

Организация связи микроконтроллера с внешней средой и временем

Материалы по дисциплине «Микропроцессорные системы»
Специальность «Компьютерные системы и комплексы»
Составитель: Торгашин Р.Г

ОГОУ СПО Борисоглебский индустриальный техникум

Порты ввода/вывода

Каждый МК имеет некоторое количество линий ввода/вывода, которые объединены в многоразрядные (чаще 8-разрядные) параллельные порты ввода/вывода. В памяти МК каждому порту ввода/вывода соответствует свой адрес регистра данных. Обращение к регистру данных порта ввода/вывода производится теми же командами, что и обращение к памяти данных. Кроме того, во многих МК отдельные разряды портов могут быть опрошены или установлены командами битового процессора.

С точки зрения внешнего устройства порт представляет собой обычный источник или приемник информации со стандартными цифровыми логическими уровнями (обычно ТТЛ), а с точки зрения микропроцессора — это ячейка памяти, в которую можно записывать данные или в которой сама собой появляется информация.

В качестве внешнего устройства может служить любой объект управления или источник информации (различные кнопки, датчики, микросхемы, дополнительная память, исполнительные механизмы, двигатели, реле и так далее).

В зависимости от реализуемых функций различают следующие типы параллельных портов:

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

В качестве простейшего порта вывода может быть использован параллельный регистр, так как он позволяет запоминать данные, выводимые микропроцессором и хранить их до тех пор, пока подается питание. Все это время сигналы с выхода этого регистра поступают на внешнее устройство. Упрощенная функциональная схема порта вывода показана на рисунке 1.

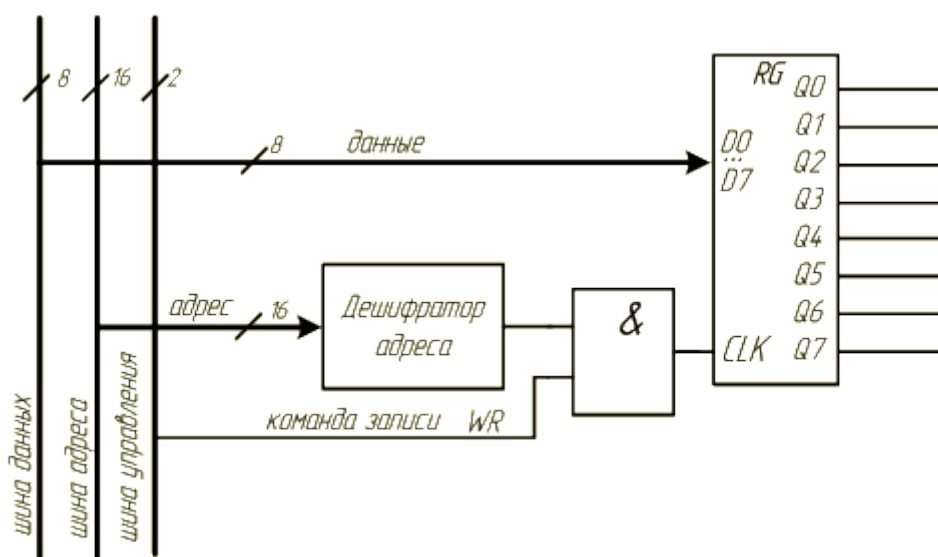


Рисунок 1 Функциональная схема порта вывода

Двоичные слова с системной шины данных микропроцессора записываются в регистр по сигналу “WR”. Выходы “Q” регистра могут быть использованы как источники логических уровней для управления внешними устройствами. Этот регистр называется регистром данных порта вывода.

Для отображения этого регистра только в одну ячейку памяти адресного пространства микропроцессорного устройства в составе порта ввода-вывода всегда присутствует дешифратор адреса. На выходе дешифратора адреса устанавливается единица лишь тогда, когда двоичное слово на его входе имеет заданное строго определенное значение, соответствующее адресу ячейки памяти, во всех остальных случаях на выходе дешифратора адреса удерживается логический ноль.

В качестве **порта ввода** может быть использован шинный формирователь. Для построения порта ввода выходы шинного формирователя (Q0-Q7) подключены к внутренней шине данных микропроцессора, а на его вход подключаются сигналы, которые нужно ввести в микропроцессорную систему. Упрощенная функциональная схема порта ввода приведена на рисунке 2.

