

**АНАЛОГОВЫЕ ПЕРИФЕРИЙНЫЕ МОДУЛИ****- 10-разрядный АЦП**

Производительность до 200 тыс. преобразований в секунду.

До 17 или 13 внешних входов (программируются как однофазные или дифференциальные).

Использование в качестве опорного напряжения (VREF) напряжения с внешнего вывода, напряжения внутреннего ИОН или напряжения питания VDD.

Встроенный датчик температуры.

Внешний вход запуска преобразования.

**- Два компаратора****- Внутренний источник опорного напряжения****- Прецизионная схема слежения за напряжением питания/детектор снижения напряжения питания****USB-контроллер**

- Совместимость со спецификацией USB 2.0.

- Полноскоростной (12 Мбит/с) и низкоскоростной (1.5 Мбит/с) режимы работы.

- Интегрированная схема восстановления синхронизации; не требуется внешний кварцевый резонатор для работы в полноскоростном или низкоскоростном режимах.

- Поддержка восьми настраиваемых конечных точек подключения (Endpoint).

- Буфер памяти USB объемом 1 Кбайт.

- Интегрированный приемопередатчик; не требуются внешние резисторы.

**ВСТРОЕННЫЕ СРЕДСТВА ОТЛАДКИ**

- Встроенный отладчик обеспечивает «неразрушающую» внутрисистемную отладку в режиме реального времени (без эмулятора).

- Расстановка точек останова, пошаговая отладка, контроль/модификация памяти и регистров.

- Производительность на уровне эмуляторов с отладочными кристаллами, специальными адаптерами и разъемами.

**РЕГУЛЯТОР НАПРЯЖЕНИЯ**

- Входное напряжение от 4.0В до 5.25В.

**ВЫСОКОПРОИЗВОДИТЕЛЬНОЕ 8051-СОВМЕСТИМОЕ ПРОЦЕССОРНОЕ ЯДРО**

- Конвейерная архитектура; 70% команд выполняются за 1 или 2 системных тактовых цикла.

- Производительность до 25MIPS при тактовой частоте 25MHz.

- Развитая система прерываний.

**ПАМЯТЬ**

- 2304 (1k + 256 + 1k USB FIFO) байт внутреннего ОЗУ.

- 16 Кбайт FLASH-памяти; возможно внутрисистемное программирование FLASH-памяти секторами по 512 байт.

**ЦИФРОВЫЕ ПЕРИФЕРИЙНЫЕ МОДУЛИ**

- 25/21 портов ввода/вывода с допустимым напряжением на выводах до 5В и высоким втекающим током.

- Аппаратные последовательные интерфейсы SPI, УАПП и SMBus.

- Четыре 16-разрядных таймера/счетчика общего назначения.

- Программируемый массив 16-разрядных таймеров/счетчиков (ПМС) с пятью модулями захвата/сравнения.

- Режим генератора реального времени (используются внешний источник тактирования и Таймер или ПМС).

**ИСТОЧНИКИ ТАКТИРОВАНИЯ**

- Внутренний генератор с точностью 0.25% и возможностью восстановления синхронизации. Поддерживает все режимы USB и УАПП.

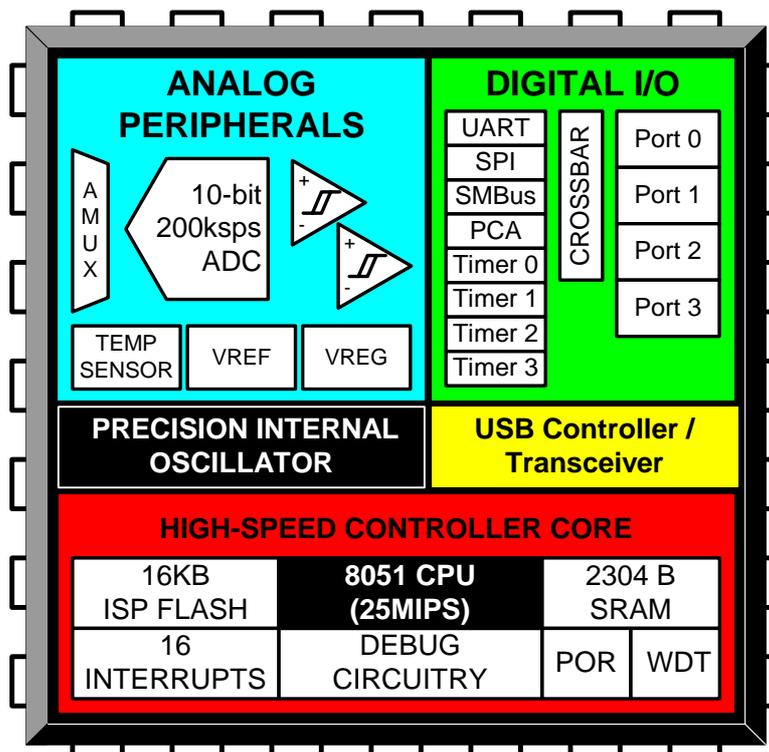
- Внешний генератор: кварцевый, RC-, C-, или счетчик (1-выв. или 2-выв. режимы).

- Возможность переключения между источниками тактирования «на лету» (полезно в критичных к энергопотреблению приложениях).

**КОРПУСА:**

32-выв. LQFP (C8051F320)

28-выв. MLP (C8051F321).

**РАБОЧАЯ ТЕМПЕРАТУРА: -40°C...+85°C**

## Примечания

## СОДЕРЖАНИЕ

<b>1. КРАТКИЙ ОБЗОР.....</b>	<b>17</b>
1.1. Процессорное ядро CIP-51™.....	20
1.1.1. Полная совместимость со стандартом 8051.....	20
1.1.2. Улучшенная производительность.....	20
1.1.3. Дополнительные функции.....	21
1.2. Встроенная память.....	22
1.3. USB-контроллер.....	23
1.4. Регулятор напряжения.....	23
1.5. Встроенные средства отладки.....	24
1.6. Программируемые цифровые порты ввода/вывода и матрица соединений.....	25
1.7. Последовательные порты.....	25
1.8. Программируемый массив счетчиков (ПМС).....	26
1.9. 10-разрядный аналого-цифровой преобразователь.....	27
1.10. Компараторы.....	28
<b>2. ПРЕДЕЛЬНО ДОПУСТИМЫЕ ПАРАМЕТРЫ.....</b>	<b>29</b>
<b>3. ОСНОВНЫЕ ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ.....</b>	<b>30</b>
<b>4. ОПИСАНИЕ КОРПУСОВ И ВЫВОДОВ.....</b>	<b>31</b>
<b>5. 10-разрядный АЦП.....</b>	<b>39</b>
5.1. Аналоговый мультиплексор.....	40
5.2. Датчик температуры.....	41
5.3. Режимы работы АЦП.....	42
5.3.1. Запуск преобразования.....	42
5.3.2. Режимы слежения.....	43
5.3.3. Время установления.....	44
5.4. Программируемый детектор диапазона АЦП.....	50
5.4.1. Детектор диапазона в однофазном режиме.....	52
5.4.2. Детектор диапазона в дифференциальном режиме.....	53
<b>6. ИСТОЧНИК ОПОРНОГО НАПРЯЖЕНИЯ.....</b>	<b>55</b>
<b>7. КОМПАРАТОРЫ.....</b>	<b>57</b>
<b>8. РЕГУЛЯТОР НАПРЯЖЕНИЯ (REG0).....</b>	<b>67</b>
8.1. Выбор режима регулятора.....	68
8.2. Детектирование VBUS.....	69
<b>9. ПРОЦЕССОРНОЕ ЯДРО CIP-51.....</b>	<b>73</b>
9.1. Система команд.....	75
9.1.1. Команды и тактирование.....	75
9.1.2. Команда MOVX и память программ.....	75
9.2. Организация памяти.....	79
9.2.1. Память программ.....	79
9.2.2. Память данных.....	80
9.2.3. Регистры общего назначения.....	80
9.2.4. Ячейки памяти с битовой адресацией.....	80
9.2.5. Стек.....	80
9.2.6. Регистры специального назначения.....	81
9.2.7. Описание регистров.....	84

9.3. Обработка прерываний.....	87
9.3.1. Источники и векторы прерываний.....	87
9.3.2. Внешние прерывания.....	88
9.3.3. Приоритеты прерываний.....	88
9.3.4. Задержка обработки прерываний.....	88
9.3.5. Описание регистров прерываний.....	90
9.4. Режимы управления электропитанием.....	96
9.4.1. Режим ожидания.....	96
9.4.2. Режим остановки.....	96
<b>10. ИСТОЧНИКИ СБРОСА.....</b>	<b>99</b>
10.1. Сброс при включении питания.....	100
10.2. Сброс при исчезновении питания.....	101
10.3. Внешний сброс.....	102
10.4. Сброс от детектора исчезновения тактирования.....	102
10.5. Сброс от Компаратора 0.....	102
10.6. Сброс от сторожевого таймера ПМС.....	102
10.7. Сброс от контроллера Flash-памяти.....	102
10.8. Программный сброс.....	103
10.9. Сброс от USB-контроллера.....	103
<b>11. FLASH-ПАМЯТЬ.....</b>	<b>107</b>
11.1. Программирование FLASH-памяти.....	107
11.1.1. Блокировка Flash-памяти и ключевой код.....	107
11.1.2. Процедура стирания Flash-памяти.....	107
11.1.3. Процедура записи Flash-памяти.....	108
11.2. Долговременное хранение данных.....	109
11.3. Защита FLASH-памяти.....	109
<b>12. ВНЕШНЕЕ ОЗУ.....</b>	<b>113</b>
12.1. Доступ к пользовательской памяти XRAM.....	113
12.2. Доступ к USB FIFO.....	114
<b>13. ГЕНЕРАТОРЫ.....</b>	<b>117</b>
13.1. Программируемый внутренний генератор.....	117
13.1.1. Программирование внутреннего генератора в МК C8051F320/1.....	118
13.1.2. Экономичный режим внутреннего генератора.....	118
13.2. Схема возбуждения внешнего генератора.....	120
13.2.1. Тактирование таймеров непосредственно от внешнего генератора.....	120
13.2.2. Пример использования внешнего резонатора.....	120
13.2.3. Пример использования RC-генератора.....	121
13.2.4. Пример использования внешнего генератора с конденсатором.....	121
13.3. 4-кратный множитель тактовой частоты.....	123
13.4. Выбор источника сигнала тактирования системы и сигнала тактирования USB.....	124
<b>14. ПОРТ ВВОДА/ВЫВОДА.....</b>	<b>127</b>
14.1. Приоритетный декодер матрицы.....	129
14.2. Инициализация порта ввода/вывода.....	131
14.3. Порт ввода/вывода общего назначения.....	134
<b>15. USB-КОНТРОЛЛЕР (USB0).....</b>	<b>143</b>
15.1. Адресация Endpoint.....	144
15.2. Приемопередатчик USB.....	144
15.3. Доступ к регистрам USB-контроллера.....	146



---

15.4. Настройка тактирования USB-контроллера.....	150
15.5. Управление буфером FIFO.....	151
15.5.1. Распределение FIFO.....	151
15.5.2. Двойная буферизация FIFO.....	151
15.5.3. Доступ к FIFO.....	152
15.6. Адресация устройства на шине USB.....	153
15.7. Настройка и управление USB0.....	154
15.8. Прерывания.....	157
15.9. Последовательный интерфейсный модуль.....	161
15.10. Endpoint0.....	161
15.10.1. Транзакции SETUP.....	162
15.10.2. Транзакции IN.....	162
15.10.3. Транзакции OUT.....	163
15.11. Настройка Endpoint1-3.....	166
15.12. Управление Endpoint1-3 IN.....	166
15.12.1. Прерывистый и блочный режимы Endpoint1-3 IN.....	166
15.12.2. Изохронный режим Endpoint1-3 IN.....	167
15.13. Управление Endpoint1-3 OUT.....	170
15.13.1. Прерывистый и блочный режимы Endpoint1-3 OUT.....	170
15.13.2. Изохронный режим Endpoint1-3 OUT.....	170
<b>16. SMBus.....</b>	<b>175</b>
16.1. Техническая документация.....	176
16.2. Подключение к шине SMBus.....	176
16.3. Протокол SMBus.....	177
16.3.1. Арбитраж.....	177
16.3.2. Растягивание тактового сигнала.....	177
16.3.3. Таймаут низкого уровня на линии SCL.....	178
16.3.4. Таймаут высокого уровня на линии SCL (шина SMBus свободна).....	178
16.4. Использование модуля SMBus.....	179
16.4.1. Регистр конфигурации SMBus.....	180
16.4.2. Регистр управления SMB0CN.....	183
16.4.3. Регистр данных.....	186
16.5. Режимы работы модуля SMBus.....	187
19.3.1. Режим ведущего передатчика.....	187
19.3.2. Режим ведущего приемника.....	188
19.3.3. Режим ведомого передатчика.....	189
19.3.4. режим ведомого приемника.....	190
16.6. Декодирование состояний SMBus.....	191
<b>17. УАППО.....</b>	<b>193</b>
17.1. Усовершенствованный режим генерации скорости передачи.....	194
17.2. Режимы работы УАППО.....	195
17.2.1. 8-разрядный УАППО.....	195
17.2.2. 9-разрядный УАППО.....	196
17.3. Поддержка связи с несколькими МК.....	197
<b>18. МОДУЛЬ SPI0.....</b>	<b>203</b>
18.1. Описание сигналов шины SPI.....	204
18.1.1. Выход ведущего, вход ведомого (MOSI).....	204
18.1.2. Вход ведущего, выход ведомого (MISO).....	204

---

18.1.3. Тактовые импульсы (SCK).....	204
18.1.4. Выбор ведомого (NSS).....	204
18.2. Функционирование SPI0 в режиме ведущего.....	205
18.3. Функционирование SPI0 в режиме ведомого.....	207
18.4. Источники прерываний модуля SPI0.....	207
18.5. Тактирование.....	208
18.6. Регистры специального назначения модуля SPI.....	210
<b>19. ТАЙМЕРЫ.....</b>	<b>217</b>
19.1. Таймер 0 и Таймер 1.....	217
19.1.1. Режим 0: 13-разрядный таймер/счетчик.....	217
19.1.2. Режим 1: 16-разрядный таймер/счетчик.....	218
19.1.3. Режим 2: 8-разрядный таймер/счетчик с автоперезагрузкой.....	219
19.1.4. Режим 3: два 8-разрядных таймера/счетчика (только Таймер 0).....	220
19.2. Таймер 2.....	225
19.2.1. 16-разрядный таймер с автоперезагрузкой.....	225
19.2.2. 8-разрядные таймеры с автоперезагрузкой.....	226
19.2.3. Режим захвата USB Start-of-Frame.....	227
19.3. Таймер 3.....	230
19.3.1. 16-разрядный таймер с автоперезагрузкой.....	230
19.3.2. 8-разрядные таймеры с автоперезагрузкой.....	231
19.3.3. Режим захвата USB Start-of-Frame.....	232
<b>20. ПРОГРАММИРУЕМЫЙ МАССИВ СЧЕТЧИКОВ.....</b>	<b>235</b>
20.1. Таймер/Счетчик модуля ПМС.....	236
20.2. Модули захват/сравнение.....	237
20.2.1. Режим захвата по фронту сигнала.....	238
20.2.2. Режим программного таймера (сравнения).....	239
20.2.3. Режим высокоскоростного выхода.....	240
20.2.4. Режим выхода заданной частоты.....	241
20.2.5. Режим 8-разрядного широтно-импульсного модулятора.....	242
20.2.6. Режим 16-разрядного широтно-импульсного модулятора.....	244
20.3. Режим сторожевого таймера.....	246
20.3.1. Функционирование сторожевого таймера.....	246
20.3.2. Использование сторожевого таймера.....	247
20.4. Описание регистров модуля ПМС.....	248
<b>21. ИНТЕРФЕЙС C2.....</b>	<b>253</b>
21.1. Регистры интерфейса C2.....	253
21.2. Разделение выводов интерфейса C2.....	255

**Примечания**

## Примечания

---

**ПЕРЕЧЕНЬ РИСУНКОВ И ТАБЛИЦ**

<b>1. КРАТКИЙ ОБЗОР.....</b>	<b>17</b>
Таблица 1.1. Сравнительная характеристика микроконтроллеров.....	17
Рисунок 1.1. Структурная схема C8051F320.....	18
Рисунок 1.2. Структурная схема C8051F321.....	19
Рисунок 1.3. Максимальная производительность различных микроконтроллеров.....	20
Рисунок 1.4. Структурная схема модуля тактирования и сброса.....	21
Рисунок 1.5. Карта распределения памяти.....	22
Рисунок 1.6. Структурная схема USB-контроллера.....	23
Рисунок 1.7. Модель отладки.....	24
Рисунок 1.8. Структурная схема цифровой матрицы.....	25
Рисунок 1.9. Структурная схема модуля ПМС.....	26
Рисунок 1.10. Структурная схема 10-разрядного АЦП.....	27
Рисунок 1.11. Структурная схема Компаратора 0.....	28
<b>2. ПРЕДЕЛЬНО ДОПУСТИМЫЕ ПАРАМЕТРЫ.....</b>	<b>29</b>
<b>3. ОСНОВНЫЕ ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ.....</b>	<b>30</b>
Таблица 3.1. Основные электрические параметры.....	30
<b>4. ОПИСАНИЕ КОРПУСОВ И ВЫВОДОВ.....</b>	<b>31</b>
Таблица 4.1. Описание выводов для C8051F320/1.....	31
Рисунок 4.1. Цоколевка корпуса LQFP-32 (вид сверху).....	33
Рисунок 4.2. Чертеж корпуса LQFP-32.....	34
Таблица 4.2. Размеры корпуса LQFP-32.....	34
Рисунок 4.3. Цоколевка корпуса MLP-28 (вид сверху).....	35
Рисунок 4.4. Чертеж корпуса MLP-28.....	36
Таблица 4.3. Размеры корпуса MLP-28.....	36
Рисунок 4.5. Типичная схема посадочного места под корпус MLP-28.....	37
Рисунок 4.6. Трафарет для нанесения припоя для корпуса MLP-28.....	38
<b>5. 10-разрядный АЦП (АЦПО).....</b>	<b>39</b>
Рисунок 5.1. Функциональная схема АЦПО.....	39
Рисунок 5.2. Передаточная характеристика датчика температуры.....	41
Рисунок 5.3. Временные диаграммы процесса преобразования 10-разр. АЦП.....	43
Рисунок 5.4. Эквивалентные схемы входов АЦПО.....	44
Рисунок 5.5. AMXOP: Регистр выбора положительного канала AMUX0.....	45
Рисунок 5.6. AMXON: Регистр выбора отрицательного канала AMUX0.....	46
Рисунок 5.7. ADC0CF: Регистр конфигурации АЦПО.....	47
Рисунок 5.8. ADC0H: Регистр старшего байта слова данных АЦПО.....	47
Рисунок 5.9. ADC0L: Регистр младшего байта слова данных АЦПО.....	48
Рисунок 5.10. ADC0CN: Регистр управления АЦПО.....	49
Рисунок 5.11. ADC0GTH: Регистр старшего байта нижней границы диапазона.....	50
Рисунок 5.12. ADC0GTL: Регистр младшего байта нижней границы диапазона.....	50
Рисунок 5.13. ADC0LTH: Регистр старшего байта верхней границы диапазона.....	51
Рисунок 5.14. ADC0LTL: Регистр младшего байта верхней границы диапазона.....	51
Рисунок 5.15. Пример использования детектора диапазона 10-разрядного АЦПО (данные выровнены вправо, вход однофазный).....	52
Рисунок 5.16. Пример использования детектора диапазона 10-разрядного	

АЦПО (данные выровнены вправо, вход дифференциальный).....	52
Рисунок 5.17. Пример использования детектора диапазона 10-разрядного АЦПО (данные выровнены влево, вход однофазный).....	53
Рисунок 5.18. Пример использования детектора диапазона 10-разрядного АЦПО (данные выровнены влево, вход дифференциальный).....	53
Таблица 5.1. Электрические характеристики 10-разрядного АЦПО.....	54
<b>6. ИСТОЧНИК ОПОРНОГО НАПРЯЖЕНИЯ.....</b>	<b>55</b>
Рисунок 6.1. Функциональная схема источника опорного напряжения.....	55
Рисунок 6.2. REF0CN: Регистр управления источника опорного напряжения.....	56
Таблица 6.1. Электрические характеристики источника опорного напряжения.....	56
<b>7. КОМПАРАТОРЫ.....</b>	<b>57</b>
Рисунок 7.1. Функциональная схема Компаратора 0.....	57
Рисунок 7.2. Функциональная схема Компаратора 1.....	58
Рисунок 7.3. Гистерезис компаратора.....	59
Рисунок 7.4. CPT0CN: Регистр управления Компаратора 0.....	60
Рисунок 7.5. CPT0MX: Регистр выбора канала мультимплексора Компаратора 0.....	61
Рисунок 7.6. CPT0MD: Регистр выбора режима Компаратора 0.....	62
Рисунок 7.7. CPT1CN: Регистр управления Компаратора 1.....	63
Рисунок 7.8. CPT1MX: Регистр выбора канала мультимплексора Компаратора 1.....	64
Рисунок 7.9. CPT1MD: Регистр выбора режима Компаратора 1.....	65
Таблица 7.1. Электрические характеристики компараторов.....	66
<b>8. РЕГУЛЯТОР НАПРЯЖЕНИЯ (REG0).....</b>	<b>67</b>
Таблица 8.1. Электрические характеристики регулятора напряжения.....	69
Рисунок 8.1. Конфигурация REG0: Питание от шины USB.....	70
Рисунок 8.2. Конфигурация REG0: Собственный (не USB) источник питания.....	70
Рисунок 8.3. Конфигурация REG0: Собственный (не USB) источник питания, регулятор отключен.....	71
Рисунок 8.4. Конфигурация REG0: Шина USB не подключена.....	71
Рисунок 8.5. REG0CN: Регистр управления регулятором напряжения.....	72
<b>9. ПРОЦЕССОРНОЕ ЯДРО CIP-51.....</b>	<b>73</b>
Рисунок 9.1. Структурная схема CIP-51.....	73
Таблица 9.1. Система команд CIP-51.....	76
Рисунок 9.2. Карта распределения памяти.....	79
Таблица 9.2. Распределение регистров специального назначения в памяти.....	81
Таблица 9.3. Регистры специального назначения.....	81
Рисунок 9.3. DPL: Младший байт указателя данных.....	84
Рисунок 9.4. DPH: Старший байт указателя данных.....	84
Рисунок 9.5. SP: Указатель стека.....	85
Рисунок 9.6. PSW: Слово состояния программы.....	85
Рисунок 9.7. ACC: Аккумулятор.....	86
Рисунок 9.8. В: Регистр В.....	86
Таблица 9.4. Источники прерываний.....	89
Рисунок 9.9. IE: Регистр разрешения прерываний.....	90
Рисунок 9.10. IP: Регистр приоритетов прерываний.....	91
Рисунок 9.11. EIE1: Дополнительный регистр разрешения прерываний 1.....	92
Рисунок 9.12. EIP1: Дополнительный регистр приоритетов прерываний 1.....	93
Рисунок 9.13. EIE2: Дополнительный регистр разрешения прерываний 2.....	94
Рисунок 9.14. EIP2: Дополнительный регистр приоритетов прерываний 2.....	94



---

Рисунок 9.15. IT01CF: Регистр конфигурации INT0/INT1 .....	95
Рисунок 9.16. PCON: Регистр управления электропитанием.....	97
<b>10. ИСТОЧНИКИ СБРОСА.....</b>	<b>99</b>
Рисунок 10.1. Структурная схема источников сброса.....	99
Рисунок 10.2. Временная диаграмма работы схемы слежения за напряжением питания...100	
Рисунок 10.3. VDM0CN: Регистр управления схемой слежения за напряжением питания.....	101
Рисунок 10.4. RSTSRC: Регистр источников сброса.....	104
Таблица 10.1. Электрические параметры источников сброса.....	105
<b>11. FLASH ПАМЯТЬ.....</b>	<b>107</b>
Таблица 11.1. Электрические параметры FLASH-памяти.....	108
Рисунок 11.1. Карта распределения и байты защиты FLASH-памяти программ.....	110
Рисунок 11.2. PSCTL: Регистр управления чтением/записью памяти программ.....	110
Рисунок 11.3. FLKEY: Регистр блокировки и ключевого кода FLASH-памяти .....	111
Рисунок 11.4. FLSCL: Регистр управления контроллером FLASH-памяти.....	111
<b>12. ВНЕШНЕЕ ОЗУ.....</b>	<b>113</b>
Рисунок 12.1. Карта распределения памяти внешнего ОЗУ.....	113
Рисунок 12.2. Расширенное представление памяти XRAM.....	114
Рисунок 12.3. EMIOCN: Регистр управления интерфейсом внешней памяти.....	115
<b>13. ГЕНЕРАТОРЫ.....</b>	<b>117</b>
Рисунок 13.1. Структурная схема генератора.....	117
Рисунок 13.2. OSCICN: Регистр управления внутренним генератором.....	119
Рисунок 13.3. OSCICL: Регистр калибровки внутреннего генератора.....	119
Рисунок 13.4. OSCXCN: Регистр управления внешним генератором.....	122
Рисунок 13.5. CLKMUL: Регистр управления множителем тактовой частоты.....	123
Таблица 13.1. Типичные настройки для высокоскоростного режима USB.....	124
Таблица 13.2. Типичные настройки для низкоскоростного режима USB.....	124
Рисунок 13.6. CLKSEL: Регистр выбора тактового генератора.....	125
Таблица 13.3. Электрические параметры внутреннего генератора.....	126
<b>14. ПОРТ ВВОДА/ВЫВОДА.....</b>	<b>127</b>
Рисунок 14.1. Функциональная схема порта ввода/вывода.....	127
Рисунок 14.2. Структурная схема ячейки порта ввода/вывода.....	128
Рисунок 14.3. Приоритетный декодер матрицы без пропуска каких-либо выводов.....	129
Рисунок 14.4. Приоритетный декодер матрицы с пропуском выводов подключения кварцевого резонатора.....	130
Рисунок 14.5. XBR0: Регистр 0 матрицы порта ввода/вывода.....	132
Рисунок 14.6. XBR1: Регистр 1 матрицы порта ввода/вывода.....	133
Рисунок 14.7. P0: Регистр данных Порта 0.....	135
Рисунок 14.8. P0MDIN: Регистр настройки входов Порта 0.....	135
Рисунок 14.9. P0MDOUT: Регистр настройки выходов Порта 0.....	136
Рисунок 14.10. P0SKIP: Регистр выбора выводов Порта 0, пропускаемых матрицей.....	136
Рисунок 14.11. P1: Регистр данных Порта 1.....	137
Рисунок 14.12. P1MDIN: Регистр настройки входов Порта 1.....	137
Рисунок 14.13. P1MDOUT: Регистр настройки выходов Порта 1.....	137
Рисунок 14.14. P1SKIP: Регистр выбора выводов Порта 1, пропускаемых матрицей.....	138
Рисунок 14.15. P2: Регистр данных Порта 2.....	139
Рисунок 14.16. P2MDIN: Регистр настройки входов Порта 2.....	139

---

Рисунок 14.17. P2MDOUT: Регистр настройки выходов Порта 2.....	140
Рисунок 14.18. P2SKIP: Регистр выбора выводов Порта 2, пропускаемых матрицей.....	140
Рисунок 14.19. P3: Регистр данных Порта 3.....	141
Рисунок 14.20. P3MDIN: Регистр настройки входов Порта 3.....	141
Рисунок 14.21. P3MDOUT: Регистр настройки выходов Порта 3.....	142
Таблица 14.1. Электрические характеристики порта ввода/вывода.....	142
<b>15. USB-КОНТРОЛЛЕР (USB0).....</b>	<b>143</b>
Рисунок 15.1. Структурная схема USB0.....	143
Таблица 15.1. Схема адресации Endpoint.....	144
Рисунок 15.2. USB0XCEN: Регистр управления приемопередатчиком USB0.....	145
Рисунок 15.3. Схема доступа к регистрам USB0.....	146
Рисунок 15.4. USB0ADR: Регистр косвенного адреса USB0.....	147
Рисунок 15.5. USB0DAT: Регистр данных USB0.....	148
Рисунок 15.6. INDEX: Регистр индекса Endpoint USB0 (USB регистр).....	148
Таблица 15.2. Регистры управления USB0.....	149
Рисунок 15.7. CLKREC: Регистр управления восстановлением синхронизации (USB регистр).....	150
Рисунок 15.8. Распределение USB FIFO.....	151
Таблица 15.3. Конфигурации FIFO.....	152
Рисунок 15.9. FIFOEN: Регистры доступа к буферам FIFO Endpoint USB0 (USB регистры).....	152
Рисунок 15.10. FADDR: Адрес USB0 на шине USB (USB регистр).....	153
Рисунок 15.11. POWER: Питание USB0 (USB регистр).....	155
Рисунок 15.12. FRAMEL: Младший байт номера пакета USB0 (USB регистр).....	156
Рисунок 15.13. FRAMEH: Старший байт номера пакета USB0 (USB регистр).....	156
Рисунок 15.14. IN1INT: Регистр флагов прерываний от IN Endpoint USB0 (USB регистр).....	157
Рисунок 15.15. OUT1INT: Регистр флагов прерываний от OUT Endpoint USB0 (USB регистр).....	158
Рисунок 15.16. CMINT: Регистр общих прерываний от USB0 (USB регистр).....	159
Рисунок 15.17. IN1IE: Регистр разрешения прерываний от IN Endpoint USB0 (USB регистр).....	160
Рисунок 15.18. OUT1IE: Регистр разрешения прерываний от OUT Endpoint USB0 (USB регистр).....	160
Рисунок 15.19. CMIE: Регистр разрешения общих прерываний от USB0 (USB регистр).....	161
Рисунок 15.20. E0CSR: Регистр управления Endpoint0 USB0 (USB регистр).....	164
Рисунок 15.21. E0CSR: Регистр счетчика данных Endpoint0 USB0 (USB регистр).....	165
Рисунок 15.22. E1NCSRL: Старший байт регистра управления IN Endpoint USB0 (USB регистр).....	168
Рисунок 15.23. E1NCSRH: Младший байт регистра управления IN Endpoint USB0 (USB регистр).....	169
Рисунок 15.24. E0OUTCSRL: Старший байт регистра управления OUT Endpoint USB0 (USB регистр).....	171
Рисунок 15.25. E0OUTCSRH: Младший байт регистра управления OUT Endpoint USB0 (USB регистр).....	172
Рисунок 15.26. E0OUTCNTL: Младший байт счетчика байт данных OUT Endpoint USB0 (USB регистр).....	172
Рисунок 15.27. E0OUTCNTH: Старший байт счетчика байт данных	

---

OUT Endpoint USB0 (USB регистр).....	172
Таблица 15.4. Электрические параметры приемопередатчика USB.....	173
<b>16. SMBUS.....</b>	<b>175</b>
Рисунок 16.1. Структурная схема модуля SMBus.....	175
Рисунок 16.2. Подключение к шине SMBus .....	176
Рисунок 16.3. Формат сообщения SMBus.....	177
Таблица 16.1. Выбор источника тактирования SMBus.....	180
Рисунок 16.4. Формирование сигнала SCL.....	181
Таблица 16.2. Минимальные значения времени установления и времени удержания сигнала SDA.....	181
Рисунок 16.5. SMB0CF: Регистр настройки модуля SMBus0.....	182
Рисунок 16.6. SMB0CN: Регистр управления модуля SMBus0.....	184
Таблица 16.3. События, вызывающие аппаратную установку/сброс битов регистра SMB0CN.....	185
Рисунок 16.7. SMB0DAT: Регистр данных модуля SMBus0.....	186
Рисунок 16.8. Передача данных в режиме ведущего.....	187
Рисунок 16.9. Прием данных в режиме ведущего.....	188
Рисунок 16.10. Прием данных в режиме ведомого.....	189
Рисунок 16.11. Передача данных в режиме ведомого.....	190
Таблица 16.4. Декодирование состояний SMBus.....	191
<b>17. УАППО.....</b>	<b>193</b>
Рисунок 17.1. Структурная схема УАППО.....	193
Рисунок 17.2. Логика генератора скорости передачи данных УАППО.....	194
Рисунок 17.3. Примеры использования УАППО.....	195
Рисунок 17.4. Временные диаграммы 8-разр. УАППО.....	195
Рисунок 17.5. Временные диаграммы 9-разр. УАППО.....	196
Рисунок 17.6. Пример использования УАППО в многопроцессорном режиме.....	197
Рисунок 17.7. SCON0: Регистр управления УАППО.....	198
Рисунок 17.8. SBUF0: Регистр буфера данных УАППО.....	199
Таблица 17.1. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внутреннего генератора.....	200
Таблица 17.2. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внешнего генератора.....	200
Таблица 17.3. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внешнего генератора.....	201
Таблица 17.4. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внешнего генератора.....	201
Таблица 17.5. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внешнего генератора.....	202
Таблица 17.6. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании системы от внешнего генератора.....	202
<b>18. МОДУЛЬ SPI (SPI0).....</b>	<b>203</b>
Рисунок 18.1. Структурная схема модуля SPI0.....	203
Рисунок 18.2. Схема включения в режиме с несколькими ведущими.....	206
Рисунок 20.3. Схема соединения одного ведущего и одного ведомого с использованием 3-х проводной шины SPI.....	206
Рисунок 18.4. Схема соединения одного ведущего и нескольких ведомых	

---

с использованием 4-х проводной шины SPI.....	206
Рисунок 18.5. Временные диаграммы сигналов данных/тактирования ведущего SPI.....	208
Рисунок 18.6. Временные диаграммы сигналов данных/тактирования ведомого SPI (СКРНА = 0).....	209
Рисунок 18.7. Временные диаграммы сигналов данных/тактирования ведомого SPI (СКРНА = 1).....	209
Рисунок 18.8. SPI0CFG: Регистр конфигурации модуля SPI0.....	210
Рисунок 18.9. SPI0CN: Регистр управления модуля SPI0.....	211
Рисунок 18.10. SPI0CKR: Регистр установки тактовой частоты модуля SPI0.....	212
Рисунок 18.11. SPI0DAT: Регистр данных модуля SPI0.....	213
Рисунок 18.12. Временные диаграммы ведомого SPI (СКРНА = 0).....	214
Рисунок 18.13. Временные диаграммы ведомого SPI (СКРНА = 1).....	214
Рисунок 18.14. Временные диаграммы ведомого SPI (СКРНА = 0).....	214
Рисунок 18.15. Временные диаграммы ведомого SPI (СКРНА = 1).....	214
Таблица 18.1. Временные параметры ведомого SPI.....	216
<b>19. ТАЙМЕРЫ.....</b>	<b>217</b>
Рисунок 19.1. Структурная схема таймера 0 в режиме 0.....	218
Рисунок 19.2. Структурная схема таймера 0 в режиме 2.....	219
Рисунок 19.3. Структурная схема таймера 0 в режиме 3.....	220
Рисунок 19.4. TCON: Регистр управления таймерами 0 и 1.....	221
Рисунок 19.5. TMOD: Регистр режима таймеров 0 и 1.....	222
Рисунок 19.6. SKCON: Регистр управления тактированием таймеров.....	223
Рисунок 19.7. TL0: Младший байт таймера 0.....	224
Рисунок 19.8. TL1: Младший байт таймера 1.....	224
Рисунок 19.9. TH0: Старший байт таймера 0.....	224
Рисунок 19.10. TH1: Старший байт таймера 1.....	224
Рисунок 19.11. Структурная схема Таймера 2 в 16-разрядном режиме.....	225
Рисунок 19.12. Структурная схема Таймера 2 в 8-разрядном режиме.....	226
Рисунок 19.13. Таймер 2 в режиме захвата SOF (T2SPLIT = '0').....	227
Рисунок 19.14. Таймер 2 в режиме захвата SOF (T2SPLIT = '1').....	227
Рисунок 19.15. TMR2CN: Регистр управления Таймера 2.....	228
Рисунок 19.16. TMR2RLL: Младший байт регистра перезагрузки Таймера 2.....	229
Рисунок 19.17. TMR2RLH: Старший байт регистра перезагрузки Таймера 2.....	229
Рисунок 19.18. TMR2L: Младший байт Таймера 2.....	229
Рисунок 19.19. TMR2H: Старший байт Таймера 2.....	229
Рисунок 19.20. Структурная схема Таймера 3 в 16-разрядном режиме.....	230
Рисунок 19.21. Структурная схема Таймера 3 в 8-разрядном режиме.....	231
Рисунок 19.22. Таймер 3 в режиме захвата SOF (T3SPLIT = '0').....	232
Рисунок 19.23. Таймер 3 в режиме захвата SOF (T3SPLIT = '1').....	232
Рисунок 19.24. TMR3CN: Регистр управления Таймера 3.....	233
Рисунок 19.25. TMR3RLL: Младший байт регистра перезагрузки Таймера 3.....	234
Рисунок 19.26. TMR3RLH: Старший байт регистра перезагрузки Таймера 3.....	234
Рисунок 19.27. TMR3L: Младший байт Таймера 3.....	234
Рисунок 19.28. TMR3H: Старший байт Таймера 3.....	234
<b>20. ПРОГРАММИРУЕМЫЙ МАССИВ СЧЕТЧИКОВ.....</b>	<b>235</b>
Рисунок 20.1. Структурная схема ПМС.....	235
Рисунок 20.2. Структурная схема таймера/счетчика ПМС.....	236
Таблица 20.1. Выбор тактового сигнала для ПМС.....	236
Рисунок 20.3. Схема формирования прерывания от ПМС.....	237

---

Таблица 20.2. Настройка модулей захват/сравнение в регистре PCA0CPM.....	237
Рисунок 20.4 Структурная схема ПМС в режиме захвата.....	238
Рисунок 20.5. Структурная схема ПМС в режиме программного таймера.....	239
Рисунок 20.6. Структурная схема ПМС в режиме высокоскоростного выхода.....	240
Рисунок 20.7. Структурная схема ПМС в режиме выхода заданной частоты.....	241
Рисунок 20.8. Структурная схема ПМС в 8-разр. режиме ШИМ.....	243
Рисунок 20.9. Структурная схема ПМС в 16-разр. режиме ШИМ.....	244
Рисунок 20.10. Модуль 4 захвата/сравнения ПМС с включенным сторожевым таймером.....	246
Таблица 20.3. Значения таймаута сторожевого таймера.....	247
Рисунок 20.11. PCA0CN: Регистр управления ПМС.....	248
Рисунок 20.12. PCA0MD: Регистр режима ПМС.....	249
Рисунок 20.13. PCA0CPMn: Регистры управления модулями захват/сравнение.....	250
Рисунок 20.14. PCA0L: Младший байт таймера/счетчика ПМС.....	251
Рисунок 20.15. PCA0H: Старший байт таймера/счетчика ПМС.....	251
Рисунок 20.16. PCA0CPLn: Младший байт модуля захвата ПМС.....	252
Рисунок 20.17. PCA0CPHn: Старший байт модуля захвата ПМС.....	252
<b>21. ИНТЕРФЕЙС C2.....</b>	<b>253</b>
Рисунок 21.1. C2ADD: Регистр адреса C2.....	253
Рисунок 21.2. DEVICEID: Регистр идентификатора C2-устройства .....	253
Рисунок 21.3. REVID: Регистр идентификатора версии C2.....	254
Рисунок 21.4. FPCTL: Регистр управления программированием Flash-памяти посредством интерфейса C2.....	254
Рисунок 21.5. FPDAT: Регистр данных интерфейса C2, используемый при программировании Flash-памяти.....	254
Рисунок 21.6. Разделение выводов интерфейса C2.....	255

## Примечания

## 1. КРАТКИЙ ОБЗОР

Микроконтроллеры C8051F320/1 представляют собой полностью интегрированные на одном кристалле системы для обработки смешанных (аналого-цифровых) сигналов. Отличительные особенности данных МК перечислены ниже. Сравнительная характеристика МК приведена в таблице 1.1.

- Высокопроизводительное микропроцессорное ядро CIP-51 с конвейерной архитектурой, совместимое со стандартом 8051 (максимальная производительность – 25 MIPS).
- Встроенные средства отладки, обеспечивающие внутрисистемную, «неразрушающую» отладку в режиме реального времени.
- USB-контроллер с 8 настраиваемыми конечными точками подключения (Endpoint), встроенным приемопередатчиком и 1k FIFO ОЗУ.
- Регулятор напряжения питания (5В → 3В).
- 10-разрядный 17-канальный АЦП (максимальная производительность – 200 тыс. преобр./сек.) с однофазными/дифференциальными входами и аналоговым мультиплексором.
- Встроенные источник опорного напряжения и датчик температуры.
- Встроенные компараторы напряжения (2).
- Высокоточный программируемый 12 МГц внутренний генератор и 4-кратный множитель тактовой частоты.
- 16 Кбайт встроенной Flash-памяти.
- 2304 байт встроенного ОЗУ (256 + 1k + 1k USB FIFO).
- Аппаратно реализованные последовательные интерфейсы I<sup>2</sup>C/SMBus, расширенные SPI и УАПЧ.
- Четыре 16-разрядных таймера общего назначения.
- Программируемый массив счетчиков/таймеров (ПМС) с пятью модулями захвата/сравнения и сторожевым таймером.
- Встроенные схема сброса по включению питания, схема слежения за напряжением питания и детектор исчезновения сигнала тактирования.
- 25/21 портов ввода/вывода с допустимым напряжением на выводах 5В.

МК C8051F320/1 имеют встроенные схему сброса по включению питания, схему слежения за напряжением питания, регулятор напряжения, сторожевой таймер, тактовый генератор и представляют собой, таким образом, функционально-законченную систему на кристалле. Имеется возможность внутрисхемного программирования Flash-памяти, что обеспечивает долговременное (энергонезависимое) хранение данных, а также позволяет осуществлять обновление программного обеспечения в готовых изделиях. Программа пользователя может полностью управлять всеми периферийными модулями, а также может индивидуально отключить любой из них с целью уменьшения энергопотребления.

Встроенный двухпроводный Silicon Labs Development Interface (интерфейс C2) позволяет производить «неразрушающую» (не используются внутренние ресурсы) внутрисхемную отладку в режиме реального времени, используя МК, установленные в конечное изделие. Средства отладки обеспечивают проверку и модификацию памяти и регистров, расстановку точек останова, пошаговое выполнение программы, а также поддерживают команды запуска и остановки. В процессе отладки с использованием интерфейса C2 все аналоговые и цифровые периферийные модули полностью сохраняют свою работоспособность. Два вывода интерфейса C2 могут использоваться для других пользовательских функций, что позволяет осуществлять внутрисистемную отладку, не занимая для этого отдельные выводы корпуса.

Каждый МК предназначен для работы в промышленном температурном диапазоне (-40°C...+85°C) при напряжении питания 2,7В...3,6В. (Примечание: для взаимодействия по шине USB требуется напряжение 3.0В...3.6В). На порты ввода/вывода и вывод /RST могут быть поданы входные сигналы напряжением до 5В. МК C8051F320/1 выпускаются в 32-выводных корпусах типа LQFP и 28-выводных корпусах типа MLP.

**Таблица 1.1. Сравнительная характеристика микроконтроллеров**

	MIPS (макс.)	FLASH - память	ОЗУ	Калибруемый внутренний генератор	USB	Регулятор напряжения питания	SMBus/I2C	Расширенный SPI	UART	Таймеры (16-разр.)	Программируемый массив счетчиков	Цифровые порты ввода/вывода	10-разр. АЦП (200 тыс. преобразований/сек.)	Датчик температуры	Источник опорного напряжения	Аналоговые компараторы	Тип корпуса
C8051F320	25	16k	2304	√	√	√	√	√	√	4	√	25	√	√	√	2	LQFP-32
C8051F321	25	16k	2304	√	√	√	√	√	√	4	√	21	√	√	√	2	MLP-28

Рисунок 1.1. Структурная схема C8051F320

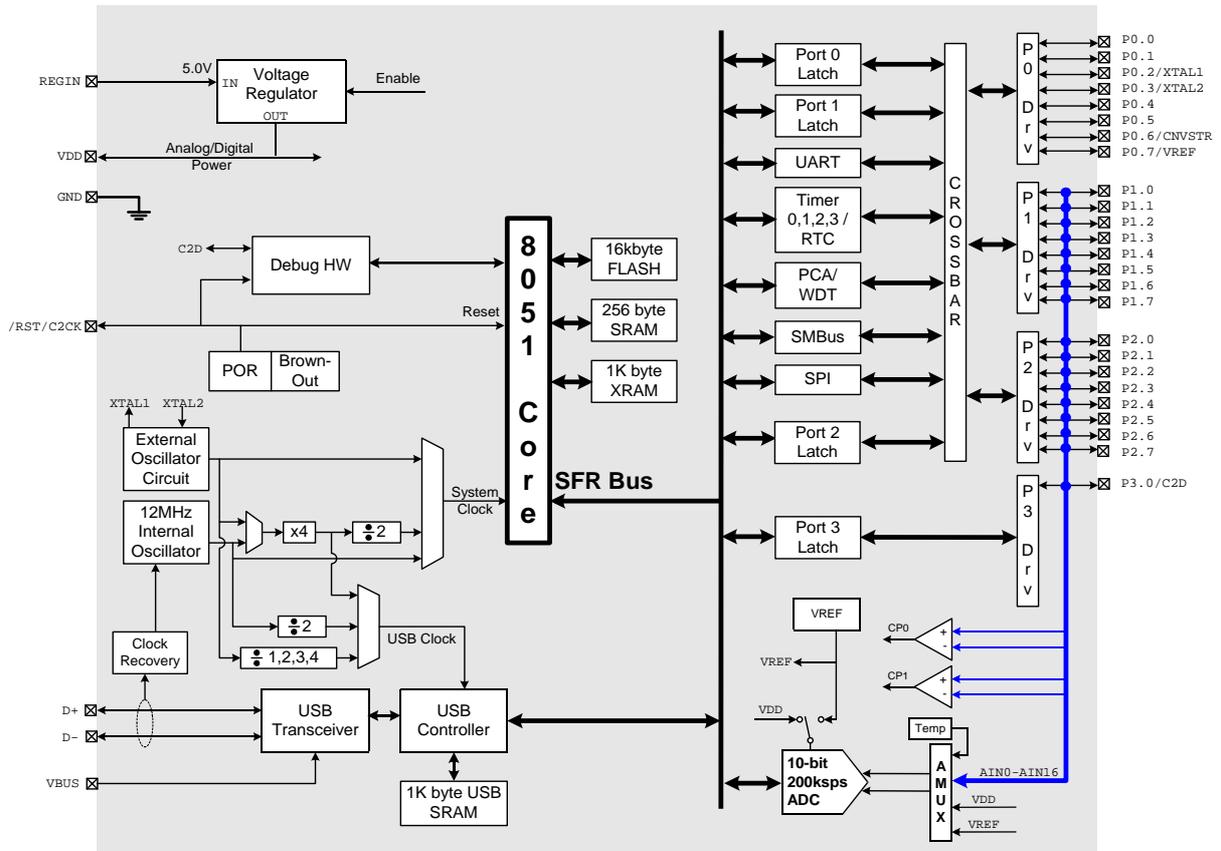
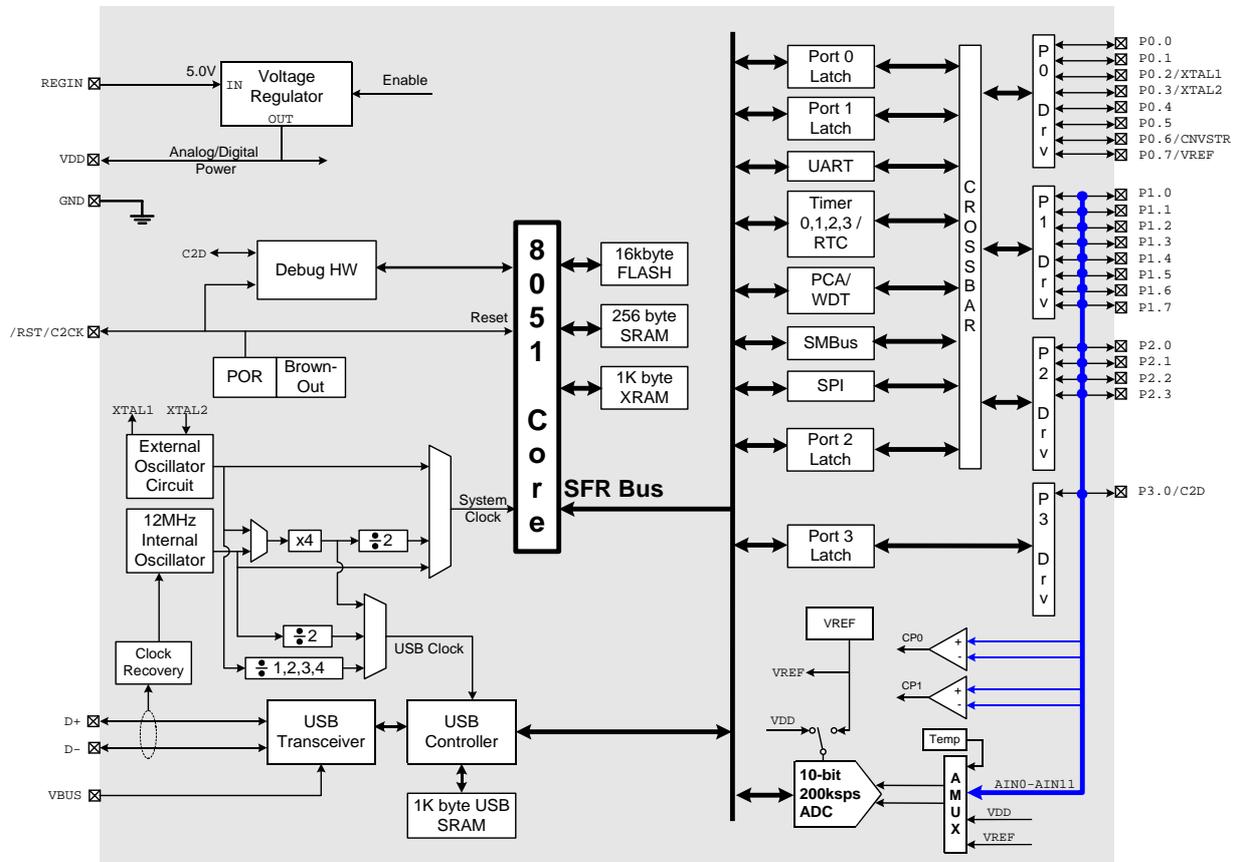


Рисунок 1.2. Структурная схема C8051F321



## 1.1. Процессорное ядро CIP-51™

### 1.1.1. Полностью 8051-совместимая архитектура

МК семейства C8051F320/1 используют разработанное фирмой Silicon Labs' процессорное ядро CIP-51, которое по системе команд полностью совместимо с ядром MCS-51™. Для разработки программного обеспечения могут использоваться стандартные 803х/805х ассемблеры и компиляторы. Ядро содержит всю периферию, соответствующую стандарту 8052, включая четыре 16-разрядных таймера/счетчика, полнодуплексный УАПЧ с усовершенствованным генератором скорости передачи, расширенный SPI интерфейс, 2304 байт встроенного ОЗУ, 128 байт адресного пространства регистров специального назначения, а также 25/21 портов ввода/вывода.

### 1.1.2. Улучшенная производительность

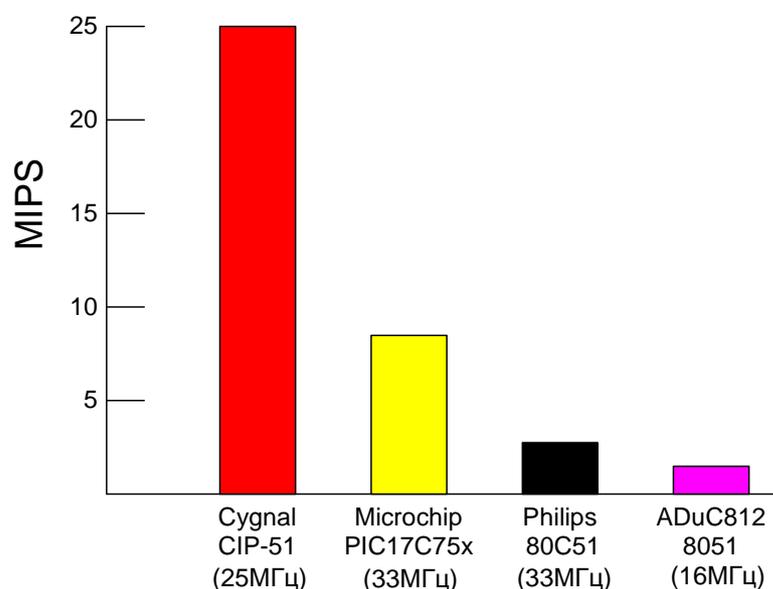
CIP-51 использует конвейерную архитектуру, что существенно повышает скорость выполнения команд по сравнению со стандартной архитектурой 8051. В МК с архитектурой 8051 все команды, кроме MUL и DIV, исполняются за 12 или 24 системных тактовых цикла при максимальной тактовой частоте 12...24 МГц. МК с ядром CIP-51 исполняют 70% своих команд за один или два системных тактовых цикла, и только четыре команды требуют более четырех системных тактовых циклов.

Система команд CIP-51 состоит из 109 команд, которые требуют от одного до восьми системных тактовых циклов:

<b>Количество команд</b>	26	50	5	14	7	3	1	2	1
<b>Количество системных тактовых циклов</b>	1	2	2/3	3	3/4	4	4/5	5	8

При работе на тактовой частоте 25 МГц производительность ядра CIP-51 может достигать 25 MIPS. На рис.1.3 показана пиковая производительность различных 8-разрядных МК, работающих на максимально возможных для них частотах.

**Рисунок 1.3. Максимальная производительность различных микроконтроллеров**



## 1.1.3. Дополнительные функции

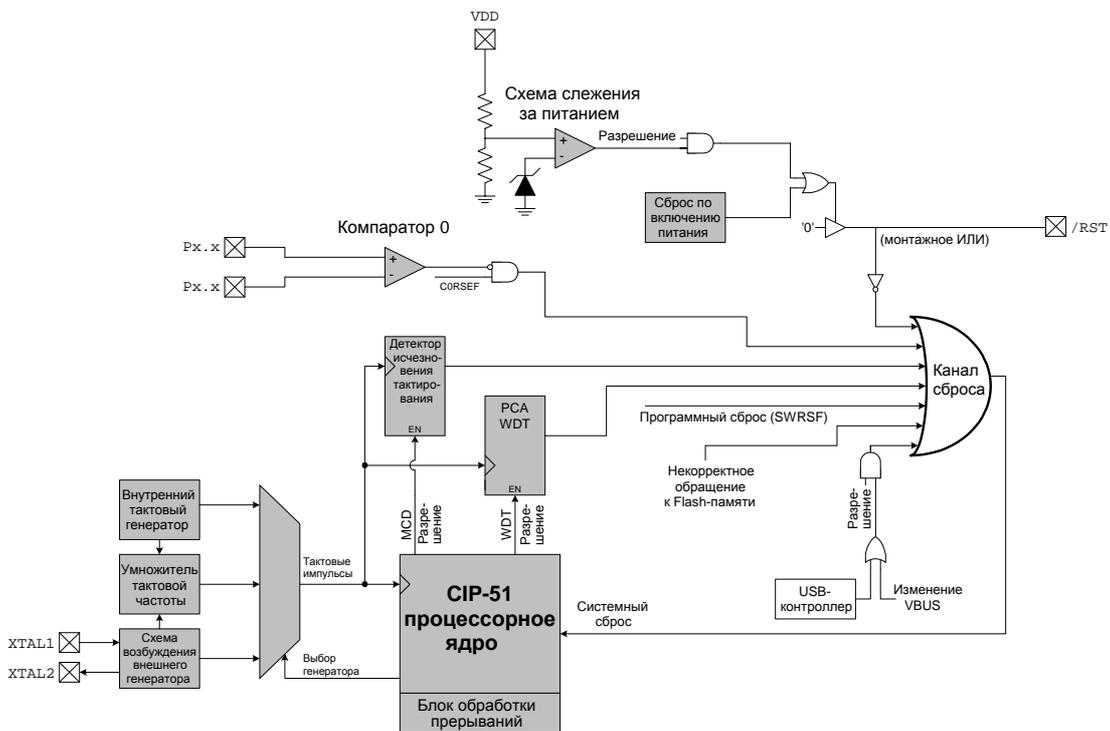
МК семейства C8051F320/1 имеют ряд важных особенностей, которые позволяют улучшить общую производительность и упростить использование МК в конечных приложениях.

16 источников прерываний (8051 имеет 7 источников прерываний) позволяют многочисленным аналоговым и цифровым периферийным модулям прерывать работу МК. Система управления прерываниями требует меньшего вмешательства со стороны программы, что улучшает ее производительность. Дополнительные источники прерываний очень полезны при построении многозадачных систем, работающих в режиме реального времени.

Имеется девять источников сброса: схема сброса по включению питания (POR), встроенная схема слежения за напряжением питания (вызывает сброс в случае, если напряжение питания опускается ниже  $V_{RST}$ , приведенного в таблице 10.1 на стр.105), USB-контроллер, сторожевой таймер, детектор исчезновения тактирования, Компаратор 0, принудительный программный сброс, внешний вывод сброса и схема предотвращения некорректных операций чтения/записи Flash-памяти. Любой источник сброса, за исключением POR, входного вывода сброса и сброса от ошибки обращения к Flash-памяти, может быть отключен программно. Сторожевой таймер может быть включен после сброса типа POR (сброс при включении питания) в процессе инициализации МК.

Внутренний генератор калибруется в процессе изготовления с точностью  $12\text{МГц} \pm 1.5\%$ . Период внутреннего генератора можно запрограммировать с шагом  $\sim 0.25\%$ . Внутренний генератор можно использовать в качестве источника тактирования USB в низкоскоростном режиме. Механизм восстановления синхронизации позволяет использовать внутренний генератор с 4-кратным множителем тактовой частоты в качестве источника тактирования USB в высокоскоростном режиме. Внешние генераторы также могут использоваться совместно с 4-кратным множителем тактовой частоты. Кроме этого, имеется схема возбуждения внешнего генератора, которая позволяет использовать для генерации системных тактовых импульсов внешний кварцевый или керамический резонатор, конденсатор, RC-цепочку или внешний КМОП источник импульсов. МК можно настроить таким образом, чтобы в качестве системного тактового сигнала использовался выходной сигнал внутреннего генератора, внешнего генератора или умножителя тактовой частоты, деленный по частоте на два. При необходимости можно переключать источник тактирования системы «на лету». В приложениях с пониженным энергопотреблением крайне полезным может быть режим работы МК с медленным (мало потребляющим) внешним кварцевым генератором с периодическим переключением (при необходимости) на быстрый внутренний генератор.

**Рисунок 1.4. Структурная схема модуля тактирования и сброса**

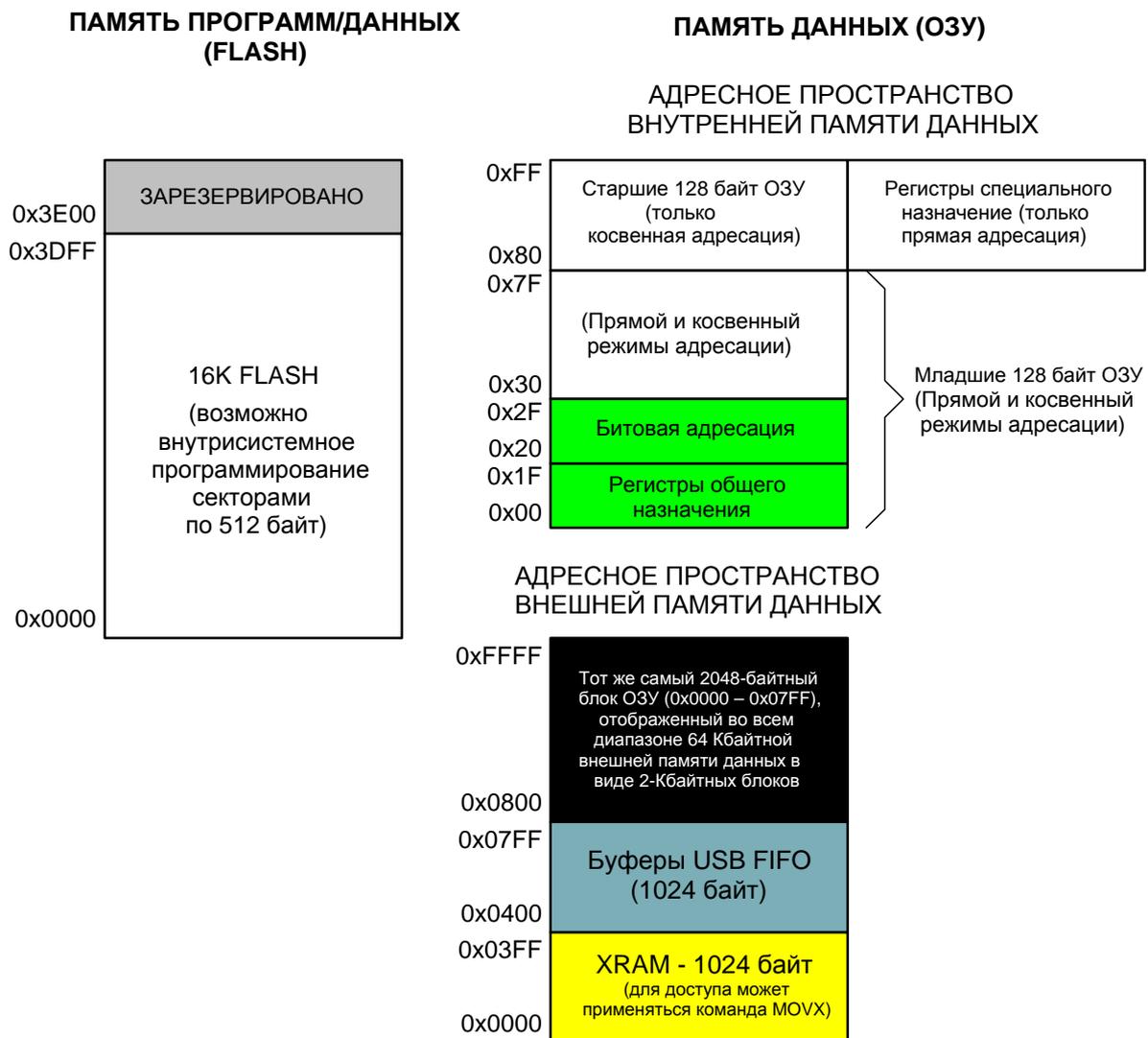


### 1.2. Встроенная память

CIP-51 имеет стандартную (8051) структуру адресного пространства памяти программ и данных. В состав памяти входит ОЗУ объемом 256 байт, старшие 128 байт которого имеют двойную конфигурацию. В режиме косвенной адресации осуществляется доступ к старшим 128 байтам ОЗУ общего назначения, а в режиме прямой адресации осуществляется доступ к 128 байтам адресного пространства регистров специального назначения (SFR). Младшие 128 байт ОЗУ доступны как для прямой, так и для косвенной адресации. Из них первые 32 байта адресуются как четыре банка регистров общего назначения, а следующие 16 байт адресуются побайтно или побитно.

Память программ МК состоит из 16 Кбайт Flash-памяти. Эта память может перепрограммироваться внутрисистемно секторами по 512 байт, не требуя при этом специального внешнего напряжения программирования. На рис.1.5 приведена карта распределения памяти МК.

Рисунок 1.5. Карта распределения памяти



### 1.3. USB-контроллер

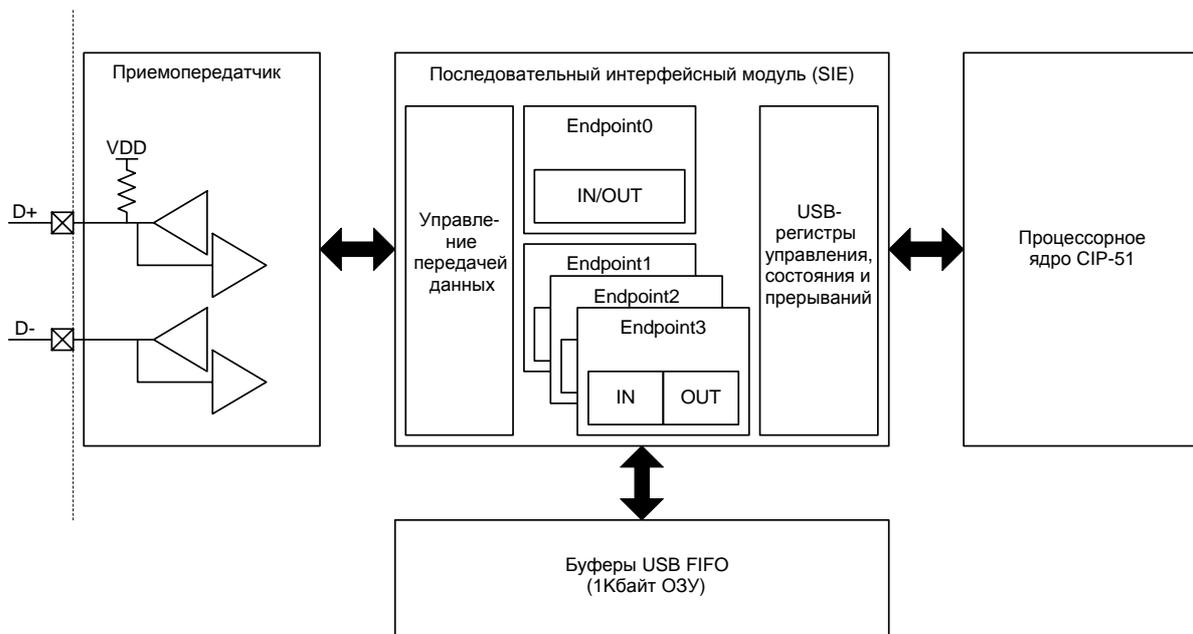
USB-контроллер (USB0) поддерживает высокоскоростной и низкоскоростной режимы функционирования в соответствии с USB2.0 и содержит встроенные приемопередатчик, а также ОЗУ типа FIFO конечных точек подключения (Endpoint). Всего доступно восемь Endpoint: управляющая двунаправленная Endpoint (Endpoint0) и три пары IN/OUT Endpoint (Endpoint1-3 IN/OUT).

В качестве буфера FIFO USB-контроллера используется отдельный блок памяти XRAM объемом 1Кбайт. Этот буфер FIFO делят между собой Endpoint0-3; слоты Endpoint1-3 FIFO можно настроить как IN (на прием данных), OUT (на передачу данных) или IN/OUT (раздельный режим). Максимальный размер FIFO составляет 512 байт (Endpoint3).

USB0 может функционировать в высокоскоростном или низкоскоростном режимах. Встроенные 4-кратный умножитель тактовой частоты и схема восстановления синхронизации позволяют использовать для тактирования USB в высокоскоростном или низкоскоростном режимах встроенный прецизионный генератор. Также для генерации тактового сигнала USB можно использовать внешний генератор совместно с 4-кратным умножителем тактовой частоты. Сигналы тактирования процессорного ядра (CPU) и USB независимы друг от друга.

Приемопередатчик USB0 совместим со спецификацией USB2.0 и содержит встроенные согласующие и подтягивающие резисторы. Подтягивающие резисторы могут программно подключаться/отключаться и будут появляться на выводах D+ и D- в зависимости от выбранного (высокоскоростной/низкоскоростной) режима работы.

Рисунок 1.6. Структурная схема USB-контроллера



### 1.4. Регулятор напряжения

МК C8051F320/1 содержат регулятор напряжения (REG0) 5В → 3В. Если разрешено, то выходной сигнал REG0 появляется на выводе VDD и может использоваться для питания внешних устройств. REG0 можно программно включить/отключить.

## 1.5. Встроенные средства отладки

МК C8051F320/1 имеют встроенный двухпроводный Silicon Labs Development Interface (интерфейс C2), который позволяет производить «неразрушающую» внутрисхемную отладку в режиме реального времени, используя МК, установленные в конечное изделие.

Средства отладки Silicon Labs' обеспечивают проверку и модификацию памяти и регистров, расстановку точек останова, а также пошаговое выполнение программы. При этом не требуется никаких специальных дополнительных ОЗУ, памяти программ, таймеров или каналов связи. Во время отладки все цифровые и аналоговые периферийные модули не отключаются и работают корректно. При остановке МК в точке останова или при пошаговой отладке работа всех периферийных модулей (кроме USB, АЦП и SMBus) блокируется, что необходимо для удержания их в режиме синхронизации с выполнением команд.

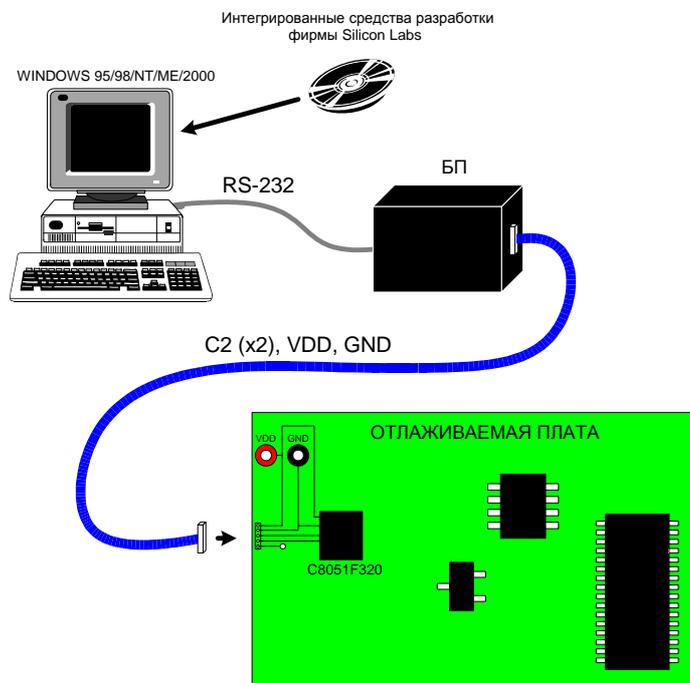
Комплект средств разработки C8051F310DK содержит все аппаратные и программные средства, необходимые для разработки программного кода и выполнения внутрисхемной отладки систем на основе МК C8051F320/1. Этот комплект включает в себя программный пакет с интегрированной средой разработки и отладки, интегрированный ассемблер стандарта 8051, а также блок-преобразователь (БП) RS-232/C2. Кроме этого имеется демонстрационная плата с установленным МК и свободным местом для макетирования, кабели RS-232 и C2, а также блок питания в отдельном корпусе. Для работы необходим компьютер с ОС Windows 95/98/NT/ME/2000 и одним свободным последовательным портом RS-232. Как показано на рис.1.7, компьютер через порт RS-232 подключается к БП. БП соединяется с платой пользователя шестижильным плоским кабелем, два провода которого используются интерфейсом C2, а еще два провода необходимы для подачи питания (VDD и GND). БП получает питание с платы пользователя. Если это невозможно, то можно подать питание непосредственно на БП от отдельного источника.

По сравнению со стандартными симуляторами интерфейс Silicon Labs IDE обеспечивает следующие преимущества:

- не требуется отладочный кристалл;
- не используются специализированные кабели;
- не требуется использовать разъем для установки МК на плату.

Предлагаемый Silicon Labs' способ отладки обеспечивает удобство работы с прецизионными аналоговыми периферийными модулями и при этом не ухудшает их производительности.

