**АРХИТЕКТУРА МИКРОКОНТРОЛЛЕРОВ**

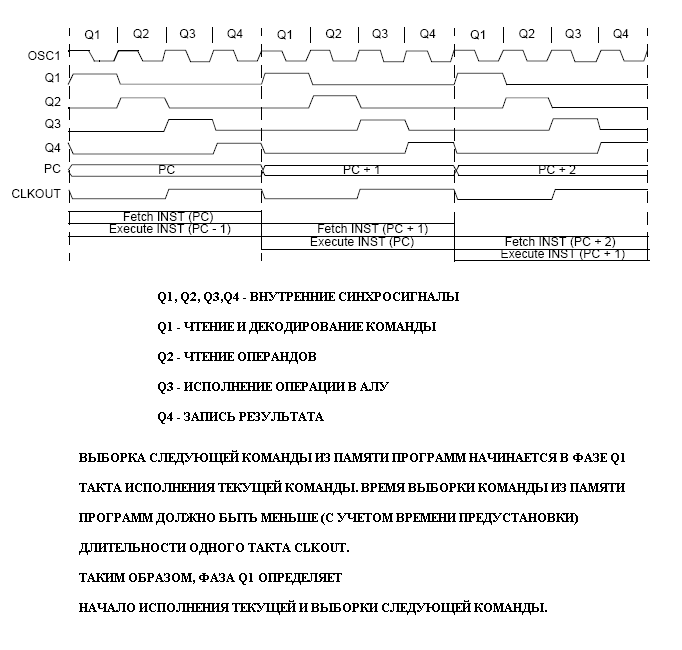
**ОСОБЕННОСТИ ISA 8-МИ РАЗРЯДНЫХ МИКРОКОНТРОЛЛЕРОВ**

* упрощенная и ориентированная на ПОСТРОЕНИЕ СИСТЕМ ПРОГРАММНО-ЛОГИЧЕСКОГО УПРАВЛЕНИЯ АРХИТЕКТУРА системЫ команд (RISC – AVR, PIC; CISC – MCS51)
* простейшие режимы адресации ЦЕЛОЧИСЛЕННЫХ операндов (НЕПОСРЕДСТВЕННАЯ, ПРЯМАЯ, КОСВЕННАЯ)
* КОМАНДЫ ОБРАБОТКИ БИТОВЫХ ДАННЫХ

**ОСОБЕННОСТИ МИКРОАРХИТЕКТУРЫ 8-МИ РАЗРЯДНЫХ МИКРОКОНТРОЛЛЕРОВ**

* гарвардская архитектура
* ОГРАНИЧЕННАЯ ёмкость внутренней памяти (ОЗУ ДАННЫХ, FLASH, EPROM, EEPROM – ПАМЯТЬ ПРОГРАММ) С ВОЗМОЖНОСТЬЮ ПОДКЛЮЧЕНИЯ ВНЕШНЕЙ ПАМЯТИ ПРОГРАММ И ДАННЫХ (MCS51)
* АККУМУЛЯТОРНОЕ (MCS51, PIC) ИЛИ РЕГИСТРОВОЕ (AVR) АЛУ
* ДВУХСТУПЕНЧАТЫЙ КОНВЕЙЕР КОМАНД (PIC)
* ВВИДУ ИНТЕГРАЦИИ НА КРИСТАЛЛЕ МИКРОКОНТРОЛЛЕРА БОЛЬШОГО ЧИСЛА ИНТЕРФЕЙСНЫХ УСТРОЙСТВ (ДЛЯ КАЖДОЙ КОНКРЕТНОЙ ЗАДАЧИ, КАК ПРАВИЛО ИЗБЫТОЧНОГО) И ОГРАНИЧЕНИЙ НА КОЛИЧЕСТВО ВЫВОДОВ КОРПУСА МИКРОКОНТРОЛЛЕРА ДЛЯ ОПТИМИЗАЦИИ СИСТЕМ ШИРОКО ИСПОЛЬЗУЮТСЯ ВЫВОДЫ С ПРОГРАММИРУЕМЫМИ ФУНКЦИЯМИ

**ДВУХСТУПЕНЧАТЫЙ КОНВЕЙЕР КОМАНД В RISC-МИКРОКОНТРОЛЛЕРЕ**

****

**ОСНОВНЫЕ ХАРАКТЕРИСТИКИ АРХИТЕКТУРЫ MCS51**

* 8-разряднОЕ процессорНОЕ ЯДРО, с системой команд, оптимизированной под задачи управления
* КОМАНДЫ РАЗЛИЧНОЙ ДЛИНЫ (1-3 БАЙТА) И ВРЕМЕНИ ВЫПОЛНЕНИЯ (1-4 ПРОЦЕССОРНОГО ЦИКЛА)
* ПРОЦЕССОРНЫЙ ЦИКЛ – 12 ТАКТОВ;
* РазвитАЯ группА команд обработки битовых данных (АКТУАЛЬНО ДЛЯ СИСТЕМ ПРОГРАММНО-ЛОГИЧЕСКОГО УПРАВЛЕНИЯ)
* внутреннЯЯ память данных (ОЗУ) ёмкостью 128 байт (В РАСШИРЕННЫХ ВЕРСИЯХ 256 ) - области контекстно-переключаемых банков регистров, стека , пользовательских данных, в том числе битовых
* внутреннЯЯ память программ (FLASH, EPROM, EEPROM) объемом 4 Кбайт (МОЖЕТ ОТСУТСТВОВАТЬ)
* возможность подключения внешней памяти программ И ДАННЫХ объемом 64 Кб
* ДвухуровневАЯ приоритетнАЯ системА прерываний от трёх внутренних и двух внешних источников

**ПЕРИФЕРИЯ АРХИТЕКТУРЫ MCS51**

* 32 двунаправленныЕ и индивидуально программируемыЕ GPIO (пользовательские контакты), сгруппированныЕ в четыре 8-разрядных порта P0… P3, (В РАСШИРЕННЫХ ВЕРСИЯХ 6 ПОРТОВ)
* ПОРТЫ (Р0 И Р2) в альтернативном режиме используются для организации внешней шины
* два 16-разрядных таймера/счётчика TC0 и TC1 ДЛЯ ФОРМИРОВАНИЯ И ИЗМЕРЕНИЯ ВРЕМЕННЫХ ИНТЕРВАЛОВ, ПОДСЧЕТА ВНЕШНИХ СОБЫТИЙ
* полнодуплексный асинхронный последовательный приемопередатчик (UART) ДЛЯ ОРГАНИЗАЦИИ ВЗАИМОДЕЙСТВИЯ С ВНЕШНИМИ УСТРОЙСТВАМИ ПО ИНТЕРФЕЙСУ RS-232;
* В РАСШИРЕННЫХ (ADVANCED) ВЕРСИЯХ – СТОРОЖЕВОЙ ТАЙМЕР ДЛЯ ЗАЩИТЫ ОТ ПРОГРАММНЫХ СБОЕВ, КОНТРОЛЛЕР ШИРОТНО-ИМПУЛЬСНОЙ МОДУЛЯЦИИ (ШИМ), МНОГОКАНАЛЬНЫЙ АЦП, КОНТРОЛЛЕР ИНТЕРФЕЙСА I2C