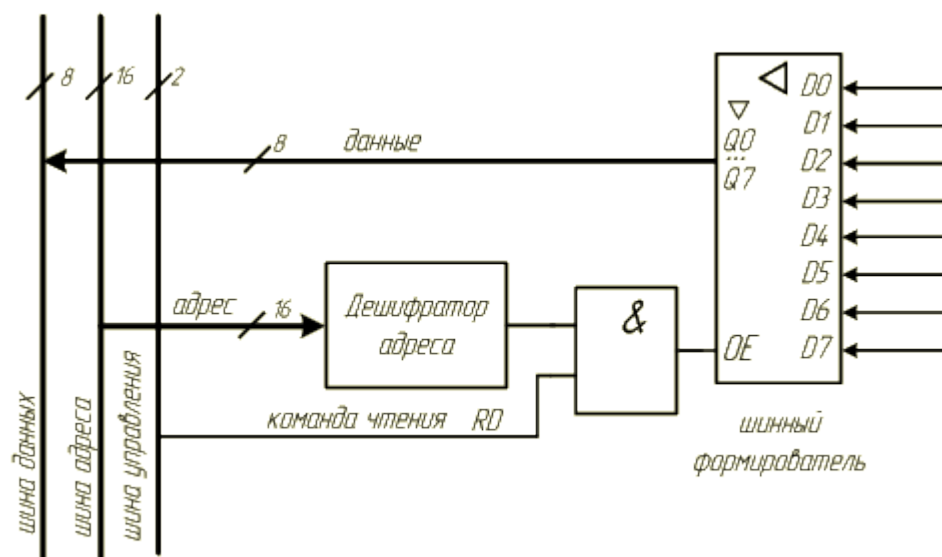


Рисунок 2 Функциональная схема порта ввода

Шинный формирователь работает следующим образом: данные с входов D0-D7 поступают на выходы Q0-Q7 лишь тогда когда на входе OE (Output Enable — разрешение выхода) установлен высокий логический уровень (“1”), когда же на OE логический ноль, выходы переходят в третье состояние и никак не влияют на шину данных. Значение сигнала с внешнего вывода порта передается на шину данных (считывается) по управляющему сигналу “RD”.

Для отображения шинного формирователя порта ввода в один адрес пространства адресов микроконтроллера используется дешифратор адреса. Выделяемый дешифратором адрес называют адресом регистра данных порта ввода. Из порта ввода возможно только чтение информации.

Так как из порта ввода возможно только чтение информации (команд “RD”), а в порт вывода только запись (команда “WR”), то для портов ввода и вывода можно отвести один и тот же адрес в адресном пространстве микропроцессора.

Порты выполняют роль устройств временного согласования функционирования МК и объекта управления, которые в общем случае работают асинхронно. Различают три типа

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

- режим простого программного ввода/вывода;
- режим ввода/вывода со стробированием;
- режим ввода/вывода с полным набором сигналов подтверждения обмена.

Типичная схема двунаправленного порта ввода/вывода МК приведена на рисунке 3

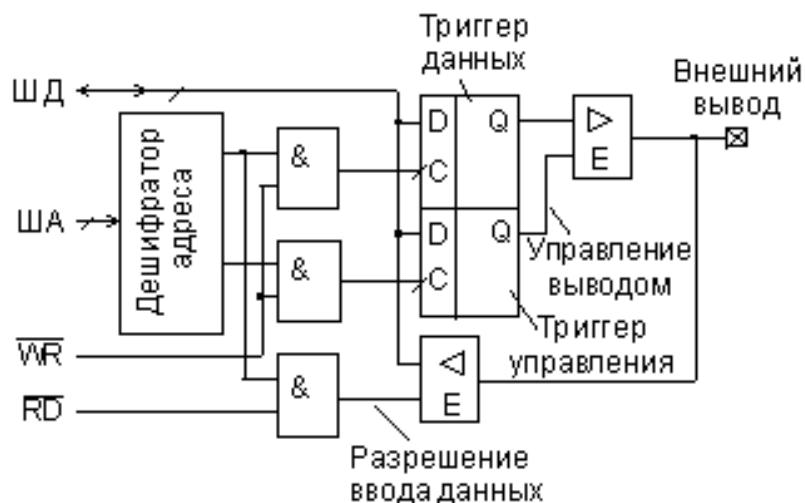


Рисунок 3 Типовая схема двунаправленного порта ввода/вывода МК.

Триггер управления разрешает вывод данных на внешний вывод. В современных МК, как правило, обеспечивается индивидуальный доступ к триггерам данных и управления, что позволяет использовать каждую линию независимо в режиме ввода или вывода.

Необходимо обратить особое внимание на то, что при вводе данных считывается значение сигнала, поступающее на внешний вывод, а не содержимое триггера данных. Если к внешнему выводу МК подключены выходы других устройств, то они могут установить свой уровень выходного сигнала, который и будет считан вместо ожидаемого значения триггера данных.

Другим распространенным вариантом схемотехнической организации порта ввода/вывода является вывод с "открытым истоком", называемый еще "квазидвунаправленным". Такая организация вывода позволяет создавать шины с объединением устройств по схеме "монтажное И".

Один разряд регистра порта представляет собой D-триггер, запись входных данных в который происходит по высокому уровню синхросигнала (вход триггера "С"). На рисунке такой сигнал назван "запись в защелку". Мощность сигнала с инвертирующего выхода триггера усиливается при помощи МОП (металл — окись — полупроводник) транзистора, а за тем поступает на внешний вывод микросхемы. Внутренний подтягивающий резистор служит для обеспечения выходного тока порта при установленном высоком логическом уровне. Как правило, вместо резистора в портах микроконтроллера используется управляемый генератор стабильного тока (ГСТ), собранный на МОП транзисторах. Следует обратить внимание, что у некоторых портов вывода, внутреннего генератора тока может и не быть, пример тому P0.