Рисунок 1.7. Модель отладки

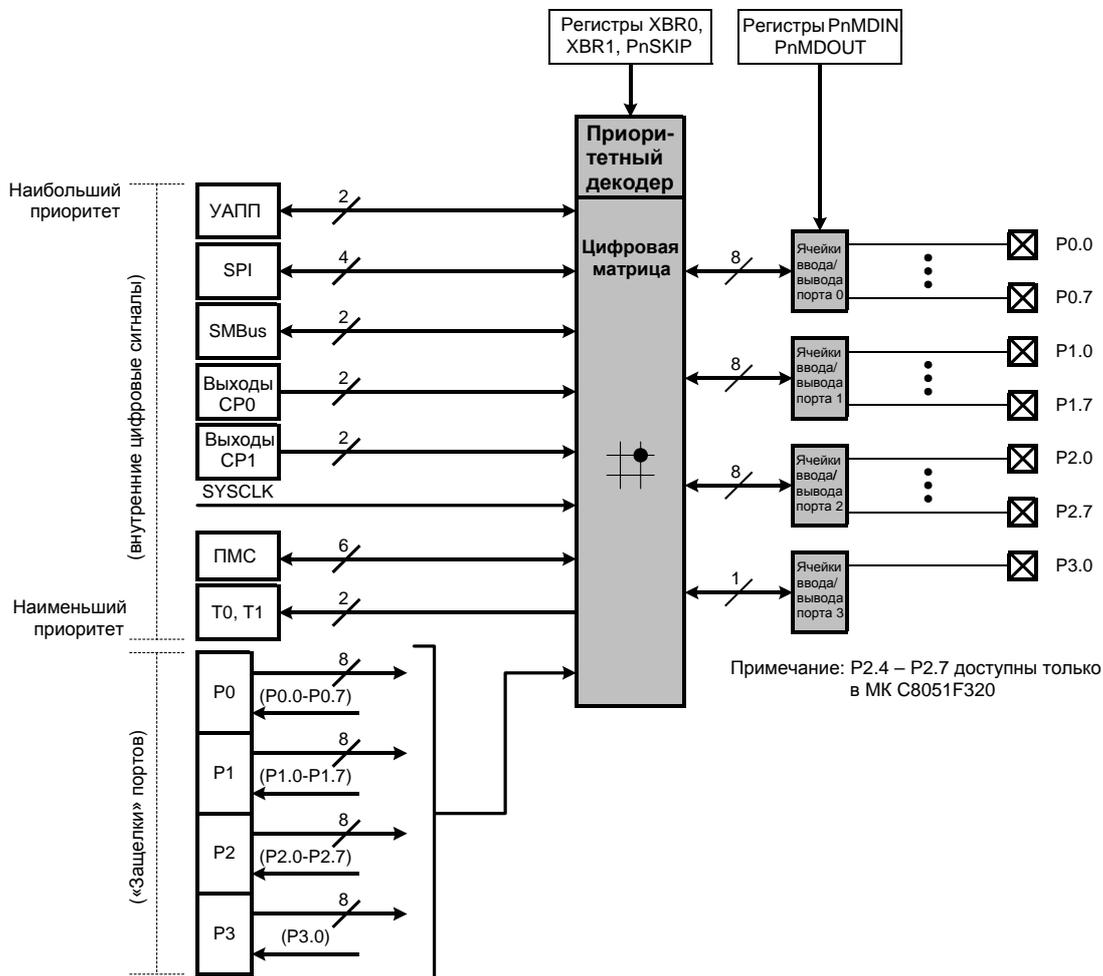


## 1.6. Программируемые цифровые порты ввода/вывода и матрица соединений

МК C8051F320 имеет 25 выводов ввода/вывода (три 8-разрядных порта и один 1-разрядный порт); МК C8051F321 имеет 21 вывод ввода/вывода (два 8-разрядных порта, один 4-разрядный порт и один 1-разрядный порт). Порты МК C8051F320/1 функционируют в соответствии со стандартом 8051 с некоторыми дополнительными возможностями. Каждый вывод порта можно настроить как аналоговый вход или как цифровой вход-выход. Выводы, настроенные как цифровые входы-выходы, можно кроме этого настроить как двухтактные цифровые выходы или выходы с открытым стоком. Также допускается глобальное отключение слаботочковых подтягивающих резисторов, что позволяет еще более снизить энергопотребление в критичных к этому параметру приложениях.

Цифровая матрица позволяет необходимым образом соединять внутренние цифровые системные ресурсы с выводами портов ввода/вывода (см. рис.1.8). При помощи регистров управления матрицей на выводы портов могут быть выведены сигналы от внутренних таймеров/счетчиков, от последовательных интерфейсов, аппаратные прерывания, выходы компараторов и другие цифровые сигналы. Это позволяет пользователю выбрать точную комбинацию связей между портами ввода/вывода общего назначения и цифровыми ресурсами, необходимую для каждого конкретного приложения.

Рисунок 1.8. Структурная схема цифровой матрицы



## 1.7. Последовательные порты

В МК семейства C8051F320/1 встроены следующие последовательные интерфейсы:

- полнодуплексный УАПП с усовершенствованным генератором скорости передачи данных;
- усовершенствованный SPI;
- I2C/SMBus.

Каждый из этих интерфейсов реализован на аппаратном уровне и широко использует прерывания, требуя лишь незначительного вмешательства со стороны программы пользователя.

### 1.8. Программируемый массив счетчиков (ПМС)

МК семейства C8051F320/1 кроме четырех 16-разрядных таймеров/счетчиков общего назначения имеют внутренний программируемый массив счетчиков (ПМС). ПМС состоит из специального 16-разрядного таймера/счетчика временных интервалов с пятью программируемыми модулями захват/сравнение. В качестве тактового сигнала ПМС могут использоваться:

- сигнал системного тактового генератора с частотой, деленной на 12;
- сигнал системного тактового генератора с частотой, деленной на 4;
- сигнал переполнения таймера 0;
- сигнал от внешнего входа тактирования (ECI – external clock input);
- системный тактовый сигнал;
- сигнал внешнего генератора с частотой, деленной на 8.

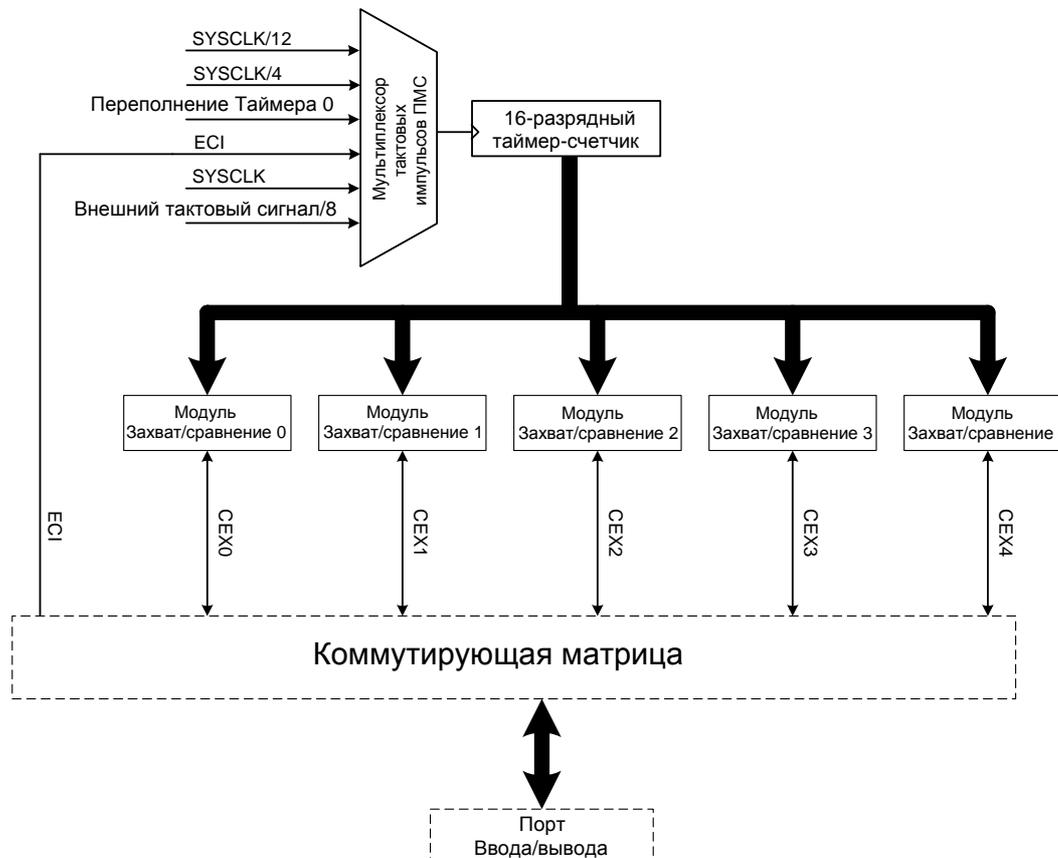
Выбор внешнего источника тактирования полезно использовать для реализации режима тактирования сигналом реального времени, когда ПМС тактируется от внешнего источника, а внутренний генератор формирует системный тактовый сигнал.

Каждый модуль захват/сравнение может быть настроен на работу в одном из шести режимах:

- захват, управляемый фронтом (сигнала);
- программный таймер;
- высокоскоростной выход;
- выход заданной частоты;
- 8-разрядный широтно-импульсный модулятор;
- 16-разрядный широтно-импульсный модулятор.

Кроме этого, модуль захвата/сравнения 4 может работать в режиме сторожевого таймера (WDT). После сброса модуль 4 настраивается и включается в режиме WDT. Входы/выходы модулей захвата/сравнения ПМС и внешний вход тактирования (ECI) соединяются с портами ввода/вывода МК через цифровую коммутирующую матрицу.

Рисунок 1.9. Структурная схема модуля ПМС



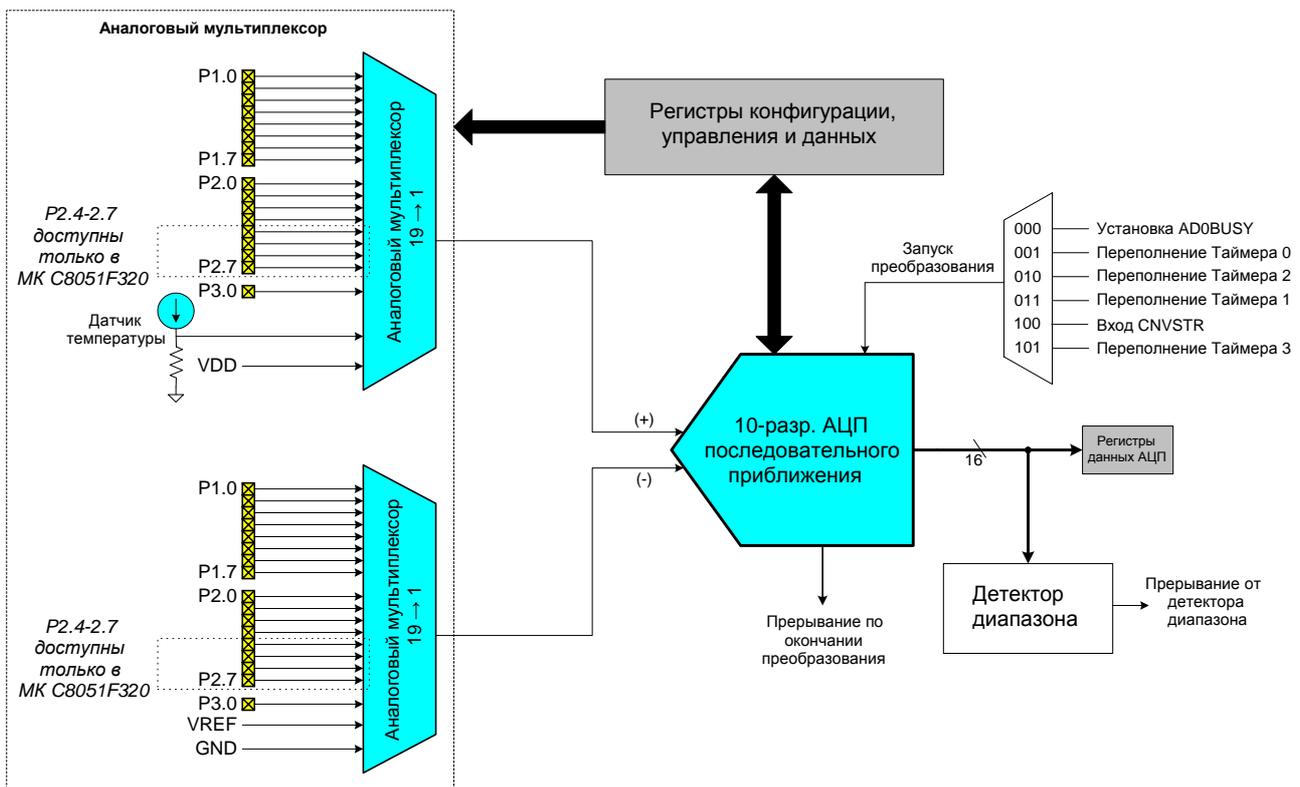
## 1.9. 10-разрядный аналого-цифровой преобразователь

МК C8051F320/1 имеют встроенный 10-разрядный АЦП последовательного приближения с 17-канальным дифференциальным входным мультиплексором. При максимальной производительности 200 тыс. преобразований в секунду этот АЦП обеспечивает 10-битную точность преобразования с нелинейностью на уровне  $\pm 1\text{МЗР}$ . Система АЦП включает настраиваемый аналоговый мультиплексор, посредством которого осуществляется выбор как положительного, так и отрицательного входов АЦП. Порты 1-3 доступны как аналоговые входы; кроме этого, входными сигналами АЦП могут быть выходной сигнал встроенного датчика температуры и напряжение питания (VDD). Программа пользователя может отключать АЦП с целью уменьшения энергопотребления.

Для запуска преобразования могут использоваться: программная команда, переполнение таймеров 0, 1, 2 или 3, внешний сигнал запуска. Такая гибкость позволяет осуществлять запуск преобразований при возникновении определенных программных событий, периодически (при переполнениях таймера) или сигналом от внешних устройств. При завершении преобразования полученное 10-разрядное слово данных «защелкивается» в регистрах данных АЦП, устанавливается специальный бит состояния и генерируется прерывание (если оно разрешено).

АЦП может быть настроен таким образом, чтобы генерировать прерывание лишь при попадании или непопадании результата преобразования в заданный диапазон значений (окно). АЦП может непрерывно отслеживать сигнал в фоновом режиме, но не прерывать МК до тех пор, пока преобразованные данные находятся в пределах или вне пределов заданного окна.

Рисунок 1.10. Структурная схема 10-разрядного АЦП

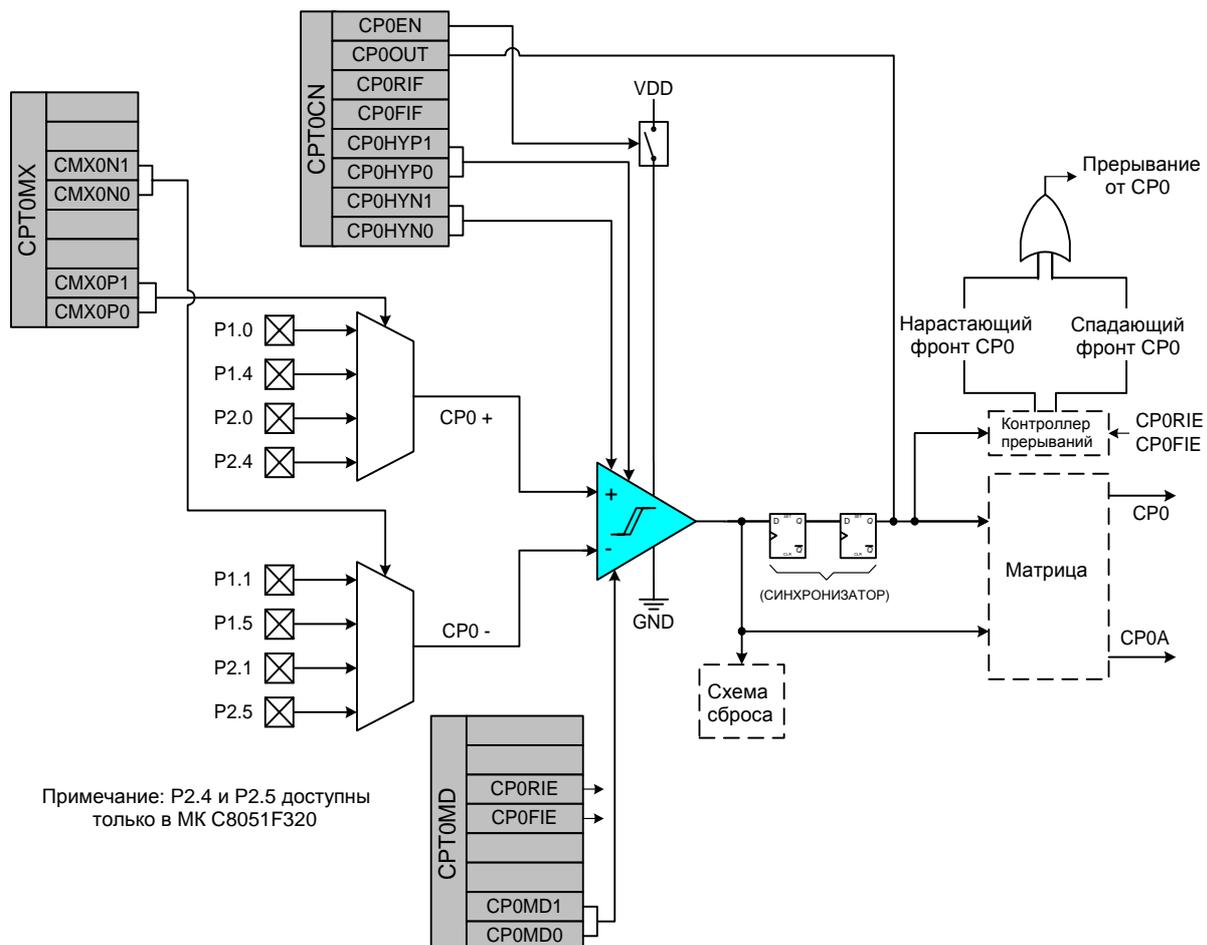


### 1.10. Компараторы

МК C8051F320/1 имеют два встроенных компаратора напряжения, которые включаются/отключаются и настраиваются программой пользователя. Порты ввода/вывода подключаются к входам компаратора через мультиплексор. При необходимости на вывод порта можно вывести (с помощью матрицы) два выходных сигнала компаратора: фиксируемый и/или нефиксируемый (асинхронный). Время отклика компаратора программируется, что позволяет выбирать между высокоскоростным и энергосберегающим режимами работы. Величина положительного и отрицательного гистерезиса также программируется.

Каждый компаратор может генерировать прерывание по переднему или заднему фронту петли гистерезиса, либо по обоим фронтам; эти прерывания могут вывести МК из режима остановки. Компаратор 0 можно использовать также в качестве источника сброса. Структурная схема Компаратора 0 приведена на рис.1.11.

Рисунок 1.11. Структурная схема Компаратора 0



## 2. ПРЕДЕЛЬНО ДОПУСТИМЫЕ ПАРАМЕТРЫ\*

Предельная рабочая температура.....	от -55°C до +125°C
Температура хранения.....	от -65°C до +150°C
Напряжение на любом выводе порта ввода/вывода или на выводе /RST относительно GND.....	от -0.3В до 5.8В
Напряжение на выводе VDD относительно GND.....	от -0.3В до 4.2В
Максимальный суммарный ток по выводам VDD и GND.....	500мА
Максимальный выходной втекающий ток по любому порту ввода/вывода или по выводу /RST .....	100мА

\* Выход за указанные значения может привести к необратимым повреждениям микроконтроллера. Работа микроконтроллера в предельном режиме в течение длительного времени не предусмотрена. Длительная эксплуатация микроконтроллера в недопустимых условиях может повлиять на его надежность.

### 3. ОСНОВНЫЕ ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ

Таблица 3.1. Основные электрические параметры

Температура от -40°C до +85°C, тактовая частота 25МГц, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Напряжение питания цифровых цепей	(Примечание 1)	2.7	3.3	3.6	В
Ток потребления от источника питания цифровых цепей при активном CPU	VDD = 3.3В, Частота = 24 МГц VDD = 3.3В, Частота = 1 МГц VDD = 3.3В, Частота = 32 кГц		10 0.6 30		мА мА мкА
Ток потребления от источника питания цифровых цепей при активном CPU и активном USB (высокоскоростной или низкоскоростной режим)	VDD = 3.3В, Частота = 24 МГц VDD = 3.3В, Частота = 6 МГц		неопр. неопр.		мА мА
Ток потребления от источника питания цифровых цепей с остановленным CPU (нет обращений к Flash-памяти)	VDD = 3.3В, Частота = 24 МГц VDD = 3.3В, Частота = 1 МГц VDD = 3.3В, Частота = 32 кГц		5 0,3 14		мА мА мкА
Ток потребления от источника питания цифровых цепей в режиме пониженного энергопотребления или в режиме отключения	Генератор остановлен.		< 0.1		мкА
Напряжение сохранения данных ОЗУ			1.5		В
Рабочая температура		-40		+85	°С
SYSCLK (системная тактовая частота)	(Примечание 2)	0		25	МГц
Tsysl (длительность низкого уровня сигнала SYSCLK)		18			нс
Tsysh (длительность высокого уровня сигнала SYSCLK)		18			нс

Примечание 1: USB требует минимального напряжения питания 3В.

Примечание 2: Отладка невозможна при SYSCLK менее 32 кГц.



## 4. ОПИСАНИЕ КОРПУСОВ И ВЫВОДОВ

Таблица 4.1. Описание выводов МК C8051F320/1

Обозначение вывода	Номер вывода		Тип	Описание
	'F320	'F321		
VDD	6	6	Power in Power out	Вход напряжения питания (2,7В...3,6В). Выход регулятора напряжения 3.3В. См. раздел 8.
GND	3	3		Общий вывод питания («земля»).
/RST	9	9	D I/O	Сброс МК. Выход с открытым стоком внутренней схемы слежения за напряжением питания. Внешний источник может вызвать сброс МК, установив низкий логический уровень на этом выводе в течение не менее 1.5 мкс. См. раздел 10.
C2CK			D I/O	Сигнал тактирования для интерфейса отладки C2.
P3.0/	10	10	D I/O	Порт 3.0. См. полное описание в разделе 14.
C2D			D I/O	Двунаправленный сигнал данных для интерфейса отладки C2.
REGIN	7	7	Power in	Вход 5В встроенного регулятора напряжения.
VBUS	8	8	D In	Вход считывания VBUS. На этот вывод следует подать сигнал VBUS шины USB. Наличие напряжения 5В на этом выводе означает, что шина USB подключена.
D+	4	4	D I/O	USB D+
D-	5	5	D I/O	USB D-
P0.0	2	2	D I/O	Порт 0.0. См. полное описание в разделе 14.
P0.1	1	1	D I/O	Порт 0.1. См. полное описание в разделе 14.
P0.2/			D I/O	Порт 0.1. См. полное описание в разделе 14.
XTAL1	32	28	A In	Вход внешнего тактового сигнала. Этот вывод является выходом внешнего генератора в случае использования кварцевого или керамического резонатора. См. раздел 13.
P0.3/	31	27	D I/O	Порт 0.3. См. полное описание в разделе 14.
XTAL2			A In или D In	Выход внешнего тактового сигнала. Этот вывод является выходом схемы возбуждения для кварцевого или керамического резонатора, или входом внешнего тактового сигнала в случае использования КМОП-счетчика, C- или RC-генератора. См. раздел 13.
P0.4	30	26	D I/O	Порт 0.4. См. полное описание в разделе 14.
P0.5	29	25	D I/O	Порт 0.5. См. полное описание в разделе 14.
P0.6	28	24		Порт 0.6. См. полное описание в разделе 14.
CNVSTR				Внешний вход запуска преобразования АЦПО. См. раздел 5.
P0.7/	23	27	D I/O	Порт 0.7. См. полное описание в разделе 14.
VREF			A I/O	Вход или выход внешнего опорного напряжения. См. раздел 6.

Таблица 4.1. Описание выводов (продолжение)

Обозначение вывода	Номер вывода		Тип	Описание
	'F320	'F321		
P1.0	26	22	D I/O или A In	Порт 1.0. См. полное описание в разделе 14.
P1.1	25	21	D I/O или A In	Порт 1.1. См. полное описание в разделе 14.
P1.2	24	20	D I/O или A In	Порт 1.2. См. полное описание в разделе 14.
P1.3	23	19	D I/O или A In	Порт 1.3. См. полное описание в разделе 14.
P1.4	22	18	D I/O или A In	Порт 1.4. См. полное описание в разделе 14.
P1.5	21	17	D I/O или A In	Порт 1.5. См. полное описание в разделе 14.
P1.6	20	16	D I/O или A In	Порт 1.6. См. полное описание в разделе 14.
P1.7	19	15	D I/O или A In	Порт 1.7. См. полное описание в разделе 14.
P2.0	18	14	D I/O или A In	Порт 2.0. См. полное описание в разделе 14.
P2.1	17	13	D I/O или A In	Порт 2.1. См. полное описание в разделе 14.
P2.2	16	12	D I/O или A In	Порт 2.2. См. полное описание в разделе 14.
P2.3	15	11	D I/O или A In	Порт 2.3. См. полное описание в разделе 14.
P2.4	14		D I/O или A In	Порт 2.4. См. полное описание в разделе 14.
P2.5	13		D I/O или A In	Порт 2.5. См. полное описание в разделе 14.
P2.6	12		D I/O или A In	Порт 2.6. См. полное описание в разделе 14.
P2.7	11		D I/O или A In	Порт 2.7. См. полное описание в разделе 14.

Рисунок 4.1. Цоколевка корпуса LQFP-32 (вид сверху)

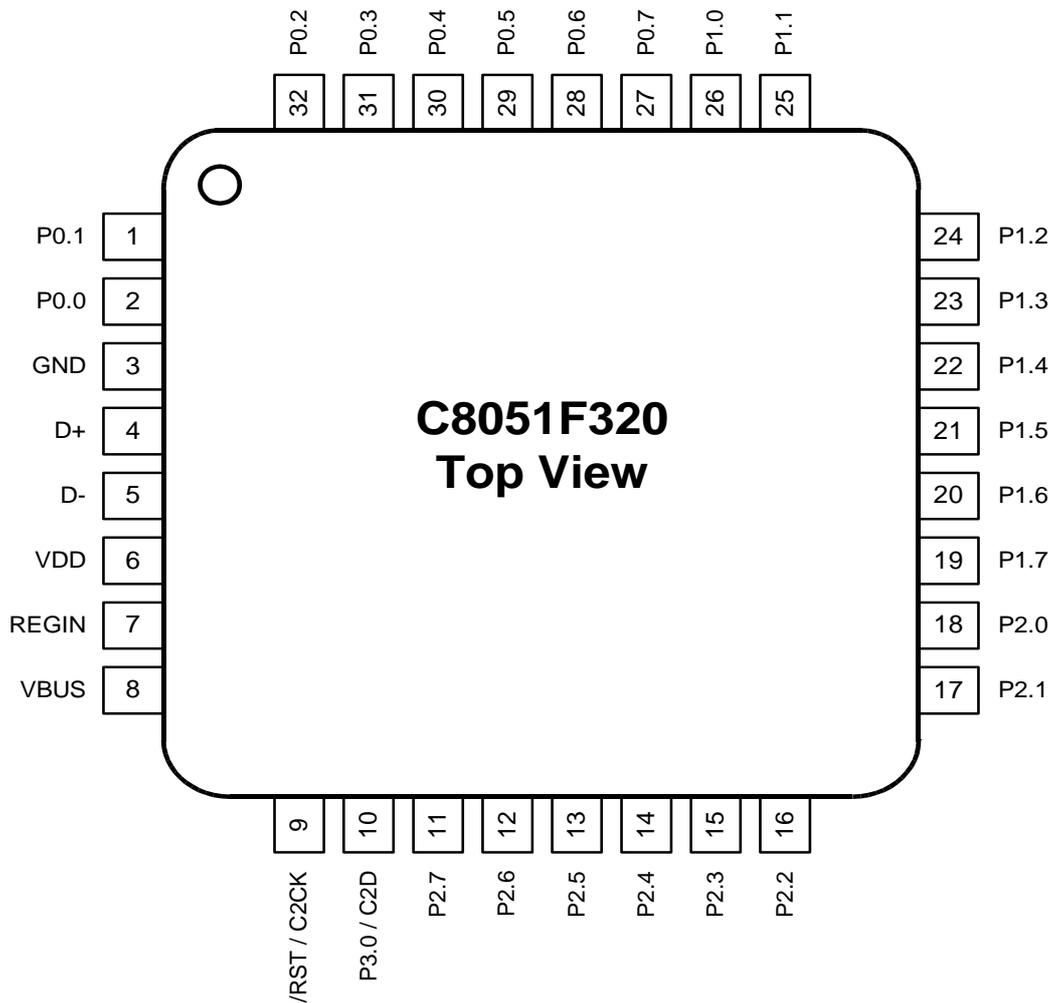
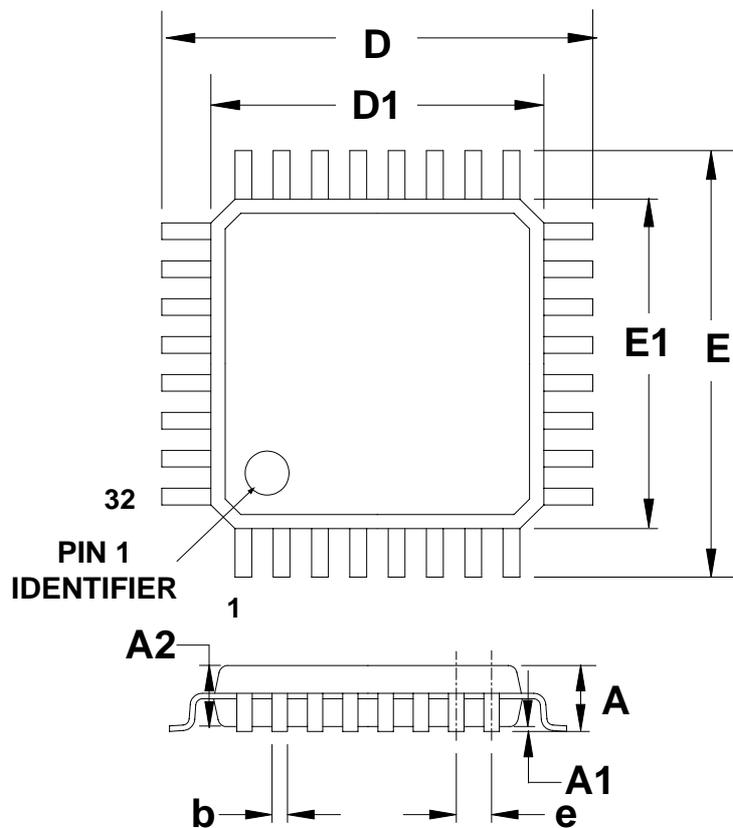


Рисунок 4.2. Чертеж корпуса LQFP-32

Таблица 4.2. Размеры корпуса LQFP-32



	MM		
	MIN	TYP	MAX
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.30	0.37	0.45
D	-	9.00	-
D1	-	7.00	-
e	-	0.80	-
E	-	9.00	-
E1	-	7.00	-

Рисунок 4.3. Цоколевка корпуса MLP-28

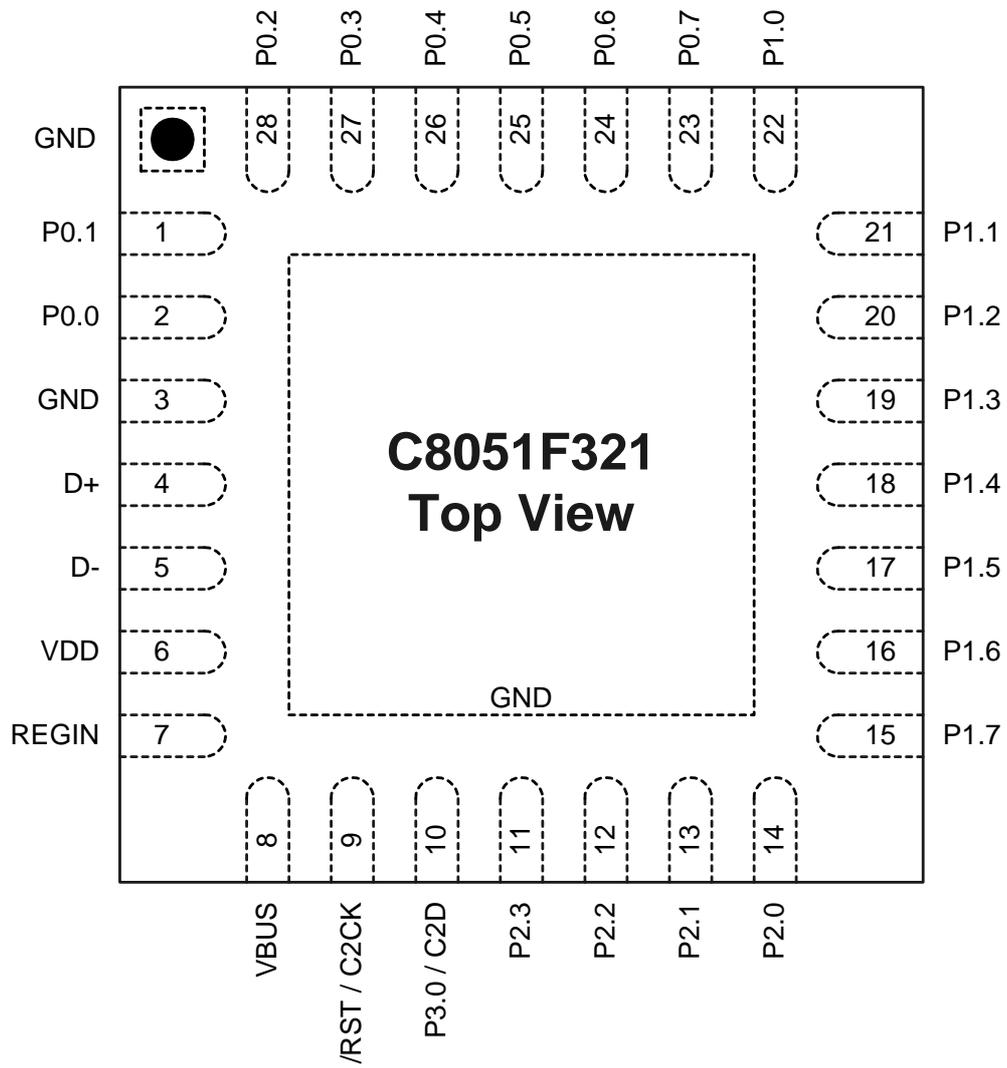
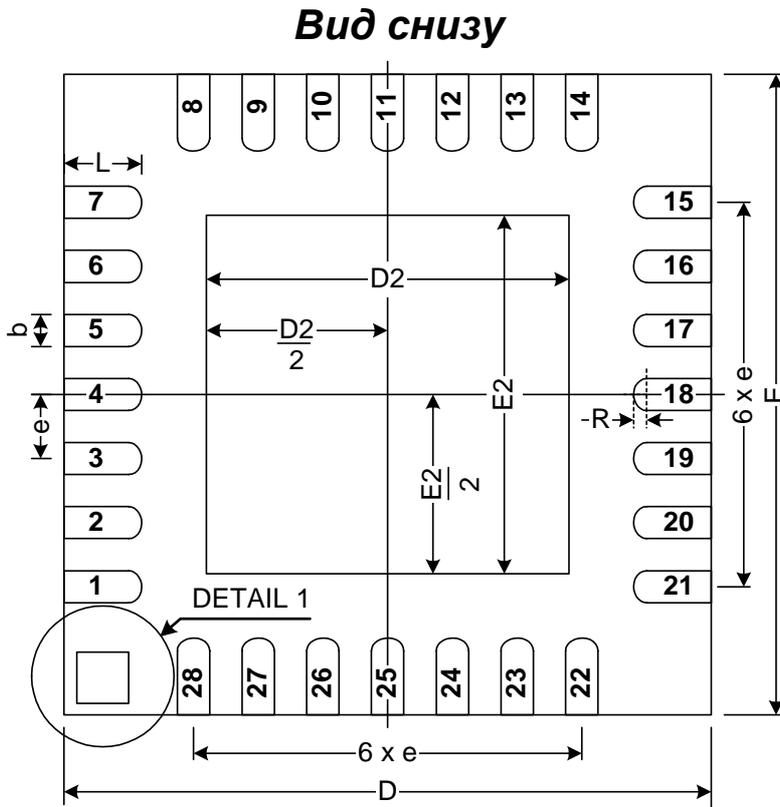


Рисунок 4.4. Чертеж корпуса MLP-28

Таблица 4.3. Размеры корпуса MLP-28



	MM		
	MIN	TYP	MAX
A	0.80	0.90	1.00
A1	0	0.02	0.05
A2	0	0.65	1.00
A3	-	0.25	-
b	0.18	0.23	0.30
D	-	5.00	-
D2	2.90	3.15	3.35
E	-	5.00	-
E2	2.90	3.15	3.35
e	-	0.5	-
L	0.45	0.55	0.65
N	-	28	-
ND	-	7	-
NE	-	7	-
R	0.09	-	-
AA	-	0.435	-
BB	-	0.435	-
CC	-	0.18	-
DD	-	0.18	-

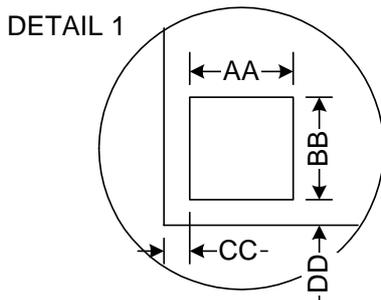
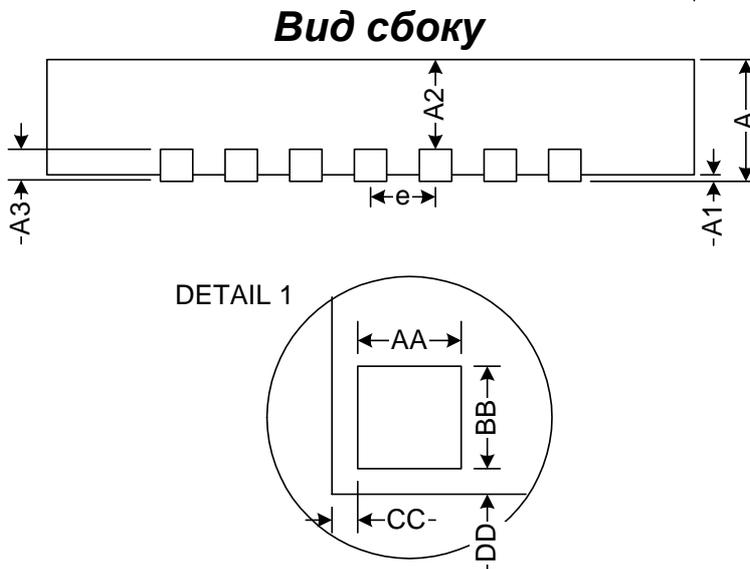
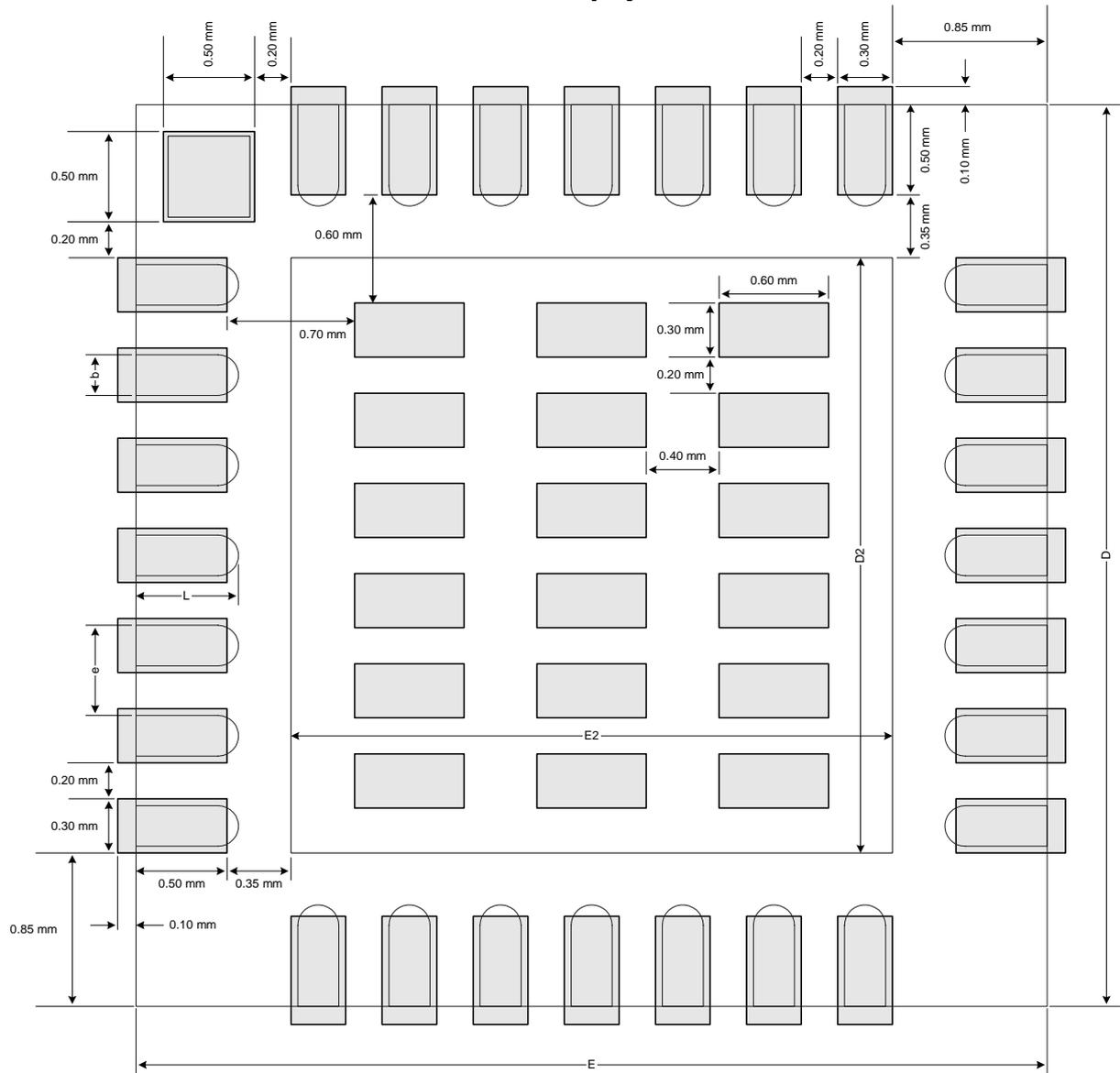




Рисунок 4.6. Трафарет для нанесения припоя для корпуса MLP-28

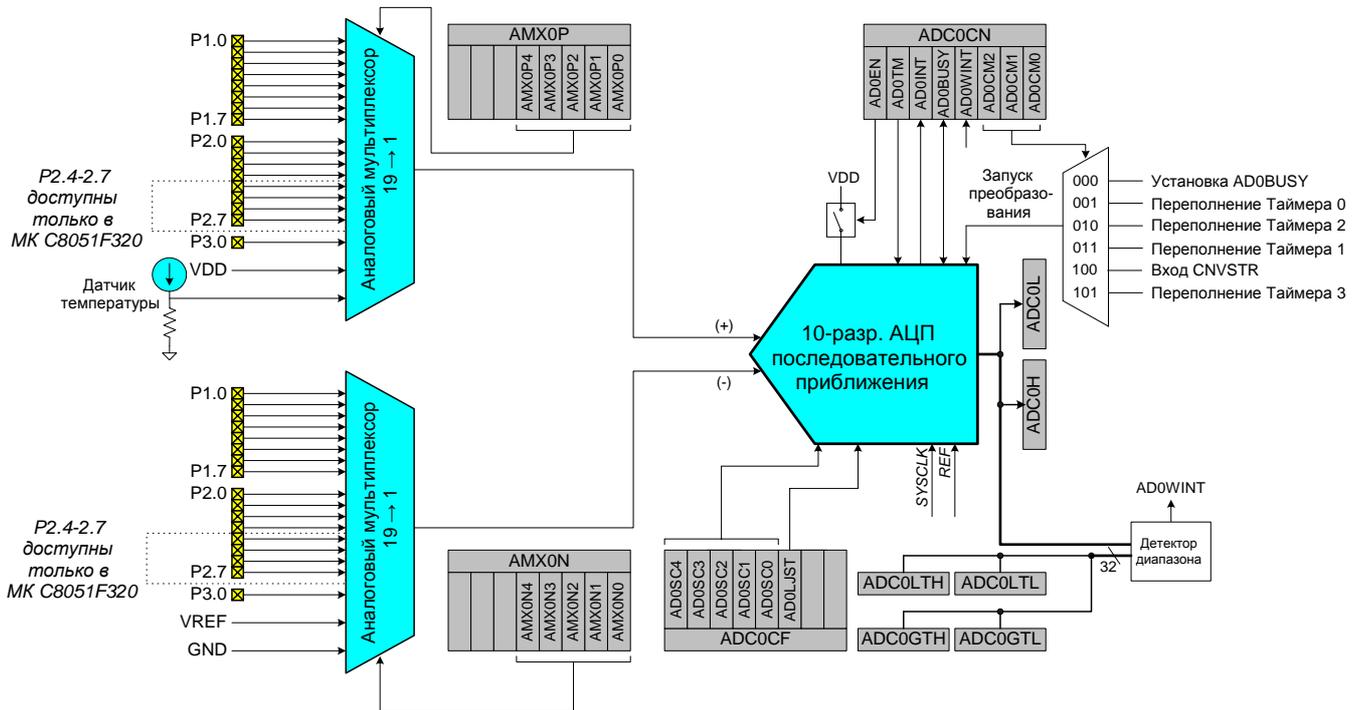
Вид сверху



## 5. 10-разрядный АЦП (АЦПО)

Модуль АЦПО МК C8051F320/1 состоит из двух 17-канальных аналоговых мультиплексоров (обозначаются вместе как AMUX0) и 10-разрядного АЦП последовательного приближения (максимальная производительность - 200 тыс. преобразований в секунду) с интегрированным устройством выборки-хранения (УВХ) и программируемым детектором диапазона. AMUX0, режимы преобразования и детектор диапазона настраиваются программным путем при помощи регистров специального назначения (см. рис.5.1). АЦПО функционирует как в однофазном, так и в дифференциальном режимах, и может быть настроен на измерение напряжения на выводах P1.0 – P3.0, выходного напряжения датчика температуры или напряжения VDD относительно P1.0 – P3.0, VREF или GND. Модуль АЦПО включен только тогда, когда бит AD0EN регистра управления АЦПО (ADC0CN) установлен в 1. Сброс этого бита в 0 переводит АЦПО в режим отключения с пониженным энергопотреблением.

**Рисунок 5.1. Функциональная схема АЦПО**



## 5.1. Аналоговый мультиплексор

AMUX0 осуществляет выбор положительного и отрицательного входов АЦП.

В качестве положительного входа можно выбрать:

- P1.0 – P3.0;
- Выходной сигнал встроенного датчика температуры;
- Положительное напряжение питания (VDD).

В качестве отрицательного входа можно выбрать:

- P1.0 – P3.0;
- VREF;
- Общий вывод питания GND.

Если в качестве отрицательного входа выбран GND, то АЦП функционирует в однофазном режиме; в остальных случаях АЦП функционирует в дифференциальном режиме. Входные каналы АЦП выбираются в регистрах AMX0P и AMX0N (см. рис.5.5 и рис.5.6).

Формат получаемого результата преобразования различен для однофазного и дифференциального режимов. По окончании каждого преобразования в регистры ADC0H и ADC0L записываются соответственно старший и младший байты результата преобразования. Данные могут быть выровнены вправо или влево в зависимости от значения бита AD0LJST (ADC0CN.0). В однофазном режиме результаты преобразований представлены в виде 10-разрядных целых чисел без знака. Диапазон измерения входных напряжений: 0 ... VREF\*1023/1024. Ниже приведены примеры результатов преобразований при выравнивании как вправо, так и влево. Неиспользуемые биты в регистрах ADC0H и ADC0L установлены в '0'.

Входное напряжение (однофазный режим)	ADC0H:ADC0L с выравниванием вправо (AD0LJST=0)	ADC0H:ADC0L с выравниванием влево (AD0LJST=1)
VREF*1023/1024	0x03FF	0xFFC0
VREF*512/1024	0x0200	0x8000
VREF*256/1024	0x0100	0x4000
0	0x0000	0x0000

В дифференциальном режиме результаты преобразований представлены в виде 10-разрядных чисел в дополнителном коде со знаком. Диапазон измерения входных напряжений: - VREF ... VREF\*511/512. Ниже приведены примеры результатов преобразований при выравнивании как вправо, так и влево. При выравнивании вправо неиспользуемые старшие разряды регистра ADC0H являются знаковым расширением слова данных. При выравнивании влево неиспользуемые младшие разряды регистра ADC0L установлены в '0'.

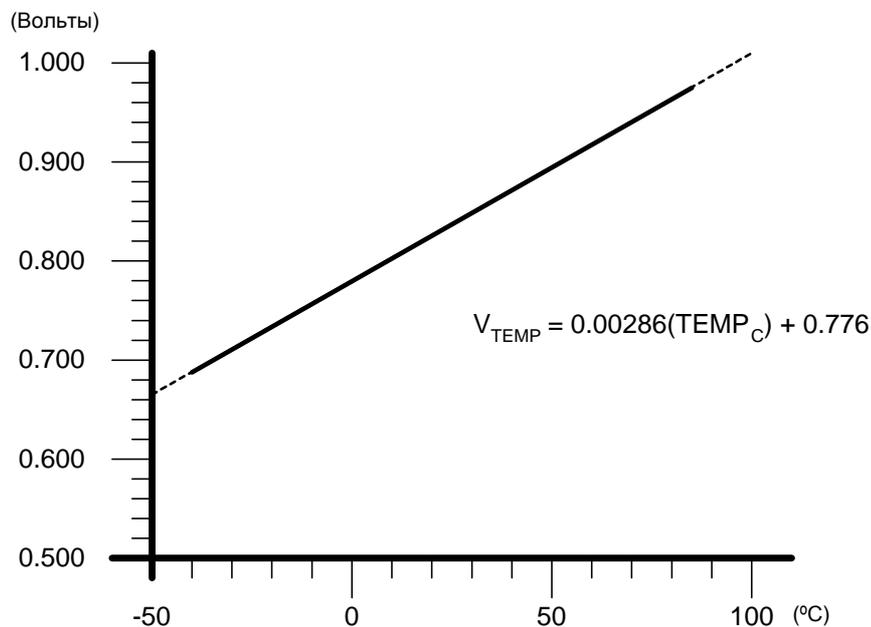
Входное напряжение (дифференциальный режим)	ADC0H:ADC0L с выравниванием вправо (AD0LJST=0)	ADC0H:ADC0L с выравниванием влево (AD0LJST=1)
VREF*511/512	0x01FF	0x7FC0
VREF*256/512	0x0100	0x4000
0	0x0000	0x0000
- VREF*256/512	0xFF00	0xC000
- VREF	0xFE00	0x8000

**Важное замечание относительно настройки входов АЦП:** Выводы порта, выбранные в качестве входов АЦП, должны быть настроены как аналоговые входы, и должны пропускаться матрицей при назначении выводов. Чтобы настроить вывод порта как аналоговый вход, следует сбросить в 0 соответствующий бит в регистре PnMDIN (для n = 0,1,2,3). Чтобы заставить матрицу пропускать вывод порта при назначении выводов, следует установить в 1 соответствующий бит в регистре PnSKIP (для n = 0,1,2). Более подробная информация о настройке порта ввода/вывода приведена в разделе 14 «Порт ввода/вывода» на стр.127.

## 5.2. Датчик температуры

Типичная передаточная характеристика датчика температуры приведена на рис.5.2. Выходное напряжение  $V_{TEMP}$  является положительным входом АЦП в том случае, если датчик температуры выбран битами AMX0P4-0 в регистре AMX0P.

**Рисунок 5.2. Передаточная характеристика датчика температуры**



Следует иметь в виду, что параметры, влияющие на результат измерения АЦП, особенно значение опорного напряжения, будут также влиять и на измерение температуры.

### 5.3. Режимы работы АЦП

Максимальная скорость преобразования АЦПО – 200 тыс. преобразований в секунду. Частота дискретизации АЦПО определяется частотой системного тактового сигнала, деленной на значение, задаваемое битами AD0SC регистра ADC0CF, т.е.  $\text{SYSCLK}/(\text{AD0SC} + 1)$  для  $0 \leq \text{AD0SC} \leq 31$ .

#### 5.3.1. Запуск преобразования

Запуск преобразования может быть осуществлен одним из шести способов, в зависимости от состояния битов режима запуска преобразования АЦПО (AD0CM2-0) в регистре ADC0CN. Преобразование может быть инициировано:

- 1) установкой в 1 бита AD0BUSY в регистре ADC0CN;
- 2) переполнением Таймера 0 (т.е. непрерывное по времени преобразование);
- 3) переполнением Таймера 2;
- 4) переполнением Таймера 1;
- 3) нарастающим фронтом внешнего входного сигнала CNVSTR (вывод P0.6);
- 4) переполнением Таймера 3.

Установка в 1 бита AD0BUSY позволяет осуществлять программное управление АЦПО, т.е. выполнять преобразования «по требованию». Бит AD0BUSY устанавливается в 1 во время преобразования и сбрасывается в 0 после окончания преобразования. При сбросе бита AD0BUSY инициируется прерывание (если оно разрешено) и устанавливается флаг прерывания от АЦПО (AD0INT). Примечание: При определении окончания преобразования методом опроса следует использовать флаг прерывания от АЦПО (AD0INT). Преобразованные данные доступны в регистрах старшего и младшего слова данных АЦПО, ADC0H и ADC0L соответственно, когда AD0INT=1. Следует иметь в виду, что если запуск преобразования осуществляется переполнением Таймера 2 или Таймера 3, то используются переполнения младшего байта, когда Таймер 2/3 работает в 8-разрядном режиме, и переполнения старшего байта, когда Таймер 2/3 работает в 16-разрядном режиме. Информация о настройке таймеров приведена в разделе 19 «Таймеры» на стр.217.

**Важное замечание относительно использования CNVSTR:** Входной вывод CNVSTR функционирует также как вывод порта P0.6. Если вход CNVSTR используется для запуска преобразования АЦПО, то вывод порта P0.6 должен пропускаться матрицей при назначении выводов. Чтобы заставить матрицу пропускать P0.6, следует установить в 1 бит 6 в регистре POSKIP. Более подробная информация о настройке порта ввода/вывода приведена в разделе 14 «Порт ввода/вывода» на стр.127.

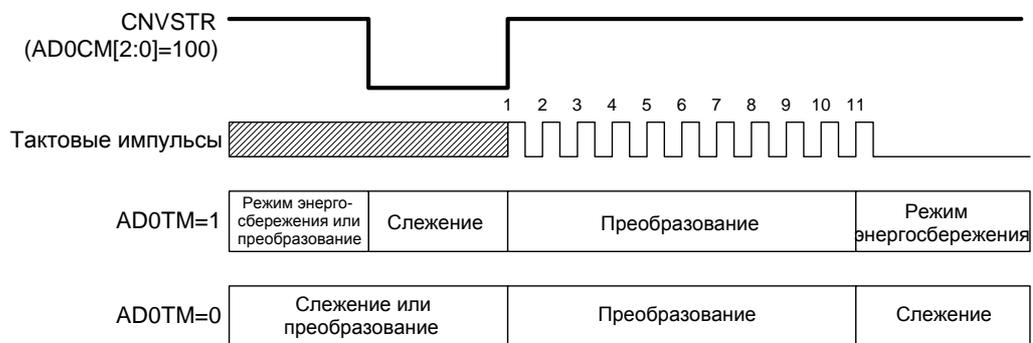


## 5.3.2. Режимы слежения

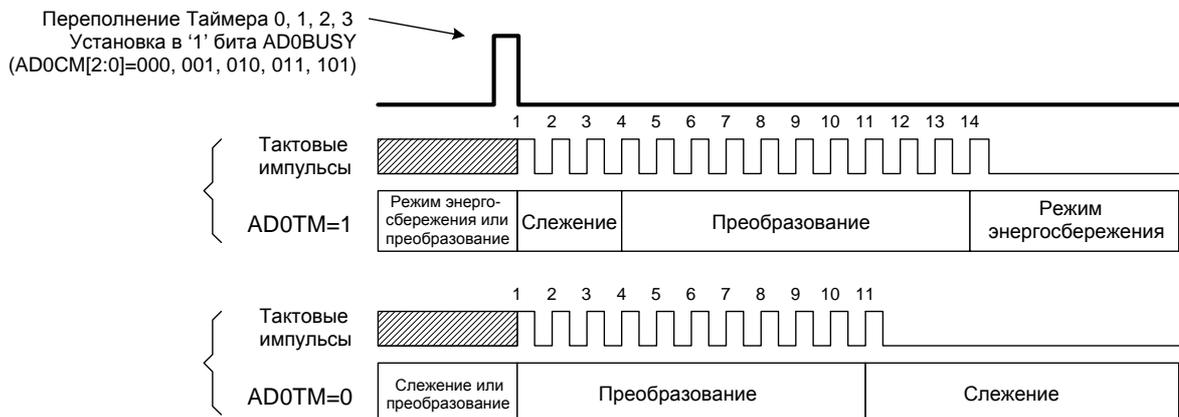
Бит AD0TM регистра ADC0CN управляет режимом выборки-хранения АЦПО. По умолчанию состояние входа АЦПО отслеживается непрерывно, за исключением момента преобразования. Установка в 1 бита AD0TM переводит АЦПО в энергосберегающий режим выборки-хранения. В этом режиме каждому преобразованию предшествует (после сигнала запуска преобразования) период выборки, равный трем периодам сигнала дискретизации АЦП. Если для запуска преобразования в энергосберегающем режиме выборки-хранения используется сигнал CNVSTR, то АЦПО отслеживает входной сигнал только тогда, когда на входе CNVSTR присутствует сигнал низкого уровня; преобразование запускается нарастающим фронтом сигнала на входе CNVSTR (см. рис.5.3). Кроме этого слежение может быть запрещено (отключено), когда весь МК переведен в мало потребляющие режимы ожидания или остановки. Энергосберегающий режим выборки-хранения также полезен в том случае, когда параметры AMUX часто изменяются. Этот режим позволяет гарантировать, что время установления соответствует заданным требованиям (см. раздел 5.3.3 на стр.44).

**Рисунок 5.9. Временные диаграммы процесса преобразования 10-разр. АЦП**

### А. Временные диаграммы АЦПО с внешним источником запуска



### В. Временные диаграммы АЦПО с внутренним источником запуска



### 5.3.3. Время установления

Если конфигурация входов АЦПО изменяется (т.е. изменяются настройки AMUX0), то после этого для обеспечения точности преобразования необходимо выдержать паузу длительностью не менее минимального времени установления сигнала. Время установления определяется сопротивлением AMUX0, емкостью накопительного конденсатора VBX, сопротивлением внешнего источника сигнала и требуемой точностью преобразования. Следует отметить, что в энергосберегающем режиме выборки-хранения после запуска каждого преобразования выборка длится три периода сигнала дискретизации АЦП. Для большинства приложений эти три периода сигнала дискретизации будут соответствовать требованиям, предъявляемым ко времени установления.

На рис.5.4 показаны эквивалентные схемы входов АЦПО как для дифференциального, так и для однофазного режимов работы. Следует отметить, что эквивалентная постоянная времени для обеих схем одинакова. Требуемое время установления для заданной точности установления (settling accuracy – SA) можно приблизительно определить из уравнения 5.1. Если измеряется выходное напряжение датчика температуры или напряжение VDD относительно GND, то  $R_{TOTAL} = R_{MUX}$ . Абсолютная величина минимального времени установления (выборки) приведена в таблице 5.1.

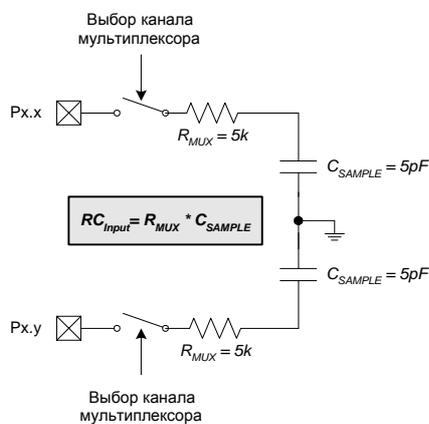
#### Уравнение 5.1. Время установления сигнала АЦПО

$$t = \ln(2^n/SA) \times R_{TOTAL}C_{SAMPLE}$$

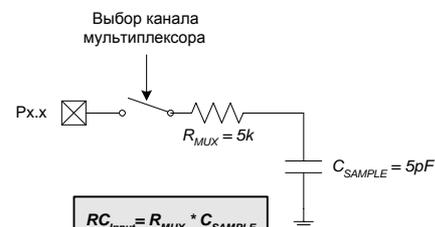
где: SA – точность установления, задаваемая в долях МЗР (например, 0.25 для установления в пределах ¼ МЗР)  
 t - требуемое время установления в секундах  
 $R_{TOTAL}$  – сумма сопротивления AMUX0 и сопротивления внешнего источника сигнала  
 n - разрешение АЦП в битах (10)

Рисунок 5.4. Эквивалентные схемы входов АЦПО

#### Дифференциальный режим



#### Однофазный режим



**Рисунок 5.5. AMX0P: Регистр выбора положительного канала AMUX0**

R	R	R	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
-	-	-	AMX0P4	AMX0P3	AMX0P2	AMX0P1	AMX0P0	SFR Адрес: 0xBB
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-5: **Не используются**: читаются как 000b. Запись не оказывает никакого влияния.

Биты 4-0: AMX0P4-0: Биты выбора положительного входа AMUX0.

AMX0P4-0	Положительный вход АЦПО
00000	P1.0
00001	P1.1
00010	P1.2
00011	P1.3
00100	P1.4
00101	P1.5
00110	P1.6
00111	P1.7
01000	P2.0
01001	P2.1
01010	P2.2
01011	P2.3
01100*	P2.4*
01101*	P2.5*
01110*	P2.6*
01111*	P2.7*
10000	P3.0
10001-11101	Зарезервированы
11110	Датчик температуры
11111	VDD

\*Только для C8051F320; в МК C8051F321 будет «Зарезервировано».

Рисунок 5.6. AMX0N: Регистр выбора отрицательного канала AMUX0

R	R	R	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	-	AMX0N4	AMX0N3	AMX0N2	AMX0N1	AMX0N0	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xBA

Биты 7-5: **Не используются**: читаются как 000b. Запись не оказывает никакого влияния.

Биты 4-0: AMX0N4-0: Биты выбора отрицательного входа AMUX0.

Следует иметь в виду, что если в качестве отрицательного входа выбрано GND, то АЦПО работает в однофазном режиме. Во всех других случаях АЦПО работает в дифференциальном режиме.

AMX0N4-0	Положительный вход АЦПО
00000	P1.0
00001	P1.1
00010	P1.2
00011	P1.3
00100	P1.4
00101	P1.5
00110	P1.6
00111	P1.7
01000	P2.0
01001	P2.1
01010	P2.2
01011	P2.3
01100*	P2.4*
01101*	P2.5*
01110*	P2.6*
01111*	P2.7*
10000	P3.0
10001-11101	Зарезервированы
11110	VREF
11111	GND (АЦП в однофазном режиме)

\*Только для C8051F320; в МК C8051F321 будет «Зарезервировано».

**Рисунок 5.7. ADC0CF: Регистр конфигурации АЦПО**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 11111000 SFR Адрес: 0xBC
AD0SC4	AD0SC3	AD0SC2	AD0SC1	AD0SC0	AD0LJST	-	-	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-3: AD0SC4-0: Биты установки периода сигнала дискретизации АЦПО  
 Частота сигнала дискретизации АЦПО определяется частотой системного тактового сигнала в соответствии со следующим уравнением:

$$AD0SC = (SYSCLK/CLK_{SAR0}) - 1,$$

где AD0SC – 5-разрядное значение, задаваемое битами AD0SC4-0  
 CLK<sub>SAR0</sub> – необходимая частота сигнала дискретизации АЦПО  
 Максимальное значение частоты сигнала дискретизации АЦПО приведено в табл.5.1.

Бит 2: AD0LJST: Бит выравнивания результата преобразования  
 0: Данные в регистровой паре ADC0H:ADC0L выровнены вправо.  
 1: Данные в регистровой паре ADC0H:ADC0L выровнены влево.

Биты 1-0: **Не используются:** читаются как 00b. Запись не оказывает никакого влияния.

**Рисунок 5.8. ADC0H: Регистр старшего байта слова данных АЦПО**

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xBE							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: Старшие биты слова данных АЦПО.  
 Для AD0LJST = 0: Биты 7-2 являются знаковым расширением бита 1. Биты 1-0 представляют собой старшие 2 бита 10-разрядного слова данных АЦПО.  
 Для AD0LJST = 1: Биты 7-0 являются старшими 8 битами 10-разрядного слова данных АЦПО.

Рисунок 5.9. ADC0L: Регистр младшего байта слова данных АЦПО

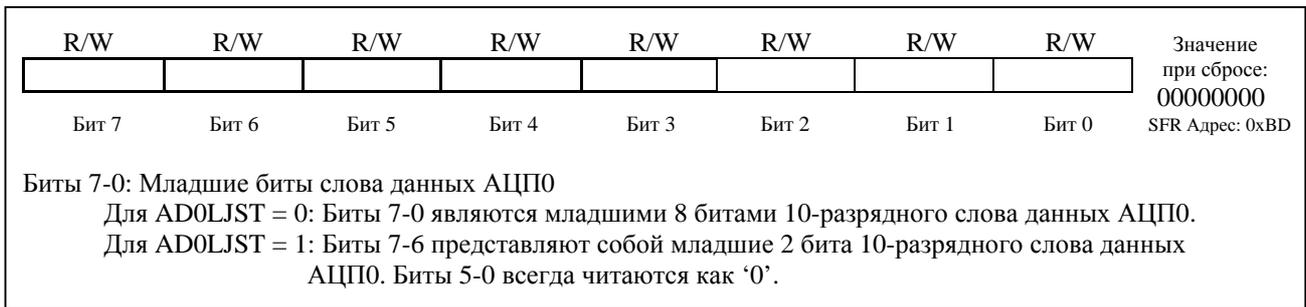


Рисунок 5.10. ADC0CN: Регистр управления АЦПО

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xE8
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM2	AD0CM1	AD0CM0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	
<p><b>Бит7: AD0EN:</b> Бит включения АЦПО  0: АЦПО отключен. АЦПО находится в режиме пониженного энергопотребления.  1: АЦПО включен. АЦПО находится в активном режиме и готов к преобразованию данных.</p> <p><b>Бит 6: AD0TM:</b> Бит установки режима слежения (выборки-хранения) АЦПО  0: Нормальный режим: Когда АЦПО включен, слежение осуществляется всегда, за исключением момента преобразования.  1: Энергосберегающий режим: Режим слежения определяется битами AD0CM2-0 (см. ниже).</p> <p><b>Бит 5: AD0INT:</b> Флаг прерывания от АЦПО (устанавливается при завершении преобразования)  0: АЦПО не закончил преобразование данных (с момента последнего обнуления этого флага)  1: АЦПО закончил преобразование данных</p> <p><b>Бит 4: AD0BUSY:</b> Бит занятости АЦПО  Чтение:  0: Преобразование данных завершено или в данный момент преобразование не осуществляется.  При аппаратном обнулении этого бита флаг AD0INT устанавливается в 1.  1: Идет процесс преобразования данных.  Запись  0: Не оказывает никакого влияния.  1: Иницирует запуск преобразования АЦПО, если биты AD0CM2-0 = 000b</p> <p><b>Бит 3: AD0WINT:</b> Флаг прерывания от детектора диапазона АЦПО.  0: Преобразованные данные не соответствуют заданному диапазону (с момента последнего обнуления этого флага).  1: Преобразованные данные соответствуют заданному диапазону.</p> <p><b>Биты 2-0: AD0CM1-0:</b> Биты выбора режима запуска преобразования АЦПО.  Если AD0TM = 0:  000: Запуск преобразования осуществляется установкой в 1 бита AD0BUSY.  001: Запуск преобразования осуществляется при переполнении Таймера 0.  010: Запуск преобразования осуществляется при переполнении Таймера 2.  011: Запуск преобразования осуществляется при переполнении Таймера 1.  100: Запуск преобразования осуществляется нарастающим фронтом внешнего сигнала CNVSTR.  101: Запуск преобразования осуществляется при переполнении Таймера 3.  11x: Зарезервировано.  Если AD0TM = 1:  000: слежение (выборка) начинается в момент установки в 1 бита AD0BUSY и длится 3 периода сигнала дискретизации АЦПО, затем начинается преобразование данных.  001: слежение (выборка) начинается при переполнении Таймера 0 и длится 3 периода сигнала дискретизации АЦПО, затем начинается преобразование данных.  010: слежение (выборка) начинается при переполнении Таймера 2 и длится 3 периода сигнала дискретизации АЦПО, затем начинается преобразование данных.  011: слежение (выборка) начинается при переполнении Таймера 1 и длится 3 периода сигнала дискретизации АЦПО, затем начинается преобразование данных.  100: слежение (выборка) происходит лишь при низком уровне сигнала на входе CNVSTR; преобразование запускается нарастающим фронтом сигнала на входе CNVSTR.  101: слежение (выборка) начинается при переполнении Таймера 3 и длится 3 периода сигнала дискретизации АЦПО; затем начинается преобразование данных.  11x: Зарезервировано.</p>								

#### 5.4. Программируемый детектор диапазона АЦПО

Программируемый детектор диапазона АЦПО постоянно проверяет результаты преобразований АЦПО на соответствие заданному пользователем диапазону значений и уведомляет систему при обнаружении несоответствия. Это особенно эффективно в управляемых прерываниями системах, т.к. позволяет уменьшить объем кода и улучшить производительность при одновременном уменьшении времени реакции системы. Флаг прерывания от детектора диапазона (бит AD0WINT в регистре ADC0CN) можно использовать также в режиме программного опроса. Старшие и младшие байты граничных значений загружаются в регистры нижней и верхней границ диапазона АЦПО (ADC0GTH, ADC0GTL, ADC0LTH и ADC0LTL). На рис.5.21, рис.5.22, рис.5.23 и рис.5.24 приведены примеры использования детектора диапазона. Следует отметить, что флаг прерывания от детектора диапазона может устанавливаться как при попадании, так и при непопадании результата преобразования в заданный диапазон, в зависимости от значений, записанных в регистры ADC0GTH и ADC0LTH.

Формат значений, записываемых в регистры детектора диапазона, должен соответствовать текущей конфигурации АЦП (выравнивание влево/вправо, со знаком/без знака).

**Рисунок 5.11. ADC0GTH: Регистр старшего байта нижней границы диапазона**

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xC4							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

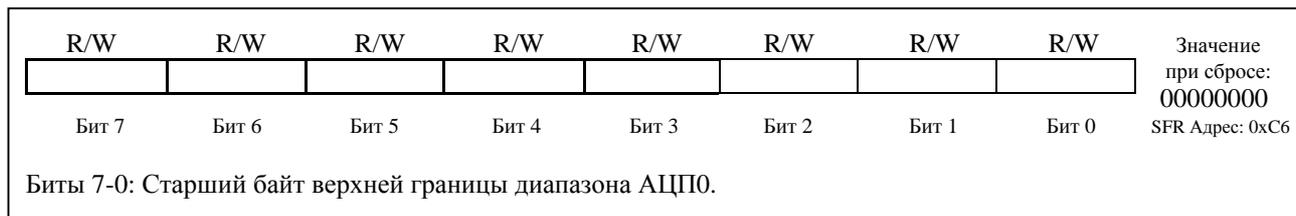
Биты 7-0: Старший байт нижней границы диапазона АЦПО.

**Рисунок 5.12. ADC0GTL: Регистр младшего байта нижней границы диапазона**

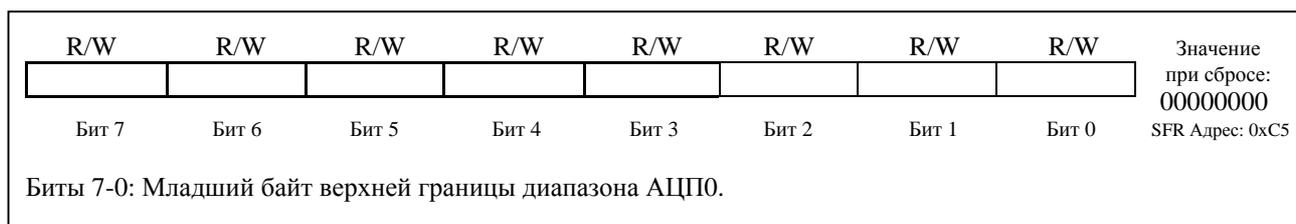
R/W	Значение при сбросе: 11111111 SFR Адрес: 0xC3							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: Младший байт нижней границы диапазона АЦПО.

