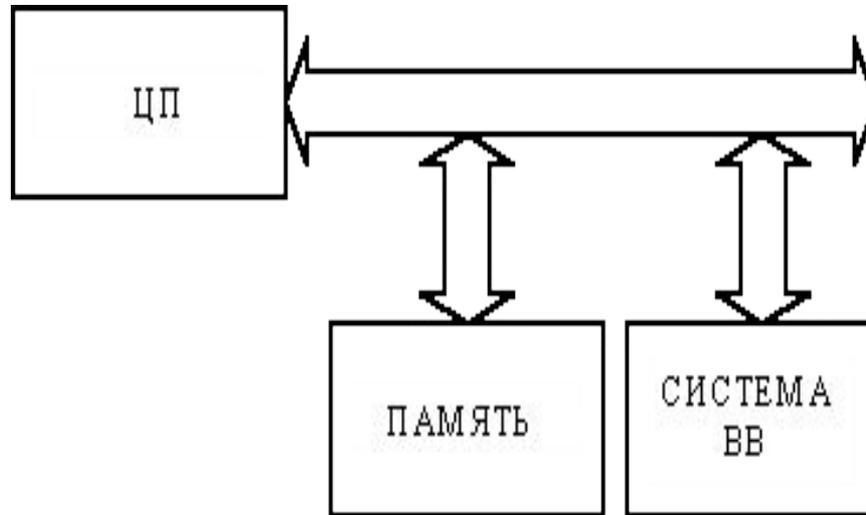


**Микроконтроллеры –  
отдельный класс  
микропроцессорных  
устройств для  
встроенных применений.  
Их особенности и  
выполняемые функции.**

- 1. Фон Неймановская архитектура микропроцессорной системы.**
- 2. Гарвардская архитектура микропроцессорной системы.**
- 3. Система шин микропроцессорной системы.**
- 4. Методы адресации.**
- 5. Структурная схема микроконтроллера MCS-51.**
- 6. Карта памяти программ и данных.**
- 7. Булевы операции.**

Базовая структура микропроцессорной системы имеет вид



Организация, при которой для хранения программ и данных используется одно пространство памяти, называется **фон Неймановской архитектурой** (по имени математика, предложившего кодирование программ в формате, соответствующем формату данных). Программы и данные хранятся в едином пространстве, и нет никаких признаков, указывающих на тип информации в ячейке памяти. Преимуществами такой архитектуры являются более простая внутренняя структура микропроцессора и меньшее количество управляющих сигналов.

Организация, при которой память программ CSEG (Code Segment) и память данных DSEG (Data Segment) разделены и имеют свои собственные адресные пространства и способы доступа к ним, называется **Гарвардской архитектурой** ( по имени лаборатории Гарвардского Университета, предложившей ее). Такая архитектура является более сложной и требует дополнительных управляющих сигналов. Однако, она позволяет осуществлять более гибкие манипуляции информацией, реализовывать компактно кодируемый набор машинных команд и, в ряде случаев, ускорять работу микропроцессора. Представителями такой архитектуры являются микроконтроллеры семейства MCS-51 фирмы Intel.

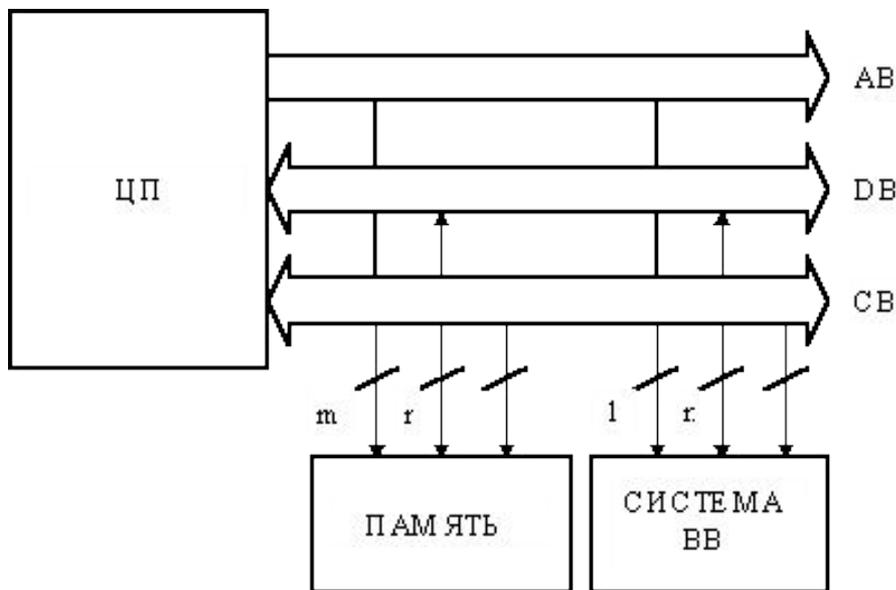
В настоящее время выпускаются микропроцессоры со смешанной архитектурой, в которых CSEG и DSEG имеют единое адресное пространство, однако различные механизмы доступа к ним. Конкретным примером являются микропроцессоры семейства 80x86 фирмы Intel.

На физическом уровне микропроцессор взаимодействует с памятью и системой ввода-вывода через единый набор системных шин - **внутрисистемную магистраль.**

Она, в общем случае состоит из:

- шины данных DB (Data Bus), по которой производится обмен данными между ЦП, памятью и системой ВВ;
- шины адреса АВ (Address Bus), используемой для передачи адресов ячеек памяти и портов ВВ, к которым осуществляется обращение;
- шины управления СВ (Control Bus), по которой передаются управляющие сигналы, реализующие циклы обмена информацией и управляющие работой системы.

Этот же набор шин применяется для организации канала ПДП. Магистраль такого типа носит название **демультиплексной** или **трехшинной с отдельными шинами адреса и данных**



В некоторых микропроцессорах с целью сокращения ширины физической магистрали вводят **совмещенную шину адреса-данных AD** (Address/Data Bus), по которой передаются как адреса так и данные.

Этап передачи адресной информации отделен по времени от этапа передачи данных и стробируется специальным сигналом ALE (Address Latch Enable), который включен в состав СВ. Данную магистраль обычно называют **мультиплексной** или **двухшинной с совмещенными шинами адреса и данных**.