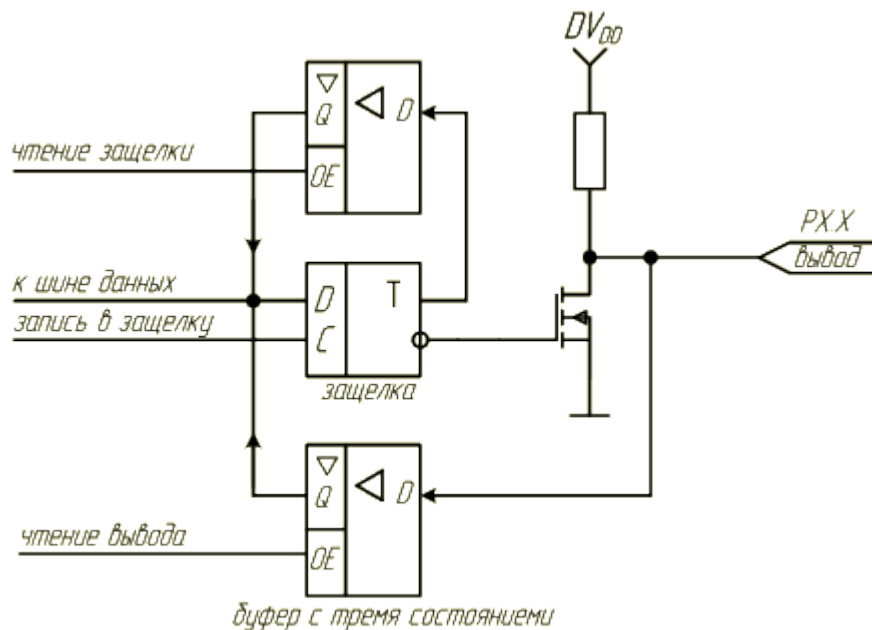


Рисунок 4 Упрощенная схема одного бита параллельного квазидвунаправленного порта (архитектура MCS-51)

Подключение внешних устройств к параллельному порту микроконтроллера

Внешними устройствами называются любые устройства, которыми управляет, от которых получает или которым передает информацию микроконтроллер. В качестве внешних устройств могут выступать устройства ввода-вывода информации — светодиодный индикатор, дисплей, датчик, кнопка, клавиатура, другой микропроцессор.

Внутренняя схема порта построена таким образом, чтобы максимально упростить подключение внешних устройств к микроконтроллеру. При этом необходимо следить за максимальной допустимой мощностью, рассеиваемой на микросхеме и отдельных выводах порта.

Например присутствие в схеме порта выходного транзистора позволяет подключить к выводам порта светодиодные индикаторы непосредственно, без усилителя мощности, как это показано на рисунке 5.

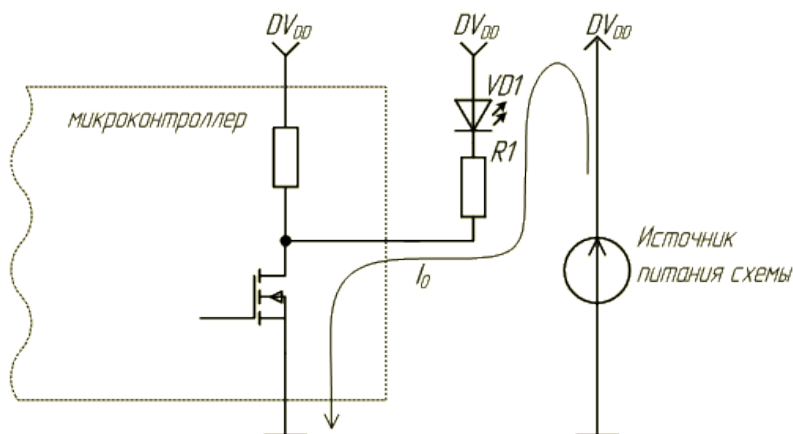


Рисунок 5 Эквивалентная схема подключения светодиодного индикатора к параллельному порту

Светодиод в такой схеме загорается при низком потенциале на выводе порта, для этого в порт должен быть записан логический ноль. Резистор R1 ограничивает ток через светодиод, если этот резистор не поставить, то ток по цепи индикатора может достигнуть недопустимой величины и светодиод или внутренний транзистор выйдут из строя.

Если же требуется обеспечить больший ток нагрузки, то для усиления тока порта используют внешний транзистор, как показано на рисунке 6.

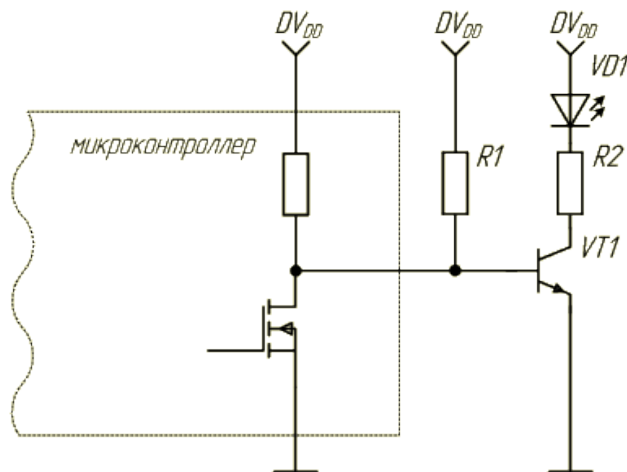


Рисунок 6 Схема подключения светодиодного индикатора через транзисторный ключ

В приведенной схеме база транзистора подключена непосредственно к выводу порта. Если выходного порта контроллера достаточно для открывания транзистора, то резистор R1 может быть исключен из схемы. Этот резистор нужен для увеличения тока базы транзисторного ключа. Резистор R2 рассчитывается исходя из допустимого тока светодиода. Для зажигания светодиода необходимо в соответствующий разряд параллельного порта записать логическую единицу, а для его гашения — логический ноль.

