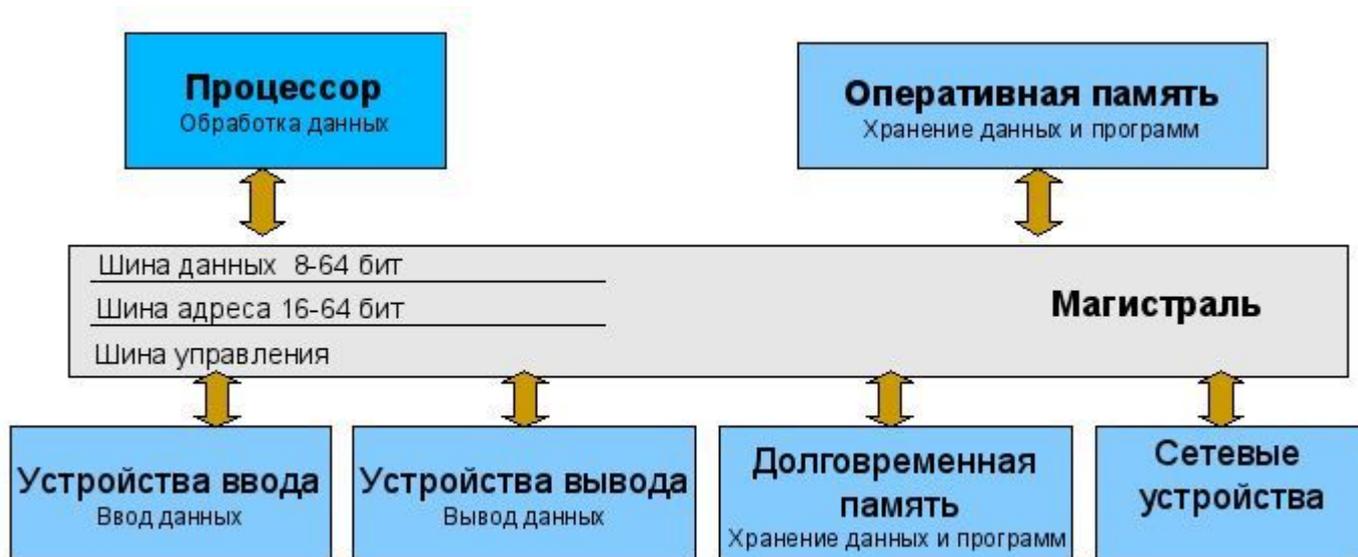


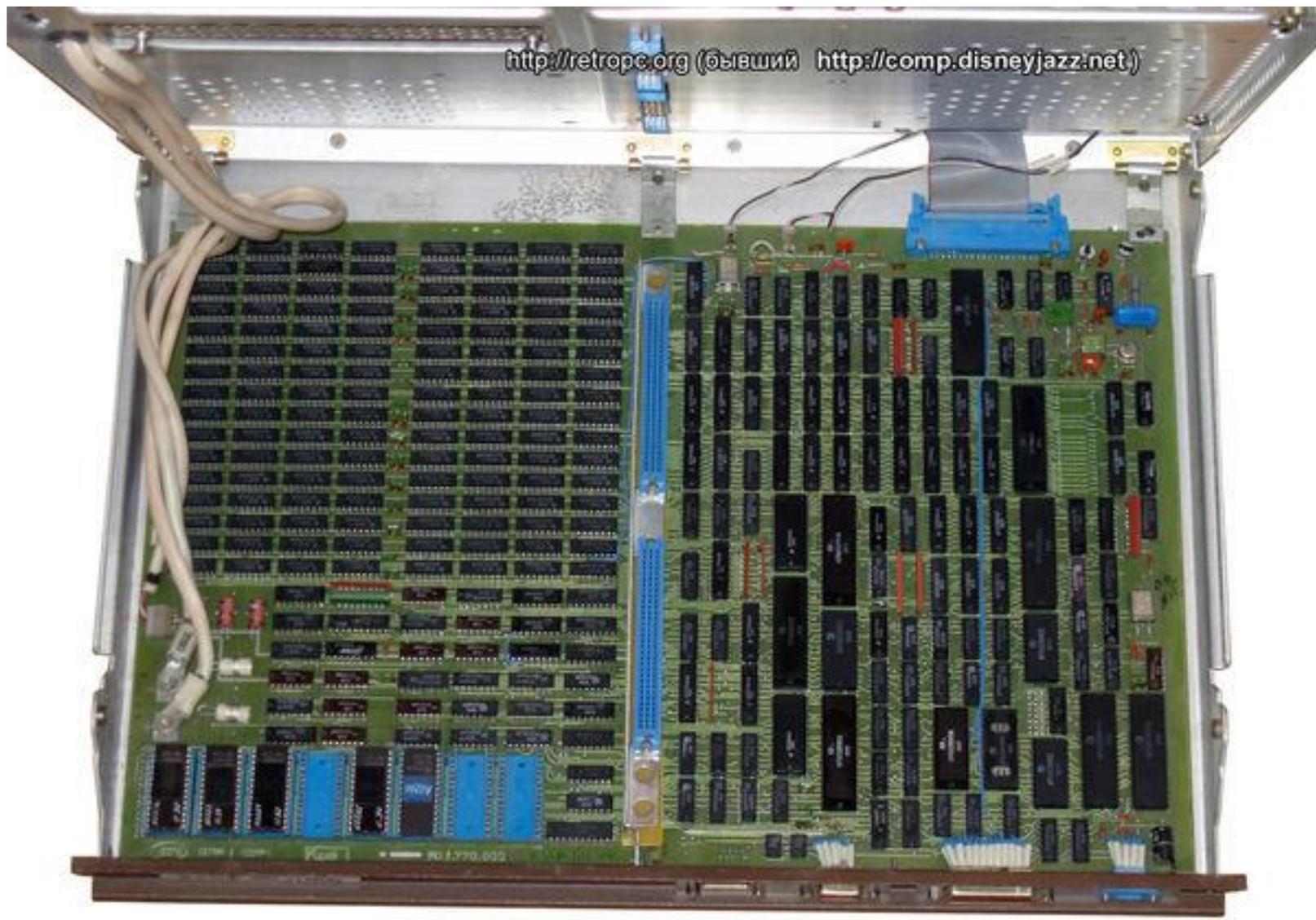
Микроконтроллеры

Классификация и структура микроконтроллеров

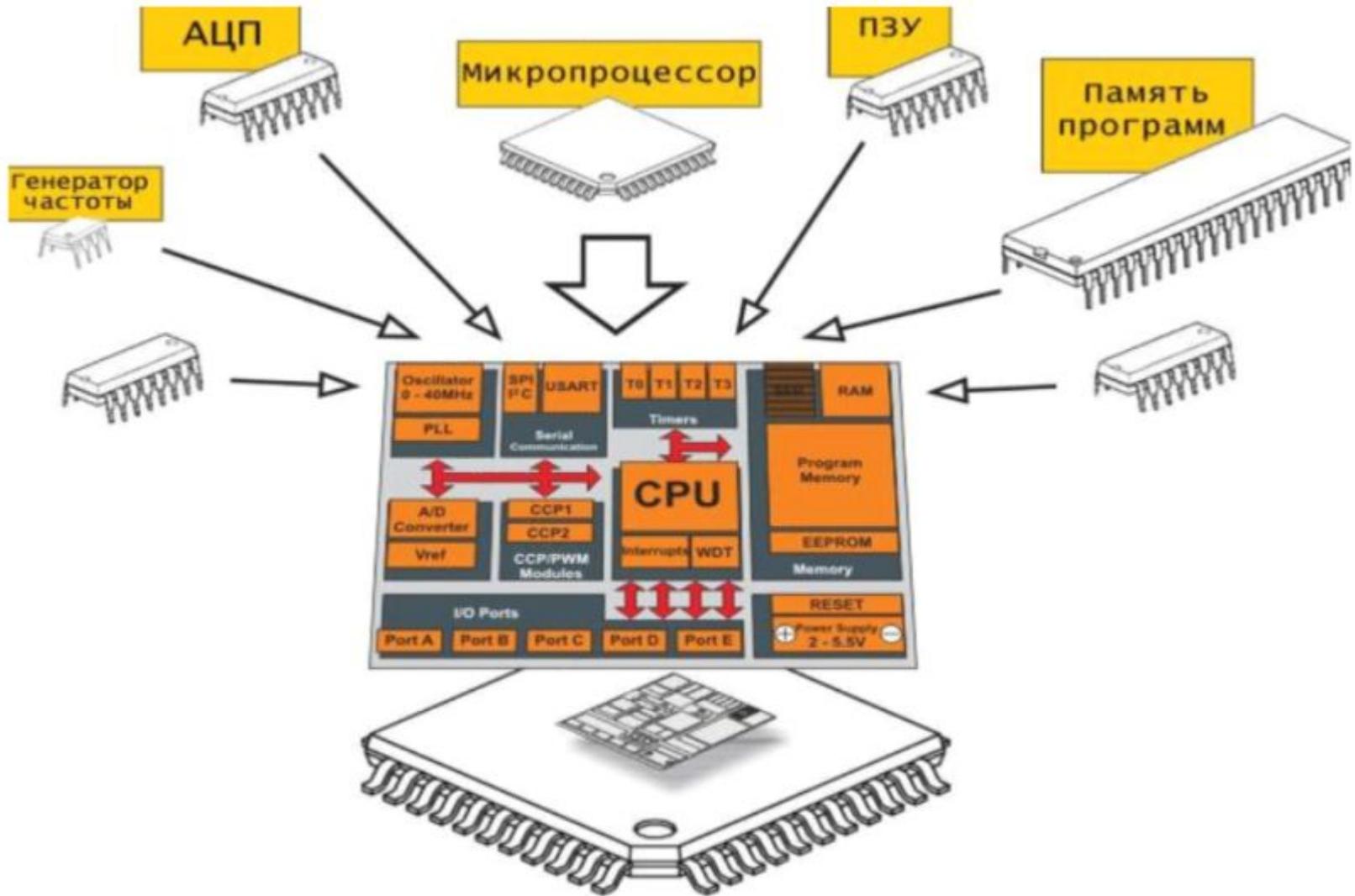
Основная особенность современных микропроцессорных систем (МПС) состоит в завершении перехода **от систем, выполненных на основе нескольких больших интегральных схем (ИС), к однокристалльным системам**, которые объединяют в одном кристалле все основные элементы ЭВМ:

- **центральный процессор (ЦП),**
- **постоянное запоминающее устройство (ПЗУ),**
- **оперативное запоминающее устройство (ОЗУ),**
- **порты ввода/вывода (УВВ),**
- **устройства формирования временных интервалов - таймеры и т.д.**





Персональный компьютер ДВК-3816



Однокристальная микро ЭВМ

Микроконтроллер – это полноценный компьютер на одной микросхеме.

Предназначен для управления различными электронными устройствами и осуществления взаимодействия между ними в соответствии с заложеной в микроконтроллер программой. В отличие от микропроцессоров, используемых в персональных компьютерах, микроконтроллеры содержат встроенные дополнительные устройства. Эти устройства выполняют свои задачи под управлением микропроцессорного ядра микроконтроллера.



Основным классификационным признаком микроконтроллеров (МК) является разрядность данных, обрабатываемых арифметико-логическим устройством (АЛУ).

По этому признаку они делятся на 4-, 8-, 16-, 32- и 64-разрядные.

Все типы **МК** можно условно разделить на три основных класса:

- 8-разрядные **МК** для встраиваемых приложений;
- 16- и 32-разрядные управляющие **МК**;
- цифровые сигнальные процессоры (DSP) для обработки данных.

Наиболее распространенным представителем семейства **МК** являются **8-разрядные приборы**. потому что основная область их применения находится **в системах управления реальными объектами**, где применяются, в основном, **алгоритмы с преобладанием логических операций**, скорость обработки которых практически не зависит от разрядности процессора.

Основные фирмы производители

1. **Intel** – i8048, **i8051** (CISC архитектура)
2. **Atmel** – серия **AT89** (CISC архитектура),
серия **AVR** AT90xx, Atmega, ATxmega (RISC архитектура)
3. **Microchip** – серия **PIC** (PIC12xx, PIC16xx и т.д.) (RISC архитектура)
4. **Motorola** – серия **M68** (M68HC05, HC08, HC11) (CISC архитектура)
5. **Texas Instruments** – серия **TMS320** – DSP (RISC архитектура)
6. **STMicroelectronics** – серия STM8, STM32 (RISC архитектура)

Тактовая частота, определяет, сколько вычислений может быть выполнено за единицу времени. В основном **производительность микроконтроллера** и **потребляемая им мощность** увеличиваются с повышением тактовой частоты.

Производительность микроконтроллера измеряют в **MIPS** (Million Instructions per Second – миллион инструкций в секунду)

Первый микроконтроллер

Первый микроконтроллер появился на свет в 1976 году, через 5 лет после создания первого микропроцессора Intel 4004.

