

Специализированные процессоры. Процессоры для встраиваемых применений. Конспект лекций. Разделы 1- 2.

Кустарев Павел Валерьевич.
Кафедра Вычислительной Техники
СПбГИТМО (ТУ)
Санкт-Петербург 2002г.

email: kustarev@lmt.ifmo.ru
URL: <http://embedded.ifmo.ru>

Оглавление.

1	Общие сведения.....	2
1.1	Основные понятия.....	2
1.2	Классификация специализированных процессоров.....	3
2	Процессоры встраиваемых систем.....	5
2.1	Общие сведения.....	5
2.2	Типовая структура процессора для встраиваемых систем.....	6
2.3	Процессорное ядро.....	7
2.4	Модули резидентной памяти.....	8
2.5	Периферийные модули процессоров для встраиваемых применений.....	9
2.5.1	Порты ввода-вывода.....	9
2.5.2	Таймеры-счетчики.....	16
2.5.3	Модули таймеров-счетчиков со схемами входного захвата, выходного сравнения и выработки сигналов с ШИМ (CAPTURE/COMPARE/PWM (CCP)).	17
2.5.4	Аналого-цифровой преобразователь (АЦП).....	20
2.5.5	Цифро-аналоговый преобразователь (ЦАП).....	22
2.5.6	Контроллеры последовательных интерфейсов.....	23
2.6	Организация прерываний в управляющих процессорах.....	24
2.7	Подсистема синхронизации.....	26
2.8	Механизмы начальной инициализации встроенной памяти.....	28

1 Общие сведения.

1.1 Основные понятия.

Процессор – основной блок ЭВМ, осуществляющий обработку данных. В настоящее время основным типом процессоров являются цифровые программно управляемые процессоры, построенные на базе цифровых электронных схем и использующих для обработки данных методы Булевой алгебры и двоичной арифметики, реализующие произвольные алгоритмы, описанный в ограниченном базисе некоторой системы команд.

Микропроцессор – процессор, реализованный в виде одной большой интегральной схемы (БИС). Существуют блочно-секционные процессоры, собираемые из нескольких одинаковых микросхем - секций. За счет этого достигается произвольная разрядность процессора. К блочно-секционным процессорам относится, например, микропроцессорный комплект К1804.

Однокристалльная ЭВМ (ОКМЭВМ) или микроконтроллер – БИС, структурная схема которой содержит все функциональные узлы, необходимые для обеспечения автономной работы в качестве вычислительного или управляющего устройства. На кристалле ОКМЭВМ располагаются: процессор, блоки постоянной и оперативной памяти (ПЗУ и ОЗУ), периферийные устройства различного типа, блоки управления и синхронизации и, возможно, некоторые другие блоки.

Универсальный процессор – процессор, имеющий архитектуру, набор структурных блоков, систему команд и конструктивно-технологическое исполнение, позволяющее одинаково эффективно применять его для решения достаточно широкого круга разнотипных задач и использовать в различных условиях.

Специализированный процессор – процессор, у которого особенности архитектуры, набора структурных блоков, системы команд или конструктивно-технологического исполнения, позволяют значительно повысить эффективность решения достаточно узкого круга специальных задач по сравнению с иными применениями.

Архитектура процессора или вычислительной системы – это совокупность ключевых решений по следующим направлениям:

- выбор структурных элементов процессора (АЛУ, блоков регистровой памяти, контроллера прерываний, блока синхронизации и др.) и их интерфейсов для связи между собой и с внешними устройствами;
- поведение каждого из этих структурных элементов, специфицированное в кооперациях с другими элементами (то есть описание функционирования элемента с учетом взаимодействия с ним других элементов);
- составление из этих элементов все более крупных подсистем;
- единый архитектурный стиль, направляющий и определяющий всю организацию процессора (его элементов, их интерфейсов и взаимодействия).

Архитектура процессора влияет на следующие характеристики:

- стоимость использования;
- надежность функционирования;
- реактивность-быстродействие-производительность;
- простота применения (проектирования систем на его основе);

- способность к реконфигурации;
- другие.

Стоимость использования процессора включает кроме стоимости микросхем(ы), также и стоимость проектирования архитектуры, аппаратуры, ПО, тестирования, документирования систем на базе процессора;

Надежность функционирования, в простейшем случае, измеряется в параметрах среднего времени между отказами.

Реактивность (задержка обслуживания), быстродействие и средняя/пиковая производительность взаимосвязанные характеристики производительности системы в различных ситуациях.

Способность к реконфигурации – возможность и время переконфигурирования системы при сбоях.

1.2 Классификация специализированных процессоров.

1. По выполняемым функциям и области применения:
 - a. Процессоры встраиваемых (управляющих) систем (embedded processor):
 - Универсальные;
 - С расширенными коммуникационными возможностями;
 - С расширенными возможностями дискретного ввода-вывода;
 - С расширенными возможностями обработки аналоговых сигналов (mixed signal processor (MSP));
 - b. Коммуникационные процессоры – имеют специальную поддержку аппаратных интерфейсов и протоколов коммуникационных систем:
 - Сетевые – поддерживают распространенные сетевые и современные периферийные интерфейсы: Ethernet, HDLC, X.25, T1, ATM, USB, High Speed UART и др. Выпускаются фирмами Motorola (MC683xx, MPC8xx), AMD (Am186CC);
 - Модемные – поддерживают протоколы серий V2x, V3x передачи данных по синхронным и асинхронным модемным каналам.
 - c. Процессоры цифровой обработки сигналов (digital signal processor (DSP)) – реализуют методы цифровой обработки сигналов: фильтрацию, спектральный анализ, смешение сигналов, масштабирование. Отличительная особенность DSP-процессоров – поточная обработка больших объемов данных в реальном времени, требующая высокой производительности, но ограниченного набора операций. Основные производители – Texas Instruments (семейство TMS320), Analog Devices (семейство ADSP21xxx), Motorola (семейства DSP56xxx, DSP96xxx);
 - d. Медийные процессоры – ориентированные на обработку видео- и звуковой информации:
 - С аппаратной поддержкой мультимедийной обработки (медиапроцессоры) – имеют аппаратные блоки и развитую систему команд для обработки и передачи аудио и графических данных, видеоизображений. Используются в аудио- и видеоадаптерах в персональном компьютере, в игровых приставках, в бытовой технике. Примеры: медиапроцессор систем связи компании MicroUtility, универсальные медиапроцессоры Trimedia (Philips), MediaGX (Ciryx), видеопроцессоры серии NV1..5 (Nvidia);

- С мультимедийным расширением набора команд (Intel MMX и более старшие модели, UltraSPARC (Sun Microsystems)).
 - e. Со-процессоры:
 - Математические;
 - Ввода-вывода.
 - f. Транспьютеры – процессор для построения массово-параллельных систем. Имеет на кристалле локальное процессорное ядро и память, а также специальные быстродействующие каналы связи для взаимодействия с другими транспьютерами (линки). Основным разработчиком и производителем транспьютеров – компания Inmos, Inc (Великобритания).
 - g. Другие типы спецпроцессоров:
 - Нейропроцессоры – ориентированы на построение систем с архитектурой нейронных сетей. Такие системы применяются для решения плохо формализуемых задач: распознавание образов, предсказание поведения систем, оптимизация и т.п.;
 - Процессоры языков высокого уровня (JAVA, Forth и др.);
 - Узкоспециализированные (медицинская техника, военные, для научно-исследовательских систем и т.п.).
2. По полноте вычислительного ядра:
- a. Со-процессоры (арифметические, ввода-вывода);
 - b. Полнофункциональные (обычные) процессоры;
 - c. С расширенной периферией (например, процессоры для встроенных применений i386EX (Intel), elan520(AMD));
 - d. ОКМЭВМ;
 - e. Однокристалльные мультипроцессорные системы (TriCore (Motorola)).
3. По архитектуре вычислительного ядра:
- a. По разрядности ядра:
 - i. Типовые (с разрядностью 4, 8, 16, 32, 64);
 - ii. С большей разрядностью. На данный момент не распространены;
 - iii. С нестандартной (не кратной 8-ми) разрядностью. Серийных моделей не выпускается, могут разрабатываться для узкоспециализированных применений;
 - iv. Масштабируемые (блочнo-секционные).
 - b. По организации памяти:
 - С Гарвардской архитектурой – с отдельной памятью программ и памятью данных. Не допускается запуск исполнения кода, расположенного в памяти данных;
 - С Принстонской архитектурой (архитектурой Фон-Неймана) – с единой памятью для хранения команд и данных. Данные могут интерпретироваться как исполняемый код.
 - c. По системе команд:
 - CISC - с традиционным набором команд. Команды могут иметь разные форматы, различную длину и время исполнения. В результате для их дешифрации и исполнения необходимо более сложное и соответственно медленное устройство управления, затруднена конвейерная обработка потока команд;
 - RISC – с сокращенным набором команд. Используется небольшое число форматов команд с одинаковой длиной и временем исполнения. За счет этого упрощаются блоки выборки, дешифрации и исполнения

команд, увеличивается их быстродействие, облегчается организация конвейерной обработки;

d. По уровню распараллеливания вычислительного процесса:

- Суперскалярные – с единым вычислительным протоком и распараллеливанием команд на этапе исполнения. Процессор самостоятельно выбирает операции для параллельной обработки из последовательности команд и организует такую обработку. Суперскалярная обработка в большинстве случаев организуется помощью конвейера команд. В более сложных процессорах присутствуют несколько конвейеров и устройств обработки команд, но командный поток остается один. Суперскалярную архитектуру имеют большинство современных серийно выпускаемых процессоров;
- С длинным командным словом (VLIW) – распараллеливание команд выполняется на уровне программного кода: в каждом кодовом слове перечисляются несколько операций, которые будут выполняться параллельно разными операционными устройствами процессора. Такое распараллеливание выполняется специальными трансляторами с языков высокого уровня;
- Мультискалярные – имеют несколько операционных устройств, обрабатывающих собственные потоки команд. Процессор выбирает из программы несколько задач и выдает их для исполнения различным операционным устройствам. Выделение задач выполняется компиляторами и самим процессором. Мультискалярная архитектура в основном пока находится на стадии экспериментальных разработок;

4. Другие варианты классификации:

- a. По энергопотреблению;
- b. По технологическому исполнению.

2 Процессоры встраиваемых систем.

2.1 Общие сведения.

Встраиваемые вычислительные системы (ВС) – системы непосредственно, без посредничества человека, взаимодействующие с обслуживаемыми объектами посредством датчиков и исполнительных устройств - актуаторов.

Управляющие системы – класс встраиваемых систем, реализующих функции автоматического или автоматизированного управления.

Специфика встраиваемых систем:

- непосредственное подключение к объекту;
- работа в режимах жесткого реального времени;
- обычно, невысокая сложность математических вычислений, но необходима поддержка алгоритмов автоматического регулирования (различных регуляторов, конечных автоматов);
- растущие требования к коммуникационным возможностям для организации комплексных распределенных систем управления.
- повышенные требования к надежности и безопасности функционирования;
- жесткие условия эксплуатации (широкий диапазон температур, помехи);

- использование в малогабаритных, автономных и переносных системах предъявляет требования невысокого энергопотребления, малых габаритов, минимального числа вспомогательных элементов.

Процессоры для встраиваемых систем должны эффективно решать перечисленные выше задачи за счет адаптации аппаратной структуры, системы команд и конструктивно-технологического исполнения.

Области применения встраиваемых систем и процессоров для встраиваемых систем:

- промышленные информационно-управляющие системы (ИУС);
- транспортные ИУС;
- бытовые электронные приборы и системы;
- охранные системы;
- военные, медицинские, космические и другие ИУС и системы сбора данных.

2.2 Типовая структура процессора для встраиваемых систем.

В настоящее время выпускается большое количество разнообразных по структуре и функциям процессоров для применения во встроенных системах. Эта номенклатура постоянно расширяется, чтобы обеспечить решение специфических задач в различных прикладных задачах. Возможность разработки и производства новых моделей в сжатые сроки обеспечивает *модульный принцип структурной организации*.

При модульном принципе построения все процессоры одного семейства содержат в себе одинаковый базовый функциональный блок – процессорное ядро, и изменяемый функциональный блок.