Двунаправленный порт позволяет получать цифровую информацию от различных источников — кнопок, датчиков, микросхем. Согласование микросхем между собой не представляет трудностей, так как практически все современные цифровые микросхемы по входу и выходу согласованы между собой и имеют строго определенные значения логических уровней. С датчиками и кнопками дело обстоит несколько сложнее. Все датчики выполняются так, что они с точки зрения электрической схемы представляют собой контакты, работающие на замыкание-размыкание. Поэтому схема подключения датчиков и кнопок принципиально не отличаются. Со стороны микроконтроллера замыкание-размыкание должно интерпретироваться как подача на вход логических уровней. Простейшая схема, преобразующая замыкание контактов в логические уровни показана на рисунке 7.

Когда контакт К разомкнут то на вход микроконтроллера через резистор R подается напряжение питания, интерпретируемое контроллером как логическая единица. Если контакт замкнут, то на вход подается потенциал общего провода, что соответствует логическому нулю.

Для того чтобы ввести информацию с параллельного порта в него должны быть записаны логические единицы. Дело в том, что если в разряд записать логическую единицу, то выходной внутренний транзистор закрывается и на выводе микросхемы за счет внутреннего подтягивающего резистора устанавливается высокий уровень. Этот уровень может быть изменен на нулевой потенциал замыканием вывода микросхемы на общий провод. И информация может быть правильно интерпретирована микропроцессором. В случае же когда в разряд порта записан логический ноль, внутренний транзистор открыт, на выходе появляется низкий потенциал, изменить который извне невозможно. Поэтому, перед

тем как осуществить ввод информации через параллельный порт, соответствующий разряд необходимо настроить на ввод — записать в него логическую единицу.

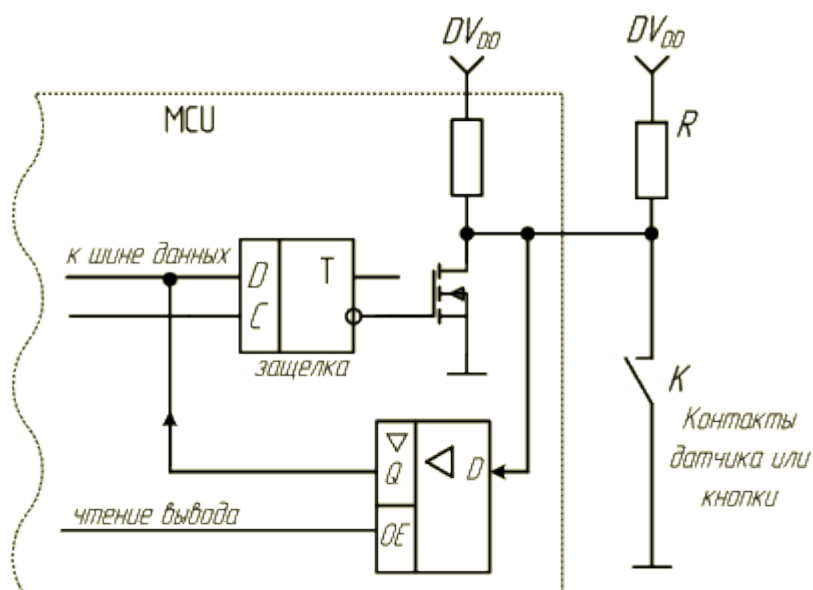


Рисунок 7 Подключение источника цифровой информации

Таймеры и процессоры событий

Большинство задач управления, которые реализуются с помощью МК, требуют исполнения их в реальном времени. Под этим понимается способность системы получить информацию о состоянии управляемого объекта, выполнить необходимые расчетные процедуры и выдать управляющие воздействия в течение интервала времени, достаточного для желаемого изменения состояния объекта.

Возлагать функции формирования управления в реальном масштабе времени только на центральный процессор неэффективно, так как это занимает ресурсы, необходимые для расчетных процедур. Поэтому в большинстве современных МК используется аппаратная поддержка работы в реальном времени с использованием таймера (таймеров).

Модули таймеров служат для приема информации о времени наступления тех или иных событий от внешних датчиков событий, а также для формирования управляющих воздействий во времени.

Модуль таймера 8-разрядного МК представляет собой 8-ми или 16-разрядный счетчик со схемой управления. Схемотехникой МК обычно предусматривается возможность использования таймера в режиме счетчика внешних событий, поэтому его часто называют таймером / счетчиком. Структура типичного 16-разрядного таймера / счетчика в составе МК приведена на рисунке 8

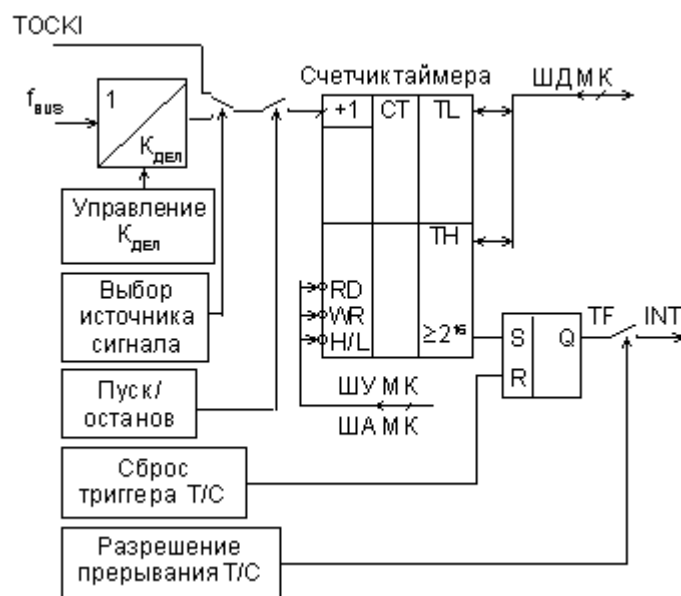


Рисунок 8 Структура модуля таймера/счетчика.