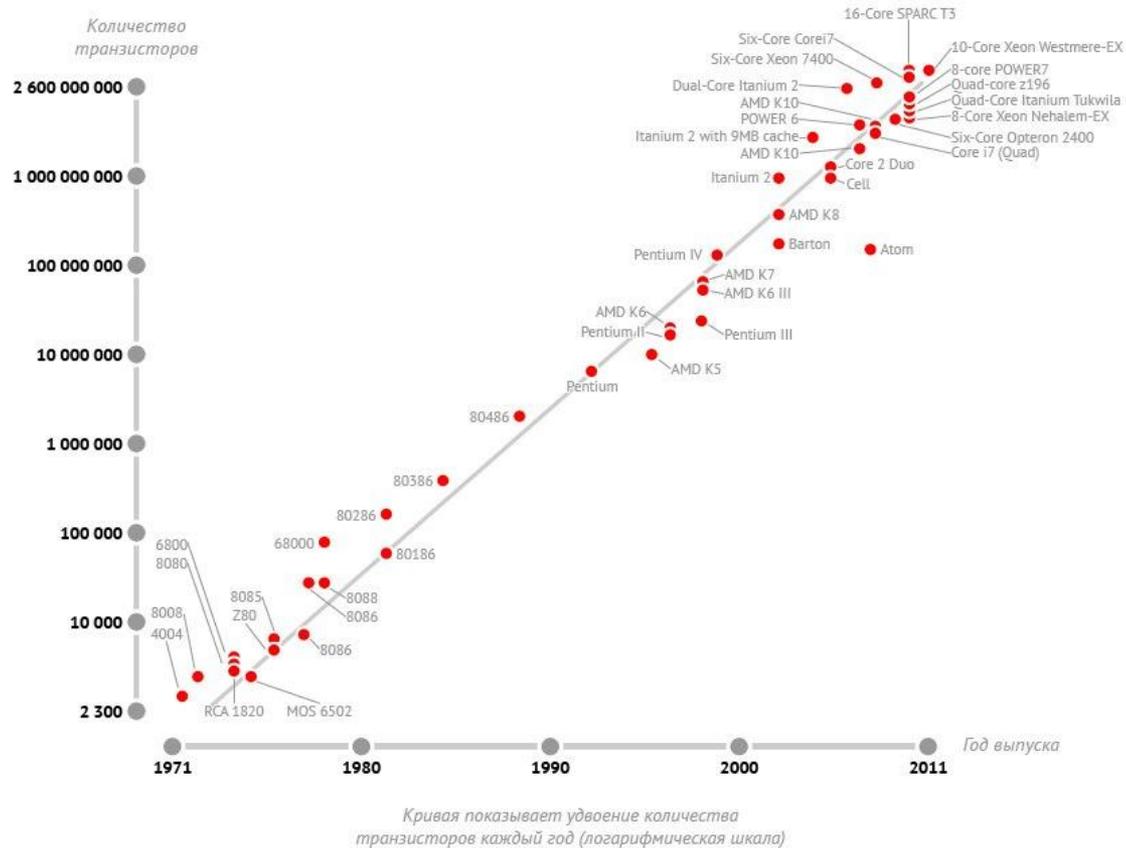


Это была микросхема фирмы **Intel**, получившая имя **i8048**.

Помимо центрального процессора, на кристалле находились 1 кбайт памяти программ, 64 байта памяти данных, два 8-битных таймера, тактовый генератор и 27 портов ввода/вывода.

Микроконтроллеры семейства 8048 использовались в игровых консольных приставках Magnavox Odyssey, в клавиатурах первых IBM PC и в ряде других устройств.

Закон Мура и количество транзисторов в процессорах



Закон Мура. В 1965 году один из основателей Intel *Гордон Мур* отметил, что новые модели микросхем разрабатывались спустя более-менее одинаковые периоды времени — 18-24 месяца — после появления их предшественников, при этом количество используемых в них транзисторов при этом возрастало каждый раз примерно вдвое.

Наблюдение Мура впоследствии блестяще подтвердилось, а обнаруженная им закономерность наблюдается и в наши дни.

За 40 лет, истекшие с момента появления микропроцессора **i4004** в 1971 году и вплоть до выпуска процессора Xeon Westmere-EX®, количество транзисторов выросло более чем в 1,3 миллиона раз – с 2 300 до 2,6 миллиардов.

Отличительные признаки микроконтроллеров

1. **модульная организация**, при которой на базе одного **процессорного ядра** (центрального процессора) проектируется ряд (линейка) **МК**, различающихся объемом и типом **памяти программ**, объемом **памяти данных**, набором периферийных модулей, частотой синхронизации;
2. **использование закрытой архитектуры МК**, которая характеризуется отсутствием линий магистралей адреса и данных на выводах корпуса **МК**. Таким образом, **МК** представляет собой **законченную систему обработки данных**, наращивание возможностей которой с использованием параллельных магистралей адреса и данных не предполагается;
3. **использование типовых функциональных периферийных модулей** (таймеры, процессоры событий, контроллеры последовательных интерфейсов, аналого-цифровые преобразователи и др.), имеющих незначительные отличия в алгоритмах работы в **МК** различных производителей;
4. **расширение числа режимов работы периферийных модулей**, которые задаются в процессе инициализации регистров специальных функций **МК**.

При модульном принципе построения все **МК** одного семейства содержат **процессорное ядро**, одинаковое для всех **МК** данного семейства, и изменяемый функциональный блок, который отличает **МК** разных моделей.

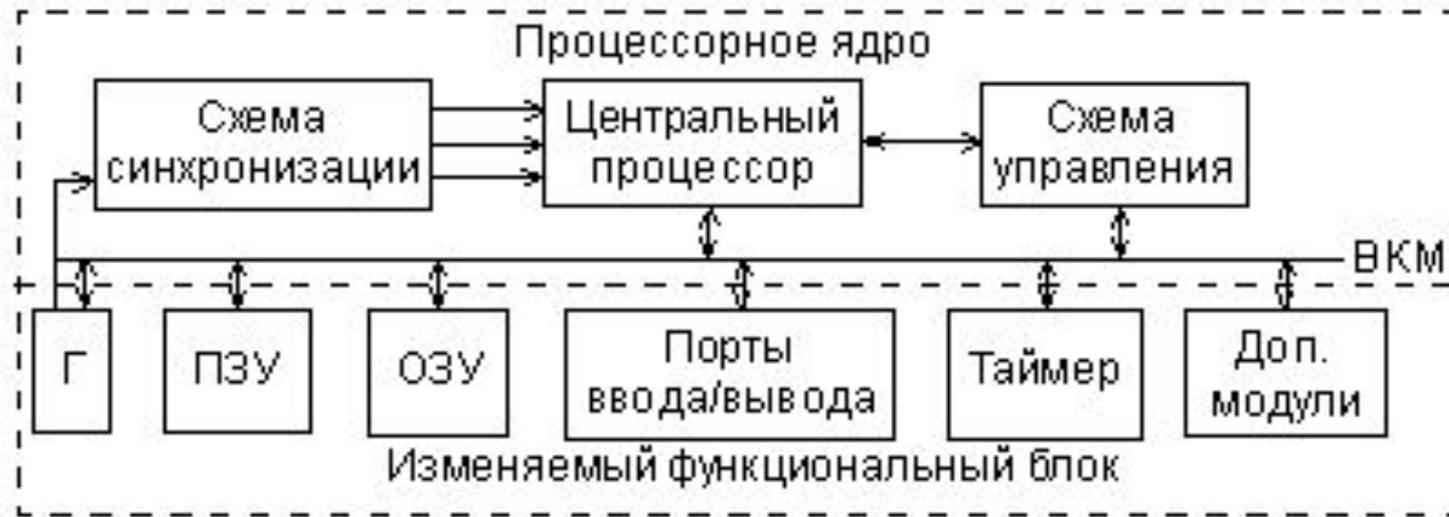


Рис. 1. Модульная организация МК

В состав процессорного ядра функционального блока могут входить и такие **дополнительные модули** как:

1. **центральный процессор;**
2. **порты параллельного и последовательного ввода-вывода,**
3. **внутреннюю контроллерную магистраль (ВКМ) в составе шин адреса, компараторы напряжения, данные и управления;**
4. **Аналого-цифровые преобразователи (АЦП)**
5. **схему синхронизации МК;**
6. **Цифро-аналоговые преобразователи (ЦАП)**
7. **схему управления режимами работы МК, включая поддержку режимов пониженного энергопотребления, начального запуска (сброса) и т.д.**
8. **Генераторы широтно-импульсной модуляции (ШИМ) и другие.**

INTEL

Наиболее массовый микроконтроллер **Intel 8051**, выпущенный в 1980 году, стал классическим образцом устройств данного класса. Этот 8-битный чип положил начало целому семейству микроконтроллеров, которые господствовали на рынке вплоть до недавнего времени.

Большинство фирм производителей микроконтроллеров и сегодня выпускают устройства, основанные на этой архитектуре – **x51**. Среди них Philips, Atmel, Dallas, OKI, Siemens и т.д. (можно перечислить более полутора десятков имен).

Но **семейство x51 (система команд MCS-51) постепенно сдает свои позиции** более молодым и совершенным микроконтроллерам.

Microchip

Первые значительные перемены произошли с появлением **PIC**-контроллеров фирмы Microchip. Эти чипы предлагались по рекордно низким ценам, что позволило им в короткий срок захватить значительную часть рынка микроконтроллеров. К тому же кристаллы от Microchip оказались не уступающими, а нередко и превосходящими микроконтроллеры **x51** по производительности и не требовали дорогостоящего программатора.

Atmel

Настоящая революция в мире микроконтроллеров произошла в 1996 году, когда корпорация Atmel представила свое семейство чипов на новом прогрессивном ядре **AVR**.

Более продуманная архитектура AVR, быстродействие, превосходящее контроллеры Microchip, привлекательная ценовая политика и доступные средства разработки способствовали оттоку симпатий многих разработчиков от недавних претендентов на звание контроллера номер 1.

Микроконтроллеры AVR имеют **более развитую систему команд**, насчитывающую до 133 инструкций, высокую **производительность**, приближающуюся к 1 MIPS/МГц, **Flash ПЗУ программ с возможностью внутрисхемного перепрограммирования**. Многие чипы имеют функцию самопрограммирования.

AVR-архитектура оптимизирована под язык высокого уровня Си.

Кроме того, все кристаллы семейства совместимы "снизу вверх".

У Atmel много бесплатно распространяемых программных продуктов. Фирма Atmel уделяет этому вопросу большое внимание. Чрезвычайно удачная и совершенно бесплатная среда разработки **Atmel Studio**, работающая под Windows. Немаловажным является и то, что для программирования AVR можно обойтись вовсе без аппаратного программатора.

AVR становится еще одним индустриальным стандартом среди 8-разрядных микроконтроллеров общего назначения.

МИКРОКОНТРОЛЛЕРЫ ATMEL AVR: ИСТОРИЯ СОЗДАНИЯ

Идея разработки нового прогрессивного RISC-ядра зародилась в норвежском городе Тронхейм в головах двух студентов Norwegian University of Science and Technology (NTNU) **Альфа Богена** и **Вегарда Воллена**. Они создали такую архитектуру микроконтроллера, которая стала одной из самых удачных на мировом рынке микроконтроллеров.

В 1995 году Боген и Воллен предложили американской корпорации Atmel, известной на тот момент своим "ноу-хау" изготовления чипов с Flash-памятью, выпускать новый 8-битный RISC-микроконтроллер и снабдить его Flash-памятью программ на кристалле. Идея настолько понравилась руководству Atmel Corp., что было принято решение незамедлительно инвестировать в данный проект.

В конце 1996 года был выпущен опытный кристалл **AT90S1200**, а во второй половине 1997-го корпорация Atmel приступила к серийному производству нового семейства микроконтроллеров, к их рекламной и технической поддержке.

Новое ядро было запатентовано и получило название AVR, которое по прошествии уже нескольких лет стало трактоваться самыми различными способами. Кто-то утверждает, что это не иначе как **Advanced Virtual RISC**, другие полагают, что не обошлось здесь без **Alf Egil Bogen Vegard Wollan RISC**.

Держателями патента при этом являются: Wollan Vegard (NO); Bogen Alf-Egil (NO); Myklebust Gaute (NO); Bryant John, D. (US).

Система команд и внутреннее устройство чипов AVR разрабатывались совместно с фирмой **IAR Systems** – производителем компиляторов языков программирования C/C++, что обеспечило уникальные характеристики этих микроконтроллеров. В результате для AVR стало возможным получать **высокую плотность кода при использовании языков высокого уровня**, практически не теряя в производительности по сравнению с программами, написанными на низкоуровневом языке Ассемблера.

Использование прогрессивной технологии конвейеризации у AVR сокращало цикл "выборка – исполнение" команды.

Например, у микроконтроллеров семейства **x51** короткая команда выполняется за **12 тактов** генератора. В **PIC**-контроллерах фирмы Microchip, где уже реализован конвейер, короткая команда выполняется за **4 периода** тактовой частоты. В микроконтроллерах **AVR** короткая команда в общем потоке выполнялась всего за **один период** тактирующего сигнала. Такое построение кристалла обеспечило существенное повышение производительности, которая в пределе может достигать значения 1MIPS на 1МГц. **Это** во многих случаях при заданной производительности **позволяло снизить тактовую частоту**, а значит, **и потребляемую мощность устройства**.

AVR-микроконтроллеры предоставляли **более широкие возможности по оптимизации производительности/энергопотребления**, что было особенно важно при разработке приложений с батарейным питанием.