**Рисунок 5.13. ADC0LTH: Регистр старшего байта верхней границы диапазона**



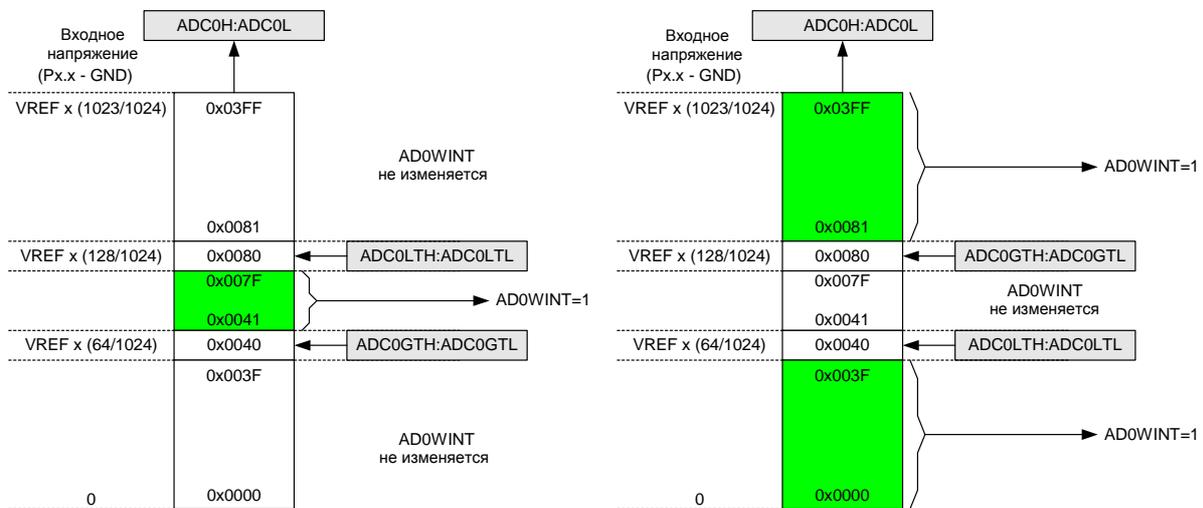
**Рисунок 5.14. ADC0LTL: Регистр младшего байта верхней границы диапазона**



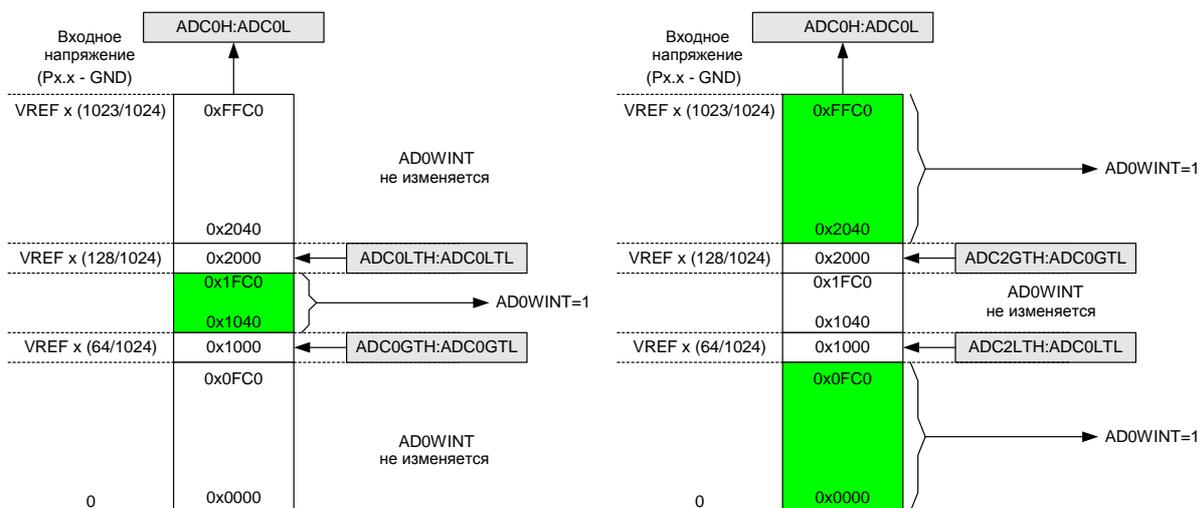
### 5.4.1. Детектор диапазона в однофазном режиме.

На рис.5.15 показаны два примера использования детектора диапазона при измерении однофазного входного сигнала и выравнивании результата преобразования вправо ( $ADC0LTH:ADC0LTL = 0x0080$  (128d) и  $ADC0GTH:ADC0GTL = 0x0040$  (64d)). В однофазном режиме напряжение входного сигнала может быть от 0 до  $VREF * (1023/1024)$  относительно GND, а результат преобразования представлен в виде 10-разрядного целого числа без знака. На примере слева прерывание от флага AD0WINT будет генерироваться в том случае, если результат преобразования АЦП ( $ADC0H:ADC0L$ ) попадает в диапазон, определяемый значениями регистров  $ADC0GTH:ADC0GTL$  и  $ADC0LTH:ADC0LTL$  (т.е. если  $0x0040 < ADC0H:ADC0L < 0x0080$ ). На примере справа прерывание от флага AD0WINT будет генерироваться в том случае, если результат преобразования АЦП ( $ADC0H:ADC0L$ ) не попадает в диапазон, определяемый значениями регистров  $ADC0GTH:ADC0GTL$  и  $ADC0LTH:ADC0LTL$  (т.е. если  $ADC0H:ADC0L < 0x0040$  или  $ADC0H:ADC0L > 0x0080$ ). На рис.5.16 показан пример использования детектора диапазона с такими же значениями регистров границ диапазона при измерении однофазного входного сигнала и выравнивании результата преобразования влево.

**Рисунок 5.15. Пример использования детектора диапазона 10-разрядного АЦП (данные выровнены вправо, вход однофазный)**



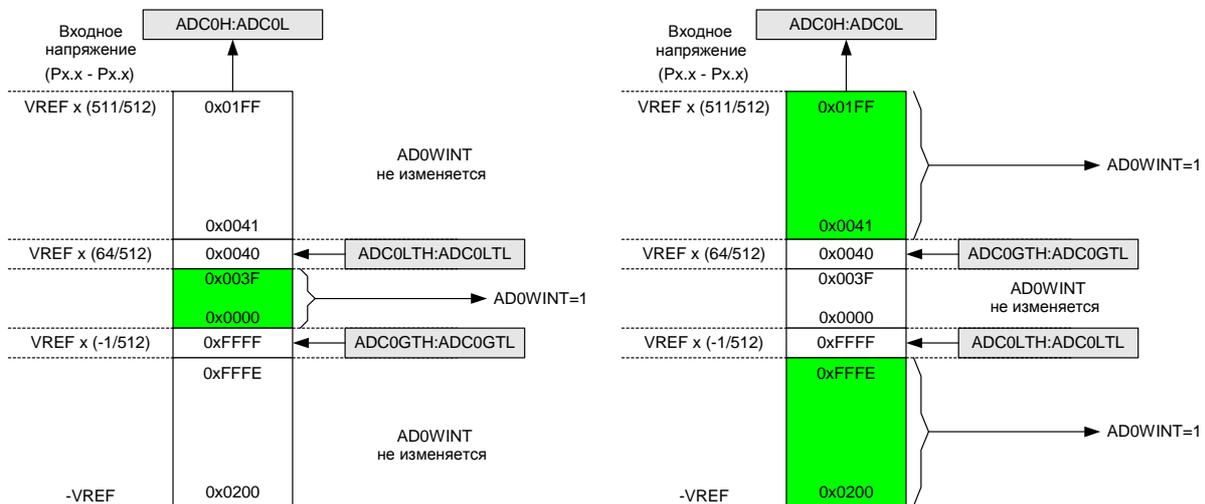
**Рисунок 5.16. Пример использования детектора диапазона 10-разрядного АЦП (данные выровнены влево, вход однофазный)**



## 5.4.2. Детектор диапазона в дифференциальном режиме.

На рис.5.17 показаны два примера использования детектора диапазона при измерении дифференциального входного сигнала и выравнивании результата преобразования вправо (ADC0LTH:ADC0LTL = 0x0040 (+64d) и ADC0GTH:ADC0GTL = 0xFFFF (-1d)). В дифференциальном режиме измеряемое напряжение между дифференциальными входами АЦП может быть от  $-VREF$  до  $VREF * (511/512)$ . Результат преобразования представлен в дополнительном коде в виде 10-разрядного целого числа со знаком. На примере слева прерывание от флага AD0WINT будет генерироваться в том случае, если результат преобразования АЦПО (ADC0H:ADC0L) попадает в диапазон, определяемый значениями регистров ADC0GTH:ADC0GTL и ADC0LTH:ADC0LTL (т.е. если  $0xFFFF (-1d) < ADC0H:ADC0L < 0x0040 (+64d)$ ). На примере справа прерывание от флага AD0WINT будет генерироваться в том случае, если результат преобразования АЦПО (ADC0H:ADC0L) не попадает в диапазон, определяемый значениями регистров ADC0GTH:ADC0GTL и ADC0LTH:ADC0LTL (т.е. если  $ADC0H:ADC0L < 0xFFFF (-1d)$  или  $ADC0H:ADC0L > 0x0040 (+64d)$ ). На рис.5.18 показан пример использования детектора диапазона с такими же значениями регистров границ диапазона при измерении дифференциального входного сигнала и выравнивании результата преобразования влево.

**Рисунок 5.17. Пример использования детектора диапазона 10-разрядного АЦПО (данные выровнены вправо, вход дифференциальный)**



**Рисунок 5.18. Пример использования детектора диапазона 10-разрядного АЦПО (данные выровнены влево, вход дифференциальный)**

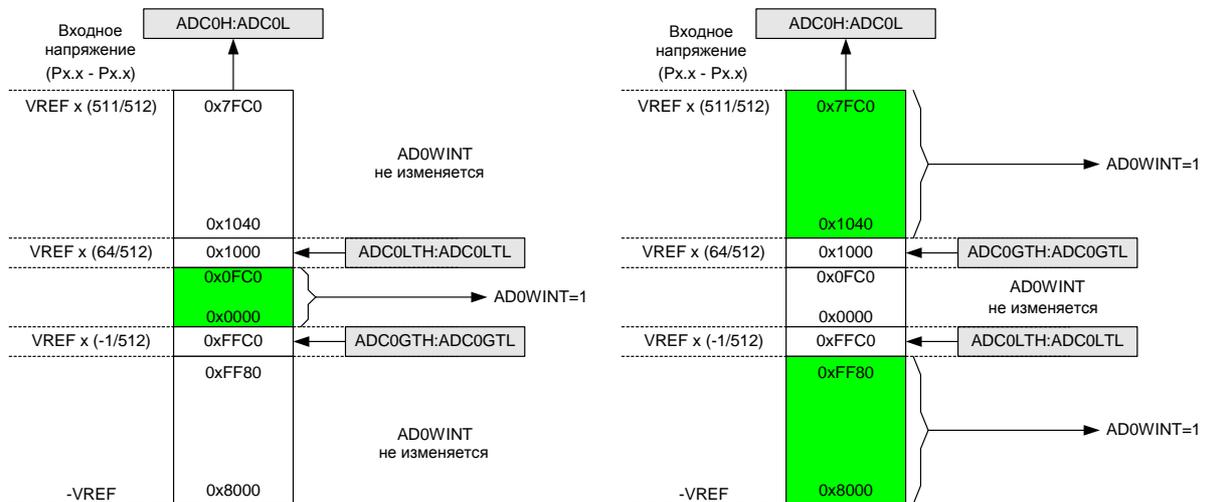


Таблица 5.1. Электрические характеристики 10-разрядного АЦПО

VDD = 3.0В, VREF = 2.4В, T = -40°C ... +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
<b>Точность преобразования</b>					
Разрядность			10		бит
Интегральная нелинейность			±0,5	±1	МЗР
Дифференциальная нелинейность	Монотонность преобразования		±0,5	±1	МЗР
Погрешность смещения нуля			0		МЗР
Погрешность полной шкалы	Дифференциальный режим		- 1		МЗР
Температурный коэффициент смещения нуля			10 <sup>-3</sup>		%/°C
<b>Динамические характеристики (10кГц синусоидальный однофазный входной сигнал, от 0 до -1dB полной шкалы, 200 тыс. преобразований/сек.)</b>					
Отношение сигнал/шум плюс искажение		53	55,5		dB
Общее нелинейное искажение	До 5-й гармоники		-67		dB
Динамический диапазон			78		dB
<b>Скорость преобразования</b>					
Максимальная частота дискретизации				3	МГц
Время преобразования в периодах сигнала дискретизации		10			такты
Время заряда УВХ		300			нс
Производительность				200000	преобр./с
<b>Аналоговые входы</b>					
Диапазон входных напряжений АЦП	Однофазный режим: (AIN+) – GND Дифференциальный режим: (AIN+) – (AIN-)	0 - VREF		VREF VREF	В В
Абсолютное напряжение на выводе относительно GND	Дифференциальный или однофазный режим	0		VDD	В
Входная емкость			5		пФ
<b>Датчик температуры</b>					
Нелинейность	Примечание 1		±0,1		°C
Крутизна характеристики	Примечание 2		2.86		мВ/°C
Смещение нуля	Примечания 1,2 (Температура = 0°C)		776 ±8,5		мВ
<b>Параметры питания</b>					
Ток потребления АЦПО по выв. VDD	Активный режим, 200 тыс. преобразований/сек		400	900	мкА
Нестабильность по напряжению питания			±0.3		мВ/В

Примечание 1: Учитывает нестабильность коэффициента смещения, коэффициента усиления и линейности АЦП.

Примечание 2: Означает одно стандартное отклонение от средней величины.



## 6. Источник опорного напряжения (ИОН)

В качестве опорного напряжения в МК C8051F320/1 можно использовать напряжение от внешнего источника, подаваемое через вывод VREF, напряжение внутреннего ИОН или напряжение питания VDD (см. рис.6.1). Выбор ИОН осуществляется с помощью бита REFSL регистра управления ИОН REF0CN. Для выбора внутреннего ИОН или внешнего напряжения следует сбросить в 0 бит REFSL; для выбора напряжения питания VDD в качестве опорного напряжения следует установить в 1 бит REFSL.

Бит BIASE включает внутренний генератор напряжения смещения АЦП, который необходим для работы АЦП и внутреннего тактового генератора. Этот генератор включается при включении любого из вышеупомянутых модулей. Его также можно включить вручную путем установки в 1 бита BIASE регистра REF0CN (подробное описание регистра REF0CN приведено на рис.6.2). Опорный генератор напряжения смещения (см. рис.6.1) необходим для работы внутреннего ИОН, датчика температуры и умножителя тактовой частоты. Этот генератор автоматически включается при включении любого из вышеупомянутых модулей. Электрические параметры ИОН и схем смещения приведены в табл.6.1.

**Важное замечание относительно использования входа VREF:** Вывод порта P0.7 используется как вход внешнего опорного напряжения VREF. При использовании внешнего опорного напряжения P0.7 должен быть настроен как аналоговый вход и должен пропускаться матрицей при назначении выводов. Чтобы настроить P0.7 как аналоговый вход, следует сбросить в 0 бит 7 регистра POMDIN. Чтобы заставить матрицу пропускать P0.7 при назначении выводов, следует установить в 1 бит 7 регистра POSKIP. Подробная информация о настройке портов ввода/вывода приведена в разделе 14 «Порт ввода/вывода» на стр.127.

Датчик температуры подключается к положительному входу АЦПО через аналоговый мультиплексор (см. раздел 5.1 «Аналоговый мультиплексор» на стр.40). Бит TEMPE регистра REF0CN разрешает/запрещает работу датчика температуры. В случае запрещения датчик температуры по умолчанию переводится в высокоимпедансное состояние. Любые аналого-цифровые измерения показаний запрещенного датчика температуры возвратят незначащие данные.

**Рисунок 6.1. Функциональная схема источника опорного напряжения**

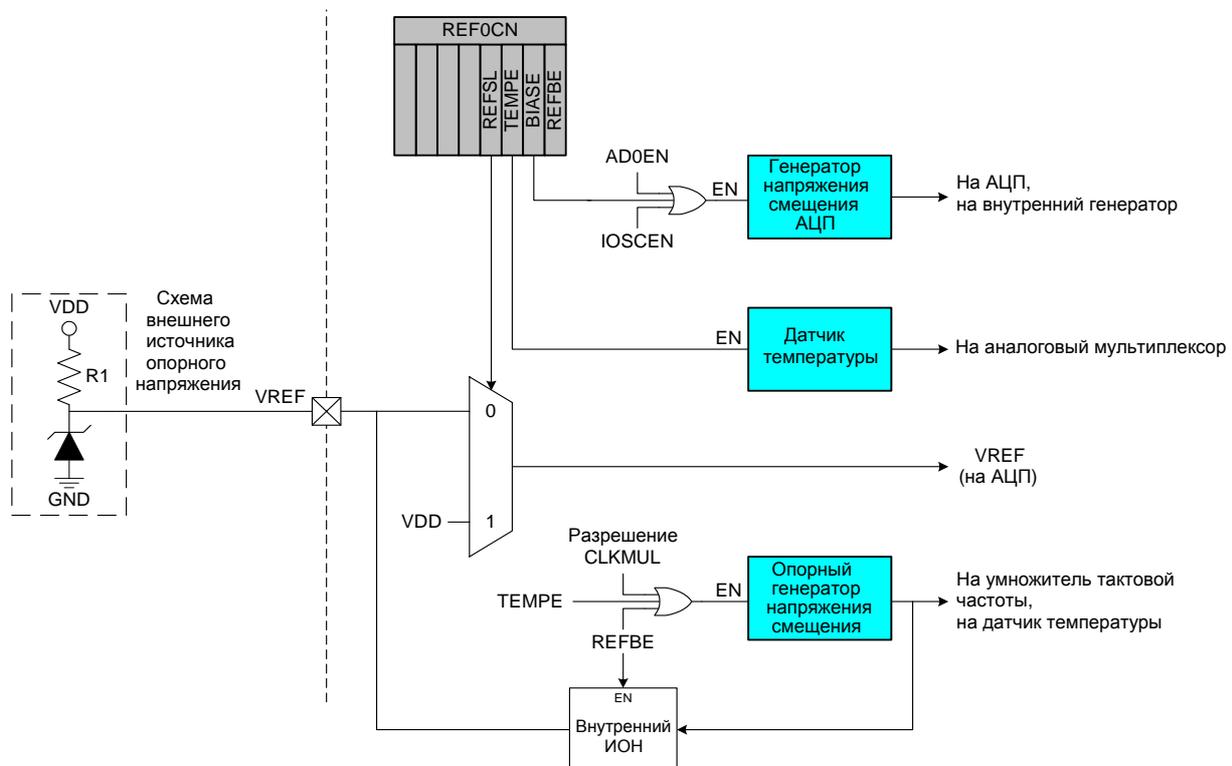


Рисунок 6.2. REF0CN: Регистр управления ИОН

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD1							
-	-	-	-	REFSL	TEMPE	BIASE	REFBE	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: **Не используются**: читаются как 0000b. Запись не оказывает никакого влияния.

Бит 3: REFSL: Бит выбора источника опорного напряжения.  
0: В качестве источника опорного напряжения используется вывод VREF.  
1: В качестве источника опорного напряжения используется напряжение питания VDD.

Бит 2: TEMPE: Бит включения датчика температуры  
0: Внутренний датчик температуры выключен.  
1: Внутренний датчик температуры включен.

Бит 1: BIASE: Бит включения внутреннего аналогового генератора напряжения смещения.  
0: Внутренний генератор напряжения смещения отключен.  
1: Внутренний генератор напряжения смещения включен

Бит 0: REFBE: Бит управления выходным буфером внутреннего ИОН  
0: Буфер внутреннего ИОН выключен.  
1: Буфер внутреннего ИОН включен. Напряжение от внутреннего ИОН подается на вывод VREF.

Таблица 6.1. Электрические характеристики ИОН

VDD = 3.0В, T = от -40°C до +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
<b>Внутренний ИОН (REFBE = 1)</b>					
Выходное напряжение	T <sub>окр. ср.</sub> = 25°C	2.38	2.44	2.50	В
Ток короткого замыкания через вывод VREF				10	мА
Температурный коэффициент нестабильности напряжения на выводе VREF			0,0015		%/°C
Нестабильность тока по нагрузке	Ток нагрузки = (0-200мкА) в цепь GND		1.5 x 10 <sup>-4</sup>		%/мкА
Время стабилизации напряжения на выводе VREF (1)	Танталовый шунтирующий конденсатор емкостью 4.7мкФ, керамический шунтирующий конденсатор емкостью 0.1мкФ		2		мс
Время стабилизации напряжения на выводе VREF (2)	Керамический шунтирующий конденсатор емкостью 0.1мкФ		20		мкс
Время стабилизации напряжения на выводе VREF (3)	Без шунтирующего конденсатора		10		мкс
Нестабильность по напряжению питания			0.014		%/В
<b>Внешний ИОН (REFBE = 0)</b>					
Входное напряжение		0		VDD	В
Входной ток	Частота выборки = 200000 с <sup>-1</sup> VREF = 3.0В		12		мкА
<b>Генераторы напряжений смещения</b>					
Генератор напряжения смещения АЦП	BIASE = '1'		100		мкА
Опорный генератор напряжения смещения			40		мкА



## 7. КОМПАРАТОРЫ

МК C8051F320/1 имеют два встроенных программируемых компаратора напряжения: Компаратор 0 показан на рис.7.1, Компаратор 1 показан на рис.7.2. Эти компараторы функционируют одинаково за исключением следующих моментов:

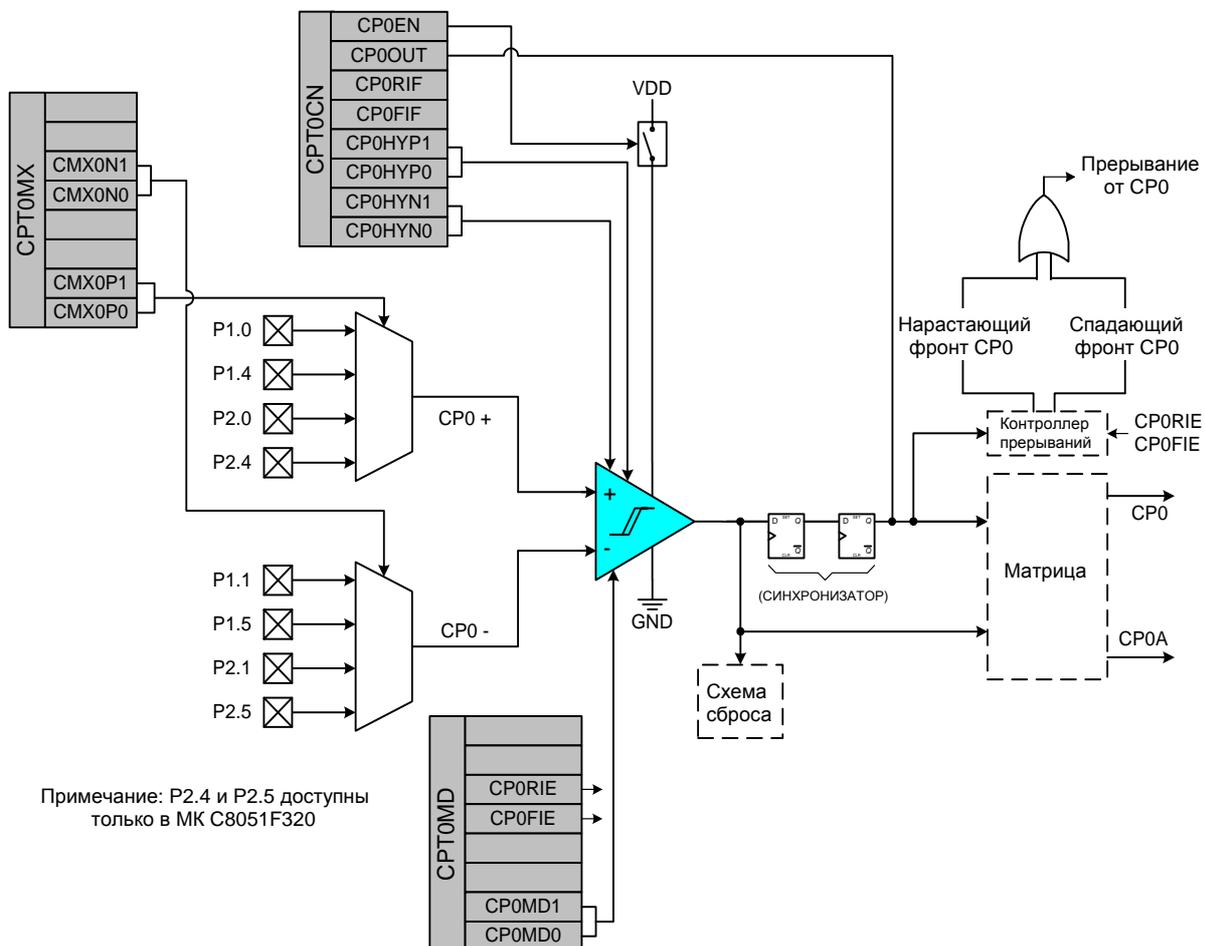
1. В качестве входов компараторов используются различные выводы (см. рис.7.1 и рис.7.2).
2. Компаратор 0 можно использовать в качестве источника сброса.

Время отклика и гистерезис каждого компаратора можно настраивать программно. Каждый компаратор имеет аналоговый входной мультиплексор и два выхода, которые доступны (при необходимости) на выводах порта: синхронный «фиксируемый» выход (CP0, CP1), или асинхронный «сырой» выход (CP0A, CP1A). Асинхронный сигнал доступен даже в том случае, если остановлено тактирование системы. Это позволяет компараторам функционировать и генерировать выходной сигнал при нахождении МК в режиме остановки. Выходы компаратора, разведенные на внешние выводы, можно настроить как выходы с открытым стоком или как цифровые двухтактные выходы (см. раздел 14.2 «Инициализация порта ввода/вывода» на стр.131). Компаратор 0 можно использовать также в качестве источника сброса (см. раздел 10.5 «Сброс от Компаратора 0» на стр.102).

Входы Компаратора 0 выбираются в регистре CPT0MX (см. рис.7.5). Биты CMX0P1-CMX0P0 выбирают положительный вход Компаратора 0; биты CMX0N1-CMX0N0 выбирают отрицательный вход Компаратора 0. Входы Компаратора 1 выбираются в регистре CPT1MX (см. рис.7.8). Биты CMX1P1-CMX1P0 выбирают положительный вход Компаратора 1; биты CMX1N1-CMX1N0 выбирают отрицательный вход Компаратора 1.

**Важное примечание относительно использования входов компараторов:** Выводы порта, выбранные в качестве входов компаратора, должны быть настроены в соответствующих регистрах конфигурации порта как аналоговые входы, а также должны быть настроены как пропускаемые матрицей выводы (см. подробную информацию о настройке порта в разделе 14.3 «Порт ввода/вывода общего назначения» на стр.134).

**Рисунок 7.1. Функциональная схема Компаратора 0**



Выходы компаратора можно опрашивать программно, использовать как источники прерываний и/или развести на вывод порта. Выходы компаратора, разведенные на вывод порта, доступны как асинхронные выходы или синхронные с тактовым сигналом выходы; асинхронный выходной сигнал доступен даже в режиме остановки (когда тактирование системы остановлено). В отключенном состоянии на выходе компаратора по умолчанию устанавливается низкий логический уровень, а ток потребления падает до 100 нА и менее. Подробная информация о настройке выходов компаратора с помощью матрицы приведена в разделе 14.1 «Приоритетный декодер матрицы» на стр.129. На входы компараторов можно подавать напряжение от  $-0.25\text{В}$  до  $(\text{VDD})+0.25\text{В}$ , не опасаясь повреждения МК или сбоев в его работе. Полные электрические характеристики компараторов приведены в таблице 7.1.

Время отклика компараторов можно настроить программно, используя регистры CPTnMD (см. рис.7.6 и рис.7.9). Увеличение времени отклика компаратора позволяет снизить потребляемую им мощность. Временные параметры и данные о потребляемой мощности приведены в таблице 7.1.

**Рисунок 7.2. Функциональная схема Компаратора 1**

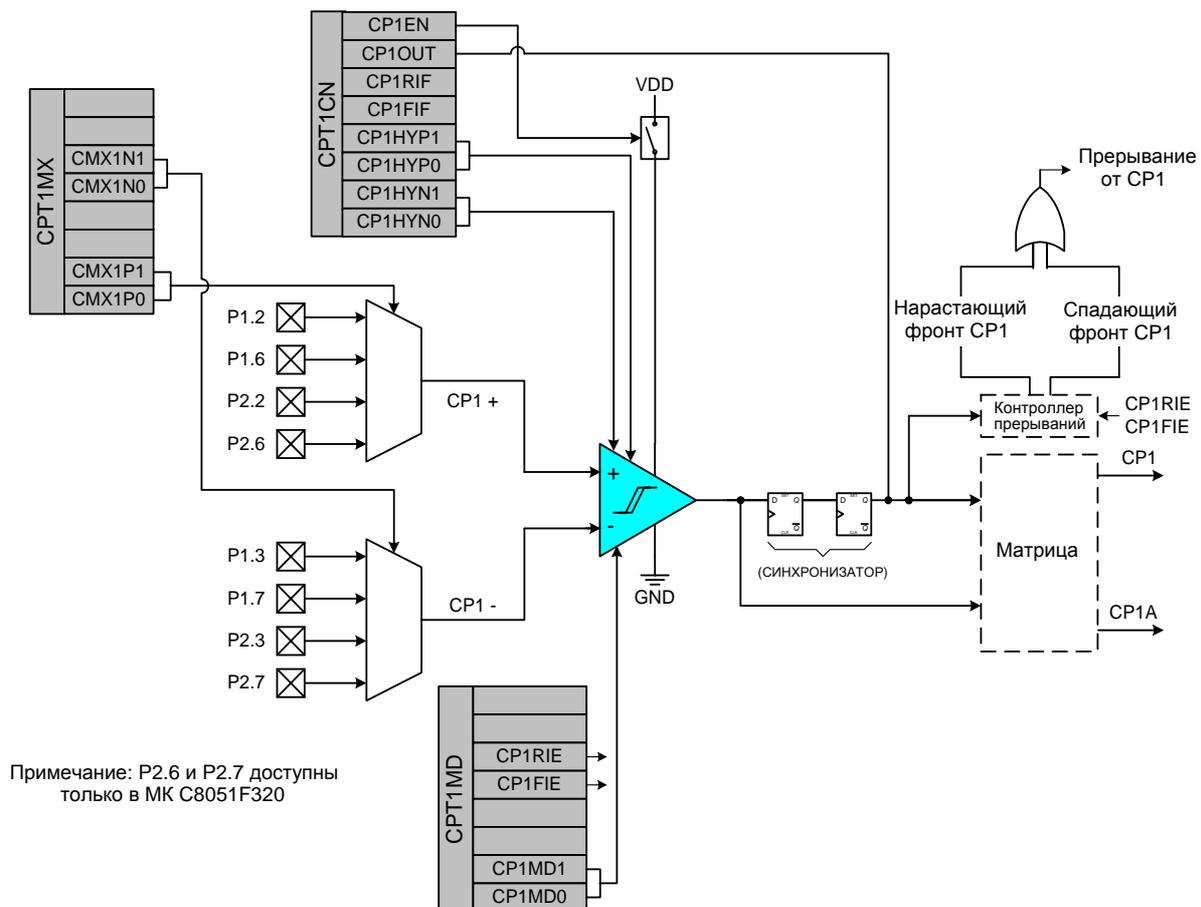
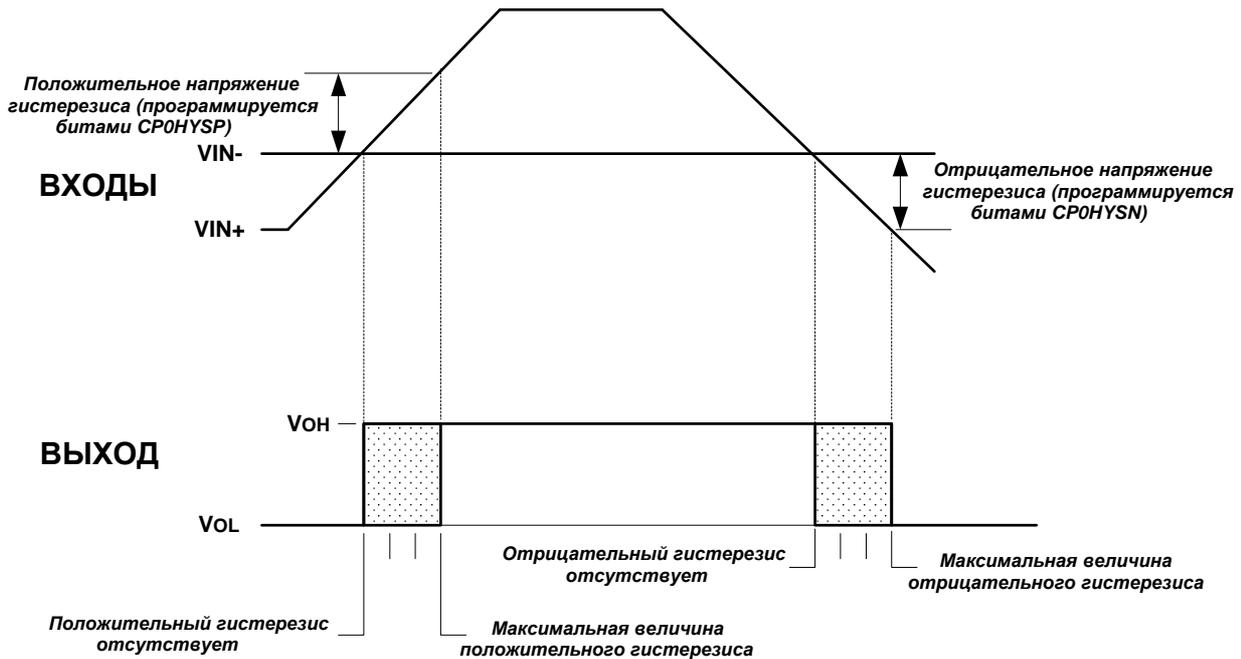
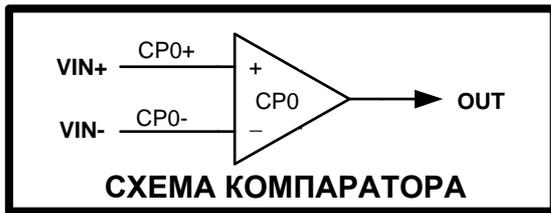


Рисунок 7.3. Гистерезис компаратора



Параметры гистерезиса каждого компаратора настраиваются программно при помощи бит 3-0 в регистрах управления компараторами CPnCN (см. рис.7.4 и рис.7.7). Величина отрицательного напряжения гистерезиса определяется битами CPnHYN. Как показано на рис.7.3, величина отрицательного гистерезиса может быть 20, 10 или 5 мВ, либо отрицательный гистерезис может отсутствовать вовсе. Аналогично величина положительного напряжения гистерезиса определяется битами CPnHYP.

Прерывания от компараторов могут быть сгенерированы по переднему или заднему фронту выходного сигнала. (Обработка прерываний и управление приоритетами описаны в разделе 9.3 «Обработка прерываний» на стр.87). Флаг CPnFIF устанавливается в 1 при возникновении прерывания по заднему фронту выходного сигнала компаратора. Флаг CPnRIF устанавливается в 1 при возникновении прерывания по переднему фронту выходного сигнала компаратора. Эти флаги остаются установленными до тех пор, пока не будут сброшены программно. Состояние выхода компаратора можно получить в любой момент опросом бита CPnOUT. Компаратор включается установкой в 1 соответствующего ему бита CPnEN и отключается сбросом в 0 этого бита.

Рисунок 7.4. CP0CN: Регистр управления Компаратора 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x9B
CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0HYP1	CP0HYP0	CP0HYN1	CP0HYN0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: CP0EN: Бит включения Компаратора 0. 0: Компаратор 0 отключен. 1: Компаратор 0 включен.</p> <p>Бит 6: CP0OUT: Флаг состояния выхода Компаратора 0. 0: Напряжение на вх. CP0+ &lt; CP0-. 1: Напряжение на вх. CP0+ &gt; CP0-.</p> <p>Бит 5: CP0RIF: Флаг прерывания по переднему фронту выходного сигнала Компаратора 0. 0: Прерывания по переднему фронту выходного сигнала Компаратора 0 с момента последнего обнуления флага CP0RIF не было. 1: Произошло прерывание по переднему фронту выходного сигнала Компаратора 0.</p> <p>Бит 4: CP0FIF: Флаг прерывания по заднему фронту выходного сигнала Компаратора 0. 0: Прерывания по заднему фронту выходного сигнала Компаратора 0 с момента последнего обнуления флага CP0RIF не было. 1: Произошло прерывание по заднему фронту выходного сигнала Компаратора 0.</p> <p>Биты 3-2: CP0HYP1-0: Биты управления положительным гистерезисом Компаратора 0. 00: Положительный гистерезис отсутствует. 01: Положительный гистерезис = 5 мВ. 10: Положительный гистерезис = 10 мВ. 11: Положительный гистерезис = 20 мВ.</p> <p>Биты 1-0: CP0HYN1-0: Биты управления отрицательным гистерезисом Компаратора 0. 00: Отрицательный гистерезис отсутствует. 01: Отрицательный гистерезис = 5 мВ. 10: Отрицательный гистерезис = 10 мВ. 11: Отрицательный гистерезис = 20 мВ.</p>								



**Рисунок 7.5. СРТ0МХ: Регистр выбора канала мультиплексора Компаратора 0**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	CMX0N1	CMX0N0	-	-	CMX0P1	CMX0P0	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x9F

Биты 7-6: **Не используются:** читаются как 00b. Запись не оказывает никакого влияния.

Биты 5-4: CMX0N1-CMX0N0: Выбор отрицательного входа Компаратора 0.

Эти биты определяют, какой вывод порта используется в качестве отрицательного входа Компаратора 0.

CMX0N1	CMX0N0	Отрицательный вход
0	0	P1.1
0	1	P1.5
1	0	P2.1
1	1	P2.5*

Биты 3-2: **Не используются:** читаются как 00b. Запись не оказывает никакого влияния.

Биты 1-0: CMX0P1-CMX0P0: Выбор положительного входа Компаратора 0.

Эти биты определяют, какой вывод порта используется в качестве положительного входа Компаратора 0.

CMX0P1	CMX0P0	Положительный вход
0	0	P1.0
0	1	P1.4
1	0	P2.0
1	1	P2.4*

*\*Примечание: P2.4 и P2.5 доступны только в МК C8051F320; в МК C8051F321 будет «Зарезервировано».*

Рисунок 7.6. CP0MD: Регистр выбора режима Компаратора 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	CP0RIE	CP0FIE	-	-	CP0MD1	CP0MD0	00000010
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x9D

Биты 7-6: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

Бит 5: CP0RIE: Флаг разрешения прерывания от нарастающего фронта выходного сигнала Компаратора 0.  
 0: Прерывание от нарастающего фронта выходного сигнала Компаратора 0 запрещено.  
 1: Прерывание от нарастающего фронта выходного сигнала Компаратора 0 разрешено.

Бит 4: CP0FIE: Флаг разрешения прерывания от спадающего фронта выходного сигнала Компаратора 0.  
 0: Прерывание от спадающего фронта выходного сигнала Компаратора 0 запрещено.  
 1: Прерывание от спадающего фронта выходного сигнала Компаратора 0 разрешено.

Биты 3-2: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

Биты 1-0: CP0MD1-CP0MD0: Выбор режима Компаратора 0.  
 Эти биты определяют время отклика Компаратора 0.

Режим	CP0MD1	CP0MD0	Типичное время отклика CP0
0	0	0	100 нс
1	0	1	175 нс
2	1	0	320 нс
3	1	1	1050 нс

**Рисунок 7.7. CPT1CN: Регистр управления Компаратора 1**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x9A
CP1EN	CP1OUT	CP1RIF	CP1FIF	CP1HYP1	CP1HYP0	CP1HYN1	CP1HYN0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: CP1EN: Бит включения Компаратора 1. 0: Компаратор 1 отключен. 1: Компаратор 1 включен.</p> <p>Бит 6: CP1OUT: Флаг состояния выхода Компаратора 1. 0: Напряжение на вх. CP1+ &lt; CP1-. 1: Напряжение на вх. CP1+ &gt; CP1-.</p> <p>Бит 5: CP1RIF: Флаг прерывания по переднему фронту выходного сигнала Компаратора 1. 0: Прерывания по переднему фронту выходного сигнала Компаратора 1 с момента последнего обнуления флага CP1RIF не было. 1: Произошло прерывание по переднему фронту выходного сигнала Компаратора 1.</p> <p>Бит 4: CP1FIF: Флаг прерывания по заднему фронту выходного сигнала Компаратора 1. 0: Прерывания по заднему фронту выходного сигнала Компаратора 1 с момента последнего обнуления флага CP1RIF не было. 1: Произошло прерывание по заднему фронту выходного сигнала Компаратора 1.</p> <p>Биты 3-2: CP1HYP1-0: Биты управления положительным гистерезисом Компаратора 1. 00: Положительный гистерезис отсутствует. 01: Положительный гистерезис = 5 мВ. 10: Положительный гистерезис = 10 мВ. 11: Положительный гистерезис = 20 мВ.</p> <p>Биты 1-0: CP1HYN1-0: Биты управления отрицательным гистерезисом Компаратора 1. 00: Отрицательный гистерезис отсутствует. 01: Отрицательный гистерезис = 5 мВ. 10: Отрицательный гистерезис = 10 мВ. 11: Отрицательный гистерезис = 20 мВ.</p>								

Рисунок 7.8. СРТ1МХ: Регистр выбора канала мультиплексора Компаратора 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x9E
-	-	CMX1N1	CMX1N0	-	-	CMX1P1	CMX1P0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-6: **Не используются**: читаются как 00b. Запись не оказывает никакого влияния.

Биты 5-4: CMX1N1-CMX1N0: Выбор отрицательного входа Компаратора 1.

Эти биты определяют, какой вывод порта используется в качестве отрицательного входа Компаратора 1.

CMX1N1	CMX1N0	Отрицательный вход
0	0	P1.3
0	1	P1.7
1	0	P2.3
1	1	P2.7*

Биты 3-2: **Не используются**: читаются как 00b. Запись не оказывает никакого влияния.

Биты 1-0: CMX1P1-CMX1P0: Выбор положительного входа Компаратора 1.

Эти биты определяют, какой вывод порта используется в качестве положительного входа Компаратора 1.

CMX1P1	CMX1P0	Положительный вход
0	0	P1.2
0	1	P1.6
1	0	P2.2
1	1	P2.6*

*\*Примечание: P2.6 и P2.7 доступны только в МК C8051F320; в МК C8051F321 будет «Зарезервировано».*



**Рисунок 7.9. CP1MD: Регистр выбора режима Компаратора 1**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	CP1RIE	CP1FIE	-	-	CP1MD1	CP1MD0	00000010
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x9C

Биты 7-6: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

Бит 5: CP1RIE: Флаг разрешения прерывания от нарастающего фронта выходного сигнала Компаратора 1.  
 0: Прерывание от нарастающего фронта выходного сигнала Компаратора 1 запрещено.  
 1: Прерывание от нарастающего фронта выходного сигнала Компаратора 1 разрешено.

Бит 4: CP1FIE: Флаг разрешения прерывания от спадающего фронта выходного сигнала Компаратора 1.  
 0: Прерывание от спадающего фронта выходного сигнала Компаратора 1 запрещено.  
 1: Прерывание от спадающего фронта выходного сигнала Компаратора 1 разрешено.

Биты 3-2: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

Биты 1-0: CP1MD1-CP1MD0: Выбор режима Компаратора 1.  
 Эти биты определяют время отклика Компаратора 1.

Режим	CP1MD1	CP1MD0	Типичное время отклика CP1
0	0	0	100 нс
1	0	1	175 нс
2	1	0	320 нс
3	1	1	1050 нс

Таблица 11.1. Электрические характеристики компаратора

VDD = 3.0В, T = - 40°C ... +85°C, если не указано иное.

Все параметры относятся к обоим компараторам (CP0 и CP1), если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Время отклика: Режим 0, $V_{\text{син}}^* = 1.5\text{В}$	$(\text{CP0+}) - (\text{CP0-}) = 100\text{ мВ}$		100		нс
	$(\text{CP0+}) - (\text{CP0-}) = -100\text{ мВ}$		250		нс
Время отклика: Режим 1, $V_{\text{син}}^* = 1.5\text{В}$	$(\text{CP0+}) - (\text{CP0-}) = 100\text{ мВ}$		175		нс
	$(\text{CP0+}) - (\text{CP0-}) = -100\text{ мВ}$		500		нс
Время отклика: Режим 2, $V_{\text{син}}^* = 1.5\text{В}$	$(\text{CP0+}) - (\text{CP0-}) = 100\text{ мВ}$		320		нс
	$(\text{CP0+}) - (\text{CP0-}) = -100\text{ мВ}$		1100		нс
Время отклика: Режим 3, $V_{\text{син}}^* = 1.5\text{В}$	$(\text{CP0+}) - (\text{CP0-}) = 100\text{ мВ}$		1050		нс
	$(\text{CP0+}) - (\text{CP0-}) = -100\text{ мВ}$		5200		нс
Коэффициент подавления синфазного сигнала			1.5	4	мВ/В
Положительный гистерезис 1	CP0HYR1-0 = 00		0	1	мВ
Положительный гистерезис 2	CP0HYN1-0 = 01	2	5	10	мВ
Положительный гистерезис 3	CP0HYR1-0 = 10	7	10	20	мВ
Положительный гистерезис 4	CP0HYN1-0 = 11	15	20	30	мВ
Отрицательный гистерезис 1	CP0HYN1-0 = 00		0	1	мВ
Отрицательный гистерезис 2	CP0HYR1-0 = 01	2	5	7	мВ
Отрицательный гистерезис 3	CP0HYN1-0 = 10	7	10	20	мВ
Отрицательный гистерезис 4	CP0HYR1-0 = 11	15	20	30	мВ
Напряжение на инвертирующем или неинвертирующем входах		-0.25		(VDD) + 0.25	В
Входная емкость			3		пФ
Входной ток смещения			0.001		нА
Входное напряжение смещения		-5		+5	мВ
<b>Параметры питания</b>					
Время включения			10		мкс
Нестабильность напряжения питания			0.1		мВ/В
Ток потребления	Режим 0		7,6		мкА
	Режим 1		3,2		мкА
	Режим 2		1,3		мкА
	Режим 3		0,4		мкА

\*  $V_{\text{син}}$  – напряжение синфазного сигнала на выводах CP0+ и CP0-.

## 8. РЕГУЛЯТОР НАПРЯЖЕНИЯ (REG0)

МК C8051F320/1 имеют встроенный 5В → 3В регулятор напряжения (REG0). Если REG0 включен, то его выходное напряжение появляется на выводе VDD и может использоваться для питания внешних устройств. REG0 можно программно включить/отключить, используя бит REGEN в регистре REG0CN. Электрические характеристики REG0 приведены в таблице 8.1.

Следует иметь в виду, что если МК используется на шине USB, то на вывод VBUS должен быть подан сигнал VBUS. Сигнал VBUS следует подавать на вывод REGIN только в том случае, если МК получает питание от шины USB. Варианты конфигурации REG0 приведены на рис.8.1 – рис.8.4.

## 8.1. Выбор режима регулятора

REG0 может работать в экономичном режиме, который предназначен для использования в том случае, когда МК функционирует в режиме пониженного энергопотребления. В этом экономичном режиме параметры выходного напряжения REG0 соответствуют заданным значениям, однако динамические характеристики REG0 (длительность переходных процессов) ухудшаются. Значения тока потребления REG0 для нормального и экономичного режимов работы приведены в таблице 8.1. Выбор режима REG0 осуществляется с помощью бита REGMOD в регистре REG0CN.



## 8.2. Детектирование VBUS

Если используется USB-контроллер (см. раздел 15 «USB-контроллер (USB0)» на стр.143), то сигнал VBUS должен быть подан на вывод VBUS. Бит VBSTAT в регистре REG0CN показывает текущий логический уровень сигнала VBUS. Если уровень сигнала VBUS совпадет с полярностью, выбранной битом VBPOL в регистре REG0CN, то будет сгенерировано прерывание VBUS, если оно разрешено. Прерывание VBUS является чувствительным к уровню и не имеет связанного с ним флага прерывания. Прерывание VBUS будет активно до тех пор, пока сигнал VBUS совпадает с полярностью, выбранной битом VBPOL. Параметры входа VBUS приведены в таблице 8.1.

**Важное примечание:** Если USB выбран в качестве источника сброса, то в случае совпадения сигнала VBUS с полярностью, выбранной битом VBPOL, будет сгенерирован системный сброс. Подробная информация о выборе USB в качестве источника сброса приведена в разделе 10 «Источники сброса» на стр.99.

**Таблица 8.1. Электрические характеристики регулятора напряжения**

VDD = 3.0В, T = - 40°C ... +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Диапазон входного напряжения		4.0		5.25	В
Выходное напряжение	Выходной ток = 1 ... 100 мА	3.0	3.3	3.6	В
Порог обнаружения VBUS		1.0	1.8	4.0	В
Ток потребления	Нормальный режим (REGMOD = '0')		90	неопр.	мкА
	Экономичный режим (REGMOD = '1')		60	неопр.	

Рисунок 8.1. Конфигурация REG0: Питание от шины USB

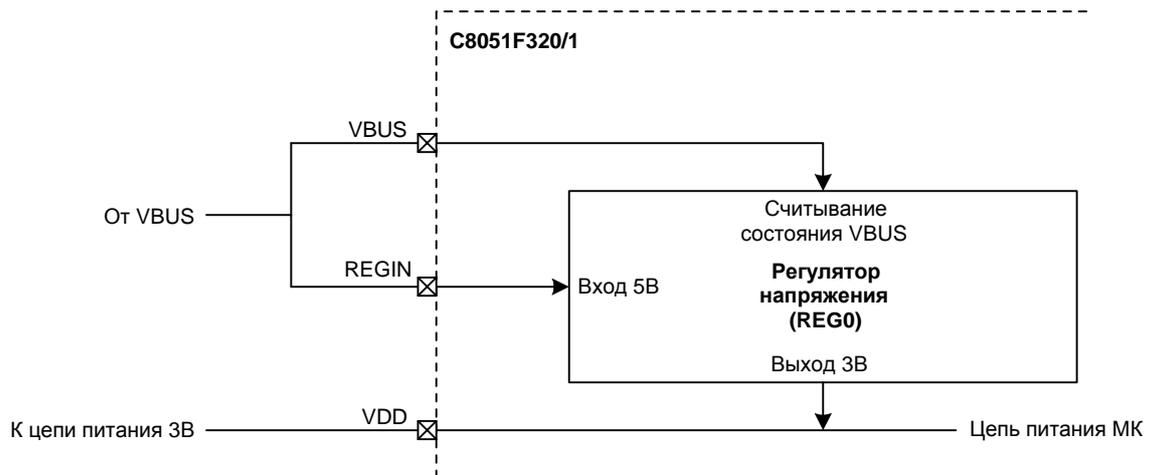
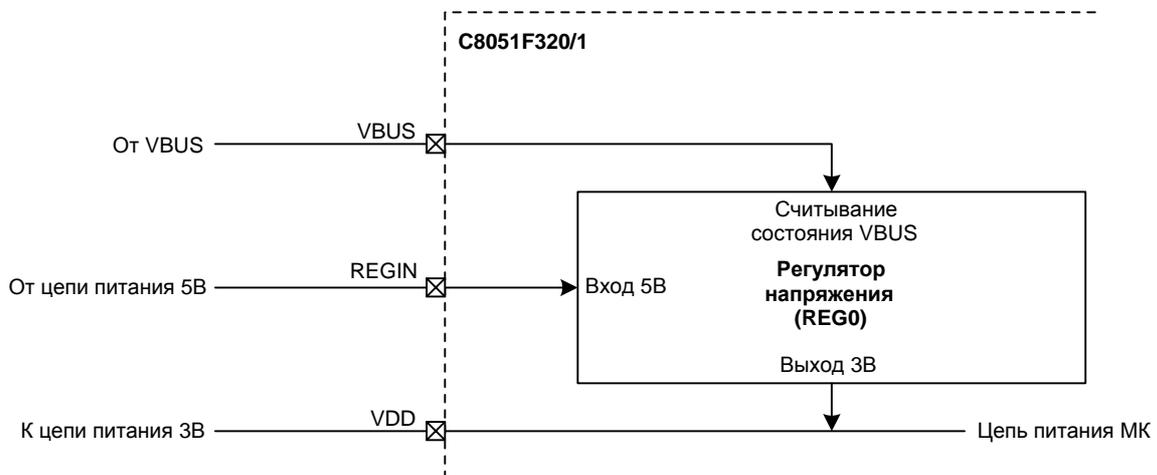
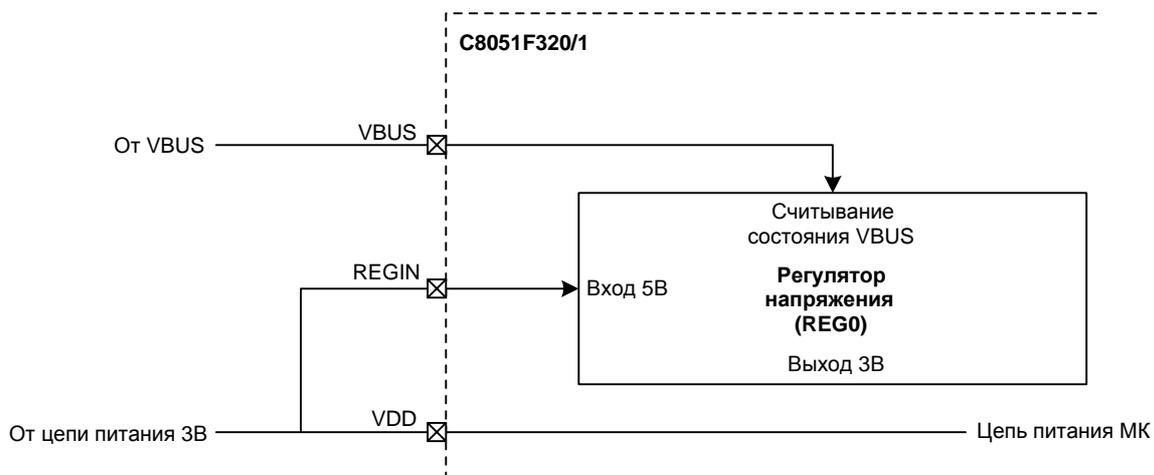


Рисунок 8.2. Конфигурация REG0: Собственный (не USB) источник питания



**Рисунок 8.3. Конфигурация REG0: Собственный (не USB) источник питания, регулятор отключен**



**Рисунок 8.4. Конфигурация REG0: Шина USB не подключена**

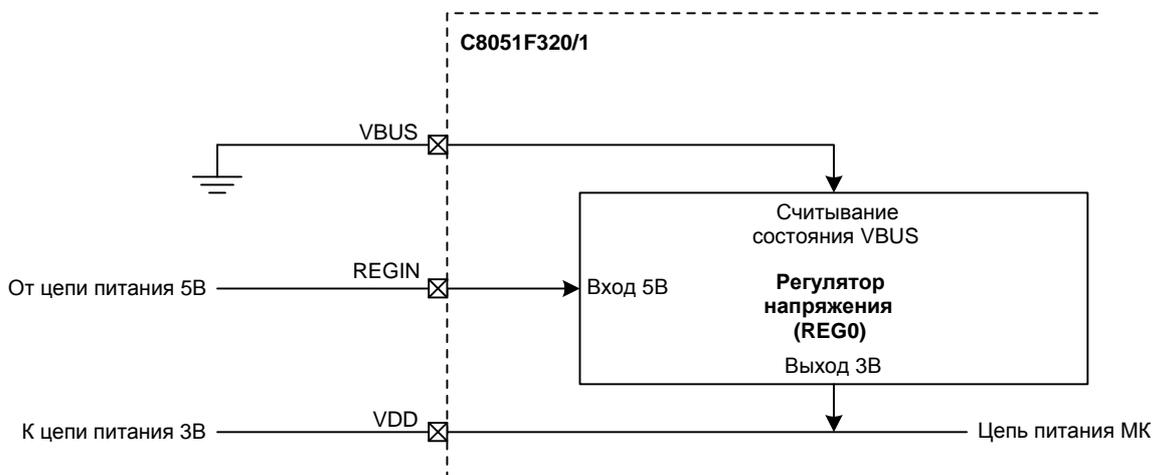


Рисунок 8.5. REG0CN: Регистр управления регулятором напряжения

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
REGDIS	VBSTAT	VBPOL	REGMOD	Зарезерв.	Зарезерв.	Зарезерв.	Зарезерв.	SFR Адрес: 0xC9
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: REGDIS: Бит отключения регулятора напряжения.  
0: Регулятор напряжения включен.  
1: Регулятор напряжения отключен.

Бит 6: VBSTAT: Состояние сигнала VBUS.  
0: Сигнал VBUS в текущий момент отсутствует (МК не подключен к шине USB).  
1: Сигнал VBUS в текущий момент присутствует (МК подключен к шине USB).

Бит 5: VPOL: Выбор полярности прерывания VBUS.  
Этот бит выбирает полярность прерывания VBUS.  
0: Прерывание VBUS активно, если VBUS имеет низкий логический уровень.  
1: Прерывание VBUS активно, если VBUS имеет высокий логический уровень.

Бит 4: REGMOD: Выбор режима регулятора напряжения.  
Этот бит выбирает режим регулятора напряжения. Если REGMOD = '1', то регулятор напряжения функционирует в режиме пониженного энергопотребления (экономичный режим).  
0: Регулятор напряжения USB0 функционирует в нормальном режиме.  
1: Регулятор напряжения USB0 функционирует в режиме пониженного энергопотребления.

Биты 3-0: Зарезервированы: Читаются как 0000b. Должны быть записаны значением 0000b.



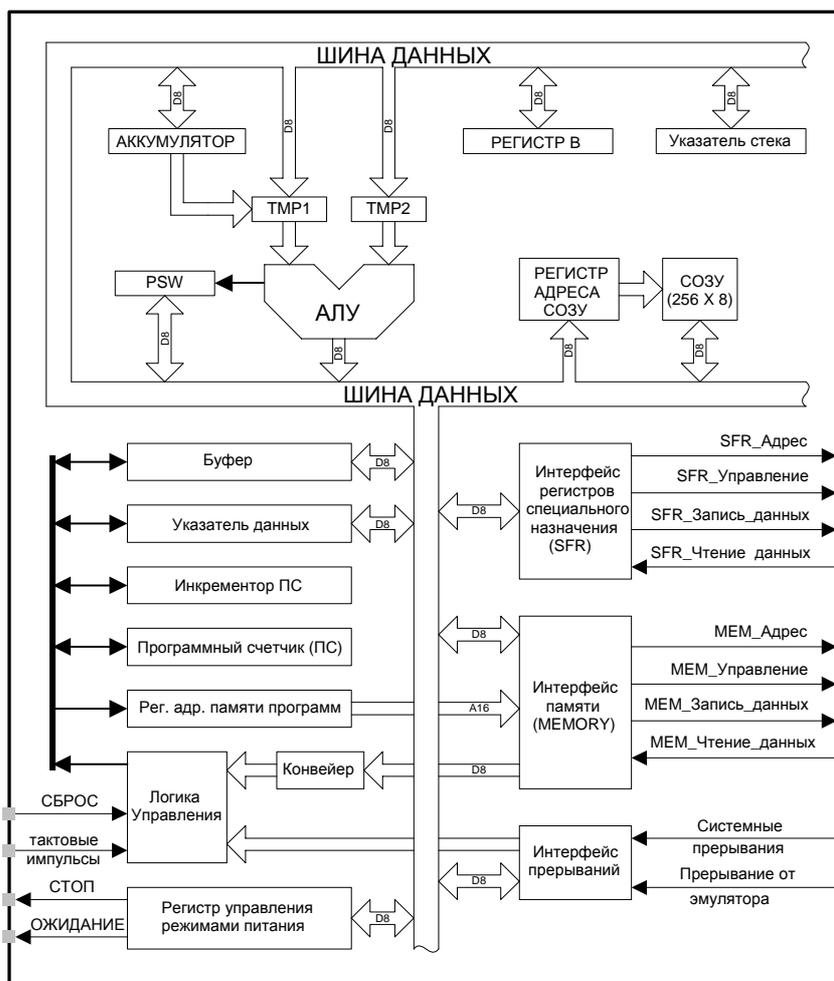
## 9. ПРОЦЕССОРНОЕ ЯДРО CIP-51

МК C8051F320/1 используют процессорное ядро CIP-51, которое по системе команд полностью совместимо с ядром MCS-51<sup>TM</sup>. Для разработки программного обеспечения могут использоваться стандартные (803x/805x) ассемблеры и компиляторы. В состав МК данного семейства входят все периферийные модули, соответствующие стандарту 8051, включая четыре 16-разрядных таймера/счетчика (см. описание в разделе 19), усовершенствованный полнодуплексный УАПП (см. описание в разделе 17), усовершенствованный SPI (см. описание в разделе 18), 256 байт внутреннего ОЗУ, 128 байт адресного пространства для регистров специального назначения (Special Function Register – SFR)(см. раздел 9.2.6), а также 25 портов ввода/вывода (см. описание в разделе 14). Ядро CIP-51 содержит встроенные аппаратные средства отладки (см. описание в разделе 21), а также средства взаимодействия с аналоговыми и цифровыми модулями МК, что позволяет построить на одной микросхеме законченную систему управления или сбора данных.

МК на основе CIP-51 имеют стандартную для архитектуры 8051 структуру и периферию. Кроме этого введены дополнительные специализированные периферийные модули и функции, улучшающие возможности МК (см. рис.9.1). Ниже перечислены основные характеристики ядра CIP-51:

- Полная совместимость с MCS-51 по системе команд
- Пиковая производительность 25 MIPS на частоте 25 МГц
- Тактовая частота от 0 до 25МГц
- 256 байт внутреннего ОЗУ
- 25 портов ввода-вывода
- Развитая система прерываний
- Вход сброса
- Различные режимы энергопотребления
- Встроенные средства отладки
- Защита памяти программ и данных

**Рисунок 9.1. Структурная схема CIP-51**



### Производительность

CIP-51 использует конвейерную архитектуру, что существенно повышает скорость выполнения команд по сравнению со стандартной архитектурой 8051. В МК с архитектурой 8051 все команды, кроме MUL и DIV, исполняются за 12 или 24 тактовых цикла, а максимальная тактовая частота составляет обычно 12 МГц. МК с ядром CIP-51 исполняют 70% своих команд за один или два тактовых цикла, и ни одна команда не требует более восьми тактовых циклов.

При работе на максимальной тактовой частоте 25 МГц производительность ядра CIP-51 может достигать 25 MIPS. Система команд CIP-51 состоит из 109 команд, которые требуют от одного до восьми тактовых циклов:

<b>Количество команд</b>	26	50	5	14	7	3	1	2	1
<b>Количество тактовых циклов</b>	1	2	2/3	3	3/4	4	4/5	5	8

### Средства поддержки программирования и отладки

Последовательный интерфейс C2 предназначен для внутрисистемного программирования Flash-памяти программ и взаимодействия со встроенными средствами отладки. Кроме этого перепрограммируемая Flash-память может быть прочитана и изменена прикладной программой в любое время в побайтном режиме, используя команды MOVC и MOVX. Эта возможность позволяет использовать память программ для долговременного хранения данных, а также обновлять программный код под управлением программы.

Встроенные аппаратные средства отладки позволяют осуществлять внутрисхемную отладку в режиме реального времени, при этом возможны расстановка точек останова, запуск, остановка и пошаговое исполнение программы (включая процедуры обработки прерываний), проверка программного стека, чтение/запись содержимого регистров и памяти. Этот метод отладки является полностью «неразрушающим» и не требует использования внутренних ресурсов МК (например, ОЗУ, стека, таймеров и др.). Подробное описание C2 приведено в разделе 21 «Интерфейс C2» на стр.253.

CIP-51 поддерживается аппаратными и программными средствами разработки от фирмы Silicon Laboratories и других фирм. Фирма Silicon Laboratories предлагает интегрированную среду проектирования (IDE), которая включает в себя редактор, макроассемблер, отладчик и программатор. Отладчик и программатор, входящие в состав IDE, взаимодействуют с CIP-51 посредством интерфейса C2, что позволяет осуществлять быстрое и эффективное программирование МК и его отладку. Доступны также макроассемблеры и C-компиляторы независимых фирм-производителей.



## 9.1. СИСТЕМА КОМАНД

Система команд CIP-51 полностью совместима с системой команд MCS-51™, поэтому разработка программного обеспечения может осуществляться с использованием средств проектирования для стандартной архитектуры 8051. Все команды CIP-51 являются двоичным и функциональным эквивалентом аналогичных команд MCS-51™, включая коды операций, режимы адресации и изменение флагов состояния. Однако, по времени выполнения команды отличаются.

### 9.1.1. Команды и тактирование

Во многих МК с архитектурой 8051 существует различие между машинным циклом и циклом тактирования, при этом машинный цикл длится от 2 до 12 циклов тактирования. Однако, CIP-51 основан исключительно на синхронизации тактовым сигналом и все временные параметры команд приводятся на основе циклов тактирования.

Благодаря конвейерной архитектуре CIP-51, количество тактовых циклов, требуемых для выполнения большинства команд, равно количеству байтов в команде. Команды условных переходов требуют для завершения на один цикл меньше, если переход не происходит (по сравнению с тем случаем, когда переход происходит). Система команд CIP-51 приведена в таблице 9.1, которая содержит мнемонику, количество байтов и количество тактовых циклов для каждой команды.

### 9.1.2. Команда MOVX и память программ

Команда MOVX обычно используется для доступа к внешней памяти данных (Примечание: C8051F320/1 не поддерживают внешнюю память программ или данных). В CIP-51 команда MOVX используется для доступа к внешнему ОЗУ (XRAM) и к адресному пространству встроенной памяти программ, которая реализована в виде перепрограммируемой Flash-памяти. Возможность доступа к Flash-памяти программ обеспечивает механизм, посредством которого программа пользователя может обновлять программный код и использовать область памяти программ для долговременного хранения данных (см. раздел 11 «Flash-память» на стр.107).

Таблица 9.1. Система команд CIP-51

Мнемоника команды	Описание	Байты	Циклы
<b>АРИФМЕТИЧЕСКИЕ ОПЕРАЦИИ</b>			
ADD A,Rn	Сложение аккумулятора с регистром (n = 0...7)	1	1
ADD A,direct	Сложение аккумулятора с прямо-адресуемым байтом	2	2
ADD A,@Ri	Сложение аккумулятора с косвенно-адресуемым байтом ОЗУ	1	2
ADD A,#data	Сложение аккумулятора с константой	2	2
ADDC A,Rn	Сложение аккумулятора с регистром и переносом	1	1
ADDC A,direct	Сложение аккумулятора с прямо-адресуемым байтом и переносом	2	2
ADDC A,@Ri	Сложение аккумулятора с косвенно-адресуемым байтом ОЗУ и переносом	1	2
ADDC A,#data	Сложение аккумулятора с константой и переносом	2	2
SUBB A,Rn	Вычитание из аккумулятора регистра и заема	1	1
SUBB A,direct	Вычитание из аккумулятора прямо-адресуемого байта и заема	2	2
SUBB A,@Ri	Вычитание из аккумулятора косвенно-адресуемого байта ОЗУ и заема	1	2
SUBB A,#data	Вычитание из аккумулятора константы и заема	2	2
INC A	Инкремент аккумулятора	1	1
INC Rn	Инкремент регистра	1	1
INC direct	Инкремент прямо-адресуемого байта	2	2
INC @Ri	Инкремент косвенно-адресуемого байта ОЗУ	1	2
DEC A	Декремент аккумулятора	1	1
DEC Rn	Декремент регистра	1	1
DEC direct	Декремент прямо-адресуемого байта	2	2
DEC @Ri	Декремент косвенно-адресуемого байта ОЗУ	1	2
INC DPTR	Инкремент указателя данных	1	1
MUL AB	Умножение аккумулятора на регистр В	1	4
DIV AB	Деление аккумулятора на регистр В	1	8
DA A	Десятичная коррекция аккумулятора	1	1
<b>ЛОГИЧЕСКИЕ ОПЕРАЦИИ</b>			
ANL A,Rn	Логическое И аккумулятора и регистра	1	1
ANL A,direct	Логическое И аккумулятора и прямо-адресуемого байта	2	2
ANL A,@Ri	Логическое И аккумулятора и косвенно-адресуемого байта ОЗУ	1	2
ANL A,#data	Логическое И аккумулятора и константы	2	2
ANL direct,A	Логическое И прямо-адресуемого байта и аккумулятора	2	2
ANL direct,#data	Логическое И прямо-адресуемого байта и константы	3	3
ORL A,Rn	Логическое ИЛИ аккумулятора и регистра	1	1
ORL A,direct	Логическое ИЛИ аккумулятора и прямо-адресуемого байта	2	2
ORL A,@Ri	Логическое ИЛИ аккумулятора и косвенно-адресуемого байта ОЗУ	1	2
ORL A,#data	Логическое ИЛИ аккумулятора и константы	2	2
ORL direct,A	Логическое ИЛИ прямо-адресуемого байта и аккумулятора	2	2
ORL direct,#data	Логическое ИЛИ прямо-адресуемого байта и константы	3	3
XRL A,Rn	Исключающее ИЛИ аккумулятора и регистра	1	1
XRL A,direct	Исключающее ИЛИ аккумулятора и прямо-адресуемого байта	2	2
XRL A,@Ri	Исключающее ИЛИ аккумулятора и косвенно-адресуемого байта ОЗУ	1	2
XRL A,#data	Исключающее ИЛИ аккумулятора и константы	2	2
XRL direct,A	Исключающее ИЛИ прямо-адресуемого байта и аккумулятора	2	2
XRL direct,#data	Исключающее ИЛИ прямо-адресуемого байта и константы	3	3
CLR A	Сброс аккумулятора	1	1
CPL A	Инверсия аккумулятора	1	1
RL A	Сдвиг аккумулятора влево циклический	1	1
RLC A	Сдвиг аккумулятора влево через перенос	1	1

Мнемоника команды	Описание	Байты	Циклы
RR A	Сдвиг аккумулятора вправо циклический	1	1
RRC A	Сдвиг аккумулятора вправо через перенос	1	1
SWAP A	Обмен местами тетрад в аккумуляторе	1	1
<b>КОМАНДЫ ПЕРЕДАЧИ ДАННЫХ</b>			
MOV A,Rn	Пересылка в аккумулятор из регистра (n = 0...7)	1	1
MOV A,direct	Пересылка в аккумулятор прямо-адресуемого байта	2	2
MOV A,@Ri	Пересылка в аккумулятор косвенно-адресуемого байта ОЗУ	1	2
MOV A,#data	Загрузка в аккумулятор константы	2	2
MOV Rn,A	Пересылка в регистр из аккумулятора	1	1
MOV Rn,direct	Пересылка в регистр прямо-адресуемого байта	2	2
MOV Rn,#data	Загрузка в регистр константы	2	2
MOV direct,A	Пересылка по прямому адресу аккумулятора	2	2
MOV direct,Rn	Пересылка по прямому адресу регистра	2	2
MOV direct,direct	Пересылка прямо-адресуемого байта по прямому адресу	3	3
MOV direct,@Ri	Пересылка косвенно-адресуемого байта ОЗУ по прямому адресу	2	2
MOV direct,#data	Пересылка по прямому адресу константы	3	3
MOV @Ri,A	Пересылка в косвенно-адресуемую ячейку ОЗУ аккумулятора	1	2
MOV @Ri,direct	Пересылка в косвенно-адресуемую ячейку ОЗУ прямо-адресуемого байта	2	2
MOV @Ri,#data	Пересылка в косвенно-адресуемую ячейку ОЗУ константы	2	2
MOV DPTR,#data16	Загрузка указателя данных	3	3
MOVC A,@A+DPTR	Пересылка в аккумулятор байта из памяти программ	1	3
MOVC A,@A+PC	Пересылка в аккумулятор байта из памяти программ	1	3
MOVX A,@Ri	Пересылка в аккумулятор байта из внешней памяти данных	1	3
MOVX @Ri,A	Пересылка байта из аккумулятора во внешнюю память данных	1	3
MOVX A,@DPTR	Пересылка в аккумулятор из расширенной внешней памяти данных	1	3
MOVX @DPTR,A	Пересылка из аккумулятора в расширенную внешнюю память данных	1	3
PUSH direct	Загрузка в стек	2	2
POP direct	Извлечение из стека	2	2
XCH A,Rn	Обмен аккумулятора с регистром	1	1
XCH A,direct	Обмен аккумулятора с прямо-адресуемым байтом	2	2
XCH A,@Ri	Обмен аккумулятора с косвенно-адресуемым байтом ОЗУ	1	2
XCHD A,@Ri	Обмен младшей тетрады аккумулятора с младшей тетрадой косвенно-адресуемого байта ОЗУ	1	2
<b>ОПЕРАЦИИ С БИТАМИ</b>			
CLR C	Сброс переноса	1	1
CLR bit	Сброс бита	2	2
SETB C	Установка переноса	1	1
SETB bit	Установка бита	2	2
CPL C	Инверсия переноса	1	1
CPL bit	Инверсия бита	2	2
ANL C,bit	Логическое И бита и переноса	2	2
ANL C,/bit	Логическое И инверсии бита и переноса	2	2
ORL C,bit	Логическое ИЛИ бита и переноса	2	2
ORL C,/bit	Логическое ИЛИ инверсии бита и переноса	2	2
MOV C,bit	Пересылка бита в перенос	2	2
MOV bit,C	Пересылка переноса в бит	2	2
JC rel	Переход, если перенос равен единице	2	2/3

Мнемоника команды	Описание	Байты	Циклы
JNC rel	Переход, если перенос равен нулю	2	2/3
JB bit,rel	Переход, если бит равен единице	3	3/4
JNB bit,rel	Переход, если бит равен нулю	3	3/4
JBC bit,rel	Переход, если бит установлен, с последующим сбросом бита	3	3/4
ПРОГРАММНЫЕ ПЕРЕХОДЫ			
ACALL addr11	Абсолютный вызов подпрограммы в пределах страницы в 2 Кбайта	2	3
LCALL addr16	Длинный вызов подпрограммы	3	4
RET	Возврат из подпрограммы	1	5
RETI	Возврат из подпрограммы обработки прерывания	1	5
AJMP addr11	Абсолютный переход внутри страницы в 2 Кбайта	2	3
LJMP addr16	Длинный переход в полном объеме памяти программ	3	4
SJMP rel	Короткий относительный переход внутри страницы в 256 байт	2	3
JMP @A+DPTR	Косвенный относительный переход	1	3
JZ rel	Переход, если аккумулятор равен нулю	2	2/3
JNZ rel	Переход, если аккумулятор не равен нулю	2	2/3
CJNE A,direct,rel	Сравнение аккумулятора с прямо-адресуемым байтом и переход, если не равно	3	3/4
CJNE A,#data,rel	Сравнение аккумулятора с константой и переход, если не равно	3	3/4
CJNE Rn,#data,rel	Сравнение регистра с константой и переход, если не равно	3	3/4
CJNE @Ri,#data,rel	Сравнение косвенно-адресуемого байта ОЗУ с константой и переход, если не равно	3	4/5
DJNZ Rn,rel	Декремент регистра и переход, если не нуль	2	2/3
DJNZ direct,rel	Декремент прямо-адресуемого байта и переход, если не нуль	3	3/4
NOP	Холостая команда	1	1

#### Условные обозначения:

**Rn** - Регистр R0-R7 выбранного банка регистров.

**@Ri** – Ячейка ОЗУ данных, адресуемая косвенно через регистры R0-R1

**rel** - 8-битное смещение со знаком (в дополнительном коде) относительно первого байта следующей команды. Используется командой SJMP и всеми командами условных переходов.