В памяти МК 16-разрядный счетчик отображается двумя регистрами: TH — старший байт счетчика, TL — младший байт. Регистры доступны для чтения и для записи. Направление счета — только прямое, то есть при поступлении входных импульсов содержимое счетчика инкрементируется. В зависимости от настройки счетчик может использовать один из источников входных сигналов:

- импульсную последовательность с выхода управляемого делителя частоты f_{BUS} ;
- сигналы внешних событий, поступающие на вход ТОСКІ контроллера.

В первом случае говорят, что счетчик работает в режиме таймера, во втором — в режиме счетчика событий. При переполнении счетчика устанавливается в "единицу" триггер переполнения TF, который генерирует запрос на прерывание, если прерывания от таймера разрешены. Пуск и останов таймера могут осуществляться только под управлением программы. Программным способом можно также установить старший и младший биты счетчика в произвольное состояние или прочитать текущий код счетчика.

Рассмотренный "классический" модуль таймера / счетчика широко применяется в различных моделях относительно простых МК. Он может использоваться для измерения временных интервалов и формирования последовательности импульсов. Основными недостатками "классического" таймера / счетчика являются:

- потери времени на выполнение команд пуска и останова таймера, приводящие к появлению ошибки при измерении временных интервалов и ограничивающие минимальную длительность измеряемых интервалов времени единицами мс;
- сложности при формировании временных интервалов (меток времени), отличных от периода полного коэффициента счета, равного $(K_{дел}/f_{BUS}) \cdot 2^{16}$;
- невозможность одновременного обслуживания (измерения или формирования импульсного сигнала) сразу нескольких каналов.

Первые два из перечисленных недостатков были устранены в усовершенствованном модуле таймера / счетчика, используемом в МК семейства MCS-51 (Intel). Дополнительная логика счетного входа позволяет тактовым импульсам поступать на вход счетчика, если уровень сигнала на одной из линий ввода равен "1". Такое решение повышает точность измерения временных интервалов, так как пуск и останов таймера производится аппаратно. Также в усовершенствованном таймере реализован режим перезагрузки счетчика

произвольным кодом в момент переполнения. Это позволяет формировать временные последовательности с периодом, отличным от периода полного коэффициента счета.

Однако эти усовершенствования не устраняют главного недостатка модуля "классического" таймера — одноканального режима работы. Совершенствование подсистемы реального времени МК ведется по следующим направлениям:

- увеличение числа модулей таймеров / счетчиков. Этот путь характерен для фирм, выпускающих МК со структурой MCS-51, а также для МК компаний Mitsubishi и Hitachi;
- модификация структуры модуля таймера / счетчика, при которой увеличение числа каналов достигается не за счет увеличения числа счетчиков, а за счет введения дополнительных аппаратных средств входного захвата (input capture — IC) и выходного сравнения (output compare — OC). Такой подход используется, в частности, в МК компании Motorola.

Принцип действия канала входного захвата таймера / счетчика иллюстрирует рисунок 9

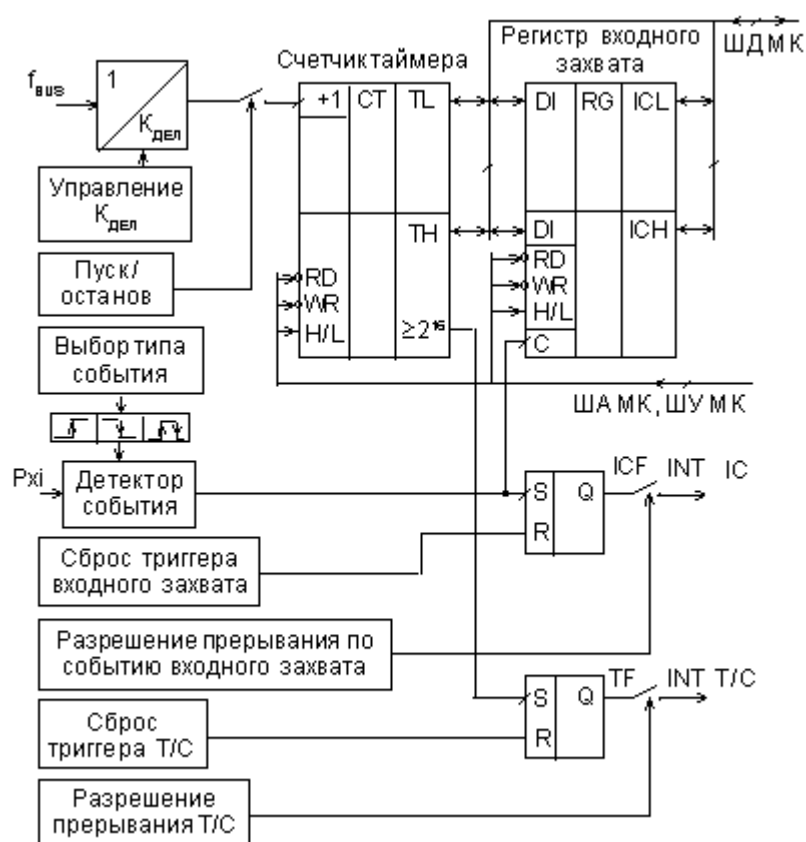


Рисунок 9 Структурная схема канала входного захвата таймера.

