приставка	Объем	Название
10^{-3}	милли	10^3	кило	2^{10}	Кбайт
10^{-6}	микро	10^6	мега	2^{20}	Мбайт
10^{-9}	нано	10^9	гига	2^{30}	Гбайт
10^{-12}	пико	10^{12}	тера	2^{40}	Тбайт
10^{-15}	фемто	10^{15}	пета	10^3 бит/с	Кбит/с
10^{-18}	атто	10^{18}	экса	10^6 бит/с	Мбит/с
10^{-21}	зепто	10^{21}	зета	10^9 бит/с	Гбит/с
10^{-24}	йокто	10^{24}	йота	10^{12} бит/с	Тбит/с