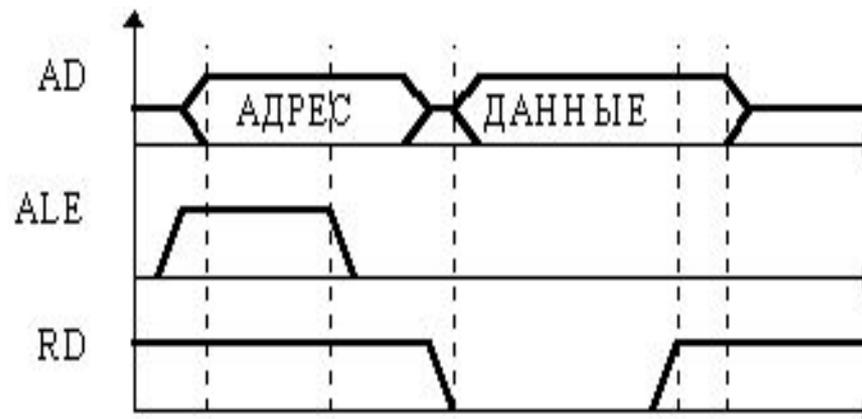
Цикл чтения памяти по демультимплексной магистрали.



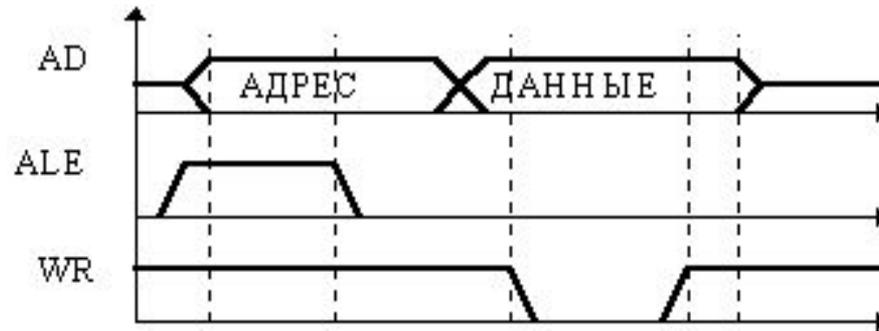
Цикл записи в память по демultipлексной магистрали.



Цикл чтения из памяти по мультиплексной магистрали



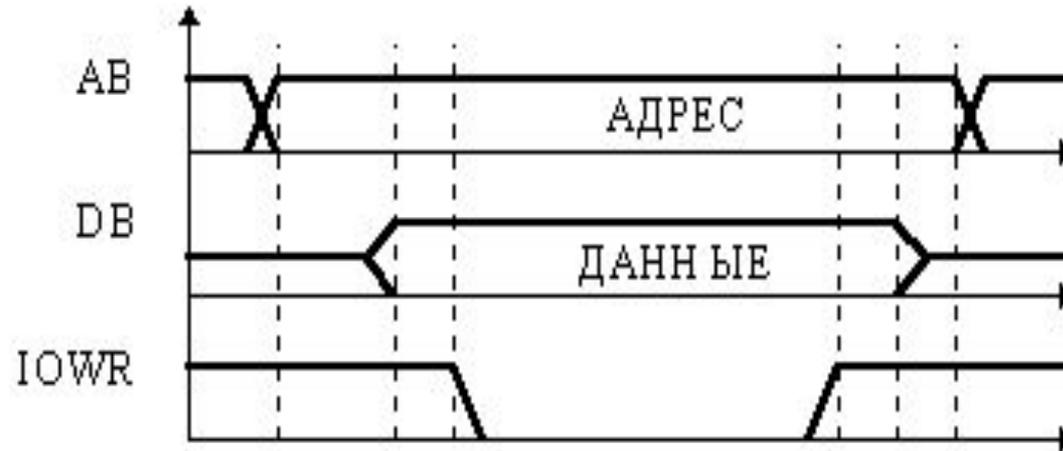
# Цикл записи в память по мультиплексной магистрали.



# Цикл чтения порта ВВ по демultipлексной магистрали

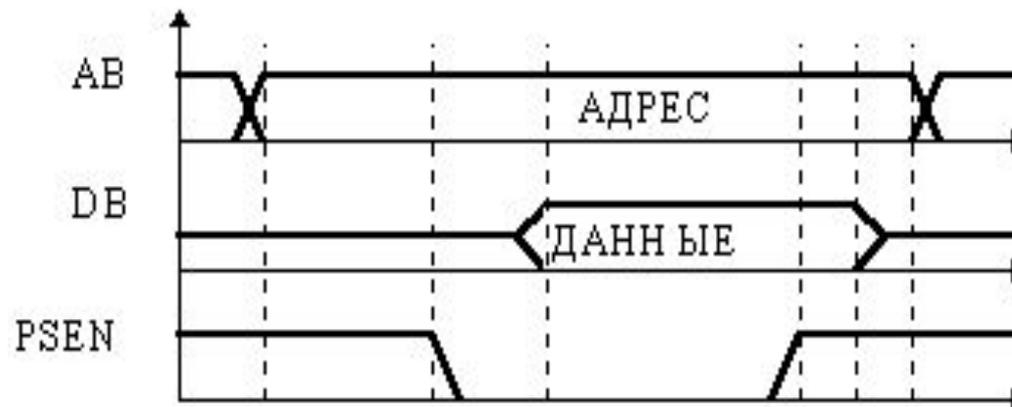


Цикл записи в порт ВВ по демультимплексной магистрали.



В случае архитектуры Гарвардского типа, когда память программ и память данных разделены, вводится также **цикл чтения памяти программ**.

Цикл чтения памяти программ по демультимплексной магистрали.



Кроме CSEG и DSEG практически все современные микропроцессоры имеют специально выделенное пространство данных небольшого объема, называемое **набором программно-доступных регистров RSEG** (Register Segment). В отличие от CSEG и DSEG регистры RSEG располагаются внутри ЦП в непосредственной близости от его АЛУ, что обеспечивает быстрый физический доступ к информации, хранящейся в них. В них, как правило, хранятся промежуточные результаты вычислений, часто используемые ЦП. Область RSEG может быть полностью изолирована от пространства данных DSEG, может частично пересекаться с ней, и может полностью являться подчастью DSEG. Внутренняя логическая организация RSEG очень разнообразна.

Регистры микропроцессора функционально неоднородны: одни служат для хранения данных или адресной информации, другие - для управления работой ЦП. В соответствии с этим все регистры можно разделить на **регистры данных, указатели и регистры специального назначения.** Регистры данных участвуют в арифметических и логических операциях в качестве источников операндов и приемников результата, адресные регистры или указатели используются для вычисления адресов данных и команд, расположенных в основной памяти.