Схема детектора события "наблюдает" за уровнем напряжения на одном из входов МК. Чаще всего это одна из линий порта ввода/вывода. При изменении уровня логического сигнала с "0" на "1" и наоборот вырабатывается строб записи, и текущее состояние счетчика таймера записывается в 16-разрядный регистр входного захвата. Описанное действие в микропроцессорной технике называют событием захвата. Предусмотрена возможность выбора типа сигнала на входе, и это воспринимается как событие:

- положительный (передний) фронт сигнала;
- отрицательный (задний) фронт сигнала;
- любое изменение логического уровня сигнала.

Выбор типа события захвата устанавливается в процессе инициализации таймера и

может неоднократно изменяться в ходе выполнения программы. Каждое событие захвата приводит к установке в "1" триггера входного захвата и появлению на его выходе флага (признака) входного захвата ICF. Состояние триггера входного захвата может быть считано программно, а если прерывания по событию захвата разрешены — формируется запрос на прерывание INT IC.

Использование режима входного захвата позволяет исключить ошибки измерения входного интервала времени, связанные со временем перехода к подпрограмме обработки прерывания, так как копирование текущего состояния счетчика осуществляется аппаратными, а не программными средствами. Однако время перехода на подпрограмму обработки прерывания накладывает ограничение на длительность измеряемого интервала времени, так как предполагается, что второе событие захвата произойдет позже, чем код первого события будет считан МК.

Структура аппаратных средств канала выходного сравнения представлена на рисунке 10

Цифровой компаратор непрерывно сравнивает текущий код счетчика таймера с кодом, который записан в 16-разрядном регистре выходного сравнения. В момент равенства кодов на одном из выходов МК (Pxj на рисунке 10) устанавливается заданный уровень логического сигнала. Обычно предусмотрено три типа изменения сигнала на выходе Pxj в момент события выходного сравнения:

- установка высокого логического уровня;
- установка низкого логического уровня;
- инвертирование сигнала на выходе.

При наступлении события сравнения устанавливаются в "1" триггер выходного сравнения и соответствующий ему признак выходного сравнения OCF. Аналогично режиму входного захвата состояние триггера выходного сравнения может быть считано программно, а если прерывания по событию сравнения разрешены — формируется запрос на прерывание INT OC.

Режим выходного сравнения предназначен, прежде всего, для формирования временных интервалов заданной длительности. Длительность сформированного временного интервала определяется только разностью кодов, последовательно загружаемых в регистр выходного сравнения, и не зависит от программного обеспечения МК. Время, необходимое для записи нового значения кода в регистр канала сравнения, ограничивает минимальную длительность формируемого временного интервала.