**direct** - 8-битный адрес ячейки внутреннего ОЗУ данных. Это может быть ячейка ОЗУ данных прямого доступа (0x00-0x7F) или регистр специального назначения SFR (0x80-0xFF).

**#data** - 8-битная константа

**#data 16** - 16-битная константа

**bit** – Прямо-адресуемый бит ячейки ОЗУ данных или регистра специального назначения SFR.

**addr 11** - 11-битный адрес перехода, используемый командами ACALL и AJMP. Переход должен осуществляться в пределах той 2-Кбайтной страницы памяти программ, в которой расположен первый байт следующей команды.

**addr 16** - 16-битный адрес перехода, используемый командами LCALL и LJMP. Переход может осуществляться в пределах всего 64-Кбайтного пространства памяти программ.

Существует один неиспользуемый код операции (0xA5), который исполняется аналогично команде NOP.

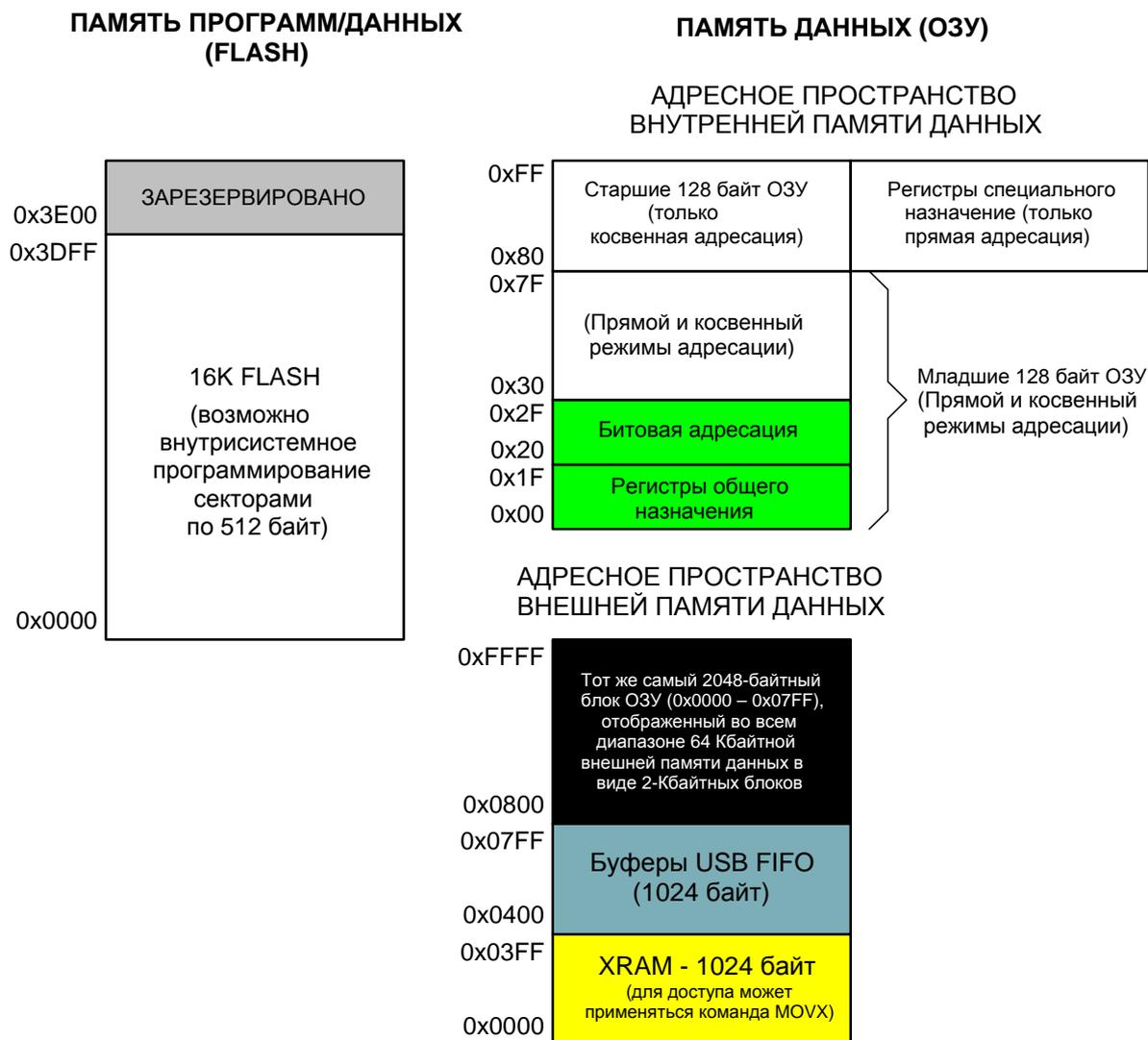
На всю мнемонику распространяется авторское право © Intel Corporation 1980.



## 9.2. ОРГАНИЗАЦИЯ ПАМЯТИ

Организация памяти МК с ядром CIP-51 соответствует стандарту 8051. Имеется две отдельные области памяти, память программ и память данных, которые разделяют одно и то же адресное пространство, но доступ к ним осуществляется командами различного типа. Организация памяти CIP-51 показана на рис.9.2.

**Рисунок 9.2. Карта распределения памяти**



### 9.2.1. Память программ

CIP-51 имеет адресное пространство памяти программ 64 Кбайт. В МК C8051F320/1 физически реализовано 16 Кбайт этой памяти программ, которая является внутрисистемной перепрограммируемой Flash-памятью, занимающей непрерывный блок адресов от 0x0000 до 0x3FFF. Адреса, превышающие 0x3FFF, зарезервированы.

По умолчанию память программ настраивается только для чтения. Однако CIP-51 может записывать данные в память программ (с использованием команды MOVX), для чего необходимо установить в 1 бит разрешения записи памяти программ (PSCTL.0). Эта возможность позволяет CIP-51 обновлять программный код и использовать память программ для долговременного хранения данных. Подробная информация о работе с Flash-памятью приведена в разделе 11 «Flash-память» на стр.107.

### 9.2.2. Память данных

Физически реализовано 256 байт внутреннего ОЗУ, отображенного в пространстве памяти данных с адресами от 0x00 до 0xFF. Младшие 128 байт памяти данных используются для регистров общего назначения (РОН) и сверхоперативного ЗУ (СОЗУ). Для доступа к младшим 128 байтам памяти данных можно использовать либо прямую, либо косвенную адресацию. Ячейки с адресами от 0x00 до 0x1F разбиты на четыре банка РОН, каждый банк состоит из восьми однобайтовых регистров. Следующие 16 байт (0x20 - 0x2F) могут адресоваться побайтно или побитно как 128 бит, доступные в режиме прямой битовой адресации.

Старшие 128 байт памяти данных доступны только в режиме косвенной адресации. Эта область памяти занимает то же самое адресное пространство, что и регистры специального назначения (Special Function Registers - SFR), но физически отделена от них. При обращении к ячейкам памяти с адресами 0x7F - 0xFF использующийся в команде режим адресации определяет, к чему осуществляется доступ: к старшим 128 байтам памяти данных или к SFR. Команды, которые используют режим прямой адресации, будут обращаться к SFR. Команды, использующие режим косвенной адресации, будут обращаться к старшим 128 байтам памяти данных. На рис.9.2 показана организация памяти данных CIP-51.

### 9.2.3. Регистры общего назначения

Младшие 32 байта памяти данных (0x00 - 0x1F) разбиты на четыре банка регистров общего назначения. Каждый банк состоит из восьми однобайтовых регистров, обозначаемых R0-R7. В конкретный момент времени может быть активен лишь один банк, определяемый битами RS0 (PSW.3) и RS1 (PSW.4) в слове состояния программы (program status word) PSW (см. описание PSW на рис.9.6). Это позволяет осуществлять быстрое переключение контекста при вызове подпрограмм и процедур обработки прерываний. Режимы косвенной адресации используют регистры R0 и R1 в качестве индексных регистров.

### 9.2.4. Ячейки памяти с битовой адресацией

Кроме прямого (побайтного) доступа к памяти данных 16 ячеек этой памяти с адресами 0x20 - 0x2F доступны также как 128 индивидуально адресуемых бит. Каждый бит имеет битовый адрес от 0x00 до 0x7F. Бит 0 байта 0x20 имеет битовый адрес 0x00, а бит 7 байта 0x20 имеет битовый адрес 0x07. Бит 7 байта 0x2F имеет битовый адрес 0x7F. Битовый доступ можно отличить от байтового доступа по типу используемой команды (операнды исходных данных и результата в первом случае являются битами, во втором – байтами).

Ассемблер MCS-51™ допускает альтернативную запись для режима битовой адресации в форме XX.B, где XX – адрес байта, а B – позиция бита внутри этого байта. Например, команда:

```
MOV C, 22h.3
```

присваивает значение бита 0x13 (бит 3 в ячейке с адресом 0x22) флагу переноса.

### 9.2.5. Стек

Программный стек может быть размещен в любом месте 256-байтной памяти данных. Область стека определяется с использованием указателя стека (Stack Pointer - SP, 0x81). SP будет указывать на последнюю использованную ячейку. Следующее значение, загружаемое в стек, размещается по адресу SP+1, и затем SP инкрементируется. При сбросе SP инициализируется значением 0x07. Поэтому первое значение, загружаемое в стек, размещается по адресу 0x08, которое также является первым регистром (R0) регистрового банка 1. Таким образом, если требуется использовать более одного банка регистров, SP следует инициализировать адресом ячейки ОЗУ, не используемой для хранения данных. Стек может иметь глубину до 256 байт.



## 9.2.6. Регистры специального назначения

Ячейки памяти данных с адресами от 0x80 до 0xFF, доступные в режиме прямой адресации, образуют регистры специального назначения (special function registers - SFR). SFR позволяют управлять ресурсами ядра CIP-51 и периферийными модулями, а также осуществлять обмен данными с ними. CIP-51 дублирует SFR, типичные для архитектуры 8051, и содержит дополнительные SFR, используемые для настройки подсистем, уникальных для данного семейства МК, и доступа к ним. Это позволяет реализовать новые возможности при сохранении совместимости с системой команд MCS-51™. В таблице 9.2 перечислены все SFR МК на основе CIP-51.

Регистры SFR доступны в любое время, когда для доступа к ячейкам памяти с адресами от 0x80 до 0xFF используется режим прямой адресации. SFR с адресами, оканчивающимися на 0x0 или 0x8 (т.е. P0, TCON, SCON, IE, и т.д.), адресуются как побайтно, так и побитно. Все другие SFR адресуются только побайтно. Незанятые адреса в области SFR зарезервированы для дальнейшего использования. Обращение к ячейкам из этой области даст неопределенный результат и должно быть исключено. Подробное описание каждого регистра приведено на соответствующей странице данного руководства, указанной в табл. 9.3.

**Таблица 9.2. Распределение регистров специального назначения в памяти**

F8	SPI0CN	PCA0L	PCA0H	PCA0CPL0	PCA0CPH0	PCA0CPL4	PCA0CPH4	VDM0CN
F0	B	P0MDIN	P1MDIN	P2MDIN	P3MDIN		EIP1	EIP2
E8	ADC0CN	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2	PCA0CPL3	PCA0CPH3	RSTSRC
E0	ACC	XBR0	XBR1		IT01CF		EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2	PCA0CPM3	PCA0CPM4	
D0	PSW	REF0CN			P0SKIP	P1SKIP	P2SKIP	USB0XCEN
C8	TMR2CN	REG0CN	TMR2RLL	TMR2RLH	TMR2L	TMR2H		
C0	SMB0CN	SMB0CF	SMB0DAT	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH	
B8	IP	CLKMUL	AMX0N	AMX0P	ADC0CF	ADC0L	ADC0H	
B0	P3	OSCXCEN	OSCICN	OSCICL			FLSCL	FLKEY
A8	IE	CLKSEL	EMI0CN					
A0	P2	SPI0CFG	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT	P2MDOUT	P3MDOUT
98	SCON0	SBUF0	CPT1CN	CPT0CN	CPT1MD	CPT0MD	CPT1MX	CPT0MX
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H	USB0ADR	USB0DAT
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON

↑  
 0(8)      1(9)      2(A)      3(B)      4(C)      5(D)      6(E)      7(F)

Доступны в режиме побитной адресации

**Таблица 9.3. Регистры специального назначения**

SFR перечислены в алфавитном порядке. Все неопределенные ячейки в области SFR зарезервированы.

Адрес	Регистр	Описание	№ стр.
0xE0	ACC	Аккумулятор	86
0xBC	ADC0CF	Конфигурация АЦПО	47
0xE8	ADC0CN	Управление АЦПО	49
0xC4	ADC0GTH	Нижняя граница диапазона АЦПО (старший байт)	50
0xC3	ADC0GTL	Нижняя граница диапазона АЦПО (младший байт)	50
0xBE	ADC0H	Слово данных АЦПО (старший байт)	47

Адрес	Регистр	Описание	№ стр.
0xBD	ADC0L	Слово данных АЦПО (младший байт)	48
0xC6	ADC0LTH	Верхняя граница диапазона АЦПО (старший байт)	51
0xC7	ADC0LTL	Верхняя граница диапазона АЦПО (младший байт)	51
0xBA	AMX0N	Выбор отрицательного канала AMUX0	46
0xBB	AMX0P	Выбор положительного канала AMUX0	45
0xF0	B	Регистр B	86
0x8E	CKCON	Управление тактированием таймеров 0, 1, 2 и 3	223
0xA9	CLKSEL	Выбор источника тактирования	125
0x9B	CPT0CN	Управление Компаратором 0	60
0x9D	CPT0MD	Выбор режима Компаратора 0	62
0x9F	CPT0MX	Выбор канала мультиплексора Компаратора 0	61
0x9A	CPT1CN	Управление Компаратором 1	63
0x9C	CPT1MD	Выбор режима Компаратора 1	65
0x9E	CPT1MX	Выбор канала мультиплексора Компаратора 1	64
0x83	DPH	Указатель данных (старший байт)	84
0x82	DPL	Указатель данных (младший байт)	84
0xE6	EIE1	Дополнительный регистр разрешения прерываний 1	92
0xE7	EIE2	Дополнительный регистр разрешения прерываний 2	94
0xF6	EIP1	Дополнительный регистр приоритетов прерываний 1	93
0xF7	EIP2	Дополнительный регистр приоритетов прерываний 2	94
0xAA	EMIOCN	Управление интерфейсом внешней памяти	115
0xB7	FLKEY	Блокировка и ключевой код Flash-памяти	111
0xB6	FLSCL	Делитель модуля Flash-памяти	111
0xA8	IE	Разрешение прерываний	90
0xB8	IP	Управление приоритетами прерываний	91
0xE4	IT01CF	Конфигурация INT0/INT1	95
0xB3	OSCICL	Калибровка внутреннего генератора	119
0xB2	OSICN	Управление внутренним генератором	119
0xB1	OSCXCN	Управление внешним генератором	122
0x80	P0	Регистр-защелка Порты 0	135
0xF1	P0MDIN	Регистр настройки входов Порты 0	135
0xA4	P0MDOUT	Регистр настройки выходов Порты 0	136
0xD4	P0SKIP	Регистр выбора выводов Порты 0, пропускаемых матрицей	136
0x90	P1	Регистр-защелка Порты 1	137
0xF2	P1MDIN	Регистр настройки входов Порты 1	137
0xA5	P1MDOUT	Регистр настройки выходов Порты 1	138
0xD5	P1SKIP	Регистр выбора выводов Порты 1, пропускаемых матрицей	138
0xA0	P2	Регистр-защелка Порты 2	139
0xF3	P2MDIN	Регистр настройки входов Порты 2	139
0xA6	P2MDOUT	Регистр настройки выходов Порты 2	140
0xD6	P2SKIP	Регистр выбора выводов Порты 2, пропускаемых матрицей	140

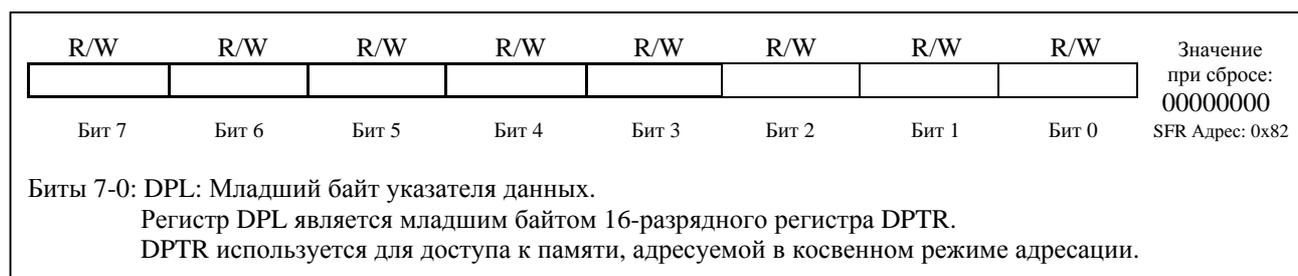
Адрес	Регистр	Описание	№ стр.
0xB0	P3	Регистр-защелка Порты 3	141
0xF4	P3MDIN	Регистр настройки входов Порты 3	141
0xA7	P3MDOUT	Регистр настройки выходов Порты 3	142
0xD8	PCA0CN	Управление программируемым массивом счетчиков (ПМС) 0	248
0xFC	PCA0CPH0	Слово данных модуля захвата 0 (старший байт)	252
0xEA	PCA0CPH1	Слово данных модуля захвата 1 (старший байт)	252
0xEC	PCA0CPH2	Слово данных модуля захвата 2 (старший байт)	252
0xEE	PCA0CPH3	Слово данных модуля захвата 3 (старший байт)	252
0xFE	PCA0CPH4	Слово данных модуля захвата 4 (старший байт)	252
0xFB	PCA0CPL0	Слово данных модуля захвата 0 (младший байт)	252
0xE9	PCA0CPL1	Слово данных модуля захвата 1 (младший байт)	252
0xEB	PCA0CPL2	Слово данных модуля захвата 2 (младший байт)	252
0xED	PCA0CPL3	Слово данных модуля захвата 3 (младший байт)	252
0xFD	PCA0CPL4	Слово данных модуля захвата 4 (младший байт)	252
0xDA	PCA0CPM0	Настройка модуля захват/сравнение 0	250
0xDB	PCA0CPM1	Настройка модуля захват/сравнение 1	250
0xDC	PCA0CPM2	Настройка модуля захват/сравнение 2	250
0xDD	PCA0CPM3	Настройка модуля захват/сравнение 3	250
0xDE	PCA0CPM4	Настройка модуля захват/сравнение 4	250
0xFA	PCA0H	Слово данных таймера/счетчика ПМС (старший байт)	251
0xF9	PCA0L	Слово данных таймера/счетчика ПМС (младший байт)	251
0xD9	PCA0MD	Режим ПМС 0	249
0x87	PCON	Управление питанием	97
0x8F	PSCTL	Управление режимом чтения/записи памяти программ	110
0xD0	PSW	Слово состояния программы	85
0xD1	REF0CN	Регистр управления ИОН	56
0xEF	RSTSRC	Регистр источника сброса	104
0x99	SBUF0	Буфер данных УАППО	199
0x98	SCON0	Управление последовательным портом УАППО	198
0xC1	SMB0CF	Конфигурация модуля SMBus0	182
0xC0	SMB0CN	Управление модулем SMBus0	184
0xC2	SMB0DAT	Регистр данных модуля SMBus 0	186
0x81	SP	Указатель стека	85
0xA1	SPI0CFG	Конфигурация модуля SPI	210
0xA2	SPI0CKR	Установка частоты тактирования модуля SPI	212
0xF8	SPI0CN	Управление модулем SPI	211
0xA3	SPI0DAT	Регистр данных модуля SPI	213
0x88	TCON	Управление таймером/счетчиком	221
0x8C	TH0	Слово данных таймера/счетчика 0 (старший байт)	224
0x8D	TH1	Слово данных таймера/счетчика 1 (старший байт)	224
0x8A	TL0	Слово данных таймера/счетчика 0 (младший байт)	224
0x8B	TL1	Слово данных таймера/счетчика 1 (младший байт)	224

Адрес	Регистр	Описание	№ стр.
0x89	TMOD	Режим таймера/счетчика	222
0xC8	TMR2CN	Управление таймером 2	228
0xCD	TMR2H	Старший байт таймера 2	229
0xCC	TMR2L	Младший байт таймера 2	229
0xCB	TMR2RLH	Старший байт регистра перезагрузки таймера 2	229
0xCA	TMR2RLL	Младший байт регистра перезагрузки таймера 2	229
0x91	TMR3CN	Управление таймером 3	233
0x95	TMR3H	Старший байт таймера 3	234
0x94	TMR3L	Младший байт таймера 3	234
0x93	TMR3RLH	Старший байт регистра перезагрузки таймера 3	234
0x92	TMR3RLL	Младший байт регистра перезагрузки таймера 3	234
0xFF	VDT0CN	Управление схемой слежения за напряжением питания	101
0xE1	XBR0	Конфигурация 1 матрицы портов ввода/вывода	132
0xE2	XBR1	Конфигурация 2 матрицы портов ввода/вывода	133
0x84-0x86, 0xAB-0xAF, 0xB4, 0xB5, 0xBF, 0xC7, 0xCE, 0xCF, 0xD2, 0xD3, 0xDF, 0xE3, 0xE5, 0xF5		Зарезервированы	

### 9.2.7. Описания регистров

Ниже приведены описания регистров SFR, связанных с работой ядра CIP-51. Зарезервированные биты не следует устанавливать в логическую 1. Будущие версии МК могут использовать эти биты для реализации новых функций, тогда при сбросе в эти биты будут записаны логические нули, выбирая тем самым состояние по умолчанию для новых функций. Подробные описания остальных SFR включены в разделы настоящего руководства, посвященные описанию связанных с ними системных модулей и функций.

**Рисунок 9.3. DPL: Младший байт указателя данных**



**Рисунок 9.4. DPH: Старший байт указателя данных**

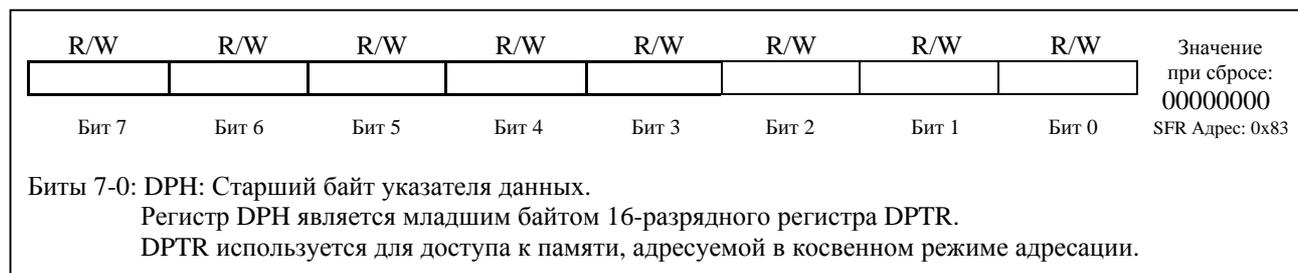


Рисунок 9.5. SP: Указатель стека

R/W	Значение при сбросе: 00000111 SFR Адрес: 0x81							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: SP: Указатель стека.  
Указатель стека содержит адрес вершины стека. Указатель стека инкрементируется перед каждой операцией PUSH. После сброса регистр SP содержит значение по умолчанию 0x07.

Рисунок 9.6. PSW: Слово состояния программы

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD0							
CY	AC	F0	RS1	RS0	OV	F1	PARITY	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Бит 7: CY: Флаг переноса.  
Этот бит устанавливается, если в результате последней арифметической операции произошел перенос (сложение) или заем (вычитание). Он сбрасывается в 0 всеми другими арифметическими операциями.

Бит 6: AC: Флаг десятичного переноса.  
Этот бит устанавливается, если в результате последней арифметической операции произошел перенос (сложение) в старший полубайт или заем (вычитание) из старшего полубайта. Он сбрасывается в 0 всеми другими арифметическими операциями.

Бит 5: F0: Флаг пользователя 0.  
Это доступный в битовом режиме адресации флаг общего назначения, предназначенный для использования под управлением программы.

Биты 4-3: RS1-RS0: Биты выбора банка регистров.  
Эти биты определяют активный банк регистров.

RS1	RS0	Банк регистров	Адреса
0	0	0	0x00-0x07
0	1	1	0x08-0x0F
1	0	2	0x10-0x17
1	1	3	0x18-0x1F

Бит 2: OV: Флаг переполнения.  
Этот бит устанавливается в 1 в следующих случаях:

- если в результате выполнения команды ADD, ADDC или SUBB произошло переполнение с изменением знака;
- если в результате выполнения команды MUL произошло переполнение (результат превышает значение 255);
- если при выполнении команды DIV произошло деление на ноль.

Бит OV сбрасывается в 0 командами ADD, ADDC, SUBB, MUL и DIV во всех других случаях.

Бит 1: F1: Флаг пользователя 1.  
Это доступный в битовом режиме адресации флаг общего назначения, предназначенный для использования под управлением программы.

Бит 0: PARITY: Флаг четности.  
Этот бит устанавливается в 1, если сумма восьми бит в аккумуляторе нечетная и сбрасывается, если сумма четная.

Рисунок 9.7. ACC: Аккумулятор

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xE0							
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: ACC: Аккумулятор  
Этот регистр является аккумулятором для арифметических операций.

Рисунок 9.8. В: Регистр В

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xF0							
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: В: Регистр В  
Этот регистр работает в качестве второго аккумулятора для некоторых арифметических операций.



## 9.3. ОБРАБОТКА ПРЕРЫВАНИЙ

CIP-51 имеет развитую систему прерываний, поддерживающую в общей сложности 16 источников прерываний с двумя уровнями приоритета. Распределение источников прерываний между встроенными периферийными модулями и внешними входными выводами зависит от конкретного типа МК. Каждый источник прерываний имеет один или несколько связанных с ним флагов прерываний, размещенных в SFR. Когда периферийный модуль или внешний источник прерываний регистрирует событие, удовлетворяющее условию прерывания, соответствующий флаг прерывания устанавливается в 1.

Если прерывание от источника прерываний разрешено, то при установке флага прерывания генерируется запрос прерывания. Как только выполнение текущей команды завершится, будет сгенерирована команда LCALL перехода по предопределенному адресу, откуда начнется исполнение процедуры обслуживания прерывания (interrupt service routine - ISR). Каждая ISR должна заканчиваться командой RETI, которая возвращает управление прерванной программе и приводит к выполнению той команды, которая исполнилась бы, если бы запроса прерывания не было. Если прерывания не разрешены, флаг прерывания игнорируется и выполнение программы продолжается в нормальном режиме. (Флаг прерывания устанавливается в 1 независимо от того, разрешены прерывания или запрещены).

Прерывание от каждого источника прерываний может быть разрешено или запрещено с помощью соответствующих битов разрешения прерываний в регистрах SFR (IE-EIE2). Однако, сначала прерывания необходимо разрешить глобально установкой в 1 бита EA (IE.7), только после этого состояние индивидуальных флагов разрешения прерываний будет иметь силу. Сброс в 0 бита EA запрещает прерывания от всех источников прерываний независимо от состояния индивидуальных флагов разрешения прерываний.

Некоторые флаги прерываний сбрасываются автоматически аппаратными средствами при переходе к процедуре ISR. Однако большинство флагов прерываний не сбрасываются аппаратно и должны быть сброшены программно до возвращения из процедуры ISR. Если флаг прерывания остается установленным после завершения выполнения команды возврата из прерывания (RETI), то сразу же будет сгенерирован новый запрос прерывания и после завершения выполнения следующей команды произойдет повторный переход к процедуре ISR.

### 9.3.1. Источники и векторы прерываний

Данное семейство МК поддерживает 16 источников прерываний. Программа может симулировать прерывание установкой в 1 любого флага прерывания. Если прерывание для этого флага разрешено, будет сгенерирован запрос прерывания и произойдет переход по адресу процедуры ISR, связанной с этим флагом прерывания. Источники прерываний МК, соответствующие им адреса прерываний, уровень приоритета и биты управления перечислены в таблице 9.4 на стр.89. Подробная информация относительно условий возникновения прерываний от периферийных модулей и поведения их флагов прерываний приведена в разделах данного руководства, посвященных работе соответствующих периферийных модулей.

### 9.3.2. Внешние прерывания

Два внешних источника прерываний (/INT0 и /INT1) настраиваются как входы с активным низким или активным высоким уровнем, чувствительные к уровню или к фронту сигнала. Биты IN0PL (полярность /INT0) и IN1PL (полярность /INT1) в регистре IT01CF выбирают активный уровень (высокий или низкий); биты IT0 и IT1 регистра TCON (см. раздел 19.1 «Таймер 0 и Таймер 1» на стр.217) определяют чувствительность (к уровню или к фронту). В приведенных ниже таблицах показаны возможные конфигурации.

IT0	IN0PL	Прерывание /INT0
1	0	Активный уровень – низкий, Чувствительность – к фронту.
1	1	Активный уровень – высокий, Чувствительность – к фронту.
0	0	Активный уровень – низкий, Чувствительность – к уровню.
0	1	Активный уровень – высокий, Чувствительность – к уровню.

IT1	IN1PL	Прерывание /INT1
1	0	Активный уровень – низкий, Чувствительность – к фронту.
1	1	Активный уровень – высокий, Чувствительность – к фронту.
0	0	Активный уровень – низкий, Чувствительность – к уровню.
0	1	Активный уровень – высокий, Чувствительность – к уровню.

/INT0 и /INT1 разводятся на выводы порта в соответствии с настройками регистра IT01CF (см. рис.9.15). Следует иметь в виду, что назначение выводов порта /INT0 и /INT1 не зависит от любых назначений выводов, сделанных матрицей. /INT0 и /INT1 будут отслеживать состояние связанных с ними выводов порта, не мешая функционированию периферийных модулей, которым назначены (посредством матрицы) эти выводы. Чтобы назначить вывод(ы) порта только /INT0 и/или /INT1, следует заставить матрицу пропускать эти порты при назначении выводов, настроив ее соответствующим образом. Для этого необходимо установить соответствующий бит в регистре XBR0 (подробная информация о настройке матрицы приведена в разделе 14.1 «Приоритетный декодер матрицы» на стр.129).

Биты IE0 (TCON.1) и IE1 (TCON.3) служат флагами прерываний для внешних прерываний /INT0 и /INT1 соответственно. Если вход /INT0 или /INT1 настроен как чувствительный к фронту, то соответствующий флаг прерывания автоматически сбрасывается аппаратными средствами при переходе к процедуре ISR. Если вход /INT0 или /INT1 настроен как чувствительный к уровню, то флаг прерывания остается установленным в 1 все то время, пока на соответствующем входе присутствует активный уровень сигнала, определяемый соответствующим битом полярности (IN0PL или IN1PL); флаг прерывания сбрасывается в 0, когда вход становится неактивным. Внешний источник прерывания должен удерживать на входе прерывания активный уровень до тех пор, пока запрос прерывания не будет распознан. Затем необходимо деактивировать запрос прерывания до окончания выполнения процедуры ISR, иначе будет сгенерирован другой запрос прерывания.

### 9.3.3. Приоритеты прерываний

Каждому источнику прерываний можно программно присвоить один из двух уровней приоритета: низкий или высокий. Процедура ISR с низким приоритетом может быть прервана прерыванием с высоким приоритетом. Прерывание с высоким приоритетом не может быть прервано. Каждое прерывание имеет связанный с ним бит приоритета в регистрах SFR (IP или EIP2), используемый для настройки уровня приоритета. По умолчанию присваивается низкий приоритет. Если два прерывания происходят одновременно, прерывание с более высоким приоритетом обслуживается первым. Если оба прерывания имеют одинаковый приоритет, то для арбитража используется фиксированный уровень приоритета, приведенный в табл.9.4.

### 9.3.4. Задержка обработки прерывания

Время реакции на прерывание зависит от состояния процессорного ядра в момент возникновения прерывания. Опрос флага прерывания и декодирование приоритета осуществляется каждый системный тактовый цикл. Поэтому, наименее возможное время реакции на прерывание составляет 5 тактовых циклов: 1 цикл для определения прерывания и 4 цикла для выполнения команды LCALL перехода к процедуре ISR. Если в момент выполнения команды RETI появляется прерывание, то до выполнения команды LCALL перехода на процедуру обслуживания этого прерывания будет исполнена одна команда основной программы. Поэтому максимальное время реакции на прерывание (если в настоящий момент не обслуживается другое прерывание или если новое прерывание имеет более высокий приоритет) будет тогда, когда выполняется команда RETI, а следом за ней должна выполняться команда DIV. В этом случае время реакции составляет 18 тактовых циклов: 1 цикл для определения прерывания, 5 циклов для выполнения команды RETI, 8 циклов для выполнения команды DIV и 4 цикла для выполнения команды LCALL перехода на процедуру ISR. Если выполняется процедура ISR для прерывания с равным или более высоким приоритетом, новое прерывание не будет



обслужено до тех пор, пока не завершится текущая процедура ISR, включая команду RETI и следующую команду.

Следует иметь в виду, что процессорное ядро (CPU) останавливается во время выполнения операций записи/стирания Flash-памяти и обращений к USB FIFO с помощью команды MOVX (см. раздел 12.2 «Доступ к USB FIFO» на стр.114). Для прерываний, возникающих во время остановки CPU, время реакции на прерывание будет больше. В этом случае задержка обработки прерывания будет определяться стандартной задержкой ISR (как описано выше) и временем остановки CPU.

**Таблица 9.4. Источники прерываний**

Источник прерывания	Вектор прерывания	Приоритет	Флаг прерывания	Битовая адресация ?	Аппаратный сброс ?	Бит разрешения	Управление приоритетом
Сброс	0x0000	Наивысший	Нет	N/A	N/A	Разрешен всегда	Всегда наивысший
Внешнее прерывание 0 (/INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
Переполнение Таймера 0	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
Внешнее прерывание 1 (/INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
Переполнение Таймера 1	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
Последовательный порт УАППО	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y	N	ES0 (IE.4)	PS0 (IP.4)
Переполнение Таймера 2	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)	PT2 (IP.5)
Модуль SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	Y	N	ESPI0 (IE.6)	PSPI0 (IP.6)
Модуль SMBus0	0x003B	7	SI (SMB0CN.0)	Y	N	ESMB0 (EIE1.0)	PSMB0 (EIP1.0)
USB0	0x0043	8	Специальный	N	N	EUSB0 (EIE1.1)	PUSB0 (EIP1.1)
Детектор диапазона АЦПО	0x004B	9	ADWINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.2)	PWADC0 (EIP1.2)
Завершение преобразования АЦПО	0x0053	10	ADC0INT (ADC0CN.5)	Y	N	EADC0 (EIE1.3)	PADC0 (EIP1.3)
Программируемый массив счетчиков	0x005B	11	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.4)	PPCA0 (EIP1.4)
Компаратор 0	0x0063	12	CP0FIF/CP0RIF (CPT0CN.4/5)	N	N	ECP0 (EIE1.5)	PCP0 (EIP1.5)
Компаратор 1	0x006B	13	CP1FIF/CP1RIF (CPT1CN.4/5)	N	N	ECP1 (EIE1.6)	PCP1 (EIP1.6)
Переполнение Таймера 3	0x0073	14	TF3H (TMR3CN.7) TF3L (TMR3CN.6)	N	N	ET3 (EIE1.7)	PT3 (EIP1.7)
Уровень VBUS	0x007B	15	N/A	N/A	N/A	EVBUS (EIE2.0)	PVBUS (EIP2.0)

### 9.3.5. Описание регистров прерываний

Регистры SFR, используемые для разрешения/запрещения источников прерываний и установки их приоритетов, описаны ниже. Подробная информация относительно условий возникновения прерываний от периферийных модулей и поведения их флагов прерываний приведена в разделах данного руководства, посвященных работе соответствующих периферийных модулей.

**Рисунок 9.9. IE: Регистр разрешения прерываний**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	SFR Адрес: 0xA8
<p><b>Бит 7: EA:</b> Бит разрешения всех прерываний.            Это бит глобально разрешает/запрещает все прерывания. Будучи сброшенным в 0, он перекрывает индивидуальные маски прерываний            0: Все источники прерываний запрещены.            1: Каждое прерывание разрешено/запрещено в соответствии с его индивидуальной маской.</p>								
<p><b>Бит 6: ESPI0:</b> Бит разрешения прерываний от модуля SPI0.            Этот бит устанавливает маскирование прерывания от модуля SPI0.            0: Все прерывания от модуля SPI0 запрещены.            1: Разрешены запросы прерываний, генерируемые при установке флага SPI0.</p>								
<p><b>Бит 5: ET2:</b> Бит разрешения прерывания от Таймера 2.            Этот бит устанавливает маскирование прерывания от Таймера 2.            0: Все прерывания от Таймера 2 запрещены.            1: Разрешены запросы прерываний, генерируемые при установке флагов TF2L или TF2H.</p>								
<p><b>Бит 4: ES0:</b> Бит разрешения прерываний от последовательного порта УАППО.            Этот бит устанавливает маскирование прерывания от последовательного порта УАППО.            0: Прерывания от УАППО запрещены.            1: Прерывания от УАППО разрешены.</p>								
<p><b>Бит 3: ET1:</b> Бит разрешения прерывания от Таймера 1.            Этот бит устанавливает маскирование прерывания от Таймера 1.            0: Все прерывания от Таймера 1 запрещены.            1: Разрешены запросы прерываний, генерируемые при установке флага TF1.</p>								
<p><b>Бит 2: EX1:</b> Бит разрешения внешнего прерывания 1.            Этот бит устанавливает маскирование внешнего прерывания 1.            0: Внешнее прерывание 1 запрещено.            1: Разрешены запросы прерываний, генерируемые сигналом на входе /INT1.</p>								
<p><b>Бит 1: ET0:</b> Бит разрешения прерывания от Таймера 0.            Этот бит устанавливает маскирование прерывания от Таймера 0.            0: Все прерывания от Таймера 0 запрещены.            1: Разрешены запросы прерываний, генерируемые при установке флага TF0.</p>								
<p><b>Бит 0: EX0:</b> Бит разрешения внешнего прерывания 0.            Этот бит устанавливает маскирование внешнего прерывания 0.            0: Внешнее прерывание 0 запрещено.            1: Разрешены запросы прерываний, генерируемые сигналом на входе /INT0.</p>								



Рисунок 9.10. IP: Регистр приоритетов прерываний

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 10000000 SFR Адрес: 0xB8
-	PSPI0	PT2	PS0	PT1	PX1	PT0	PX0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	
<p>Бит 7: Не используется. Читается как 1b. Запись не оказывает никакого влияния.</p> <p>Бит 6: PSPI0: Управление приоритетом прерывания от модуля SPI0. Этот бит устанавливает приоритет прерывания от модуля SPI0. 0: Прерыванию от модуля SPI0 назначается низкий уровень приоритета. 1: Прерыванию от модуля SPI0 назначается высокий уровень приоритета.</p> <p>Бит 5: PT2: Управление приоритетом прерывания от Таймера 2. Этот бит устанавливает приоритет прерываний от Таймера 2. 0: Прерыванию от Таймера 2 назначается низкий уровень приоритета. 1: Прерываниям от Таймера 2 назначается высокий уровень приоритета.</p> <p>Бит 4: PS0: Управление приоритетом прерывания от последовательного порта УАПП0. Этот бит устанавливает приоритет прерываний от последовательного порта УАПП0. 0: Прерываниям от УАПП0 назначается низкий уровень приоритета. 1: Прерываниям от УАПП0 назначается высокий уровень приоритета.</p> <p>Бит 3: PT1: Управление приоритетом прерывания от Таймера 1. Этот бит устанавливает приоритет прерываний от Таймера 1. 0: Прерываниям от Таймера 1 назначается низкий уровень приоритета. 1: Прерываниям от Таймера 1 назначается высокий уровень приоритета.</p> <p>Бит 2: PX1: Управление приоритетом внешнего прерывания 1. Этот бит устанавливает приоритет внешнего прерывания 1. 0: Внешнему прерыванию 1 назначается низкий уровень приоритета. 1: Внешнему прерыванию 1 назначается высокий уровень приоритета.</p> <p>Бит 1: PT0: Управление приоритетом прерывания от Таймера 0. Этот бит устанавливает приоритет прерываний от Таймера 0. 0: Прерываниям от Таймера 0 назначается низкий уровень приоритета. 1: Прерываниям от Таймера 0 назначается высокий уровень приоритета.</p> <p>Бит 0: PX0: Управление приоритетом внешнего прерывания 0. Этот бит устанавливает приоритет внешнего прерывания 0. 0: Внешнему прерыванию 0 назначается низкий уровень приоритета. 1: Внешнему прерыванию 0 назначается высокий уровень приоритета.</p>								

Рисунок 9.11. EIE1: Дополнительный регистр разрешения прерываний 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xE6
ET3	ЕСР1	ЕСР0	ЕРСА0	ЕАDС0	ЕWАDС0	ЕUСВ0	ЕSМВ0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: ET3: Бит разрешения прерывания от Таймера 3. Этот бит устанавливает маскирование прерывания от Таймера 3. 0: Все прерывания от Таймера 3 запрещены. 1: Разрешены запросы прерываний, генерируемые при установке флагов TF3L или TF3H.</p> <p>Бит 6: ECP1: Бит разрешения прерываний от Компаратора 1 (CP1). Этот бит устанавливает маскирование прерывания от CP1. 0: Прерывания от CP1 запрещены. 1: Разрешены запросы прерываний, генерируемые при установке флагов CP1RIF и CP1FIF.</p> <p>Бит 5: ECP0: Бит разрешения прерываний от Компаратора 0 (CP0). Этот бит устанавливает маскирование прерывания от CP0. 0: Прерывания от CP0 запрещены. 1: Разрешены запросы прерываний, генерируемые при установке флагов CP0RIF и CP0FIF.</p> <p>Бит 4: ERSA0: Бит разрешения прерываний от программируемого массива счетчиков (PCA0). Этот бит устанавливает маскирование прерывания от PCA0. 0: Все прерывания от PCA0 запрещены. 1: Разрешены запросы прерываний, генерируемые PCA0.</p> <p>Бит 3: EADC0: Бит разрешения прерываний, возникающих при завершении преобразования АЦП0. Этот бит устанавливает маскирование прерывания, возникающего при завершении преобразования АЦП0. 0: Прерывание, возникающее при завершении преобразования АЦП0, запрещено. 1: Разрешены запросы прерываний, генерируемые при установке флага AD0INT.</p> <p>Бит 2: EWADC0: Бит разрешения прерываний от детектора диапазона АЦП0. Этот бит устанавливает маскирование прерывания от детектора диапазона АЦП0. 0: Прерывание от детектора диапазона АЦП0 запрещено. 1: Разрешены запросы прерываний, генерируемые детектором диапазона АЦП0 (т.е. при установке флага AD0WINT).</p> <p>Бит 1: EUSB0: Бит разрешения прерываний от USB-контроллера (USB0). Этот бит устанавливает маскирование прерывания от USB0. 0: Все прерывания от USB0 запрещены. 1: Разрешены запросы прерываний, генерируемые USB0.</p> <p>Бит 0: ESMB0: Бит разрешения прерываний от модуля SMBus (SMB0). Этот бит устанавливает маскирование прерывания от модуля SMB0. 0: Все прерывания от модуля SMB0 запрещены. 1: Разрешены запросы прерываний, генерируемые модулем SMB0.</p>								



**Рисунок 9.12. EIP1: Дополнительный регистр приоритетов прерываний 1**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xF6
PT3	PCP1	PCP0	PPCA0	PADC0	PWADC0	PUSB0	PSMB0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p><b>Бит 7: PT3:</b> Управление приоритетом прерывания от Таймера 3.  Этот бит устанавливает приоритет прерываний от Таймера 3.  0: Прерыванию от Таймера 3 назначается низкий уровень приоритета.  1: Прерываниям от Таймера 3 назначается высокий уровень приоритета.</p> <p><b>Бит 6: PCP1:</b> Управление приоритетом прерываний от Компаратора 1 (CP1).  Этот бит устанавливает приоритет прерывания от CP1.  0: Прерыванию от CP1 назначается низкий уровень приоритета.  1: Прерыванию от CP1 назначается высокий уровень приоритета.</p> <p><b>Бит 5: PCP0:</b> Управление приоритетом прерываний от Компаратора 0 (CP0).  Этот бит устанавливает приоритет прерывания от CP0.  0: Прерыванию от CP0 назначается низкий уровень приоритета.  1: Прерыванию от CP0 назначается высокий уровень приоритета.</p> <p><b>Бит 4: PPCA0:</b> Управление приоритетом прерывания от программируемого массива счетчиков (PCA0).  Этот бит устанавливает приоритет прерывания от PCA0.  0: Прерыванию от PCA0 назначается низкий уровень приоритета.  1: Прерыванию от PCA0 назначается высокий уровень приоритета.</p> <p><b>Бит 3: PADC0:</b> Управление приоритетом прерывания от флага завершения преобразования АЦПО.  Этот бит устанавливает приоритет прерывания от флага завершения преобразования АЦПО.  0: Прерыванию от флага завершения преобразования АЦПО назначается низкий уровень приоритета.  1: Прерыванию от флага завершения преобразования АЦПО назначается высокий уровень приоритета.</p> <p><b>Бит 2: PWADC0:</b> Управление приоритетом прерывания от детектора диапазона АЦПО.  Этот бит устанавливает приоритет прерывания от детектора диапазона АЦПО.  0: Прерыванию от детектора диапазона АЦПО назначается низкий уровень приоритета.  1: Прерыванию от детектора диапазона АЦПО назначается высокий уровень приоритета.</p> <p><b>Бит 1: PUSB0:</b> Управление приоритетом прерывания от USB-контроллера (USB0).  Этот бит устанавливает приоритет прерывания от модуля USB0.  0: Прерыванию от модуля USB0 назначается низкий уровень приоритета.  1: Прерыванию от модуля USB0 назначается высокий уровень приоритета.</p> <p><b>Бит 0: PSMB0:</b> Управление приоритетом прерывания от модуля SMBus (SMB0).  Этот бит устанавливает приоритет прерывания от модуля SMB0.  0: Прерыванию от модуля SMB0 назначается низкий уровень приоритета.  1: Прерыванию от модуля SMB0 назначается высокий уровень приоритета.</p>								

Рисунок 9.13. EIE2: Дополнительный регистр разрешения прерываний 2

R/W	Значение при сбросе:							
-	-	-	-	-	-	-	EVBUS	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xE7

Биты 7-1: **Не используются.** Читаются как 0000000b. Запись не оказывает никакого влияния.

Бит 0: EVBUS: Бит разрешения прерываний по уровню сигнала VBUS.  
Этот бит устанавливает маскирование прерывания VBUS.  
0: Все прерывания VBUS запрещены.  
1: Разрешены запросы прерываний, генерируемые сигналом VBUS.

Рисунок 9.14. EIP2: Дополнительный регистр приоритетов прерываний 2

R/W	Значение при сбросе:							
-	-	-	-	-	-	-	PVBUS	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xF7

Биты 7-1: **Не используются.** Читаются как 0000000b. Запись не оказывает никакого влияния.

Бит 6: PVBUS: Управление приоритетом прерывания от уровня сигнала VBUS.  
Этот бит устанавливает приоритет прерываний VBUS.  
0: Прерываниям VBUS назначается низкий уровень приоритета.  
1: Прерываниям VBUS назначается высокий уровень приоритета.



Рисунок 9.15. IT01CF: Регистр конфигурации INT0/INT1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000001 SFR Адрес: 0xE4
IN1PL	IN1SL2	IN1SL1	IN1SL0	IN0PL	IN0SL2	IN0SL1	IN0SL0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Примечание: Описание выбора типа чувствительности (к уровню или к фронту) INT0/INT1 приведено на рис.19.4.

Бит 7: IN1PL: Полярность /INT1.  
0: Вход /INT1 имеет низкий активный уровень.  
1: Вход /INT1 имеет высокий активный уровень.

Биты 6-4: IN1SL2-0: Биты выбора выводов порта для /INT1.  
Эти биты определяют, какой вывод порта будет назначен для /INT1. Следует иметь в виду, что назначение этого вывода не зависит от настроек матрицы. /INT1 будет отслеживать состояние связанного с ним вывода порта, не мешая функционированию периферийных модулей, которым назначен (посредством матрицы) этот вывод. Матрица не будет назначать этот вывод порта периферийным модулям, если настроить ее соответствующим образом, заставив пропускать этот порт при назначении выводов (для этого необходимо установить в 1 соответствующий бит в регистре POSKIP).

IN1SL2-0	Вывод порта /INT1
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6
111	P0.7

Бит 3: IN0PL: Полярность /INT0.  
0: Вход /INT0 имеет низкий активный уровень.  
1: Вход /INT0 имеет высокий активный уровень.

Биты 2-0: IN0SL2-0: Биты выбора выводов порта для /INT0.  
Эти биты определяют, какой вывод порта будет назначен для /INT0. Следует иметь в виду, что назначение этого вывода не зависит от настроек матрицы. /INT0 будет отслеживать состояние связанного с ним вывода порта, не мешая функционированию периферийных модулей, которым назначен (посредством матрицы) этот вывод. Матрица не будет назначать этот вывод порта периферийным модулям, если настроить ее соответствующим образом, заставив пропускать этот порт при назначении выводов (для этого необходимо установить в 1 соответствующий бит в регистре POSKIP).

IN0SL2-0	Вывод порта /INT0
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6
111	P0.7

## 9.4. Режимы управления электропитанием

Ядро CIP-51 имеет два программируемых режима управления электропитанием: режим ожидания и режим остановки. В режиме ожидания процессорное ядро (CPU) останавливается, а периферийные модули и тактирование остаются активными. В режиме остановки CPU и внутренний тактовый генератор останавливаются, все источники прерываний неактивны (аналоговые периферийные модули остаются в выбранном для них состоянии; режим работы внешнего генератора не изменяется). После того, как тактовые генераторы переведены в режим ожидания, энергопотребление зависит от системной тактовой частоты и количества периферийных модулей, оставленных в активном режиме до входа в режим ожидания. В режиме остановки энергопотребление наименьшее. На рис.9.16 описан регистр управления питанием (PCON), используемый для настройки режимов электропитания CIP-51.

Хотя CIP-51 имеет встроенные режимы ожидания и остановки (как любой МК со стандартной архитектурой 8051), управление питанием всего МК наиболее эффективно осуществляется путем включения/отключения отдельных периферийных модулей, а также с помощью настройки режима тактирования системы. Каждый аналоговый периферийный модуль, когда он не используется, может быть запрещен и переведен в режим пониженного энергопотребления. Цифровые периферийные модули, такие как таймеры или последовательные интерфейсы, потребляют мало энергии, когда не используются. Отключение генераторов значительно снижает энергопотребление, но после этого требуется сброс для запуска МК.

Внутренний генератор можно перевести в экономичный режим (см. раздел 13 «Генераторы» на стр.117). В экономичном режиме внутренний генератор останавливается до тех пор, пока не произойдет событие USB, требующее обработки, или пока входной сигнал VBUS не совпадет с полярностью, выбранной битом VBPOL в регистре REGOCN (см. рис.8.5 на стр.72).

### 9.4.1. Режим ожидания

Установка в 1 бита выбора режима ожидания (PCON.0) заставит CIP-51 остановить процессорное ядро и перейти в режим ожидания сразу же после завершения команды, которая устанавливает этот бит. Все внутренние регистры и память сохраняют свои данные. Все аналоговые и цифровые периферийные модули могут оставаться активными во время режима ожидания.

Выйти из режима ожидания можно или по сигналу сброса, или по прерыванию. Любой из разрешенных сигналов прерывания приведет к сбросу бита PCON.0 и возобновлению работы процессорного ядра. Прерывание будет обслужено и после выхода из прерывания (RETI) будет исполнена команда, которая следует в программе за командой, установившей бит выбора режима ожидания (PCON.0). Если режим ожидания завершается по внутреннему или внешнему сигналу сброса, то CIP-51 отработает последовательность действий, обычную для сброса, и начнет выполнение программы с адреса 0x0000.

Если сторожевой таймер включен, то со временем он вызовет сброс от сторожевого таймера, что приведет к выходу из режима ожидания. Эта возможность защищает систему от непреднамеренного постоянного отключения в случае случайной записи регистра PCON. Когда такое поведение нежелательно, сторожевой таймер может быть отключен программно до входа в режим ожидания, если первоначально он был настроен на разрешение такой операции. Это обеспечивает возможность дополнительного уменьшения энергопотребления, т.к. система остается в режиме ожидания неопределенно долгое время, ожидая внешнего сигнала пробуждения системы. Подробная информация об использовании и настройке сторожевого таймера приведена в разделе 10.6 «Сброс от сторожевого таймера ПМС» на стр. 102.

### 9.4.2. Режим остановки

Установка в 1 бита выбора режима остановки (PCON.1) заставит CIP-51 перейти в режим остановки сразу же после завершения команды, которая устанавливает этот бит. В режиме остановки внутренний тактовый генератор, процессорное ядро и все цифровые периферийные модули останавливаются; состояние схемы внешнего тактового генератора не изменяется. Каждый аналоговый периферийный модуль (включая схему внешнего тактового генератора) можно отключить индивидуально до перехода в режим остановки. Выйти из режима остановки можно только по внутреннему или внешнему сигналу сброса. При сбросе CIP-51 отработает последовательность действий, обычную для сброса, и начнет выполнение программы с адреса 0x0000.

Если включен детектор исчезновения тактовых импульсов (Missing Clock Detector – MCD), то он вызовет внутренний сброс и тем самым выведет МК из режима остановки. Детектор исчезновения тактовых импульсов следует отключить, если необходимо перевести МК в режим остановки на время, большее времени задержки MCD (100 мкс).



**Рисунок 9.16. PCON: Регистр управления электропитанием**

R/W	Значение при сбросе: 00000000 SFR Адрес: 0x87							
GF5	GF4	GF3	GF2	GF1	GF0	STOP	IDLE	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-2: GF5-GF0: Флаги 5-0 общего назначения.  
 Это флаги общего назначения, доступные для использования в программе пользователя.

Бит 1: STOP: Выбор режима остановки.  
 Установка в 1 этого бита переведет CIP-51 в режим остановки. Этот бит всегда будет читаться как '0'.  
 1: Переход в режим остановки (внутренний тактовый генератор остановлен).

Бит 0: IDLE: Выбор режима ожидания.  
 Установка в 1 этого бита переведет CIP-51 в режим ожидания. Этот бит всегда будет читаться как '0'.  
 1: Переход в режим ожидания. (Отключение тактирования процессорного ядра, однако тактирование таймеров, модулей прерываний, последовательных портов и аналоговых модулей сохраняется).

## Примечания

## 10. ИСТОЧНИКИ СБРОСА

Схема сброса позволяет легко перевести МК в predetermined по умолчанию состояние. При переходе к этому состоянию сброса происходит следующее:

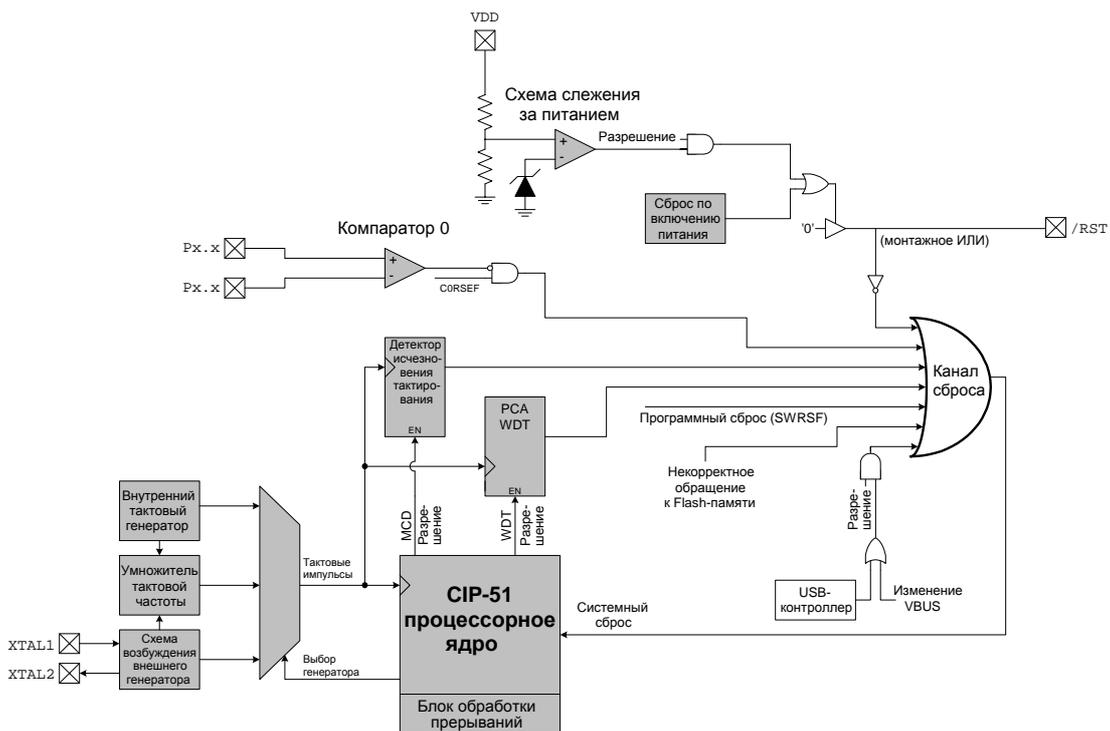
- CIP-51 останавливает выполнение программы;
- регистры SFR инициализируются значениями по умолчанию;
- выводы внешних портов устанавливаются в известное состояние;
- прерывания и таймеры запрещаются.

Все регистры SFR принимают значения по умолчанию. В подробном описании каждого регистра SFR приведено значение, загружаемое в этот регистр при сбросе. Содержимое внутренней памяти данных не изменяется при сбросе и ранее сохраненные данные остаются неизменными. Однако, т.к. регистр указателя стека сбрасывается, стек фактически теряется, хотя данные в нем не изменяются.

Защелки портов ввода/вывода сбрасываются в состояние 0xFF (все логические единицы) и переводятся в режим с открытым стоком. Слаботочковые подтяжки активируются в течение и после сброса. В случае сброса от схемы слежения за напряжением питания или при сбросе типа POR на выводе /RST удерживается низкий логический уровень до выхода МК из состояния сброса.

При выходе из состояния сброса программный счетчик (PC) сбрасывается, а тактирование системы осуществляется от внутреннего генератора. Информация о выборе и настройке источника системного тактового сигнала приведена в разделе 13 «Генераторы» на стр.117. Сторожевой таймер включен и использует тактовый сигнал SYSCLK/12 (подробная информация об использовании сторожевого таймера приведена в разделе 20.3 «Режим сторожевого таймера» на стр.246). Выполнение программы начинается с адреса 0x0000.

**Рисунок 10.1. Структурная схема источников сброса**



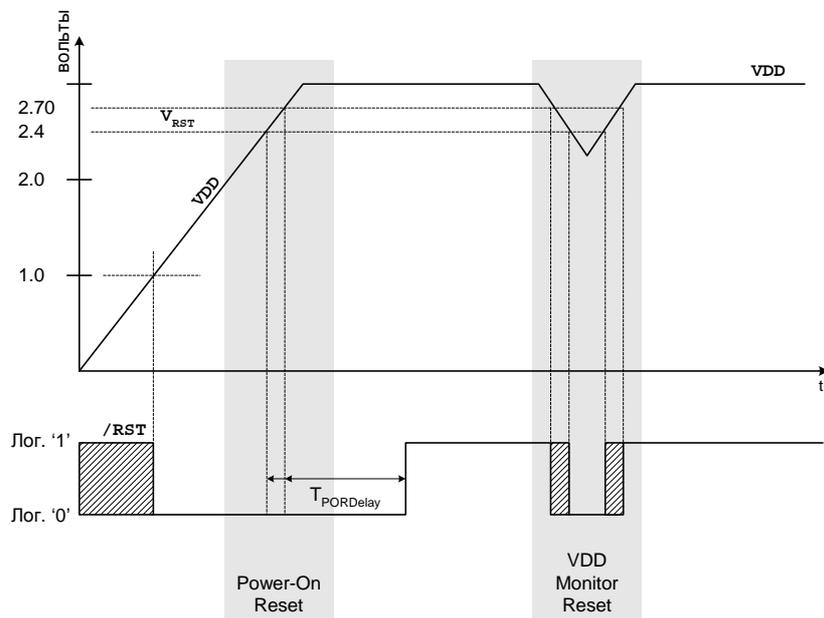
### 10.1. Сброс при включении питания (Power-on Reset - POR)

Во время включения питания МК удерживается в состоянии сброса и на выводе /RST удерживается низкий логический уровень до тех пор, пока напряжение VDD не превысит в процессе включения уровень  $V_{RST}$ . Задержка по включению питания ( $T_{PORDelay}$ ) предшествует выходу МК из состояния сброса; эта задержка обычно не превышает 0.3мс. На рис.10.2 приведены временные диаграммы сброса типа POR и сброса от схемы слежения за напряжением питания.

При завершении сброса типа POR флаг PORSF (RSTSRC.1) аппаратно устанавливается в 1. Если флаг PORSF установлен в 1, то все другие флаги сброса в регистре RSTSRC являются неопределенными. Флаг PORSF сбрасывается в 0 при сбросе от любого другого источника. Т.к. при сбросе любого типа выполнение программы начинается с одного и того же адреса (0x0000), то программа может опросить флаг PORSF, чтобы определить, было ли включение питания причиной сброса. Содержимое внутренней памяти данных после сброса типа POR следует считать неопределенным. После сброса типа POR включается схема слежения за напряжением питания.

Программа может вызвать сброс типа POR, установив в 1 бит PINRFS в регистре RSTSRC.

**Рисунок 10.2. Временная диаграмма работы схемы слежения за напряжением питания**



## 10.2. Сброс при исчезновении питания (Power-fail Reset – PFR)/Схема слежения за напряжением питания

Когда при выключении или сбое питания напряжение питания VDD опускается ниже уровня V<sub>RST</sub>, схема слежения за напряжением питания установит на выводе /RST низкий логический уровень и переведет CIP-51 в состояние сброса (см. рис.10.2). Когда VDD превысит уровень V<sub>RST</sub>, CIP-51 выйдет из состояния сброса. Следует иметь в виду, что, хотя содержимое внутренней памяти данных и не изменяется при сбросе типа PFR, невозможно определить, опускалось ли напряжение VDD ниже уровня, необходимого для сохранения данных. Если флаг PORSF установлен в 1, данные нельзя более считать действительными. Схема слежения за напряжением питания включается после сброса типа POR; однако ее состояние (включена/отключена) на изменяется после сброса от любого другого источника. Например, если схема слежения за напряжением питания включена и произошел программный сброс, то схема слежения за напряжением питания будет оставаться включенной после этого сброса.

**Важное примечание:** Схему слежения за напряжением питания необходимо включить до выбора ее в качестве источника сброса. Выбор схемы слежения за напряжением питания в качестве источника сброса до ее включения и стабилизации приведет к сбросу системы. Ниже приведена процедура настройки схемы слежения за напряжением питания в качестве источника сброса:

Шаг 1. Включить схему слежения за напряжением питания (VDM0CN = '1').

Шаг 2. Ожидать стабилизации схемы слежения за напряжением питания (время включения схемы слежения за напряжением питания приведено в таблице 10.1).

Шаг 3. Выбрать схему слежения за напряжением питания в качестве источника сброса (RSTSRC = '1').

Временная диаграмма работы схемы слежения за напряжением питания приведена на рис.10.2. Электрические характеристики схемы слежения за напряжением питания приведены в таблице 10.1.

### Рисунок 10.3. VDM0CN: Регистр управления схемой слежения за напряжением питания

R/W	R	R	R	R	R	R	R	Значение при сбросе: XXXXXXXX
VDMEN	VDDSTAT	Зарезерв.	Зарезерв.	Зарезерв.	Зарезерв.	Зарезерв.	Зарезерв.	SFR Адрес: 0xFF
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: VDMEN: Включение схемы слежения за напряжением питания.  
 Этот бит включает/отключает схему слежения за напряжением питания. Схема слежения за напряжением питания не может генерировать системные сбросы до тех пор, пока она не будет выбрана также в качестве источника сброса в регистре RSTSCR (см. рис.10.4). Схема слежения за напряжением питания должна стабилизироваться до выбора ее в качестве источника сброса. **Выбор схемы слежения за напряжением питания в качестве источника сброса до ее стабилизации вызовет сброс системы.** Минимальное время включения схемы слежения за напряжением питания приведено в таблице 10.1. Схема слежения за напряжением питания включается после любого сброса типа POR.  
 0: Схема слежения за напряжением питания отключена.  
 1: Схема слежения за напряжением питания включена.

Бит 6: VDDSTAT: Состояние VDD.  
 Этот бит показывает текущее состояние напряжения питания (выход схемы слежения за напряжением питания).  
 0: VDD не превышает порог схемы слежения за напряжением питания.  
 1: VDD превышает схемы слежения за напряжением питания.

Биты 5-0: Зарезервированы. Результат чтения различен. Запись не оказывает никакого влияния.

### 10.3. Внешний сброс

Внешний вывод /RST позволяет внешней схеме перевести МК в состояние сброса. Подача на вывод /RST сигнала с низким активным уровнем заставит МК перейти в состояние сброса. Желательно обеспечить на выводе /RST внешние подтягивающие и/или развязывающие цепи, чтобы предотвратить ложные сбросы, вызванные шумом. Электрические параметры для вывода /RST приведены в таблице 10.1. При завершении внешнего сброса устанавливается в 1 флаг PINRSF (RSTSRC.0).

### 10.4. Сброс от детектора исчезновения тактирования

Детектор исчезновения тактирования (Missing Clock Detector – MCD) фактически представляет собой одновибратор, который управляется системным тактовым сигналом. Если интервал времени между нарастающими фронтами импульсов системного тактового сигнала превысит 100мкс, то одновибратор сработает и сгенерирует сброс. После сброса типа MCD будет установлен в 1 флаг MCDRSF (RSTSRC.2), показывая, что источником сброса был MCD; в иных случаях этот бит читается как 0. Детектор исчезновения тактирования включается установкой бита MCDRSF (RSTSRC.2) в 1 и отключается сбросом в 0 этого бита. Этот сброс не влияет на состояние вывода /RST.

### 10.5. Сброс от Компаратора 0

Установка в 1 флага CORSEF (RSTSRC.5) настраивает Компаратор 0 как вход сброса с низким активным уровнем. До записи бита CORSEF необходимо включить Компаратор 0 и дождаться его стабилизации, чтобы дребезг на выходе не привел к генерации нежелательного сброса. Если напряжение на неинвертирующем входе (CP0+) меньше, чем напряжение на инвертирующем входе (CP0-), то генерируется системный сброс. После сброса от Компаратора 0 флаг CORSEF (RSTSRC.5) будет читаться как 1, показывая, что Компаратор 0 был источником сброса; в иных случаях этот бит читается как 0. Этот сброс не влияет на состояние вывода /RST.

### 10.6. Сброс от сторожевого таймера ПМС

Программируемый сторожевой таймер (WDT) ПМС может использоваться для предотвращения выхода программы из-под контроля в случае системного сбоя. ПМС WDT можно включать/отключать программно, как описано в разделе 20.3 «Режим сторожевого таймера» на стр.246; после любого сброса WDT включается, при этом используется тактовый сигнал SYSCLK/12. Если из-за системного сбоя программа пользователя не может обновить WDT, то генерируется сброс и устанавливается в 1 бит WDTRSF (RSTSRC.5). Этот сброс не влияет на состояние вывода /RST.

### 10.7. Сброс от контроллера Flash-памяти

Если операции чтения/записи/стирания Flash-памяти или чтения памяти программ обращаются по некорректному адресу, то генерируется системный сброс. Это может быть вызвано одной из следующих причин:

- Попытка записать или стереть Flash-память выше пользовательского кодового пространства. Это происходит в том случае, если PSWE = 1 и операция записи, использующая команду MOVX, пытается обратиться по адресу, превышающему 0x3DFF.
- Попытка прочитать Flash-память выше пользовательского кодового пространства. Это происходит в том случае, когда команда MOVC пытается обратиться по адресу, превышающему 0x3DFF.
- Попытка прочитать память программ выше пользовательского кодового пространства. Это происходит в том случае, когда в программе пользователя встречается переход по адресу, превышающему 0x3DFF.
- Доступ к Flash-памяти для операций чтения, записи или стирания ограничен с помощью опций защиты Flash-памяти (см. раздел 11.3 «Защита Flash-памяти» на стр.109).

После сброса, вызванного некорректным обращением к Flash-памяти, устанавливается в 1 бит FERROR (RSTSRC.6). Этот сброс не влияет на состояние вывода /RST.



## 10.8. Программный сброс

Программа может вызвать сброс установкой в 1 бита SWRSF (RSTSRC.4). После сброса, вызванного программой, бит SWRSF будет читаться как '1'. Этот сброс не влияет на состояние вывода /RST.

## 10.9. Сброс от USB-контроллера

Установка в 1 бита USBRSF в регистре RSTSRC выбирает USB0 в качестве источника сброса. Если USB0 выбран в качестве источника сброса, то системный сброс будет генерироваться при любом из следующих случаев:

1. На шине USB обнаружено событие RESET. Чтобы событие RESET распознавалось, USB-контроллер (USB0) должен быть включен. Информация о USB-контроллере приведена в разделе 15 «USB-контроллер (USB0)» на стр. 143.
2. Напряжение на выводе VBUS совпадает с полярностью, выбранной битом VBPOL в регистре REG0CN. Подробная информация о схеме детектирования VBUS приведена в разделе 8 «Регулятор напряжения (REG0)» на стр. 67.

После сброса USB бит USBRSF будет читаться как '1'. Этот сброс не влияет на состояние вывода /RST.

Рисунок 10.4. RSTSRC: Регистр источников сброса

R/W	R	R/W	R/W	R	R/W	R/W	R	Значение при сбросе: XXXXXXXX SFR Адрес: 0xEF
USBRSF	FERROR	CORSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: USBRSF: Флаг сброса от USB.

0: **Чтение:** Последний сброс не был сбросом от USB;

**Запись:** Сброс от USB запрещен.

1: **Чтение:** Последний сброс был сбросом от USB;

**Запись:** Сброс от USB разрешен.

Бит 6: FERROR: Флаг некорректного обращения к Flash-памяти.

0: Источником последнего сброса не является некорректная операция чтения/записи/стирания Flash-памяти.

1: Источником последнего сброса была некорректная операция чтения/записи/стирания Flash-памяти.

Бит 5: CORSEF: Разрешение и флаг сброса от Компаратора 0.

0: **Чтение:** Последний сброс не был сбросом от Компаратора 0;

**Запись:** Компаратор 0 не является источником сброса.

1: **Чтение:** Последний сброс был сбросом от Компаратора 0;

**Запись:** Компаратор 0 является источником сброса (активный уровень - низкий).

Бит 4: SWRSF: Бит инициации и флаг программного сброса

0: **Чтение:** Предыдущий сброс не был вызван установкой в 1 бита SWRSF;

**Запись:** Не вызывает никаких действий.

1: **Чтение:** Предыдущий сброс был вызван установкой в 1 бита SWRSF;

**Запись:** Вызывает системный сброс.

Бит 3: WDTRSF: Флаг сброса от сторожевого таймера

0: WDT не был источником предыдущего сброса.

1: WDT был источником предыдущего сброса.

Бит 2: MCDRSF: Флаг сброса от детектора исчезновения тактирования (MCD)

0: **Чтение:** MCD не был источником предыдущего сброса;

**Запись:** Детектор исчезновения тактирования отключен.

1: **Чтение:** MCD был источником предыдущего сброса;

**Запись:** Детектор исчезновения тактирования включен; вызывает сброс, если обнаружено условие, означающее исчезновение тактовых импульсов.

Бит 1: PORSF: Флаг сброса типа POR (сброс при включении питания)/Флаг сброса от схемы слежения за напряжением питания.

Этот бит устанавливается в 1 каждый раз при сбросе типа POR. Запись этого бита используется для выбора схемы слежения за напряжением питания в качестве источника сброса. **Примечание: установка в 1 этого бита до включения и стабилизации схемы слежения за напряжением питания может привести к системному сбросу.** (См. описание регистра VDM0CN на рис.10.3).

0: **Чтение:** Предыдущий сброс не был сбросом типа POR или сбросом от схемы слежения за напряжением питания;

**Запись:** Схема слежения за напряжением питания не является источником сброса.

1: **Чтение:** Предыдущий сброс был сбросом типа POR или сбросом от схемы слежения за напряжением питания; состояние всех других флагов сброса неопределено;

**Запись:** Схема слежения за напряжением питания является источником сброса.

Бит 0: PINRSF: Флаг сброса от вывода /RST

0: Предыдущий сброс не был сбросом от вывода /RST.