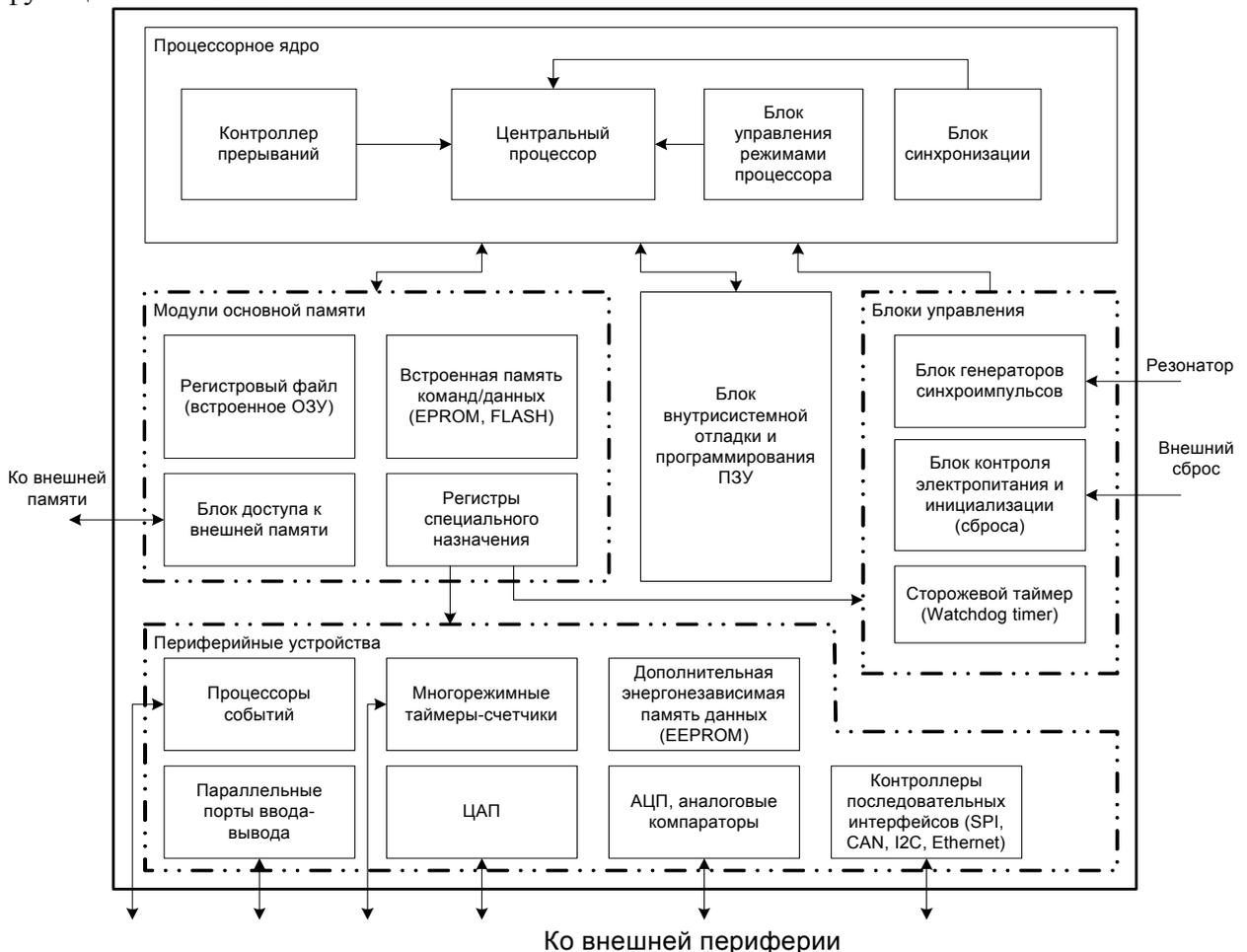


Рисунок 1 Типовая структура процессора для встроенных систем.

Базовый блок (процессорное ядро) включает:

- Центральный процессор;
- Внутренние магистрали адреса, данных и управления;
- Блок формирования множества сигналов с различными фазами и частотами для синхронизации центрального процессора и внутренних магистралей.
- Блок управления режимами работы процессора, который может настраивать процессор на активный режим, режим обработки прерываний, несколько режимов пониженного энергопотребления, режим рестарта.

Процессорное ядро является основным отличительным признаком архитектуры определенного семейства процессоров, поэтому его (ядро) называют по названию семейства. Например, ядро MCS-51 или ядро PIC16.

Изменяемый функциональный блок включает:

- модули памяти различных типов: оперативную память данных типа SRAM, постоянную память команд (программ) типов ROM, EPROM или FLASH, энергонезависимую память данных типа EEPROM;
- Модули периферийных устройств;
- Модули управления и синхронизации.

В различных микросхемах семейства может иметься различный набор модулей изменяемого функционального блока. Общую совокупность модулей, реализованных в микросхемах одного семейства, называют *библиотекой периферийных модулей* данного семейства. В данную библиотеку входят, как говорилось, не только периферийные, но и модули памяти, встроенные генераторы синхронизации, блок контроля электропитания и формирования сигналов рестарта системы в случае сбоев или «внешнего сброса», модули внутрисхемной отладки и программирования. В последнее время активно развивается направление «System-On-Chip», когда конечный пользователь сам может формировать структуру специализированного процессора из предоставленной библиотеки периферийных модулей, а также самостоятельно разрабатывать новые модули.

2.3 Процессорное ядро.

Техническое решение процессорного ядра определяют следующие параметры:

- Архитектурные – набор регистров, организация памяти, способы адресации операндов в памяти, система команд для обработки этих данных.
- Схемотехнические решения – схемы регистров, АЛУ, схемы управления магистралями и т.п. Схемотехника определяет также внутреннюю диаграмму функционирования – последовательность перемещения данных по магистралям между регистрами, памятью, АЛУ.
- Технология производства – определяет допустимую сложность схемы, максимальную частоту переключений, энергопотребление.

В современных процессорах для встраиваемых систем реализуют как CISC-архитектуру (Motorola HC11, Intel MCS-51, AMD Am186 и др.), так и RISC-архитектуру (MicrochipPIC, Atmel AVR, Triscend E7-ARM).

Производительность процессорного ядра определяется комплексом факторов:

- Частотой тактирования межмодульных магистралей адреса и данных Fbus. Она определяется из частоты генератора синхронизации Fxclk по соотношению, индивидуальному для каждого процессорного ядра. Например, для MCS51 – $Fxclk/Fbus = 12$ и при частоте генератора 12МГц ядро работает на частоте 1МГц; для Am186ES - $Fxclk/Fbus = 1$; для Motorola HC08 существует режим умножения входной частоты и $Fxclk/Fbus < 1$.
- Количеством пересылок регистр-регистр за единицу времени. Для RISC-процессоров это одна пересылка за такт шины, для CISC – 1..3 пересылки (они медленнее).

- Производительностью при выполнении операций наиболее используемым в конкретном алгоритме управления. Например, для ПИД – регуляторов – это операции умножения/деления; для простых конечных автоматов – это логические операции.
- Временем вызова/возврата подпрограммы обработки прерывания. Этот параметр значим для функционирования в режиме жесткого реального времени и определяет максимальную интенсивность обрабатываемых событий.

2.4 Модули резидентной памяти.

Термин «модуль памяти» подразумевает объединение собственно массивов ячеек памяти со специальными аналоговым и цифровыми схемами управления режимами записи–стирания, со схемами (и иногда источниками) электропитания, с регистрами управления режимами.

Модули памяти делятся на ПЗУ и ОЗУ.

Модули ПЗУ бывают:

- ПЗУ масочного типа (MaskROM) – записывается на заводе-изготовителе и не может быть изменено пользователем. Обладают высоким качеством хранения. Самый дешевый тип ПЗУ. Используются для изделий, выпускаемых большими партиями в несколько десятков тысяч штук.
- ПЗУ, однократно программируемые пользователем (One-Time Programmable ROM (OTPROM)). Программируется пользователем. При выпуске на заводе все ячейки имеют значения 0xFF. Подачей импульсов напряжения в биты могут быть записаны «0», но обратная запись – в «1» уже не возможна. Имеют высокое качество хранения при строгом соблюдении режимов программирования (уровни напряжения, временная диаграмма, режимы проверки), в противном случае через некоторое время (месяцы-годы) биты могут самопроизвольно «распрограммироваться» - перейти в состояние «1». Дешевое ПЗУ. Используется для небольших партий изделий.
- ПЗУ программируемое пользователем, со стиранием ультрафиолетовыми или рентгеновскими лучами (EPROM). Допускается многократное перепрограммирование (несколько десятков раз). Технология программирования схожа с OTPROM, но может быть осуществлено стирание всех запрограммированных в «0» ячеек в состояние «1» под УФ лучами. Для того на корпусе есть специальное окно из кварцевого стекла. Нарушение режимов стирания программирования приводит к резкому сокращению числа циклов перепрограммирования и времени хранения. Очень дорогая память (примерно на порядок дороже чем OTPROM). Используется в отладочных образцах.
- ПЗУ программируемое пользователем с электрическим стиранием (Electrically Erasable Programmable ROM – EEPROM или E²PROM). Допускается перезапись произвольной ячейки. При этом стирание выполняется автоматически, прозрачно для пользователя. Число циклов перезаписи до 10000..100000 шт. Значительное время хранения (годы .. 10 лет). Однако, блоки EEPROM имеют ограниченный объем (байты...десятки кБ), в связи с чем их почти всегда используют как память данных.
- ПЗУ с электрическим стиранием типа FLASH является модификацией EEPROM со значительно увеличенным объемом. Для увеличения объема удалены схемы стирания каждого бита по отдельности и стирание выполняется страницами размером от десятков байт до десятков кБ. Также доступно стирание блоками по несколько страниц или всей памяти разом. Такой режим работы (страничное стирание) неудобен для хранения данных, но приемлем для записи программ. Поэтому память FLASH используется в качестве памяти программ. Объем

встраиваемой FLASH-памяти – десятки-сотни кБ. Число циклов перепрограммирования – до 100000. Время хранения – до 10 лет. Время стирания – десятки ms на килобайт, время программирования – десятки мкс на байт. Напряжение питания от 1.8 В. ПЗУ типа FLASH в настоящее время выходит на ведущие позиции в секторе встраиваемых и внешних модулей (микросхем) памяти ПЗУ.

В качестве встроенного ОЗУ в большинстве случаев используются модули статической памяти (SRAM). Выбор данного типа ОЗУ, а не динамического, определяется возможностью хранения данных при снижении частоты вплоть до полной остановки процессора. Такой режим используется для энергосбережения, например, при питании от батарейки.

На современном этапе модули встроенного ОЗУ не обладают значительным объемом – единицы иногда десятки килобайт. В случае мощных систем требующих больших объемов ОЗУ подключаются внешние микросхемы памяти. Это могут быть как микросхемы статического ОЗУ, так и динамическое ОЗУ (DRAM, SDRAM). В последнем случае процессор (например, AMD Am186ED, Mitsubishi M16C и др.) имеет модуль интерфейса динамического ОЗУ, который поддерживает интерфейс классических микросхем DRAM, или SDRAM, а так же регенерацию динамической памяти.

Второй важной особенностью современных модулей ОЗУ – низкое, примерно 1 В, напряжение хранения информации. Это позволяет сохранять данные при провалах питания. С этой же целью иногда в модуль статического ОЗУ включают батарейку электропитания, позволяющую хранить данные до 10 лет (семейство DS5000 Dallas Semiconductor).

2.5 Периферийные модули процессоров для встраиваемых применений.

2.5.1 Порты ввода-вывода.

Каждый процессор для встраиваемых применений имеет некоторое количество внешних линий ввода-вывода, подключенных к внешним выводам микросхемы и называемых внешними портами. Одиночные (одноразрядные, состоящие из одной линии) порты ввода-вывода объединяются в группы, обычно, по 4, 8 или 16 линий, которые называются параллельными портами. Разрядность параллельных портов может быть нестандартной, например, 5-разрядный порт у микроконтроллера PIC16F84.

Через порты процессорное ядро взаимодействует с различными внешними устройствами – считывает значения входных сигналов и устанавливает значения выходных сигналов. Во встраиваемых системах в качестве внешних устройств чаще всего рассматриваются датчики, исполнительные устройства, устройства ввода-вывода данных оператором, устройства внешней памяти.