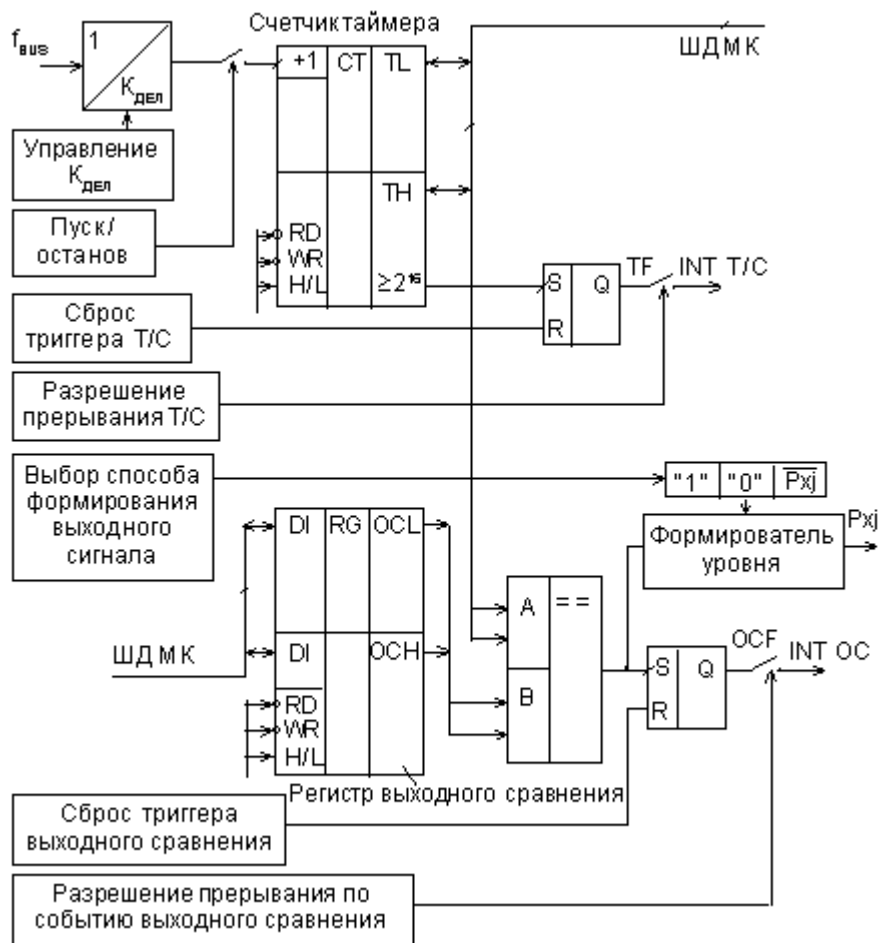


Рисунок 10 Структурная схема канала выходного сравнения таймера

Модули усовершенствованного таймера используются в составе МК в различных модификациях. При этом число каналов входного захвата и выходного сравнения в модуле может быть различным. Так, в МК семейства HC05 фирмы Motorola типовыми решениями являются модули 1IC+1OC или 2IC+2OC, а модуль таймера в составе МК только один. В ряде модулей каналы могут быть произвольно настроены на функцию входного захвата или выходного сравнения посредством инициализации. Счетчик модуля усовершенствованного таймера может не иметь функции программного останова. В этом случае состояние счетчика нельзя синхронизировать с каким-либо моментом работы МК, и такой счетчик характеризуется как свободно считающий (free counter).

Аппаратные средства усовершенствованного таймера позволяют решить многие задачи управления в реальном времени. Однако по мере роста сложности алгоритмов управления отчетливо проявляются ограничения модулей усовершенствованного таймера, а именно:

- недостаточное число каналов захвата и сравнения, принадлежащих одному счетчику временной базы. Это не позволяет сформировать синхронизированные между собой многоканальные импульсные последовательности;
- однозначно определенная конфигурация канала (или захват или сравнение) часто не удовлетворяет потребностям решаемой задачи;
- формирование сигналов по методу широтно-импульсной модуляции (ШИМ) требует программной поддержки, что снижает максимально достижимую частоту выходного сигнала.

Процессоры событий

Поэтому следующим этапом развития модулей подсистемы реального времени МК

стали модули процессоров событий. Впервые модули процессоров событий были использованы компанией Intel в МК семейства 8xC51Fx. Этот модуль получил название программируемого счетного массива (Programmable Counter Array — PCA).

PCA обеспечивает более широкие возможности работы в реальном масштабе времени и в меньшей степени расходует ресурсы центрального процессора, чем стандартный и усовершенствованный таймеры / счетчики. К преимуществам PCA также можно отнести более простое программирование и более высокую точность. К примеру, PCA может обеспечить лучшее временное разрешение, чем таймеры 0, 1 и 2 МК семейства MCS-51, так как счетчик PCA способен работать с тактовой частотой, втрое большей, чем у этих таймеров. PCA также может решать многие задачи, выполнение которых с использованием таймеров требует дополнительных аппаратных затрат (например, определение фазового сдвига между импульсами или генерация ШИМ-сигнала). PCA состоит из 16-битного таймера - счетчика и пяти 16-битных модулей сравнения-защелки, как показано рисунке 11.

Таймер - счетчик PCA используется в качестве базового таймера для функционирования всех пяти модулей сравнения-защелки. Вход таймера - счетчика PCA может быть запрограммирован на счет сигналов от следующих источников:

- выход делителя на 12 тактового генератора МК;
- выход делителя на 4 тактового генератора МК;
- сигнал переполнения таймера 0;
- внешний входной сигнал на выводе ECI (P1.2).

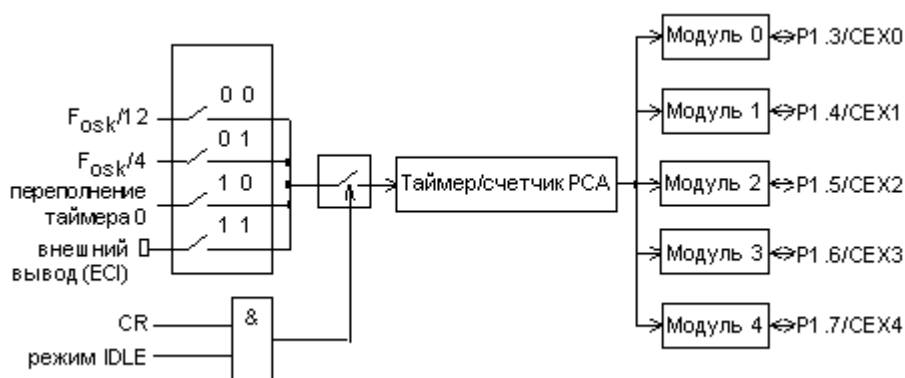


Рисунок 11 Структура процессора событий МК семейства Intel 8xC51Fx

Любой из модулей сравнения-защелки может быть запрограммирован для работы в следующих режимах:

- защелкивания по фронту и/или спаду импульса на входе CEX_i;
- программируемого таймера ;
- высокоскоростного выхода;
- широтно-импульсного модулятора.

Модуль 4 может быть также запрограммирован как сторожевой таймер (Watchdog Timer – WDT).

Режим защелкивания по импульсу на входе МК эквивалентен режиму входного захвата (IC) усовершенствованного таймера. Режимы программируемого таймера и высокоскоростного выхода близки по своим функциональным возможностям к режиму выходного сравнения (OC).