1: Предыдущий сброс был сбросом от вывода /RST.

**Примечание:** Для бит, которые функционируют как биты разрешения источников сброса (при записи) и флаги сброса (при чтении), команды типа чтение-модификация-запись читают и модифицируют только разрешения источников сброса. Это касается бит: USBRSF, CORSEF, SWRSF, MCDRSF, PORSF.



Таблица 10.1. Электрические параметры источников сброса

T = от -40°C до +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Выходное напряжение низкого уровня на выводе /RST	$I_{OL} = 8.5\text{mA}$ , $V_{DD} = 2.7 \dots 3.6\text{V}$			0.6	В
Входное напряжение высокого уровня на выводе /RST		0.7 x VDD			В
Входное напряжение низкого уровня на выводе /RST				0.3 x VDD	В
Входной ток утечки вывода /RST	/RST = 0.0В		25	40	мкА
Пороговое значение напряжения VDD для сброса типа POR ( $V_{RST}$ )		2.40	2.55	2.70	В
Таймаут детектора исчезновения тактирования	Время от нарастающего фронта последнего тактового импульса до генерации сброса	100	220	500	мкс
Время задержки после сброса	Задержка между выходом из состояния любого сброса и выполнением кода по адресу 0x0000	5.0			мкс
Минимальная длительность низкого уровня сигнала на выводе /RST, необходимая для генерации системного сброса		15			мкс
Время включения схемы слежения за напряжением питания		100			мкс
Ток потребления схемы слежения за напряжением питания			20	50	мкА

## Примечания

## 11. FLASH-ПАМЯТЬ

Встроенная перепрограммируемая Flash-память предназначена для хранения программного кода и долговременного хранения данных. Flash-память может программироваться внутрисистемно (по одному байту за раз) посредством JTAG-интерфейса или из программы, используя команду MOVX. Если Flash-бит сброшен в 0, то для того, чтобы установить его в 1, его необходимо стереть. Байты перед программированием обычно стираются (устанавливаются в 0xFF). Временные интервалы операций записи и стирания, необходимые для корректной работы, устанавливаются автоматически аппаратными средствами. Опрос данных для определения завершения операции записи/стирания не требуется. Выполнение программного кода останавливается во время операций записи/стирания Flash-памяти. Электрические параметры Flash-памяти приведены в таблице 11.1.

### 11.1. Программирование Flash-памяти

Самый простой способ программирования Flash-памяти заключается в использовании C2-интерфейса и средств программирования, предлагаемых фирмой Silicon Labs или независимыми производителями. Это единственный способ программирования неинициализированного МК. Подробная информация об использовании C2-команд для программирования Flash-памяти приведена в разделе 21 «Интерфейс C2» на стр. 253.

**Чтобы гарантировать целостность содержимого Flash-памяти, строго рекомендуется включить схему слежения за напряжением питания. Это касается любых систем, которые содержат код, осуществляющий запись/стирание Flash-памяти программным путем.**

#### 11.1.1. Блокировка Flash-памяти и ключевой код

Функции блокировки и ключевой код защищают Flash-память от операций записи и стирания. Операции с Flash-памятью невозможны, если перед ними не записать в регистр блокировки и ключевого кода Flash-памяти (FLKEY) корректные ключевые коды в определенной последовательности. Эти коды следующие: 0xA5, 0xF1. Временные интервалы не имеют значения, но эти коды должны быть записаны в правильной последовательности. Если эти коды записаны в неверном порядке или если записаны неверные коды, то операции записи и стирания Flash-памяти будут запрещены до следующего системного сброса. Операции записи и стирания Flash-памяти будут запрещены также в том случае, если попытка записать или стереть Flash-память была произведена до корректной записи ключевых кодов. Блокировка Flash-памяти восстанавливается после каждой операции записи или стирания; следующая операция записи или стирания Flash-памяти возможна только после повторной записи ключевых кодов. Подробное описание регистра FLKEY приведено на рис. 11.3.

#### 11.1.2. Процедура стирания Flash-памяти

Flash-память можно программировать программным путем, используя команду MOVX с адресом и байтом данных в качестве обычных операндов. Перед записью во Flash-память с использованием команды MOVX операции записи Flash-памяти необходимо разрешить:

- 1) записью ключевых кодов Flash-памяти в правильном порядке в регистр FLKEY;
- 2) установкой в 1 бита разрешения записи памяти программ PSWE (PSCTL.0). Это приведет к тому, что операции записи с помощью команды MOVX будут относиться не к памяти XRAM, а к Flash-памяти. Бит PSWE остается установленным в 1 до сброса его программным путем.

Запись во Flash-память может сбросить в 0 биты, но не может установить их в 1. Только операция стирания может установить в 1 биты во Flash-памяти. **Поэтому ячейку памяти, которую требуется запрограммировать, необходимо предварительно стереть, чтобы можно было записать новое значение.** Flash-память организована секторами по 512 байт. Операция стирания применяется ко всему сектору целиком (все байты в секторе устанавливаются в 0xFF). Ниже приведен алгоритм стирания 512-байтной страницы Flash-памяти:

1. Запретить прерывания (рекомендуется).
2. Записать первый ключевой код в регистр FLKEY: 0xA5.
3. Записать второй ключевой код в регистр FLKEY: 0xF1.
4. Установить в 1 бит PSEE (PSCTL.1).
5. Установить в 1 бит PSWE (PSCTL.0).
6. Используя команду MOVX, записать байт данных в любую ячейку внутри 512-байтного сектора, который требуется стереть.
7. Сбросить в 0 бит PSWE (PSCTL.0).
8. Сбросить в 0 бит PSEE (PSCTL.1).

### 11.1.3. Процедура записи Flash-памяти

Ниже приведен алгоритм программирования Flash-памяти из программы пользователя:

1. Запретить прерывания (рекомендуется).
2. Стереть 512-байтную страницу Flash-памяти, которая содержит требуемую ячейку памяти, как описано в разделе 11.1.2.
3. Записать первый ключевой код в регистр FLKEY: 0xA5.
4. Записать второй ключевой код в регистр FLKEY: 0xF1.
5. Установить в 1 бит PSWE (PSCCTL.0).
6. Сбросить в 0 бит PSEE (PSCCTL.1).
7. Используя команду MOVX, записать один байт данных в требуемую ячейку памяти внутри 512-байтного сектора.
8. Сбросить в 0 бит PSWE (PSCCTL.0).

Шаги 3-8 необходимо повторять для каждого записываемого байта. После завершения операций записи Flash-памяти бит PSWE следует сбросить в 0, чтобы при выполнении команд MOVX происходило обращение к памяти программ.

**Таблица 11.1. Электрические параметры FLASH-памяти**

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Объем Flash-памяти	C8051F320/1	16384*			Байт
Число циклов программирования		20000	100000		Стирание/Запись
Время цикла стирания	SYSCLK = 25 МГц	10	15	20	мс
Время цикла записи	SYSCLK = 25 МГц	40	55	70	мкс

\*Примечание: 512 байт по адресам 0x3E00 – 0x3FFF зарезервированы.

## 11.2. Долговременное хранение данных

Flash-память может использоваться не только для хранения программного кода, но и для долговременного хранения данных. Это позволяет рассчитывать и сохранять во время выполнения программы такие данные, как калибровочные коэффициенты. Данные записываются с использованием команды MOVX и считываются с использованием команды MOVC. Примечание: команды чтения MOVX всегда относятся к памяти XRAM.

## 11.3. Защита FLASH-памяти

CIP-51 имеет опции защиты, позволяющие защитить Flash-память от случайной модификации со стороны программы и исключить возможность просмотра патентованного программного кода и констант. Биты разрешения записи памяти программ PSWE (PSCTL.0) и разрешения стирания памяти программ PSEE (PSCTL.1) защищают Flash-память от случайной модификации со стороны программы. Бит PSWE должен быть явно установлен в 1, чтобы программа могла модифицировать Flash-память; оба бита PSWE и PSEE должны быть явно установлены в 1, чтобы программа могла стирать Flash-память. Дополнительные функции защиты предотвращают чтение и изменение патентованного программного кода и констант посредством C2-интерфейса.

Байт блокировки защиты, расположенный в последнем байте пользовательского пространства Flash-памяти, обеспечивает защиту Flash-памяти программ от доступа (операции чтения, записи или стирания) со стороны незащищенного программного кода или через C2-интерфейс. Механизм защиты Flash-памяти позволяет пользователю заблокировать  $n$  512-байтных страниц Flash-памяти, начиная со страницы 0 (адреса 0x0000 – 0x01FF), где  $n$  – обратное значение байта блокировки защиты. Ниже приведен пример:

Байт блокировки защиты	11111101b
<u>Обратное значение</u>	<u>00000010b</u>
Блокируемые страницы Flash-памяти	2
Блокируемые адреса	0x0000 – 0x03FF

### Важные замечания относительно защиты Flash-памяти:

1. Сброс в 0 любого бита в байте блокировки заблокирует страницу Flash-памяти, содержащую байт блокировки (помимо выбранных страниц).
2. Заблокированные страницы нельзя читать, записывать или стирать посредством C2-интерфейса.
3. Заблокированные страницы нельзя читать, записывать или стирать из программы пользователя, выполняющейся в незаблокированном разделе памяти.
4. Программа пользователя, выполняющаяся в заблокированной странице, может читать и записывать любую заблокированную или незаблокированную страницу Flash-памяти, кроме зарезервированной области.
5. Программа пользователя, выполняющаяся в заблокированной странице, может стирать любую заблокированную или незаблокированную страницу Flash-памяти, кроме зарезервированной области и страницы, содержащей байт блокировки.
6. Заблокированные страницы можно разблокировать только посредством C2-интерфейса с помощью C2-команды стирания всей памяти МК.
7. Если попытка доступа к Flash-памяти из программы пользователя отклонена (согласно описанным выше ограничениям #3, #4 и #5), то будет сгенерирован системный сброс от Flash-контроллера.

Рисунок 11.1. Карта распределения Flash-памяти программ и байты защиты

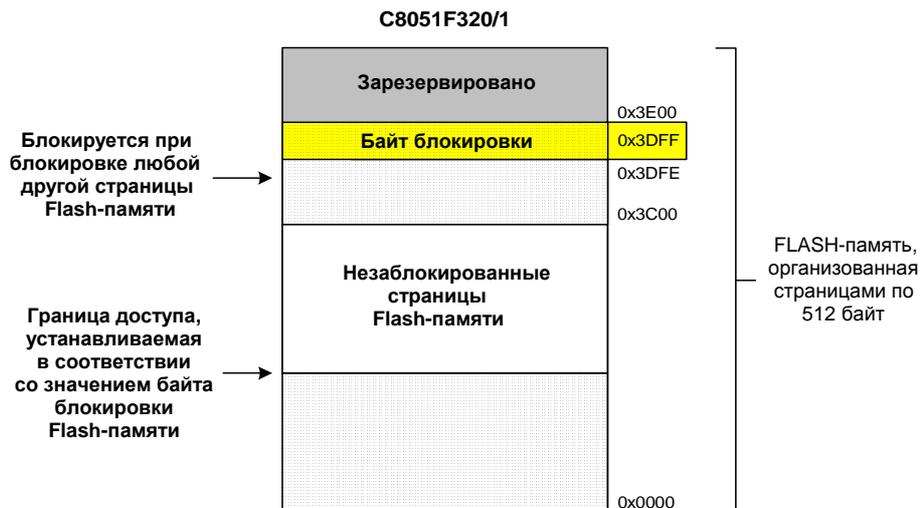


Рисунок 11.2. PSCTL: Регистр управления записью/стиранием памяти программ

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	-	-	-	Зарезерв.	PSEE	PSWE	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x8F

Биты 7-3: Не используются. Читаются как 00000b. Запись не оказывает никакого влияния.

Бит 2: Зарезервирован. Читается как 0b. Должен быть записан значением 0b.

Бит 1: PSEE: Разрешение стирания памяти программ.  
Установка этого бита разрешает стереть целую страницу Flash-памяти программ при условии, что бит PSWE также установлен. Если PSEE = 1 и PSWE = 1, то запись во Flash-память с использованием команды MOVX сотрет целую страницу, которая содержит ячейку, адресуемую командой MOVX (значение записываемого байта данных не важно).  
0: Стирание Flash-памяти программ запрещено.  
1: Стирание Flash-памяти программ разрешено.

Бит 0: PSWE: Разрешение записи памяти программ.  
Установка этого бита разрешает запись байта данных во Flash-память программ, используя команду MOVX. Адресуемая в команде MOVX ячейка памяти должна быть стертой.  
0: Запись во Flash-память программ запрещена.  
1: Запись во Flash-память программ разрешена; команда записи MOVX обращается к Flash-памяти.

**Рисунок 11.3. FLKEY: Регистр блокировки и ключевого кода Flash-памяти**

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xB7

Биты 7-0: FLKEY: Регистр блокировки и ключевого кода Flash-памяти.

**Запись:**  
 Чтобы можно было записывать или стирать Flash-память, необходимо сначала записать ключевые коды в этот регистр. Flash-память остается заблокированной до тех пор, пока в этот регистр не будет записана следующая последовательность ключевых кодов: 0xA5, 0xF1. Временные интервалы не имеют значения, но эти коды должны быть записаны в правильной последовательности. Ключевые коды должны записываться для каждой операции записи или стирания Flash-памяти. Flash-память будет заблокирована до следующего системного сброса в том случае, если записаны неверные коды или если попытка выполнить операцию с Flash-памятью осуществляется до корректной записи этих кодов.

**Чтение:**  
 При чтении биты 1-0 показывают текущее состояние блокировки Flash-памяти.  
 00: Flash-память заблокирована для операций записи/стирания.  
 01: Первый ключевой код записан (0xA5).  
 10: Flash-память разблокирована (операции записи/стирания разрешены).  
 11: Операции записи/стирания Flash-памяти запрещены до следующего системного сброса.

**Рисунок 11.4. FLSCL: Регистр управления контроллером Flash-памяти**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
FOSE	Зарезерв.	10000000						
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xB6

Бит 7: FOSE: Включение ждущего таймера модуля Flash-памяти  
 Этот бит включает ждущий таймер для операций чтения Flash-памяти. Когда ждущий таймер Flash-памяти отключен, усилитель считывания данных Flash-памяти включен весь тактовый цикл во время операций чтения Flash-памяти. При системной тактовой частоте менее 10 МГц отключение ждущего таймера Flash-памяти приведет к увеличению энергопотребления.  
 0: Ждущий таймер модуля Flash-памяти выключен.  
 1: Ждущий таймер модуля Flash-памяти включен.

Биты 6-0: Зарезервированы. Читаются как 0000000b. В эти биты следует записать значение 0000000b.

## Примечания

## 12. ВНЕШНЕЕ ОЗУ

МК C8051F320/1 содержат 2048 байт встроенной памяти XRAM. Пространство этой памяти XRAM разделено на ОЗУ пользователя (адреса 0x0000 – 0x03FF) и буфер FIFO модуля USB0 (адреса 0x0400 – 0x07FF).

**Рисунок 12.1. Карта распределения памяти внешнего ОЗУ**



### 12.1. Доступ к пользовательской памяти XRAM

Для доступа к пространству памяти XRAM можно использовать команду MOVX и указатель данных DPTR, или команду MOVX с косвенным режимом адресации. Если команда MOVX используется с 8-разрядным операндом адреса (например, @R1), то старший байт 16-разрядного адреса определяется регистром управления интерфейсом внешней памяти EMIOCN (см. рис.12.3). Примечание: команда MOVX используется также для записи во Flash-память (см. раздел 11 «Flash-память» на стр.107). По умолчанию команда MOVX обращается к памяти XRAM.

В любом режиме адресации старшие 5 бит 16-разрядного адреса внешней памяти данных «не имеют значения». Таким образом, 2048-байтное ОЗУ отображается помодульно во всем 64-Кбайтном адресном пространстве внешней памяти данных. Например, байт XRAM с адресом 0x0000 можно также адресовать с помощью адресов 0x0800, 0x1000, 0x1800, 0x2000 и т.д.

**Важное замечание:** Старшие 1Кб из 2Кб XRAM функционируют как USB FIFO. Подробная информация о доступе к этой области памяти приведена в разделе 12.2.

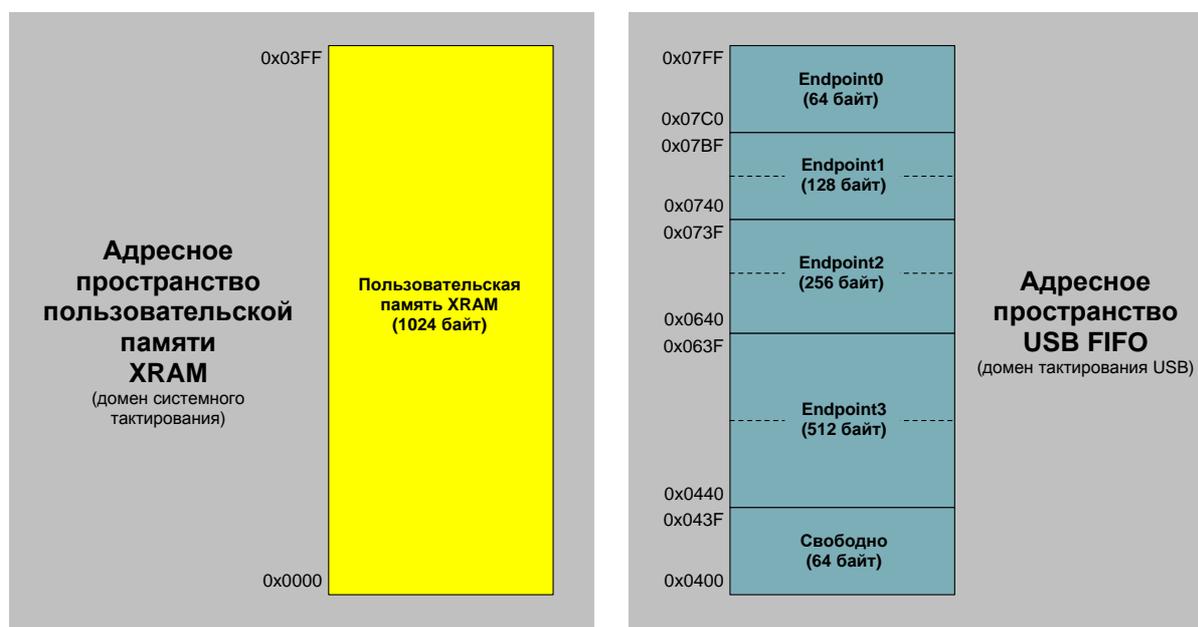
## 12.2. Доступ к USB FIFO

Старшие 1Кб памяти XRAM функционируют как область буферов USB FIFO. На рис.12.2 показано расширенное представление области FIFO и пользовательской памяти XRAM. Доступ к области USB FIFO осуществляется с помощью USB FIFO регистров; подробная информация о доступе к этим FIFO приведена в разделе 15.5 «Управление буфером FIFO» на стр.151. Команда MOVX не должна использоваться для загрузки и модификации USB-данных в области FIFO.

Неиспользованные области пространства FIFO можно использовать в качестве памяти XRAM общего назначения, доступной как описано в разделе 12.1. Блок FIFO функционирует в области тактирования USB; поэтому тактирование USB должно быть активировано при доступе к пространству FIFO. Следует иметь в виду, что при доступе к пространству USB FIFO для выполнения команды MOVX требуется большее количество циклов SYSCLK.

**Важное замечание:** При доступе к области FIFO тактирование USB должно быть активировано.

**Рисунок 12.2. Расширенное представление памяти XRAM**



**Рисунок 12.3. ЕМІ0СN: Регистр управления интерфейсом внешней памяти**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	-	-	-	PGSEL2	PGSEL1	PGSEL0	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0хAA

Биты 7-3: Не используются. Читаются как 00000b.

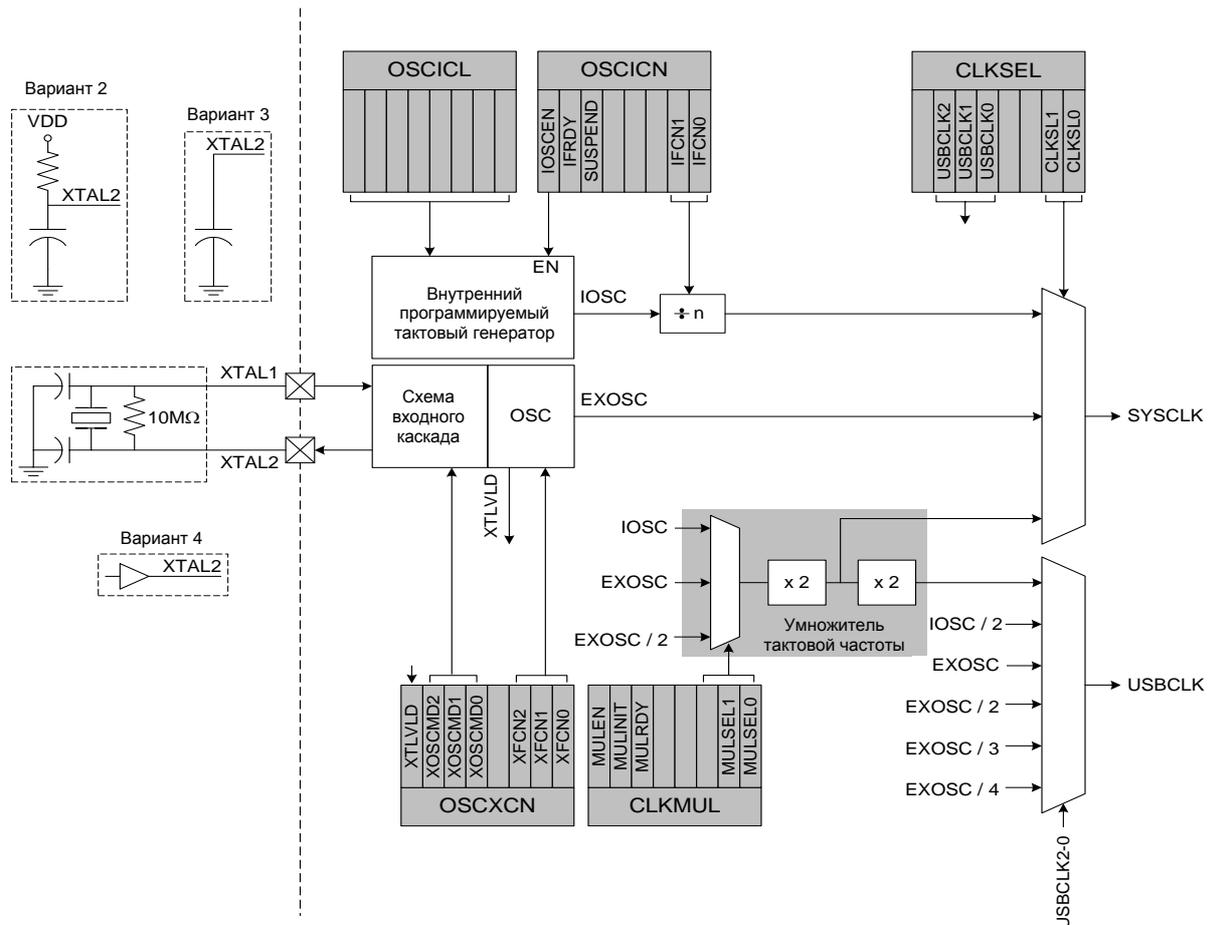
Биты 2-0: PGSEL[2:0]: Биты выбора страницы XRAM.  
 Биты выбора страницы XRAM образуют старший байт 16-битного адреса внешней памяти данных при использовании команды MOVX с 8-битным адресом, задавая таким образом необходимую 256-байтную страницу ОЗУ. Старшие 5 бит «не имеют значения», таким образом 2-Кбайтные блоки адресов повторяются помодульно во всем 64-Кбайтном адресном пространстве памяти данных.

## Примечания

## 13. ГЕНЕРАТОРЫ

МК C8051F320/1 содержат программируемый внутренний генератор, схему возбуждения внешнего генератора и 4-кратный умножитель тактовой частоты. Внутренний генератор можно включать/отключать и калибровать с помощью регистров OSCICN и OSCICL, как показано на рис.13.1. Системный тактовый сигнал (SYSCLK) может быть получен от внутреннего генератора, от схемы внешнего генератора или от 4-кратного умножителя частоты с делением на 2. Сигнал тактирования USB (USBCLK) может быть получен от внутреннего генератора, от внешнего генератора или от 4-кратного умножителя частоты. Электрические параметры генератора приведены в таблице 13.3. на стр.126.

**Рисунок 13.1. Структурная схема генератора**



### 13.1. Программируемый внутренний генератор

Все МК C8051F320/1 содержат программируемый внутренний генератор, который после сброса системы является по умолчанию системным тактовым генератором. Частоту внутреннего генератора можно программировать с помощью регистра OSCICL (см. уравнение 13.1).

**Уравнение 13.1. Типичное изменение периода выходного сигнала внутреннего генератора, задаваемое с помощью регистра OSCICL.**

$$\Delta T \approx 0.0025 \times (1 / f_{BASE}) \times \Delta OSCICL$$

где  $f_{BASE}$  – частота внутреннего генератора, устанавливаемая после сброса;  
 $\Delta T$  – изменение периода выходного сигнала внутреннего генератора;  
 $\Delta OSCICL$  – изменение значения регистра OSCICL.

OSCICL калибруется при изготовлении МК таким образом, чтобы частота внутреннего генератора после сброса ( $f_{BASE}$ ) составляла 12 МГц. В разделе 13.1.1 подробно описано программирование генератора в МК C8051F320/1. Электрические параметры прецизионного внутреннего генератора приведены табл.13.3 на стр. 126. Следует иметь в виду, что системная тактовая частота может быть получена из частоты внутреннего генератора, деленной на 1, 2, 4 или 8, в зависимости от значения битов IFCN регистра OSCICN. После сброса по умолчанию устанавливается коэффициент деления 8.

### 13.1.1. Программирование внутреннего генератора в МК C8051F320/1

Значение после сброса регистра OSCICL задается (при изготовлении МК) таким образом, чтобы частота внутреннего генератора составляла 12 МГц с точностью  $\pm 1.5\%$ ; такая частота подходит для использования в качестве тактовой частоты USB (см. раздел 13.4). Программа может изменить частоту внутреннего генератора, как описано ниже.

**Важное замечание:** После изменения частоты внутреннего генератора его более нельзя использовать в качестве источника тактирования USB, как описано в разделе 13.4. Любой сброс МК установит частоту внутреннего генератора равной откалиброванному при изготовлении значению, после чего этот генератор можно использовать в качестве источника тактирования USB.

Чтобы получить требуемую частоту, программа должна прочитать и изменить значение регистра OSCICL в соответствии с уравнением 13.1. В приведенном ниже примере показано, как получить частоту внутреннего генератора 11.6 МГц.

- $f_{BASE}$  – частота внутреннего генератора, устанавливаемая после сброса;
- $T_{BASE}$  – период выходного сигнала внутреннего генератора, устанавливаемого после сброса;
- $f_{DES}$  – требуемая частота внутреннего генератора;
- $T_{DES}$  – требуемый период выходного сигнала внутреннего генератора.

$$f_{BASE} = 12000000 \text{ Гц}$$

$$f_{DES} = 11600000 \text{ Гц}$$

$$T_{BASE} = (1 / 12000000) \text{ сек}$$

$$T_{DES} = (1 / 11600000) \text{ сек}$$

Требуемое изменение периода ( $\Delta T_{DES}$ ) равно разнице между базовым периодом и требуемым периодом.

$$\Delta T_{DES} = (1 / 11600000) - (1 / 12000000) = 2.87 \times 10^{-9} \text{ сек}$$

Используя уравнение 13.1 и приведенные выше данные, находим  $\Delta OSCICL$ :

$$2.87 \times 10^{-9} = 0.0025 \times (1 / f_{BASE}) \times \Delta OSCICL$$

$$\Delta OSCICL = 13.79$$

$\Delta OSCICL$  округляется до ближайшего целого числа (14) и добавляется к значению регистра OSCICL, устанавливаемому после сброса.

**Важное замечание:** Если сумма значения регистра OSCICL, устанавливаемого после сброса, и  $\Delta OSCICL$  больше 31 или меньше 0, то это означает, что данный МК не способен генерировать сигнал с требуемой частотой.

### 13.1.2. Экономичный режим внутреннего генератора

Внутренний генератор можно перевести в экономичный режим, установив в 1 бит SUSPEND в регистре OSCICN. В экономичном режиме внутренний генератор останавливается до тех пор, пока не будет обнаружено требующее обработки событие на шине USB (см. раздел 15) или пока сигнал VBUS не совпадет с полярностью, выбранной битом VBPOL в регистре REG0CN (см. раздел 8.2). Следует иметь в виду, что обнаружение события на шине USB возможно только при включенном приемопередатчике USB.



**Рисунок 13.2. OSCICN: Регистр управления внутренним генератором**

R/W	R	R/W	R	R/W	R/W	R/W	R/W	Значение при сбросе:
IOSCEN	IFRDY	SUSPEND	-	-	-	IFCN1	IFCN0	00010100
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xB2

Бит 7: IOSCEN: Бит включения внутреннего генератора  
 0: Внутренний генератор выключен  
 1: Внутренний генератор включен

Бит 6: IFRDY: Флаг стабилизации частоты внутреннего генератора  
 0: Частота внутреннего генератора не соответствует частоте, задаваемой битами IFCN.  
 1: Частота внутреннего генератора соответствует частоте, задаваемой битами IFCN.

Бит 5: SUSPEND: Бит перехода в экономичный режим работы.  
 Установка в 1 этого бита вызовет остановку внутреннего генератора. Этот генератор будет запущен вновь после следующего события на шине USB, требующего обработки (например, событие RESUME), или после прерывания VBUS (см. рис.8.5).

Биты 4-2: Не используются. Читаются как 000b. Запись не оказывает никакого влияния.

Биты 1-0: IFCN1-0: Биты управления частотой внутреннего генератора  
 00: SYSCLK равна частоте внутреннего генератора, деленной на 8.  
 01: SYSCLK равна частоте внутреннего генератора, деленной на 4.  
 10: SYSCLK равна частоте внутреннего генератора, деленной на 2.  
 11: SYSCLK равна частоте внутреннего генератора, деленной на 1.

**Рисунок 13.3. OSCICL: Регистр калибровки внутреннего генератора**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	-	OSCCAL				-	различно для каждого экземпляра МК
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xB3

Биты 4-0: OSCCAL: Значение калибровки внутреннего генератора.  
 Эти биты определяют период выходного сигнала внутреннего генератора в соответствии с уравнением 13.1.  
**Примечание: Если схема восстановления синхронизации включена, то содержимое этого регистра является неопределенным. Подробная информация о схеме восстановления синхронизации приведена в разделе 15.4 «Настройка тактирования USB-контроллера» на стр. 150.**

### 13.2. Схема возбуждения внешнего генератора.

Для внешнего генератора можно использовать кварцевый или керамический резонатор, конденсатор или RC-цепочку. Кроме этого, системный тактовый сигнал может подаваться от внешнего КМОП-счетчика. Если используется кварцевый или керамический резонатор, то он должен подключаться к выводам XTAL1 и XTAL2, как показано на рис.13.1 (вариант 1). При использовании этой конфигурации между выводами XTAL1 и XTAL2 должен быть подключен также резистор сопротивлением 10Мом. При использовании RC-цепочки, конденсатора или КМОП-счетчика источник тактовых импульсов следует подключать к выводу XTAL2, как показано на рис.13.1 (варианты 2, 3 или 4). Тип внешнего генератора следует выбрать в регистре OSCXCN. В соответствии с выбранным типом внешнего генератора следует установить биты управления его частотой XFCN (см. рис.13.4).

**Важное замечание относительно использования внешнего генератора:** При использовании схемы внешнего генератора выводы порта должны быть правильно настроены. Если схема возбуждения внешнего генератора используется совместно с кварцевым/керамическим резонатором, то выводы порта P0.2 и P0.3 используются как XTAL1 и XTAL2 соответственно. Если схема возбуждения внешнего генератора используется совместно с RC-цепочкой, конденсатором или КМОП-счетчиком, то вывод порта P0.3 используется как XTAL2. Матрица должна быть настроена таким образом, чтобы при назначении выводов она пропускала порты, используемые схемой генератора; настройка матрицы описана в разделе 14.1 «Приоритетный декодер матрицы» на стр. 129. Кроме этого при использовании схемы возбуждения внешнего генератора совместно с кварцевым/керамическим резонатором, конденсатором или RC-цепочкой задействованные выводы портов должны быть настроены как **аналоговые входы**. При использовании КМОП-счетчика задействованный вывод порта должен быть настроен как **цифровой вход**. Подробная информация о выборе режима входов порта приведена в разделе 14.2 «Инициализация порта ввода/вывода» на стр.131.

#### 13.2.1. Тактирование таймеров непосредственно от внешнего генератора

Сигнал внешнего генератора, деленный по частоте на 8, может быть одним и сигналов тактирования таймеров (см. раздел 19 «Таймеры» на стр.217) и программируемого массива счетчиков (см. раздел 20 «Программируемый массив счетчиков» на стр.235). Если внешний генератор используется для тактирования этих периферийных модулей, но не используется для тактирования системы, то частота внешнего генератора не должна превышать частоту системного тактового сигнала. При такой конфигурации тактовый сигнал, подаваемый на периферийные модули (частота внешнего генератора/8) синхронизирован с системным тактовым сигналом; неустойчивость данной синхронизации ограничена на уровне  $\pm 0.5$  цикла системного тактового сигнала.

#### 13.2.2. Пример использования внешнего резонатора

Если бы для генерации системной тактовой частоты МК использовался кварцевый или керамический резонатор, то схема была бы такой, как показано на рис.13.1, вариант 1. При выборе значения битов управления частотой внешнего генератора (XFCN) следует использовать столбец «Резонатор» таблицы, приведенной на рис.13.4 (регистр OSCICN). Например, для резонатора с частотой 12 МГц значение битов XFCN должно быть 111b.

При первом включении кварцевого генератора схема детектирования амплитуды сигнала генератора выходит на стабильный режим работы не сразу, а в течение определенного времени установления. Введение задержки длительностью как минимум 1мс между включением генератора и проверкой бита XTLVLD предотвратит преждевременный переход системы на тактирование от внешнего генератора. Переключение на работу от внешнего генератора до выхода его на устойчивый режим работы может привести к непредсказуемому поведению МК. Рекомендуется следующая последовательность действий:

1. Включить внешний генератор.
2. Выдержать паузу длительностью как минимум 1мс.
3. Опрашивать бит XTLVLD до обнаружения перехода его состояния из '0' в '1'.
4. Переключиться на тактирование от внешнего генератора.

**Важное примечание:** Схемы кварцевых генераторов достаточно чувствительны к разводке печатной платы. Кварцевый резонатор следует размещать как можно ближе к выводам XTAL микроконтроллера, добиваясь минимальной длины проводников, а также экранировать слоем «земли» от любых других проводников, которые могли бы быть причиной шумов и помех.



### 13.2.3. Пример использования RC-генератора

Если бы для генерации системной тактовой частоты МК использовалась RC-цепочка, то схема была бы такой, как показано на рис.13.1, вариант 2. Емкость конденсатора должна быть не более 100пФ, однако использование конденсатора с очень маленькой емкостью приведет к увеличению частотного дрейфа вследствие влияния паразитной емкости печатной платы. Чтобы определить значение битов управления частотой внешнего генератора (XFCN) регистра OSCXCN, сначала нужно выбрать значения компонентов RC-цепи, необходимые для получения требуемой частоты генерации. Например, если требуется частота 100кГц, то можно взять  $R = 246\text{кОм}$  и  $C = 50\text{пФ}$ :

$$f = 1.23(10^3)/RC = 1.23(10^3) / [246 * 50] = 0.1\text{МГц} = 100\text{кГц}$$

$$XFCN \geq \log_2(f/25\text{кГц})$$

$$XFCN \geq \log_2(100\text{кГц}/25\text{кГц}) = \log_2(4)$$

$$XFCN \geq 2, \text{ или код } 010$$

Программирование бит XFCN в RC-режиме более высокими значениями улучшит точность задания частоты, но приведет к увеличению тока потребления внешнего генератора.

### 13.2.4. Пример использования внешнего генератора с конденсатором

Если бы для генерации системной тактовой частоты МК использовался внешний конденсатор, то схема была бы такой, как показано на рис.13.1, вариант 3. Емкость конденсатора должна быть не более 100пФ, однако использование конденсатора с очень маленькой емкостью приведет к увеличению погрешности установки частоты вследствие влияния паразитной емкости печатной платы. Чтобы определить значение битов управления частотой внешнего генератора (XFCN) регистра OSCXCN, сначала необходимо выбрать емкость используемого конденсатора и вычислить частоту генерации из уравнения, приведенного ниже. Например, для  $VDD = 3.0\text{В}$  и  $C = 50\text{пФ}$  получим:

$$f = KF / (C * VDD) = KF / (50 * 3) \text{ МГц}$$

$$f = KF / 150 \text{ МГц}$$

Если требуется получить частоту около 150кГц, то из таблицы на рис.13.4 необходимо выбрать  $KF = 22$ :

$$f = 22 / 150 = 0.146\text{МГц}, \text{ или } 146\text{кГц}$$

Поэтому, значение битов XFCN для этого примера составляет 011b.

Рисунок 13.4. OSCXCN: Регистр управления внешним генератором

R	R/W	R/W	R/W	R	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xB1
XTLVLD	XOSCMD2	XOSCMD1	XOSCMD0	-	XFCN2	XFCN1	XFCN0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: XTLVLD: Флаг стабилизации кварцевого генератора.

**(Читается только в том случае, если XOSCMD = 11x.)**

0: Кварцевый генератор не используется или еще нестабилен.

1: Кварцевый генератор работает и стабилен.

Биты 6-4: XOSCMD2-0: Биты выбора режима внешнего генератора.

00x: Внешний генератор выключен.

010: Тактовые импульсы поступают от внешнего КМОП-счетчика.

011: Тактовые импульсы поступают от внешнего КМОП-счетчика через внутренний делитель на 2.

100: Режим RC-генератора.

101: Режим C-генератора.

110: Режим кварцевого генератора

111: Режим кварцевого генератора с делением тактовой частоты на 2.

Бит 3: Зарезервирован. Читается как 0b. Запись не оказывает никакого влияния.

Биты 2-0: XFCN2-0: Биты управления частотой внешнего генератора

000-111: см. таблицу ниже:

XFCN	Резонатор (XOSCMD = 11x)	RC (XOSCMD = 10x)	C (XOSCMD = 10x)
000	$f \leq 32 \text{ кГц}$	$f \leq 25 \text{ кГц}$	KF = 0.87
001	$32 \text{ кГц} < f \leq 84 \text{ кГц}$	$25 \text{ кГц} < f \leq 50 \text{ кГц}$	KF = 2.6
010	$84 \text{ кГц} < f \leq 225 \text{ кГц}$	$50 \text{ кГц} < f \leq 100 \text{ кГц}$	KF = 7.7
011	$225 \text{ кГц} < f \leq 590 \text{ кГц}$	$100 \text{ кГц} < f \leq 200 \text{ кГц}$	KF = 22
100	$590 \text{ кГц} < f \leq 1,5 \text{ МГц}$	$200 \text{ кГц} < f \leq 400 \text{ кГц}$	KF = 65
101	$1,5 \text{ МГц} < f \leq 4 \text{ МГц}$	$400 \text{ кГц} < f \leq 800 \text{ кГц}$	KF = 180
110	$4 \text{ МГц} < f \leq 10 \text{ МГц}$	$800 \text{ кГц} < f \leq 1,6 \text{ МГц}$	KF = 664
111	$410 \text{ МГц} < f \leq 30 \text{ МГц}$	$1,6 \text{ МГц} < f \leq 3,2 \text{ МГц}$	KF = 1590

#### Режим генератора с кварцевым или керамическим резонатором

(Схема на рис.13.1, Вариант 1; XOSCMD = 11x)

Выберите значение XFCN, соответствующее частоте кварцевого или керамического резонатора.

#### Режим RC-генератора

(Схема на рис.13.1, Вариант 2; XOSCMD = 10x)

Выберите значение XFCN, соответствующее диапазону генерируемых частот:

$$f = 1.23(10^3) / (R * C), \text{ где}$$

f = генерируемая частота в [МГц]

C = емкость конденсатора в [пФ]

R = сопротивление подтягивающего резистора в [кОм]

#### Режим генератора с конденсатором

(Схема на рис.13.1, Вариант 3; XOSCMD = 10x)

Выберите коэффициент K (KF) для требуемой частоты:

$$f = KF / (C * VDD), \text{ где}$$

f = генерируемая частота в [МГц]

C = емкость конденсатора, подключенного к выводу XTAL2, в [пФ]

VDD = напряжение питания МК в [В]



### 13.3. 4-кратный умножитель тактовой частоты

4-кратный умножитель тактовой частоты позволяет получить от 12МГц генератора тактовый сигнал с частотой 48МГц, требуемый для функционирования USB в высокоскоростном режиме (см. раздел 15.4 «Настройка тактирования USB-контроллера» на стр.150). Деленный по частоте выходной сигнал этого умножителя можно также использовать в качестве системного тактового сигнала. Подробная информация о выборе источников системного тактирования и тактирования USB приведена в разделе 13.4.

4-кратный умножитель тактовой частоты настраивается с помощью регистра CLKMUL. Ниже приведена процедура настройки и включения 4-кратного умножителя тактовой частоты:

1. Сбросить умножитель, записав байт 0x00 в регистр CLKMUL.
2. Выбрать источник входного сигнала умножителя, используя биты MULSEL.
3. Включить умножитель с помощью бита MULEN (CLKMUL | = 0x80).
4. Задержка длительностью не менее 5 мкс.
5. Инициализировать умножитель с помощью бита MULINIT (CLKMUL | = 0xC0).
6. Опрашивать бит MULRDY до обнаружения перехода его в лог '1'.

**Важное замечание:** Если в качестве входного сигнала 4-кратного умножителя тактовой частоты используется сигнал внешнего генератора, то к моменту инициализации умножителя внешний генератор должен быть включен и стабилен (см. раздел 13.4).

**Рисунок 13.5. CLKMUL: Регистр управления умножителем тактовой частоты**

R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xB9
MULEN	MULINIT	MULRDY	-	-	-	MULSEL			
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0		

Бит 7: MULEN: Бит включения умножителя тактовой частоты.  
(**Действителен, только если XOSCMD = 11x.**)  
0: Умножитель тактовой частоты отключен.  
1: Умножитель тактовой частоты включен.

Бит 6: MULINIT: Бит инициализации умножителя тактовой частоты.  
Когда умножитель тактовой частоты включен, этот бит должен быть сброшен в 0. Установка в 1 этого бита после включения умножителя тактовой частоты вызовет его инициализацию. Когда умножитель тактовой частоты стабилизируется, бит MULRDY читается как '1'.

Бит 5: MULRDY: Флаг готовности умножителя тактовой частоты.  
Этот предназначенный только для чтения бит показывает состояние умножителя тактовой частоты.  
0: Умножитель тактовой частоты не готов к работе (нестабилен).  
1: Умножитель тактовой частоты готов к работе (стабилен).

Биты 4-2: Не используются. Читаются как 000b. Запись не оказывает никакого влияния.

Биты 1-0: MULSEL: Биты выбора входного сигнала умножителя тактовой частоты.  
Эти биты определяют, какой сигнал будет подаваться на вход умножителя тактовой частоты.

MULSEL	Источник входного сигнала умножителя тактовой частоты
00	Внутренний генератор
01	Внешний генератор
10	Внешний генератор/2
11	ЗАРЕЗЕРВИРОВАНО

### 13.4. Выбор источника сигнала тактирования системы и сигнала тактирования USB

Внутренний генератор требует небольшого времени запуска и может быть выбран в качестве источника системного тактового сигнала или сигнала тактирования USB сразу же после операции записи регистра OSCICN, которая включает внутренний генератор. Внешним кварцевым и керамическим резонаторам обычно требуется определенное время для выхода на стабильный режим работы, после истечения которого они готовы к использованию. Когда внешний генератор будет стабилен, флаг стабилизации кварцевого генератора (XTLVLD в регистре OSCXCN) аппаратно установится в 1. **Чтобы исключить чтение некорректного значения XTLVLD, при использовании кварцевого резонатора программа должна выдержать паузу длительностью как минимум 1мс между включением внешнего генератора и проверкой бита XTLVLD.** RC- и C-режимы обычно не требуют времени запуска.

#### 13.4.1. Выбор источника системного тактового сигнала

Биты CLKSL[1:0] регистра CLKSEL определяют, какой генератор используется в качестве источника системного тактового сигнала. Биты CLKSL[1:0] следует установить в 01b, чтобы тактирование системы осуществлялось от внешнего генератора; однако внешний генератор может также тактировать периферийные модули (таймеры, ПМС, USB), в то время как внутренний генератор выбран в качестве источника системного тактового сигнала. Источник системного тактового сигнала можно переключать «на лету» между внутренним генератором, внешним генератором и 4-кратным умножителем тактовой частоты при условии, что выбранный генератор включен и стабилен.

#### 13.4.2. Выбор источника сигнала тактирования USB

Биты USBCLK[2:0] регистра CLKSEL определяют, какой генератор используется в качестве источника сигнала тактирования USB. В качестве сигнала тактирования USB можно использовать выходной сигнал 4-кратного умножителя тактовой частоты, деленный по частоте сигнал внутреннего генератора или деленный по частоте сигнал внешнего генератора. Следует иметь в виду, что при функционировании USB0 в высокоскоростном режиме необходимо использовать 48МГц сигнал тактирования USB; при функционировании USB0 в низкоскоростном режиме необходимо использовать 6МГц сигнал тактирования USB. Варианты выбора сигнала тактирования USB приведены на рис.13.6.

Ниже приведены примеры некоторых конфигураций тактирования USB:

**Таблица 13.1. Типичные настройки для высокоскоростного режима USB**

Внутренний генератор		
Тактовый сигнал	Выбор источника входного сигнала	Значения бит в регистрах
Сигнал тактирования USB	Умножитель тактовой частоты	USBCLK = 000b
Вход умножителя тактовой частоты	Внутренний генератор*	MULSEL = 00b
Внутренний генератор	Деленный по частоте на 1	IFSN = 11b
Внешний генератор		
Тактовый сигнал	Выбор источника входного сигнала	Значения бит в регистрах
Сигнал тактирования USB	Умножитель тактовой частоты	USBCLK = 000b
Вход умножителя тактовой частоты	Внешний генератор	MULSEL = 01b
Внешний генератор	Кварцевый генератор 12МГц кварцевый резонатор	XOSCMD = 110b XFSN = 111b

\* Для этой конфигурации схема восстановления синхронизации должна быть включена.

**Таблица 13.2. Типичные настройки для низкоскоростного режима USB**

Внутренний генератор		
Тактовый сигнал	Выбор источника входного сигнала	Значения бит в регистрах
Сигнал тактирования USB	Внутренний генератор/2	USBCLK = 001b
Внутренний генератор	Деленный по частоте на 1	IFSN = 11b
Внешний генератор		
Тактовый сигнал	Выбор источника входного сигнала	Значения бит в регистрах
Сигнал тактирования USB	Внешний генератор/4	USBCLK = 101b
Внешний генератор	Кварцевый генератор 24МГц кварцевый резонатор	XOSCMD = 110b XFSN = 111b



**Рисунок 13.6. CLKSEL: Регистр выбора тактового генератора**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
-	USBCLK			-	-	-	CLKSL	Значение при сбросе: 00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xA9

Бит 7: Не используется. Читается как 0b. Запись не оказывает никакого влияния.

Биты 6-4: USBCLK: Выбор сигнала тактирования USB.  
 Эти биты выбирают сигнал тактирования модуля USB0. При функционировании USB0 в высокоскоростном режиме необходимо выбрать 48МГц сигнал тактирования. При функционировании USB0 в низкоскоростном режиме необходимо выбрать 6МГц сигнал тактирования.

USBCLK	Выбранный источник тактового сигнала
000	4-кратный множитель тактовой частоты
001	Внутренний генератор/2
010	Внешний генератор
011	Внешний генератор/2
100	Внешний генератор/3
101	Внешний генератор/4
110	ЗАРЕЗЕРВИРОВАНО
111	ЗАРЕЗЕРВИРОВАНО

Биты 3-2: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

Биты 1-0: CLKSL1-0: Биты выбора источника системного тактового сигнала.  
 Эти биты выбирают источник системного тактового сигнала.

CLKSL	Выбранный источник тактового сигнала
00	Внутренний генератор (как задано битами IFCN в регистре OSCICN)
01	Внешний генератор
10	4-кратный множитель тактовой частоты/2
11	ЗАРЕЗЕРВИРОВАНО

Таблица 13.3. Электрические параметры внутреннего генератора

T = от -40°C до +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Частота внутреннего генератора	Частота, устанавливаемая при сбросе	11.82	12	12.18	МГц
Ток потребления внутреннего генератора (по цепи VDD)	OSCICN.7 = 1		450		мкА
Частота тактирования USB*	Высокоскоростной режим	47.88	48	48.12	МГц
	Низкоскоростной режим	5.91	6	6.09	

\* Только при использовании внешнего генератора

## 14. ПОРТ ВВОДА/ВЫВОДА

Доступ к аналоговым и цифровым ресурсам МК осуществляется через 25 (C8051F320) или 21 (C8051F321) выводов порта ввода/вывода. Организация выводов порта показана на рис.14.1. Каждый вывод порта можно определить как порт ввода/вывода общего назначения или как аналоговый вход; выводы порта P0.0 – P2.3 могут быть назначены одному из внутренних цифровых модулей, как показано на рис.14.3. Разработчик системы определяет, какие цифровые ресурсы будут назначены внешним выводам, ограничиваясь только количеством доступных выводов. Гибкость при распределении ресурсов достигается благодаря использованию приоритетного декодера матрицы. Следует иметь в виду, что состояние вывода порта ввода/вывода всегда можно прочитать из соответствующего регистра-защелки порта независимо от настройки матрицы.

Матрица назначает выводы порта ввода/вывода выбранным внутренним цифровым ресурсам, используя приоритетный декодер (см. рис.14.3 и рис.14.4). Для выбора внутренних цифровых ресурсов используются регистры XBR0 и XBR1 (см. рис.14.5 и рис.14.6).

Допустимое напряжение любого порта ввода/вывода составляет 5В (см. схему ячейки порта на рис.14.2). С помощью регистров настройки выходов порта (PnMDOUT, где n = 0,1,2,3) ячейки порта ввода/вывода настраиваются либо как двухтактные цифровые выходы, либо как выходы с открытым стоком. Электрические характеристики порта ввода/вывода приведены в табл. 14.1 на стр.142.

**Рисунок 14.1. Функциональная схема порта ввода/вывода**

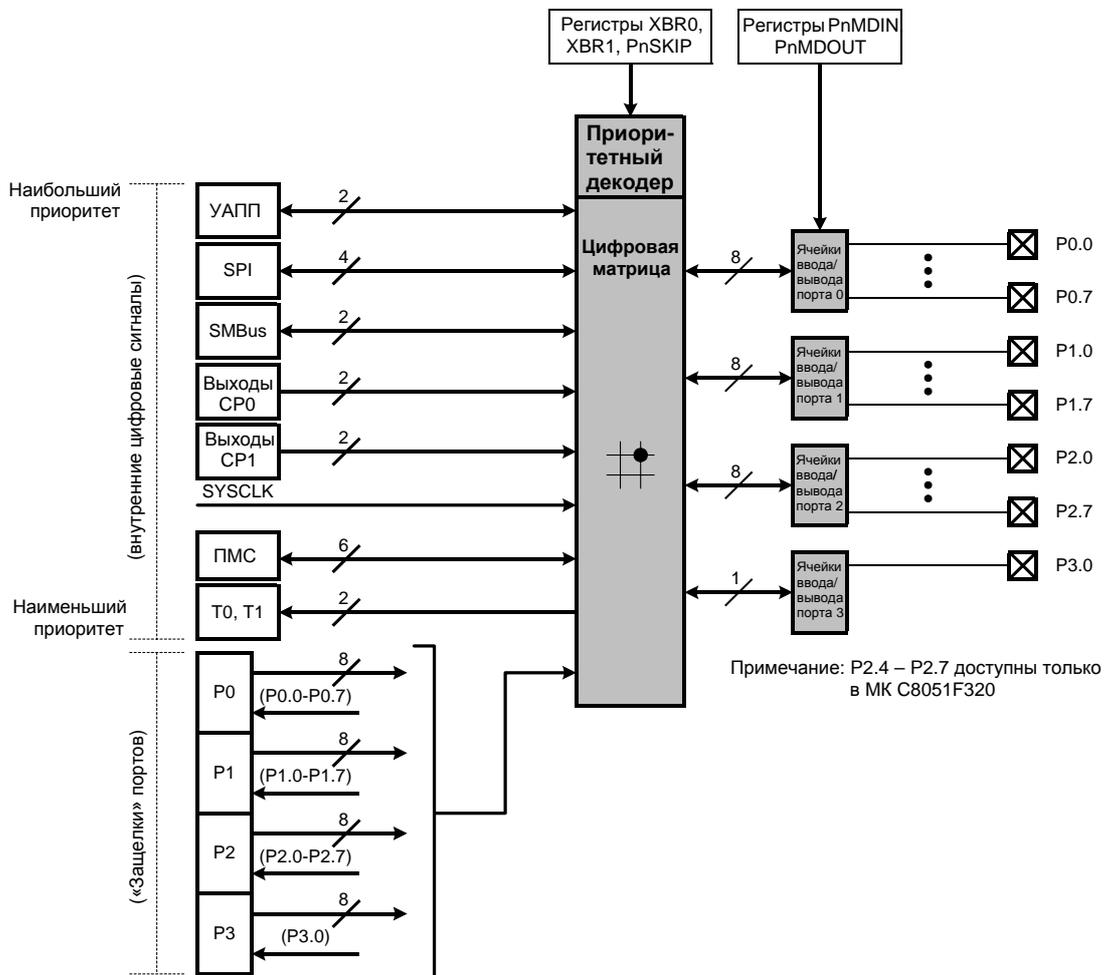
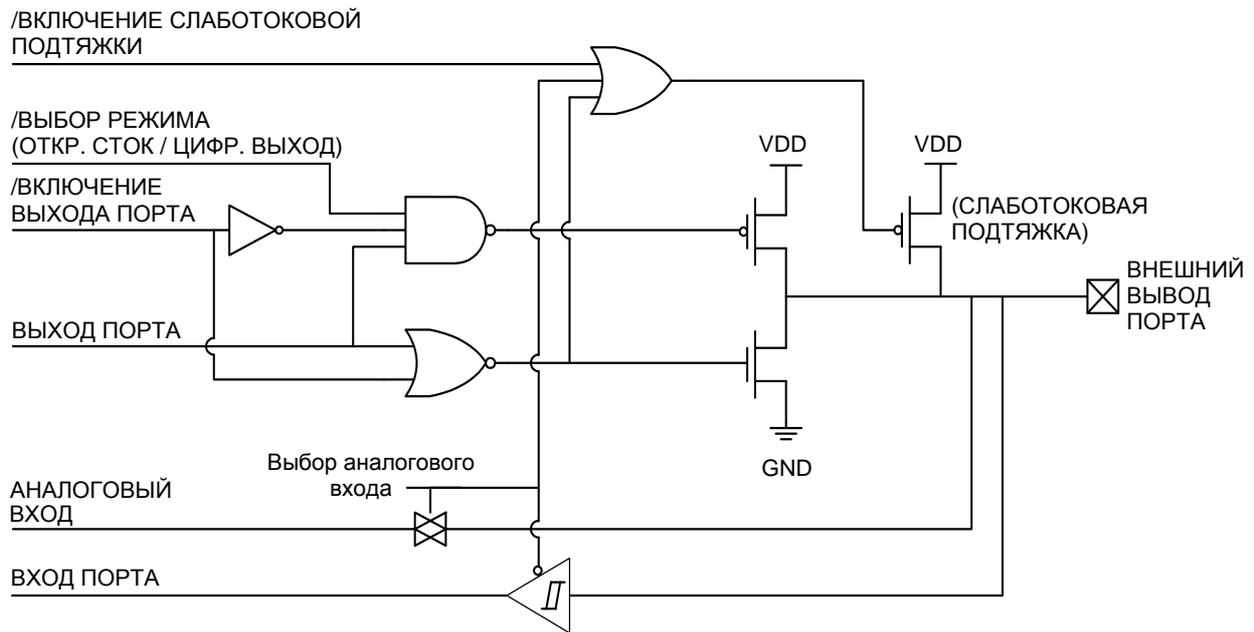


Рисунок 14.2. Структурная схема ячейки порта ввода/вывода



## 14.1. Приоритетный декодер матрицы

Приоритетный декодер матрицы (см. рис.14.3) назначает приоритет каждой функции ввода/вывода, начиная с выводов УАППО. Если какой-либо цифровой ресурс выбран, то этому ресурсу назначается неназначенный вывод порта с наименьшим приоритетом (кроме УАППО, которому всегда назначаются выводы 4 и 5). Если вывод порта назначен, то матрица пропускает этот вывод при назначении следующего выбранного ресурса. Кроме этого, матрица будет пропускать выводы порта, если соответствующие им биты в регистрах PnSKIP установлены в 1. Регистры PnSKIP позволяют программе настроить матрицу таким образом, чтобы она пропускала выводы порта, используемые в качестве аналоговых входов, специализированных портов или портов ввода/вывода общего назначения.

**Важное замечание относительно конфигурации матрицы:** Если вывод порта закреплен за периферийным модулем без использования матрицы, то соответствующий ему бит в регистре PnSKIP должен быть установлен в 1. Это касается P0.7, если используется VREF, P0.3 и/или P0.2, если включена схема возбуждения внешнего генератора, P0.6, если АЦП настроен на использование внешнего сигнала запуска преобразования (CNVSTR), а также любых выводов АЦП или компараторов. Матрица пропускает выбранные выводы, как если бы они были уже назначены, и переходит к следующему неназначенному выводу. На рис.14.3 показаны приоритеты декодера матрицы без пропуска каких-либо выводов порта (POSKIP, P1SKIP, P2SKIP = 0x00); На рис.14.4 показаны приоритеты декодера матрицы с пропуском XTAL1 (P0.2) и XTAL2 (P0.3) (POSKIP = 0x0C).

**Рисунок 14.3. Приоритетный декодер матрицы без пропуска каких-либо выводов**

	P0								P1								P2							
SF Сигналы	XTAL1		XTAL2		CNVSTR		VREF																	
PIN I/O	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
TX0																								
RX0																								
SCK																								
MISO																								
MOSI																								
NSS*									*NSS разводится только в 4-х проводном режиме SPI															
SDA																								
SCL																								
CP0																								
CP0A																								
CP1																								
CP1A																								
SYSClk																								
CEX0																								
CEX1																								
CEX2																								
CEX3																								
CEX4																								
ECI																								
T0																								
T1																								
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	POSKIP[0:7]								P1SKIP[0:7]								P2SKIP[0:3]							

Недоступные сигналы

- Выводы порта, потенциально доступные периферийным модулям.
- SF Сигналы. Сигналы специальных функций, которые не назначаются матрицей. Если эти сигналы разрешены, то матрица должна быть настроена таким образом, чтобы пропускать соответствующие им выводы портов при назначении выводов.

Рисунок 14.4. Приоритетный декодер матрицы с пропуском выводов подключения кварцевого резонатора

	P0								P1								P2							
SF Сигналы	XTAL1		XTAL2		CNVSTR		VREF																	
PIN I/O	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
TX0																								
RX0																								
SCK																								
MISO																								
MOSI																								
NSS*																								
SDA																								
SCL																								
CP0																								
CP0A																								
CP1																								
CP1A																								
SYSCLK																								
CEX0																								
CEX1																								
CEX2																								
CEX3																								
CEX4																								
ECI																								
T0																								
T1																								
	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	P0SKIP[0:7]								P1SKIP[0:7]								P2SKIP[0:3]							



Выводы порта, потенциально доступные периферийным модулям.

SF Сигналы

Сигналы специальных функций, которые не назначаются матрицей. Если эти сигналы разрешены, то матрица должна быть настроена таким образом, чтобы пропускать соответствующие им выводы портов при назначении выводов.

Недоступные сигналы

Регистры XBR0 и XBR1 используются для назначения цифровых ресурсов внешним выводам порта ввода/вывода. Следует иметь в виду, что если выбран SMBus, то матрица назначает оба вывода, связанные с модулем SMBus (SDA и SCL); если выбран УАПП, то матрица назначает оба вывода, связанные с модулем УАПП (TX и RX). Назначение выводов УАППО фиксировано с целью обеспечения возможности самозагрузки: TX0 всегда назначается выводу P0.4; RX0 всегда назначается выводу P0.5. Назначение приоритетных функций выводам порта ввода/вывода осуществляется последовательно в порядке их следования.

**Важное замечание:** SPI может функционировать в 3- или 4-проводном режимах, в зависимости от состояния бит NSSMD1-NSSMD0 в регистре SPI0CN. В соответствии с режимом работы SPI сигнал NSS либо назначается, либо не назначается выводу порта.



## 14.2. Инициализация порта ввода/вывода

Инициализация порта ввода/вывода осуществляется следующим образом:

1. Выбрать тип входа (аналоговый или цифровой) для всех выводов порта, используя регистр настройки входов порта (PnMDIN).
2. Выбрать тип выхода (с открытым стоком или двухтактный цифровой) для всех выводов порта, используя регистр настройки выходов порта (PnMDOUT).
3. Выбрать все выходы, которые должны пропускаться матрицей при назначении выводов, используя регистры выбора пропускаемых выводов (PnSKIP).
4. Назначить выходы порта требуемым периферийным модулям (XBR0, XBR1).
5. Включить матрицу (XBARE = '1').

Все выходы порта должны быть настроены как аналоговые или как цифровые входы. Любые выходы, используемые в качестве входов компаратора или АЦП, должны быть настроены как аналоговые входы. Если вывод настроен как аналоговый вход, то его слаботочная подтяжка, цифровой драйвер и цифровой приемник отключаются. Это позволяет снизить энергопотребление и уменьшить уровень шумов на аналоговом входе. Выводы, настроенные как цифровые входы, все равно могут использоваться аналоговыми периферийными модулями; однако это не рекомендуется. Чтобы настроить вывод порта как цифровой вход, следует сбросить в 0 соответствующий бит в регистре PnMDOUT и установить в 1 соответствующий бит регистра-защелки порта (регистр Pn).

Кроме этого, все аналоговые входы необходимо настроить таким образом, чтобы они пропускались матрицей при назначении выводов (это достигается установкой в 1 соответствующих бит в регистрах PnSKIP). Тип входа устанавливается с помощью соответствующих бит регистра PnMDIN (1 – цифровой вход, 0 – аналоговый вход). При сбросе все выходы настраиваются по умолчанию как цифровые входы. Подробное описание регистров PnMDIN приведено на рис.14.8.

Параметры выходных драйверов выводов порта задаются в помощью регистров настройки выходов порта (PnMDOUT). Выходной драйвер каждого порта можно настроить либо как цифровой двухтактный выход, либо как выход с открытым стоком. Такая настройка не осуществляется автоматически; ее необходимо выполнить даже для цифровых ресурсов, выбранных в регистрах XBRn. Единственным исключением из этого правила являются выходы SMBus (SDA, SCL), которые настраиваются как выходы с открытым стоком независимо от значения PnMDOUT. Если бит WEAKPUD в регистре XBR1 сброшен в 0, то слаботочная подтяжка отключается у всех выводов портов, настроенных как выходы с открытым стоком. Бит WEAKPUD не влияет на выходы, настроенные как цифровые двухтактные выходы. Более того, слаботочная подтяжка отключается у выхода, на который выведен лог.'0', чтобы предотвратить нежелательное увеличение энергопотребления.

Для выбора цифровых ресурсов, требуемых для конкретного проекта, необходимо загрузить регистры XBR0 и XBR1 соответствующими значениями. Установка в 1 бита XBARE в регистре XBR1 включает матрицу. До включения матрицы внешние выходы остаются стандартными портами ввода/вывода (настроенными на вход), независимо от значений регистров XBRn. Зная значение регистров XBRn, можно определить разводку выводов, используя таблицу декодирования приоритетов; мастер конфигурации программного пакета Silicon Labs IDE также позволяет определить разводку выводов портов на основе значений регистров XBRn.

**Важное замечание:** Чтобы использовать порты P0, P1 и P2.0 – P2.3 как стандартные порты ввода/вывода в режиме выходов, необходимо включить матрицу. Выходные драйверы этих портов отключаются при выключении матрицы. P2.4 – P2.7 и P3.0 всегда функционируют как стандартные порты ввода/вывода общего назначения.

Рисунок 14.5. XBR0: Регистр 0 матрицы портов ввода/вывода

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
CP1AE	CP1E	CP0AE	CP0E	SYSCKE	SMB0E	SPI0E	URT0E	SFR Адрес: 0xE1
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: CP1AE: Бит подключения асинхронного выхода Компаратора 1 (CP1) 0: Асинхронный выход CP1 не соединен с выводом порта. 1: Асинхронный выход CP1 соединен с выводом порта.</p>								
<p>Бит 6: CP1E: Бит подключения выхода Компаратора 1 (CP1) 0: CP1 не соединен с выводом порта. 1: CP1 соединен с выводом порта.</p>								
<p>Бит 5: CP0AE: Бит подключения асинхронного выхода Компаратора 0 (CP0) 0: Асинхронный выход CP0 не соединен с выводом порта. 1: Асинхронный выход CP0 соединен с выводом порта.</p>								
<p>Бит 4: CP0E: Бит подключения выхода Компаратора 0 (CP0) 0: CP0 не соединен с выводом порта. 1: CP0 соединен с выводом порта.</p>								
<p>Бит 3: SYSCKE: Бит подключения выхода /SYSCLK 0: Выход /SYSCLK не соединен с выводом порта. 1: Выход /SYSCLK соединен с выводом порта.</p>								
<p>Бит 2: SMB0E: Бит подключения входов/выходов модуля SMBus0 0: Входы/выходы модуля SMBus0 не соединены с выводами порта. 1: Входы/выходы модуля SMBus0 соединены с выводами порта.</p>								
<p>Бит 1: SPI0E: Бит подключения входов/выходов модуля SPI0 0: Входы/выходы модуля SPI0 не соединены с выводами порта. 1: Входы/выходы модуля SPI0 соединены с выводами порта.</p>								
<p>Бит 0: URT0E: Бит подключения входов/выходов УАППО 0: Входы/выходы УАППО не соединены с выводами порта. 1: TX0 и RX0 соединены с выводами P0.4 и P0.5 соответственно.</p>								



**Рисунок 14.6. XBR1: Регистр 1 матрицы портов ввода/вывода**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
WEAKPUD	XBARE	T1E	T0E	ECIE	PCA0ME			00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xE2

Бит 7: WEAKPUD: Бит отключения слаботочковых подтяжек портов ввода/вывода.  
 0: Слаботочковые подтяжки включены (кроме портов, чьи выходы настроены как аналоговые входы или двухтактные цифровые выходы)  
 1: Слаботочковые подтяжки отключены

Бит 6: XBARE: Бит включения матрицы.  
 0: Матрица отключена. Все драйверы портов отключены.  
 1: Матрица включена

Бит 5: T1E: Бит подключения T1.  
 0: T1 не соединен с выводом порта.  
 1: T1 соединен с выводом порта.

Бит 4: T0E: Бит подключения T0.  
 0: T0 не соединен с выводом порта.  
 1: T0 соединен с выводом порта.

Бит 3: ECIE: Бит подключения внешнего входа счетчика ПМС.  
 0: ЕСИ не соединен с выводом порта.  
 1: ЕСИ соединен с выводом порта.

Биты 2-0: PCA0ME: Биты подключения входов/выходов модуля ПМС  
 000: Все входы/выходы модуля ПМС не соединены с выводами порта.  
 001: СЕХ0 соединен с выводом порта.  
 010: СЕХ0, СЕХ1 соединены с двумя выводами порта.  
 011: СЕХ0, СЕХ1, СЕХ2 соединены с тремя выводами порта.  
 100: СЕХ0, СЕХ1, СЕХ2, СЕХ3 соединены с четырьмя выводами порта.  
 101: СЕХ0, СЕХ1, СЕХ2, СЕХ3, СЕХ4 соединены с пятью выводами порта.  
 110: Зарезервировано  
 111: Зарезервировано

### 14.3. Порт ввода/вывода общего назначения

Выводы портов, которые не назначены матрицей и не используются аналоговыми периферийными модулями, можно использовать в качестве выводов ввода/вывода общего назначения. Порты 3-0 доступны с помощью соответствующих SFR-регистров как в побайтном, так и в побитном режимах адресации. При записи в порт значение, записываемое в SFR-регистр, «защелкивается»; это позволяет удерживать на каждом выводе порта выходное значение. При чтении логические уровни входных выводов портов возвращаются независимо от значений регистров XBRn (т.е. даже если вывод назначен матрицей другому сигналу, регистр порта все равно может прочитать логическое состояние на соответствующем входе). Исключением являются команды типа *чтение-модификация-запись*. При работе с SFR-регистром порта командами типа *чтение-модификация-запись* являются следующие команды: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ, а также MOV, CLR или SETB, если они адресуют отдельный бит в SFR-регистре порта. В случае использования этих команд считывается, модифицируется и записывается обратно значение регистра (а не вывода).



## Рисунок 14.7. P0: Регистр данных Porta 0

R/W	Значение при сбросе: 11111111 SFR Адрес: 0x80							
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: P0.[7:0]: Биты выходной защелки порта 0.  
 Запись – выходной сигнал появляется на внешних выводах в зависимости от состояния регистров матрицы (если XBARE = 1).  
 0: Выход в состоянии лог. 0  
 1: Выход в состоянии лог. 1 (в высокоимпедансном состоянии, если соответствующий бит POMDOUT.n = 0)  
 Чтение – Всегда читается как '0', если в регистре POMDIN выбран аналоговый вход. Читается непосредственно состояние вывода порта, если настроен как цифровой вход.  
 0: На выводе P0.n низкий логический уровень.  
 1: На выводе P0.n высокий логический уровень.

## Рисунок 14.8. POMDIN: Регистр настройки входов Porta 0

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xF1							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: POMDIN.[7:0]: Биты выбора режима входов Porta 0.  
 Если вывод настроен как аналоговый вход, то его слаботочковая подтяжка, цифровой драйвер и цифровой приемник отключаются.  
 0: Вывод P0.n настроен как аналоговый вход.  
 1: Вывод P0.n не настроен как аналоговый вход.

Рисунок 14.9. P0MDOUT: Регистр настройки выходов Porta 0

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xA4

Биты 7-0: P0MDOUT.[7:0]: Биты настройки выходного драйвера порта 0: игнорируются, если соответствующий бит в регистре P0MDIN сброшен в 0.  
 0: Соответствующий вывод P0.n настроен как выход с открытым стоком.  
 1: Соответствующий вывод P0.n настроен как цифровой двухтактный выход.

Примечание: Если сигналы SDA и SCL появляются на любом выводе порта, то каждый из этих выводов будет настроен как выход с открытым стоком независимо от значения регистра P0MDOUT.

Рисунок 14.10. P0SKIP: Регистр выбора выводов Porta 0, пропускаемых матрицей

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0xD4

Биты 7-0: P0SKIP.[7:0]: Биты выбора выводов порта 0, пропускаемых матрицей при назначении выводов. Эти биты выбирают выводы порта, пропускаемые декодером матрицы. Выводы порта, используемые как аналоговые входы (для АЦП или компаратора) или используемые для специальных целей (вход VREF, схема внешнего генератора, вход CNVSTR) должны пропускаться матрицей.  
 0: Соответствующий вывод P0.n не пропускается матрицей при назначении выводов.  
 1: Соответствующий вывод P0.n пропускается матрицей при назначении выводов.

## Рисунок 14.11. P1: Регистр данных Порты 1

R/W	Значение при сбросе: 11111111 SFR Адрес: 0x90							
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: P1.[7:0]: Биты выходной защелки порта 1.

Запись – выходной сигнал появляется на внешних выводах в зависимости от состояния регистров матрицы (если XBARE = 1).

0: Выход в состоянии лог. 0

1: Выход в состоянии лог. 1 (в высокоимпедансном состоянии, если соответствующий бит P1MDOUT.n = 0)

Чтение – Всегда читается как '0', если в регистре P1MDIN выбран аналоговый вход. Читается непосредственно состояние вывода порта, если настроен как цифровой вход.

0: На выводе P1.n низкий логический уровень.

1: На выводе P1.n высокий логический уровень.

## Рисунок 14.12. P1MDIN: Регистр настройки входов Порты 1

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xF2							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: P1MDIN.[7:0]: Биты выбора режима входов Порты 1.

Если вывод настроен как аналоговый вход, то его слаботочковая подтяжка, цифровой драйвер и цифровой приемник отключаются.

0: Вывод P1.n настроен как аналоговый вход.

1: Вывод P1.n не настроен как аналоговый вход.

Рисунок 14.13. P1MDOUT: Регистр настройки выходов Porta 1

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xA5							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: P1MDOUT.[7:0]: Биты настройки выходного драйвера порта 1: игнорируются, если соответствующий бит в регистре P1MDIN сброшен в 0.  
 0: Соответствующий вывод P1.n настроен как выход с открытым стоком.  
 1: Соответствующий вывод P1.n настроен как цифровой двухтактный выход.