Специальные регистры служат для индикации текущего состояния ЦП и управления работой его составных частей. Возможна архитектура, при которой одни и те же регистры используются для хранения как данных, так и адресной информации. Такие регистры называются **регистрами общего назначения (РОН)**. Способы использования того или иного вида регистров определяют конкретные особенности архитектуры микропроцессора.

Среди регистров данных часто выделяют один регистр, называемый **аккумулятором А** (Accumulator), с которым связывают большинство команд арифметической и логической обработки данных.

Это означает, что арифметические и логические команды используют в качестве одного из своих операндов содержимое аккумулятора и сохраняют в нем результат операции. Ссылка на него производится неявно с помощью кода операции. При этом нет необходимости в коде команды выделять специальную область для адресов операнда и результата. Такой тип архитектуры микропроцессора называется **аккумуляторным**. К недостаткам такой архитектуры можно отнести относительно низкое быстродействие, объясняемое тем, что аккумулятор является "узким местом", в которое каждый раз необходимо сначала занести операнд перед выполнением операции.

Примером такой архитектуры могут служить микроконтроллеры семейства MCS-51 фирмы Intel.

Другим примером организации регистров данных являются т.н. “рабочие регистры” R0, R1, и т.д. В этом случае операнды и результаты арифметических и логических операций могут храниться не в одном, а в нескольких регистрах, что расширяет возможности по манипуляции данными. В отличие от аккумулятора рабочие регистры адресуются явно в коде команды. Такой тип архитектуры микропроцессора называется **регистровым**.

Примером такой организации могут служить микропроцессоры семейства 80x86 фирмы Intel. В ряде микропроцессоров, предназначенных для работы в реальном масштабе времени, предусмотрены не один, а несколько наборов рабочих регистров. В каждый момент времени доступен лишь один из наборов регистров, выбор которого осуществляется записью соответствующей информации в определенный служебный регистр. Примером таких устройств могут служить микроконтроллеры семейства MCS-48 фирмы Intel.

Служебные регистры, расположенные внутри микропроцессора, предназначены для различных функций управления его работой и индикации состояния его составных частей. Их состав и организация зависят от конкретной архитектуры процессора и различаются в каждом конкретном случае. Наиболее часто встречающимися регистрами специальных функций являются **“программный счетчик”** PC (Program Counter), **“указатель стека”** SP (Stack Pointer) и **“слово состояния программы”** PSW (Program Status Word). Программный счетчик PC в каждый конкретный момент времени содержит адрес команды, следующей в CSEG за той, которая в данный момент выполняется.

. Указатель стека *SP* хранит текущий адрес вершины стека. Слово состояния программы *PSW* содержит набор текущих признаков результата выполнения операции. С каждым признаком результата связывается одноразрядная переменная-флажок, соответствующая определенному биту *PSW*.

К типовым флажкам-признакам относятся:

- **CF** (Carry Flag) - флажок переноса из старшего разряда АЛУ. Равен 1, если в результате выполнения арифметической операции или операции сдвига произошел перенос из старшего разряда результата;
- **ZF** (Zero Flag) - флажок признака нуля. Равен 1, если результат операции равен 0;
- **SF** (Sign Flag) - флажок знака результата.

Дублирует знаковый разряд результата операции;

- **AF** (Auxiliary Carry Flag) - флажок дополнительного переноса. Равен 1, если в результате выполнения арифметической операции или операции сдвига произошел перенос из младшей тетрады результата в старшую. Часто используется в двоично-десятичной арифметике;
- **OF** (Overflow Flag) - флажок переполнения. Равен 1, если в результате выполнения арифметической операции произошло переполнение разрядной сетки результата;
- **PF** (Parity Flag) - флажок четности. Равен 1, если число 1 в результате операции нечетно и наоборот.
- **IF** (Interrupt Flag) - флажок разрешения прерывания. Индицирует, разрешены ли прерывания в системе.

Под **методом адресации** понимается метод кодирования адреса операнда или результата операции в коде команды.

В общем случае код команды микропроцессора можно представить в следующем виде

КОП	АОП1	АОП2	...	АР
-----	------	------	-----	----

где,

КОП - код операции;

АОП1 - поле адреса первого операнда;

АОП2 - поле адреса второго операнда;

АР - поле адреса результата.

Наиболее распространенными методами адресации, используемыми в современных моделях микропроцессоров являются:

- **Регистровая адресация.** Операнд находится в регистре. Адрес регистра включен в код операции. Поле адреса в команде отсутствует;
- **Прямая адресация.** Физический адрес операнда расположен в соответствующем поле адреса.
- **Непосредственная адресация.** Непосредственное значение операнда расположено в соответствующем поле адреса.
- **Косвенная регистровая адресация.** Физический адрес операнда расположен в регистре косвенного адреса DP (Data Pointer). Адрес регистра включен в код операции. Поле адреса в команде отсутствует.

- **Косвенная автоинкрементная/автодекрементная адресация.** Физический адрес операнда расположен в регистре косвенного адреса DP. Адрес регистра включен в код операции. Поле адреса в команде отсутствует. После (либо до) выполнения операции содержимое DP автоматически инкрементируется/декрементируется, чтобы указывать на следующий элемент таблицы.
- **Адресация по базе со смещением.** Базовый адрес операнда расположен в регистре базы BP (Base Pointer). Адрес регистра включен в код операции. Смещение адреса операнда относительно базового адреса расположено в соответствующем поле адреса. В качестве BP может выступать РОН или специальный адресный регистр;
- **Индексная адресация.** Базовый адрес операнда расположен в соответствующем поле адреса. Смещение адреса операнда относительно базового адреса расположено в индексном регистре X (Index). В качестве X может выступать РОН или специальный адресный регистр;

- **Адресация по базе с индексированием.** Базовый адрес операнда расположен в регистре базы BP , смещение адреса операнда относительно базового адреса расположено в индексном регистре X . Адреса регистров включены в код операции. Поле адреса в команде отсутствует; В качестве X и BP могут выступать РОН или специальные адресные регистры;

- **Сегментная адресация.** Вся память разбита на сегменты определенного объема. Адрес сегмента хранится в сегментном регистре SR (Segment Register), смещение адреса относительно начала сегмента расположено в соответствующем поле адреса либо в индексном регистре X. В качестве X может выступать РОН или специальный адресный регистр;

Одно из современных направлений развития архитектуры микропроцессоров основано на том, чтобы путем сокращения числа возможных команд и методов адресации добиться того, чтобы любая команда выполнялась за один машинный цикл. Такие процессоры называются RISC-процессорами (Reduced Instruction Set Computer).