По типу сигнала различают порты:

1. Дискретные (цифровые) – используются для ввода-вывода дискретных значений логического «0» или «1».
2. Аналоговые – через них вводятся сигналы на вход АЦП или других аналоговых схем и выводятся выходные сигналы ЦАП или других аналоговых схем.
3. Перестраиваемые – настраиваются на аналоговый или цифровой режим работы.

По направлению передачи сигнала различают:

1. Однонаправленные порты, предназначенные только для ввода (входные порты, порты ввода) или только для вывода (выходные порты, порты вывода).

2. Двухнаправленные порты, направление передачи которых определяется в процессе программно-управляемой настройки схемы.
3. Порты с альтернативной функцией. Отдельные линии этих портов связаны со встроенными периферийными устройствами, такими, как таймер, контроллеры последовательных приемопередатчиков. Если соответствующий периферийный модуль не задействован, то линии можно использовать как обычные порты, если модуль активизирован, то связанные с ним линии автоматически или «вручную» (программно) конфигурируются в соответствии с функциональным назначением и не могут быть использованы в качестве универсальных портов ввода-вывода. В некоторых случаях порты могут использоваться только для связи с периферийным модулем (например, входы АЦП в некоторых процессорах).

По алгоритму обмена различают порты:

1. С программно-управляемым (программным) вводом-выводом – установка и считывание данных определяется только ходом вычислительного процесса. Нет защиты от повторного считывания-записи одного и того же (не изменившегося) значения на выводе и считывания-записи во время переходного процесса на выводе.
2. Со стробированием – каждая операция ввода вывода подтверждается импульсом синхронизации (стробом) со стороны источника сигнала (при выводе – процессор, при вводе – внешнее устройство). Считывание информации приемником происходит только по стробу, что позволяет защититься от приема данных во время переходного процесса входного сигнала. Пример: порт PSP (Parallel slave port) в ОКМЭВМ PICmicro.
3. С полным квитированием. Данный режим чаще всего используется для обмена данными с другой вычислительной системой по параллельной шине. Кроме сигналов синхронизации со стороны передатчика используются сигналы подтверждения (готовности к следующему обмену) со стороны приемника. Это позволяет управлять интенсивностью обмена обоим взаимодействующим сторонам и предотвращает потерю данных, когда одна из них перегружена. Пример порта с квитированием, порт LPT персонального компьютера. Во встроенных модулях процессоров данный режим чаще всего реализуется программно-аппаратно.

Аналоговые порты (или перестраиваемые порты в аналоговом режиме) – используются подключения внешних сигналов к ЦАП, АЦП или аналоговым компараторам, встроенным приемопередатчикам. В режиме работы с ЦАП, АЦП или компаратором порты обычно позволяют вводить сигнал в диапазоне от $0V^-$ до $U_{пит}^+$ (индексы + и – означают чуть больше и чуть меньше, примерно на 200..300мВ). В режиме приемопередатчика параметры сигналов определяются конкретным интерфейсом. (В большинстве случаев аналоговые или цифровые линии подключения к приемопередатчикам вообще не называют портами, хотя они по схемотехнике и по месту в структуре процессора близки к универсальным портам ввода-вывода). Реализация входных и выходных каскадов зависит от схемы АЦП, компаратора, ЦАП или приемопередатчика.

Дискретные порты ввода-вывода.

В большинстве современных процессоров для встраиваемых применений поддерживается как независимое управление каждой линией параллельного порта, так и групповое управление всеми разрядами. Так как схемотехника отдельных линий в рамках одного 4-х, 8-ми или 16-разрядного порта одинакова, то дальше будет рассматриваться устройство и функционирование одиночного разряда.

Однонаправленные порты.

Порты ввода.

Схема однонаправленного порта ввода представлена на рисунке.

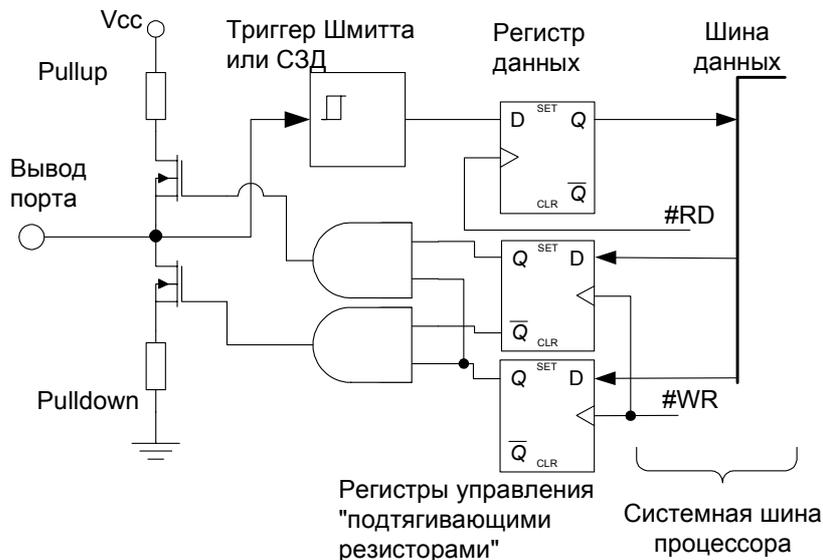


Рисунок 2 Однонаправленный порт ввода

Внешние данные считываются через вывод порта (ножку микросхемы), проходят через триггер Шмитта (ТШ) или схему защиты от дребезга (СЗД) и по внутреннему сигналу чтения фиксируются в регистре данных, с выхода которого, в свою очередь, данные считываются процессором.

ТШ (используется в большинстве процессоров для встроенного применения) имеет гистерезис по уровню входного напряжения и предотвращает многократное переключение входных схем при пологом фронте сигнала или помехах.

СЗД (например, в семействе Zilog Z8) вводит инерционность переключения и отсекает реакцию на короткие по длительности импульсы. Используется для защиты от помех. К входу также могут подключаться так называемые «резисторы поддержки» логической «1» (Pullup) или логического «0» (Pulldown). Эти резисторы предназначены для перевода входов в устойчивое состояние «0» или «1» и предотвращения произвольных переключений от помех в моменты, когда на них (входы) не подается внешний сигнал, например, неиспользуемых и неподключенных к внешним схемам входов («открытых входов»). Через специальные управляющие регистры «схемы поддержки» могут быть отключены полностью или включены в режим Pullup или Pulldown.

Все перечисленные блоки – триггер Шмитта, СЗД и «схемы поддержки» используются для защиты от случайных переключений в результате помех и помогают снизить энергопотребление, которое резко возрастает в момент переключений входных схем.

Порты вывода бывают:

- с двухтактной выходной схемой (комплементарные);
- с одноктактной выходной схемой и внутренней нагрузкой;
- с открытым выходом (открытым коллектором или стоком).

Порты вывода с двухтактной выходной схемой (см. рис 3) являются самыми распространенными и реализованы, например, в семействах Atmel AVR, Microchip PICmicro, AMD AM186, Motorola HC08, HC11 и многих других.

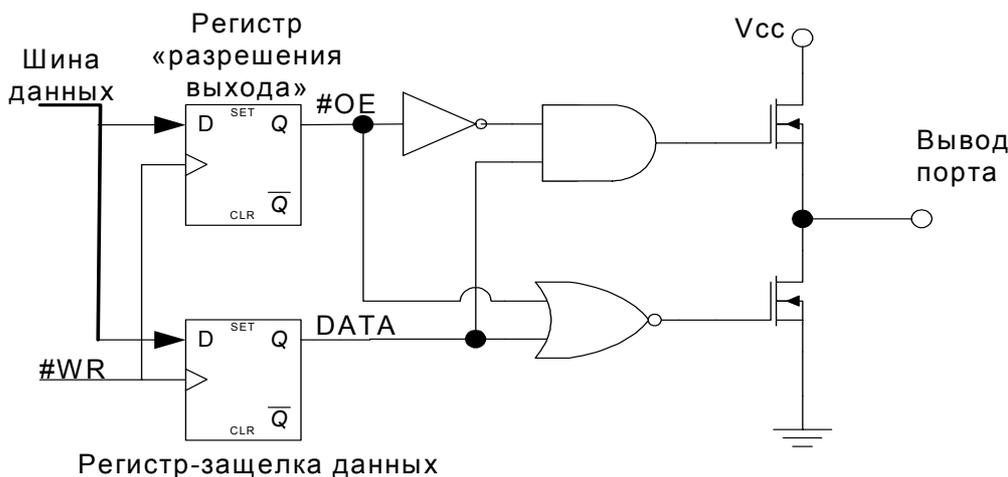


Рисунок 3 Порт вывода с двухтактной схемой

Рассмотрим функционирование данной схемы.

Выходные данные записываются в регистр-защелку данных по внутреннему сигналу записи #WR и через простейшую логическую схему управляют выходными транзисторами. Если в регистр записано значение «1», то открыт верхний по схеме транзистор, а нижний закрыт: на выходе Vcc, то есть «1». Если в регистр записано значение «0», то открыт нижний по схеме транзистор, а верхний закрыт: выход соединен с минусовой шиной питания, то есть там установлен «0».

Верхний по схеме регистр управляет сигналом #OE - «разрешение выходов». Если в регистр записан «0», то схема работает, как было описано выше. Если записана «1», то оба транзистора закрываются и схема переводится в «высокоомное» состояние (Z-состояние). В этом состоянии выходное сопротивление порта очень высокое и он фактически «оторван» от микропроцессора. Это необходимо:

- Если к выходному порту подключены выходы других схем и необходимо разделять линии передачи данных с этими устройствами. Например: наш процессор используется как периферийный контроллер и его выходной порт подключен к периферийной шине другого процессора (мастера), и к шине также подсоединены еще несколько периферийных контроллеров;
- В схемах двунаправленных портов (см. ниже).

Достоинства:

Значительный максимальный втекающий (в состоянии «0») и вытекающий (в состоянии «1») ток выхода: 2..6mA для каскадов с нормальной нагрузочной способностью (например, Fujitsu MB90) и 5..30mA для каскадов с повышенной нагрузочной способностью (например, PICmicro, AVR). Встречаются отдельные микросхемы со сверхвысокой нагрузочной способностью – до 60..90mA (например, PIC17). Большой выходной ток позволяет непосредственно с ножки, без схем усиления и согласования сигнала, управлять достаточно мощной нагрузкой: светодиодами, реле, мощным электронным ключом (транзистор, тиристор). Это значительно упрощает схему устройств.

Недостатки:

- При программировании необходимо управлять дополнительным битовым регистром «разрешение выходов»;
- Значительное энергопотребление и уровень помех при переключении. Последний особо зависит от скорости переключения. Для ограничения токов в момент переключений иногда используют специальные демпфирующие схемы. Однако они снижают быстродействие портов. Наибольшее применение демпфирующие схемы находят в портах ПЛИС в силу их особо высокого быстродействия;

- Относительно сложная внутренняя схема повышающая сложность и стоимость микросхемы в целом. Однако на нынешнем этапе, в связи с успехами технологии производства микросхем, это уже не является проблемой.

Порты вывода с одноконтной выходной схемой и внутренней нагрузкой применяются, например, в семействе MCS-51. Они имеют более простую внутреннюю схему.

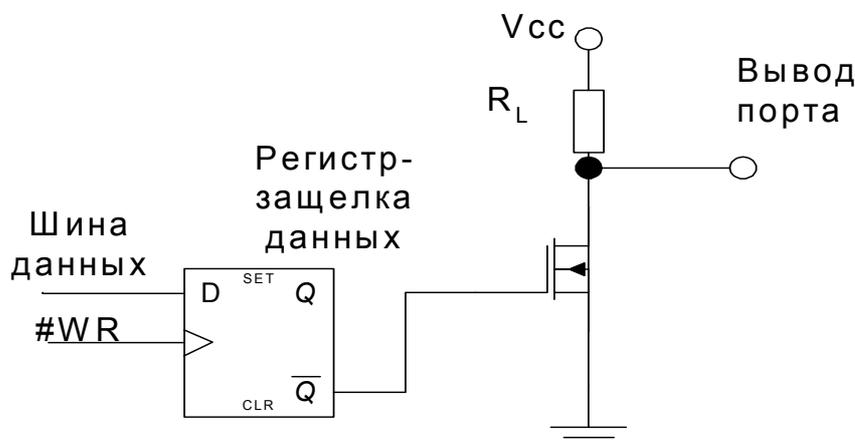


Рисунок 4 Порт вывода с одноконтной схемой

Когда в регистр-защелку записано значение «1», транзистор закрыт и на выходе через резистор R_L устанавливается V_{cc} – логическая «1». Когда же в регистр-защелку записан «0», открывается транзистор и соединяет выход с минусовой шиной питания, то есть там устанавливается «0». При этом резистор R_L оказывается подключенным между шинами питания. Во избежания высокого тока через резистор и его перегрева, сопротивление делают достаточно высоким – 10..100кОм. Высокое сопротивление резистора позволяет непосредственно соединять несколько выходов, не опасаясь их встречного включения, так как если «0» на одном из выходов «подсадит» «1» на другом, то мощность, выделяемая на «подсаженном» резисторе будет мала, он не перегреется и каскад не выйдет из строя.