Рисунок 14.14. P1SKIP: Регистр выбора выводов Porta 1, пропускаемых матрицей

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD5							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: P1SKIP.[7:0]: Биты выбора выводов порта 1, пропускаемых матрицей при назначении выводов. Эти биты выбирают выводы порта, пропускаемые декодером матрицы. Выводы порта, используемые как аналоговые входы (для АЦП или компаратора) или используемые для специальных целей (вход VREF, схема внешнего генератора, вход CNVSTR) должны пропускаться матрицей.  
 0: Соответствующий вывод P1.n не пропускается матрицей при назначении выводов.  
 1: Соответствующий вывод P1.n пропускается матрицей при назначении выводов.

### Рисунок 14.15. P2: Регистр данных Порты 2

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xA0							
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: P2.[7:0]: Биты выходной защелки порта 1.

Запись – выходной сигнал появляется на внешних выводах в зависимости от состояния регистров матрицы (если XBARE = 1).

0: Выход в состоянии лог. 0

1: Выход в состоянии лог. 1 (в высокоимпедансном состоянии, если соответствующий бит P2MDOUT.n = 0)

Чтение – Всегда читается как '0', если в регистре P1MDIN выбран аналоговый вход. Читается непосредственно состояние вывода порта, если настроен как цифровой вход.

0: На выводе P2.n низкий логический уровень.

1: На выводе P2.n высокий логический уровень.

Примечание: P2.4 – P2.7 доступны только в МК C8051F320. Для записи в эти порты не требуется XBARE = '1'.

### Рисунок 14.16. P2MDIN: Регистр настройки входов Порты 2

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xF3							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: P2MDIN.[7:0]: Биты выбора режима входов Порты 2.

Если вывод настроен как аналоговый вход, то его слаботочковая подтяжка, цифровой драйвер и цифровой приемник отключаются.

0: Вывод P2.n настроен как аналоговый вход.

1: Вывод P2.n не настроен как аналоговый вход.

Примечание: P2.4 – P2.7 доступны только в МК C8051F320.

Рисунок 14.17. P2MDOUT: Регистр настройки выходов Porta 2

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xA6							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: P2MDOUT.[7:0]: Биты настройки выходного драйвера порта 2: игнорируются, если соответствующий бит в регистре P2MDIN сброшен в 0.  
 0: Соответствующий вывод P2.n настроен как выход с открытым стоком.  
 1: Соответствующий вывод P2.n настроен как цифровой двухтактный выход.

Примечание: P2.4 – P2.7 доступны только в МК C8051F320.

Рисунок 14.18. P2SKIP: Регистр выбора выводов Porta 2, пропускаемых матрицей

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD6							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Биты 3-0: P2SKIP.[3:0]: Биты выбора выводов порта 2, пропускаемых матрицей при назначении выводов. Эти биты выбирают выводы порта, пропускаемые декодером матрицы. Выводы порта, используемые как аналоговые входы (для АЦП или компаратора) или используемые для специальных целей (вход VREF, схема внешнего генератора, вход CNVSTR) должны пропускаться матрицей.  
 0: Соответствующий вывод P2.n не пропускается матрицей при назначении выводов.  
 1: Соответствующий вывод P2.n пропускается матрицей при назначении выводов.

### Рисунок 14.19. P3: Регистр данных Порты 3

R/W	Значение при сбросе: 11111111 SFR Адрес: 0xB0							
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Биты 7-0: P3.[7:0].

Запись – выходной сигнал появляется на внешних выводах.

0: Выход в состоянии лог. 0

1: Выход в состоянии лог. 1 (в высокоимпедансном состоянии, если соответствующий бит P3MDOUT.n = 0)

Чтение – Всегда читается как '0', если в регистре P3MDIN выбран аналоговый вход. Читается непосредственно состояние вывода порта, если настроен как цифровой вход.

0: На выводе P3.n низкий логический уровень.

1: На выводе P3.n высокий логический уровень.

### Рисунок 14.20. P3MDIN: Регистр настройки входов Порты 3

R/W	Значение при сбросе: 00000001 SFR Адрес: 0xF4							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-1: Не используются. Читаются как 0000000b. Запись не оказывает никакого влияния.

Бит 0: Бит выбора режима входа порта P3.0.  
Если вывод настроен как аналоговый вход, то его слаботочковая подтяжка, цифровой драйвер и цифровой приемник отключаются.

0: Вывод P3.0 настроен как аналоговый вход.

1: Вывод P3.0 не настроен как аналоговый вход.

Рисунок 14.21. P3MDOUT: Регистр настройки выходов Порты 3

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xA7							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-1: Не используются. Читаются как 0000000b. Запись не оказывает никакого влияния.

Бит 0: Бит настройки выходного драйвера порта P3.0: игнорируется, если соответствующий бит в регистре P3MDIN сброшен в 0.  
 0: Вывод P3.0 настроен как выход с открытым стоком.  
 1: Вывод P3.0 настроен как цифровой двухтактный выход.

Таблица 14.1. Электрические характеристики порта ввода/вывода

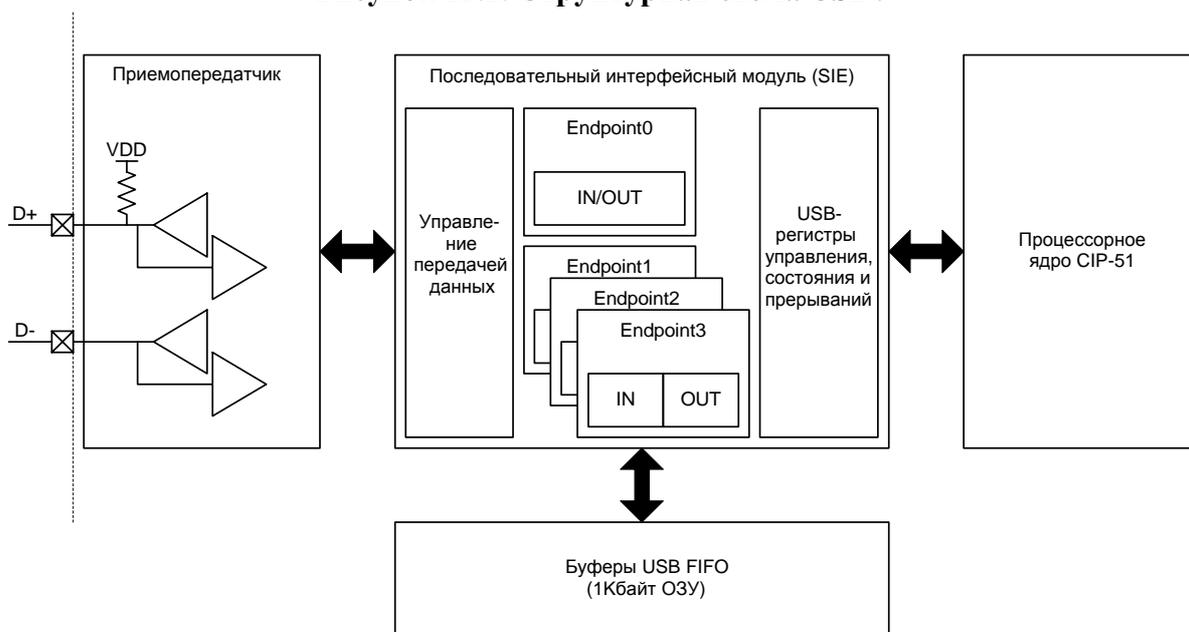
VDD = 2.7 ... 3.6В, T = от -40°C до +85°C, если не указано иное.

ПАРАМЕТР	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
Выходное напряжение высокого уровня (V <sub>OH</sub> )	I <sub>OH</sub> = -10мкА, порт ввода/вывода – цифровой вход/выход	VDD – 0.1			В
	I <sub>OH</sub> = -3мА, порт ввода/вывода – цифровой вход/выход	VDD – 0.7			
	I <sub>OH</sub> = -10мА, порт ввода/вывода – цифровой вход/выход	VDD – 0.8			
Выходное напряжение низкого уровня (V <sub>OL</sub> )	I <sub>OL</sub> = 10мкА		1.0	0.1	В
	I <sub>OL</sub> = 8.5мА			0.6	
	I <sub>OL</sub> = 25мА				
Входное напряжение высокого уровня (V <sub>IH</sub> )		2.0			В
Входное напряжение низкого уровня (V <sub>IL</sub> )				0.8	В
Ток утечки входа	слаботочковая подтяжка отключена			±1	мкА
	слаботочковая подтяжка включена, V <sub>IN</sub> = 0В		25	50	

## 15. USB-контроллер (USB0)

МК C8051F320/1 содержат законченный высокоскоростной/низкоскоростной функциональный USB-контроллер, позволяющий разрабатывать периферийные устройства для шины USB\*. Функциональный USB-контроллер (USB0) состоит из последовательного интерфейсного модуля (Serial Interface Engine - SIE), USB-приемопередатчика (включая согласующие резисторы и настраиваемые подтягивающие резисторы), буфера FIFO объемом 1Кбайт и схемы восстановления синхронизации, предназначенной для обеспечения возможности функционирования без кварцевого резонатора. Никакие внешние компоненты не требуются. Функциональный USB-контроллер и USB-приемопередатчик совместимы со спецификацией USB 2.0.

**Рисунок 15.1. Структурная схема USB0**



**Важное примечание:** Этот документ предполагает полное понимание протокола USB. Термины и аббревиатуры, используемые в этом документе, определены в спецификации USB. Рекомендуется ознакомиться с последней версией спецификации USB.

**Примечание переводчика:** В этой главе будут использоваться следующие понятия:

*Endpoint* – конечная точка подключения

*IN Endpoint* – конечная точка подключения, настроенная на прием данных

*OUT Endpoint* – конечная точка подключения, настроенная на передачу данных

\*МК C8051F320/1 нельзя использовать на шине USB в качестве хост-контроллера.

### 15.1. Адресация Endpoint

Всего доступно восемь каналов Endpoint. Управляющая Endpoint (Endpoint0) всегда функционирует как двунаправленная IN/OUT Endpoint. Остальные Endpoint реализованы как три пары каналов IN/OUT Endpoint:

**Таблица 15.1. Схема адресации Endpoint**

Endpoint	Связанные с Endpoint каналы	Адрес протокола USB
Endpoint0	Endpoint0 IN	0x00
	Endpoint0 OUT	0x00
Endpoint1	Endpoint1 IN	0x81
	Endpoint1 OUT	0x01
Endpoint2	Endpoint2 IN	0x82
	Endpoint OUT	0x02
Endpoint3	Endpoint3 IN	0x83
	Endpoint3 OUT	0x03

### 15.2. Приемопередатчик USB

USB-приемопередатчик настраивается в регистре USB0XCN (см. рис.15.2). С помощью этого регистра можно включить/отключить приемопередатчик, подключить/отключить подтягивающий резистор, а также выбрать скоростной режим (высокоскоростной/низкоскоростной). Если бит SPEED = 1, то USB0 функционирует в высокоскоростном режиме и встроенный подтягивающий резистор (если он подключен) появляется на выводе D+. Если бит SPEED = 0, то USB0 функционирует в низкоскоростном режиме и встроенный подтягивающий резистор (если он подключен) появляется на выводе D-. Биты 4-0 регистра USB0XCN можно использовать для тестирования приемопередатчика, как описано на рис.15.2. Подтягивающий резистор подключается только в том случае, если присутствует напряжение VBUS (подробная информация об обнаружении VBUS приведена в разделе 8.2 «Детектирование VBUS» на стр.69).

**Важное примечание:** Тактирование USB должно быть активировано до включения приемопередатчика.



**Рисунок 15.2. USB0XCN: Регистр управления приемопередатчиком USB0**

R/W	R/W	R/W	R/W	R/W	R	R	R	Значение при сбросе: 00000000 SFR Адрес: 0xD7																				
PREN	PHYEN	SPEED	PHYTST1	PHYTST0	DFREC	Dp	Dn																					
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0																					
<p><b>Бит 7: PREN:</b> Подключение внутреннего подтягивающего резистора.  Место подключения подтягивающего резистора (D+ или D-) определяется битом SPEED.  0: Внутренний подтягивающий резистор отключен (МК физически изолирован от шины USB).  1: Внутренний подтягивающий резистор подключен, если присутствует VBUS (МК подключен к шине USB).</p> <p><b>Бит 6: PHYEN:</b> Активация физического уровня.  Этот бит включает/отключает приемопередатчик физического уровня USB0.  0: Приемопередатчик отключен (остановлен).  1: Приемопередатчик включен (нормальный режим работы).</p> <p><b>Бит 5: SPEED:</b> Выбор скоростного режима USB0.  Этот бит выбирает скорость передачи данных USB0.  0: USB0 функционирует в низкоскоростном режиме. Если внутренний подтягивающий резистор подключен, то он появляется на линии D-.  1: USB0 функционирует в высокоскоростном режиме. Если внутренний подтягивающий резистор подключен, то он появляется на линии D+.</p> <p><b>Биты 4-3: PHYTST1-0:</b> Тестирование физического уровня.  Эти биты можно использовать для тестирования приемопередатчика USB0.</p> <table border="1" data-bbox="336 1070 1257 1234"> <thead> <tr> <th>PHYTST[1-0]</th> <th>Режим</th> <th>D+</th> <th>D-</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>Режим 0: Нормальный режим (не тестирование)</td> <td>X</td> <td>X</td> </tr> <tr> <td>01b</td> <td>Режим 1: Дифференциальная '1'</td> <td>1</td> <td>0</td> </tr> <tr> <td>10b</td> <td>Режим 2: Дифференциальный '0'</td> <td>0</td> <td>1</td> </tr> <tr> <td>11b</td> <td>Режим 3: Синфазный '0'</td> <td>0</td> <td>0</td> </tr> </tbody> </table> <p><b>Бит 2: DFREC:</b> Дифференциальный приемник.  Состояние этого бита показывает текущее дифференциальное значение, присутствующее на линиях D+ и D-, когда PHYEN = 1.  0: На шине присутствует дифференциальный '0'.  1: На шине присутствует дифференциальная '1'.</p> <p><b>Бит 1: Dp:</b> Состояние сигнала D+.  Этот бит показывает текущий логический уровень на выводе D+.  0: На выводе D+ в текущий момент установлен лог.'0'.  1: На выводе D+ в текущий момент установлена лог.'1'.</p> <p><b>Бит 0: Dn:</b> Состояние сигнала D-.  Этот бит показывает текущий логический уровень на выводе D-.  0: На выводе D- в текущий момент установлен лог.'0'.  1: На выводе D- в текущий момент установлена лог.'1'.</p>									PHYTST[1-0]	Режим	D+	D-	00b	Режим 0: Нормальный режим (не тестирование)	X	X	01b	Режим 1: Дифференциальная '1'	1	0	10b	Режим 2: Дифференциальный '0'	0	1	11b	Режим 3: Синфазный '0'	0	0
PHYTST[1-0]	Режим	D+	D-																									
00b	Режим 0: Нормальный режим (не тестирование)	X	X																									
01b	Режим 1: Дифференциальная '1'	1	0																									
10b	Режим 2: Дифференциальный '0'	0	1																									
11b	Режим 3: Синфазный '0'	0	0																									

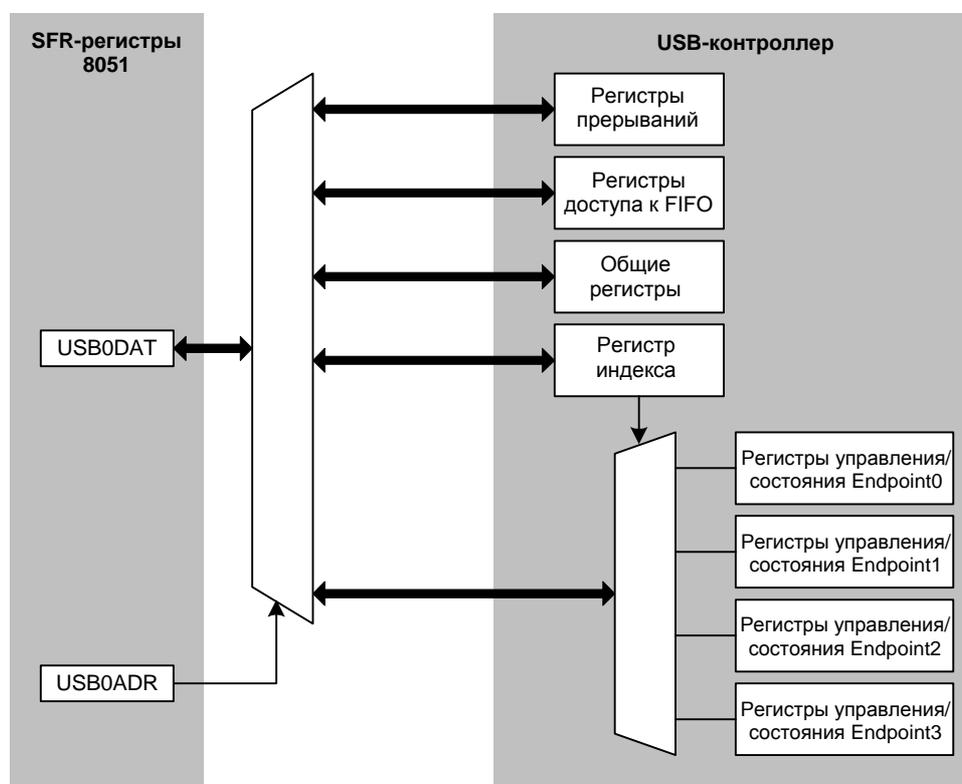
### 15.3. Доступ к регистрам USB-контроллера

Доступ к регистрам контроллера USB0, перечисленным в таблице 15.2, осуществляется с помощью двух SFR-регистров: регистр адреса USB0 (USB0ADR) и регистр данных USB0 (USB0DAT). Регистр USB0ADR определяет, к какому USB-регистру будет происходить обращение при чтении/записи регистра USB0DAT. См. рис.15.3.

Для доступа к регистрам управления/состояния Endpoint необходимо сначала записать в USB-регистр INDEX номер целевой Endpoint. После того как номер целевой Endpoint записан в регистр INDEX, можно обращаться к регистрам управления/состояния, связанным с этой Endpoint. Регистры управления/состояния Endpoint перечислены в разделе «Индексные регистры» таблицы 15.2.

**Важное примечание:** Тактирование USB должно быть активировано при обращении к USB-регистрам

Рисунок 15.3. Схема доступа к регистрам USB0



**Рисунок 15.4. USB0ADR: Регистр косвенного адреса USB0**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x96
BUSY	AUTORD	USBADDR						
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: **BUSY**: Флаг занятости, показывающий, что идет чтение регистра USB0.  
 Этот бит используется при косвенном обращении к регистру USB0. Программа должна установить этот бит в 1, чтобы инициировать операцию чтения регистра USB0, адресуемого битами USBADDR (USB0ADR.[5-0]). Целевой адрес и бит BUSY можно записывать одной командой записи в регистр USB0ADR. После установки в 1 бита BUSY модуль USB0 начнет чтение данных и сбросит в 0 этот бит только тогда, когда данные будут готовы в регистре USB0DAT. Перед записью в регистр USB0DAT программа должна убедиться, что BUSY = 0.  
 Запись:  
 0: Не оказывает никакого влияния.  
 1: Иницируется операция чтения регистра USB0, адресуемого битами USBADDR.  
 Чтение:  
 0: Данные в регистре USB0DAT корректные.  
 1: Модуль USB0 занят (обращается к косвенному регистру); данные в регистре USB0DA некорректные (не готовы).

Бит 6: **AUTORD**: Флаг автоматического режима чтения регистра USB0.  
 Этот бит используется для блочных операций чтения FIFO.  
 0: Бит BUSY должен записываться вручную для каждой операции чтения косвенного регистра USB0.  
 1: Следующая операция чтения косвенного регистра будет инициирована при чтении программой регистра USB0DAT (биты USBADDR не будут изменяться).

Биты 5-0: **USBADDR**: Адрес косвенного регистра USB0.  
 Этот бит содержит 6-разрядный адрес, используемый для косвенного доступа к регистрам ядра USB0. В таблице 15.2 перечислены регистры ядра USB0 и их косвенные адреса. Операции чтения и записи регистра USB0DAT будут обращаться к регистру, на который указывают биты USBADDR.

Рисунок 15.5. USB0DAT: Регистр данных USB0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x97
USB0DAT								
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Этот SFR-регистр используется для косвенных операций чтения и записи регистров USB0.

Процедура записи:

1. Опрашивать BUSY (USB0ADR.7) до обнаружения перехода его в '0'.
2. Загрузить адрес целевого регистра USB0 в биты USBADDR регистра USB0ADR.
3. Записать данные в регистр USB0DAT.
4. Повторять (Шаг 2 можно пропустить, если запись осуществляется в тот же регистр USB0).

Процедура чтения:

1. Опрашивать BUSY (USB0ADR.7) до обнаружения перехода его в '0'.
2. Загрузить адрес целевого регистра USB0 в биты USBADDR регистра USB0ADR.
3. Установить в 1 бит BUSY в регистре USB0ADR (Шаг 2 и шаг 3 можно выполнить одной командой).
4. Опрашивать BUSY (USB0ADR.7) до обнаружения перехода его в '0'.
5. Прочитать данные из регистра USB0DAT.
6. Повторять с шага 2 (Шаг 2 можно пропустить, если осуществляется чтение того же регистра USB0; шаг 3 можно пропустить, если бит AUTORD (USB0ADR.6) установлен в 1).

Рисунок 15.6. INDEX: Регистр индекса Endpoint USB0 (USB регистр)

R	R	R	R	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 USB Адрес: 0x0E
-	-	-	-	EPSEL				
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Биты 3-0: EPSEL: Выбор Endpoint.  
Эти биты определяют, какая Endpoint будет использоваться при обращении к индексным регистрам USB0.

INDEX	Целевая Endpoint
0x0	0
0x1	1
0x2	2
0x3	3
0x4 – 0xF	Зарезервировано



Таблица 15.2. Регистры управления USB0

Обозначение USB-регистра	Адрес USB-регистра	Описание	Номер страницы
<b>Регистры прерываний</b>			
IN1INT	0x02	Флаги прерываний Endpoint0 и Endpoint1-3 IN	157
OUT1INT	0x04	Флаги прерываний Endpoint1-3 OUT	158
CMINT	0x06	Общие флаги прерываний USB	159
IN1IE	0x07	Разрешение прерываний Endpoint0 и Endpoint1-3 IN	160
OUT1IE	0x09	Разрешение прерываний Endpoint1-3 OUT	160
CMIE	0x0B	Разрешение общих прерываний USB	161
<b>Общие регистры</b>			
FADDR	0x00	Адрес USB0 на шине USB	153
POWER	0x01	Управление электропитанием	155
FRAMEL	0x0C	Младший байт номера пакета	156
FRAMEH	0x0D	Старший байт номера пакета	156
INDEX	0x0E	Выбор индекса Endpoint	148
CLKREC	0x0F	Управление схемой восстановления синхронизации	150
FIFO <sub>n</sub>	0x20 – 0x23	Буферы FIFO Endpoint0-3	152
<b>Индексные регистры</b>			
E0CSR	0x11	Управление/состояние Endpoint0	164
EINCSRL		Младший байт регистра управления/состояния Endpoint IN	168
EINCSRH	0x12	Старший байт регистра управления/состояния Endpoint IN	169
EOUTCSRL	0x14	Младший байт регистра управления/состояния Endpoint OUT	171
EOUTCSRH	0x15	Старший байт регистра управления/состояния Endpoint OUT	172
E0CNT	0x16	Количество принятых байт в буфере FIFO Endpoint0	165
E0CNTL		Младший байт счетчика пакетов Endpoint OUT	172
E0CNTH	0x17	Старший байт счетчика пакетов Endpoint OUT	172

### 15.4. Настройка тактирования USB-контроллера

USB0 может функционировать в высокоскоростном или низкоскоростном режимах. Скоростной режим определяется битом SPEED в регистре USB0XCN. При работе в низкоскоростном режиме частота тактирования USB0 должна быть 6МГц. При работе в высокоскоростном режиме частота тактирования USB0 должна быть 48МГц. Источники тактирования описаны в разделе 13 «Генераторы» на стр.117. Выбор источника тактирования USB0 осуществляется с помощью регистра CLKSEL (см. рис.13.6 на стр.125).

Схема восстановления синхронизации использует входящий поток USB-данных для подстройки внутреннего генератора; это позволяет обеспечить соответствие внутреннего генератора (и 4-кратного умножителя тактовой частоты) требованиям, предъявляемым к точности задания частоты тактирования USB. Восстановление синхронизации должно использоваться в следующих конфигурациях:

Скоростной режим	Источник тактирования USB	Вход 4-кратного умножителя тактовой частоты
Высокоскоростной	4-кратный умножитель тактовой частоты	Внутренний генератор
Низкоскоростной	Внутренний генератор/2	-

Если USB0 функционирует в низкоскоростном режиме со схемой восстановления синхронизации, то программа должна установить в 1 бит CRLOW, чтобы включить низкоскоростной режим восстановления синхронизации. Обычно в низкоскоростном режиме восстановление синхронизации не требуется.

При высоком уровне шумов в шине USB для блокировки схемы восстановления синхронизации может быть полезен пошаговый режим. В обычных сетях USB этот режим не требуется (или не рекомендуется).

**Рисунок 15.7. CLKREC: Регистр управления восстановлением синхронизации (USB регистр)**

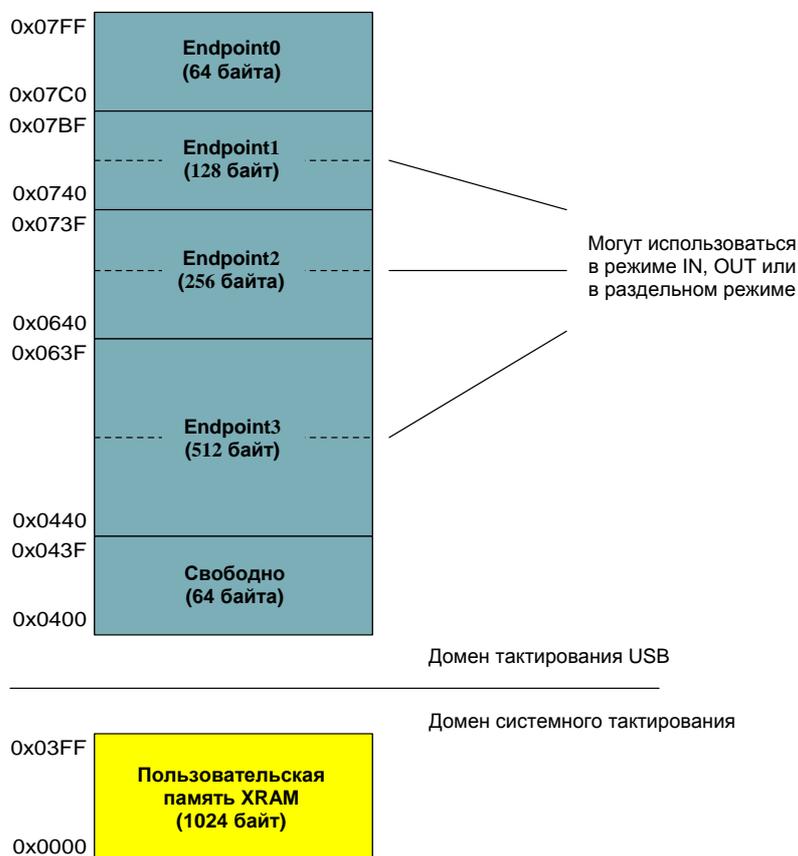
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00001001 USB Адрес: 0x0F
CRE	CRSEN	CRLOW	Зарезервированы					
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: CRE: Разрешение восстановления синхронизации. Этот бит включает/отключает функцию восстановления синхронизации USB. 0: Восстановление синхронизации отключено. 1: Восстановление синхронизации включено.</p>								
<p>Бит 6: CRSEN: Пошаговый режим восстановления синхронизации. Этот бит переводит калибровку генератора в процессе восстановления синхронизации в «пошаговый» режим. 0: Нормальный режим калибровки. 1: Пошаговый режим.</p>								
<p>Бит 5: CRLOW: Низкоскоростной режим восстановления синхронизации. Этот бит должен быть установлен в 1, если USB-устройство функционирует в низкоскоростном режиме и использует восстановление синхронизации. 0: Высокоскоростной режим. 1: Низкоскоростной режим.</p>								
<p>Биты 4-0: Зарезервированы. При чтении возвращают различные значения. Должны быть записаны значением 1001b.</p>								



## 15.5. Управление буфером FIFO

1024 байт памяти XRAM используются как буферы FIFO для USB0. Это пространство FIFO делится между Endpoint0-3, как показано на рис.15.8. Буферы FIFO, выделенные для Endpoint1-3, могут настраиваться на прием данных (IN), на передачу данных (OUT) или на прием/передачу (раздельный режим: половина IN, половина OUT).

**Рисунок 15.8. Распределение USB FIFO**



### 15.5.1. Распределение FIFO

Пространство FIFO для Endpoint1-3 может быть разделено таким образом, чтобы старшая половина пространства FIFO использовалась для IN Endpoint, а младшая половина – для OUT Endpoint. Например: если для FIFO Endpoint3 выбран раздельный режим, то старшие 256 байт (0x0540 – 0x063F) используются для Endpoint3 IN, а младшие 256 байт (0x0440 – 0x053F) используются для Endpoint3 OUT.

Если для FIFO Endpoint не выбран раздельный режим, то области FIFO для пар Endpoint IN/OUT объединяются и образуют единственную область FIFO: IN FIFO или OUT FIFO. В этом случае в один момент времени можно использовать только одно направление передачи данных пары Endpoint IN/OUT. Направление Endpoint (IN/OUT) определяется битом DIRSEL в соответствующем выбранной Endpoint регистре EINCSRH (см. рис.15.23).

### 15.5.2. Двойная буферизация FIFO

Буферы FIFO для Endpoint1-3 могут настраиваться на использование режима двойной буферизации. В этом режиме максимальный размер пакета уменьшается в два раза и FIFO может содержать одновременно два пакета. Этот режим доступен для Endpoint1-3. Если для Endpoint выбран раздельный режим, то двойную буферизацию можно разрешить для IN Endpoint и/или для OUT Endpoint. Если раздельный режим не выбран, то двойную буферизацию можно разрешить для всей области Endpoint FIFO. Максимальные размеры пакетов для каждой конфигурации FIFO приведены в таблице 15.3.

Таблица 15.3. Конфигурации FIFO

Номер Endpoint	Раздельный режим выбран?	Максимальный размер пакета IN (Двойная буферизация запрещена/разрешена)	Максимальный размер пакета IN (Двойная буферизация запрещена/разрешена)
0	-	64	
1	Нет	128/64	
	Да	64/32	64/32
2	Нет	256/128	
	Да	128/64	128/64
3	Нет	512/256	
	Да	256/128	256/128

### 15.5.3. Доступ к FIFO

Доступ к области FIFO каждой Endpoint осуществляется с помощью соответствующего регистра FIFOн. Операция чтения регистра FIFOн выгружает один байт из области FIFO; запись в регистр FIFOн загружает один байт в область FIFO. Если для FIFO Endpoint выбран раздельный режим, то чтение регистра FIFOн выгружает один байт из области FIFO Endpoint OUT, а запись в регистр FIFOн загружает один байт в область FIFO Endpoint IN.

**Рисунок 15.9. FIFOн: Регистры доступа к буферам FIFO Endpoint USB0 (USB регистры)**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
FIFODATA								USB Адрес: 0x20 – 0x23
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Адреса USB 0x20 – 0x23 обеспечивают доступ к четырем парам буферов Endpoint FIFO:

IN/OUT Endpoint FIFO	Адрес USB
0	0x20
1	0x21
2	0x22
3	0x23

Операция записи по адресу FIFO загружает данные в буфер IN FIFO соответствующей Endpoint.  
Операция чтения из адреса FIFO выгружает данные из буфера OUT FIFO соответствующей Endpoint.

## 15.6. Адресация устройства на шине USB

Регистр FADDR содержит текущий адрес USB0 на шине USB. Программа должна записать в регистр FADDR назначенный хост-контроллером 7-битный адрес, полученный в составе команды SET\_ADDRESS. Новый адрес, записанный в регистр FADDR, не будет активирован (т.е. USB0 не будет откликаться на новый адрес) до тех пор, пока не закончится текущий сеанс передачи данных (который обычно заканчивается передачей фазы состояния команды SET\_ADDRESS). После записи программой нового адреса в регистр FADDR бит UPDATE (FADDR.7) аппаратно устанавливается в 1. Бит UPDATE сбрасывается аппаратно после активирования нового адреса, как описано выше.

**Рисунок 15.10. FADDR: Адрес USB0 на шине USB (USB регистр)**

R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
UPDATE	Адрес USB0 на шине USB							00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	USB Адрес: 0x00

Бит 7: UPDATE: Бит обновления адреса USB0 на шине USB.  
 Устанавливается в 1, когда программа записывает регистр FADDR. USB0 сбрасывает этот бит в 0 после активирования нового адреса.  
 0: Последний адрес, записанный в регистр FADDR, активирован.  
 1: Последний адрес, записанный в регистр FADDR, еще не активирован.

Биты 6-0: Адрес USB0 на шине USB.  
 Эти биты содержат 7-битный адрес USB0 на шине USB. Программа должна записывать этот адрес в случае получения в Endpoint0 запроса стандартного устройства SET\_ADDRESS. Новый адрес активируется после окончания запроса устройства.

## 15.7. Настройка и управление USB0

USB-регистр POWER (см. рис.15.11) используется для настройки и управления USB0 на уровне физического устройства (включение/отключение, управление сбросом, остановкой и возобновлением работы, и т.д).

**USB сброс:** Бит USBRST (POWER.3) аппаратно устанавливается в 1, если на шине USB обнаружено событие RESET (сброс). В этом случае происходит следующее:

1. Адрес USB0 сбрасывается (FADDR = 0x00).
2. Буферы Endpoint FIFO очищаются.
3. Регистры управления/состояния сбрасываются в 0x00 (E0CSR, E1NCSRL, E1NCSRH, E0UTCSRL, E0UTCSRH).
4. USB-регистр INDEX сбрасывается в 0x00.
5. Все USB-прерывания (кроме прерывания SUSPEND) разрешаются и соответствующие им флаги сбрасываются.
6. Генерируется прерывание по сбросу USB (если оно разрешено).

Установка бита USBRST вызовет асинхронный сброс от USB0. После асинхронного сброса все USB-регистры принимают значения по умолчанию.

**Режим остановки (Suspend mode):** Если детектирование события SUSPEND разрешено (SUSEN = 1), то USB0 перейдет в режим остановки при обнаружении на шине USB события SUSPEND. Будет сгенерировано прерывание, если оно разрешено (SUSINTE = 1). Процедура обработки прерывания SUSPEND должна выполнить специфичные для конкретного приложения задачи по настройке МК, например, отключение некоторых периферийных модулей и/или перевод источников тактирования в режимы пониженного энергопотребления. Подробная информация о настройке внутреннего генератора (включая описание экономичного режима внутреннего генератора) приведена в разделе 13 «Генераторы» на стр.117.

Вывести USB0 из режима остановки может любое из следующих событий:

1. Обнаружено или сгенерировано событие RESUME (Возобновить работу).
2. Обнаружено событие RESET (Сброс).
3. Произошел сброс МК или USB.

Все вышеперечисленные события могут вывести внутренний генератор из экономичного режима.

**Событие RESUME (Возобновить работу):** USB0 выйдет из режима остановки, если детектирует на шине USB событие RESUME. В случае обнаружения события RESUME будет сгенерировано прерывание RESUME, если оно разрешено (RESINTE = 1). Программа может вызвать Удаленное Пробуждение, установив в 1 бит RESUME (POWER.2). В этом случае для завершения события RESUME программа должна сбросить бит RESUME в 0 спустя 10-15 мс после инициации Удаленного Пробуждения (т.е. после установки RESUME в 1).

**Обновление ISO (ISO Update):** Если программа установит в 1 бит ISOUP (POWER.7), то будет разрешена функция обновления ISO. Если обновление ISO разрешено, то новые пакеты, записанные в ISO IN Endpoint, не будут передаваться до тех пор, пока не будет получен новый признак SOF (Start-Of-Frame – начало пакета). Если ISO IN Endpoint получит признак IN ранее, чем SOF, то USB0 передаст пакет нулевой длины. Если ISOUP = 1, то обновление ISO разрешено для всех ISO Endpoint.

**Включение USB:** После сброса типа POR USB0 отключается. Для включения USB0 необходимо сбросить в 0 бит USBINH (POWER.4). После сброса в 0 бит USBINH может быть установлен в 1 только следующими событиями:

1. Сброс типа POR.
2. Асинхронный сброс от USB0, генерируемый при установке в 1 бита USBRST (POWER.3).

Программа должна полностью настроить USB0 до его включения. Для этого необходимо:

1. Выбрать и включить источник тактирования USB.
2. Сбросить USB0, установив в 1 бит USBRST.
3. Настроить и включить USB-приемопередатчик.
4. Настроить все функции USB0 (прерывания, детектирование SUSPEND).
5. Включить USB0, сбросив в 0 бит USBINH.



**Рисунок 15.11. POWER: Питание USB0 (USB регистр)**

R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	Значение при сбросе:
ISOUP	-	-	USBINH	USBRST	RESUME	SUSMD	SUSEN	00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	USB Адрес: 0x01

**Бит 7: ISOUP: Обновление ISO.**  
 Этот бит влияет на все изохронные IN Endpoint.  
 0: Если программа установит бит INPRDY в 1, то USB0 передаст пакет данных после получения следующего признака IN.  
 1: Если программа установит бит INPRDY в 1, то USB0 передаст пакет данных нулевой длины.

**Биты 6-5:** Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

**Бит 4: USBINH: Запрет USB0.**  
 Этот бит устанавливается в 1 после сброса типа POR или после асинхронного сброса от USB0 (см. описание бита 3). Программа должна сбросить этот бит после окончания процесса инициализации USB0 и приемопередатчика. Программа не может установить этот бит в 1.  
 0: USB0 включен.  
 1: USB0 запрещен. Весь USB-трафик игнорируется.

**Бит 3: USBRST: Детектирование сброса.**  
 Установка этого бита в 1 вызывает асинхронный сброс от USB0. Чтение этого бита позволяет обнаружить состояние сброса на шине USB.  
 Чтение:  
 0: Событие RESET на шине USB не обнаружено.  
 1: На шине USB обнаружено событие RESET.

**Бит 2: RESUME: Возобновление работы.**  
 Программа может инициировать событие RESUME на шине USB с целью вывести USB0 из режима остановки. Если USB0 находится в режиме остановки (SUSMD = 1), то установка этого бита в 1 заставит USB0 генерировать событие RESUME на шине USB (событие «Удаленное Пробуждение»). Чтобы закончить событие RESUME, программа должна через 10-15 мс сбросить в 0 бит RESUME. Когда программа записывает RESUME = '0', генерируется прерывание и бит SUSMD аппаратно сбрасывается в 0.

**Бит 1: SUSMD: Режим остановки.**  
 Когда USB0 переходит в режим остановки, этот бит аппаратно устанавливается в 1. Этот бит аппаратно сбрасывается в 0, когда программа записывает RESUME = '0' или считывает регистр CMINT после обнаружение события RESUME на шине USB.  
 0: USB0 не работает в режиме остановки.  
 1: USB0 работает в режиме остановки.

**Бит 0: SUSEN: Разрешение детектирования события SUSPEND.**  
 0: Детектирование события SUSPEND запрещено. USB0 будет игнорировать событие SUSPEND на шине USB.  
 1: Детектирование события SUSPEND разрешено. USB0 перейдет в режим остановки, если обнаружит на шине USB событие SUSPEND.

Рисунок 15.12. FRAMEL: Младший байт номера пакета USB0 (USB регистр)

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x0C
Младший байт номера пакета								
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: Младший байт номера пакета.  
Этот регистр содержит биты 7 – 0 номера последнего полученного пакета данных.

Рисунок 15.13. FRAMEH: Старший байт номера пакета USB0 (USB регистр)

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x0D
-	-	-	-	-	Старший байт номера пакета			
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-3: Не используются. Читаются как 00000b. Запись не оказывает никакого влияния.

Биты 2-0: Старший байт номера пакета.  
Этот регистр содержит биты 10 – 8 номера последнего полученного пакета данных.



## 15.8. Прерывания

Флаги прерываний модуля USB0 (доступные только для чтения) размещены в USB-регистрах, показанных на рис.15.14.- рис.15.16. Соответствующие биты разрешения прерываний размещены в USB-регистрах, показанных на рис.15.17.- рис.15.19. Прерывание от модуля USB0 генерируется при установке в 1 любого из флагов прерываний USB. Прерывание от модуля USB0 разрешается в SFR-регистре EIE1 (см. раздел 9.3 «Обработка прерываний» на стр.87).

**Важное примечание:** Чтение регистра флагов USB прерываний сбросит в 0 все флаги в этом регистре.

**Рисунок 15.14. IN1INT: Регистр флагов прерываний от IN Endpoint USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x02
-	-	-	-	IN3	IN2	IN1	EP0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Бит 3: IN3: Флаг запроса прерывания от IN Endpoint3.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от IN Endpoint3.  
1: Прерывание от IN Endpoint3 активно.

Бит 2: IN2: Флаг запроса прерывания от IN Endpoint2.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от IN Endpoint2.  
1: Прерывание от IN Endpoint2 активно.

Бит 1: IN1: Флаг запроса прерывания от IN Endpoint1.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от IN Endpoint1.  
1: Прерывание от IN Endpoint1 активно.

Бит 0: EP0: Флаг запроса прерывания от Endpoint0.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от Endpoint0.  
1: Прерывание от Endpoint0 активно.

**Рисунок 15.15. OUT1INT: Регистр флагов прерываний от OUT Endpoint USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x04
-	-	-	-	OUT3	OUT2	OUT1	-	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

**Бит 3: OUT3:** Флаг запроса прерывания от OUT Endpoint3.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от OUT Endpoint3.  
1: Прерывание от OUT Endpoint3 активно.

**Бит 2: OUT2:** Флаг запроса прерывания от OUT Endpoint2.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от OUT Endpoint2.  
1: Прерывание от OUT Endpoint2 активно.

**Бит 1: OUT1:** Флаг запроса прерывания от OUT Endpoint1.  
Этот бит сбрасывается в 0 при чтении программой регистра IN1INT.  
0: Нет прерывания от OUT Endpoint1.  
1: Прерывание от OUT Endpoint1 активно.

**Бит 0:** Не используется. Читаются как 0b. Запись не оказывает никакого влияния.



**Рисунок 15.16. CMINT: Регистр общих прерываний от USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x06
-	-	-	-	SOF	RSTINT	RSUINT	SUSINT	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

**Бит 3: SOF:** Флаг запроса прерывания от признака SOF (Start-Of-Frame – Начало пакета).  
Устанавливается аппаратно при получении признака SOF. Это прерывание синхронизируется аппаратно: прерывание будет генерироваться тогда, когда аппаратура ожидает событие SOF, даже в том случае, если реальный сигнал SOF потерян или поврежден.  
Этот бит сбрасывается в 0 при чтении программой регистра CMINT.  
0: Нет SOF-прерывания.  
1: SOF-прерывание активно.

**Бит 2: RSTINT:** Флаг запроса прерывания от события RESET (Сброс).  
Этот бит устанавливается аппаратно в том случае, если USB0 обнаруживает на шине USB событие RESET.  
Этот бит сбрасывается в 0 при чтении программой регистра CMINT.  
0: Нет прерывания от события RESET.  
1: Прерывание от события RESET активно.

**Бит 1: RSUINT:** Флаг запроса прерывания от события RESUME (Возобновить работу).  
Этот бит устанавливается аппаратно в том случае, если USB0, находясь в режиме остановки, обнаруживает на шине USB событие RESUME.  
Этот бит сбрасывается в 0 при чтении программой регистра CMINT.  
0: Нет прерывания от события RESUME.  
1: Прерывание от события RESUME активно.

**Бит 0: SUSINT:** Флаг запроса прерывания от события SUSPEND.  
Если обнаружение события SUSPEND разрешено (бит SUSEN в регистре POWER), то этот бит устанавливается аппаратно тогда, когда на шине USB обнаруживается событие SUSPEND.  
Этот бит сбрасывается в 0 при чтении программой регистра CMINT.  
0: Нет прерывания от события SUSPEND.  
1: Прерывание от события SUSPEND активно.

**Рисунок 15.17. IN1IE: Регистр разрешения прерываний от IN Endpoint USB0 (USB регистр)**

R/W	Значение при сбросе: 00001111 USB Адрес: 0x07							
-	-	-	-	IN3E	IN2E	IN1E	EP0E	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Бит 3: IN3E: Бит разрешения прерывания от IN Endpoint3.  
0: Прерывание от IN Endpoint3 запрещено.  
1: Прерывание от IN Endpoint3 разрешено.

Бит 2: IN2E: Бит разрешения прерывания от IN Endpoint2.  
0: Прерывание от IN Endpoint2 запрещено.  
1: Прерывание от IN Endpoint2 разрешено.

Бит 1: IN1E: Бит разрешения прерывания от IN Endpoint1.  
0: Прерывание от IN Endpoint1 запрещено.  
1: Прерывание от IN Endpoint1 разрешено.

Бит 0: EP0E: Бит разрешения прерывания от Endpoint0.  
0: Прерывание от Endpoint0 запрещено.  
1: Прерывание от Endpoint0 разрешено.

**Рисунок 15.18. OUT1IE: Регистр разрешения прерываний от OUT Endpoint USB0 (USB регистр)**

R/W	Значение при сбросе: 00001110 USB Адрес: 0x09							
-	-	-	-	OUT3E	OUT2E	OUT1E	-	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Бит 3: OUT3E: Бит разрешения прерывания от OUT Endpoint3.  
0: Прерывание от OUT Endpoint3 запрещено.  
1: Прерывание от OUT Endpoint3 разрешено.

Бит 2: OUT2E: Бит разрешения прерывания от OUT Endpoint2.  
0: Прерывание от OUT Endpoint2 запрещено.  
1: Прерывание от OUT Endpoint2 разрешено.

Бит 1: OUT1E: Бит разрешения прерывания от OUT Endpoint1.  
0: Прерывание от OUT Endpoint1 запрещено.  
1: Прерывание от OUT Endpoint1 разрешено.

Бит 0: Не используется. Читаются как 0b. Запись не оказывает никакого влияния.



**Рисунок 15.19. СМІЕ: Регистр разрешения общих прерываний от USB0 (USB регистр)**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе:
-	-	-	-	SOFE	RSTINTE	RSUINTE	SUSINTE	00000110
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	USB Адрес: 0x0B

Биты 7-4: Не используются. Читаются как 0000b. Запись не оказывает никакого влияния.

Бит 3: SOFE: Бит разрешения прерывания от признака SOF (Start-Of-Frame – Начало пакета).  
0: SOF-прерывание запрещено.  
1: SOF-прерывание разрешено.

Бит 2: RSTINTE: Бит разрешения прерывания от события RESET (Сброс).  
0: Прерывание от события RESET запрещено.  
1: Прерывание от события RESET разрешено.

Бит 1: RSUINTE: Бит разрешения прерывания от события RESUME (Возобновить работу).  
0: Прерывание от события RESUME запрещено.  
1: Прерывание от события RESUME разрешено.

Бит 0: SUSINTE: Бит разрешения прерывания от события SUSPEND.  
0: Прерывание от события SUSPEND запрещено.  
1: Прерывание от события SUSPEND разрешено.

## 15.9. Последовательный интерфейсный модуль

Последовательный интерфейсный модуль (Serial Interface Engine – SIE) выполняет все низкоуровневые функции протокола USB, прерывая процессор при успешной отправке или получении данных. При приеме данных SIE будет прерывать процессор после получения полного пакета данных; соответствующие сигналы квитирования генерируются модулем SIE автоматически. При передаче данных SIE будет прерывать процессор после отправки полного пакета данных и получения соответствующего сигнала подтверждения.

SIE не будет прерывать процессор в случае получения поврежденного/некорректного пакета данных.

## 15.10. Endpoint0

Для управления Endpoint0 используется регистр E0CSR (см. рис.15.20). Для доступа к регистру E0CSR необходимо загрузить в регистр INDEX значение 0x00.

Прерывание от Endpoint0 генерируется в следующих случаях:

1. Пакет данных (OUT или SETUP) получен и загружен в Endpoint0 FIFO. Бит OPRDY (E0CSR.0) аппаратно устанавливается в 1.
2. Пакет данных IN успешно выгружен из Endpoint0 FIFO и передан хост-контроллеру; бит INPRDY аппаратно сбрасывается в 0.
3. Транзакция IN завершена (это прерывание генерируется в течение фазы состояния транзакции).
4. Модуль USB0 аппаратно устанавливает в 1 бит STSTL (E0CSR.2) после окончания управляющей транзакции из-за несоблюдения (ошибок) протокола.
5. Модуль USB0 аппаратно устанавливает в 1 бит SUEND (E0CSR.4), т.к. управляющая транзакция завершилась до программной установки бита DATAEND (E0CSR/3).

Регистр E0CNT (см. рис.15.21) содержит количество полученных в Endpoint0 FIFO байт данных.

Модуль USB0 будет автоматически обнаруживать ошибки протокола и отправлять в ответ условие STALL. Программа может вызвать условие STALL, чтобы прервать текущий сеанс передачи данных. Когда генерируется условие STALL, бит STSTL будет установлен в 1 и генерируется прерывание. Следующие события могут заставить модуль USB0 сгенерировать условие STALL:

1. Хост отправляет признак OUT в течение фазы OUT-данных после установки в 1 бита DATAEND.
  2. Хост отправляет признак IN в течение фазы IN-данных после установки в 1 бита DATAEND.
  3. Хост отправляет пакет, размер которого превышает максимальный размер пакета для Endpoint0.
  4. Хост отправляет пакет DATA1 ненулевой длины в течение фазы состояния транзакции IN.
- Программа устанавливает бит SDSTL (E0CSR.5) в 1.

### **15.10.1. Транзакции SETUP**

Все сеансы передачи команд управления должны начинаться с пакета SETUP. Пакеты SETUP похожи на пакеты OUT, содержащие 8-разрядное поле данных и посылаемые хостом. Пакеты SETUP, содержащие поле команд длиной, отличающейся от 8 байт, будут автоматически отбрасываться модулем USB0. Прерывание от Endpoint0 генерируется, когда данные из пакета SETUP загружаются в FIFO Endpoint0. Программа должна выгрузить команду из FIFO Endpoint0, декодировать ее, выполнить все необходимые действия и установить бит SOPRDY, чтобы показать, что она обслужила пакет OUT.

### **15.10.2. Транзакции IN**

Если получен запрос SETUP, требующий от USB0 передать данные хосту, то хосту будут отправлены один или несколько запросов IN. Для первой транзакции IN программа должна загрузить пакет IN в FIFO Endpoint0 и установить бит INPRDY (E0CSR.1). Прерывание будет сгенерировано в случае успешной передачи пакета IN. Следует иметь в виду, что прерывание не будет генерироваться, если запрос IN получен прежде, чем программа загрузила пакет в FIFO Endpoint0. Если объем запрашиваемых данных превышает максимальный размер пакета для Endpoint0 (переданный хосту), то данные должны быть разделены на несколько пакетов; каждый пакет должен иметь максимальный размер, за исключением последнего пакета, который содержит остаток данных. Если запрашиваемые данные разделяются на несколько пакетов максимального размера (для Endpoint0) без остатка, то последним пакетом данных должен быть пакет нулевой длины, обозначающий конец передачи. Программа должна установить бит DATAEND в 1 после загрузки в FIFO Endpoint0 последнего пакета данных, предназначенного для передачи.

Получение первого признака IN в сеансе передачи команд управления указывает Endpoint0 на то, что она должна перейти в режим передачи. В этом режиме хост должен посылать в Endpoint0 только признаки IN. Если Endpoint0, находясь в режиме передачи, получает признак SETUP или OUT, то бит SUEND (E0CSR.4) устанавливается в 1.

Endpoint0 будет оставаться в режиме передачи до тех пор, пока не произойдет одно из следующих событий:

1. USB0 получает в Endpoint0 признак SETUP или OUT.
2. Программа посылает пакет длиной менее, чем максимальный размер пакета Endpoint0.
3. Программа посылает пакет нулевой длины.

Программа должна установить бит DATAEND (E0CSR.3) в 1, когда выполняет действия, описанные выше в пунктах (2) и (3).

Если в буфере IN FIFO нет готовых пакетов (INPRDY = 0), то SIE должен передать NAK в ответ на признак IN.



### 15.10.3. Транзакции OUT

Если получен запрос SETUP, требующий от хоста передать данные в USB0, то хост отправит один или несколько запросов OUT. Если модуль USB0 успешно примет пакет OUT, то аппаратно установится в 1 бит OPRDY (E0CSR.0) и будет сгенерировано прерывание от Endpoint0. Вслед за этим прерыванием программа должна выгрузить пакет OUT из FIFO Endpoint0 и установить в 1 бит SOPRDY (E0CSR.6).

Если объем данных, требуемых для передачи, превышает максимальный размер пакета для Endpoint0, то данные будут разделены на несколько пакетов. Если запрашиваемые данные разделяются на несколько пакетов максимального для Endpoint0 размера (переданного хосту) без остатка, то хост будет посылать пакет данных нулевой длины, обозначающий конец передачи.

Получение первого признака OUT в сеансе передачи команд управления указывает Endpoint0 на то, что она должна перейти в режим приема. В этом режиме хост должен посылать в Endpoint0 только признаки OUT. Если Endpoint0, находясь в режиме приема, получает признак SETUP или IN, то бит SUEND (E0CSR.4) устанавливается в 1.

Endpoint0 будет оставаться в режиме приема до тех пор, пока не произойдет одно из следующих событий:

1. SEI получает признак SETUP или IN.
2. Хост посылает пакет длиной менее, чем максимальный размер пакета Endpoint0.
3. Хост посылает пакет нулевой длины.

Программа должна установить бит DATAEND (E0CSR.3) в 1, когда получено ожидаемое количество данных. SEI передаст (сгенерирует) условие STALL в том случае, если хост посылает признак OUT после того, как программа установила в 1 бит DATAEND. После передачи STALL будет сгенерировано прерывание и установится в 1 бит STSTL (E0CSR.2).

Рисунок 15.20. E0CSR: Регистр управления Endpoint0 USB0 (USB регистр)

R/W	R/W	R/W	R	R/W	R/W	R/W	R	Значение при сбросе: 00000000 USB Адрес: 0x11
SSUEND	SOPRDY	SDSTL	SUEND	DATAEND	STSTL	INPRDY	OPRDY	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Бит 7: SSUEND: Событие SETUP END обслужено.  <b>Запись:</b> Программа должна установить этот бит в 1 после обработки события SETUP END (бит SUEND). Если программа устанавливает в 1 бит SSUEND, то бит SUEND аппаратно сбрасывается в 0.  <b>Чтение:</b> Это бит всегда читается как '0'.</p>								
<p>Бит 6: SOPRDY: OPRDY обслужено.  <b>Запись:</b> Программа должна установить этот бит в 1 после обработки полученного в Endpoint0 пакета. Если программа устанавливает в 1 бит SOPRDY, то бит OPRDY аппаратно сбрасывается в 0.  <b>Чтение:</b> Это бит всегда читается как '0'.</p>								
<p>Бит 5: SDSTL: Передача STALL.  Программа может установить этот бит в 1 для завершения текущего сеанса передачи данных (в случае обнаружения ошибки, неожиданной передачи запроса, и т.д.). Модуль USB0 аппаратно сбросит этот бит в 0 после передачи сигнала STALL.</p>								
<p>Бит 4: SUEND: SETUP END.  Этот доступный только для чтения бит аппаратно устанавливается в 1 в том случае, если управляющая транзакция завершается прежде, чем программа установит в 1 бит DATAEND. Если программа устанавливает в 1 бит SSUEND, то бит SUEND аппаратно сбрасывается в 0.</p>								
<p>Бит 3: DATAEND: Конец данных.  Программа должна установить этот бит в 1 в следующих случаях:</p> <ol style="list-style-type: none"> <li>1. При установке в 1 бита INPRDY для последнего пакета исходящих данных.</li> <li>2. При установке в 1 бита INPRDY для пакета данных нулевой длины.</li> <li>3. При установке в 1 бита SOPRDY после обработки последнего входящего пакета данных.</li> </ol> <p>Это бит автоматически аппаратно сбрасывается в 0.</p>								
<p>Бит 2: STSTL: Передан STALL.  Модуль USB0 аппаратно устанавливает этот бит в 1 после передачи сигнала STALL. Этот флаг должен сбрасываться программно.</p>								
<p>Бит 1: INPRDY: Пакет IN готов.  Программа должна установить этот бит в 1 после загрузки пакета данных для передачи в FIFO Endpoint0. Модуль USB0 аппаратно сбрасывает этот бит в 0 и генерирует прерывание в следующих случаях:</p> <ol style="list-style-type: none"> <li>1. Пакет передан.</li> <li>2. Пакет переписан входящим пакетом SETUP.</li> <li>3. Пакет переписан входящим пакетом OUT.</li> </ol>								
<p>Бит 0: OPRDY: Пакет OUT готов.  Модуль USB0 аппаратно устанавливает в 1 этот доступный только для чтения бит и генерирует прерывание при получении пакета данных. Этот бит сбрасывается в 0 только в том случае, если программа устанавливает в 1 бит SOPRDY.</p>								



**Рисунок 15.21. E0CSR: Регистр счетчика данных Endpoint0 USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x16
-	E0CNT							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: Не используется. Читается как 0b. Запись не оказывает никакого влияния.

Биты 6-0: E0CNT: Счетчик данных Endpoint0.  
 Это 7-битное число показывает количество байт данных, полученных в FIFO Endpoint0. Это число является корректным только в том случае, если OPRDY = '1'.

### 15.11. Настройка Endpoint1-3

Для управления и настройки Endpoint1-3 используются их собственные наборы регистров управления/состояния: IN-регистры (EINCSRL и EINCSRH) и OUT-регистры (EOUTCSRL и EOUTCSRH). Одновременно в адресном пространстве USB-регистров может отображаться только один набор регистров управления/состояния, определяемый значением регистра INDEX (см. рис.15.6).

Endpoint1-3 можно настроить на прием данных (IN), на передачу данных (OUT) или использовать в раздельном режиме (IN/OUT), как описано в разделе 15.5.1. Режим работы Endpoint (раздельный/нормальный) выбирается в регистре EINCSRH с помощью бита SPLIT.

Если SPLIT = '1', то буфер FIFO соответствующей Endpoint разделен и доступны каналы как приема (IN), так и передачи (OUT) данных.

Если SPLIT = '0', то соответствующая Endpoint настроена либо на прием (IN), либо на передачу (OUT) данных; направление обмена данными выбирается в регистре EINCSRH с помощью бита DIRSEL.

### 15.12. Управление Endpoint1-3 IN

Для управления Endpoint1-3 IN используются USB-регистры EINCSRL и EINCSRH. Все IN Endpoint могут использоваться в прерывистом, блочном или изохронном режимах передачи данных. Для включения изохронного режима (ISO) следует установить в 1 бит ISO в регистре EINCSRH. Управление блочным и прерывистым режимами передачи данных осуществляется аппаратно идентичным образом.

Прерывание от Endpoint1-3 IN генерируется при любом из следующих случаев:

1. Пакет IN успешно передан хосту.
2. Программа устанавливает в 1 бит FLUSH (EINCSRL.3), когда целевой буфер FIFO не пуст.
3. Модуль USB0 аппаратно генерирует событие STALL.

#### 15.12.1. Прерывистый и блочный режимы Endpoint1-3 IN

Если бит ISO (EINCSRH.6) = '0', то целевая Endpoint функционирует в блочном или прерывистом режимах передачи данных. Как только Endpoint настроена на работу в блочном/прерывистом IN режиме (обычно после команды SET\_INTERFACE для Endpoint0), программа должна загрузить пакет IN в буфер IN FIFO Endpoint и установить в 1 бит INPRDY (EINCSRL.0). При получении признака IN модуль USB0 передаст данные, сбросит в 0 бит INPRDY и сгенерирует прерывание.

Установка в 1 бита INPRDY без записи каких-либо данных в FIFO Endpoint вызовет передачу пакета нулевой длины после получения следующего признака IN.

Блочный или прерывистый канал передачи данных можно отключить (или остановить), установив в 1 бит SDSTL (EINCSRL.4). Если SDSTL = 1, то модуль USB0 будет отвечать на все запросы IN сигналом STALL. Каждый раз при генерировании события STALL будет генерироваться прерывание и устанавливаться в 1 бит STSTL (EINCSRL.5). Бит STSTL должен сбрасываться в 0 программно.

Модуль USB0 автоматически сбросит в 0 бит INPRDY при открытии в FIFO Endpoint области, занимаемой пакетом. Следует иметь в виду, что если для целевой Endpoint разрешена двойная буферизация, то программа имеет возможность загрузить в буфер IN FIFO Endpoint одновременно два пакета данных. В этом случае модуль USB0 сбросит в 0 бит INPRDY сразу же после того, как программа загрузит первый пакет в буфер FIFO и установит в 1 бит INPRDY. Прерывание не будет генерироваться в этом случае; прерывание будет генерироваться только тогда, когда передан пакет данных.

Если программа установит в 1 бит FCDT (EINCSRH.3), то бит Data\_Toggle каждого пакета IN будет переключаться непрерывно, независимо от сигналов квитирования, получаемых от хоста. Эту функцию обычно используют прерывистые Endpoint, функционирующие в качестве обратного канала взаимодействия для изохронных Endpoint. Если FCDT = '0', то бит Data\_Toggle будет переключаться только при отправке АСК хостом в ответ на пакет IN.



## 15.12.2. Изохронный режим Endpoint1-3 IN

Если бит ISO (EINCSRH.6) установлен в 1, то целевая Endpoint функционирует в изохронном (ISO) режиме. Если Endpoint настроена на работу в ISO IN режиме, то хост будет посылать один признак IN (запрос данных) на фрейм; положение данных внутри каждого фрейма может изменяться. В связи с этим рекомендуется разрешить двойную буферизацию для ISO IN Endpoints.

Модуль USB0 автоматически сбросит в 0 бит INPRDY (EINCSRL.0) при открытии в FIFO Endpoint области, занимаемой пакетом. Следует иметь ввиду, что если для целевой Endpoint разрешена двойная буферизация, то программа имеет возможность загрузить в буфер IN FIFO Endpoint одновременно два пакета данных. В этом случае модуль USB0 сбросит в 0 бит INPRDY сразу же после того, как программа загрузит первый пакет в буфер FIFO и установит в 1 бит INPRDY. Прерывание не будет генерироваться в этом случае; прерывание будет генерироваться только тогда, когда передан пакет данных.

Если модуль USB0 получает от хоста признак IN, а в буфере FIFO Endpoint нет готового пакета данных, то USB0 передаст пакет данных нулевой длины и установит в 1 бит UNDRUN (EINCSRL.2).

Функция обновления ISO (ISO Update) (см. раздел 15.7) может быть полезна при запуске ISO IN Endpoint с двойной буферизацией. Если хост уже настроил ISO IN канал (начал передачу признаков IN), когда программа записывает первый пакет данных в FIFO Endpoint, то следующий признак IN может поступить и первый пакет данных может быть отправлен прежде, чем программа запишет второй пакет данных в FIFO Endpoint. Функция обновления ISO гарантирует, что никакой пакет данных, записанный в FIFO Endpoint, не будет отправлен в течение текущего фрейма; пакет будет отправлен только после получения сигнала SOF.

**Рисунок 15.22. EINCSRL: Старший байт регистра управления  
IN Endpoint USB0 (USB регистр)**

R	W	R/W	R/W	W	R/W	R/W	R/W	Значение при сбросе: 00000000
-	CLRDT	STSTL	SDSTL	FLUSH	UNDRUN	FIFONE	INPRDY	USB Адрес: 0x11
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: Не используется. Читается как 0b. Запись не оказывает никакого влияния.

Бит 6: CLRDT: Сброс бита переключения данных.  
**Запись:** Программа должна установить этот бит в 1 для сброса в '0' бита переключения данных IN Endpoint.  
**Чтение:** Это бит всегда читается как '0'.

Бит 5: STSTL: Передан STALL.  
 Модуль USB0 аппаратно устанавливает этот бит в 1 после передачи сигнала STALL. FIFO очищается и бит INPRDY сбрасывается в 0. Этот флаг должен сбрасываться программно.

Бит 4: SDSTL: Передача STALL.  
 Программа должна установить этот бит в 1 для генерации события STALL в ответ на признак IN. Программа должна сбросить этот бит в 0 для завершения сигнала STALL. Этот бит игнорируется в изохронном режиме.

Бит 3: FLUSH: Очистка FIFO.  
 Установка этого бита в 1 очищает следующий пакет, который должен передаваться из IN FIFO Endpoint. Указатель FIFO сбрасывается, бит INPRDY сбрасывается в 0. Если FIFO содержит несколько пакетов, то программа должна для каждого пакета установить в 1 бит FLUSH. При завершении очистки FIFO модуль USB0 аппаратно сбрасывает в 0 бит FLUSH.

Бит 2: UNDRUN: Недогрузка данных.  
 Функционирование этого бита зависит от режима работы IN Endpoint:  
 ISO: Этот бит устанавливается в 1, когда пакет нулевой длины отправлен после получения признака IN при INPRDY = '0'.  
 Прерывистый/блочный: Этот бит устанавливается в 1, когда в ответ на признак IN возвращен NAK. Этот бит должен сбрасываться программно.

Бит 1: FIFONE: Буфер FIFO не пуст.  
 0: Буфер FIFO IN Endpoint пуст.  
 1: Буфер FIFO IN Endpoint содержит один или более пакетов.

Бит 0: INPRDY: Пакет IN готов.  
 Программа должна установить этот бит в 1 после загрузки пакета данных в буфер FIFO IN Endpoint. Модуль USB0 аппаратно сбрасывает этот бит в 0 в следующих случаях:

1. Пакет данных передан.
2. Двойная буферизация разрешена и открыта область FIFO, занимаемая пакетом.
3. Если Endpoint функционирует в изохронном режиме (ISO = '0') и ISOUP = '1', то INPRDY будет читаться как '0' до получения следующего признака SOF.

**Если аппаратный сброс бита INPRDY вызван передачей пакета, то будет сгенерировано прерывание (если оно разрешено).**

**Рисунок 15.23. EINCSRH: Младший байт регистра управления  
IN Endpoint USB0 (USB регистр)**

R/W	R/W	R/W	R	R/W	R/W	R	R	Значение при сбросе: 00000000 USB Адрес: 0x12
DBIEN	ISO	DIRSEL	-	FCDT	SPLIT	-	-	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: DBIEN: Разрешение двойной буферизации IN Endpoint.  
0: Двойная буферизация для выбранной IN Endpoint запрещена.  
1: Двойная буферизация для выбранной IN Endpoint разрешена.

Бит 6: ISO: Включение изохронного режима передачи данных.  
Этот бит включает/отключает изохронный режим передачи данных для текущей Endpoint.  
0: Endpoint функционирует в прерывистом/блочном режиме передачи данных.  
1: Endpoint функционирует в изохронном режиме передачи данных.

Бит 5: DIRSEL: Выбор направления передачи данных Endpoint.  
Этот бит является корректным только в том случае, если выбранная Endpoint не работает в отдельном режиме (SPLIT = '0').  
0: Endpoint настроена на передачу данных (OUT Endpoint).  
1: Endpoint настроена на прием данных (IN Endpoint).

Бит 4: Не используется. Читается как 0b. Запись не оказывает никакого влияния.

Бит 3: FCDT: Режим переключения бита Data\_Toggle.  
0: Бит Data\_Toggle Endpoint переключается только тогда, когда после передачи пакета данных получен ACK.  
1: Бит Data\_Toggle Endpoint переключается после передачи каждого пакета данных, независимо от получения ACK.

Бит 2: SPLIT: Включение отдельного режима FIFO.  
Если SPLIT = '0', то буфер FIFO выбранной Endpoint функционирует в отдельном режиме. IN Endpoint использует старшую половину буфера FIFO; OUT Endpoint использует младшую половину буфера FIFO.

Биты 1-0: Не используются. Читаются как 00b. Запись не оказывает никакого влияния.

### 15.13. Управление Endpoint1-3 OUT

Для управления Endpoint1-3 OUT используются USB-регистры EOUTCSRL и EOUTCSRH. Все OUT Endpoint могут использоваться в прерывистом, блочном или изохронном режимах передачи данных. Для включения изохронного режима (ISO) следует установить в 1 бит ISO в регистре EOUTCSRH. Управление блочным и прерывистым режимами передачи данных осуществляется аппаратно идентичным образом.

Прерывание от Endpoint1-3 IN генерируется при любом из следующих случаев:

1. Модуль USB0 аппаратно устанавливает в 1 бит OPRDY (EINCSRL.0).
2. Модуль USB0 генерирует событие STALL.

#### 15.13.1. Прерывистый и блочный режимы Endpoint1-3 OUT

Если бит ISO (EOUTCSRH.6) = '0', то целевая Endpoint функционирует в блочном или прерывистом режимах передачи данных. После того, как Endpoint настроена на работу в блочном/прерывистом OUT режиме (обычно после команды SET\_INTERFACE для Endpoint0), модуль USB0 будет устанавливать в 1 бит OPRDY (EOUTCSRL.0) и генерировать прерывание при получении признака OUT и пакета данных. Количество байт в текущем пакете OUT-данных (т.е. в пакете, готовом к выгрузке из FIFO) задается в регистрах EOUTCNTH и EOUTCNTL. В ответ на это прерывание программа должна выгрузить пакет данных из буфера OUT FIFO и сбросить в 0 бит OPRDY.

Блочный или прерывистый канал передачи данных можно отключить (или остановить), установив в 1 бит SDSTL (EOUTCSRL.5). Если SDSTL = 1, то модуль USB0 будет отвечать на все запросы OUT сигналом STALL. Каждый раз при генерировании события STALL будет генерироваться прерывание и устанавливаться в 1 бит STSTL (EOUTCSRL.6). Бит STSTL должен сбрасываться в 0 программно.

Модуль USB0 автоматически сбросит в 0 бит OPRDY, когда пакет будет готов к выгрузке в буфере OUT FIFO. Следует иметь в виду, что если для целевой Endpoint разрешена двойная буферизация, то одновременно в буфере OUT FIFO может быть два готовых к выгрузке пакета данных. В этом случае модуль USB0 установит в 1 бит OPRDY сразу же после того, как программа выгрузит первый пакет и сбросит в 0 бит OPRDY. В этом случае будет генерироваться второе прерывание.

#### 15.13.2. Изохронный режим Endpoint1-3 OUT

Если бит ISO (EOUTCSRH.6) установлен в 1, то целевая Endpoint функционирует в изохронном (ISO) режиме. Если Endpoint настроена на работу в ISO OUT режиме, то хост будет посылать ровно один пакет данных на фрейм; однако, положение данных внутри каждого фрейма может изменяться. В связи с этим рекомендуется разрешить двойную буферизацию для ISO OUT Endpoints.

Каждый раз при получении пакета данных модуль USB0 будет аппаратно загружать полученный пакет в FIFO Endpoint, устанавливать в 1 бит OPRDY (EOUTCSRL.0) и генерировать прерывание (если оно разрешено). Программа должна использовать это прерывание для выгрузки пакета данных из FIFO Endpoint и сброса в 0 бита OPRDY.

Если пакет данных получен, когда в FIFO Endpoint нет свободного места, то будет сгенерировано прерывание и установится в 1 бит OVRUN (EOUTCSRL.2). Если USB0 получит пакет ISO-данных с неверной контрольной суммой, то пакет данных будет загружен в буфер FIFO Endpoint, бит OPRDY будет установлен в 1, будет сгенерировано прерывание (если оно разрешено) и будет установлен в 1 бит DATERR (EOUTCSRL.3). Программа должна проверять бит DATERR каждый раз при выгрузке пакета данных из ISO OUT FIFO Endpoint.



**Рисунок 15.24. EOUCSRL: Старший байт регистра управления  
OUT Endpoint USB0 (USB регистр)**

W	R/W	R/W	W	R	R/W	R	R/W	Значение при сбросе: 00000000
CLRDT	STSTL	SDSTL	FLUSH	DATERR	OVRUN	FIFOFUL	OPRDY	USB Адрес: 0x14
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: CLRDT: Сброс бита переключения данных.  
**Запись:** Программа должна установить этот бит в 1 для сброса в '0' бита переключения данных OUT Endpoint.  
**Чтение:** Это бит всегда читается как '0'.

Бит 6: STSTL: Передан STALL.  
 Модуль USB0 аппаратно устанавливает этот бит в 1 после передачи сигнала STALL. Этот флаг должен сбрасываться программно.

Бит 5: SDSTL: Передача STALL.  
 Программа должна установить этот бит в 1 для генерации события STALL. Программа должна сбросить этот бит в 0 для завершения сигнала STALL. Этот бит игнорируется в изохронном режиме.

Бит 4: FLUSH: Очистка FIFO.  
 Установка этого бита в 1 очищает следующий пакет, который должен считываться из OUT FIFO Endpoint. Указатель FIFO сбрасывается, бит OPRDY сбрасывается в 0. Если FIFO содержит несколько пакетов, то программа должна для каждого пакета установить в 1 бит FLUSH. При завершении очистки FIFO модуль USB0 аппаратно сбрасывает в 0 бит FLUSH.

Бит 3: DATERR: Ошибка данных.  
 В изохронном режиме этот бит аппаратно устанавливается в 1, если полученный пакет имеет ошибку контрольной суммы или ошибку подстановки битов. Этот бит сбрасывается при программном сбросе бита OPRDY. Этот бит является корректным только в изохронном режиме.