В составе системы ВВ также можно выделить ряд функционально законченных устройств, которые оформляются в виде модулей подключаемых непосредственно к единой магистрали системы. В простейшем случае это адресуемые ЦП буферные регистры - **порты ВВ**. Более сложные программно-управляемые подсистемы ВВ, содержащие блоки портов, получили название **периферийных адаптеров**. В случае, когда средства ВВ предназначены для управления специальным внешним оборудованием и реализации специальных функций ВВ, их называют **периферийными контроллерами**.

Наиболее сложными из современных средств обмена с внешними устройствами ВВ считают **сопроцессоры ВВ**, которые работают по собственным программам, хранящимся в собственной памяти, и по сути дела представляют собой отдельные микропроцессорные системы.

В зависимости от того, в каком формате процессор способен воспринимать и обрабатывать данные, различают **микропроцессоры с фиксированной точкой** и **микропроцессоры с плавающей точкой**. При заданной точности вычислений и разрядности, диапазон чисел, представимых в формате с плавающей точкой значительно превышает диапазон чисел в формате с фиксированной точкой.

Существуют микропроцессоры, архитектура которых адаптирована для выполнения вычислений определенного рода. К числу таких процессоров относятся т.н. **“процессоры цифровой обработки сигналов”** DSP (Digital Signal Processor). Все эти процессоры построены, как правило по Гарвардской архитектуре. Современные DSP имеют отдельные шины адреса/данных для CSEG и DSEG, что позволяет им с помощью одной команды осуществить доступ к различным видам памяти и произвести несколько операций над данными.

Основной особенностью DSP является то, что кроме обыкновенного АЛУ, которое присутствует во всех процессорах, они имеют еще несколько вычислительных устройств. К числу таких устройств в первую очередь относится т.н. “умножитель-аккумулятор” MAU (Multiple-Accumulator Unit), способный с помощью одной команды умножить два многоразрядных числа и сложить результат удвоенной разрядности с результатом предыдущей команды.

Микроконтроллер – вычислительно-управляющее устройство, предназначенное для выполнения функций контроля и управления периферийным оборудованием.

Уклон в сторону управления накладывает отпечаток на особенность архитектуры микроконтроллеров. Основной из этих особенностей является то, что наряду с процессорным ядром микроконтроллера имеют в своём составе подсистему ввода-вывода и, возможно, подсистему памяти. В последнем случае принято говорить об однокристальных микро-ЭВМ. Рассмотрим особенности организации каждой из подсистем микроконтроллеров.

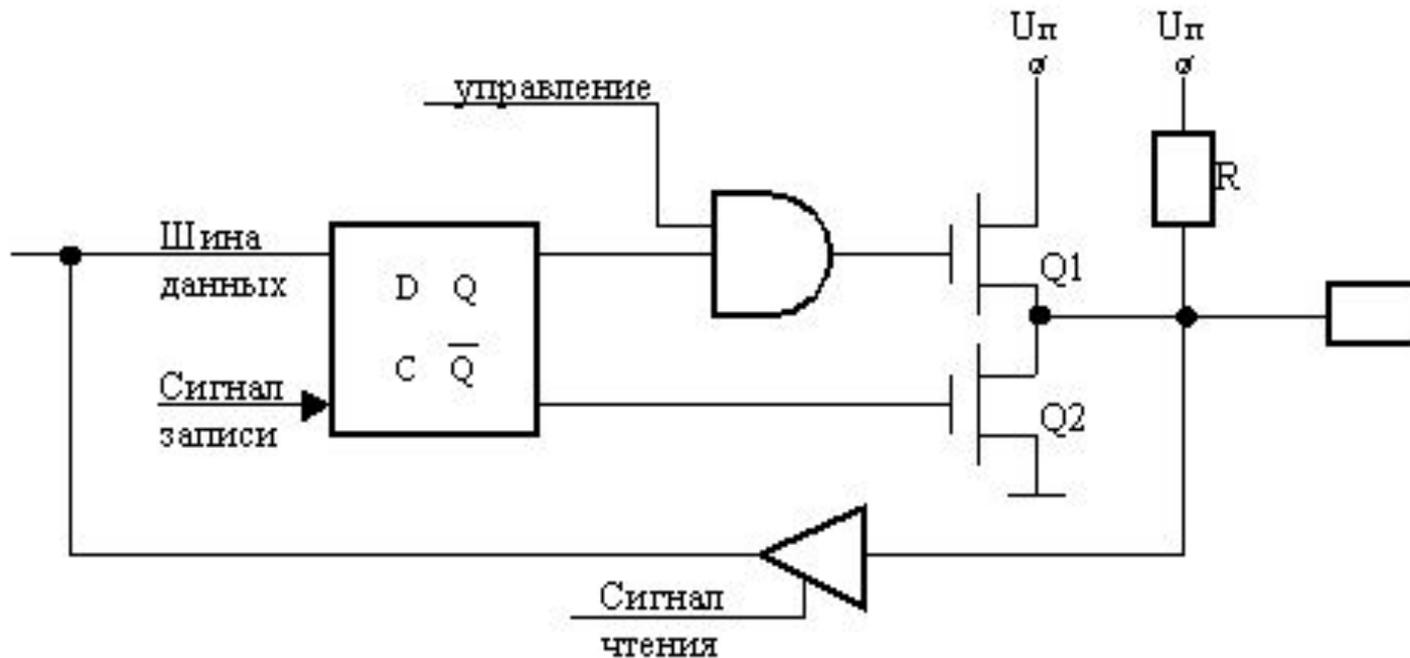
Современные микроконтроллеры могут быть построены как по Гарвардской (MCS-51 Intel), так и по Фон Неймановской архитектуре (MCS-96 Intel, 80C166 Siemens). Все они имеют внешнюю системную магистраль для обмена данными с внешней памятью и дополнительными периферийными устройствами. Классические семейства микроконтроллеров (MCS-51) имеют, как правило, мультиплексные шины адреса/данных, что было обусловлено необходимостью минимизировать размер микросхемы. Однако современные быстродействующие микроконтроллеры используют уже демultipлексную шину, что ускоряет работу системы. Некоторые модели микроконтроллеров имеют возможность работать либо мультиплексной либо с демultipлексной шиной, в зависимости от требуемой конфигурации системы. В случае демultipлексной шины контроллер быстрее обменивается данными по магистрали.