В режиме ШИМ на соответствующем выводе МК формируется последовательность

импульсов с периодом, равным периоду базового таймера / счетчика PCA. Значение 8-разрядного кода, записанное в младший байт регистра-защелки соответствующего модуля задает скважность формируемого сигнала. При изменении кода от 0 до 255 скважность меняется от 100 % до 0,4 %.

Режим ШИМ очень прост с точки зрения программного обслуживания. Если изменения скважности не предполагается, то достаточно один раз занести соответствующий код в регистр данных модуля, проинициализировать режим ШИМ, и импульсная последовательность будет воспроизводиться с заданными параметрами без вмешательства программы.

Назначение и особенности работы сторожевого таймера будут рассмотрены далее отдельно.

При работе модуля сравнения-защелки в режиме защелки, программируемого таймера или высокоскоростного выхода модуль может сформировать сигнал прерывания. Сигналы от всех пяти модулей сравнения-защелки и сигнал переполнения таймера PCA разделяют один вектор прерывания. Иными словами, если прерывания разрешены, то и сигнал переполнения таймера PCA и сигнал от любого из модулей вызывают одну и ту же подпрограмму прерываний, которая должна сама идентифицировать источник, вызвавший ее.

Для работы с внешними устройствами таймер - счетчик PCA и модули сравнения-защелки используют выходы P1 порта МК. Если какой-либо вывод порта не используется при работе PCA, или PCA не задействован, порт может применяться стандартным образом.

Реализованный в 8xС51FX PCA оказался настолько удачным, что архитектура данных МК стала промышленным стандартом де-факто, а сам PCA многократно воспроизводился в различных модификациях микроконтроллеров разных фирм.

Тенденция развития подсистемы реального времени современных МК находит свое отражение в увеличении числа каналов процессоров событий и расширении их функциональных возможностей.

Модуль прерываний МК

Обработка прерываний в МК происходит в соответствии с общими принципами обработки прерываний в МПС. Модуль прерываний принимает запросы прерывания и организует переход к выполнению определенной прерывающей программы. Запросы прерывания могут поступать как от внешних источников, так и от источников, расположенных в различных внутренних модулях МК. В качестве входов для приема запросов от внешних источников чаще всего используются выходы параллельных портов ввода/вывода, для которых эта функция является альтернативной. Источниками запросов внешних прерываний также могут быть любые изменения внешних сигналов на некоторых специально выделенных линиях портов ввода/вывода.

Источниками внутренних запросов прерываний могут служить следующие события:

- переполнение таймеров / счетчиков ;
- сигналы от каналов входного захвата и выходного сравнения таймеров / счетчиков или от процессора событий ;
- готовность памяти EEPROM;
- сигналы прерывания от дополнительных модулей МК, включая завершение передачи или приема информации по одному из последовательных портов и другие.

Любой запрос прерывания поступает на обработку, если прерывания в МК разрешены и разрешено прерывание по данному запросу. Адрес, который загружается в программный счетчик при переходе к обработке прерывания, называется "вектор прерывания". В зависимости от организации модуля прерываний конкретного МК различные источники прерываний могут иметь разные векторы или использовать некоторые из них совместно. Использование различными прерываниями одного вектора обычно не вызывает проблем при разработке программного обеспечения, так как аппаратная часть МК фиксирована, а контроллер чаще всего выполняет одну-единственную программу.

Вопрос о приоритетах при одновременном поступлении нескольких запросов на прерывание решается в различных МК по-разному. Есть МК с одноуровневой системой приоритетов (все запросы равноценны), многоуровневой системой с фиксированными приоритетами и многоуровневой программируемой системой приоритетов.

Отдельно необходимо описать аппаратные прерывания, связанные с включением питания, подачей сигнала "сброс" и переполнением сторожевого таймера. Они имеют немаскируемый характер и чаще всего разделяют один общий вектор прерывания. Это вполне логично, поскольку результатом каждого из событий является начальный сброс МК.

Вопросы для контроля

Что такое порт ввода/вывода?

Как порт ввода вывода представляется с точки зрения внешнего устройства? Сточки зрения микропроцессора?

В чем отличие параллельных и последовательных портов?

Перечислите и кратко опишите четыре вида портов(классификация по направлению перемещения данных)

Опишите, на основании функциональной схемы, конструкцию и работу порта ввода

Опишите, на основании функциональной схемы, конструкцию и работу порта вывода

Опишите, на основании функциональной схемы, конструкцию и работу

двунаправленного порта ввода/вывода

Опишите, на основании функциональной схемы одного бита, конструкцию и работу квазидвунаправленного порта

Как подключить к параллельному порту светодиодный индикатор?

Как подключить к параллельному порту светодиодный индикатор с усилением тока нагрузки?

Как подключить к параллельному порту источник цифровой информации?

Для чего в МК используются таймеры?

Перечислите функциональные узлы таймера/счетчика указанные на схеме и назовите их назначение.

В чем недостатки "классического" таймера? Как их преодолеть?

Для чего используются процессоры событий?

Опишите структуру процессора событий на примере Intel8xC51Fx

В каких режимах может работать модуль сравнения?

Для чего нужен модуль прерываний?

Какие события могут служить источниками внутренних запросов?

Что такое вектор прерывания?

Источники

<http://www.labfor.ru/guidance/mpu-leso1/2>