Бит 2: OVRUN: Переполнение данных.  
 Этот бит аппаратно устанавливается в 1 в том случае, если входящий пакет данных не может быть загружен в OUT FIFO Endpoint. Этот бит является корректным только в изохронном режиме и должен сбрасываться программно.  
 0: Нет переполнения данных.  
 1: Пакет данных потерян вследствие переполнения буфера FIFO после последнего обнуления этого флага.

Бит 1: FIFOFUL: Заполнение буфера FIFO.  
 Этот бит показывает содержимое буфера OUT FIFO. Если двойная буферизация для Endpoint разрешена (DBIEN = '1'), то FIFO заполнен, если в нем содержится два пакета. Если DBIEN = '0', то FIFO заполнен, если он содержит один пакет.  
 0: Буфер OUT FIFO Endpoint не заполнен.  
 1: Буфер OUT FIFO Endpoint заполнен.

Бит 0: OPRDY: Пакет OUT готов.  
 Модуль USB0 аппаратно устанавливает этот бит в 1 и генерирует прерывание, если пакет данных доступен. Программа должна сбрасывать этот бит в 0 после выгрузки каждого пакета данных из буфера FIFO OUT Endpoint.

**Рисунок 15.25. EOOUTCSRH: Младший байт регистра управления OUT Endpoint USB0 (USB регистр)**

R/W	R/W	R/W	R/W	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x15
DBOEN	ISO	-	-	-	-	-	-	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: DBOEN: Разрешение двойной буферизации.  
0: Двойная буферизация для выбранной OUT Endpoint запрещена.  
1: Двойная буферизация для выбранной OUT Endpoint разрешена.

Бит 6: ISO: Включение изохронного режима передачи данных.  
Этот бит включает/отключает изохронный режим передачи данных для текущей Endpoint.  
0: Endpoint функционирует в прерывистом/блочном режиме передачи данных.  
1: Endpoint функционирует в изохронном режиме передачи данных.

Биты 5-0: Не используются. Читаются как 000000b. Запись не оказывает никакого влияния.

**Рисунок 15.26. EOOUTCNTL: Младший байт счетчика байт данных OUT Endpoint USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x16
EOCL								
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: EOCL: Младший байт счетчика байт данных OUT Endpoint.  
EOCL содержит младшие 8 бит 10-разрядного числа байт данных последнего полученного пакета в текущем буфере OUT FIFO Endpoint. Это число является корректным только в том случае, если OPRDY = '1'.

**Рисунок 15.27. EOOUTCNTH: Старший байт счетчика байт данных OUT Endpoint USB0 (USB регистр)**

R	R	R	R	R	R	R	R	Значение при сбросе: 00000000 USB Адрес: 0x17
-	-	-	-	-	-	EOCH		
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-20: Не используются. Читаются как 000000b. Запись не оказывает никакого влияния.

Биты 1-0: EОCH: Старший байт счетчика байт данных OUT Endpoint.  
EOCH содержит старшие 2 бита 10-разрядного числа байт данных последнего полученного пакета в текущем буфере OUT FIFO Endpoint. Это число является корректным только в том случае, если OPRDY = '1'.



**Таблица 15.4. Электрические параметры приемопередатчика USB**

VDD = 3.0В ... 3.6В, T = от -40°C до +85°C, если не указано иное.

ПАРАМЕТР	ОБОЗНАЧЕНИЕ	УСЛОВИЯ	Мин.	Тип.	Макс.	Ед. изм.
<b>ПЕРЕДАТЧИК</b>						
Выходное напряжение высокого уровня	$V_{OH}$		2.8			В
Выходное напряжение низкого уровня	$V_{OL}$				0.8	В
Выходная точка пересечения	$V_{CRS}$		1.3		2.0	В
Выходное сопротивление	$Z_{DRV}$	Высокий уровень на выходе Низкий уровень на выходе		38 38		Ом
Подтягивающий резистор	$R_{PU}$	Высокоскоростной режим (подтяжка D+) Низкоскоростной режим (подтяжка D-)	1.425	1.5	1.575	кОм
Время нарастания выходного сигнала	$T_R$	Низкоскоростной режим	75		300	нс
		Высокоскоростной режим	4		20	
Время спада выходного сигнала	$T_F$	Низкоскоростной режим	75		300	нс
		Высокоскоростной режим	4		20	
<b>ПРИЕМНИК</b>						
Чувствительность к входному дифференциальному сигналу	$V_{DI}$	$ (D+) - (D-) $	0.2			В
Диапазон синфазных входных сигналов	$V_{CM}$		0.8		2.5	В
Входной ток утечки	$I_L$	Подтягивающие резисторы отключены		< 1.0		мкА

Примечание: Временные диаграммы и параметры приведены в спецификации USB.

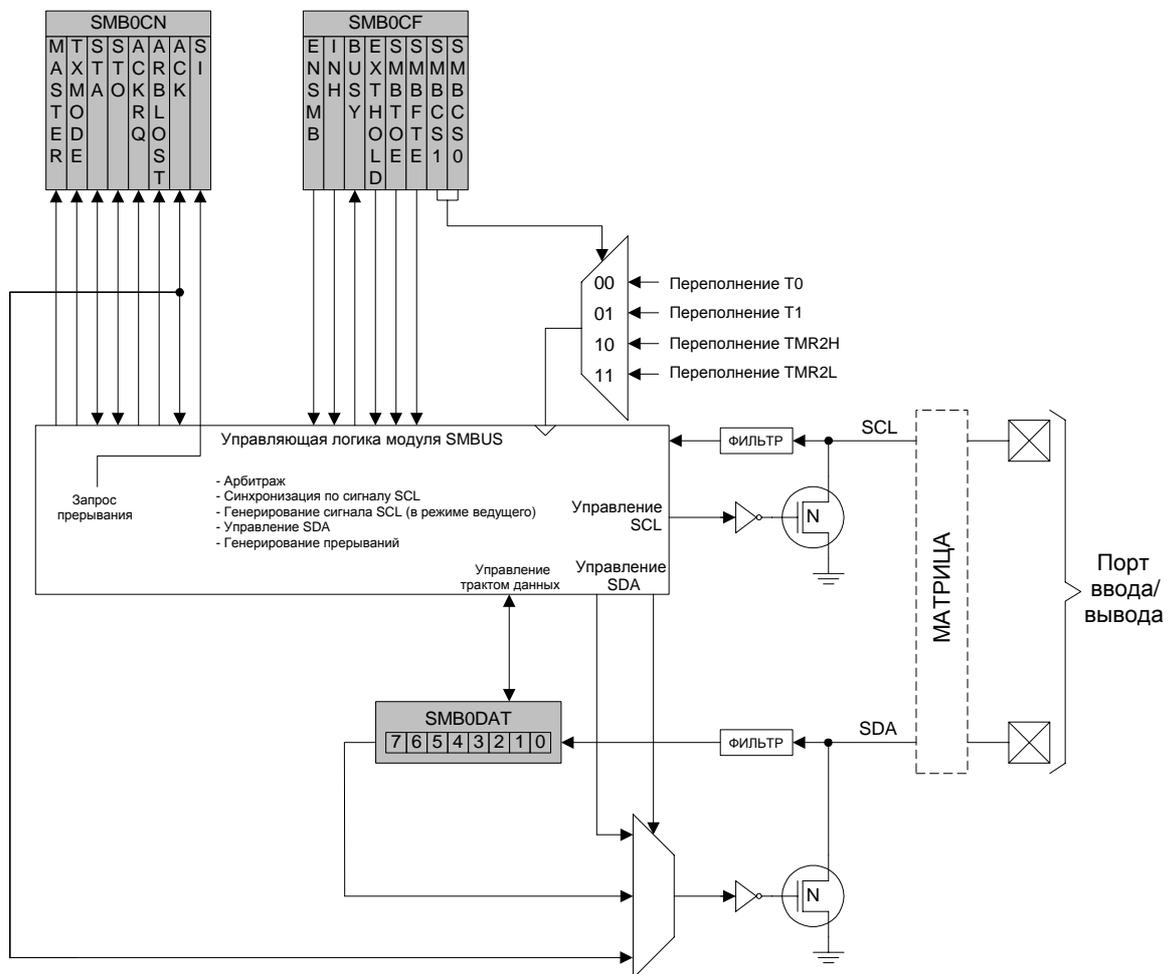
## Примечания

## 16. Модуль SMBus / I2C (SMBUS0)

Интерфейс ввода/вывода SMBus0 представляет собой двухпроводную двунаправленную последовательную шину. SMBus0 соответствует System Management Bus Specification (версия 1.1) и поддерживает обмен данными по протоколу I<sup>2</sup>C. Системный контроллер считывает данные с последовательной шины и записывает их в последовательную шину побайтно с помощью модуля SMBus, который автоматически управляет последовательной передачей данных. Данные могут передаваться со скоростью до 1/10 системной тактовой частоты как в ведущем, так и в ведомом режимах (максимально возможная скорость передачи зависит от частоты используемого тактового сигнала и может превышать допускаемую спецификацией SMBus скорость передачи данных). Для обеспечения возможности работы устройств с различной скоростью передачи данных на одной шине используется метод растягивания синхросигнала за счет удержания низкого уровня.

SMBus0 может работать в режимах ведущего и/или ведомого, а также может функционировать на шине с несколькими ведущими. SMBus0 обеспечивает управление линией SDA (последовательные данные), генерацию тактовых импульсов SCL и синхронизацию, арбитраж, управление битами START/STOP и их генерацию. Управление SMBus0 осуществляется с помощью трех SFR регистров: SMB0CF настраивает SMBus0; SMB0CN управляет состоянием SMBus0; SMB0DAT является регистром данных и используется для приема и передачи адреса ведомого и данных SMBus.

Рисунок 16.1. Структурная схема модуля SMBus0



### 16.1. Техническая документация

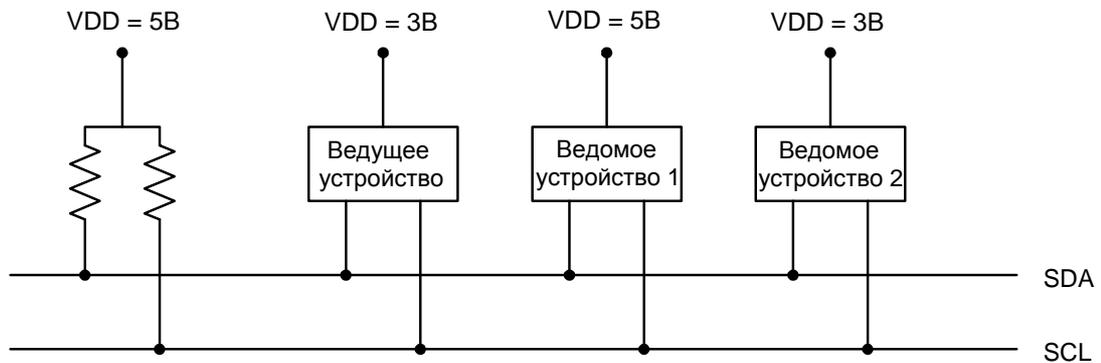
Предполагается, что читатель знаком со следующими техническими документами или имеет доступ к ним:

1. *The I<sup>2</sup>C-bus and how to use it (including specifications)*, Philips Semiconductor.
2. *The I<sup>2</sup>C-Bus Specification -- Version 2.0*, Philips Semiconductor.
3. *System Management Bus Specification -- Version 1.1*, SBS Implementers Forum.

### 16.2. Подключение к шине SMBus

На рис.16.2 приведена типичная схема подключения к шине SMBus. Интерфейс SMBus0 способен работать при любом напряжении от 3.0 до 5.0В, а различные устройства на шине могут иметь различные напряжения питания. Линии SCL (тактовые импульсы) и SDA (последовательные данные) являются двунаправленными. Необходимо подать на них положительное напряжение питания через подтягивающий резистор или подобную схему. Каждое устройство, подключенное к шине, должно иметь выход с открытым стоком или с открытым коллектором как для линии SCL, так и для линии SDA; тогда при свободной шине обе линии будут «подтянуты» к напряжению высокого логического уровня. Максимальное количество устройств на шине ограничивается только следующим требованием: время нарастания и спада сигнала на линиях шины не должно превышать 300нс и 1000нс соответственно.

Рисунок 16.2. Подключение к шине SMBus



### 16.3. Протокол SMBus

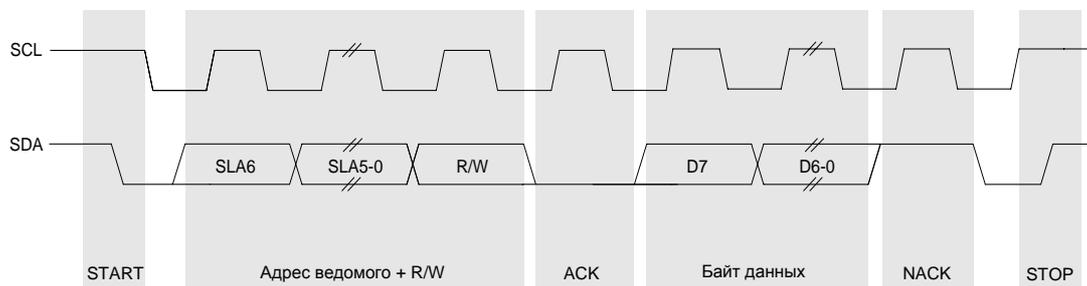
Возможны два режима передачи данных: передача данных от ведущего передатчика к адресуемому ведомому приемнику (ЗАПИСЬ) и передача данных от адресуемого ведомого передатчика к ведущему приемнику (ЧТЕНИЕ). Ведущее устройство инициирует процесс передачи данных в обоих режимах и генерирует тактовый сигнал на линии SCL. SMBus0 может функционировать в ведущем или ведомом режимах, а также поддерживает режим работы с несколькими ведущими на одной шине. Если два или более ведущих пытаются инициировать процесс передачи данных одновременно, то применяемая схема арбитража всегда определит одного ведущего, который выиграет арбитраж и захватит управление шиной. Следует иметь в виду, что нет необходимости определять какое-либо устройство как ведущее в системе; любое устройство, которое передает, бит START и адрес ведомого, становится ведущим для этого сеанса связи.

Типичное сообщение SMBus состоит из бита START, следующего за ним байта адреса (биты 7-1: 7-разрядный адрес ведомого; бит 0: бит направления передачи R/W), одного или нескольких байт данных и бита STOP. Каждый принятый (ведущим или ведомым) байт должен быть подтвержден (ACK) низким уровнем сигнала на линии SDA во время высокого уровня сигнала на линии SCL (см. рис.16.3). Если принимающее устройство не подтверждает прием, то передающее устройство воспримет этот факт как бит «нет подтверждения» (NACK), который представляет собой высокий уровень сигнала на линии SDA во время высокого уровня сигнала на линии SCL.

Бит направления занимает самый младший значащий разряд байта адреса. Бит направления устанавливается в 1 для выполнения операции чтения и сбрасывается в 0 для выполнения операции записи.

Все сеансы обмена данными инициируются ведущим, который адресует одно или несколько ведомых устройств. Ведущий генерирует бит START и затем передает адрес ведомого и бит направления. Если инициируется операция записи от ведущего к ведомому, то ведущий передает по одному байту данных за раз, ожидая бита подтверждения (ACK) от ведомого в конце каждого байта. Если осуществляется операция чтения, то ведомый передает данные, ожидая бита подтверждения (ACK) от ведущего в конце каждого байта. В конце сеанса передачи данных ведущий генерирует бит STOP, чтобы завершить сеанс и освободить шину. На рис.16.3 показан формат типичного сообщения SMBus.

**Рисунок 16.3. Формат сообщения SMBus**



#### 16.3.1. Арбитраж

Ведущий может начать сеанс передачи, только если шина свободна. Шина является свободной после бита STOP или после того, как на линиях SCL и SDA в течение определенного времени удерживается высокий уровень сигнала (см. раздел 16.3.4 «Таймаут высокого уровня на линии SCL» на стр.178). В случае, когда два или более ведущих устройства пытаются начать передачу данных одновременно, используется схема арбитража, которая заставит какое-либо ведущее устройство освободить шину. Ведущие устройства продолжают передавать до тех пор, пока один из ведущих не попытается передать на линию SDA сигнал высокого уровня, в то время как другие ведущие выдают на эту линию сигнал низкого уровня. Ведущее устройство, пытающееся передать на линию SDA сигнал высокого уровня, определит, что вместо сигнала высокого уровня на линии SDA присутствует сигнал низкого уровня, и освободит шину. Выигравший арбитраж ведущий продолжает передавать свои данные без какого-либо перерыва; потерявший арбитраж ведущий становится ведомым и принимает остаток передаваемых данных. Данная схема арбитража является неразрушающей: какое-нибудь одно устройство всегда выигрывает и никакие данные не теряются.

### 16.3.2. Растягивание тактового сигнала

Модуль SMBus обеспечивает механизм тактовой синхронизации, аналогичный I<sup>2</sup>C, который позволяет устройствам с различной скоростью передачи данных сосуществовать на одной шине. Чтобы позволить медленным ведомым устройствам обмениваться данными с быстрыми ведущими, применяется растягивание тактовых импульсов на участке с низким уровнем сигнала. Ведомый может временно удерживать линию SCL на низком уровне, тем самым удлиняя период тактового сигнала (за счет удлинения участка с низким уровнем) и уменьшая таким образом частоту тактирования шины.

### 16.3.3. Таймаут низкого уровня на линии SCL

Если линия SCL удерживается на низком уровне ведомым устройством, то дальнейший обмен данными по шине невозможен. Кроме этого, ведущий не может установить на линии SCL высокий уровень, чтобы исправить ошибочное состояние. Чтобы решить эту проблему, протокол SMBus определяет, что устройства, участвующие в обмене, должны распознавать как условие «таймаута» любой тактовый цикл, в котором сигнал удерживается на низком уровне более 25мс. Устройства, которые обнаружили такое условие таймаута, должны в течение 10мс после этого сбросить свои модули обмена.

Если бит SMBT0E в регистре SMB0CF установлен в 1, то для обнаружения таймаута низкого уровня на линии SCL используется Таймер 3. Таймер 3 будет перезагружаться, если на линии SCL высокий уровень, и будет считать, если на линии SCL низкий уровень. Если Таймер 3 включен и настроен на переполнение через 25 мс (и SMBT0E = 1), то процедуру обслуживания прерывания от Таймера 3 можно использовать для сброса (отключение и повторное включение) модуля SMBus0 в случае таймаута низкого уровня на линии SCL.

### 16.3.4. Таймаут высокого уровня на линии SCL (шина SMBus свободна)

Спецификация SMBus оговаривает, что если линии SCL и SDA удерживаются на высоком уровне более 50 мкс, то шина считается свободной. Если бит SMBFTE в регистре SMB0CF установлен в 1, то шина будет считаться свободной, если линии SCL и SDA удерживаются на высоком уровне более 10 периодов сигнала тактирования SMBus. Если устройство на шине SMBus ожидает освобождения шины, чтобы сгенерировать бит START (в ведущем режиме), то он будет сгенерирован сразу же после истечения таймаута освобождения шины. Следует иметь в виду, что для обнаружения истечения таймаута освобождения шины требуется источник тактирования, даже если реализован только ведомый режим работы.



## 16.4. Использование модуля SMBus

Модуль SMBus0 может функционировать как в ведущем, так и в ведомом режимах. Модуль SMBus0 управляет временными параметрами и сдвигом последовательных данных; протокол более высокого уровня определяется программой пользователя. Модуль SMBus0 обеспечивает следующие независимые от конкретного приложения функции:

- Побайтная последовательная передача данных.
- Выдача тактового сигнала на линию SCL (только в ведущем режиме) и синхронизация данных на линии SDA.
- Обнаружение таймаутов и ошибочных состояний шины, как определено регистром конфигурации SMB0CF.
- Генерирование и обнаружение битов START/STOP.
- Арбитраж шины.
- Генерирование прерываний.
- Информация о состоянии.

Прерывания от модуля SMBus0 генерируются для каждого переданного байта данных или адреса ведомого. При передаче это прерывание генерируется после ACK, что позволяет программе прочитать принятое значение ACK; при приеме данных это прерывание генерируется до ACK, что позволяет программе определить значение исходящего ACK. Подробная информация о работе модуля SMBus0 в различных режимах передачи данных приведена в разделе 16.5 «Режимы работы модуля SMBus» на стр.187.

Прерывания также генерируются для обозначения начала передачи данных в ведущем режиме («передан START») или для обозначения конца передачи данных в ведомом режиме («обнаружен STOP»). Программа должна прочитать регистр SMB0CN (регистр управления модулем SMBus0), чтобы определить причину прерывания. Регистр SMB0CN описан в разделе 16.4.2 «Регистр управления SMB0CN» на стр.183; таблицу 16.4 можно использовать в качестве руководства для быстрого декодирования состояний SMB0CN.

Опции настройки модуля SMBus0 включают:

- Обнаружение таймаута (таймаут низкого уровня на линии SCL и/или таймаут освобождения шины).
- Растягивание времени установления и времени удержания данных на линии SDA.
- Разрешение/запрещение ведомого режима.
- Выбор источника тактирования.

Эти опции выбираются в регистре SMB0CF, как описано в разделе 16.4.1 «Регистр конфигурации SMBus» на стр.180.

### 16.4.1. Регистр конфигурации SMBus

Регистр конфигурации модуля SMBus0 (SMB0CF) используется для разрешения ведущего и/или ведомого режимов работы, для выбора источника тактирования SMBus, а также для выбора временных параметров и значений таймаутов SMBus. Если бит ENSMB установлен в 1, то модуль SMBus0 может работать во всех ведущих и ведомых режимах. Ведомый режим можно запретить, установив в 1 бит INH. Если ведомый режим запрещен, то модуль SMBus0 все равно будет отслеживать сигналы на выводах SCL и SDA; однако модуль SMBus0 будет выдавать NACK в ответ на все принятые адреса и не будет генерировать никакие прерывания ведомого режима. Если бит INH устанавливается в 1, то ведомый режим будет запрещен после следующего START (прерывания будут генерироваться во время текущего сеанса передачи данных).

**Таблица 16.1. Выбор источника тактирования SMBus**

SMBCS1	SMBCS0	Источник тактирования SMBus
0	0	Переполнение Таймера 0
0	1	Переполнение Таймера 1
1	0	Переполнение старшего байта Таймера 2
1	1	Переполнение младшего байта Таймера 2

Биты SMBCS1-0 выбирают источник тактирования SMBus, который используется только в том случае, если модуль SMBus0 функционирует в ведущем режиме или если разрешено обнаружение таймаута освобождения шины. Если модуль SMBus0 работает в ведущем режиме, то переполнение выбранного источника тактирования определяет абсолютные минимальные длительности низкого и высокого уровней на линии SCL (см. уравнение 16.1). Следует иметь в виду, что выбранный источник тактирования может использоваться и другими периферийными модулями при условии, что таймер работает постоянно. Например, переполнения Таймера 1 могут генерировать скорость передачи данных SMBus0 и УАППО одновременно. Настройка таймеров описана в разделе 19 «Таймеры» на стр.217.

### Уравнение 16.1. Минимальная длительность высокого и низкого уровней на линии SCL

$$T_{HighMin} = T_{LowMin} = 1 / f_{ClockSourceOverflow}$$

где:

$T_{HighMin}$  – длительность высокого уровня на линии SCL

$T_{LowMin}$  – длительность низкого уровня на линии SCL

$f_{ClockSourceOverflow}$  – частота переполнения выбранного источника тактирования

Используя уравнение 16.1, следует настроить выбранный источник тактирования таким образом, чтобы установить требуемые минимальные значения высокого и низкого уровней на линии SCL. Если модуль SMBus0 работает в ведущем режиме (и никакие другие устройства на шине не изменяют сигнал SCL), то типичную скорость передачи данных по шине SMBus можно приблизительно определить из уравнения 16.2.

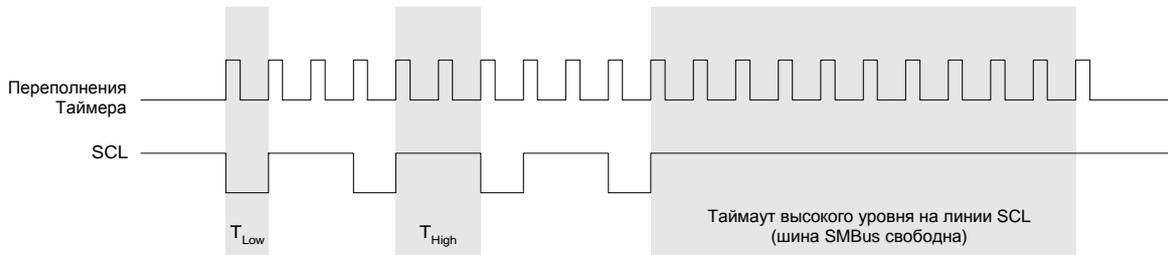
### Уравнение 16.2. Типичная скорость передачи данных по шине SMBus

$$\text{Скорость передачи данных} = f_{ClockSourceOverflow} / 3$$



На рис.16.4 показано типичное формирование сигнала SCL, описываемое уравнением 16.2. Следует обратить внимание на то, что обычно  $T_{HIGH}$  в два раза больше, чем  $T_{LOW}$ . Реальный сигнал SCL может отличаться от приведенного на рис.16.4, т.к. другие устройства на шине могут влиять на него (более медленные устройства могут удлинять низкий уровень сигнала SCL, конкурирующие ведущие устройства могут устанавливать низкий уровень на линии SCL). При работе в ведущем режиме битовая скорость никогда не превысит значения, определяемого уравнением 16.1.

**Рисунок 16.4. Формирование сигнала SCL**



Установка в 1 бит EXTHOLD удлиняет минимальное время установления и время удержания сигнала на линии SDA. Минимальное время установления сигнала на линии SDA определяет абсолютное минимальное время, в течение которого SDA стабилизируется перед переходом 0→1 сигнала SCL. Минимальное время удержания сигнала на линии SDA определяет абсолютное минимальное время, в течение которого текущее значение SDA остается стабильным после перехода 1→0 сигнала SCL. Бит EXTHOLD следует установить таким образом, чтобы минимальное время установления и время удержания соответствовали требованиям спецификации SMBus, т.е. 250 нс и 300 нс соответственно. В таблице 16.2 приведены минимальное время установления и время удержания для двух значений EXTHOLD. Удлинение времени установления и времени удержания обычно требуется, если  $SYSCLK > 10$  МГц.

**Таблица 16.2. Минимальные значения времени установления и времени удержания сигнала SDA**

EXTHOLD	Минимальное время установления SDA	Минимальное время удержания SDA
0	$T_{LOW}$ – 4 системных цикла или 1 системный цикл + задержка s/w*	3 системных цикла
1	11 системных циклов	12 системных циклов

\* Время установления для бита ACK и для старших бит всех передаваемых байт данных. Задержка s/w отсчитывается между записью регистра SMBODAT или бита ACK и сбросом бита SI. Следует иметь в виду, что если бит SI сбрасывается той же командой записи, которая определяет значение исходящего ACK, то задержка s/w равна нулю.

Если бит SMBT0E установлен в 1, то Таймер 3 должен быть настроен на переполнение через 25 мс для обнаружения таймаута низкого уровня на линии SCL (см. раздел 16.3.3 «Таймаут низкого уровня на линии SCL» на стр.178). Таймер 3 будет перезагружаться, когда  $SCL = 1$ , и будет считаться, когда  $SCL = 0$ . Процедуру обслуживания прерывания от Таймера 3 следует использовать для сброса модуля SMBus0 путем его отключения и повторного включения.

Обнаружение таймаута освобождения шины SMBus можно разрешить, установив в 1 бит SMBFTE. Если SMBFTE = 1, то шина будет считаться свободной, если линии SCL и SDA удерживаются на высоком уровне более 10 периодов сигнала тактирования SMBus (см. рис.16.4). Если таймаут освобождения шины обнаружен, то модуль SMBus будет реагировать таким же образом, как будто обнаружен бит STOP (будет сгенерировано прерывание и будет установлен в 1 бит STO).

Рисунок 16.5. SMB0CF: Регистр конфигурации модуля SMBus0

R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
ENSMB	INH	BUSY	EXTHOLD	SMBTOE	SMBFTE	SMBCS1	SMBCS0	SFR Адрес: 0xC1
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

**Бит 7: ENSMB:** Включение модуля SMBus0.  
Этот бит включает/отключает модуль SMBus0. Во включенном состоянии модуль SMBus0 постоянно отслеживает сигналы на выводах SDA и SCL.  
0: Модуль SMBus0 отключен.  
1: Модуль SMBus0 включен.

**Бит 6: INH:** Запрет ведомого режима модуля SMBus0.  
Если этот бит установлен в 1, то модуль SMBus0 не генерирует прерываний ведомого режима. Это позволяет эффективно удалить ведомый модуль SMBus0 с шины. Это бит не влияет на прерывания ведущего режима.  
0: Ведомый режим модуля SMBus0 разрешен.  
1: Ведомый режим модуля SMBus0 запрещен.

**Бит 5: BUSY:** Флаг занятости шины SMBus.  
Этот бит аппаратно устанавливается в 1, если идет передача данных. Этот бит сбрасывается в 0, если обнаружен бит STOP или таймаут освобождения шины.

**Бит 4: EXTHOLD:** Разрешение удлинения времени установления и времени удержания сигнала SDA.  
Этот бит управляет удлинением времени установления и времени удержания сигнала SDA.  
0: Удлиненное время установления и время удержания сигнала SDA запрещены.  
1: Удлиненное время установления и время удержания сигнала SDA разрешены.

**Бит 3: SMBTOE:** Бит разрешения таймаута низкого уровня на линии SCL.  
Этот бит разрешает обнаружение таймаута низкого уровня на линии SCL. Если этот бит установлен в 1, то Таймер 3 будет перезагружаться, когда SCL = 1, и будет считать, когда SCL = 0. Таймер 3 следует настроить таким образом, чтобы он генерировал прерывание через 25 мс. Процедура обслуживания прерывания от Таймера 3 должна сбрасывать модуль SMBus0.

**Бит 2: SMBFTE:** Бит разрешения обнаружения таймаута освобождения шины SMBus  
Если этот бит установлен в 1, то шина будет считаться свободной, если линии SCL и SDA удерживаются на высоком уровне более 10 периодов сигнала тактирования SMBus.

**Биты 1-0: SMBCS1-SMBCS0:** Выбор источника тактирования SMBus.  
Эти два бита выбирают источник тактирования SMBus, который определяет скорость передачи данных SMBus. Выбранный источник тактирования должен быть настроен в соответствии с уравнением 16.1.

SMBCS1	SMBCS0	Источник тактирования SMBus
0	0	Переполнение Таймера 0
0	1	Переполнение Таймера 1
1	0	Переполнение старшего байта Таймера 2
1	1	Переполнение младшего байта Таймера 2

## 16.4.2. Регистр управления SMB0CN

Регистр управления SMB0CN используется для управления модулем SMBus0 и для предоставления информации о его состоянии (см. рис.16.6). Старшие четыре бита регистра SMB0CN (MASTER, TXMODE, STA и STO) образуют вектор состояния, который может использоваться для перехода на процедуру обслуживания прерывания. Биты MASTER и TXMODE показывают состояние (ведущий/ведомый) и режим (передача/прием) соответственно.

Биты STA и STO показывают, что с момента последнего прерывания от SMBus0 обнаружен или сгенерирован бит START и/или STOP. Кроме этого, при работе в ведущем режиме биты STA и STO используются для генерации битов START и STOP. Установка бита STA в 1 заставит модуль SMBus0 перейти в ведущий режим и сгенерировать бит START, когда шина освободится (бит STA не сбрасывается аппаратно после генерации бита START). Установка бита STO в 1 в ведущем режиме заставит модуль SMBus0 сгенерировать бит STOP и завершить текущий сеанс передачи данных после следующего цикла ACK. Если оба бита STA и STO установлены в 1 (в ведущем режиме), то будут сгенерированы сначала бит STOP, а затем бит START.

В режиме приемника запись в бит ACK определяет значение исходящего ACK; в режиме передатчика бит ACK показывает значение, полученное в последнем цикле ACK. Бит ACKRQ устанавливается в 1 каждый раз при получении байта и показывает, что требуется значение исходящего ACK. Если ACKRQ = 1, то программа должна записать требуемое исходящее значение в бит ACK до сброса бита SI. NACK будет сгенерирован в том случае, если программа не запишет бит ACK до сброса бита SI. Сигнал SDA отразит заданное значение ACK сразу же после записи бита ACK; однако SCL будет оставаться на низком уровне до сброса бита SI. Если полученный адрес ведомого не подтверждается, то дальнейшие обращения к ведомому со стороны шины будут игнорироваться до обнаружения следующего бита START.

Бит ARBLOST показывает, что модуль SMBus0 потерял арбитраж. Это может произойти в любое время, когда модуль SMBus0 (ведущий или ведомый) передает данные. Потеря арбитража при работе в режиме ведомого означает ошибку состояния шины. Бит ARBLOST сбрасывается аппаратно каждый раз при сбросе бита SI.

Бит SI (флаг прерывания от модуля SMBus0) устанавливается в 1 в начале и в конце каждой передачи, после приема каждого байта, или при потере арбитража; подробная информация приведена в таблице 16.3.

**Важное примечание относительно бита SI:** Модуль SMBus0 останавливается при SI = 1; поэтому SCL удерживается на низком уровне и обмен по шине останавливается до программного сброса бита SI.

В таблице 16.3 перечислены все источники, вызывающие аппаратное изменение бит регистра SMB0CN. Таблица 16.4 позволяет декодировать состояние модуля SMBus0, используя регистр SMB0CN.

Рисунок 16.6. SMB0CN: Регистр управления модуля SMBus0

R	R	R/W	R/W	R	R	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xC0 <small>(доступен в битовом режиме адресации)</small>
MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p><b>Бит 7: MASTER:</b> Флаг состояния ведущий/ведомый модуля SMBus0. Этот доступный только для чтения бит показывает, работает ли модуль SMBus0 в ведущем режиме. 0: Модуль SMBus0 работает в ведомом режиме. 1: Модуль SMBus0 работает в ведущем режиме.</p> <p><b>Бит 6: TXMODE:</b> Флаг режима передачи модуля SMBus0. Этот доступный только для чтения бит показывает, работает ли модуль SMBus0 в режиме передатчика. 0: Модуль SMBus0 работает в режиме приемника 1: Модуль SMBus0 работает в режиме передатчика.</p> <p><b>Бит 5: STA:</b> Флаг запуска модуля SMBus0. Запись: 0: START не генерируется. 1: При работе в режиме ведущего передается бит START, если шина свободна (если шина не свободна, то бит START передается после приема бита STOP или после обнаружения таймаута). Если бит STA устанавливается в тот момент, когда модуль SMBus0 передает/принимает данные (т.е. активен) в ведущем режиме, то после следующего цикла ACK будет сгенерирован бит «повторный START». Чтение: 0: Биты START или «повторный START» не обнаружены. 1: Обнаружен бит START или «повторный START».</p> <p><b>Бит 4: STO:</b> Флаг окончания передачи модуля SMBus0. Запись: 0: Бит STOP не передается. 1: Установка в 1 бита STO приведет к передаче бита STOP после следующего цикла ACK. Когда генерируется бит STOP, бит STO будет аппаратно сброшен в 0. Если оба бита STA и STO установлены в 1, то будут сгенерированы сначала бит STOP, а затем бит START. Чтение: 0: Бит STOP не обнаружен. 1: Обнаружен бит STOP (в ведомом режиме) или задержка (в ведущем режиме).</p> <p><b>Бит 3: ACKRQ:</b> Запрос подтверждения. Этот доступный только для чтения бит устанавливается в 1, если модуль SMBus0 уже принял байт и должен записать в бит ACK правильное ответное значение ACK.</p> <p><b>Бит 2: ARBLOST:</b> Флаг потери арбитража. Этот доступный только для чтения бит устанавливается в 1, если модуль SMBus0, работая в режиме передатчика, теряет арбитраж. Потеря арбитража при работе в ведомом режиме означает ошибку состояния шины.</p> <p><b>Бит 1: ACK:</b> Флаг подтверждения. Этот бит определяет уровень исходящего ACK и отражает уровень входящего ACK. Этот бит должен записываться каждый раз при получении байта (когда ACKRQ = 1), или считываться после передачи каждого байта. 0: NACK, т.е. «нет подтверждения», получен (в режиме передатчика) ИЛИ будет передан (в режиме приемника). 1: ACK, т.е. «подтверждение», получен (в режиме передатчика) ИЛИ будет передан (в режиме приемника).</p> <p><b>Бит 0: SI:</b> Флаг прерывания от модуля SMBus0. Этот бит устанавливается аппаратно в 1 при возникновении условий, перечисленных в таблице 16.3. Бит SI должен сбрасываться программно. Если SI = 1, то SCL удерживается на низком уровне и обмен по шине останавливается.</p>								



**Таблица 16.3. События, вызывающие аппаратную установку/сброс битов регистра SMB0CN**

Бит	Устанавливается аппаратно, если:	Сбрасывается аппаратно, если:
MASTER	- Передан бит START.	- Передан бит STOP. - Потерян арбитраж.
TXMODE	- Передан бит START. - SMB0DAT записан до начала фрейма SMBus.	- Обнаружен бит START. - Потерян арбитраж. - SMB0DAT не записан до начала фрейма SMBus.
STA	- Получен бит START и следующий за ним байт адреса.	- Должен быть сброшен программно.
STO	- Обнаружен бит STOP во время адресации ведомого. - Потерян арбитраж вследствие обнаружения бита STOP.	- Передан отложенный бит STOP.
ACKRQ	- Байт уже получен и требуется ответное значение ACK.	- После каждого цикла ACK.
ARBLOST	- В ведущем режиме обнаружен бит «повторный START» при STA = 0 (неожиданный повторный START). - На линии SCL обнаружен низкий уровень при попытке генерировать бит STOP или бит «повторный START». - На линии SDA обнаружен низкий уровень при передаче '1' (кроме битов ACK).	- Каждый раз при сбросе бита SI.
ACK	- Значение входящего ACK равно '0' («подтверждение»).	- Значение входящего ACK равно '1' («нет подтверждение»).
SI	- Бит START уже передан. - Потерян арбитраж. - Байт уже передан и ACK/NACK получен. - Байт уже получен. - Получен бит START или «повторный START» и следующие за ними адрес ведомого + бит R/W. - Бит STOP уже получен.	- Должен быть сброшен программно.

### 16.4.3. Регистр данных

Регистр данных модуля SMBus0 SMB0DAT содержит байт последовательных данных, который необходимо передать, или байт последовательных данных, который только что принят. Программа может прочитать из регистра или записать в регистр данные, когда флаг SI установлен в 1; программа не должна пытаться обратиться к регистру SMB0DAT, когда модуль SMBus включен и флаг SI сброшен в 0, т.к. в этот момент может осуществляться аппаратный сдвиг байта данных в регистр или из регистра.

Данные всегда сдвигаются старшими разрядами вперед. После приема байта первый бит принятых данных занимает старший разряд регистра SMB0DAT. Когда данные выдвигаются из регистра на шину, данные с шины одновременно вдвигаются в регистр. Поэтому регистр SMB0DAT всегда содержит последний байт данных, присутствующий в настоящий момент на шине. Таким образом, в случае потери арбитража переход от ведущего передатчика к ведомому приемнику осуществляется с корректными данными в регистре SMB0DAT.

**Рисунок 16.7. SMB0DAT: Регистр данных модуля SMBus0**

R/W	Значение при сбросе: 00000000							
								SFR Адрес: 0xC2
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: SMB0DAT: Данные модуля SMBus.

Регистр SMB0DAT содержит байт данных, которые должны передаваться последовательному интерфейсу SMBus, или данные, только что принятые от последовательного интерфейса SMBus. Читать из этого регистра или записывать в этот регистр можно всегда, когда флаг прерывания от последовательного порта SI (SMB0CN.0) установлен в 1. Последовательные данные в регистре остаются стабильными, пока SI = 1. Когда флаг SI не установлен в 1, система может находиться в процессе сдвига данных в регистр (или из регистра) SMB0DAT и обращаться к этому регистру нельзя.



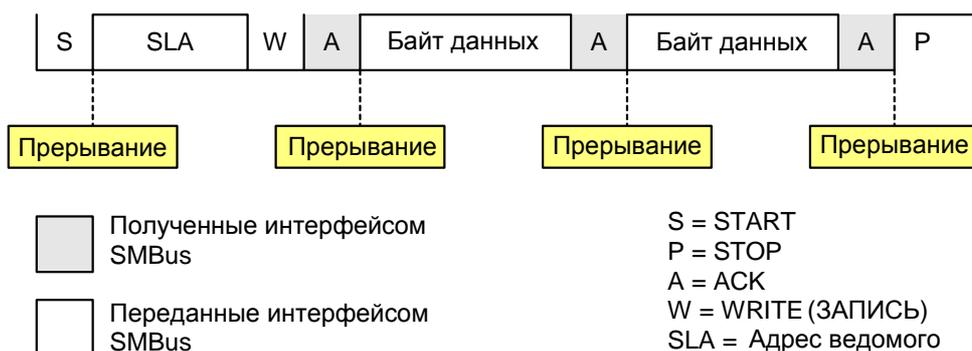
## 16.5. Режимы работы модуля SMBus

Модуль SMBus может быть настроен для работы как в режиме ведущего, так и в режиме ведомого. В любой конкретный момент времени он может работать в одном из четырех режимов: ведущий передатчик, ведущий приемник, ведомый передатчик, ведомый приемник. Модуль SMBus переходит в ведущий режим каждый раз, когда генерирует бит START, и остается в ведущем режиме до потери арбитража или до генерации бита STOP. Прерывание от модуля SMBus генерируется в конце всех сообщений SMBus; однако следует иметь в виду, что прерывание генерируется до ACK при работе в режиме приемника, и после ACK при работе в режиме передатчика.

### 16.5.1. Режим ведущего передатчика

Последовательные данные выдаются на линию SDA, а тактовые импульсы выдаются на линию SCL. SMBus0 генерирует бит START и затем передает первый байт, который содержит адрес целевого ведомого устройства и бит направления. В этом случае бит направления (R/W) будет сброшен в 0, иницируя операцию записи. Затем модуль SMBus0 передает один или несколько байт последовательных данных. После передачи каждого байта ведомый генерирует бит подтверждения. Передача заканчивается, когда бит STO устанавливается в 1 и генерируется бит STOP. Следует иметь в виду, что SMBus0 переключится в режим ведущего приемника, если после прерывания от ведущего передатчика регистр SMBODAT не был записан. На рис.16.8 показана передача данных в режиме ведущего. Показана передача двух байт данных, хотя может быть передано любое количество байт. Следует обратить внимание, что в этом режиме прерывание «байт данных передан» генерируется **после** ACK.

Рисунок 16.8. Передача данных в режиме ведущего



### 16.5.2. Режим ведущего приемника

Последовательные данные принимаются с линии SDA, а тактовые импульсы выдаются на линию SCL. Модуль SMBus0 генерирует бит START и затем передает первый байт, который содержит адрес целевого ведомого устройства и бит направления. В этом случае бит направления (R/W) будет установлен в 1, инициируя операцию чтения. Модуль SMBus0 принимает последовательные данные от ведомого по линии SDA, при этом генерирует тактовые импульсы на линии SCL. Ведомый передает один или несколько байт последовательных данных. После приема каждого байта устанавливается в 1 бит ACKRQ и генерируется прерывание. Программа должна записать бит ACK (SMB0CN.1), чтобы определить значение (тип) генерируемого подтверждения (Примечание: при записи ACK = 1 генерируется ACK, при записи ACK = 0 генерируется NACK). Программа должна сбросить бит ACK в 0 после приема последнего байта, чтобы передать (сгенерировать на шине) NACK. Модуль SMBus0 выйдет из режима ведущего приемника после установки в 1 бита STO и генерации бита STOP. Следует иметь в виду, что модуль SMBus0 перейдет в режим ведущего передатчика, если в режиме ведущего приемника записывается регистр SMB0DAT. На рис.16.9 показан прием данных в режиме ведущего. Показан прием двух байт данных, хотя может быть принято любое количество байт. Следует обратить внимание, что в этом режиме прерывание «байт данных передан» генерируется до ACK.

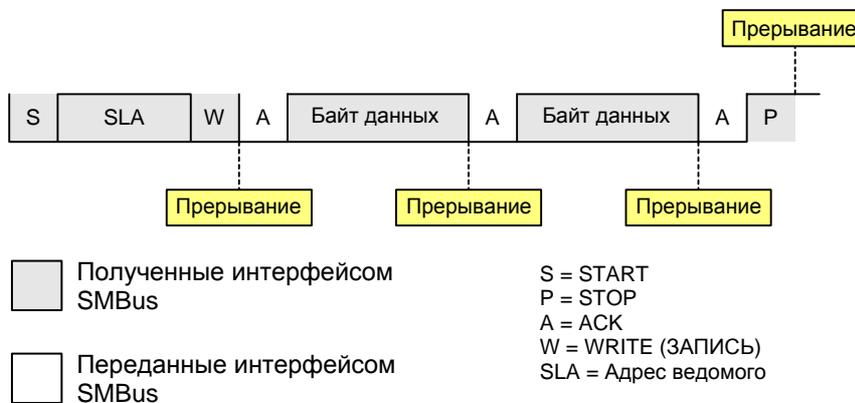
Рисунок 16.9. Прием данных в режиме ведущего



### 16.5.3. Режим ведомого приемника.

Последовательные данные принимаются с линии SDA, а тактовые импульсы принимаются с линии SCL. Если ведомый режим разрешен (INH = 0), то модуль SMBus0 перейдет в режим ведомого приемника при получении бита START и следующих за ним адреса ведомого и бита направления (в этом случае ЗАПИСЬ). При входе в режим ведомого приемника генерируется прерывание и устанавливается в 1 бит ACKRQ. Программа отвечает на принятый адрес ведомого битом ACK или игнорирует принятый адрес ведомого битом NACK. Если адрес ведомого игнорируется, то прерывания от ведомого будут запрещены до обнаружения следующего бита START. Если принятый адрес ведомого подтвержден (битом ACK), то принимаются ноль или более байт данных. Программа должна записывать бит ACK после приема каждого байта, чтобы подтвердить (ACK) или не подтвердить (NACK) принятый байт. Модуль SMBus0 выйдет из режима ведомого приемника после приема бита STOP. Следует иметь в виду, что модуль SMBus0 перейдет в режим ведомого передатчика, если в режиме ведомого приемника записывается регистр SMB0DAT. На рис.16.10 показан прием данных в режиме ведомого. Показан прием двух байт данных, хотя может быть принято любое количество байт. Следует обратить внимание, что в этом режиме прерывание «байт данных передан» генерируется до ACK.

**Рисунок 16.10. Прием данных в режиме ведомого**



### 16.5.4. Режим ведомого передатчика

Последовательные данные выдаются на линию SDA, а тактовые импульсы принимаются с линии SCL. Если ведомый режим разрешен ( $INH = 0$ ), то модуль SMBus0 перейдет в режим ведомого приемника (для приема адреса ведомого) при получении бита START и следующих за ним адреса ведомого и бита направления (в этом случае ЧТЕНИЕ). При входе в режим ведомого передатчика генерируется прерывание и устанавливается в 1 бит ACKRQ. Программа отвечает на принятый адрес ведомого битом ACK или игнорирует принятый адрес ведомого битом NACK. Если адрес ведомого игнорируется, то прерывания от ведомого будут запрещены до обнаружения следующего бита START. Если принятый адрес ведомого подтвержден (битом ACK), то передаваемые данные следует записать в регистр SMBODAT. Модуль SMBus0 перейдет в режим ведомого передатчика и передаст один или более байт данных. После передачи каждого байта ведущий посылает бит подтверждения; если битом подтверждения является ACK, то в регистр SMBODAT следует записать следующий байт данных. Если битом подтверждения является NACK, то регистр SMBODAT нельзя записывать до сброса бита SI (Примечание: Если в режиме ведомого передатчика регистр SMBODAT записывается после получения NACK, то может быть сгенерировано условие ошибки). Модуль SMBus0 выйдет из режима ведомого передатчика после приема бита STOP. Следует иметь в виду, что модуль SMBus0 перейдет в режим ведомого приемника, если после прерывания от ведомого передатчика регистр SMBODAT не записывается. На рис.16.11 показана передача данных в режиме ведомого. Показана передача двух байт данных, хотя может быть передано любое количество байт. Следует обратить внимание, что в этом режиме прерывание «байт данных передан» генерируется **после** ACK.

Рисунок 16.11. Передача данных в режиме ведомого



### 16.6. Декодирование состояний SMBus

Текущее состояние SMBus0 можно легко определить с помощью регистра SMB0CN. В приведенной ниже таблице ВЕКТОР СОСТОЯНИЯ определяется четырьмя старшими битами регистра SMB0CN: MASTER, TXMODE, STA и STO. Следует иметь в виду, что приведенные варианты ответных реакций являются только типичными ответными реакциями; специфичные для конкретного приложения процедуры также допустимы с учетом соответствия их спецификации SMBus. Выделенные цветом ответные реакции разрешены, но они не соответствуют спецификации SMBus.

Таблица 16.4. Декодирование состояний SMBus

РЕЖИМ	СЧИТЫВАЕМЫЕ ЗНАЧЕНИЯ				ТЕКУЩЕЕ СОСТОЯНИЕ SMBUS	ТИПИЧНЫЕ ОТВЕТНЫЕ РЕАКЦИИ	ЗАПИСЫВАЕМЫЕ ЗНАЧЕНИЯ		
	ВЕКТОР СОСТОЯНИЯ	ACKRQ	ARBLOST	ACK			STA	STO	ACK
Ведущий передатчик	1110	0	0	X	Передан бит START.	Загрузка SMB0DAT адресом ведомого + R/W.	0	0	X
	1100	0	0	0	Передан байт данных или адреса. Получен NACK.	Установка STA для перезапуска передатчика.	1	0	X
						Прекращение передачи.	0	1	X
		0	0	1	Передан байт данных или адреса. Получен ACK.	Загрузка следующего байта данных в SMB0DAT.	0	0	X
						Окончание передачи с генерацией бита STOP.	0	1	X
						Окончание передачи с генерацией бита STOP и начало другой передачи.	1	1	X
						Отправка бита «повторный START».	1	0	X
Переход в режим ведущего приемника (сброс SI без записи новых данных в SMB0DAT).	0	0	X						
Ведущий приемник	1000	1	0	X	Получен байт данных. Запрошен ACK.	Подтверждение (отправка ACK) полученного байта. Чтение SMB0DAT.	0	0	1
						Отправка NACK для обозначения последнего байта и выдача STOP.	0	1	0
						Отправка NACK для обозначения последнего байта, выдача STOP и затем START.	1	1	0
						Отправка ACK, затем выдача бита «повторный START».	1	0	1
						Отправка NACK для обозначения последнего байта, затем выдача бита «повторный START».	1	0	0
						Отправка ACK и переход в режим ведущего передатчика (запись в SMB0DAT перед сбросом SI).	0	0	1
						Отправка NACK и переход в режим ведущего передатчика (запись в SMB0DAT перед сбросом SI).	0	0	0

Таблица 16.4. Декодирование состояний SMBus (продолжение)

РЕЖИМ	СЧИТЫВАЕМЫЕ ЗНАЧЕНИЯ			ТЕКУЩЕЕ СОСТОЯНИЕ SMBUS	ТИПИЧНЫЕ ОТВЕТНЫЕ РЕАКЦИИ	ЗАПИСЫВАЕМЫЕ ЗНАЧЕНИЯ			
	ВЕКТОР СОСТОЯНИЯ	ACKRQ	ARBLOST			ACK	STA	STO	ACK
Ведомый передатчик	0100	0	0	0	Передан байт данных. Получен NACK.	Никаких действий не требуется (ожидается бит STOP).	0	0	X
		0	0	1	Передан байт данных. Получен ACK.	Загрузка следующего байта данных в SMB0DAT для передачи.	0	0	X
		0	1	X	Передан байт данных. Обнаружена ошибка.	Никаких действий не требуется (ожидается завершение передачи по инициативе ведущего).	0	0	X
	0101	0	X	X	Во время адресации ведомого передатчика обнаружен бит STOP.	Никаких действий не требуется (передача завершена).	0	0	X
Ведомый приемник	0010	1	0	X	Получен адрес ведомого. Запрошен ACK.	Подтверждение (отправка ACK) полученного адреса. Неподтверждение (отправка NACK) полученного адреса.	0	0	1
		1	1	X	Потерян арбитраж. Получен адрес ведомого. Запрошен ACK.	Подтверждение (отправка ACK) полученного адреса.	0	0	1
	Неподтверждение (отправка NACK) полученного адреса.					0	0	0	
	Перепланирование ошибочной передачи. Неподтверждение (отправка NACK) полученного адреса.					1	0	0	
	0010	0	1	X	Потерян арбитраж при попытке выдачи бита «повторный START».	Прекращение ошибочной передачи.	0	0	X
						Перепланирование ошибочной передачи.	1	0	X
	0010	1	1	X	Потерян арбитраж при попытке выдачи бита STOP.	Никаких действий не требуется (передача завершена/прекращена)	0	0	0
		0	0	X	Во время адресации ведомого приемника обнаружен бит STOP.	Никаких действий не требуется (передача завершена)	0	0	X
		0	1	X	Потерян арбитраж вследствие обнаружения бита STOP.	Передача прервана. Перепланирование ошибочной передачи.	0	0	X
	0000	1	0	X	Получен байт данных. Запрошен ACK.	Подтверждение (отправка ACK) полученного байта. Чтение SMB0DAT.	0	0	1
						Неподтверждение (отправка NACK) полученного байта.	0	0	0
		1	1	X	Потеря арбитраж во время передачи байта данных в ведущем режиме.	Прекращение ошибочной передачи. Перепланирование ошибочной передачи.	0	0	0
						Перепланирование ошибочной передачи.	1	0	0



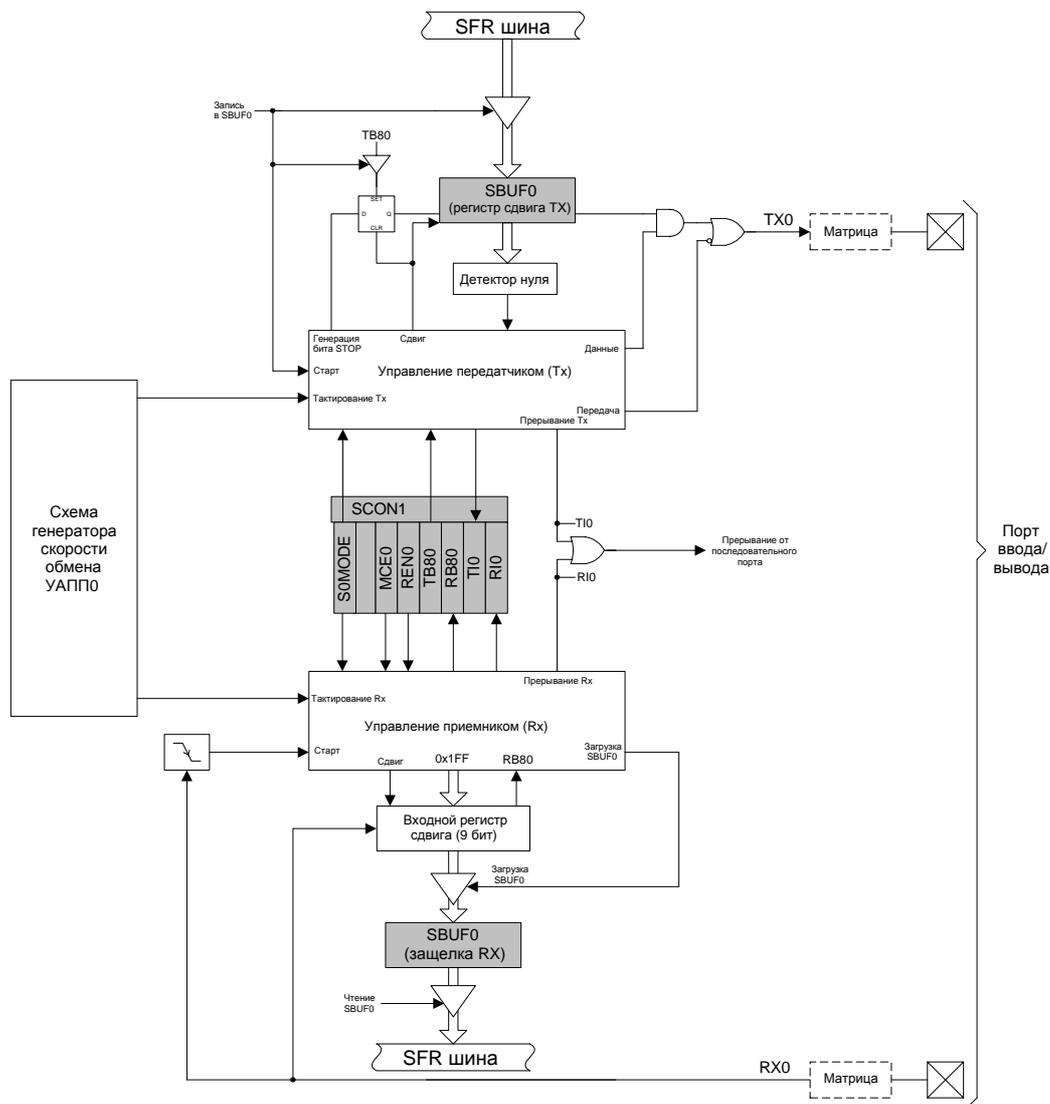
## 17. УАППО

УАППО представляет собой асинхронный полнодуплексный последовательный порт, способный работать в режимах 1 и 3 стандартного (для архитектуры 8051) УАПП. Поддержка усовершенствованного режима генерации скорости передачи данных позволяет использовать для генерации стандартных скоростей обмена различные источники тактирования (см. раздел 17.1 «Усовершенствованный режим генерации скорости передачи данных» на стр.194). Буферизация принимаемых данных позволяет УАППО начать прием второго входящего байта данных до того, как программа закончит чтение предыдущего байта данных.

С работой УАППО связаны следующие регистры специального назначения: регистр управления УАППО (SCON0) и буфер данных УАППО (SBUF0). Одна и та же ячейка памяти, адресуемая как SBUF0, обеспечивает доступ и к регистру передатчика, и к регистру приемника. **Операции записи в SBUF0 всегда обращаются к регистру передатчика. Операции чтения из SBUF0 всегда обращаются к буферизованному регистру приемника; невозможно прочесть данные из регистра передатчика.**

Если прерывания от модуля УАППО разрешены, то запрос прерывания генерируется при завершении передачи байта данных (установка в 1 флага TI0 в регистре SCON0) или при получении байта данных (установка в 1 флага RI0 в регистре SCON0). Флаги прерываний от УАППО не сбрасываются аппаратно при переходе к процедуре обслуживания прерывания. Они должны сбрасываться программно. Это позволяет программе определить причину, вызвавшую прерывание от УАППО (завершение передачи или завершение приема).

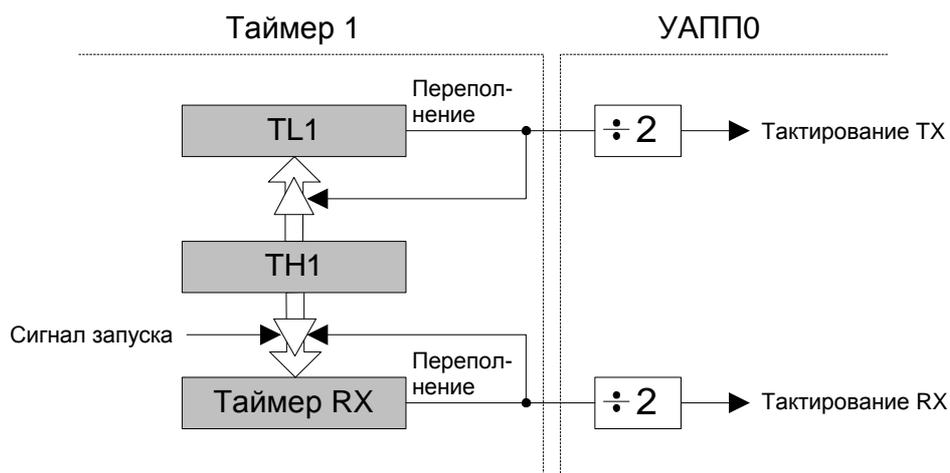
**Рисунок 17.1. Структурная схема УАППО**



### 17.1. Усовершенствованный режим генерации скорости передачи данных.

Скорость передачи данных УАППО генерируется Таймером 1, работающим в 8-разрядном режиме с автоперезагрузкой. Частота передатчика (TX) определяется переполнением регистра TL1; частота приемника определяется переполнением регистра-копии регистра TL1 (обозначенного как «RX-Таймер» на рис.17.2), который недоступен из программы пользователя. Скорость передачи данных передатчика и приемника равна деленной на два частоте переполнения регистров TL1 и RX-Таймер соответственно. RX-Таймер работает тогда, когда включен Таймер 1 и использует то же самое значение перезагрузки (TH1). Однако перезагрузка регистра RX-Таймер происходит в тот момент, когда на выводе RX обнаруживается событие START. Это позволяет начать прием данных в любой момент при обнаружении события START, независимо от состояния Таймера TX.

**Рисунок 17.2. Логика генератора скорости передачи данных УАППО**



Таймер 1 следует настроить для работы в режиме 2, т.е. как 8-разрядный таймер с автоперезагрузкой (см. раздел 19.1.3 «Режим 2: 8-разрядный таймер/счетчик с автоперезагрузкой» на стр.219). Значение перезагрузки Таймера 1 следует установить таким образом, чтобы частота переполнений таймера была в два раза больше необходимой скорости передачи данных. Частота тактового сигнала Таймера 1 может быть одной из следующих:

- 1) SYSCLK;
- 2) SYSCLK/4;
- 3) SYSCLK/12;
- 4) SYSCLK/48;
- 5) Частота внешнего генератора / 8;
- 6) Частота сигнала на внешнем входе T1.

Для любой из этих частот скорость передачи данных УАППО определяется из уравнения 17.1.

#### Уравнение 17.1. Скорость передачи данных УАППО

$$\text{Скорость передачи данных УАППО} = T1_{CLK} / ((256 - T1H) \times 2),$$

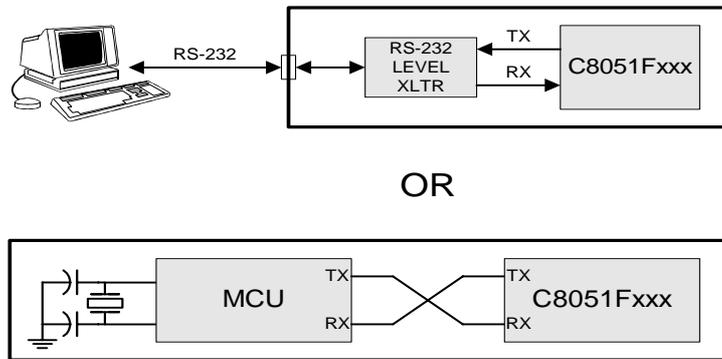
где  $T1_{CLK}$  – частота тактирования Таймера 1;  
 $T1H$  – старший байт Таймера 1 (значение перезагрузки).

Выбор частоты тактирования Таймера 1 описан в разделе 19 «Таймеры» на стр.217. В таблицах 17.1 – 17.6 приведены системные параметры для стандартных скоростей обмена при различных частотах системного тактового сигнала. Следует отметить, что внутренний генератор может генерировать системный тактовый сигнал, в то время как сигнал от внешнего генератора подается на Таймер 1.

## 17.2. Режимы работы УАППО

УАППО обеспечивает стандартный асинхронный полнодуплексный обмен данными. Режим работы УАППО (8-разрядный или 9-разрядный) выбирается при помощи бита S0MODE (SCON0.7). Типичные варианты использования УАППО приведены на рисунке ниже.

**Рисунок 17.3. Примеры использования УАППО**



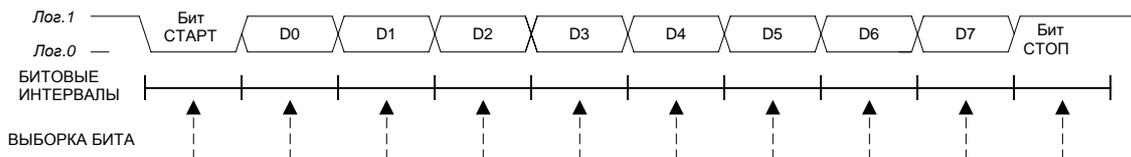
### 17.2.1. 8-разрядный УАПП.

В режиме 8-разрядного УАПП для передачи одного байта данных используются 10 бит: один стартовый бит, восемь бит данных (МЗР вперед) и один стоповый бит. Данные передаются МЗР вперед через внешний вывод TX0 и принимаются через внешний вывод RX0. При приеме в регистре SBUF0 сохраняются восемь бит данных, а бит RB80 (SCON0.2) принимает значение стопового бита.

Передача данных начинается, когда происходит запись байта данных в регистр SBUF0. Флаг прерывания от передатчика TI0 (SCON0.1) устанавливается в 1 в конце передачи (в начале передачи стопового бита). Прием данных может быть начат в любое время после установки в 1 флага включения приемника RENO (SCON0.4). После приема стопового бита байт данных будет загружен в регистр приемника SBUF0, если соблюдаются следующие условия: RI0 должен быть равен лог.0, и, если MCE0 = 1, то стоповый бит должен быть равен лог.1. В случае переполнения данных при приеме первые принятые 8 бит «защелкиваются» в регистре приемника SBUF0, а следующие данные, вызвавшие переполнение, теряются.

Если эти условия соблюдаются, то восемь бит данных сохраняются в регистре SBUF0, стоповый бит сохраняется в бите RB80 и устанавливается в 1 флаг RI0. Если эти условия не соблюдаются, то SBUF0 и RB80 не будут загружаться и флаг RI0 не устанавливается. При установке флагов TI0 или RI0 будет сгенерировано прерывание, если оно разрешено.

**Рисунок 17.4. Временные диаграммы в режиме 8-разрядного УАПП**

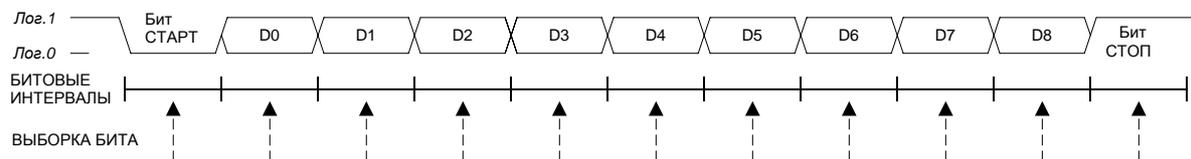


### 17.2.2. 9-разрядный УАПП.

В режиме 9-разрядного УАПП для передачи одного байта данных используются 11 бит: один стартовый бит, восемь бит данных (МЗР вперед), программируемый девятый бит данных и один стоповый бит. При передаче значение девятого бита данных определяется значением бита TB80 (SCON0.3), который устанавливается/сбрасывается программой пользователя. Значение девятого бита может соответствовать значению флага четности «P» регистра PSW (применяется для обнаружения ошибок) или использоваться для организации связи с несколькими МК. При приеме значение девятого бита сохраняется в бите RB80 (SCON0.2), а стоповый бит игнорируется.

Передача данных начинается, когда происходит запись байта данных в регистр SBUF0. Флаг прерывания от передатчика TI0 (SCON0.1) устанавливается в 1 в конце передачи (в начале передачи стопового бита). Прием данных может быть начат в любое время после установки в 1 флага включения приемника RENO (SCON0.4). После приема стопового бита байт данных будет загружен в регистр приемника SBUF0, если соблюдаются следующие условия: RI0 должен быть равен лог.0, и, если MCE0 = 1, то стоповый бит должен быть равен лог.1 (когда MCE0 = 0, состояние девятого бита данных не имеет значения). Если эти условия соблюдаются, то восемь бит данных сохраняются в регистре SBUF0, девятый бит данных сохраняется в бите RB80 и устанавливается в 1 флаг RI0. Если эти условия не соблюдаются, то SBUF0 и RB80 не будут загружаться и флаг RI0 не будет устанавливаться. При установке флагов TI0 или RI0 будет сгенерировано прерывание от модуля УАППО, если оно разрешено.

**Рисунок 17.5. Временные диаграммы в режиме 9-разрядного УАПП**



### 17.3. Поддержка связи с несколькими МК

9-разрядный режим УАПП поддерживает мультимикроконтроллерный обмен данными между ведущим МК и одним или несколькими ведомыми МК, для чего особым образом используется девятый бит данных. Когда ведущий МК хочет передать данные одному или нескольким ведомым МК, он прежде всего посылает байт адреса, чтобы выбрать конкретное(-ые) устройство(-а). Адресный байт отличается от байта данных тем, что его девятый бит равен лог.1; в байте данных девятый бит всегда равен лог.0.

Установка в 1 бита MCE0 (SCON0.5) ведомого МК настраивает его модуль УАППО таким образом, что при получении стопового бита УАППО будет генерировать прерывание только в том случае, если принятый девятый бит равен лог.1 ( $RB80 = 1$ ), уведомляя тем самым систему о том, что принят адресный байт. В процедуре обработки прерывания от УАППО необходимо сравнить принятый адрес с собственным 8-разрядным адресом ведомого. Если эти адреса совпадают, ведомый МК должен сбросить в 0 свой бит MCE0, чтобы разрешить генерацию прерываний при получении следующих байтов данных. Ведомые МК, которые не были адресованы, оставляют свои биты MCE0 равными 1 и не генерируют запросов прерываний при получении следующих байтов данных, т.е. игнорируют их. После того, как все сообщение получено, адресованный ведомый МК устанавливает в 1 бит MCE0, чтобы игнорировать все послыки до получения следующего адресного байта.

Одному ведомому устройству можно назначить несколько адресов и/или один адрес можно назначить нескольким ведомым устройствам, поэтому возможна “широковещательная” передача данных более чем одному ведомому устройству одновременно. Ведущий процессор можно настроить на прием всех сообщений или протокол обмена можно реализовать таким образом, чтобы временно менять местами ведущего и ведомого, обеспечив тем самым полудуплексный обмен данными между истинным ведущим и ведомым(-и).

Рисунок 17.6. Пример использования УАПП в многопроцессорном режиме

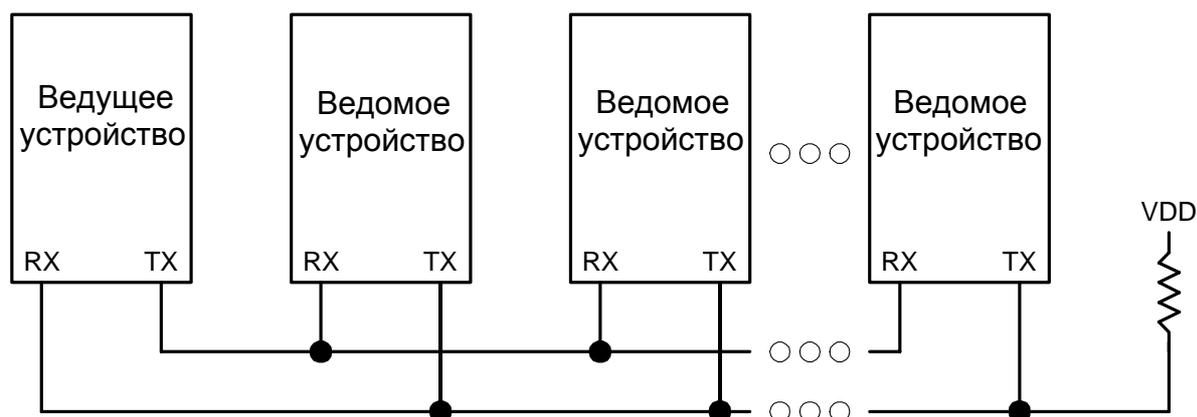


Рисунок 17.7. SCON0: Регистр управления УАПП0

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 01000000 SFR Адрес: 0x98
S0MODE	-	MCE0	REN0	TB80	RB80	TI0	RI0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

Бит 7: S1MODE: Режим работы УАПП0.  
Этот бит выбирает режим работы УАПП0.  
0: Режим 0: 8-разрядный УАПП с изменяемой скоростью передачи данных.  
1: Режим 1: 9-разрядный УАПП с изменяемой скоростью передачи данных.

Бит 6: Не используется. Читается как 1b. Запись не имеет значения.

Бит 5: MCE0: Разрешение поддержки мультимикроконтроллерного взаимодействия.  
Функционирование этого бита зависит от режима работы УАПП0.  
Режим 0: Проверка корректности стопового бита.  
0: Логический уровень стопового бита игнорируется.  
1: Флаг RI0 будет установлен только в том случае, если стоповый бит равен лог.1.  
Режим 1: Разрешение поддержки мультимикроконтроллерного взаимодействия.  
0: Логический уровень девятого бита игнорируется.  
1: Флаг RI0 устанавливается и прерывание генерируется только в том случае, если девятый бит равен лог.1.

Бит 4: REN0: Разрешение приема.  
Этот бит включает/отключает приемник УАПП0.  
0: Прием данных модулем УАПП0 запрещен.  
1: Прием данных модулем УАПП0 разрешен.