Достоинства:

- Необходимо управлять только одним регистром;
- Простая схема;
- Возможность без дополнительных схем организовать подключение на одну внешнюю шину несколько таких выходов. Легко построить квазидвунаправленный порт ввода-вывода (см. ниже).

Недостатки:

- Малый вытекающий ток (в состоянии «1»), ограниченный резистором R_L – сотни μA . Это не дает управлять относительно мощными нагрузками без дополнительных каскадов усиления либо требует обеспечивать, чтобы активным был сигнал со значением «0» («управление нулем»).

Порты вывода с открытым выходом (открытым коллектором или стоком).

Применяются во многих семействах микропроцессоров, например, AMD Am186 (там это один из режимов порта), PIC18c40. Выходной каскад построен по одноконтной схеме с внешней нагрузкой. Принцип функционирования аналогичен описанному для одноконтного выходного каскада.

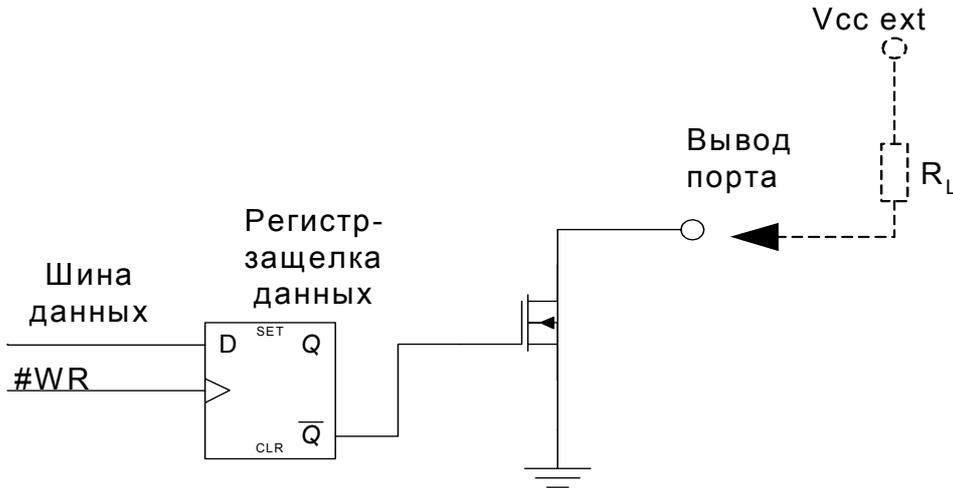


Рисунок 5 Порт вывода с открытым выходом

Достоинства:

- Внешнее напряжение питания нагрузки $V_{cc\ ext}$ может быть иным – большим или меньшим, чем питание микропроцессора. Это может быть удобным для сопряжения схем с различными уровнями логической «1», например, 3.3В и 5В. Если внешнее напряжение достаточно высокое, то можно непосредственно управлять высоковольтной нагрузкой. Например, анонсирован микроконтроллер семейства PICmicro допускающий подключение внешнего напряжения $V_{cc\ ext}$ до 15В при питании ядра 2..6В.
- Необходимо управлять только одним регистром;
- Простая схема;
- Возможность без дополнительных схем организовать подключение на одну внешнюю шину несколько таких выходов. При этом можно подбирать требуемое сопротивление R_L , например, стандарт I²C требует чтобы сопротивление было 2.2кОм. Легко построить квазидвунаправленный порт ввода-вывода (см. ниже).

Недостатки:

- Требуется внешняя нагрузка;
- Малый вытекающий ток (в состоянии «1»), ограниченный внешним нагрузочным резистором.

Двунаправленные порты и порты с альтернативной функцией.

Самой простой схемой двунаправленного порта является квазидвунаправленный порт со схемой, аналогичной схеме порта вывода с одноканальным выходным каскадом.

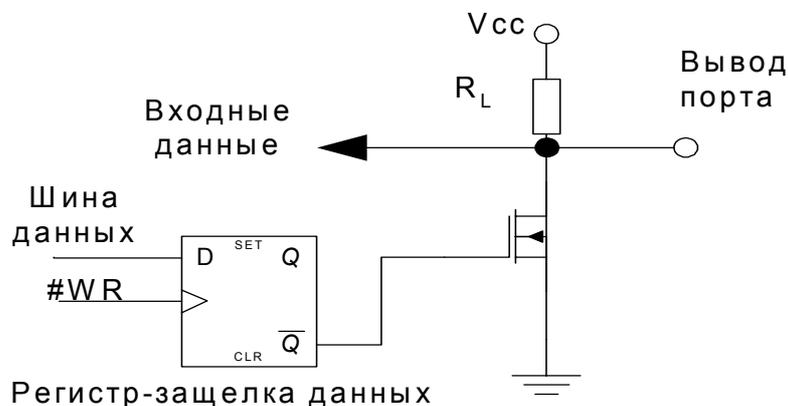


Рисунок 6 Квазидвунаправленный порт

Регистр входных данных (на схеме не показан) подключен к внешнему выводу порта. Перед считыванием входных данных необходимо предварительно записать «1» в регистр-защелку выходных данных. Это закрывает транзистор и исключит влияние порта вывода на входной сигнал. Резистор R_L останется подключенным к входному сигналу и будет являться для него дополнительной нагрузкой, однако, так как сопротивление резистора велико (10..100 кОм), то даже на маломощный входной сигнал данная нагрузка не окажет заметного влияния. Схема квазидвунаправленного порта используется в семействе MCS-51.

Более часто используется схема переключаемого двунаправленного порта с комплементарным выходным каскадом.

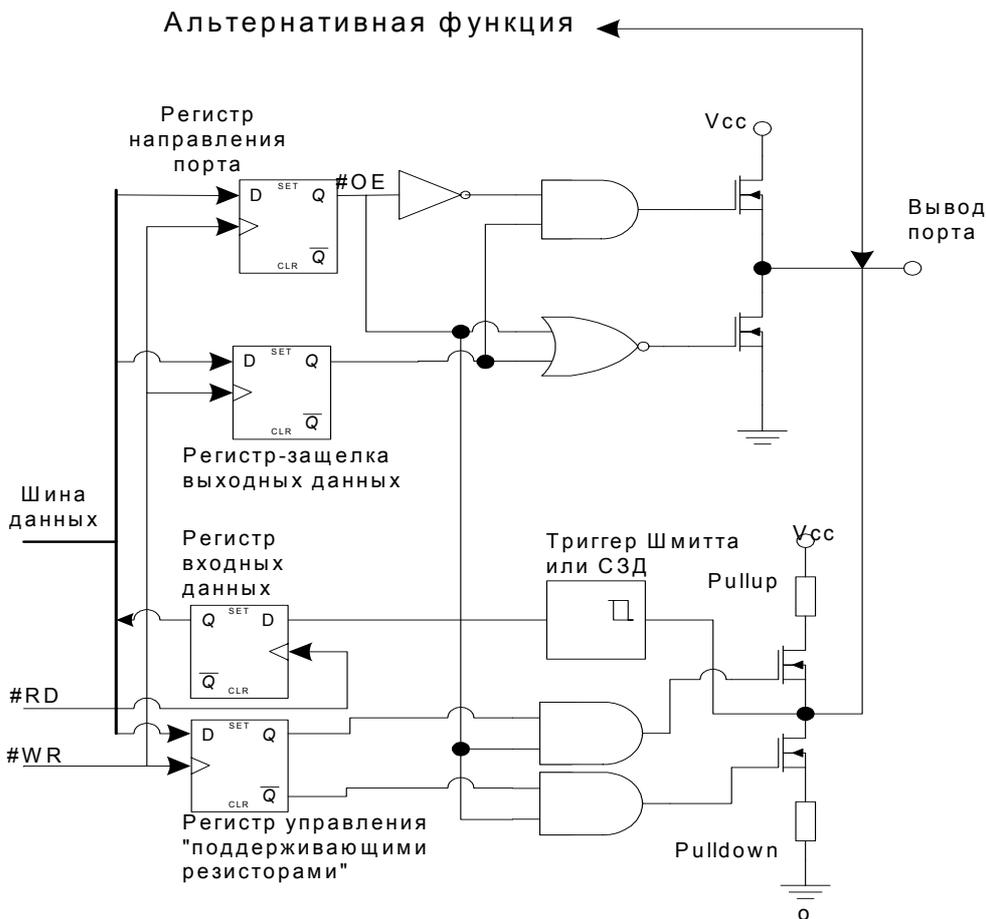


Рисунок 7 Переключаемый двунаправленный порт с комплементарным выходным каскадом

Она объединяет схемы порта ввода и порта вывода с двухтактной выходной схемой, описанные выше. Переключение порта в режим ввода осуществляется записью «1» в регистр «вход/выход». В этом случае (как было указано при описании порта вывода) оба транзистора переводятся в закрытое состояние и порт вывода не влияет на входной сигнал. В двунаправленных портах резисторы pullup и pulldown подключаются только в режиме ввода, для чего на вход соответствующей схемы управления подключается выход регистра «вход/выход» («1» - ввод).

Кроме исполнения функции порта ввода-вывода, внешние выводы микросхемы могут быть задействованы для связи с внутренними периферийными модулями микропроцессора, а так же с подсистемами процессорного ядра, схем памяти и управления (с контроллером прерываний, блоком интерфейса внешней памяти и т.п.). Данные функции называются альтернативными. Обычно, когда вывод порта используется для выполнения альтернативной функции основные схемы переводятся в состояние ввода или вообще отключаются.

2.5.2 Таймеры-счетчики.

Таймеры счетчики предназначены для:

- Подсчета временных интервалов (режим тайера);
- Подсчета числа импульсов («внешних событий») на специальном внешнем входе (режим счетчика).

Структурная схема таймера-счетчика представлена на рисунке 7.

Режим таймера.

Тактирование счетчика выполняется от сигнала внутренней синхронизации процессора Fint. Обычно это частота процессорных циклов формируемая от основного генератора.

Подсчет временных интервалов выполняется в периодах сигнала Fint.

Предделитель используется для снижения тактовой частоты, подаваемой на регистр-счетчик. Это позволяет подсчитывать в более длительные интервалы, но увеличивает шаг дискретизации, а соответственно уменьшает точность. Предделитель может быть с фиксированным или программируемым коэффициентом деления. У программируемых предделителей обычно выбирается коэффициент деления из ряда 1, 2, 4, 8, ...

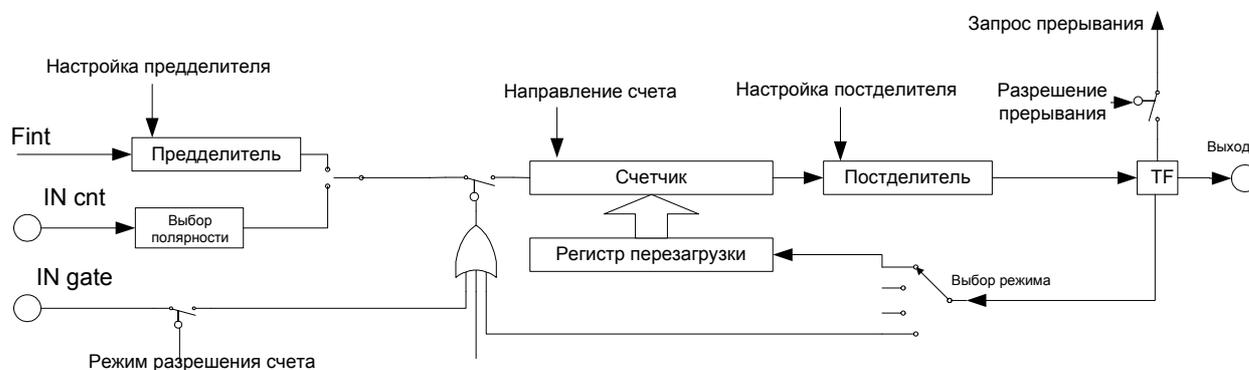


Рисунок 8 Таймер-счетчик

Регистр-счетчик накапливает (считает) значение временного интервала в единицах входных тактов счетчика (после предделителя). Разрядность регистра-счетчика определяет разрядность всего таймера-счетчика.