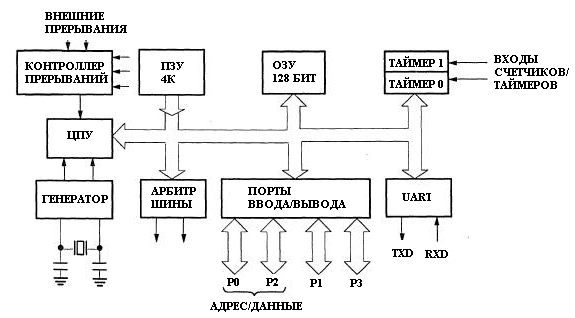
**ОСОБЕННОСТИ СИСТЕМЫ КОМАНД MCS51**

* КОЛИЧЕСТВО КОМАНД – 111 ( ГРУППЫ КОМАНД: АРИФМЕТИЧЕСКИЕ, ЛОГИЧЕСКИЕ, УПРАВЛЕНИЯ, ПЕРЕСЫЛКИ, ОБРАБОТКИ БИТОВЫХ ДАННЫХ)
* СПОСОБЫ АДРЕСАЦИИ:

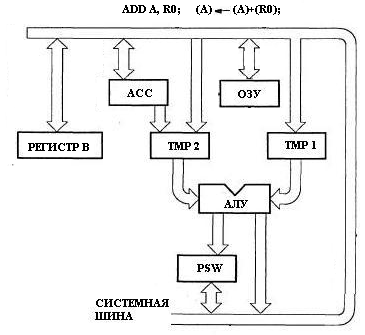
1. НЕПОСРЕДСТВЕННАЯ – ADD A, #4Eh; (A)←(A)+4Eh;

1. ПРЯМАЯ – ADD A, 4Eh; (A)←(A)+(4Eh);
2. РЕГИСТРОВАЯ – ADD A, R0; (A)←(A)+(R0);
3. КОСВЕННАЯ – ADD A, @R0; (A)←(A)+((R0)); (ТОЛЬКО ДЛЯ R0 И R1);
4. ИНДЕКСНАЯ (ТОЛЬКО ДЛЯ ЧТЕНИЯ ИЗ ПАМЯТИ ПРОГРАММ (КОД ИЛИ КОНСТАНТА)) – MOVC A, <@A+БАЗОВЫЙ РЕГИСТР> MOVC A, @A+DPTR, MOVC A, @A+PC. БАЗОВЫЙ РЕГИСТР - РЕГИСТР УКАЗАТЕЛЬ DPTR ИЛИ СЧЕТЧИК КОМАНД РС. АККУМУЛЯТОР А СОДЕРЖИТ ИНДЕКС

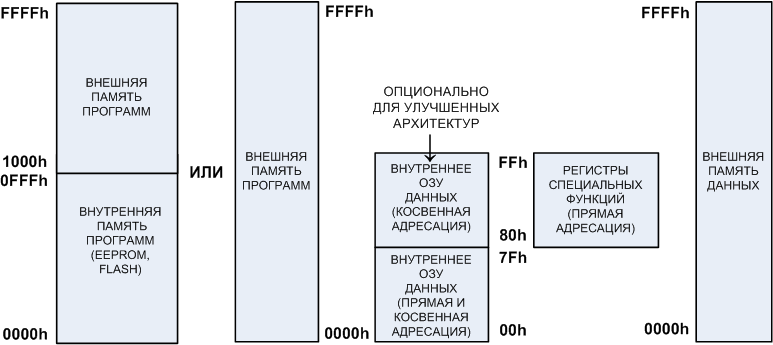
**СТРУКТУРА ЯДРА MCS51**

****

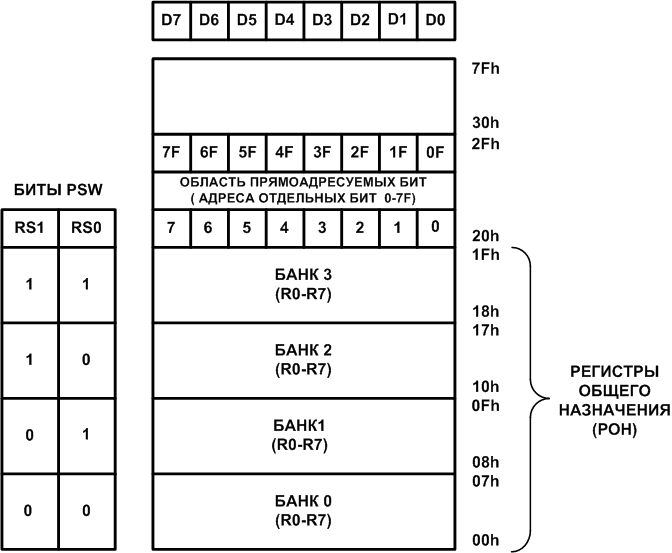
**АККУМУЛЯТОРНОЕ АЛУ MCS51**

****

**КАРТА ПАМЯТИ MCS51**

****

**КАРТА ОЗУ ДАННЫХ**

****

**ОСОБЕННОСТИ ОРГАНИЗАЦИИ СТЕКА**

* ДЛЯ ПОДДЕРЖКИ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ И ОБРАБОТКИ ПРЕРЫВАНИЙ ПРЕДУСМОТРЕН СТЕК (ВЕРХНИЙ), КОТОРОМУ ОТВОДИТСЯ КОНФИГУРИРУЕМАЯ ПОЛЬЗОВАТЕЛЕМ ОБЛАСТЬ ВО ВНУТРЕННЕЙ ПАМЯТИ ДАННЫХ
* ДЛЯ РАБОТЫ СО СТЕКОМ ИСПОЛЬЗУЕТСЯ РЕГИСТР УКАЗАТЕЛЬ СТЕКА SP. ПО УМОЛЧАНИЮ В НЕМ ЗАПИСАНО ЗНАЧЕНИЕ 07h. SP указывает на верхнюю занятую стекОМ ячейку ОЗУ
* ПРИ ЗАПИСИ В СТЕК ПО АДРЕСУ, СОДЕРЖАЩЕМУСЯ В SP, ЕГО ЗНАЧЕНИЕ ПРЕДВАРИТЕЛЬНО ИНКРЕМЕНТИРУЕТСЯ
* ПОСЛЕ СчитываниЯ из стека ПО АДРЕСУ, СОДЕРЖАЩЕМУСЯ В SP, значение SP ДЕКРЕМЕНТИРУЕТСЯ.