При работе с мультиплексной шиной, освободившиеся выходы используются как порты ввода-вывода. (MCS251 Intel, 80C166 Siemens).

Практически все микроконтроллеры выполняют только операции с фиксированной точкой. Существуют 8-разрядные (MCS-51 Intel, MC6805 Motorola), 16-разрядные (MCS-96 Intel, 80C166 Siemens, MC6816 Motorola), 32-разрядные (MC6833 Motorola, MPC500 PowerPc) микроконтроллеры. Системы команд микроконтроллеров поддерживает, как правило, широкий набор методов адресации в т. ч. битовую адресацию.

Существуют микроконтроллеры с аккумуляторной (MCS-51) и регистровой (MCS-96) организацией. Количество регистров и их разрядность зависит от конкретной модели. Зачастую микроконтроллеры имеют несколько банков регистров (MCS-48, MCS-51, 80C166).

Порты ввода-вывода. Они могут быть либо однонаправленными (выполняя функции входа или выхода соответственно), либо квази-двунаправленными. Такие порты могут выполнять функции как входа, так и выхода (в каждый конкретный момент времени либо вход, либо выход).



Если порт работает как выход, то один из выходных транзисторов открыт, а другой закрыт, в соответствии с уровнем выходного сигнала. Чтобы порт работал как вход необходимо записать в выходной триггер "1", тогда транзистор Q2 будет закрыт, а сигнал управления блокирует открытие Q1. Таким образом, образуется высокоимпедансный вход. Микроконтроллеры, поддерживающие битовую адресацию способны управлять состоянием каждого вывода порта отдельно.

Таймеры/счетчики. Они представляют собой специализированные счетчики с коммутируемыми источниками синхронизации.

Широтно-импульсный модулятор. Встроенный ШИМ предназначен для генерации широтно - модулированного сигнала на выходе микросхемы без участия процессора.

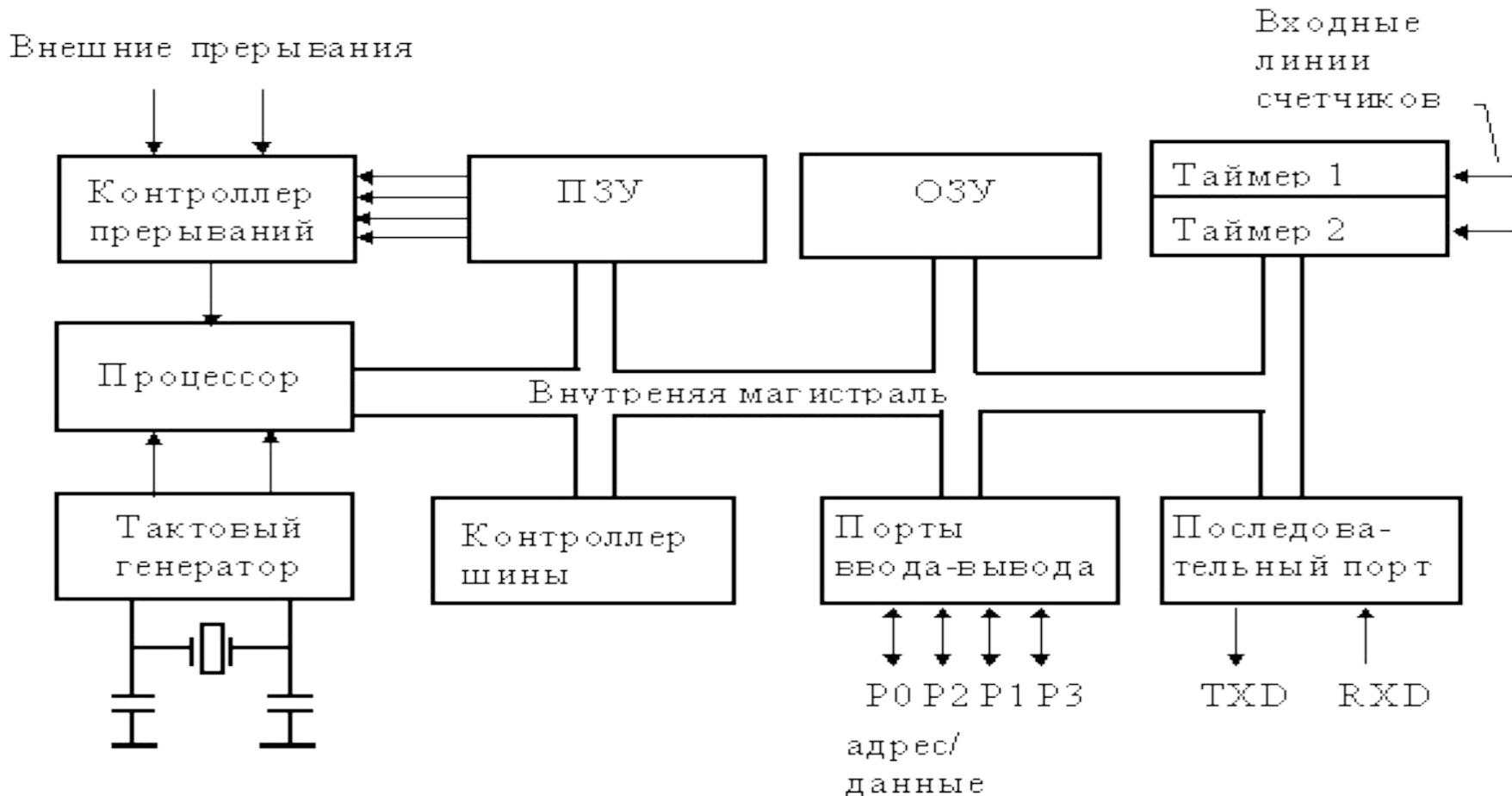
Модуль последовательного ввода-вывода используется для обмена данными между микроконтроллером и удаленным периферийным узлом. Данные передаются в последовательном коде, то есть биты передаются во временной последовательности друг за другом по одному каналу связи.

Модуль АЦП предназначен для преобразования входной аналоговой информации в цифровую и передачи ее в процессор для дальнейшей обработки. Микроконтроллеры имеют, как правило, несколько каналов АЦП.