Постделитель встречается достаточно редко (PICmicro) и служит для увеличения периода установки флага переполнения TF. Обычно постделитель – это дополнительные разряды регистра-счетчика недоступные по чтению-записи. Постделитель обычно программируемый на разные коэффициенты деления как и предделитель.

TF – флаг переполнения таймера. Устанавливается при переходе всех разрядов регистра счетчика-постделителя из 1 в 0. Обычно используется для указания окончания временного интервала. По нему может вырабатываться запрос прерывания.

От флага TF идет цепь обратной связи, задающая режим работы таймера:

- a) однократный счет: после переполнения в регистр-счетчик загружается значение 0 и счет останавливается. Запуск следующего цикла – специальной командой из программы;
- b) циклический счет с полным циклом: после переполнения в регистр-счетчик загружается значение 0 и счет начинается снова. Полный цикл счета таймера будет 2^k тактов, где k – разрядность счетчик + постделитель.
- c) циклический счет с автоперезагрузкой: после переполнения в регистр счетчик загружается значение из регистра перезагрузки. Таким образом счет можно начинать не с 0 и уменьшается (программируется) длительность цикла таймера.

Во многих процессорах имеется специальный вывод INgate, который выполняет функцию разрешения счета внешним сигналом. С помощью этого механизма легко подсчитывать длительность временного интервала, определяемого длительностью импульса на входе INgate.

Режим счетчика.

В отличие от режима таймера, в режиме счетчика выбирается тактирование от внешнего импульсного сигнала, подаваемого на вход INcnt. При этом подсчитываются импульсы внешнего сигнала. Инкрементация или декрементация счетчика происходит по перепаду (фронту) сигнала. Фронт сигнала в данном случае называют «внешним событием». Полярность фронтов можно программировать. В остальном, функционирование в режимах счетчика и таймера аналогично.

2.5.3 Модули таймеров-счетчиков со схемами входного захвата, выходного сравнения и выработки сигналов с ШИМ (CAPTURE/COMPARE/PWM (CCP)).

Модули CCP являются развитием структуры таймеров-счетчиков и выполняют схожие функции, однако требуют меньшей программной поддержки, более гибки в настройке на различные задачи, позволяют достигнуть более высокого быстродействия. Наибольшую эффективность они обеспечивают при работе с внешними периодическими или непериодическими сигналами при решении следующих задач:

- фиксация времени (момента) внешнего события (фронта);
- определение частоты и длительности импульсов внешнего сигнала, фазового сдвига нескольких сигналов;
- формирование одиночных импульсов с программируемой длительностью.
- формирование на одном или нескольких выводах периодических последовательностей импульсов и программируемой частотой, длительностью, фазовым сдвигом (в случае нескольких выходных сигналов);
- формирование сигналов с широтно-импульсной модуляцией (ШИМ, PWM). При ШИМ частота сигнала остается постоянной, а длительность положительного и отрицательного импульсов программируется. Основная характеристика сигнала с ШИМ является скважность: отношение периода к длительности положительного импульса. Для меандра скважность равна 2. Модуль ШИМ с подключенной к его выходу интегрирующей цепочкой образует простейший ЦАП. Такое использование модулей ШИМ является основным во встраиваемых системах.

Все перечисленные функции выполняются модулями CCP автономно, а вмешательство программиста требуется только на этапе настройки режимов модуля.

Схема выходного сравнения (Output Compare).

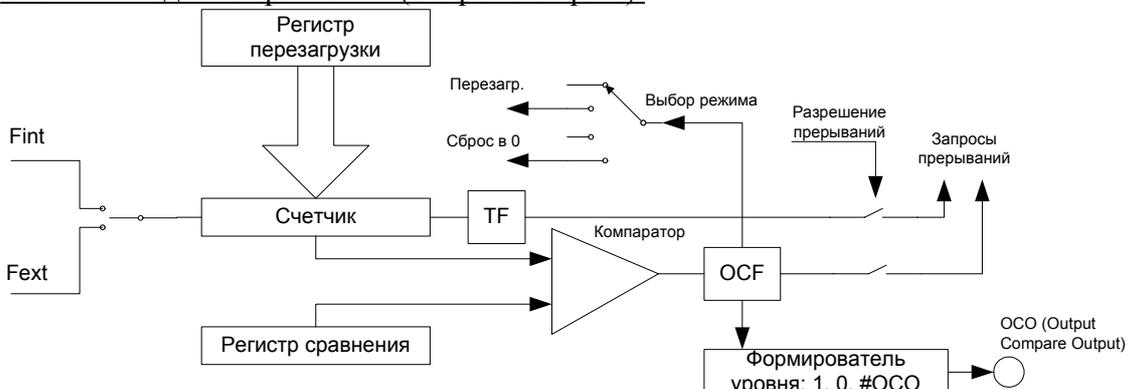


Рисунок 9 Модуль выходного сравнения (Output Compare)

Многоразрядный цифровой компаратор непрерывно сравнивает изменяющийся во времени код таймера-счетчика с кодом, который записан в регистре сравнения. В момент равенства этих кодов устанавливается флаг OCF (Output Compare Flag) и изменяется сигнал на выводе ОСО (Output Compare Output). Возможны три варианта изменения сигнала, которые могут быть настроены программно: установка «1», установка «0», инвертирование сигнала на выводе ОСО ($ОСО \leq \#ОСО$). По установке флага OCF может быть сброшен (обнулен) или перезагружен определенным значением регистр-счетчик. Кроме того, по установке флага OCF может быть выработан запрос прерывания, если данное прерывание разрешено. Запрос прерывания может вырабатываться и при переполнении таймера-счетчика.

Рассмотрим примеры типовых применений модуля CCP в режиме выходного сравнения:

1. Формирование сигнала с определенной частотой: формирователь уровня настраивают на режим инверсии ОСО, управление таймером-счетчиком в режим сброса по флагу OCF, в регистр сравнения – значение, равное полупериоду формируемой частоты. По каждому событию сравнения раз в полупериод порт ОСО инвертируется и формируется передний или задний фронт сигнала.
2. Формирование одиночного импульса определенной длительности: формирователь уровня настраивают на режим установки ОСО в «0», в регистр сравнения – длительность импульса, таймер обнуляем и одновременно устанавливаем порт ОСО в «1» (передний фронт). По событию сравнения порт обнуляется (задний фронт).
3. Ожидание определенного числа импульсов на счетном входе (сигнал Fext) таймера-счетчика: таймер настраиваем в режим счетчика, обнуляем, в регистр сравнения записываем требуемое число импульсов, разрешаем прерывание по событию сравнения (по флагу OCF). После прохождения заданного числа импульсов будет выработан запрос прерывания.
4. Делитель входной частоты на заданное число N, кратное двум: таймер-счетчик переключаем в режим счетчика, устанавливаем обнуление счетчика по флагу OCF, формирователь уровня настраивают на режим инверсии ОСО, в регистр сравнения записываем значение N/2.

Схема входного захвата (Input Capture).

Функцию входного захвата поддерживают микроконтроллеры семейств (Atmel), 8051GB(Intel), AVR(Atmel), PIC16(Microchip), ST7, ST9 (SGS-T), HC08, HC11 (Motorola) и многие другие.

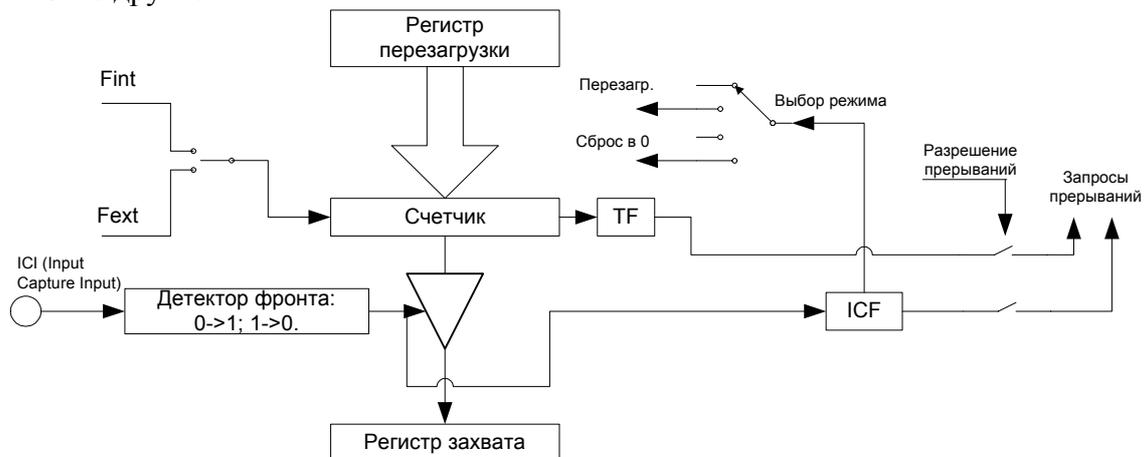


Рисунок 10 Модуль входного захвата (Input Capture)

Данная схема предназначена для фиксации времени возникновения внешнего события: когда на внешнем выводе ICI происходит событие (перепад), определяемый настройкой схемы «детектора фронта», то текущее значение регистра-счетчика переписывается в

регистр захвата, откуда может быть прочитано программно. Во многих реализациях захват может быть программно-управляемым – по команде обращения к специальному регистру.

Тактирование регистра-счетчика чаще выбирается от сигнала внутренней синхронизации процессора F_{int} , то есть счетная часть модуля Input Capture настроена на режим подсчета времени – таймера. Но так же можно использовать и внешнее тактирование. По событию захвата устанавливается флаг ICF, может вырабатываться запрос прерывания. Кроме этого может быть перезагружен «0» или определенным значением регистр-счетчик.

С помощью схемы входного захвата удобно:

1. Определять период/частоту сигнала на входе ICI;
2. Фиксация относительного времени возникновения различных событий.

Схема выработки сигнала с ШИМ.

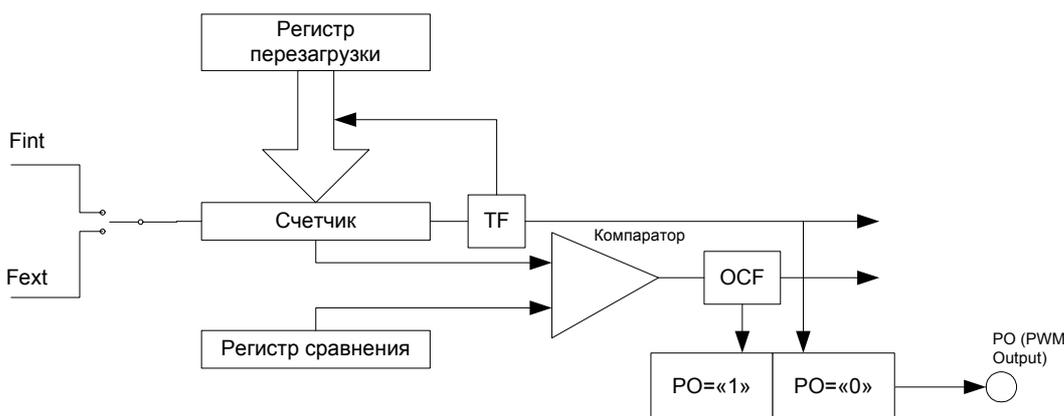


Рисунок 11 Модуль генератора ШИМ-сигнала

Данная схема является модифицированным вариантом схемы выходного сравнения (Output Compare). Разница в том, что выходом управляет как компаратор, так и схема фиксации переполнения регистра-счетчика. Передний фронт сигнала с ШИМ ($0 \rightarrow 1$) формируется по событию сравнения (когда регистр-счетчик равен регистру сравнения). Задний фронт ($1 \rightarrow 0$) – по переполнению регистра-счетчика.