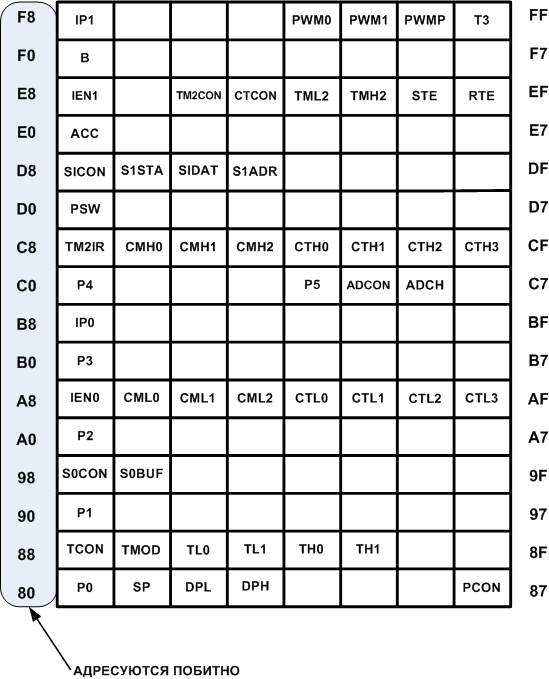
**ОБРАТИТЬ ВНИМАНИЕ:**

* ПРИ ВЫЗОВЕ ПОДПРОГРАММ И ОБРАБОТКЕ ПРЕРЫВАНИЙ В СТЕК ЗАПИСЫВАЕТСЯ ТОЛЬКО АДРЕС КОМАНДЫ ВОЗВРАТА
* ЕСЛИ В ПРОГРАММЕ ПОЛЬЗОВАТЕЛЯ ПРЕДПОЛАГАЕТСЯ ИСПОЛЬЗОВАТЬ НЕ ТОЛЬКО 0-ОЙ БАНК РОН (R0 – R7), ТО СТЕК НЕОБХОДИМО ПЕРЕНЕСТИ В ВЕРХНЮЮ ОБЛАСТЬ ВНУТРЕННЕЙ ПАМЯТИ ДАННЫХ

**НАЗНАЧЕНИЕ РЕГИСТРОВ СПЕЦИАЛЬНЫХ ФУНКЦИЙ**

1. КОНФИГУРИРОВАНИЕ И УПРАВЛЕНИЕ СИСТЕМОЙ ПРЕРЫВАНИЙ
2. КОНФИГУРИРОВАНИЕ И УПРАВЛЕНИЕ ТАЙМЕРАМИ
3. ХРАНЕНИЕ СЛОВА СОСТОЯНИЯ ПРОГРАММЫ
4. УПРАВЛЕНИЕ СТЕКОМ
5. ПАРАЛЛЕЛЬНЫЙ И ПОСЛЕДОВАТЕЛЬНЫЙ ВВОДА/ВЫВОДА – ВЫДЕЛЕННОЕ АДРЕСНОЕ ПРОСТРАНСТВО ВВОДА/ВЫВОДА (ПРЯМАЯ АДРЕСАЦИЯ), ВНУТРЕННЕЕ ОЗУ ДАННЫХ (КОСВЕННАЯ АДРЕСАЦИЯ – В ADVANCED МК). ПРИ ИСПОЛЬЗОВАНИИ ДЛЯ ВВОДА/ВЫВОДА СИСТЕМНОЙ ШИНЫ – РАЗДЕЛЯЕМОЕ АДРЕСНОЕ ПРОСТРАНСТВО ВВОДА/ВЫВОДА

**КАРТА РЕГИСТРОВ СПЕЦИАЛЬНЫХ ФУНКЦИЙ (ДЛЯ 80С522)**



**PSW – ПОБИТНО АДРЕСУЕМЫЙ РЕГИСТР СЛОВА СОСТОЯНИЯ ПРОГРАММЫ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **PSW.7** | **PSW.6** | **PSW.5** | **PSW.4** | **PSW.3** | **PSW.2** | **PSW.1** | **PSW.0** |
| CY | AC | F0 | RS1 | RS0 | OV | - | P |

**ПОРТЫ ПАРАЛЛЕЛЬНОГО ВВОДА/ВЫВОДА (Р0-Р3)**

* КАЖДЫЙ РАЗРЯД 8-МИ БИТНОГО ПОРТА СОДЕРЖИТ:
* ТРИГГЕР-ЗАЩЕЛКУ,
* ВХОДНОЙ БУФЕР
* ВЫХОДНОЙ ДРАЙВЕР
* ПОРТ Р0 – ДВУНАПРАВЛЕННЫЙ,

Р1-Р3 - КВАЗИДВУНАПРАВЛЕННЫЕ

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ:

* ПАРАЛЛЕЛЬНЫЙ И ПОБИТНЫЙ ВВОД/ВЫВОД
* ДЛЯ ИСПОЛЬЗОВАНИЯ В РЕЖИМЕ ВВОДА В ТРИГГЕР-ЗАЩЕЛКУ ЛИНИИ ПОРТА НЕОБХОДИМО ЗАПИСАТЬ «1» (УСТАНАВЛИВАЮТСЯ АВТОМАТИЧЕСКИ СИГНАЛОМ СБРОСА)

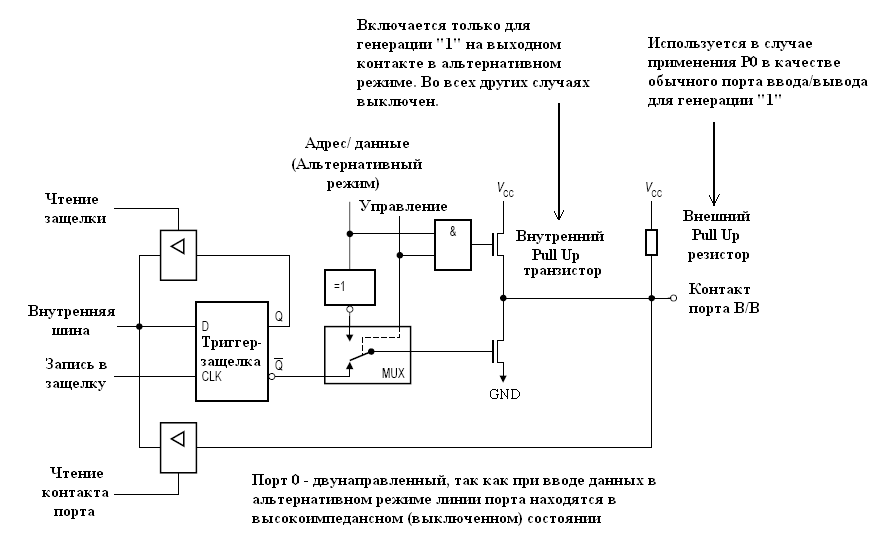
**АЛЬТЕРНАТИВНЫЕ ФУНКЦИИ ПОРТОВ**

* ПОРТЫ Р0, Р2 И Р3 МОГУТ ИСПОЛЬЗОВАТЬСЯ В АЛЬТЕРНАТИВНОМ РЕЖИМЕ
* В АЛЬТЕРНАТИВНОМ РЕЖИМЕ ЧЕРЕЗ ПОРТ Р0 В ЦИКЛЕ ОБРАЩЕНИЯ К ВНЕШНЕЙ ПАМЯТИ ПРОГРАММ/ДАННЫХ ПОСЛЕДОВАТЕЛЬНО ВЫВОДИТСЯ МЛАДШАЯ ЧАСТЬ АДРЕСА И ПРИНИМАЕМЫЕ/ВЫДАВАЕМЫЕ ДАННЫЕ
* В АЛЬТЕРНАТИВНОМ РЕЖИМЕ ЧЕРЕЗ ПОРТ Р2 В ЦИКЛЕ ОБРАЩЕНИЯ К ВНЕШНЕЙ ПАМЯТИ ДАННЫХ ВЫВОДИТСЯ СТАРШАЯ ЧАСТЬ АДРЕСА
* В АЛЬТЕРНАТИВНОМ РЕЖИМЕ ТРИГГЕРЫ-ЗАЩЕЛКИ ПОРТОВ Р0 И Р2 ОТКЛЮЧАЮТСЯ ОТ ВЫХОДНЫХ ДРАЙВЕРОВ, А ДЛЯ Р3 УСТАНАВЛИВАЮТСЯ В СОСТОЯНИЕ «1»