“Семейство MCS-51 фирмы Intel - как представитель 8-разрядных микроконтроллеров. Обобщенная архитектура семейства. Состав семейства. Типовая схема включения микроконтроллера

Семейство 8-разрядных микроконтроллеров MCS-51 было выпущено фирмой Intel в начале 80-х годов. Микроконтроллеры MCS-51 являются функционально завершенными однокристальными микроЭВМ Гарвардской архитектуры, содержащими все необходимые узлы для работы в автономном режиме, и предназначены для реализации различных цифровых алгоритмов управления. На сегодняшний день семейство MCS-51 содержит несколько десятков типов микросхем, отличающихся конкретной реализацией отдельных узлов и условиями эксплуатации. Все микросхемы семейства обладают аналогичной архитектурой и имеют целый ряд общих узлов.

# Структурная схема микроконтроллера MCS-51.



Общими узлами всех микроконтроллеров семейства являются:

- 8-разрядный центральный процессор (ЦП), ориентированный на управление исполнительными устройствами. ЦП имеет встроенную схему 8-разрядного аппаратного умножения и деления чисел. Наличие в наборе команд большого числа операций для работы с прямоадресуемыми битами дает возможность говорить о т.н. “булевом процессоре”;

- внутренняя память программ масочного или репрограммируемого типа, имеющая для различных кристаллов объем от 4 до 32 Кбайт; в некоторых версиях она отсутствует;

- не менее чем 128 байтное резидентное ОЗУ данных, которое используется для организации регистровых банков, стека и хранения пользовательских данных;

не менее 32 двунаправленных интерфейсных линий (портов), индивидуально настраиваемых на ввод или вывод информации;

- два 16-битных многорежимных счетчика/таймера, используемых для подсчета внешних событий, организации временных задержек и тактирования коммуникационного порта;

- двунаправленный дуплексный последовательный коммуникационный порт, предназначенный для организации каналов связи между микроконтроллером и внешними устройствами;

- двухуровневая приоритетная система прерываний, поддерживающая не менее 5 векторов прерываний от внутренних и внешних источников;

- встроенный тактовый генератор.

Центральный процессор микроконтроллеров MCS-51 имеет следующие технические характеристики:

Разрядность АЛУ - 8 бит.

Число выполняемых команд - 111.

Длина команд - 1, 2, или 3 байта.

Число регистров общего назначения (РОН) - 32.

Число прямоадресуемых битовых переменных - 128.

Число прямоадресуемых битов в области регистров специальных функций - 128.

Максимальный объем памяти программ - 64 Кбайта.

Максимальный объем памяти данных - 64 Кбайта.

Максимальный объем внутренней памяти данных - 256 байт.

Время выполнения команд при тактовой частоте 12 МГц:

сложение - 1 мкс;

пересылки “регистр- внешняя память данных” - 2 мкс;

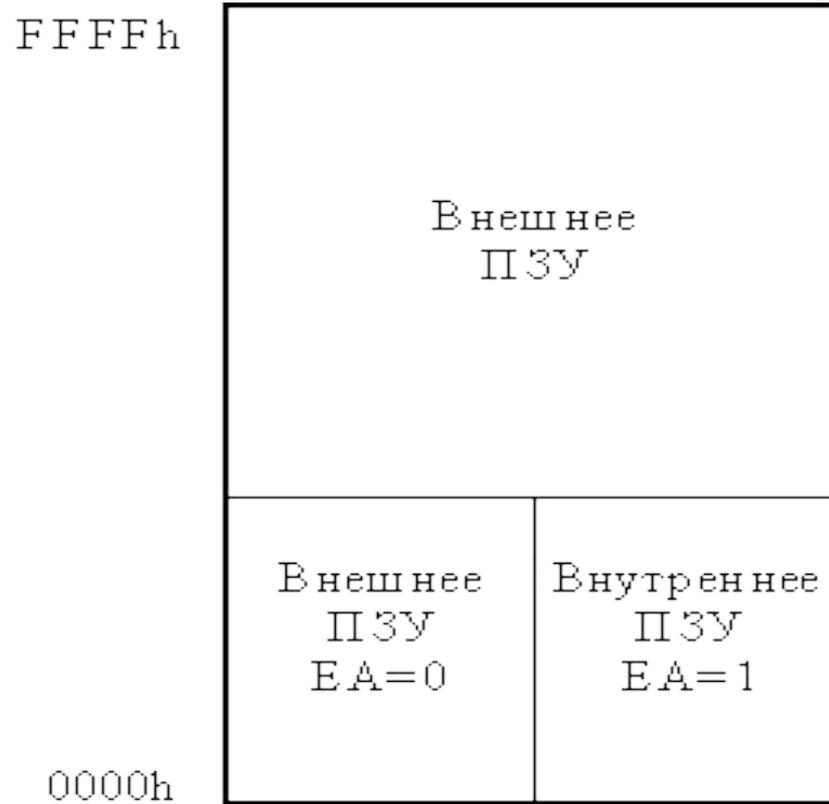
умножение/деление - 4 мкс;

Методы адресации операнда - регистровый, косвенный, прямой, непосредственный.