В 2003 году количество выпущенных Atmel Corp. микросхем с ядром AVR превысило **500 миллионов штук!**

МОДЕЛИ МИКРОКОНТРОЛЛЕРОВ AVR

В настоящее время в серийном производстве у Atmel находятся семейства AVR ATtiny, ATmega и ATxmega. Также все еще можно встретить в продаже чипы семейства Classic типа AT90.

Classic AVR – это классические (первые) модели AVR, что следует и из их названия. В настоящее время они представлены специализированными микроконтроллерами, например для работы с USB, CAN, для формирования многоканального ШИМ сигнала.

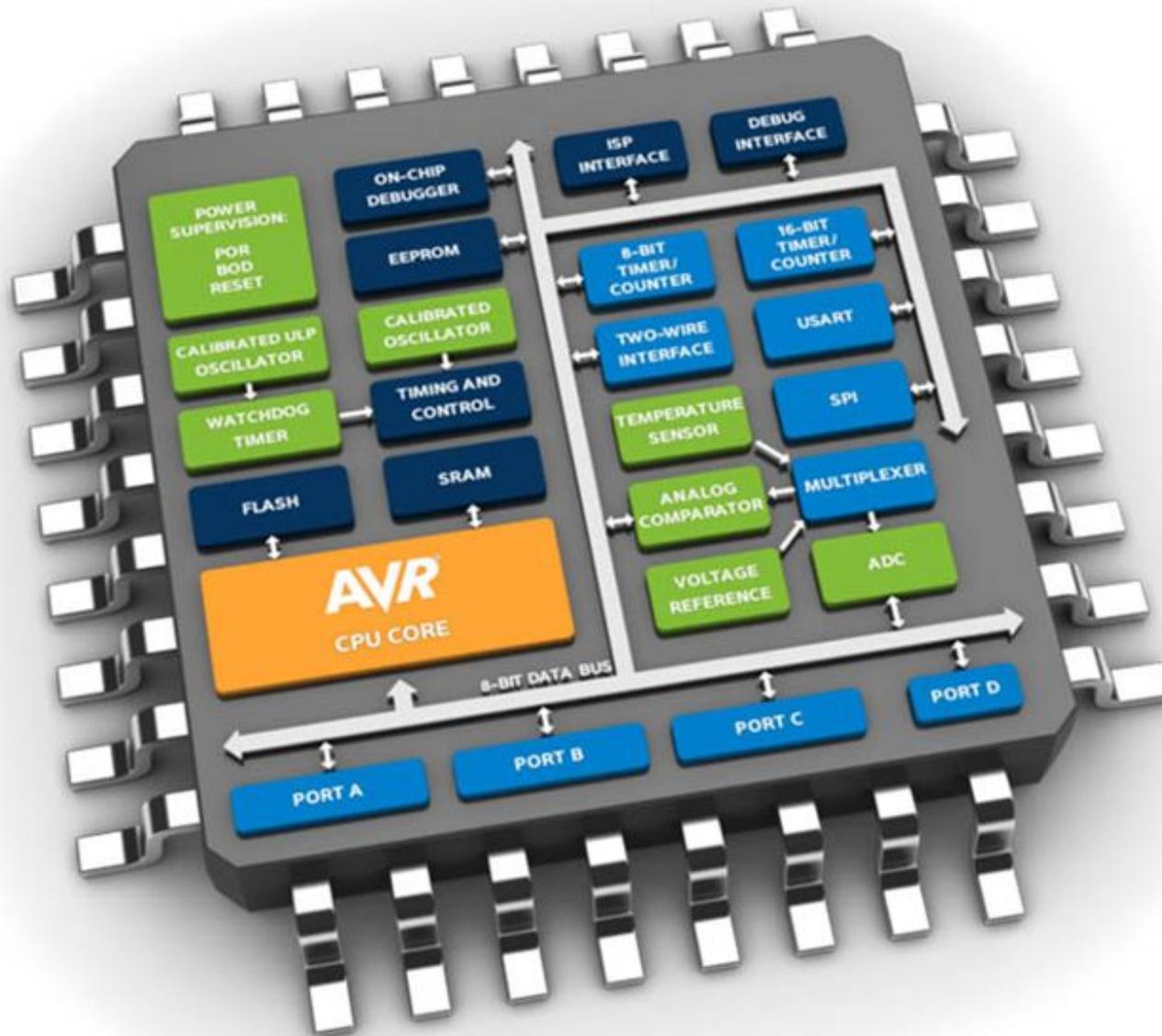
Tiny AVR – недорогие микроконтроллеры с малым количеством выходных контактов, чаще в восьмивыводном исполнении. Используются в системах, где не требуются сложные вычислительные алгоритмы.

Mega AVR – мощные микроконтроллеры, включающие большой набор периферии. Это семейство имеет самое большое разнообразие моделей для выбора.

AVR XMEGA – новое семейство компании Atmel переносит 8/16-битные микроконтроллеры на новый уровень системных характеристик. В этой связи, микроконтроллеры AVR XMEGA могут выступать в качестве эталонных 8/16-битных микроконтроллеров.

Независимо от выбора семейства микроконтроллеров AVR, следует помнить, что **система команд всех семейств совместима**, поэтому возможен простой перенос программы со слабого на более мощный микроконтроллер.

УСТРОЙСТВО МИКРОКОНТРОЛЛЕРОВ AVR



Память

В микроконтроллерах AVR реализована архитектура, в соответствии с которой разделены не только адресные пространства памяти программ и памяти данных, но и шины доступа к ним. Каждая из областей памяти данных расположена в своем адресном пространстве.

Память программ

Память программ в микроконтроллере AVR является энергонезависимой реализована по *flash* технологии и предназначена для хранения последовательности команд, управляющих функционированием микроконтроллера, и имеет 16-ти битную организацию.

Все AVR имеют Flash-память программ, которая может быть различного размера от 1 до 256 КБайт.

Программа заносится во Flash-память AVR как с помощью специального устройства – программатора, так и с помощью последовательных интерфейсов – SPI- или JTAG. Практически все микроконтроллеры AVR обладают возможностью внутрисхемного программирования (**функция ISP in system programming**) через коммуникационный интерфейс SPI.

Все микроконтроллеры семейства Mega имеют возможность **самопрограммирования**, т. е. самостоятельного изменения содержимого своей памяти программ.

Память данных

Память данных разделена на три части:

1. **регистровая память** (набор внутренних регистров микроконтроллера),
2. **оперативная память** (ОЗУ – оперативное запоминающее устройство с произвольной выборкой или RAM – Random Access Memory),
3. **энергонезависимая память** (ЭСППЗУ – электрически стираемое перепрограммируемое постоянное запоминающее устройство или EEPROM – Electrically Erasable Programmable Read-Only Memory).

Регистровая память

Регистровая память включает **32 регистра общего назначения** (РОН или GPR), объединенных в файл, и служебные **регистры ввода/вывода** (РВВ). И те и другие расположены в адресном пространстве ОЗУ, но не являются его частью.