Период сигнала с ШИМ равен частоте переполнения таймера и задается содержимым регистра перезагрузки. Длительность положительного импульса в периоде определяется как разница (максимального значения регистра-счетчика +1) и содержимого регистра сравнения.

В различных процессорах могут использоваться схемы генераторов ШИМ немного отличающиеся от данной.

Процессоры событий.

Под управлением единого счетчика могут быть объединены несколько каналов входного захвата/ выходного сравнения/ формирования сигналов ШИМ. Каждый из каналов может быть индивидуально настроен на один из перечисленных режимов. Такие сложные блоки называют *процессорами событий*, а также: массивом программируемых счетчиков - PCA (Programmable Counter Array) (Intel); блоком CAPCOM (Infineon), блоком TIM8 (Motorola).



Рисунок 12 Блок процессора событий

Процессоры событий позволяют формировать взаимно синхронизированные выходные сигналы - с фиксированным сдвигом фаз или считывать временные сдвиги между событиями, как частный случай – сдвиг фазы.

2.5.4 Аналого-цифровой преобразователь (АЦП).

Модуль аналого-цифрового преобразования (АЦП) предназначен для ввода в процессор аналоговых сигналов с датчиков физических величин и преобразования значения напряжения этих сигналов в двоичный код с целью дальнейшей программной обработки.



Рисунок 13 Модуль аналого-цифрового преобразования (АЦП) (схема приводилась на лекциях)

АЦП выполняет преобразование значения аналогового напряжения на входе в цифровой код. Полученное (преобразованное) значение записывается в регистр данных (РД). АЦП, интегрированные на кристалл процессора, обычно строят по схеме последовательного приближения. Время преобразования обычно составляет несколько десятков микросекунд, в зависимости от частоты тактирования АЦП. Завершение процесса преобразования отмечается установкой флага Фацп и (если разрешено) вырабатывается запрос прерывания. В современных управляющих процессорах и микроконтроллерах наиболее распространены АЦП с разрядностью 8, 10, реже 12 и совсем редко 14 и 16 бит.

Аналоговый коммутатор – выбирает один из возможных аналоговых входов (выводов) и подключает его к входу внутреннего АЦП для преобразования. При последовательной выборке каналов создается имитация многоканального АЦП. Применение действительно многоканальных АЦП резко повышает энергопотребление и стоимость процессора и обычно не используется (Если требуется несколько каналов и высокая скорость преобразования, то используют микросхему внешнего АЦП).

Код выбора канала может формироваться программно, то есть программист «вручную» переключается между каналами, или аппаратно (автоматически), последовательно перебирая каналы (режим сканирования).

Для большего удобства использования модуля АЦП в режиме сканирования могут быть реализованы несколько регистров данных (Fujitsu MB90, Intel 8051GB), по одному на канал. Программисту будет достаточно считывать данные из регистра, соответствующего требуемому каналу. При этом код выбора канала параллельно подается на адресные входы блока регистров данных.

Источник опорного напряжения V_{ref} и коммутатор V_{ref} . Опорное напряжение V_{ref} определяет диапазон значений напряжения на аналоговых входах и разрешающую способность АЦП, равную $V_{ref}/2^n$, где n – разрядность АЦП. Если значение напряжение на входе не велико, то точность преобразования может быть увеличена путем уменьшения V_{ref} . Диапазон допустимых значений V_{ref} обычно находится в рамках значения напряжения питания процессора.

Могут быть использованы опорные источники следующего типа:

1. внешний, подключаемые через специальные выводы микросхемы;
2. внутренний фиксированный или программируемый (с помощью встроенного ЦАП).

Подключение к АЦП внешнего или внутреннего источников выполняется с помощью коммутатора V_{ref} .

Коммутатор сигнала запуска АЦП – позволяет выбрать способ запуска процесса преобразования, а также определяет один из возможных режимов работы АЦП:

1. Периодического преобразования. В этом режиме АЦП запускается периодическим сигналом $f_{зап.вн.}$ от основного тактового генератора или встроенного таймера. Если сигнал запуска подать на двоичный счетчик, выходами подключенный к управляющим входам аналогового коммутатора и адресным линиям блока регистров данных, то таким образом легко реализовать режим последовательного сканирования каналов.
2. Внешнего запуска. Запуск осуществляется внешним сигналом $f_{зап.внеш.}$, что позволяет четко определить момент считывания значения аналогового напряжения со входа.
3. Программно управляемого запуска, по установке специального бита $F_{зап.прг.}$.

Блок управления модулем АЦП – конфигурирует и синхронизирует функционирование других (вышеперечисленных) блоков, управляется программно, через регистры специального назначения.

Аналоговый компаратор.

Аналоговый компаратор используется для сравнения напряжения двух внешних аналоговых сигналов или для сравнения напряжения внешнего аналогового сигнала с образцовым напряжением, вырабатываемым внутри процессора. Могут быть запрограммированы различные уровни образцового напряжения. Результат сравнения кодируется битом в регистре специального назначения, например, “1” – вход А больше или равно чем В, “0” – вход А меньше чем В. В случае изменения соотношения изменяется значение бита, а также может быть установлен флаг и выработан запрос прерывания.

Структура блока аналогового компаратора приведена на ниже.

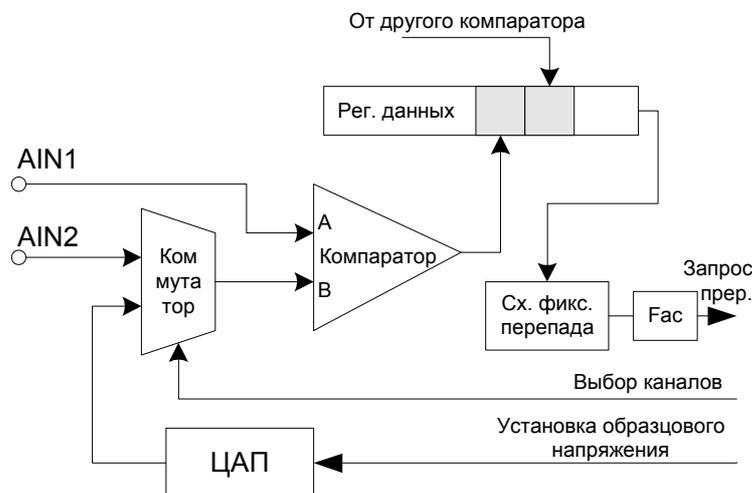


Рисунок 14 Модуль аналогового компаратора

Аналоговый коммутатор входов – выбирает аналоговые сигналы для сравнения. Один сигнал берется с внешнего входа AIN1, в качестве второго берется или сигнал с внешнего входа AIN2 или образцовое внутреннее напряжение, которое вырабатывается с помощью ЦАП.

Компаратор – сравнивает аналоговые сигналы.

ЦАП – программируемый генератор образцового напряжения.

Регистр данных – программно доступный регистр, в битах которого сохраняются результаты сравнения одного или нескольких компараторов.

Схема фиксации перепада – определяет изменение одного из бит в регистре данных (выхода одного из компараторов) и вырабатывает по этому событию запрос прерывания.

Пример использования аналогового компаратора:

- контроль превышения допустимых значений температуры, давления, тока, напряжения и других физических величин. Физическая величина преобразуется в напряжение с помощью датчика и контролируется с помощью аналогового компаратора. Порог сравнения устанавливается встроенным генератором образцового напряжения.
- Обнаружение (формирование) фронтов внешних сигналов.
- Встроенные схемы контроля напряжения питания системы.

2.5.5 Цифро-аналоговый преобразователь (ЦАП).

ЦАП предназначен для генерации аналогового сигнала с уровнем напряжения, соответствующим заданному цифровому коду.

Блок-схема ЦАП приведена ниже.

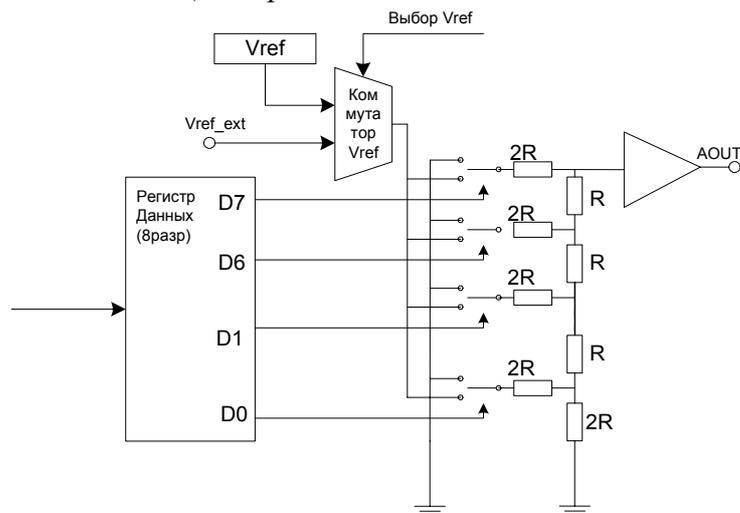


Рисунок 15 Модуль цифро-аналогового преобразователя (ЦАП)

Регистр данных – в него записывается цифровой код. Регистр данных определяет разрядность ЦАП.

Матрица R-2R – самый распространенный метод цифро-аналогового преобразования. Матрица работает по принципу деления входного напряжения на входах. Матрица имеет число входов по числу разрядов регистра данных. На каждый вход через ключ может быть подано опорное напряжение Vref или 0В. Ключи управляются разрядами регистра данных: “1” – на матрицу подается Vref, “0” – подается 0В.

Коммутатор опорного напряжения Vref позволяет выбрать внешний или встроенный источник опорного напряжения.

На практике ЦАП применяется для управления различными исполнительными устройствами (приводами) и системами: электродвигателями постоянного тока с

переменной скоростью вращения, источниками питания с управляемым напряжением, различными индикаторами и т.п. С помощью ЦАП можно синтезировать аналоговые сигналы различной формы, например, синусоидальной.

2.5.6 Контроллеры последовательных интерфейсов.

Контроллеры последовательных интерфейсов ориентированы на решение следующих задач:

- связь встраиваемой микропроцессорной системы с системой управления верхнего уровня: промышленным или офисным компьютером, программируемым контроллером. Наиболее часто для этих целей используют интерфейсы RS232C, RS422, USB, IrDA.
- Связь с внешними по отношению к микропроцессору периферийными микросхемами (памяти EEPROM, часов реального времени (RTC) и т.д.), а также с различными датчиками с последовательным цифровым выходом. Для этих целей наиболее часто применяются интерфейсы SPI, I²C, MicroWire, uLAN и другие.
- Интерфейс связи с локальной сетью в распределенных информационно-управляющих системах. В этой сфере находят применение интерфейсы RS232C, RS485, I²C, uLAN, CAN, Ethernet.
- Внутрисистемное программирование резидентной памяти программ (OTPROM, EPROM, FLASH) или данных (EEPROM) у процессоров для встраиваемых применений. Обычно для этого используется интерфейс RS232C (ADuC (Analog Devices), MB90Fxxx (Fujitsu), MSP430 (Texas Instruments)) или SPI (AVR(Atmel)).

В настоящее время встроенные контроллеры последовательных интерфейсов имеются почти у всех встраиваемых процессоров, исключая простейшие 8-16 выводные микросхемы. У большинства процессоров имеются несколько таких модулей одного или различных типов.

Среди контроллеров последовательного обмена стандартом «де-факто» стал модуль универсального синхронно-асинхронного приемо-передатчика (USART (Universal Synchronous/Asynchronous Receiver and Transmitter)). В названии часто опускают слово «синхронный» и модуль не совсем корректно именуется UART (чисто асинхронные приемо-передатчики сейчас встречаются достаточно редко). Модули UART в асинхронном режиме поддерживают протокол обмена для интерфейсов RS232C (8N1 или 9N1); в синхронном режиме – нестандартные синхронные протоколы, в некоторых случаях - протокол SPI. Упрощенная структура приемопередатчика типа UART представлена на рисунке.