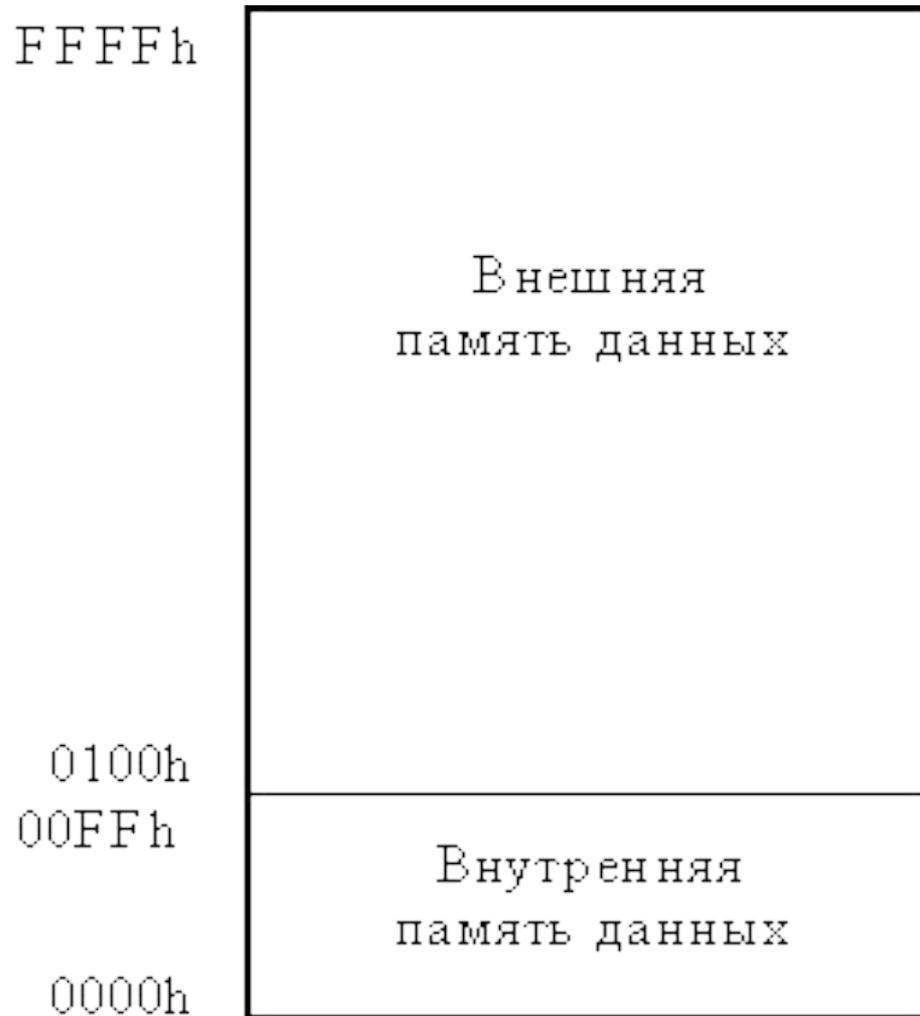
Все микроконтроллеры семейства MCS-51 имеют Гарвардскую архитектуру, т.е. отдельное адресное пространство программ и данных. Такое разделение позволяет осуществлять доступ к памяти данных по 8-битным адресам, что способствует более быстрой обработке данных в ЦП. Тем не менее, возможно использование и 16-битных адресов данных.

Программная память доступна исключительно по чтению. Общий объем ее адресного пространства составляет 64 Кбайта. В различных версиях кристаллов может присутствовать до 32 Кбайт внутренней памяти программ. При обращении к этим областям памяти используются циклы обмена по внутренней магистрали микроконтроллера. При необходимости можно расширить память программ за счет внешнего ПЗУ.

# Карта памяти программ



# Карта памяти данных



Внутренняя память данных может быть разделена на три условных блока - нижний, верхний и пространство Регистров Специальных Функций (SFR).

Карта внутренней памяти данных.

FFh ... ... 80h	Верхняя область внутреннего ОЗУ (доступна в режиме косвенной адресации)	Область регистров специальных функций (доступна в режиме прямой адресации)
7Fh ... ... 00h	Нижняя область внутреннего ОЗУ (доступна в режимах прямой и косвенной адресации)	
	косвенная адресация	прямая адресация

Внутренняя память данных всегда адресуется байтом, что подразумевает 256 байт адресного пространства. Однако, фактически можно обращаться к 384-м байтам. Прямая адресация ячейки с адресом свыше 7Fh осуществляет доступ к области регистров специальных функций, а косвенная - к верхней области внутренней памяти данных. Т. о. эти две области занимают одно и то же адресное пространство, хотя и разделены физически. Все байты в нижней 128-байтной половине внутренней памяти данных могут адресоваться как прямо, так и косвенно.

Первые 32 байта в нижней области внутреннего ОЗУ сгруппированы в 4 банка по 8 регистров. Инструкции программы могут оперировать с ними как с регистрами общего назначения R0-R7. Два бита регистра PSW определяют, какой из банков используется в текущий момент. Это позволяет более эффективно использовать память программ, поскольку регистровые инструкции короче, чем прямая адресация.

Для 16 адресов в пространстве SFR имеется возможность как битовой, так и байтовой адресации. Побитово-адресуемыми регистрами являются те, чей шестнадцатеричный адрес заканчивается на "0" или "8". Битовые адреса в этой области располагаются со значения 80h по FFh. Наиболее часто используемыми регистрами специальных функций являются:

- Аккумулятор (байтовый адрес 0E0h). Используется во всех арифметических и логических операциях в качестве операнда;
- Регистр В (байтовый адрес - 0F0h). Используется при операциях умножения и деления, а также как сверхоперативный регистр;
- Слово состояния программы PSW (байтовый адрес - 0D0h). Содержит информацию о состоянии программы;

- Указатель стека (байтовый адрес -081h).  
Используется для указания на вершину стека в операциях записи в стек и чтения из него. По аппаратному сбросу устанавливается в значение 07h (область стека в этом случае начинается с адреса 08h) и инкрементируется при каждой записи в стек. Запись в SP производится для predeterminedения положения стека во внутренней памяти данных микроконтроллера;
- Указатель данных DPTR (байтовые адреса 082h, 083h). Состоит из двух байт: старшего - DPH и младшего - DPL. Используется как 16-ти или 8-битовый указатель адреса при обращении к внешней памяти или выполнении команды перехода по косвенному адресу.

Слово состояния программы PSW содержит ряд статусных битов, отражающих текущее состояние процессора.  
Формат PSW.

Бит	Назначение
CY	Флаг переноса.
AC	Флаг дополнительного переноса.
F0	Флаг пользователя 0.
RS1, RS0	Биты выбора используемого регистрового банка: 00 - банк 0; 01 - банк 1; 10 - банк 2; 11 - банк 3.
OV	Флаг переполнения.
-	Зарезервирован. Доступен по чтению и записи.
P	Флаг четности

## **Режимы адресации.**

Набор команд MCS-51 поддерживает следующие режимы адресации:

**Прямая адресация.** Операнд определяется 8-битовым адресом в инструкции. Прямая адресация используется только для младшей половины внутренней памяти данных и регистров SFR.

**Косвенная адресация.** Инструкция адресует регистр, содержащий адрес операнда. Данный вид адресации используется для внешнего и внутреннего ОЗУ. Для указания 8-битовых адресов могут использоваться регистры R0 и R1 выбранного регистрового банка или указатель стека SP. Для 16-битовой адресации используется только регистр указателя данных DPTR.

**Регистровые инструкции.** Регистры R0-R7 текущего регистрового банка могут быть адресованы через конкретные инструкции, содержащие 3-хбитовое поле, указывающее номер регистра в самой инструкции. В этом случае соответствующее поле адреса в команде отсутствует.

**Операции с использованием специальных регистров.** Некоторые инструкции используют индивидуальные регистры. Например операции с аккумулятором, DPTR, и т.д. В данном случае адрес операнда вообще не указывается в команде. Он предопределяется кодом операции.