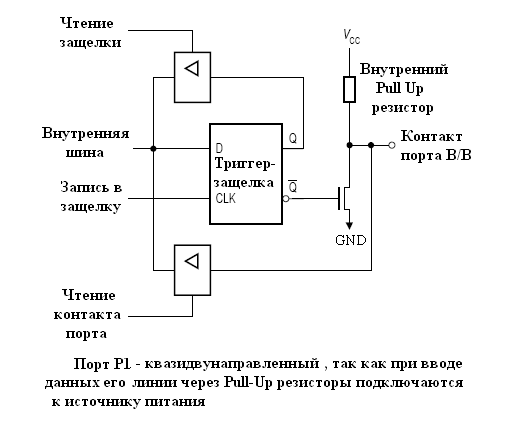
**АЛЬТЕРНАТИВНЫЕ ФУНКЦИИ ЛИНИЙ ПОРТА Р3**

|  |  |  |
| --- | --- | --- |
| **ИМЯ БИТА** | **ОБОЗНАЧЕНИЕ** | **ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ** |
| Р3.0 | RxD | Вход приемника последовательного порта (УАПП) |
| Р3.1 | TxD | Выход передатчика последовательного порта (УАПП) |
| Р3.2 | INT0 | Вход запроса внешнего прерывания 0 |
| Р3.3 | INT1 | Вход запроса внешнего прерывания 1 |
| Р3.4 | T0 | Внешний счетный вход таймера/счётчика 0 |
| Р3.5 | T1 | Внешний счетный вход таймера/счётчика 1 |
| Р3.6 | WR | Строб записи во внешнюю память данных |
| Р3.7 | RD | Строб чтения из внешней памяти данных |

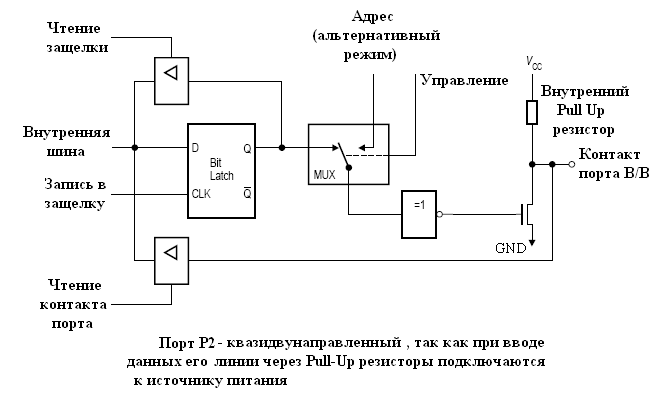
**ФУНКЦИОНАЛЬНАЯ СХЕМА ПОРТА Р0**



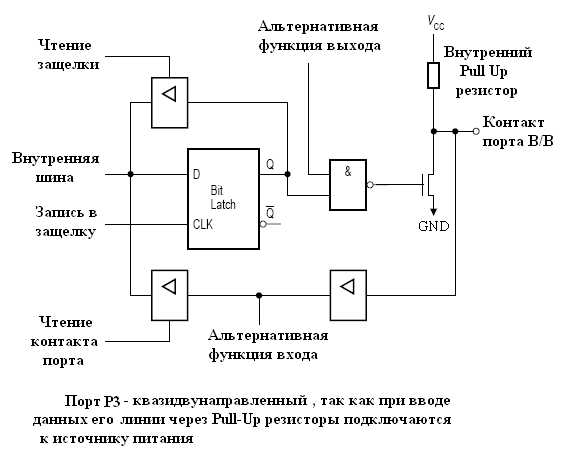
**ФУНКЦИОНАЛЬНАЯ СХЕМА ПОРТА Р1**

****

**ФУНКЦИОНАЛЬНАЯ СХЕМА ПОРТА Р2**

****

**ФУНКЦИОНАЛЬНАЯ СХЕМА ПОРТА Р3**

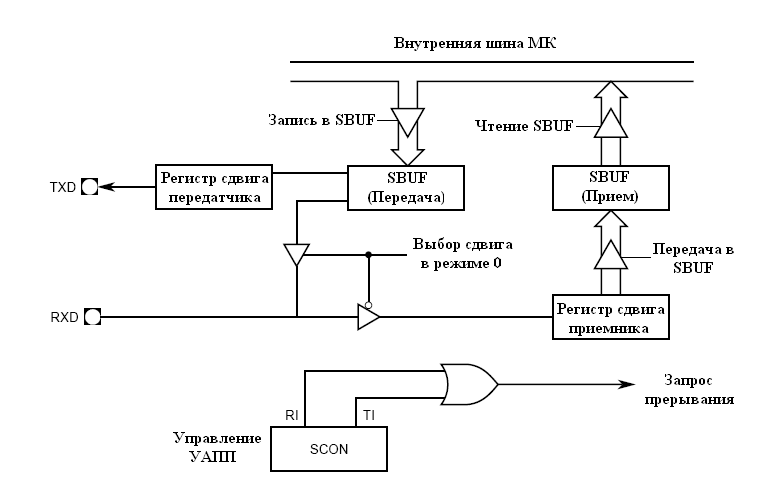
****

**ОСОБЕННОСТЬ ВВОДА ДАННЫХ ЧЕРЕЗ ПОРТЫ ВВОДА/ВЫВОДА**

* ПРИ РАБОТЕ С ПОРТАМИ ВВОДА/ВЫВОДА РЯД КОМАНД ИСПОЛЬЗУЕТ ЗНАЧЕНИЕ, УСТАНОВЛЕННОЕ НЕПОСРЕДСТЕННО НА КОНТАКТАХ ВВОДА/ВЫВОДА (MOV A,P3), А РЯД ДРУГИХ ЗНАЧЕНИЕ В ТРИГГЕРЕ-ЗАЩЕЛКЕ (ANL P1, #011h)