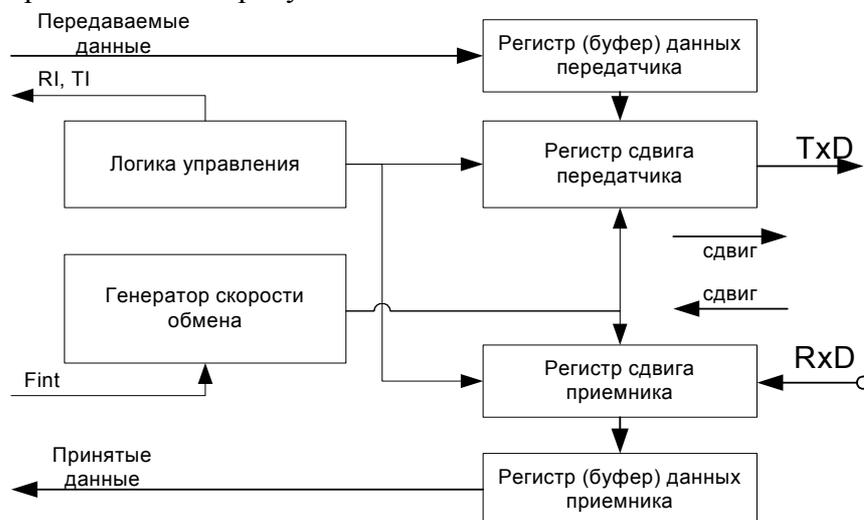


Рисунок 16 Модуль UART

Генератор скорости обмена представляет собой делитель внутренней тактовой частоты процессора f_{int} с плавно или пошагово (дискретно) программируемым коэффициентом деления. При «плавном» программировании можно настраивать требуемую скорость вне зависимости (в определенных пределах) от частоты f_{int} . Для этого используется стандартный или специально выделенный таймер-счетчик в режиме автоперезагрузки. В случае «фиксированных» коэффициентов деления для поддержания стандартного ряда скоростей необходимо выбирать определенную частоту тактирования процессора. С выхода генератора скорости сигнал синхронизации поступает на вход тактирования приемного и передающего сдвиговых регистров, которые осуществляют последовательную выдачу/прием бит данных с заданной скоростью. Полностью принятый байт попадает в регистр – буфер данных приемника. Байт для передачи помещается в сдвиговый регистр из буфера передатчика.

Процессы приема и передачи в асинхронном режиме UART происходят независимо. Таким образом, поддерживается дуплексный режим обмена. Однако требуется, чтобы приемник и передатчик были настроены на одну скорость.

Более простым является функционирование в синхронном режиме. Здесь каждый принимаемый/передаваемый бит стробируется специальным сигналом и нет необходимости точно согласовывать скорость приемника и передатчика.

Еще более упрощается функционирование в режиме SPI: приемник и передатчик работают синхронно: приему одного бита соответствует передача одного бита, начало передачи байта совпадает с началом приема, за сеанс обмена происходит прием одного байта и передача одного байта.

В большинстве случаев приемопередатчики работают с входными и выходными сигналами уровней TTL. Формирование физических сигналов с уровнями напряжения и тока, соответствующими реализуемому интерфейсу выполняется с помощью специальных микросхем – трансиверов или адаптеров физического интерфейса. Например: MAX232 (MAXIM) – RS232C, MAX485 (MAXIM) – RS422/485, PCA82C251 (Philips) – CAN.

Кроме рассмотренных приемопередатчиков USART во встраиваемых процессорах широко используются другие интерфейсы, например, USB, CAN. Они значительно сложнее и рассматриваются в специальных учебных курсах («Системы ввода-вывода», «Распределенные системы управления») и в технической литературе.

2.6 Организация прерываний в управляющих процессорах.

Источниками прерываний могут быть:

1. Внешние источники. Запрос передается перепадом напряжения на входе (из «1» в «0» или из «0» в «1») или определенным уровнем напряжения («0» или «1») на внешнем входе запроса прерывания.
2. Внутренние источники – встроенные модули памяти (обычно от модули EEPROM) или модули периферийных устройств:
 - a. Таймеры/счетчики. Запрос вырабатывается по переполнению;
 - b. Блоки захвата/сравнения. Запрос по событию входного захвата или равенства при выходном сравнении.
 - c. АЦП. Запрос по завершению преобразования.
 - d. Аналоговые компараторы. Запрос по изменению соотношения уровней входных сигналов.

- е. Приемо-передатчики последовательных интерфейсов (RS232 (SIO), SPI, I2C, USB, CAN, Ethernet, HDLC и т.п.). Запрос вырабатывается:
- i. По приему байта или пакета и доступности новых принятых данных;
 - ii. По завершению передачи байта или пакета и освобождению передатчика.

3. Программные прерывания.

Организация прерываний в процессорах для управляющих систем ничем принципиально не отличается от универсальных процессоров. В различных семействах управляющих процессоров реализованы различные механизмы обработки прерываний:

1. Векторный с жестким приоритетом (ST7, AVR, Am186);
2. Векторный с программируемым приоритетом (MCS-51, M16C, i386EX);
3. Векторный с динамической таблицей векторов (M16C);
4. С общим вектором (механизм поллинга) (PIC).

Укрупненная схема блока обработки запросов прерываний.

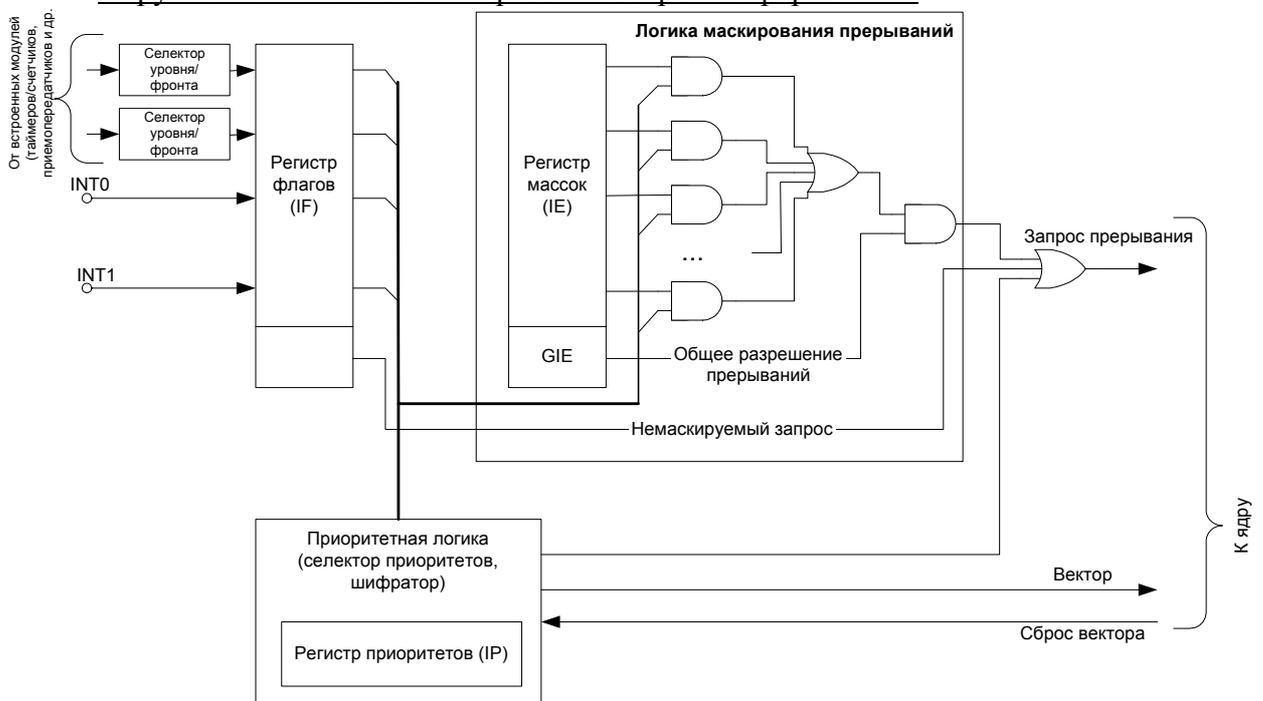


Рисунок 17 Блок обработки запросов прерываний

Селектор уровня/фронта внешнего сигнала запроса прерывания – выбирает событие, по которому вырабатывается запрос прерывания от внешнего сигнала («внешнее прерывание»). Возможны следующие настройки:

- по перепаду «1» → «0»;
- по перепаду «0» → «1»;
- по уровню «1»;
- по уровню «0».

Регистр флагов (Interrupt Flag (IF)) – при возникновении запроса прерывания в нем устанавливается бит, соответствующий источнику.

Логика маскирования (разрешения) прерываний – разрешает или запрещает выработку запросов от определенных источников или от всех источников сразу. Для разрешения прерывания необходимо установить в «1» соответствующий бит регистра масок (Interrupt

Mask (IM)) и бит общего разрешения прерываний (Global Interrupt Enable (GIE)). Логика маскирования никак не влияет на немаскируемый запрос прерывания (NMI).

Приоритетная логика – вырабатывает вектор для наиболее приоритетного запроса и передает его ядру процессора синхронно с сигналом запроса прерывания; отслеживает приоритет запроса, находящегося в обработке и вытесняет (прерывает) данный запрос, если пришел другой запрос с более высоким приоритетом. Приоритет каждого прерывания может задаваться жестко (системы с жесткими приоритетами) или программируется в регистре приоритетов (Interrupt Priority (IP)). Вектор прерывания выбирается по номеру запроса из статической или динамической (программируемой) таблицы векторов.

2.7 Подсистема синхронизации.

Подсистема синхронизации отвечает за формирование устойчивых сигналов синхронизации внутренних блоков процессора и внешних цепей (блоков) управляющих вычислительных систем, построенных на данном процессоре.

К внутренним блокам относятся:

- Вычислительное ядро;
- Периферийные устройства (таймеры/счетчики, блоки ССР, АЦП, ЦАП, приемопередатчики и др.);
- Схемы рестарта («сброса»).

Для синхронизации внешних схем (периферийных контроллеров, интерфейсных микросхем, блоков программируемой логики и др.) из процессора на специальную ножку выводится сигнал синхронизации.

Обобщенная структура подсистемы синхронизации встраиваемых процессоров приведена на рисунке.

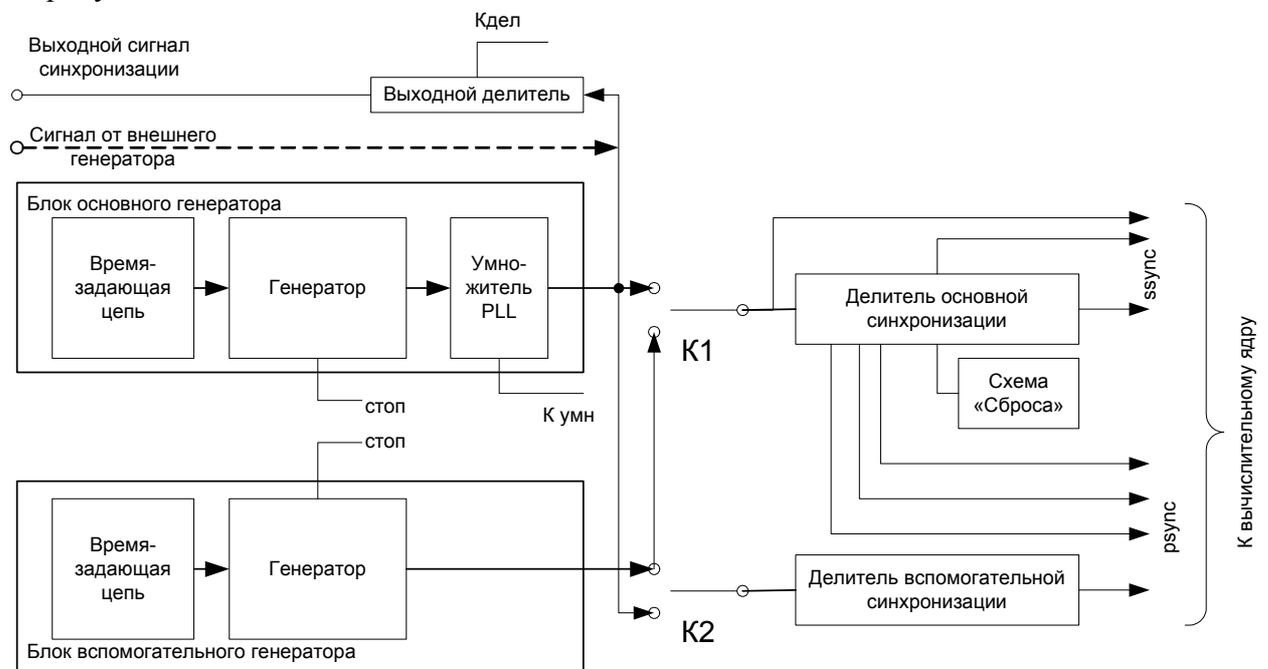


Рисунок 18 Подсистема синхронизации

Подсистема синхронизации включает блок генераторов синхроимпульсов (основной и вспомогательный генераторы, схему формирователя выходного сигнала синхронизации) и

формирователь внутренних синхросигналов процессора (коммутаторы K1 и K2, делители, схему «сброса»).