**Непосредственные константы.** Константа может находиться прямо в команде за кодом операции.

**Индексная адресация.** Индексная адресация может использоваться только для доступа к программной памяти и только в режиме чтения. В этом режиме осуществляется просмотр таблиц в памяти программ. 16-битовый регистр (DPTR или программный счетчик) указывает базовый адрес требуемой таблицы, а аккумулятор указывает на точку входа в нее.

## **Арифметические и логические инструкции.**

Пример арифметической команды: Операция сложения может быть выполнена одной из следующих команд:

```
ADD A,7Fh, ADD A,@R0
```

```
ADD A,R7,ADD A,#127
```

## **Команды передачи данных.**

Команда `MOV dest,src` позволяет пересылать данные между ячейками внутреннего ОЗУ или SFR без использования аккумулятора. При этом работа с верхней половиной внутреннего ОЗУ может осуществляться только в режиме косвенной адресации, а обращение к регистрам SFR - только в режиме прямой адресации.

## **Булевы операции.**

Микросхемы MCS-51 содержат в своем составе “булевый” процессор. Внутреннее ОЗУ имеет 128 прямоадресуемых бит. Пространство SFR может также поддерживать до 128 битовых полей. Битовые инструкции осуществляют условные переходы, пересылки, сброс, инверсии, операции “И”, “ИЛИ”. Все указанные биты доступны в режиме прямой адресации. Бит переноса CF в PSW используется как 1-битный аккумулятор булевого процессора.

## **Инструкции переходов.**

Адреса операций переходов обозначаются на языке ассемблера меткой либо реальным значением в пространстве памяти программ. Адреса условных переходов ассемблируются в в относительное смещение - знаковый байт, прибавляемый к программному счетчику PC в случае выполнения условия перехода. Границы таких переходов лежат в пределах между -128 и +127, относительно первого байта, следующего за инструкцией. В PSW отсутствует флажок нуля, поэтому инструкции JZ и JNZ проверяют условие “равно нулю” тестирование данных в аккумуляторе.

Существует три вида команды безусловного перехода: SJMP, LJMP, AJMP, различающиеся форматом адреса назначения. Инструкция SJMP кодирует адрес как относительное смещение, и занимает 2 байта. Дальность перехода ограничена диапазоном от -128 до +127 байт относительно инструкции, следующей за SJMP. В инструкции LJMP используется адрес назначения в виде 16-битной константы. Длина команды составляет 3 байта. Адрес назначения может располагаться в любом месте памяти программ. Команда AJMP использует 11-битную константу адреса. Команда состоит из 2-х байт. При выполнении этой инструкции младшие 11 бит адресного счетчика замещаются 11-битным адресом из команды. 5 старших бит РС остаются неизменными. Т.о., переход может производиться внутри 2К-байтного блока, в котором располагается инструкция, следующая за командой AJMP.

Существует два вида команды вызовы подпрограммы: LCALL и ACALL. Инструкция LCALL использует 16-битный адрес вызываемой подпрограммы. В данном случае подпрограмма может быть расположена в любом месте памяти программ. Инструкция ACALL использует 11-битный адрес подпрограммы. В этом случае вызываемая подпрограмма должна быть расположена в одном 2К-байтном блоке с инструкцией, следующей за ACALL. Оба варианта команды кладут на стек адрес следующей команды и загружают в РС соответствующее новое значение.

Подпрограмма завершается инструкцией RET, позволяющей вернуться на инструкцию, следующую за командой CALL. Эта инструкция снимает со стека адрес возврата и загружает его в РС. Инструкция RETI используется для возврата из подпрограмм обработки прерываний. Единственное отличие RETI от RET состоит в том, что RETI информирует систему о том, что обработка прерывания завершилась. Если в момент выполнения RETI нет других прерываний, то она идентична RET

Инструкция DJNZ предназначена для управления циклами. Для выполнения цикла N раз надо загрузить в счетчик байт со значением N и закрыть тело цикла командой DJNZ, указывающей на начало цикла.

Команда CJNE сравнивает два своих операнда как беззнаковые целые и производит переход по указанному в ней адресу, если сравниваемые операнды не равны. Если первый операнд меньше, чем второй, то бит переноса CF устанавливается в “1”.

## “Порты ввода-вывода. Параллельный интерфейс.”

Порты ввода-вывода предназначены для обмена логической информацией между процессором и периферийными устройствами. Они могут быть однонаправленными (т.е. осуществлять обмен информацией в одну сторону) или двунаправленными. Большинство микроконтроллеров семейства MCS-51 имеют четыре 8-битных квазидвунаправленных порта ввода-вывода P0-P4. Они называются квазидвунаправленными потому, что каждый бит данных портов может индивидуально быть настроен на работу как на ввод так и на вывод информации. Со стороны процессора эти порты представляются в виде соответствующих регистров специальных функций P0-P3. В эти регистры процессор может записывать выходную информацию порта либо считывать из них входные данные. Данные регистры могут адресоваться как побайтно, так и побитно, что облегчает манипуляцию информацией на отдельных выводах портов.

Помимо работы в качестве обычных портов ввода-вывода, линии портов могут выполнять ряд альтернативных функций:

“Последовательный порт. Последовательный интерфейс”.

Последовательный порт предназначен для организации связи по последовательному каналу между микроконтроллером и периферийными устройствами или другими микропроцессорами.

Последовательный порт микроконтроллера семейства MCS-51 является т.н. “дуплексным” портом, т.е. способен осуществлять прием и передачу данных одновременно. Входная часть порта обладает двойной буферизацией, т.е. способна осуществлять прием нового сообщения пока предыдущее еще не прочитано процессором, что уменьшает загрузку процессора.

Входным и выходным регистром последовательного порта является регистр SBUF, расположенный в области регистров специальных функций (адрес 99H). Реально это два разных регистра: входной буфер и выходной буфер. Однако, команда осуществляющая запись в SBUF, обращается к выходному буферу, а команда, осуществляющая чтение из SBUF - ко входному.

Для управления режимом работы и состоянием последовательного порта используется управляющий регистр SCON, расположенный в пространстве регистров специальных функций (адрес 98H).

Последовательный порт может работать в одном из четырех режимов.

SM0	SM1	Режим	Описание	Частота
0	0	0	Синхронный	$F_{osc}/12$
0	1	1	8 бит асинхронный	переменная
1	0	2	9 бит асинхронный	$F_{osc}/64$ или $F_{osc}/32$
1	1	3	9 бит асинхронный	переменная