* В КОМАНДАХ ВТОРОГО ТИПА ПОРТ ОДНОВРЕМЕННО ЯВЛЯЕТСЯ ОПЕРАНДОМ И МЕСТОМ НАЗНАЧЕНИЯ РЕЗУЛЬТАТА. ТАКИЕ КОМАНДЫ ВЫПОЛНЯЮТ ЦИКЛ «ЧТЕНИЕ – МОДИФИКАЦИЯ – ЗАПИСЬ»
* ИСПОЛЬЗОВАНИЕ ТРИГГЕРА-ЗАЩЕЛКИ КАК ИСТОЧНИКА ОПЕРАНДА ЯВЛЯЕТСЯ МЕХАНИЗМОМ ЗАЩИТЫ ОТ НЕПРАВИЛЬНОГО СЧИТЫВАНИЯ РАНЕЕ ЗАПИСАННЫХ В ПОРТ ДАННЫХ
* НЕПРАВИЛЬНОЕ ЗНАЧЕНИЕ НА ВЫХОДЕ ДРАЙВЕРА ПОРТА, ОТЛИЧНОЕ ОТ ЗАПИСАННОГО В ТРИГГЕР-ЗАЩЕЛКУ, МОЖЕТ БЫТЬ ОБУСЛОВЛЕНО ИЗМЕНЕНИЕМ И, КАК СЛЕДСТВИЕ, НЕОДНОЗНАЧНОЙ ИНТЕРПРЕТАЦИЕЙ УРОВНЯ СИГНАЛА НА ЛИНИИ ПОРТА ИЗ-ЗА ОСОБЕННОСТЕЙ РЕАЛИЗАЦИИ ЦЕПИ НАГРУЗКИ (НАПРИМЕР, ПРИ ПОВЫШЕНИИ ЗНАЧЕНИЯ ВЫХОДНОГО ТОКА УРОВЕНЬ СИГНАЛА МОЖЕТ СНИЖАТЬСЯ)

**ПОСЛЕДОВАТЕЛЬНЫЙ ПОРТ (УАПП). УПРОЩЕННАЯ ФУНКЦИОНАЛЬНАЯ СХЕМА**

****

**ОСОБЕННОСТИ ФУНКЦИОНИРОВАНИЯ УАПП**

* ОБЕСПЕЧИВАЕТ ПОЛНОДУПЛЕКСНЫЙ АСИНХРОННЫЙ ОБМЕН С БУФЕРИЗАЦИЕЙ ПРИНИМАЕМЫХ ДАННЫХ (РЕЖИМЫ 1,2,3) И СИНХРОННЫЙ ПРИЕМ/ВЫДАЧУ (РЕЖИМ 0) ПОСЛЕДОВАТЕЛЬНЫХ ДАННЫХ
* ПОСЛЕДОВАТЕЛЬНЫЙ ПРИЕМ ДАННЫХ ОСУЩЕСТВЛЯЕТСЯ В СДВИГОВОМ РЕГИСТРЕ ПРИЕМНИКА. ПОСЛЕ ПРИЕМА ПОСЫЛКИ СОДЕРЖИМОЕ СДВИГОВОГО РЕГИСТРА ПЕРЕДАЕТСЯ В БУФЕР - РЕГИСТР СПЕЦИАЛЬНЫХ ФУНКЦИЙ С ИМЕНЕМ SBUF
* БУФЕР ОБЕСПЕЧИВАЕТ ПРИЕМ ВТОРОЙ ПОСЫЛКИ В СДВИГОВЫЙ РЕГИСТР ПОКА МИКРОКОНТРОЛЛЕР СЧИТЫВАЕТ ПЕРВУЮ ИЗ SBUF. ЧТОБЫ НЕ ПРОИЗОШЛА ПОТЕРЯ ПРИНИМАЕМЫХ ДАННЫХ СЧИТЫВАНИЕ ПЕРВОЙ ПОСЫЛКИ ИЗ БУФЕРА ДОЛЖНО ЗАВЕРШИТЬСЯ К МОМЕНТУ ОКОНЧАНИЯ ПРИЕМА ВТОРОЙ

* БУФЕР ПЕРЕДАТЧИКА ИМЕЕТ ТАКОЕ ЖЕ ИМЯ SBUF, КАК И БУФЕР ПРИЕМНИКА (ФИЗИЧЕСКИ ЭТО РАЗНЫЕ РЕГИСТРЫ). РАССМОТРЕННАЯ ОСОБЕННОСТЬ ОБРАБОТКИ ПОСЫЛОК ДАННЫХ ДЛЯ ПРИЕМНИКА РАСПРОСТРАНИЯЕТСЯ ТАКЖЕ НА РАБОТУ ТРАКТА ПЕРЕДАТЧИКА УАПП

**РЕЖИМЫ РАБОТЫ УАПП (0,1)**

* РЕЖИМ 0 (СИНХРОНИЗИРУЕМЫЙ РЕГИСТР СДВИГА):

ЧЕРЕЗ ВЫВОД Rx УАПП ПЕРЕДАЕТСЯ ИЛИ ПРИНИМАЕТСЯ НАЧИНАЯ С МЛАДШЕГО БИТА ОДНОБАЙТОВАЯ ПОСЫЛКА. ЧЕРЕЗ ВЫВОД Тx С 1/12 ТАКТОВОЙ ЧАСТОТЫ МК ВЫДАЮТСЯ СИНХРОНИЗИРУЮЩИЕ КАЖДЫЙ СДВИГ ИМПУЛЬСЫ

* РЕЖИМ 1 (8-МИ БИТОВЫЙ УАПП С ПЕРЕМЕННОЙ СКОРОСТЬЮ ОБМЕНА):

ИСПОЛЬЗУЕТСЯ 10-ТИ БИТОВАЯ ПОСЫЛКА, КОТОРАЯ СОСТОИТ ИЗ СТАРТОВОГО БИТА («0»), 8-МИ ИНФОРМАЦИОННЫХ БИТ И СТОП-БИТА («1»)

* СТОП-БИТ ПОСЛЕ ПРИЕМА ПОСЫЛКИ ЗАПИСЫВАЕТСЯ В БИТ RB8 РЕГИСТРА SCON. СКОРОСТЬ ОБМЕНА ЗАДАЕТСЯ ТАЙМЕРОМ 1

**РЕЖИМ 2 РАБОТЫ УАПП**

* 9-ТИ БИТОВЫЙ УАПП С ФИКСИРОВАННОЙ СКОРОСТЬЮ ОБМЕНА:
* ИСПОЛЬЗУЕТСЯ 11-ТИ БИТОВАЯ ПОСЫЛКА, КОТОРАЯ СОСТОИТ ИЗ СТАРТОВОГО БИТА («0»), 8-МИ ИНФОРМАЦИОННЫХ БИТ, 9-ГО ПРОГРАММИРУЕМОГО БИТА И СТОП-БИТА («1»)
* ПРИ ПЕРЕДАЧЕ 9-Й БИТ ЗАПИСЫВАЕТСЯ В БИТ ТВ8 РЕГИСТРА SCON. ПОСЛЕ ПРИЕМА ПОСЫЛКИ В БИТ RB8 РЕГИСТРА SCON ЗАПИСЫВАЕТСЯ ЗНАЧЕНИЕ 9-ГО ПРОГРАММИРУЕМОГО БИТА, СТОП-БИТ ИГНОРИРУЕТСЯ
* В КАЧЕСТВЕ 9-ГО БИТА МОЖНО ИСПОЛЬЗОВАТЬ БИТ «Р» (PSW.O). В ЭТОМ СЛУЧАЕ ВЫПОЛНЯЕТСЯ КОНТРОЛЬ ОБМЕНА ПО ЧЕТНОСТИ
* СКОРОСТЬ ОБМЕНА РАВНА 1/32 ИЛИ 1/64 ТАКТОВОЙ ЧАСТОТЫ МК В ЗАВИСИМОСТИ ОТ ЗНАЧЕНИЯ БИТА SMOD РЕГИСТРА PCON