В области **регистров ввода/вывода** расположены различные **служебные регистры** (регистры управления микроконтроллером, регистры состояния и т. п.), а также **регистры управления периферийными устройствами**, входящими в состав микроконтроллера.

Управление микроконтроллером заключается в управлении этими регистрами

Энергонезависимая память данных

Для долговременного хранения различной информации, которая может изменяться в процессе функционирования микроконтроллерной системы, используется **EEPROM-память**. Все AVR имеют блок энергонезависимой электрически перезаписываемой памяти данных EEPROM от 64 байт до 4 кбайт. Этот тип памяти, доступный программе микроконтроллера непосредственно в ходе ее выполнения, удобен для хранения промежуточных данных, различных констант, коэффициентов, серийных номеров, ключей и т.п. EEPROM может быть загружена извне как через SPI интерфейс, так и с помощью обычного программатора. **Число циклов стирание/запись – не менее 100 тыс.**

Оперативная память

Внутренняя оперативная статическая память (Static RAM – SRAM) имеет байтовый формат организации и используется для оперативного хранения данных.

Размер оперативной памяти может варьироваться у различных чипов от 64 байт до 4 кбайт. Число циклов чтения и записи в RAM не ограничено, но при отключении питающего напряжения вся информация теряется.

Для некоторых микроконтроллеров возможна организация подключения внешнего статического ОЗУ объемом до 64 кбайт

Периферия микроконтроллера

Периферия микроконтроллеров AVR включает:

- **цифровые порты ввода/вывода** (от 3 до 48 линий ввода и вывода),
- поддержку **внешних прерываний (IRQ)**,
- **таймеры-счетчики (Т/С)**,
- **шиотно-импульсные модуляторы (PWM)**
- **сторожевой таймер (WD)**,
- **аналоговые компараторы (AC)**,
- **10-разрядный 8-канальный АЦП (ADC)**,
- Последовательные интерфейсы **UART, JTAG** и **SPI**,
- **устройство сброса** по понижению питания (**RESET**),

Цифровые порты ввода/вывода

Порт ввода-вывода – логическое объединение сигнальных линий, через которое принимаются и передаются данные.

В зависимости от реализуемых функций различают следующие типы параллельных портов:

- 1. однонаправленные порты**, предназначенные **только для ввода** или **только для вывода** информации;
- 2. двунаправленные порты**, **направление передачи** которых (ввод или вывод) **определяется в процессе инициализации МК**;
- 3. порты с альтернативной функцией** (мультиплексированные порты). Отдельные линии этих портов используются совместно со встроенными периферийными устройствами МК, такими как таймеры, АЦП, контроллеры последовательных интерфейсов;
- 4. порты с программно управляемой схемотехникой** входного/выходного буфера.

Порты ввода/вывода можно назвать Периферией Независимой от Ядра, так как входы портов являются асинхронными, состояние портов сохраняется в спящем состоянии ядра.

Входные элементы управления

В качестве входных элементов управления могут использоваться

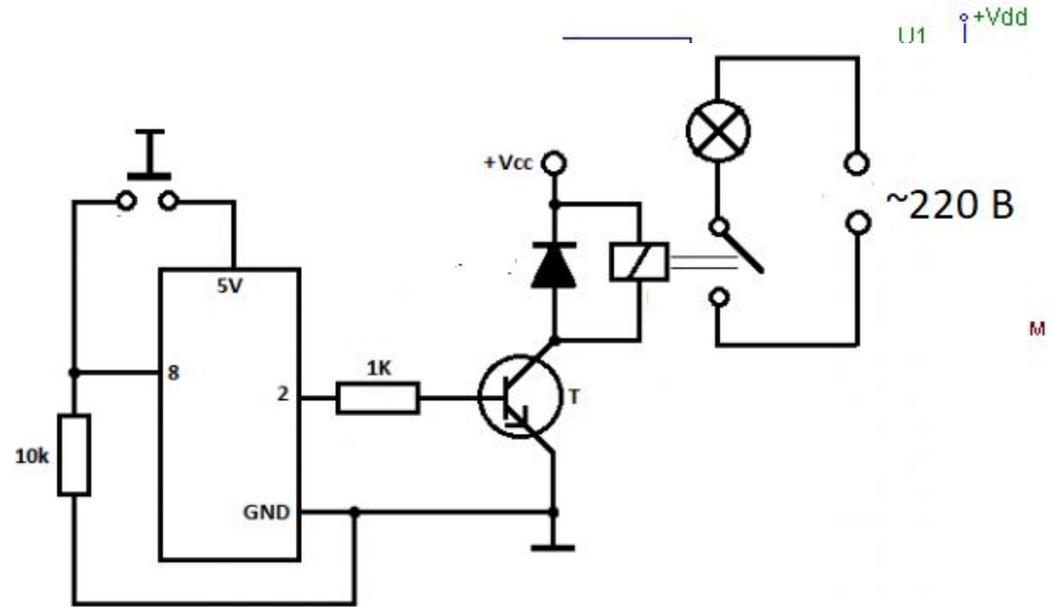
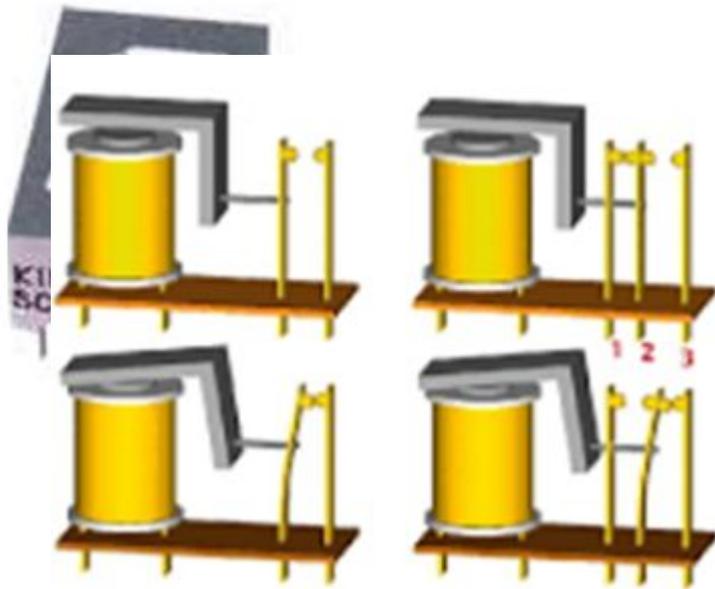
- тумблеры;
- джамперы;
- множественные переключатели;
- кнопки;
- другие элементы схемы.