Блок основного генератора – вырабатывает сигналы синхронизации для вычислительного ядра, большинства периферийных устройств, схемы рестарта. Блок включает собственно схему генератора, внешнюю или встроенную времязадающую цепочку для генератора и (не всегда) схему умножителя частоты.

Основная частота синхронизации может изменяться в очень широких пределах – от десятков килогерц до десятков и сотен мегагерц. Невысокая частота (до 1 МГц) используется в системах с пониженным энергопотреблением.

В зависимости от выбранной рабочей частоты, требований точности и стабильности параметров сигнала синхронизации необходимо могут быть использованы *различные типы времязадающих цепочек*, которые в свою очередь требуют перенастройки режимов работы генераторов. Выбор режима работы генератора и типа времязадающей цепочки осуществляется программированием специальных конфигурационных бит во встроенной памяти программ микропроцессора.

Времязадающие цепочки подключаются к специальным выводам микропроцессора. В некоторых моделях имеются встроенные цепочки. Наиболее часто используются следующие типы времязадающих цепочек:

1. Кварцевый резонатор: частоты от десятков килогерц до десятков мегагерц, высокая стабильность частоты (погрешность – сотые/тысячные доли процента), относительно высокая цена;
2. Пьезокерамический резонатор: частоты от десятков килогерц до единиц мегагерц, средняя стабильность частоты (погрешность – десятые доли процента), невысокая цена;
3. LC-цепь: частоты единицы-сотни килогерц, средняя стабильность частоты (погрешность – десятые доли/единицы процента), невысокая цена;
4. RC-цепь: частоты единицы-сотни килогерц, низкая стабильность частоты (погрешность – единицы процента), низкая цена, часто реализуется как встроенная времязадающая цепочка;

В случаях высокой частоты (свыше 30 МГц) рекомендуется встроенный генератор отключать полностью и подключать внешний. Так же можно поступать, если от одного внешнего генератора синхронизируется несколько схем, включая процессор.

Работа генератора на высокой частоте ведет за собой следующие трудности: сложность «запуска» встроенного генератора (на частотах свыше 30 МГц), специальные требования к трассировке и качеству печатных плат, высокий уровень помех от внешних высокочастотных цепей (например, цепей подключения кварцевого резонатора), невозможность оперативной перестройки частоты. Для избежания этих проблем почти во всех 16/32-разрядных процессорах используются встроенные цифровые умножители частоты с программируемым коэффициентом умножения. Наиболее распространенным умножителем на сегодняшний день является схема синтезатора частоты с фазовой автоподстройкой (PLL).

От основного генератора синхросигнал выводится на ножку микросхемы и может использоваться для тактирования внешних схем. Для снижения частоты выходного сигнала в этой цепи может использоваться управляемый или фиксированный делитель.

Блок вспомогательного генератора – обеспечивает тактирование части периферийных устройств, обычно таймеров-счетчиков (PICmicro, ATmega, Fujitsu MB90), а в некоторых режимах может принимать на себя функции основного генератора (синхронизацию ядра, периферии, схем «сброса»). Вспомогательный генератор обычно работает на частотах до 1

МГц. В случае использования его как базы часов реального времени – на частоте 32768кГц.

По структуре вспомогательный генератор аналогичен основному генератору, но почти никогда не используется умножитель частоты.

Ядром формирователя внутренних сигналов являются делители частоты основного и вспомогательного генераторов. С их выходов берутся сигналы тактирования ядра и схемы сброса **ssync** и сигналы синхронизации периферийных модулей **psync**. Обычно все внутренние сигналы (ssync и psync) получаются делением частоты генераторов на фиксированный коэффициент, но в некоторых процессорах коэффициенты деления могут программироваться, например, если необходимо снизить частоту тактирования ядра в режимах пониженного энергопотребления.

Коммутаторы синхросигналов K1 и K2 используются для выбора источника тактирования внутренних схем процессора: основного или вспомогательного генератора. В обычном режиме большинство подсистем синхронизируется от основного генератора, а вспомогательный используется как временная база таймеров, часов реального времени или сторожевого таймера. В случае нестабильности работы основного генератора или при необходимости перейти на более низкие частоты функционирования, например, в режимах энергосбережения, можно подключить вход делителя основного генератора на выход вспомогательного генератора.

2.8 Механизмы начальной инициализации встроенной памяти.

Механизмы начальной инициализации (начальной загрузки) обеспечивают запись программного кода, данных или конфигурационных параметров во встроенную энергонезависимую память процессора или однокристалльной микроЭВМ. Процесс начальной инициализации предполагает работу с «голой» аппаратурой, т.е. без помощи какой-либо инструментальной программы (загрузчика), исполняющейся в рабочем режиме процессора.

В качестве записываемого программного кода выступает или более высокоуровневый загрузчик или непосредственно прикладная программа.

Данные – обычно начальные значения рабочих параметров (уставок).

Конфигурационные параметры настраивают режимы работы аппаратуры процессора. К ним могут относиться:

- тип генератора (кварцевый, на пьезокерамическом резонаторе, LC или RC);
- используемые подсистемы сброса при сбое электропитания, автоматического сброса при включении питания (Power On Reset);
- использование сторожевого таймера (Watch Dog Timer);
- флаги защиты внутренней памяти от несанкционированного копирования;
- использование и разрядность шины внешней памяти;
- адрес старта программы (вектор сброса);
- и т.д.

Классификация механизмов начальной инициализации представлена на рисунке ниже.

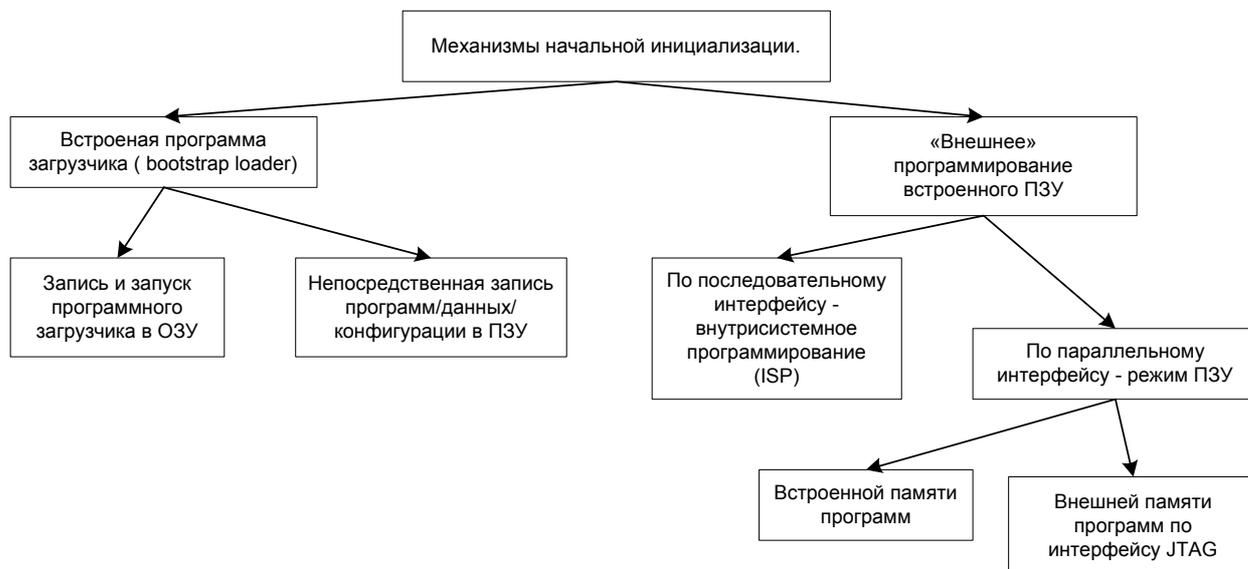


Рисунок 19 Механизмы инициализации встроенной памяти

Встроенная программа загрузчика (Bootstrap loader) - специальная программа, записанная при производстве процессора в специальный блок встроенной памяти программ ПЗУ. При выполнении bootstrap loader принимает записываемые программы или данные через последовательный порт (обычно порт UART) и записывает их в память процессора.

Возможны несколько вариантов сохранения загруженной программы:

1. Программа загружается в ОЗУ и сразу после этого ей передается управление. Это должен быть загрузчик, в свою очередь принимающий и записывающий во встроенное или внешнее ПЗУ (обычно это – FLASH-память) прикладную программу. После рестарта управление передается прикладной программе.
2. Программа записывается непосредственно во встроенное ПЗУ и начинает исполняться после перезапуска в нормальном режиме. В качестве загруженной программы может выступать загрузчик или целевой код. Такой режим используется, например, в семействах MB90F (Fujitsu), MSP430 (Texas Instruments).

Переход в режим bootstrap loader обычно выполняется подачей специального кода на конфигурационные выводы с одновременным рестартом процессора.

Внешнее программирование встроенного ПЗУ.

В данном режиме процессор со встроенным ПЗУ (OTP, EEPROM, FLASH) рассматривается как обычная микросхема ПЗУ с последовательным или параллельным интерфейсом. Ядро процессора при этом отключено; некоторые семейства, например, AVR (Atmel), требуют, чтобы функционировал генератор синхроимпульсов. Переключение процессора в режим внешнего программирования обычно осуществляется подачей на специальный вывод напряжения программирования с уровнем около 12В или подачей специального кода на конфигурационные выводы с одновременным рестартом процессора (аналогично переходу в режим bootstrap loader).

В зависимости от типа интерфейса различают два варианта внешнего программирования:

1. Программирование по параллельному интерфейсу (поддерживается семействами AVR, MCS51, Z8). Процессор или микроЭВМ программируется вне целевой системы, в специальных программаторах. Порты ввода-вывода и/или сигналы внешней шины используются в качестве линий адреса/данных/управления ПЗУ.

Достоинства:

- Простые алгоритмы программирования;
- Высокая степень защиты от случайного перепрограммирования в системе;

Недостатки:

- Необходимо использовать панельку для микропроцессора или программировать однократно перед монтажом печатной платы системы. Последнее не позволяет модифицировать (обновлять) программное обеспечение системы.
2. Программирование по последовательному интерфейсу (поддерживается семействами AVR, PICmicro). В этом режиме адреса, данные и команды доступа к ПЗУ (записи, чтения, проверки и другие) передаются по специальному или стандартному последовательному интерфейсу.

Достоинства:

- Небольшое количество сигналов (2-5 шт.) позволяет подключать программатор к микросхеме установленной на плате и программировать ее, не отключая от схемы. В связи с этим режим последовательного программирования часто называют режимом внутрисистемного программирования (In System Programming (ISP)).
- Неограниченность числа различных команд, которые можно передавать в последовательном коде, позволяет значительно увеличить функциональные возможности программатора.

Недостатки:

- Относительно сложный протокол (алгоритм) программирования.

В настоящее время в режиме внутрисистемного программирования микропроцессорных систем с внешним или внутренним ПЗУ начинает широко использоваться последовательный интерфейс JTAG (IEEE-1049). JTAG – интерфейс граничного сканирования, позволяющий устанавливать на ножках микросхемы сигналы с определенным значением и считывать значения сигналов, установленные внешними схемами или внутренними подсистемами процессора. С помощью интерфейса JTAG имитируется диаграмма записи во внешнее ПЗУ или во внутреннее ПЗУ (процессор должен находиться в режиме внешнего параллельного программирования).