**РЕЖИМ 3 РАБОТЫ УАПП**

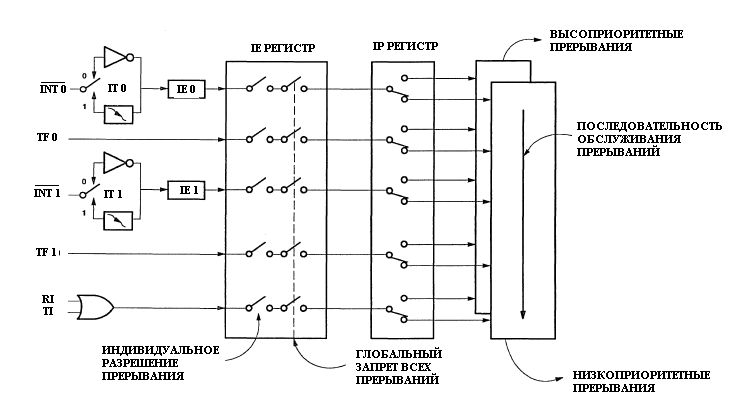
9-ТИ БИТОВЫЙ УАПП С ПЕРЕМЕННОЙ СКОРОСТЬЮ ОБМЕНА:

* ДАННЫЙ РЕЖИМ АНАЛОГИЧЕН РЕЖИМУ 2 ЗА ИСКЛЮЧЕНИЕМ ОГРАНИЧЕНИЙ НА СКОРОСТЬ ОБМЕНА
* В РЕЖИМЕ 3 СКОРОСТЬ ОБМЕНА ЯВЛЯЕТСЯ ПЕРЕМЕННОЙ И ЗАДАЕТСЯ ТАЙМЕРОМ 1

**ФОРМАТ УПРАВЛЯЮЩЕГО РЕГИСТРА SCON**

|  |  |  |
| --- | --- | --- |
| **ИМЯ БИТА** | **ОБОЗНАЧЕНИЕ** | **ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ** |
| SCON.7  SCON.6 | SM0  SM1 | Выбор режима работы УАПП |
| SCON.5 | SM2 | Разрешение работы в составе многопроцессорной системы (магистральная архитектура с общим моноканалом). В режимах 2 и 3 флаг прерывания RI не вырабатывается, если  SM2=1 и принятый 9-й бит в RB8=0. В режиме 1 при SM2=1 флаг RI не устанавливается, пока не будет получен корректный стоп-бит. В режиме 0 SM2=0. |
| SCON.4 | REN | Бит разрешения работы приемника |
| SCON.3 | TB8 | 9-й бит передаваемых данных в режимах 2 и 3 |
| SCON.2 | RB8 | 9-й бит принимаемых данных в режимах 2 и 3 |
| SCON.1 | TI | Флаг прерывания передатчика |
| SCON.0 | RI | Флаг прерывания приемника |

**СИСТЕМА ПРЕРЫВАНИЙ MCS51**

****

**ПРИОРИТЕТЫ ЗАПРОСОВ ПРЕРЫВАНИЙ**

* В МК реализована двухуровневая система приоритетов прерываний. Каждому прерыванию при помощи программирования регистра IP НАЗНАЧАЕТСЯ низкий (0) или высокий (1) уровень приоритета
* Обработка прерывания низкого уровня может быть прервана только прерыванием высокого уровня
* Обработка прерывания высокого уровня не может быть прервана никаким другим прерыванием
* При одновременном поступлении двух прерываний различного уровня приоритета предпочтение для обработки отдается прерыванию высокого уровня
* При одновременном поступлении прерываний одного уровня порядок их обработки определяется таблицей (см. следующий слайд)

**ПОРЯДОК ОБРАБОТКИ ПРЕРЫВАНИЙ**

|  |  |  |
| --- | --- | --- |
| **Источник прерывания** | **Обозначение**  **флагов** | **Приоритет в пределах одного уровня** |
| Внешнее прерывание INT0 | IE0 | Высший  Низший |
| Переполнение таймера Т0 | TF0 |
| Внешнее прерывание INT1 | IE1 |
| Переполнение таймера Т1 | TF1 |
| UART | RI/TI |

**IE – РЕГИСТР РАЗРЕШЕНИЯ ПРЕРЫВАНИЙ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **IP.6** | **IP.5** | **IP.4** | **IP.3** | **IP.2** | **IP.1** | **IP.0** |
| - | - | - | PS | PT1 | PX1 | PT0 | PX0 |

**РS - приоритет прерывания от UART**

**РT1 - приоритет прерывания от таймера Т1**

**РX1 - приоритет внешнего прерывания INT1**

**РT0 - приоритет прерывания от таймера Т0**

**РX0 - приоритет внешнего прерывания INT0**

**IР – РЕГИСТР ПРИОРИТЕТА ПРЕРЫВАНИЙ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **IP.7** | **IP.6** | **IP.5** | **IP.4** | **IP.3** | **IP.2** | **IP.1** | **IP.0** |
| - | - | - | PS | PT1 | PX1 | PT0 | PX0 |

**РS - приоритет прерывания от UART**

**РT1 - приоритет прерывания от таймера Т1**

**РX1 - приоритет внешнего прерывания INT1**

**РT0 - приоритет прерывания от таймера Т0**

**РX0 - приоритет внешнего прерывания INT0**

**ТАБЛИЦА ВЕКТОРОВ ПРЕРЫВАНИЙ**

|  |  |
| --- | --- |
| **ИСТОЧНИК ПРЕРЫВАНИЯ**  **(ФЛАГ ПРЕРЫВАНИЯ)** | **АДРЕС ВЕКТОРА ПРЕРЫВАНИЙ** |
| Внешнее прерывание (IЕ0) | 03h |
| Таймер/счётчик (TF0) | 0Bh |
| Внешнее прерывание (IЕ1) | 13h |
| Таймер/счётчик (TF1) | 1Bh |
| Последовательный порт (RI или TI) | 23h |

**ОСОБЕННОСТИ ОБРАБОТКИ ВНУТРЕННИХ ПРЕРЫВАНИЙ**

* ФЛАГИ TF0 И TF1 (ПЕРЕПОЛНЕНИЕ ТАЙМЕРОВ-СЧЕТЧИКОВ) СБРАСЫВАЮТСЯ АВТОМАТИЧЕСКИ ПРИ ВХОДЕ В ОБРАБОТЧИК ПРЕРЫВАНИЯ
* ФЛАГИ RI И TI УСТАНАВЛИВАЮТСЯ АППАРАТНО И ДОЛЖНЫ СБРАСЫВАТЬСЯ ПРОГРАММНО

* ВСЕ ФЛАГИ ПРЕРЫВАНИЙ МОГУТ УСТАНАВЛИВАТЬСЯ/СБРАСЫВАТЬСЯ ПРОГРАММНО, ТО ЕСТЬ ОБРАБОТЧИК ПРЕРЫВАНИЯ МОЖЕТ ВЫЗЫВАТЬСЯ ИЗ ПРОГРАММЫ