Выходные элементы управления

В качестве выходных элементов управления могут использоваться

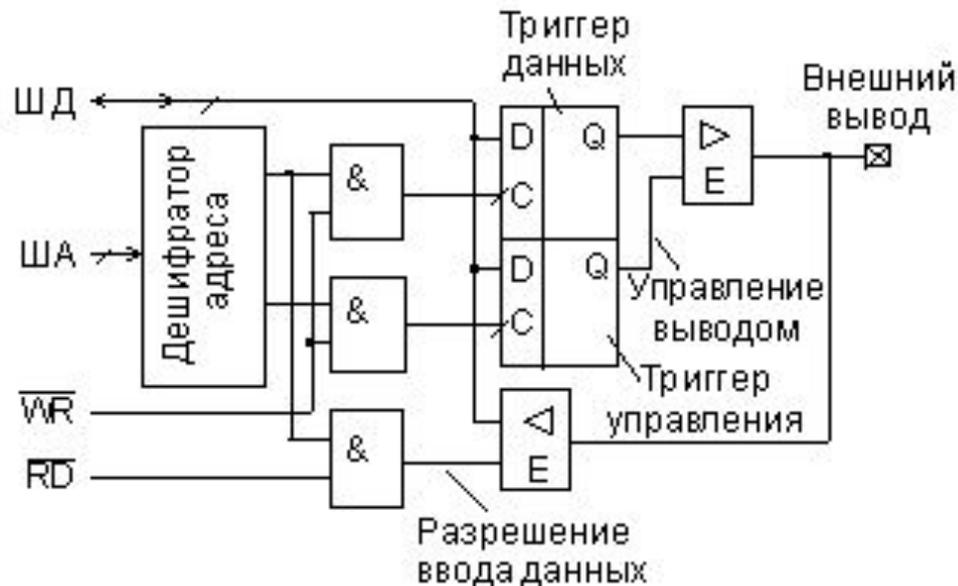
- элементы индикации (единичные светодиоды или светодиодные сборки);
- Электромагнитные реле;
- другие элементы схемы.



Порт в МК выполняет роль устройства согласования между самим **МК** и **объектом управления**, которые в общем случае **работают асинхронно**.

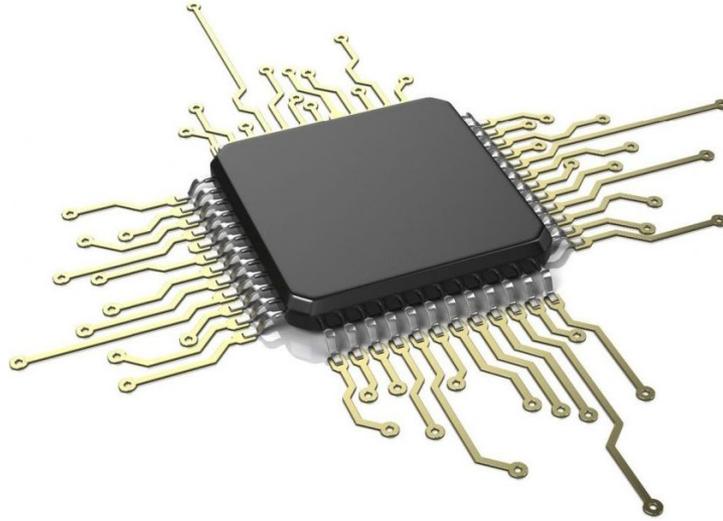
Три типа алгоритмов обмена информацией между МК и внешним устройством через параллельные порты ввода/вывода:

1. **режим простого программного ввода/вывода;**
2. **режим ввода/вывода со стробированием (синхронный);**
3. **режим ввода/вывода с полным набором сигналов подтверждения обмена (асинхронный).**



Типовая схема двунаправленного порта ввода/вывода МК.

Порты ввода/вывода AVR имеют от 3 до 53 независимых линий "вход/выход".



Каждая линия порта может быть запрограммирована на вход или на выход.

Мощные выходные драйверы обеспечивают высокую токовую нагрузочную способность 20 мА на линию порта (втекающий ток) при максимальном значении **40 мА**, что позволяет, например, непосредственно подключать к микроконтроллеру светодиоды и биполярные транзисторы. **Общая токовая нагрузка на все линии одного порта не должна превышать 80 мА** (все значения приведены для напряжения питания 5 В).

Архитектурная особенность построения портов ввода/вывода у AVR заключается в том, что **для каждого физического вывода (пина, ножки) существует 3 бита контроля/управления**, а не 2, как у других распространенных 8-разрядных микроконтроллеров. Это повышает скорость работы микроконтроллера при работе с внешними устройствами, особенно в условиях внешних электрических помех.

Прерывания (INTERRUPTS)

Система прерываний – одна из важнейших частей микроконтроллера.

Все микроконтроллеры AVR имеют многоуровневую систему прерываний. Прерывание прекращает нормальный ход программы для выполнения приоритетной задачи, определяемой внутренним или внешним событием.

Для обработки каждого такого события разрабатывается отдельная программа, которую называют подпрограммой обработки запроса на прерывание (для краткости – подпрограммой прерывания), и размещается в памяти программ.

При возникновении события, вызывающего прерывание, микроконтроллер сохраняет содержимое счетчика команд, прерывает выполнение центральным процессором текущей программы и переходит к выполнению подпрограммы обработки прерывания.

После выполнения подпрограммы прерывания осуществляется восстановление предварительно сохраненного счетчика команд и процессор возвращается к выполнению прерванной программы.

Для каждого события может быть установлен приоритет. Понятие приоритет означает, что выполняемая подпрограмма прерывания может быть прервана другим событием только при условии, что оно имеет более высокий приоритет, чем текущее. В противном случае центральный процессор перейдет к обработке нового события только после окончания обработки предыдущего.

Таймеры/счетчики

Микроконтроллеры AVR имеют в своем составе от 1 до 4 таймеров/счетчиков (TIMER/COUNTERS) с разрядностью 8 или 16 бит, которые могут работать и как таймеры от внутреннего источника тактовой частоты, и как счетчики внешних событий.

Их можно использовать:

1. для точного формирования временных интервалов,
2. подсчета внешних импульсов на выводах микроконтроллера,
3. формирования последовательности импульсов,
4. формировать широтно-импульсную модуляцию ШИМ (PWM) с программируемыми частотой и скважностью.

Таймеры/счетчики способны вырабатывать запросы на прерывания, переключая процессор на их обслуживание по событиям и освобождая его от необходимости периодического опроса состояния таймеров.

Поскольку основное применение микроконтроллеры находят в системах управления реального времени, **таймеры/счетчики являются одним из наиболее важных элементов таких систем.**

Сторожевой таймер

Сторожевой таймер (WDT – WatchDog Timer) предназначен для предотвращения катастрофических последствий от случайных сбоев программы. Он имеет свой собственный **RC-генератор**, работающий на частоте **1 МГц** от встроенного генератора.