**ОБРАБОТКА ВНЕШНИХ ПРЕРЫВАНИЙ**

* Прерывания INT0 и INT1 могут быть сконфигурированы или по изменению 1/0 или низкому уровню сигнала
* Если прерывание сконфигурировано по изменению 1/0 , то для его захвата контроллером прерывания необходимо, чтобы длительность значений «1» и «0» сигнала прерывания была бы не меньше 12 периодов тактового генератора МК (одного процессорного цикла).
* Если прерывание сконфигурировано по низкому уровню сигнала, то его длительность должна быть достаточной для захвата сигнала контроллером прерываний

* сигнал запроса прерывания ДОЛЖЕН БЫТЬ СБРОШЕН перед выходом из обработчика. ДЛЯ ЭТОГО В обработчикЕ ДОЛЖНА БЫТЬ КОМАНДА ПО КОТОРОЙ ВНЕШНЕЕ УСТРОЙСТВО СБРАСЫВАЕТ СИГНАЛ ЗАПРОСА ПРЕРЫВАНИЯ. Невыполнение данного условия приведет к повторному вызову и обработке прерывания
* ДЛЯ КОНФИГУРИРОВАНИЯ ВНЕШНИХ ПРЕРЫВАНИЙ ИСПОЛЬЗУЕТСЯ РЕГИСТР СПЕЦИАЛЬНЫХ ФУНКЦИЙ TCON

**ПЕРЕДАЧА/ВОЗВРАТ УПРАВЛЕНИЯ ПРИ ОБРАБОТКЕ ПРЕРЫВАНИЯ**

* ПРИ ОБРАБОТКЕ РАЗРЕШЕННОГО И НЕБЛОКИРУЕМОГО ПРЕРЫВАНИЯ АППАРАТНО ВЫПОЛНЯЕТСЯ КОМАНДА LCALL, АДРЕС ПЕРЕХОДА В КОТОРОЙ ОПРЕДЕЛЯЕТСЯ ВЕКТОРОМ СООТВЕТСТВУЮЩЕГО ПРЕРЫВАНИЯ
* В СТЕК ПОМЕЩАЕТСЯ СОДЕРЖИМОЕ СЧЕТЧИКА КОМАНД (И ТОЛЬКО!), А В СЧЕТЧИК КОМАНД ЗАГРУЖАЕТСЯ АДРЕС ВЕКТОРА ПРЕРЫВАНИЯ, ПО КОТОРОМУ НАХОДИТСЯ КОМАНДА ПЕРЕДАЧИ УПРАВЛЕНИЯ ОБРАБОТЧИКУ ПРЕРЫВАНИЯ
* ДЛЯ СОХРАНЕНИЯ КОНТЕКСТА ОСНОВНОЙ ПРОГРАММЫ ПОСЛЕ ВХОДА В ОБРАБОТЧИК КОМАНДОЙ PUSH В СТЕК ПОМЕЩАЮТ ЗНАЧЕНИЯ PSW, ACC, DPTR И ДРУГИХ РЕГИСТРОВ, ЗНАЧЕНИЯ В КОТОРЫХ ПЕРЕД ВЫХОДОМ ИЗ ОБРАБОТЧИКА ВОССТАНАВЛИВАЮТСЯ КОМАНДАМИ РОР
* ПОСЛЕДНЕЙ КОМАНДОЙ ОБРАБОТЧИКА ПРЕРЫВАНИЯ ЯВЛЯЕТСЯ КОМАНДА RETI, КОТОРАЯ ПОМЕЩАЕТ В СЧЕТЧИК КОМАНД АДРЕС ВОЗВРАТА В ПРЕРВАННУЮ ПРОГРАММУ

**TCON - регистр управления таймерами и внешниМИ прерываниЯМИ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TCON.7** | **TCON.6** | **TCON.5** | **TCON.4** | **TCON.3** | **TCON.2** | **TCON.1** | **TCON.0** |
| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

РЕГИСТР ОБЕСПЕЧИВАЕТ:

* КОНФИГУРИРОВАНИЕ ВНЕШНИХ ПРЕРЫВАНИЙ (ПО УРОВНЮ ИЛИ ПЕРЕКЛЮЧЕНИЮ 1/0)
* РАЗРЕШЕНИЕ/ЗАПРЕТ РАБОТЫ ТАЙМЕРОВ
* ХРАНЕНИЕ ФЛАГОВ ПЕРЕПОЛНЕНИЯ ТАЙМЕРОВ И ВНЕШНИХ ПРЕРЫВАНИЙ

**Назначение битов регистра TCON**

* **TF1** - флаг переполнения таймера Т1. Устанавливается аппаратно при переполнении и сбрасывается аппаратно при входе в обработчик этого прерывания
* **TR1** - разрешение работы таймера Т1. При TR1=1 таймер Т1 включен, при TR1=0 выключен ( устанавливается/сбрасывается программно)
* **TF0** - флаг переполнения таймера Т0. Устанавливается аппаратно при переполнении и сбрасывается аппаратно при входе в обработчик этого прерывания
* **TR0** - разрешение работы таймера Т0. При TR0=1 таймер Т0 включен, при TR0=0 выключен ( устанавливается/сбрасывается программно)
* **IE1** - флаг обнаружения сигнала прерывания на входе внешнего прерывания INT1. Устанавливается аппаратно при обнаружении и сбрасывается аппаратно при обработке прерывания;
* **IT1** - конфигурация прерывания INT1. При IT1=1 прерывание вызывается изменением 1/0, а при IT1=0 низким уровнем на входе INT1;
* **IE0** - флаг обнаружения сигнала прерывания на входе внешнего прерывания INT0. Устанавливается аппаратно при обнаружении и сбрасывается аппаратно при обработке прерывания;
* **IT0**  - конфигурация прерывания INT0. При IT0=1 прерывание вызывается изменением 1/0, а при IT0=0 низким уровнем на входе INT0.

**ФУНКЦИИ ТАЙМЕРОВ-СЧЕТЧИКОВ Т0 И Т1**

* Т0 и Т1 могут работать в качестве таймера ДЛЯ ФОРМИРОВАНИЯ временных интервалов И измерениЯ ИХ длительности, А ТАКЖЕ или счетчика внешних событий