Работа с аналоговыми сигналами

Аналоговый компаратор

Аналоговый компаратор (*Analog Comparator*) сравнивает напряжения на двух выводах (пинах) микроконтроллера. Результатом сравнения будет логическое значение, которое может быть прочитано из программы.

Выход аналогового компаратора можно включить на *прерывание от аналогового компаратора*. Пользователь может установить срабатывание прерывания по нарастающему или спадающему фронту или по переключению.

Присутствует у всех современных AVR, кроме Mega8515

Аналого-цифровой преобразователь

Аналого-цифровой преобразователь (АЦП, ADC – analog to digital converter) служит для получения числового значения напряжения, поданного на его вход. Этот результат сохраняется в регистре данных АЦП. Какой из выводов (пинов) микроконтроллера будет являться входом АЦП (их может быть до 8 у AVR), определяется числом, занесенным в соответствующий регистр.

Последовательные периферийные интерфейсы

Универсальный последовательный приемопередатчик (UART или USART)

Универсальный асинхронный или универсальный синхронно/асинхронный приемопередатчик (Universal Synchronous/Asynchronous Receiver and Transmitter – **UART** или **USART**) – удобный и простой последовательный интерфейс для организации информационного канала обмена микроконтроллера с внешним миром. Способен работать в дуплексном режиме (одновременная передача и прием данных). Он **поддерживает интерфейс стандарта RS-232**, что обеспечивает возможность организации связи с персональным компьютером. (Для стыковки МК и компьютера обязательно понадобится схема сопряжения уровней сигналов. Для этого существуют специальные микросхемы, например MAX232).

Последовательный периферийный интерфейс SPI

Последовательный периферийный трехпроводный интерфейс SPI (Serial Peripheral Interface) предназначен для организации обмена данными между двумя устройствами. С его помощью может осуществляться обмен данными между микроконтроллером и различными устройствами, такими, как цифровые потенциометры, ЦАП/АЦП, FLASH-ПЗУ и др. С помощью этого интерфейса удобно производить обмен данными между несколькими микроконтроллерами AVR.

Кроме того, **через интерфейс SPI может осуществляться программирование микроконтроллера.**

Двухпроводной последовательный интерфейс TWI

Двухпроводной последовательный интерфейс TWI (Two-wire Serial Interface) является полным аналогом базовой версии интерфейса **I2C** (двухпроводная двунаправленная синхронно-асинхронная шина) фирмы **Philips**. Этот интерфейс позволяет объединить вместе до **128** различных устройств с помощью двунаправленной шины, состоящей из двух линий - линии тактового сигнала (**SCL**) и линии данных (**SDA**).

Интерфейс JTAG

Интерфейс JTAG был разработан группой ведущих специалистов по проблемам тестирования электронных компонентов (**Joint Test Action Group**) и был зарегистрирован в качестве промышленного стандарта IEEE Std 1149.1-1990.

Четырехпроводной интерфейс JTAG используется для тестирования печатных плат, внутрисхемной отладки, программирования микроконтроллеров.

Многие микроконтроллеры семейства Mega имеют совместимый с IEEE Std 1149.1 интерфейс **JTAG** или **debugWIRE** для встроенной отладки. Кроме того, все микроконтроллеры Mega с флэш-памятью емкостью 16 кбайт и более могут программироваться через интерфейс JTAG.

Тактовый генератор

Тактовый генератор вырабатывает импульсы для синхронизации работы всех узлов микроконтроллера. Внутренний тактовый генератор AVR может запускаться от нескольких источников опорной частоты (**внешний генератор, внешний кварцевый резонатор, внутренняя или внешняя RC-цепочка**). Минимальная допустимая частота ничем не ограничена (вплоть до пошагового режима). Максимальная рабочая частота определяется конкретным типом микроконтроллера и указывается Atmel в его характеристиках, хотя практически любой AVR-микроконтроллер с заявленной рабочей частотой, например, в 10 МГц при комнатной температуре легко может быть "разогнан" до 12 МГц и выше.

Система реального времени

Система реального времени – (RTC – real time clock) реализована во всех микроконтроллерах Mega и в двух кристаллах "classic" - AT90(L)S8535. Таймер/счетчик RTC имеет отдельный предделитель, который может быть программным способом подключен или к источнику основной тактовой частоты, или к дополнительному асинхронному источнику опорной частоты (кварцевый резонатор или внешний синхросигнал). Для этой цели зарезервированы два вывода микросхемы. Внутренний осциллятор оптимизирован для работы с внешним "часовым" кварцевым резонатором **32,768 кГц**.

ТИПЫ КОРПУСОВ МИКРОСХЕМ



PDIP - Plastic Dual
In-Line Package



SOIC - Small Outline
Integrated Circuit



PLCC - Plastic Leader
Chip Carrier



TQFP - Thin Profile
Quad Flat Package



MLF - Micro Lead
Frame Package



BGA - Ball grid array

ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ AVR

Языки

Для микроконтроллеров AVR существуют различные языки программирования, но, пожалуй, наиболее подходящими являются ассемблер и Си, поскольку в этих языках в наилучшей степени реализованы все необходимые возможности по управлению аппаратными средствами микроконтроллеров.

Ассемблер – это низкоуровневый язык программирования, использующий непосредственный набор инструкций микроконтроллера. Создание программы на этом языке требует хорошего знания системы команд программируемого чипа и достаточного времени на разработку программы. Ассемблер проигрывает Си в скорости и удобстве разработки программ, но имеет заметные преимущества в размере конечного исполняемого кода, а соответственно, и скорости его выполнения.

Си – позволяет создавать программы с гораздо большим комфортом, предоставляя разработчику все преимущества языка высокого уровня.

Архитектура и система команд AVR создавалась при непосредственном участии разработчиков компилятора языка Си и в ней учтены особенности этого языка. **Компиляция исходных текстов, написанных на Си, осуществляется быстро и дает компактный, эффективный код.**

Основные преимущества Си перед ассемблером:

1. высокая скорость разработки программ;
2. универсальность, не требующая досконального изучения архитектуры микроконтроллера;
3. лучшая документируемость и читаемость алгоритма;
4. наличие библиотек функций;
5. поддержка вычислений с плавающей точкой.

В языке Си гармонично сочетаются возможности программирования низкого уровня со свойствами языка высокого уровня. Возможность низкоуровневого программирования позволяет легко оперировать непосредственно аппаратными средствами, а свойства языка высокого уровня позволяют создавать легко читаемый и модифицируемый программный код. Кроме того, практически все компиляторы Си имеют возможность использовать ассемблерные вставки для написания критичных по времени выполнения и занимаемым ресурсам участков программы.

Одним словом, **Си – наиболее удобный язык** как для начинающих знакомиться с микроконтроллерами AVR, так и для серьезных разработчиков