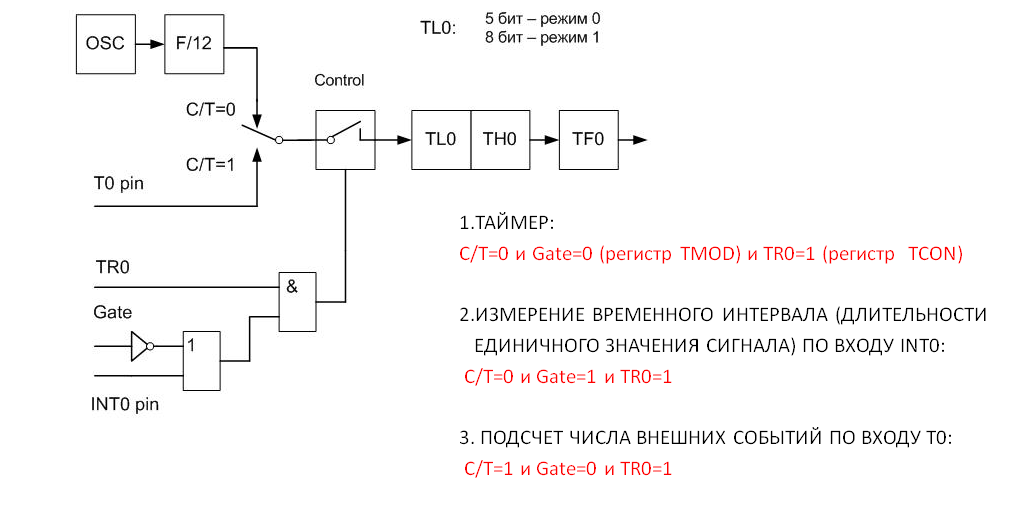
* При работе в качестве таймера состояние Т0 и Т1 инкрементируется 1 раз за машинный цикл (12 периодов тактового генератора)

* При работе в качестве счетчика внешних событий на входах Т0 и Т1 по спаду 1/0 максимальная частота таких изменений не должна превышать 1/24 частоты генератора - Для гарантированного прочтения сигналов «0» и «1» на входах Т0 и Т1 время их удержания должно быть не меньше одного машинного цикла

**РЕЖИМЫ РАБОТЫ Т0 И Т1**

* Таймеры Т0 и Т1 могут работать в одном из следующих четырех режимов: 0, 1, 2 и 3
* В режимах 0, 1 и 2 таймеры Т0 и Т1 полностью идентичны
* В режиме 3 таймер Т1 хранит свое состояние (отключен), а таймер Т0 работает как два независимых таймера
* Выбор функции КАЖДОГО таймера и режима его работы определяется программно при помощи регистра TMOD

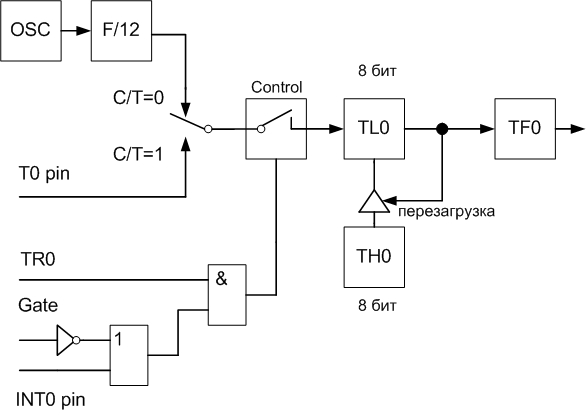
**СХЕМА ДЛЯ РЕЖИМОВ 0 И 1 ТАЙМЕРА Т0**

****

**РЕЖИМЫ 0 И 1**

* В режиме 0 регистр TL0 используется как 5-ти разрядный предделитель ( используются 5 младших разрядов, 3 старших игнорируются). В этом случае таймер конфигурируется как 13-ти разрядный двоичный счетчик. При переполнении этого счетчика устанавливается флаг TF0 и вырабатывается прерывание, если оно разрешено.
* Режим 1 идентичен режиму 0, а единственным отличием является использование всех 8-ми бит регистра TL0. В этом режиме таймер конфигурируется как 16-ти разрядный счетчик.

**СХЕМА ДЛЯ РЕЖИМА 2**

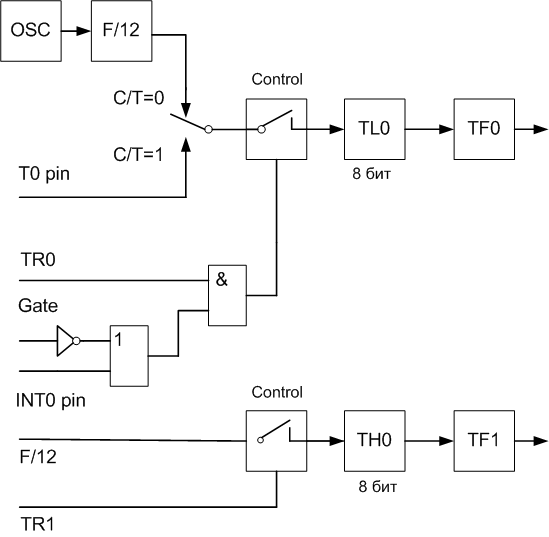
****

**РЕЖИМ 2**

* В режиме 2 таймер конфигурируется как 8-ми разрядный счетчик на основе регистра TL0 . При переполнении TL0 происходит не только установка флага TF0, но и автоматическая перезагрузка TL0 содержимым регистра TH0

* При перезаписи в TL0 значение TH0 не изменяется
* Данный режим используется при необходимости формирования периодических сигналов требуемой частоты, например, для управления скоростью передачи УАПП

**СХЕМА ДЛЯ РЕЖИМА 3**

****

**РЕЖИМ 3**

* В режиме 3 таймер Т1 отключен и регистры TL1 и TH1 сохраняют свое состояние
* Таймер Т0 представляет совокупность двух устройств: таймера/счетчика событий на основе регистра TL0 и таймера на основе регистра TH0

* таймер на основе регистра TH0 использует биты TR1 для разрешения работы и TF1 в качестве флага переполнения

* ЕСЛИ ТАЙМЕР Т0 РАБОТАЕТ В РЕЖИМЕ 3, ТО Таймер Т1 при этом может использоваться В ДРУГИХ РЕЖИМАХ для применений, не требующих использования прерываний - управление скоростью обмена УАПП
* режим 3 выбирается при необходимости использования в составе МК третьего дополнительного 8-ми разрядного таймера.

**РАЗВИТИЕ АРХИТЕКТУРЫ МК НА ОСНОВЕ ЯДРА MCS51**

* ПЕРЕХОД К ТЕХНОЛОГИИ CMOS (СНИЖЕНИЕ ЭНЕРГОПОТРЕБЛЕНИЯ)
* СОКРАЩЕНИЕ ПРОЦЕССОРНОГО ЦИКЛА (2, 6 ТАКТОВ)
* УВЕЛИЧЕНИЕ ОБЪЕМА ВНУТРЕННЕГО ОЗУ (192, 256, 512 БАЙТ)
* УВЕЛИЧЕНИЕ ОБЪЕМА ВНУТРЕННЕЙ ПАМЯТИ ПРОГРАММ (8КБ, 16КБ, 32КБ)
* ПОВЫШЕНИЕ ЧАСТОТЫ СИНХРОНИЗАЦИИ (16, 20, 24, 33 Мгц)
* ДОПОЛНИТЕЛЬНЫЕ ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА: МНОГОКАНАЛЬНЫЙ АЦП, КОНТРОЛЛЕР ШИМ, СТОРОЖЕВОЙ ТАЙМЕР, КОНТРОЛЛЕРЫ ИНТЕРФЕЙСОВ I2C И CAN, АНАЛОГОВЫЕ И ЦИФРОВЫЕ КОМПАРАТОРЫ, ДОПОЛНИТЕЛЬНЫЕ ПОРТЫ ВВОДА-ВЫВОДА