Бит 3: TB80: Девятый бит передаваемых данных.  
Значение этого бита будет передано в качестве девятого бита данных в 9-разрядном режиме работы УАПП0. В 8-разрядном режиме работы УАПП0 этот бит не используется. Бит TB80 устанавливается и сбрасывается программно.

Бит 2: RB80: Девятый бит принимаемых данных.  
Этот бит принимает значение полученного стопового бита в режиме 0. В режиме 1 бит RB80 принимает значение девятого бита данных.

Бит 1: TI0: Флаг прерывания от передатчика УАПП0.  
Устанавливается в 1 аппаратно по окончании передачи байта данных (после передачи 8-го бита в режиме 0, или в начале передачи стопового бита в режиме 1). Если прерывание от УАПП0 разрешено, то установка этого бита вызовет переход на процедуру обслуживания прерывания от УАПП0. Этот бит должен сбрасываться программно.

Бит 0: RI0: Флаг прерывания от приемника УАПП0.  
Устанавливается в 1 аппаратно при приеме байта данных (Устанавливается в момент выборки стопового бита). Если прерывание от УАПП0 разрешено, то установка этого бита вызовет переход на процедуру обслуживания прерывания от УАПП0. Этот бит должен сбрасываться программно.



**Рисунок 17.8. SBUF0: Регистр буфера данных УАППО**

R/W	Значение при сбросе: 00000000 SFR Адрес: 0x99							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: SBUF0.[7:0]: Биты (7-0) буфера данных УАППО (СЗР - МЗР)

На самом деле именем SBUF0 обозначаются два регистра: сдвиговый регистр передатчика и регистр-защелка приемника. Когда данные помещаются в регистр SBUF0, они записываются в сдвиговый регистр передатчика и сохраняются для последовательной передачи. Запись данных в SBUF0 инициирует передачу. Когда данные читаются из регистра SBUF1, они считываются из регистра-защелки приемника.

**Таблица 17.1. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внутреннего генератора**

Частота: 24,5 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внутреннего генератора	230400	-0,32%	106	SYSCLK	XX	1	0xCB
	115200	-0,32%	212	SYSCLK	XX	1	0x96
	57600	0,15%	426	SYSCLK	XX	1	0x2B
	28800	-0,32%	848	SYSCLK/4	01	0	0x96
	14400	0,15%	1704	SYSCLK/12	00	0	0xB9
	9600	-0,32%	2544	SYSCLK/12	00	0	0x96
	2400	-0,32%	10176	SYSCLK/48	10	0	0x96
	1200	0,15%	20448	SYSCLK/48	10	0	0x2B

X – Не имеет значения

\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.

**Таблица 17.2. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внешнего генератора**

Частота: 25,0 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внешнего генератора	230400	-0,47%	108	SYSCLK	XX	1	0xCA
	115200	0,45%	218	SYSCLK	XX	1	0x93
	57600	-0,01%	434	SYSCLK	XX	1	0x27
	28800	0,45%	872	SYSCLK/4	01	0	0x93
	14400	-0,01%	1736	SYSCLK/4	01	0	0x27
	9600	0,15%	2608	EXTCLK/8	11	0	0x5D
	2400	0,45%	10464	SYSCLK/48	10	0	0x93
	1200	-0,01%	20832	SYSCLK/48	10	0	0x27
SYSCLK от внутреннего генератора	57600	-0,47%	432	EXTCLK/8	11	0	0xE5
	28800	-0,47%	864	EXTCLK/8	11	0	0xCA
	14400	0,45%	1744	EXTCLK/8	11	0	0x93
	9600	0,15%	2608	EXTCLK/8	11	0	0x5D

X – Не имеет значения

\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.



**Таблица 17.3. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внешнего генератора**

Частота: 22,1184 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внешнего генератора	230400	0,00%	96	SYSCLK	XX	1	0xD0
	115200	0,00%	192	SYSCLK	XX	1	0xA0
	57600	0,00%	384	SYSCLK	XX	1	0x40
	28800	0,00%	768	SYSCLK/12	00	0	0xE0
	14400	0,00%	1536	SYSCLK/12	00	0	0xC0
	9600	0,00%	2304	SYSCLK/12	00	0	0xA0
	2400	0,00%	9216	SYSCLK/48	10	0	0xA0
	1200	0,00%	18432	SYSCLK/48	10	0	0x40
SYSCLK от внутреннего генератора	230400	0,00%	96	EXTCLK/8	11	0	0xFA
	115200	0,00%	192	EXTCLK/8	11	0	0xF4
	57600	0,00%	384	EXTCLK/8	11	0	0xE8
	28800	0,00%	768	EXTCLK/8	11	0	0xD0
	14400	0,00%	1536	EXTCLK/8	11	0	0xA0
	9600	0,00%	2304	EXTCLK/8	11	0	0x70

X – Не имеет значения

\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.

**Таблица 17.4. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внешнего генератора**

Частота: 18,432 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внешнего генератора	230400	0,00%	80	SYSCLK	XX	1	0xD8
	115200	0,00%	160	SYSCLK	XX	1	0xB0
	57600	0,00%	320	SYSCLK	XX	1	0x60
	28800	0,00%	640	SYSCLK/4	01	0	0xB0
	14400	0,00%	1280	SYSCLK/4	01	0	0x60
	9600	0,00%	1920	SYSCLK/12	00	0	0xB0
	2400	0,00%	7680	SYSCLK/48	10	0	0xB0
	1200	0,00%	15360	SYSCLK/48	10	0	0x60
SYSCLK от внутреннего генератора	230400	0,00%	80	EXTCLK/8	11	0	0xFB
	115200	0,00%	160	EXTCLK/8	11	0	0xF6
	57600	0,00%	320	EXTCLK/8	11	0	0xEC
	28800	0,00%	640	EXTCLK/8	11	0	0xD8
	14400	0,00%	1280	EXTCLK/8	11	0	0xB0
	9600	0,00%	1920	EXTCLK/8	11	0	0x88

X – Не имеет значения

\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.

**Таблица 17.5. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внешнего генератора**

Частота: 11,0592 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внешнего генератора	230400	0,00%	48	SYSCLK	XX	1	0xE8
	115200	0,00%	96	SYSCLK	XX	1	0xD0
	57600	0,00%	192	SYSCLK	XX	1	0xA0
	28800	0,00%	384	SYSCLK	XX	1	0x40
	14400	0,00%	768	SYSCLK/12	00	0	0xE0
	9600	0,00%	1152	SYSCLK/12	00	0	0xD0
	2400	0,00%	4608	SYSCLK/12	00	0	0x40
SYSCLK от внутреннего генератора	1200	0,00%	9216	SYSCLK/48	10	0	0xA0
	230400	0,00%	48	EXTCLK/8	11	0	0xFD
	115200	0,00%	96	EXTCLK/8	11	0	0xFA
	57600	0,00%	192	EXTCLK/8	11	0	0xF4
	28800	0,00%	384	EXTCLK/8	11	0	0xE8
	14400	0,00%	768	EXTCLK/8	11	0	0xD0
	9600	0,00%	1152	EXTCLK/8	11	0	0xB8

X – Не имеет значения

\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.

**Таблица 17.6. Параметры настройки таймера для стандартных скоростей передачи данных при тактировании от внешнего генератора**

Частота: 3,6864 МГц							
	Требуемая скорость передачи данных (бит/сек)	Погрешность установки скорости передачи данных	Коэффициент деления генератора	Частота сигнала тактирования	SCA1-SCA0 (выбор коэффициента предварительного деления)*	T1M*	Значение перезагрузки Таймера 1
SYSCLK от внешнего генератора	230400	0,00%	16	SYSCLK	XX	1	0xF8
	115200	0,00%	32	SYSCLK	XX	1	0xF0
	57600	0,00%	64	SYSCLK	XX	1	0xE0
	28800	0,00%	128	SYSCLK	XX	1	0xC0
	14400	0,00%	256	SYSCLK	XX	1	0x80
	9600	0,00%	384	SYSCLK	XX	1	0x40
	2400	0,00%	1536	SYSCLK/12	00	0	0xC0
	1200	0,00%	3072	SYSCLK/12	00	0	0x80
SYSCLK от внутреннего генератора	230400	0,00%	16	EXTCLK/8	11	0	0xFF
	115200	0,00%	32	EXTCLK/8	11	0	0xFE
	57600	0,00%	64	EXTCLK/8	11	0	0xFC
	28800	0,00%	128	EXTCLK/8	11	0	0xF8
	14400	0,00%	256	EXTCLK/8	11	0	0xF0
	9600	0,00%	384	EXTCLK/8	11	0	0xE8

X – Не имеет значения

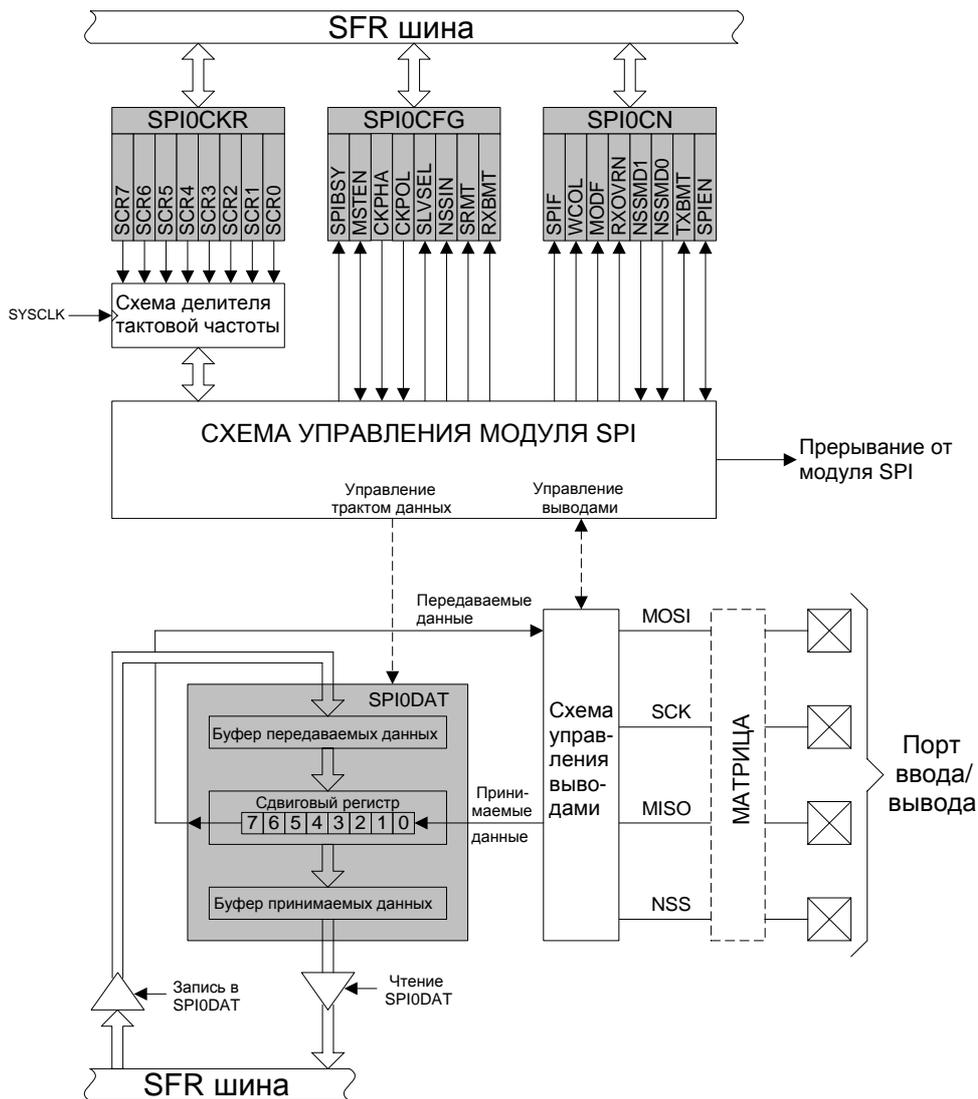
\* Определения бит SCA1 – SCA0 и T1M приведены в разделе 19.1.



### 18. МОДУЛЬ SPI (SPI0)

Расширенный модуль SPI0 обеспечивает доступ к гибкой полнодуплексной синхронной последовательной шине. SPI0 может выполнять функции ведущего или ведомого устройства в 3-х проводном или 4-х проводном режимах, а также поддерживает работу нескольких ведомых и ведущих устройств на одной шине. Сигнал выбора ведомого (NSS) можно настроить как вход выбора SPI0 в ведомом режиме или как вход отключения функций ведущего при работе на шине с несколькими ведущими, что позволяет предотвратить конфликты на шине в том случае, если два или более ведущих попытаются передать данные одновременно. Кроме этого NSS можно настроить как выход выбора кристалла в ведущем режиме или отключить при работе в 3-х проводном режиме. В ведущем режиме можно также использовать порт ввода/вывода общего назначения для выбора нескольких ведомых устройств.

Рисунок 18.1. Структурная схема модуля SPI



## 18.1. Описание сигналов

Ниже описаны четыре сигнала, используемые интерфейсом SPI0 (MOSI, MISO, SCK, NSS).

### 18.1.1. Выход ведущего, вход ведомого

Сигнал MOSI (master-out, slave-in - «выход ведущего, вход ведомого») является выходом данных ведущего устройства и входом данных ведомых устройств. Он используется для последовательной передачи данных от ведущего к ведомому. Этот сигнал является выходом, если SPI0 работает в ведущем режиме, и входом, если SPI0 работает в ведомом режиме. Данные передаются старшими значащими разрядами вперед. При работе в ведущем режиме значение сигнала MOSI определяется старшим значащим разрядом сдвигового регистра как в 3-х проводном, так и в 4-х проводном режимах.

### 18.1.2. Вход ведущего, выход ведомого

Сигнал MISO (master-in, slave-out - «вход ведущего, выход ведомого») является выходом данных ведомого устройства и входом данных ведущего устройства. Он используется для последовательной передачи данных от ведомого к ведущему. Этот сигнал является входом, если SPI0 работает в ведущем режиме, и выходом, если SPI0 работает в ведомом режиме. Данные передаются старшими значащими разрядами вперед. Вывод MISO переводится в высокоимпедансное состояние, когда модуль SPI0 отключен, а также тогда, когда модуль SPI0 работает в 4-х проводном режиме как ведомый, который не выбран. Когда модуль SPI0 работает в 3-х проводном режиме как ведомый, сигнал MISO всегда определяется старшим значащим разрядом сдвигового регистра.

### 18.1.3. Тактовые импульсы

Сигнал SCK (serial clock – «импульсы тактирования последовательного интерфейса») является выходом ведущего устройства и входом ведомых устройств. Он используется для синхронизации обмена данными между ведущим и ведомым устройствами по линиям MOSI и MISO. SPI0 генерирует этот сигнал, когда работает в ведущем режиме. В 4-х проводном ведомом режиме сигнал SCK игнорируется ведомым SPI, когда ведомый не выбран (NSS = 1).

### 18.1.4. Выбор ведомого (NSS)

Функционирование сигнала выбора ведомого (NSS) зависит от состояния бит NSSMD1 и NSSMD0 регистра SPI0CN. С помощью этих бит можно выбрать три возможных режима:

1. NSSMD[1:0] = 00: 3-х проводный ведущий или 3-х проводный ведомый режим: SPI0 работает в 3-х проводном режиме и NSS отключен. В 3-х проводном ведомом режиме SPI0 выбран всегда. Т.к. сигнал выбора отсутствует, то в 3-х проводном режиме SPI0 может быть только ведомым на шине. Этот режим предназначен для организации взаимодействия типа «точка - точка» между ведущим устройством и одним ведомым устройством.
2. NSSMD[1:0] = 01: 4-х проводный ведомый режим или режим с несколькими ведущими: SPI0 работает в 4-х проводном режиме и NSS является входом. При работе в ведомом режиме сигнал NSS является сигналом выбора данного ведомого. При работе в ведущем режиме срез (переход из состояния 1 в состояние 0) сигнала NSS отключает функции ведущего SPI0, что позволяет работать на одной SPI шине нескольким ведущим устройствам.
3. NSSMD[1:0] = 1x: 4-х проводный ведущий режим: SPI0 работает в 4-х проводном режиме и NSS является выходом. Значение бита NSSMD0 определяет, сигнал какого логического уровня будет выведен на вывод NSS. Эту конфигурацию следует использовать только тогда, когда SPI0 работает в ведущем режиме.

На рис.18.2, рис.18.3, рис.18.4 показаны типичные схемы включения для различных режимов работы. Следует иметь в виду, что состояние бит NSSMD влияет на разводку выводов МК. В 3-х проводном ведомом или 3-х проводном ведущем режимах вывод NSS не будет разводиться матрицей. Во всех других режимах сигнал NSS будет выводиться на внешний вывод МК. Подробная информация о портах ввода/вывода общего назначения и матрице приведена в разделе 14 «Порт ввода/вывода» на стр.127.



## 18.2. Функционирование SPI0 в ведущем режиме

Все сеансы обмена данными по SPI шине инициируются ведущим устройством. Модуль SPI0 переводится в ведущий режим работы установкой в 1 флага включения ведущего режима (MSTEN, SPI0CN.6). Если модуль SPI0 работает в ведущем режиме, то запись байта данных в регистр данных модуля SPI0 (SPI0DAT) вызовет загрузку буфера передатчика. Если сдвиговый регистр модуля SPI0 пуст, то в него загружается байт из буфера передатчика и начинается передача данных. Ведущий SPI сразу же начинает последовательно сдвигать данные на линию MOSI, выдавая тактовые импульсы на линию SCK. По окончании передачи устанавливается в 1 флаг SPIF (SPI0CN.7). Если прерывания разрешены, то при установке флага SPIF генерируется запрос прерывания. В полнодуплексном режиме работы в то время, когда ведущий SPI0 передает данные ведомому по линии MOSI, адресуемый ведомый одновременно передает содержимое своего регистра сдвига ведущему SPI0 по линии MISO. Поэтому флаг SPIF является как флагом окончания передачи, так и флагом готовности принимаемых данных. Байт данных, принимаемый от ведомого устройства, передается старшими значащими разрядами вперед в сдвиговый регистр ведущего. После полной загрузки сдвигового регистра полученный байт данных переписывается в буфер приемника, откуда он может быть считан процессором путем чтения регистра SPI0DAT.

Когда модуль SPI0 настроен как ведущий, он может работать в одном из трех различных режимов:

- режим работы с несколькими ведущими;
- 3-х проводный режим работы с одним ведущим;
- 4-х проводный режим работы с одним ведущим.
- 

Активным по умолчанию является режим работы с несколькими ведущими, когда NSSMD1 (SPI0CN.3) = 0 и NSSMD0 (SPI0CN.2) = 1. В этом режиме NSS функционирует как вход и используется для отключения ведущего SPI0 в то время, когда другой ведущий пытается получить доступ к шине. Если в этом режиме на вход NSS подается сигнал низкого логического уровня, то сбрасываются в 0 биты MSTEN (SPI0CN.6) и SPIEN (SPI0CN.0), выключая тем самым ведущий модуль SPI0, и устанавливается в 1 флаг ошибки режима MODF (SPI0CN.5). При установке флага ошибки режима будет сгенерировано прерывание, если оно разрешено. При данных обстоятельствах требуется программно вновь включить модуль SPI0. В системе с несколькими ведущими любое устройство обычно по умолчанию становится ведомым устройством, если оно не функционирует как ведущее устройство системы. В режиме с несколькими ведущими ведомые устройства можно адресовать индивидуально (при необходимости), используя порты ввода/вывода общего назначения. На рис.18.2 приведена схема соединений между двумя ведущими устройствами в режиме с несколькими ведущими.

3-х проводный режим работы с одним ведущим активен тогда, когда NSSMD1 (SPI0CN.3) = 0 и NSSMD0 (SPI0CN.2) = 0. В этом режиме NSS не используется и не разводится на внешний вывод порта с помощью матрицы. Все ведомые устройства, которые требуется адресовать в этом режиме работы, необходимо выбирать с помощью портов ввода/вывода общего назначения. На рис.18.3 приведена схема соединений между ведущим и ведомым устройствами в 3-х проводном режиме.

4-х проводный режим работы с одним ведущим активен тогда, когда NSSMD1 (SPI0CN.3) = 1. В этом режиме NSS функционирует как выход и может использоваться как сигнал выбора ведомого для одного устройства SPI. Логический уровень сигнала на выходе NSS определяется битом NSSMD0 (SPI0CN.2). Другие ведомые устройства можно адресовать с помощью портов ввода/вывода общего назначения. На рис.18.4 приведена схема соединений между одним ведущим и двумя ведомыми устройствами в 4-х проводном режиме.

Рисунок 18.2. Схема включения в режиме с несколькими ведущими

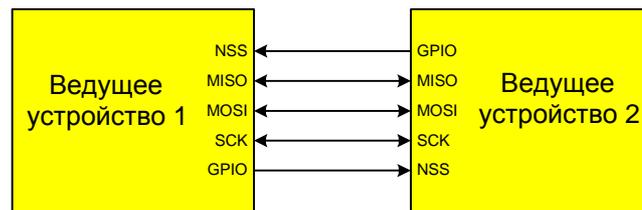


Рисунок 18.3. Схема соединения одного ведущего и одного ведомого с использованием 3-х проводной шины SPI

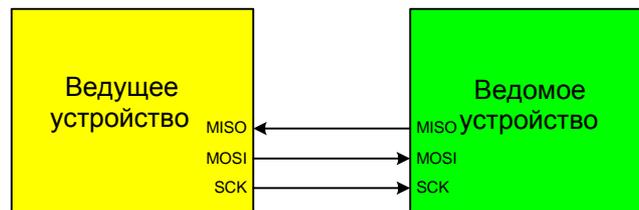
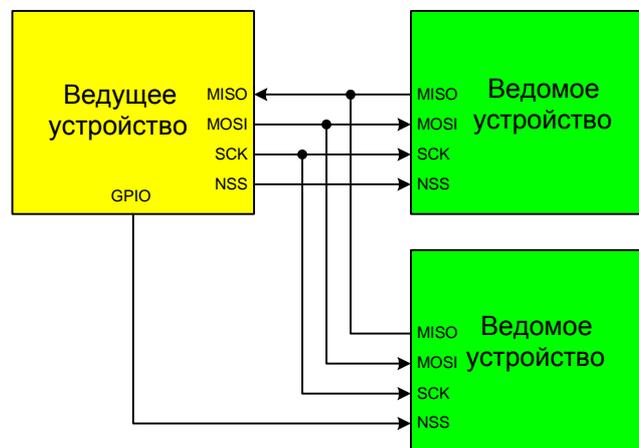


Рисунок 18.4. Схема соединения одного ведущего и нескольких ведомых с использованием 4-х проводной шины SPI



## 18.3. Функционирование SPI0 в ведомом режиме

Когда модуль SPI0 включен и не настроен как ведущий, он будет функционировать как ведомый SPI. Байты данных принимаются по линии MOSI от ведущего и передаются по линии MISO ведущему, при этом ведущее устройство управляет сигналом на линии SCK. Битовый счетчик модуля SPI0 подсчитывает фронты сигнала SCK. После того, как 8 бит данных приняты в сдвиговый регистр, устанавливается в 1 флаг SPIF и байт данных копируется в буфер приемника. Данные считываются из буфера приемника путем чтения регистра SPI0DAT. Ведомое устройство не может инициировать процесс обмена данными. Данные, которые необходимо передать ведущему, предварительно загружаются в сдвиговый регистр путем записи регистра SPI0DAT. При записи регистра SPI0DAT данные сначала загружаются в буфер передатчика. Если сдвиговый регистр пуст, то содержимое буфера передатчика будет сразу же передано в сдвиговый регистр. В том случае, если сдвиговый регистр уже содержит данные, то SPI0 будет ждать окончания их передачи, и только после этого загрузит сдвиговый регистр содержимым буфера передатчика.

Когда модуль SPI0 функционирует как ведомый, его можно настроить на работу в 3-х проводном или 4-проводном режимах. Активным по умолчанию является 4-проводный ведомый режим, когда NSSMD1 (SPI0CN.3) = 0 и NSSMD0 (SPI0CN.2) = 1. В 4-проводном режиме сигнал NSS выведен на внешний вывод порта, который настроен как цифровой вход. SPI0 включен, когда NSS = 0, и отключен, когда NSS = 1. Битовый счетчик сбрасывается по заднему фронту сигнала NSS. Следует иметь в виду, что сигнал NSS необходимо сбросить в 0 как минимум за 2 системных тактовых цикла до первого активного фронта сигнала SCK для каждого передаваемого байта. На рис.18.4 приведена схема соединений между одним ведущим и двумя ведомыми устройствами в 4-х проводном режиме.

3-проводный ведомый режим активен, когда NSSMD1 (SPI0CN.3) = 0 и NSSMD0 (SPI0CN.2) = 0. NSS не используется и не разводится на внешний вывод порта с помощью матрицы. Т.к. в этом режиме нет способа однозначной адресации устройства, то SPI0 должен быть единственным ведомым устройством, присутствующим на шине. Важно иметь в виду, что в 3-х проводном ведомом режиме отсутствуют средства для сброса битового счетчика. Данный счетчик определяет момент окончания приема байта. Этот битовый счетчик можно сбросить лишь путем выключения и повторного включения модуля SPI0 с помощью бита SPIEN. На рис.18.3 приведена схема соединений между ведущим и ведомым устройствами в 3-х проводном режиме.

## 18.4. Источники прерываний модуля SPI0

Если прерывания от модуля SPI0 разрешены, то следующие 4 флага будут генерировать прерывания при установке их в 1:

*Все приведенные ниже флаги сбрасываются программно.*

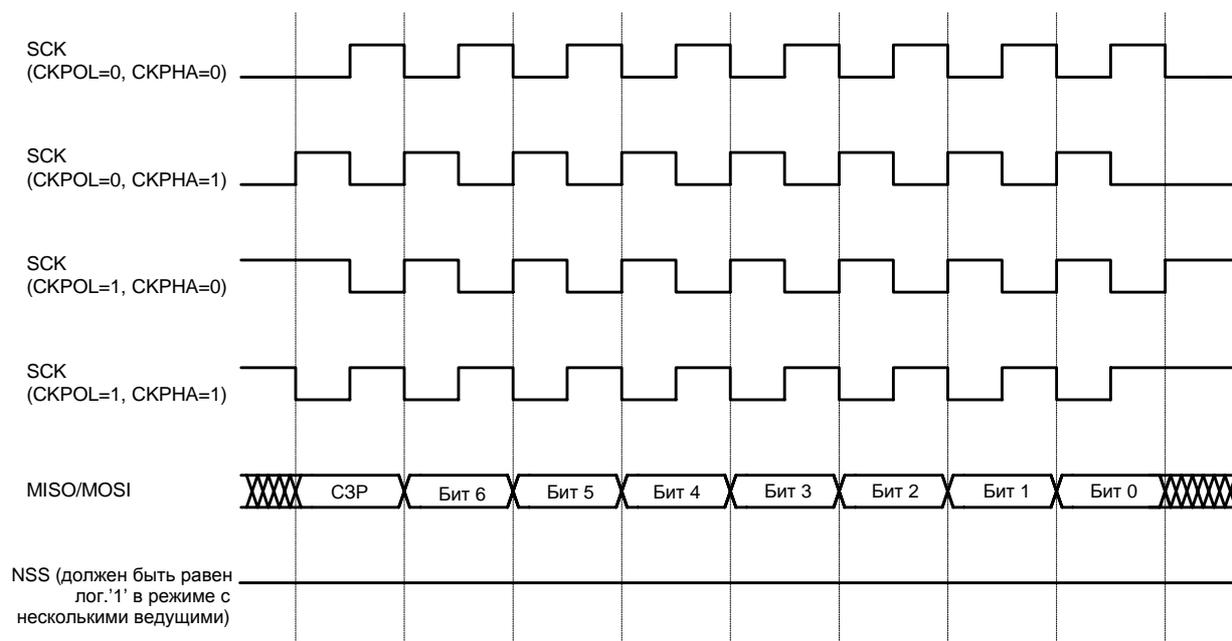
1. Флаг прерывания от модуля SPI0 SPIF (SPI0CN.7) устанавливается в 1 по окончании передачи каждого байта. Установка этого флага возможна во всех режимах работы модуля SPI0.
2. Флаг конфликта записи WCOL (SPI0CN.6) устанавливается в 1, если запись в регистр SPI0DAT происходит в тот момент, когда данные из буфера передатчика еще не переписаны в сдвиговый регистр. В этом случае запись в регистр SPI0DAT игнорируется и буфер передатчика не переписывается. Установка этого флага возможна во всех режимах работы модуля SPI0.
3. Флаг ошибки режима MODF (SPI0CN.5) устанавливается в 1, если модуль SPI0 функционирует как ведущий в режиме работы с несколькими ведущими и на входе NSS появляется сигнал с низким логическим уровнем. В этом случае будут сброшены в 0 биты MSTEN и SPIEN в регистре SPI0CN, в результате чего модуль SPI0 будет отключен. Это позволит другому ведущему устройству получить доступ к шине.
4. Флаг переполнения приемника RXOVRN (SPI0CN.4) устанавливается в 1, если при работе в режиме ведомого передача завершается, а буфер приемника все еще содержит непрочитанный байт от предыдущей передачи. Новый байт не переписывается в буфер приемника, что позволяет прочитать ранее принятый байт данных. Байт данных, который вызвал переполнение приемника, теряется.

## 18.5. Тактирование

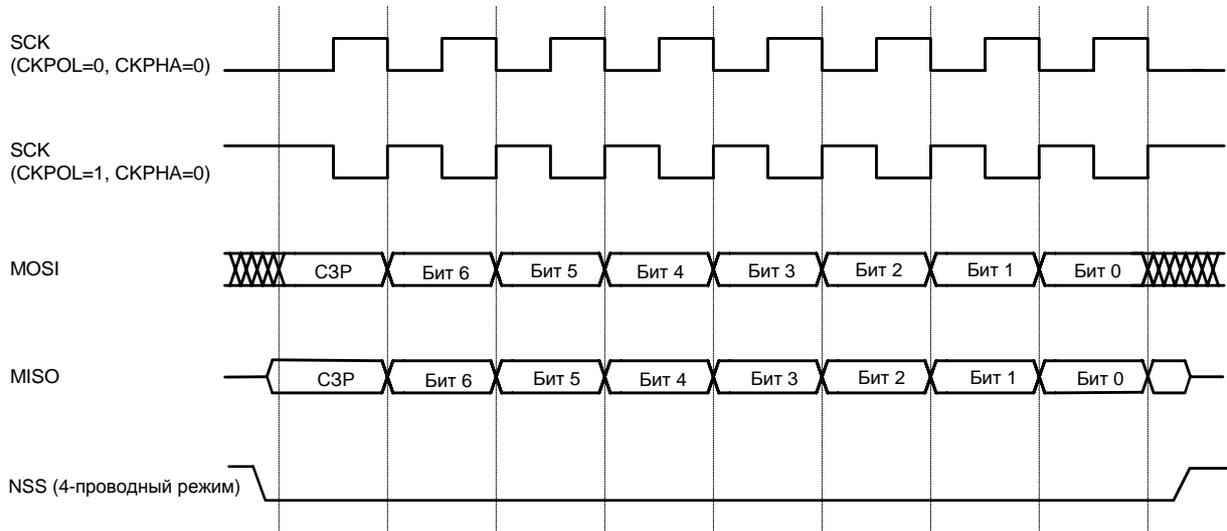
Используя биты управления тактовой частотой регистра конфигурации модуля SPI0 (SPI0CFG), можно выбрать четыре комбинации фазы и полярности импульсов тактирования последовательного интерфейса. Бит СКРНА (SPI0CFG.5) выбирает одну из двух фаз тактового сигнала (фронт, используемый для фиксации данных). Бит СКPOL (SPI0CFG.4) задает активный уровень (высокий или низкий) тактового сигнала. Как ведущий, так и ведомые устройства должны быть настроены на использование одинаковых фазы и полярности тактовых импульсов. При изменении фазы и полярности тактовых импульсов модуль SPI0 следует отключить сбросом в 0 бита SPIEN (SPI0CN.0). Временные диаграммы сигналов данных и тактирования для ведущего режима приведены на рис.18.5. Временные диаграммы сигналов данных и тактирования для ведомого режима приведены на рис.18.6 и рис.18.7. Следует иметь ввиду, что бит СКРНА должен быть сброшен в 0 как у ведущего, так и у ведомого SPI при обмене данными между любыми двумя из следующих МК: C8051F04x, C8051F06x, C8051F12x, C8051F31x, C8051F32x и C8051F33x.

Регистр установки тактовой частоты модуля SPI0 (SPI0CKR), показанный на рис.18.10, управляет частотой тактирования последовательного интерфейса при работе в ведущем режиме. При работе в ведомом режиме содержимое этого регистра игнорируется. Когда модуль SPI0 настроен как ведущий, максимальная скорость передачи данных (в бит/сек) равна половине системной тактовой частоты (12,5 МГц или меньше). Когда модуль SPI0 настроен как ведомый, максимальная скорость передачи данных (в бит/сек) для полнодуплексного режима работы равна 1/10 системной тактовой частоты, при условии, что сигналы от ведущего SCK, NSS (в 4-х проводном ведомом режиме) и последовательные входные данные синхронизированы с системной тактовой частотой ведомого. Если сигналы от ведущего SCK, NSS и последовательные входные данные асинхронны, то максимальная скорость передачи данных (в бит/сек) должна быть меньше 1/10 системной тактовой частоты. В особом случае, когда ведущему требуется только передавать данные ведомому и не требуется принимать от него данные (т.е. полудуплексный режим работы), ведомый модуль SPI может принимать данные с максимальной скоростью (в бит/сек), равной 1/4 системной тактовой частоты. Это справедливо при условии, что сигналы от ведущего SCK, NSS и последовательные входные данные синхронизированы с системной тактовой частотой ведомого.

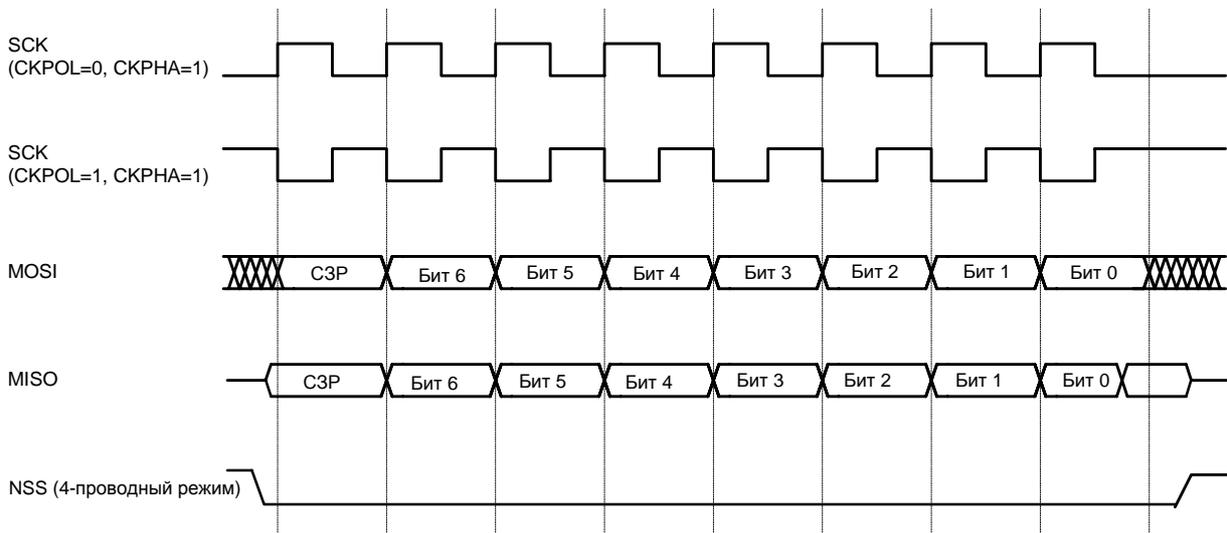
**Рисунок 18.5. Временные диаграммы сигналов данных/тактирования в режиме ведущего**



**Рисунок 18.6. Временные диаграммы сигналов данных/тактирования в режиме ведомого (СКРНА = 0)**



**Рисунок 18.7. Временные диаграммы сигналов данных/тактирования в режиме ведомого (СКРНА = 1)**



## 18.6. Регистры специального назначения модуля SPI0

Для доступа к интерфейсу SPI и управления им используются четыре регистра специального назначения: регистр управления SPI0CN, регистр данных SPI0DAT, регистр конфигурации SPI0CFG и регистр установки тактовой частоты SPI0CKR. Все эти регистры описаны в следующих разделах.

**Рисунок 18.8. SPI0CFG: Регистр конфигурации модуля SPI0**

R	R/W	R/W	R/W	R	R	R	R	Значение при сбросе: 00000111 SFR Адрес: 0xA1
SPIBSY	MSTEN	СКРНА	СКPOL	SLVSEL	NSSIN	SRMT	RXBMT	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p><b>Бит 7: SPIBSY:</b> Флаг занятости модуля SPI0 (только для чтения). Этот бит устанавливается в 1 тогда, когда SPI0 находится в процессе передачи данных (ведущий или ведомый режим).</p> <p><b>Бит 6: MSTEN:</b> Включение ведущего режима. 0: Ведущий режим отключен. Модуль SPI0 работает в ведомом режиме. 1: Ведущий режим включен. Модуль SPI0 работает в ведущем режиме.</p> <p><b>Бит 5: СКРНА:</b> Выбор активной фазы тактового сигнала модуля SPI0. Этот бит управляет фазой тактового сигнала модуля SPI0. 0: Данные фиксируются по первому фронту периода сигнала SCK.* 1: Данные фиксируются по второму фронту периода сигнала SCK.*</p> <p><b>Бит 4: СКPOL:</b> Выбор полярности тактового сигнала модуля SPI0. Этот бит управляет полярностью тактового сигнала модуля SPI0. 0: В состоянии простоя на линии SCK установлен сигнал низкого уровня. 1: В состоянии простоя на линии SCK установлен сигнал высокого уровня.</p> <p><b>Бит 3: SLVSEL:</b> Флаг выбора ведомого (только для чтения). Этот бит аппаратно устанавливается в 1 всякий раз, когда на линию NSS подан сигнал низкого уровня, и показывает, что SPI0 является выбранным ведомым. Этот бит сбрасывается в 0, если на линию NSS подан сигнал высокого уровня (ведомый не выбран). Этот бит отражает не мгновенное состояние сигнала на выводе NSS, а скорее сглаженную (без паразитных выбросов) форму этого сигнала.</p> <p><b>Бит 2: NSSIN:</b> Флаг мгновенного состояния сигнала на входном выводе NSS (только для чтения). Этот бит отражает мгновенное значение сигнала на входном выводе NSS в момент чтения этого регистра. Этот вход не является сглаженным.</p> <p><b>Бит 1: SRMT:</b> Флаг опустошения сдвигового регистра (только для чтения, действителен в ведомом режиме). Этот бит будет устанавливаться в 1 тогда, когда все данные переданы в сдвиговый регистр или из сдвигового регистра, и нет данных для считывания из буфера передатчика или записи в буфер приемника. Этот бит сбрасывается в 0, когда байт данных передается в сдвиговый регистр из буфера передатчика или при изменении сигнала SCK. Примечание: SRMT = 1 в ведущем режиме работы.</p> <p><b>Бит 0: RXBMT:</b> Флаг опустошения буфера приемника (только для чтения, действителен в ведомом режиме). Этот бит будет устанавливаться в 1 тогда, когда буфер приемника прочитан и не содержит новых данных. Если в буфере приемника имеются доступные для чтения новые данные, которые не были прочитаны, то этот бит будет сброшен в 0. Примечание: RXBMT = 1 в ведущем режиме работы.</p> <p>* В ведомом режиме данные на линии MOSI выбираются в центре каждого битового интервала. В ведущем режиме данные на линии MISO выбираются за один цикл SYSCLK до окончания каждого битового интервала, что позволяет обеспечить максимальное время установления сигнала для ведомого устройства. Временные параметры приведены в таблице 18.1.</p>								



Рисунок 18.9. SPI0CN: Регистр управления модуля SPI0

R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	Значение при сбросе: 00000110 SFR Адрес: 0xF8
SPIF	WCOL	MODF	RXOVRN	NSSMD1	NSSMD0	TXBMT	SPIEN	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	
<p><b>Бит 7: SPIF:</b> Флаг прерывания от модуля SPI0. Этот бит аппаратно устанавливается в 1 по окончании передачи данных. Если прерывания разрешены, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от модуля SPI0. Этот бит не сбрасывается аппаратно, его необходимо сбросить программно.</p> <p><b>Бит 6: WCOL:</b> Флаг конфликта записи. Этот бит аппаратно устанавливается в 1 (и генерирует прерывание от модуля SPI0) и тем самым показывает, что была произведена попытка записи в регистр данных модуля SPI0, когда текущий сеанс передачи данных еще не завершился. Этот бит необходимо сбросить программно.</p> <p><b>Бит 5: MODF:</b> Флаг ошибки режима. Этот бит аппаратно устанавливается в 1 (и генерирует прерывание от модуля SPI0) при обнаружении конфликта ведущего режима (на линии NSS низкий уровень, <math>MSTEN = 1</math> и <math>NSSMD[1:0] = 01</math>). Этот бит не сбрасывается аппаратно, его необходимо сбросить программно.</p> <p><b>Бит 4: RXOVRN:</b> Флаг переполнения приемника (только ведомый режим работы). Этот бит аппаратно устанавливается в 1 (и генерирует прерывание от модуля SPI0), если приемный буфер все еще содержит непрочитанные данные от предыдущей передачи, а последний бит текущей передачи сдвигается в сдвиговый регистр модуля SPI0. Этот бит не сбрасывается аппаратно, его необходимо сбросить программно.</p> <p><b>Биты 3-2: NSSMD1-NSSMD0:</b> Биты режима выбора ведомого. С помощью этих бит осуществляется выбор между следующими режимами функционирования вывода NSS (см. раздел 18.2 и раздел 18.3): 00: 3-х проводный ведомый или 3-х проводный ведущий режим. Сигнал NSS не разводится с помощью матрицы на внешний вывод МК. 01: 4-х проводный ведомый режим или режим работы с несколькими ведущими (по умолчанию). Вывод NSS всегда является входом. 1x: 4-проводный режим работы с одним ведущим. Вывод NSS настроен как выход и состояние сигнала на нем определяется значением бита NSSMD0.</p> <p><b>Бит 1: TXBMT:</b> Флаг опустошения буфера передатчика. Этот бит будет сбрасываться в 0 при записи новых данных в буфер передатчика. После передачи данных из буфера передатчика в сдвиговый регистр модуля SPI0 этот бит будет установлен в 1, показывая, что в буфер передатчика можно записывать новые данные.</p> <p><b>Бит 0: SPIEN:</b> Включение модуля SPI0. Этот бит включает/отключает модуль SPI0. 0: Модуль SPI0 отключен. 1: Модуль SPI0 включен.</p>								

Рисунок 18.10. SPI0CKR: Регистр установки тактовой частоты модуля SPI0

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xA2							
SCR7	SCR6	SCR5	SCR4	SCR3	SCR2	SCR1	SCR0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Биты 7-0: SCR7-SCR0: Тактовая частота модуля SPI0

Эти биты определяют частоту выходного сигнала SCK, когда модуль SPI0 работает в ведущем режиме. Частота тактового сигнала SCK представляет собой поделенную на определенный коэффициент системную тактовую частоту и задается следующим уравнением:

$$f_{\text{SCK}} = 0.5 * \text{SYSCLK} / (\text{SPI0CKR} + 1), \text{ для } 0 \leq \text{SPI0CKR} \leq 255,$$

где: SYSCLK – частота системного тактового сигнала;  
SPI0CKR – 8-разрядное значение регистра SPI0CKR.

Пример: Если SYSCLK = 2МГц и SPI0CKR = 0x04, то:

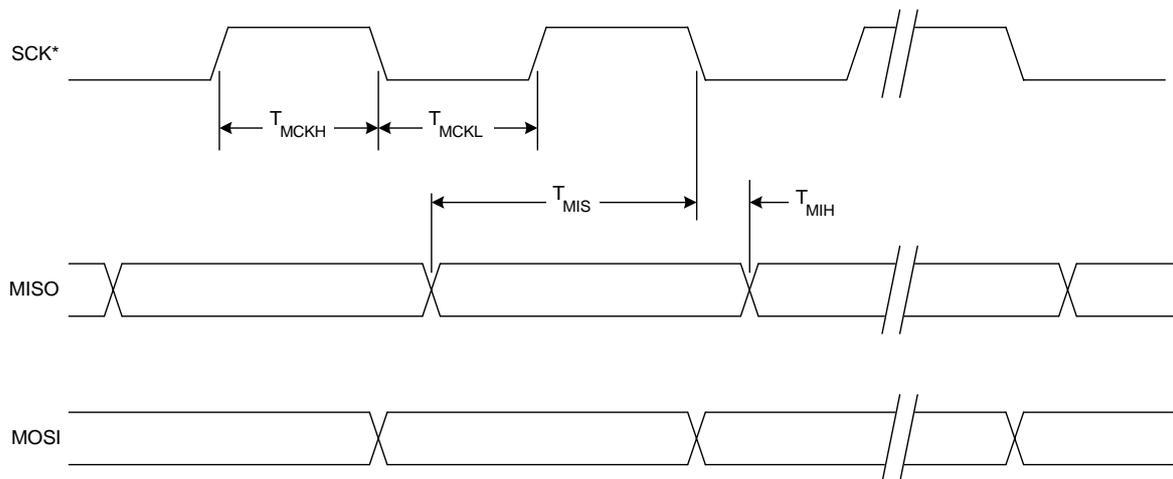
$$f_{\text{SCK}} = 0.5 * 2000000 / (4 + 1) = 200\text{кГц}.$$


**Рисунок 18.11. SPI0DAT: Регистр данных модуля SPI0**

R/W	Значение при сбросе: 00000000 SFR Адрес: 0xA3							
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

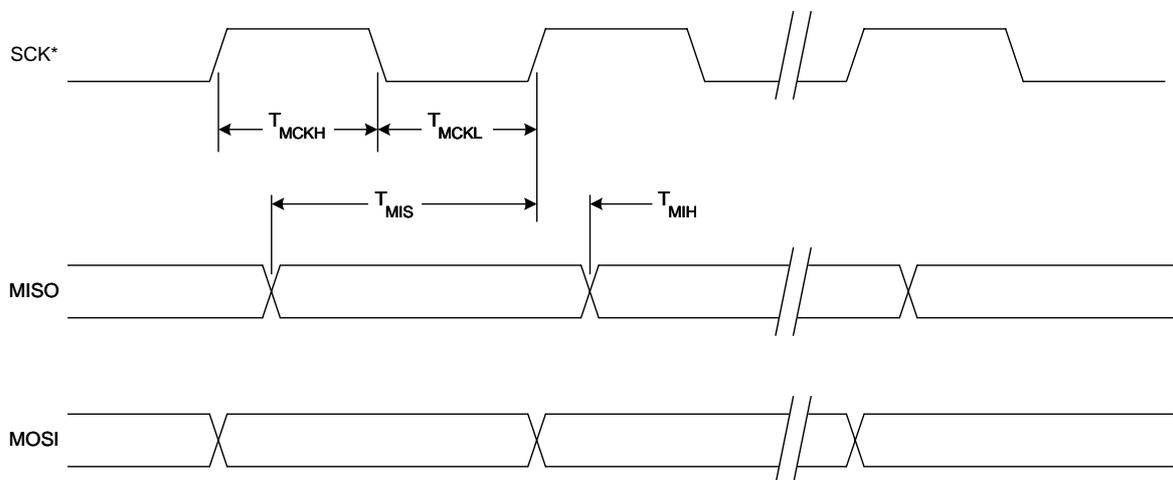
Биты 7-0: SPI0DAT: Данные передатчика и приемника модуля SPI0.  
 Регистр SPI0DAT используется для передачи и приема данных. В ведущем режиме запись данных в регистр SPI0DAT сразу же приводит к загрузке данных в буфер передатчика и инициирует сеанс передачи. Чтение регистра SPI0DAT возвратит содержимое приемного буфера.

Рисунок 18.12. Временные диаграммы ведущего SPI (СКРНА = 0)



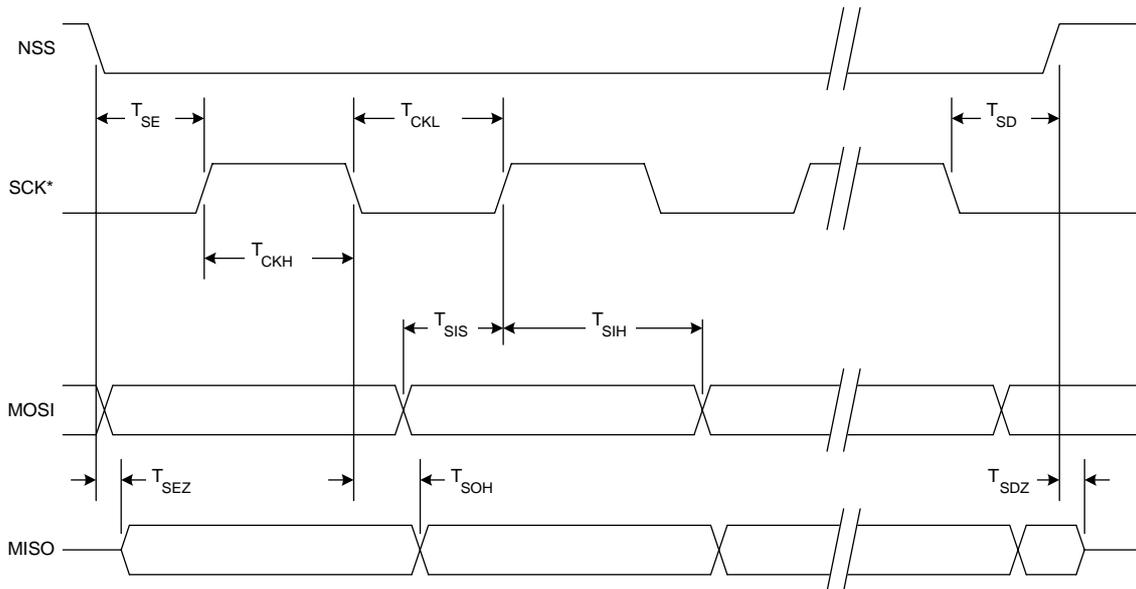
\* Состояние сигнала SCK показано для случая CKPOL = 0. Если CKPOL = 1, то состояние сигнала SCK будет противоположным.

Рисунок 18.13. Временные диаграммы ведущего SPI (СКРНА = 1)



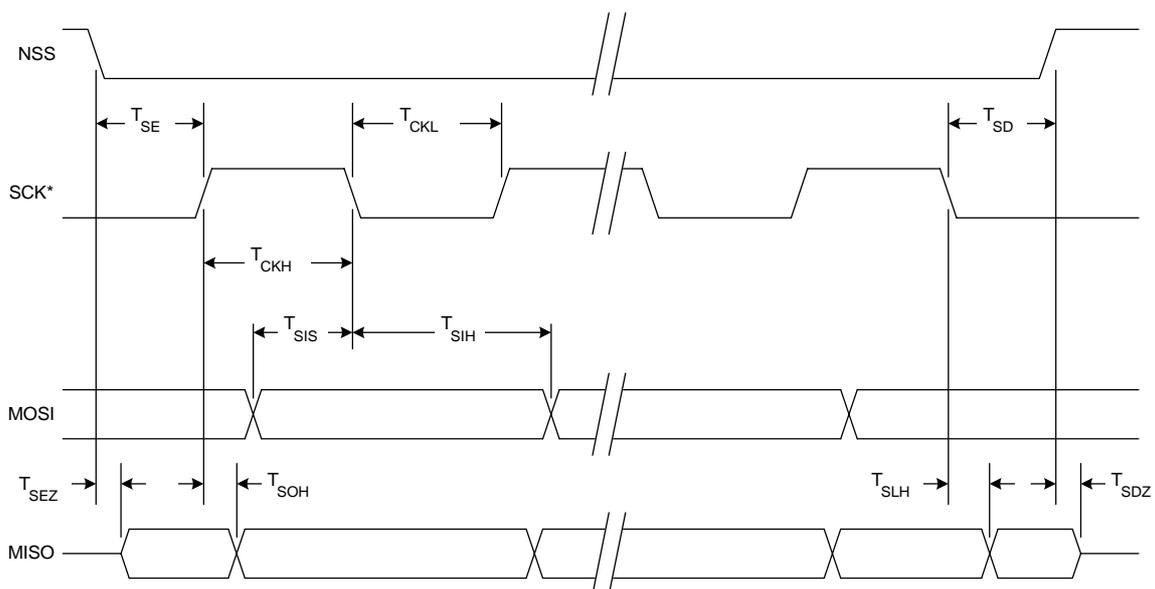
\* Состояние сигнала SCK показано для случая CKPOL = 0. Если CKPOL = 1, то состояние сигнала SCK будет противоположным.

**Рисунок 18.14. Временные диаграммы ведомого SPI (СКРНА = 0)**



\* Состояние сигнала SCK показано для случая CKPOL = 0. Если CKPOL = 1, то состояние сигнала SCK будет противоположным.

**Рисунок 18.15. Временные диаграммы ведомого SPI (СКРНА = 1)**



\* Состояние сигнала SCK показано для случая CKPOL = 0. Если CKPOL = 1, то состояние сигнала SCK будет противоположным.

Таблица 18.1. Временные параметры ведомого SPI

Параметр	Описание	Мин.	Макс.	Ед. изм.
<b>Временные параметры ведущего SPI*</b> (см. рис.18.12 и рис.18.13)				
<b>ТМСКН</b>	Длительность высокого уровня сигнала SCK	1*TSYSCLK		нс
<b>ТМСКЛ</b>	Длительность низкого уровня сигнала SCK	1*TSYSCLK		нс
<b>ТМIS</b>	Время удержания данных на линии MISO до фронта SCK, вызывающего сдвиг данных	1*TSYSCLK+ 20		нс
<b>ТМИН</b>	Длительность интервала между фронтом SCK, вызывающим сдвиг данных, и обновлением данных на линии MISO	0		нс
<b>Временные параметры ведомого SPI*</b> (см. рис.18.14 и рис.18.15)				
<b>TSE</b>	Длительность интервала между появлением сигнала низкого уровня на линии NSS и первым фронтом сигнала SCK	2*TSYSCLK		нс
<b>TSD</b>	Длительность интервала между последним фронтом сигнала SCK и появлением сигнала высокого уровня на линии	2*TSYSCLK		нс
<b>TSEZ</b>	Время установления данных на линии MISO после появления сигнала низкого уровня на линии NSS		4*TSYSCLK	нс
<b>TSDZ</b>	Длительность интервала между появлением сигнала высокого уровня на линии NSS и переходом линии MISO в высокоимпедансное состояние		4*TSYSCLK	нс
<b>ТСКН</b>	Длительность высокого уровня сигнала SCK	5*TSYSCLK		нс
<b>ТСКЛ</b>	Длительность низкого уровня сигнала SCK	5*TSYSCLK		нс
<b>ТСIS</b>	Время удержания данных на линии MOSI до фронта выборки сигнала SCK	2*TSYSCLK		нс
<b>ТСИН</b>	Длительность интервала между фронтом выборки SCK и обновлением данных на линии MOSI	2*TSYSCLK		нс
<b>ТСОН</b>	Длительность интервала между фронтом SCK, вызывающим сдвиг данных, и обновлением данных на линии MISO		4*TSYSCLK	нс
<b>ТСЛН</b>	Длительность интервала между последним фронтом SCK и обновлением данных на линии MISO (только для случая, когда СКРНА = 1)	6*TSYSCLK	8*TSYSCLK	нс
* TSYSCLK представляет собой один период системного тактового сигнала (SYSCLK).				

## 19. ТАЙМЕРЫ

Каждый МК содержит четыре таймера/счетчика (Т/С): два из них представляют собой 16-разрядные Т/С, совместимые с Т/С стандартной архитектуры 8051, а другие два являются 16-разрядными Т/С с режимом автоперезагрузки и предназначены для использования совместно с АЦП, SMBus, USB (для измерения фреймов), а также в качестве Т/С общего назначения. Эти Т/С можно использовать для измерения временных интервалов, подсчета внешних событий, а также для генерации периодических запросов прерываний. Таймер 0 и Таймер 1 почти идентичны и имеют четыре основных режима работы. Таймер 2 и Таймер 3 могут работать (каждый) как один 16-разрядный таймер или как два 8-разрядных таймера, причем во всех случаях поддерживается режим автоперезагрузки.

Режимы Таймера 0 и Таймера 1:	Режимы Таймера 2:	Режимы Таймера 3:
13-разрядный Т/С	16-разрядный Т/С с автоперезагрузкой	16-разрядный Т/С с автоперезагрузкой
16-разрядный Т/С		
8-разрядный Т/С с автоперезагрузкой	Два 8-разрядных Т/С с автоперезагрузкой	Два 8-разрядных Т/С с автоперезагрузкой
Два 8-разрядных Т/С (только Таймер 0)		

Таймеры 0 и 1 могут тактироваться от одного из пяти источников, выбор которых осуществляется с помощью битов выбора режима таймера (Т1М – Т0М) и битов выбора коэффициента деления тактовой частоты (SCA1 – SCA0). Биты выбора коэффициента деления тактовой частоты настраивают предварительный делитель тактовой частоты, сигнал с выхода которого может использоваться для тактирования Таймера 0 и/или Таймера 1 (см. рис. 19.6).

В качестве сигнала тактирования Таймеров 0 и 1 можно выбрать либо сигнал с выхода предварительного делителя тактовой частоты, либо системный тактовый сигнал. Таймер 2 и Таймер 3 могут тактироваться либо системным тактовым сигналом, либо системным тактовым сигналом, деленным по частоте на 12, либо сигналом от внешнего генератора тактовых импульсов, деленным по частоте на 8.

Таймер 0 и Таймер 1 могут также функционировать как счетчики. В этом случае регистр таймера/счетчика инкрементируется под воздействием каждого перехода внешнего сигнала на выбранном входном выводе (Т0 или Т1) из состояния лог. '1' в состояние лог. '0'. Могут подсчитываться импульсы с частотой до 1/4 системной тактовой частоты. Входной сигнал не обязательно должен быть периодическим, однако он должен удерживаться на заданном уровне как минимум в течение двух полных системных тактовых циклов, чтобы гарантировать его корректную выборку.

### 19.1. Таймер 0 и Таймер 1

Каждый таймер реализован в виде 16-разрядного регистра, доступного как два отдельных байта: младший байт (ТL0 или ТL1) и старший байт (ТН0 или ТН1). Регистр управления Т/С (ТCON) используется для включения Таймера 0 и Таймера 1, а также для индикации их состояния. Прерывания от Таймера 0 можно включить установкой в 1 бита ET0 в регистре IE (см. раздел 8.3.5 «Описания регистров прерываний» на стр.61); прерывания от Таймера 1 можно включить установкой в 1 бита ET1 в регистре IE (см. раздел 8.3.5). Оба таймера/счетчика работают в одном из четырех основных режимов, задаваемых битами выбора режима Т1М1-Т0М0 в регистре режима Т/С (ТMOD). Каждый Т/С может быть настроен независимо от другого. Ниже приведено описание каждого режима работы.

#### 19.1.1. Режим 0: 13-разрядный таймер/счетчик

В режиме 0 Таймеры 0 и 1 работают как 13-разрядный таймер/счетчик. Ниже приводится описание настройки и функционирования Таймера 0. Однако, оба таймера идентичны, и Таймер 1 настраивается точно так же, как и Таймер 0.

Регистр ТН0 содержит восемь старших бит 13-разрядного значения регистра Т/С. Регистр ТL0 содержит в разрядах ТL0.4-ТL0.0 пять младших бит 13-разрядного значения регистра Т/С. Три старших бита регистра ТL0 (ТL0.7-ТL0.5) не определены и должны маскироваться или игнорироваться при чтении регистра ТL0. При инкрементировании 13-разрядного таймера и переполнении его из состояния 0x1FFF (все единицы) в состояние 0x0000 устанавливается в 1 флаг переполнения таймера ТF0 (ТCON.5) и будет сгенерировано прерывание, если оно разрешено.

Бит С/Т0 (ТMOD.2) выбирает источник сигнала тактирования Т/С0. Если бит С/Т0 установлен в 1, то инкрементирование регистра таймера осуществляется под воздействием перехода внешнего сигнала на выбранном входном выводе (Т0) из состояния лог. 1 в состояние лог. 0. (Подробная информация о выборе и настройке внешних выводов приведена в разделе 14.1 «Приоритетный декодер матрицы» на стр.129.). Если бит

C/T0 сброшен в 0, то в качестве источника тактирования T/C0 будет использоваться сигнал, определяемый битом T0M (СКCON.3). Если бит T0M установлен в 1, то Таймер 0 тактируется системным тактовым сигналом. Если бит T0M сброшен в 0, то в качестве источника тактирования T/C0 будет использоваться сигнал, определяемый битами настройки предварительного делителя в регистре СКCON (см. рис. 19.6).

Установка в 1 бита TR0 (TCON.4) включит таймер, если либо бит GATE0 (TMOD.3) равен нулю, либо на внешнем выводе /INT0 присутствует сигнал с активным логическим уровнем, который определяется битом IN0PL в регистре INT01CF (см. рис.8.13). После установки в 1 бита GATE0 управление таймером передается внешнему сигналу /INT0 (см. раздел 8.3.5 «Описания регистров прерываний» на стр.61), что позволяет легко осуществлять измерение ширины импульсов.

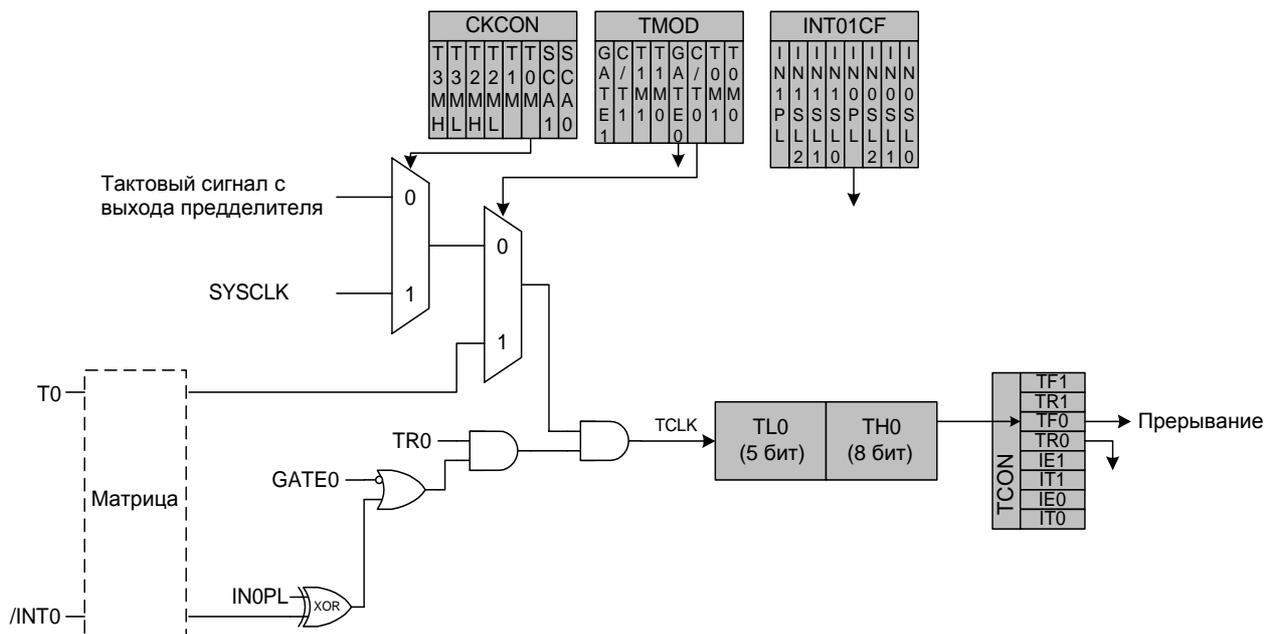
TR0	GATE0	/INT0	Таймер/Счетчик
0	X	X	Отключен
1	0	X	Включен
1	1	0	Отключен
1	1	1	Включен

X = не имеет значения

Установка TR0 не сбрасывает таймер. Регистры таймера следует инициализировать необходимыми значениями до включения таймера.

TL1 и TH1 образуют 13-разрядный регистр Таймера 1 точно так же, как описано выше для регистров TL0 и TH0. Для настройки Таймера 1 и управления им используются соответствующие биты регистров TCON и TMOD таким же образом, как и для Таймера 0. Входной сигнал /INT1 используется совместно с Таймером 1; полярность /INT1 определяется битом IN1PL в регистре INT01CF (см. рис.8.13).

Рисунок 19.1. Структурная схема Таймера 0 в режиме 0



### 19.1.2. Режим 1: 16-разрядный Таймер/Счетчик

Режим 1 аналогичен режиму 0 с тем лишь исключением, что регистры T/C используют все 16 бит. Таймеры/счетчики включаются и настраиваются в режиме 1 точно так же, как в режиме 0.



### 19.1.4. Режим 3: Два 8-разрядных таймера/счетчика (только Таймер 0)

В режиме 3 Таймер 0 функционирует как два отдельных 8-разрядных таймера/счетчика TL0 и TH0. Для управления таймером/счетчиком TL0 используются биты управления/состояния Таймера 0 (в регистрах TCON и TMOD): TR0, C/T0, GATE0 и TF0. В качестве источника тактирования TL0 может использовать либо системный тактовый сигнал, либо внешний входной сигнал. Таймер/счетчик TH0 может использовать для тактирования либо системный тактовый сигнал, либо сигнал с выхода предварительного делителя. Для включения таймера/счетчика TH0 используется управляющий бит запуска Таймера 1 (TR1). Таймер/счетчик TH0 при переполнении устанавливает флаг переполнения Таймера 1 TF1 и, таким образом, управляет прерыванием от Таймера 1.

В режиме 3 Таймер 1 неактивен. Если Таймер 0 функционирует в режиме 3, Таймер 1 может работать в режимах 0, 1 или 2, но не может тактироваться внешними сигналами, устанавливать флаг TF1 и генерировать прерывание. Однако, переполнение Таймера 1 можно использовать для генерации скорости передачи данных для модулей SMBus и/или УАПЧ, а также для запуска преобразований АЦП. Пока Таймер 0 функционирует в режиме 3, управление запуском Таймера 1 осуществляется путем настройки режима его работы. Чтобы запустить Таймер 1 в то время, когда Таймер 0 функционирует в режиме 3, необходимо установить для Таймера 1 режим работы 0, 1 или 2. Чтобы отключить Таймер 1, необходимо настроить его на работу в режиме 3.

Рисунок 19.3. Структурная схема Таймера 0 в режиме 3

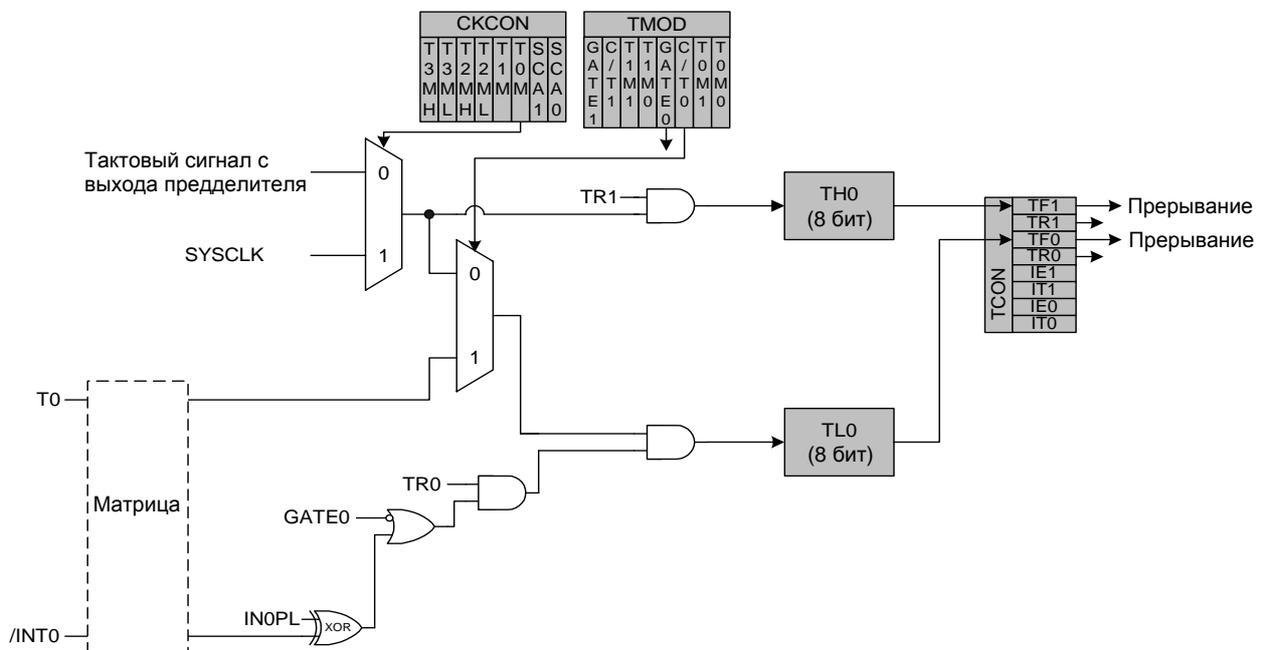


Рисунок 19.4. TCON: Регистр управления Таймерами 0 и 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x88
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	
<p><b>Бит 7: TF1: Флаг переполнения Таймера 1.</b> Устанавливается аппаратно при переполнении Таймера 1. Сбрасывается аппаратно при переходе к процедуре обслуживания прерывания от Таймера 1, но может быть сброшен и программно. 0: Переполнения Таймера 1 не обнаружено. 1: Таймер 1 переполнился.</p> <p><b>Бит 6: TR1: Управление запуском Таймера 1.</b> 0: Таймер 1 отключен. 1: Таймер 1 включен.</p> <p><b>Бит 5: TF0: Флаг переполнения Таймера 0.</b> Устанавливается аппаратно при переполнении Таймера 0. Сбрасывается аппаратно при переходе к процедуре обслуживания прерывания от Таймера 0, но может быть сброшен и программно. 0: Переполнения Таймера 0 не обнаружено. 1: Таймер 0 переполнился.</p> <p><b>Бит 4: TR0: Управление запуском Таймера 0.</b> 0: Таймер 0 отключен. 1: Таймер 0 включен.</p> <p><b>Бит 3: IE1: Внешнее прерывание 1.</b> Этот флаг аппаратно устанавливается в 1 при обнаружении активного фронта/уровня (определяется битом IT1) внешнего сигнала. Может быть сброшен программно, но при переходе к процедуре обслуживания внешнего прерывания 1 сбрасывается аппаратно, если IT1=1. При IT1=0 этот флаг устанавливается в 1, если на внешнем выводе /INT1 присутствует сигнал с активным логическим уровнем, который определяется битом IN1PL в регистре INT01CF (см. рис.8.13).</p> <p><b>Бит 2: IT1: Выбор типа внешнего прерывания 1.</b> Этот бит определяет, какое событие будет вызывать внешнее прерывание 1: фронт или активный уровень внешнего сигнала /INT1 (активный уровень внешнего сигнала /INT1 определяется битом IN1PL в регистре INT01CF (см. рис.8.13)). 0: Внешнее прерывание 1 вызывается активным уровнем сигнала /INT1. 1: Внешнее прерывание 1 вызывается фронтом сигнала /INT1.</p> <p><b>Бит 1: IE0: Внешнее прерывание 0.</b> Этот флаг аппаратно устанавливается в 1 при обнаружении активного фронта/уровня (определяется битом IT0) внешнего сигнала. Может быть сброшен программно, но при переходе к процедуре обслуживания внешнего прерывания 0 сбрасывается аппаратно, если IT0=1. При IT0=0 этот флаг устанавливается в 1, если на внешнем выводе /INT0 присутствует сигнал с активным логическим уровнем, который определяется битом IN0PL в регистре INT01CF (см. рис.8.13).</p> <p><b>Бит 0: IT0: Выбор типа внешнего прерывания 0.</b> Этот бит определяет, какое событие будет вызывать внешнее прерывание 0: фронт или активный уровень внешнего сигнала /INT0 (активный уровень внешнего сигнала /INT0 определяется битом IN0PL в регистре INT01CF (см. рис.8.13)). 0: Внешнее прерывание 0 вызывается активным уровнем сигнала /INT0. 1: Внешнее прерывание 0 вызывается фронтом сигнала /INT0.</p>								

Рисунок 19.5. TMOD: Регистр режима Таймеров 0 и 1

R/W	Значение при сбросе: 00000000 SFR Адрес: 0x89							
GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	

Бит 7: GATE1: Управление блокировкой Таймера 1.  
 0: Таймер 1 включен, если TR1 = 1, независимо от логического уровня на входе /INT1.  
 1: Таймер 1 включен только тогда, когда TR1 = 1 и на входе /INT1 активный логический уровень, определяется битом IN1PL в регистре INT01CF (см. рис.8.13).

Бит 6: C/T1: Выбор режима таймера или счетчика для T/C1.  
 0: T/C1 работает как таймер: Таймер 1 инкрементируется от внутреннего сигнала тактирования, который задается битом T1M (СКCON.4).  
 1: T/C1 работает как счетчик: Таймер 1 инкрементируется под воздействием перехода из 1 в 0 внешнего входного сигнала (T1).

Биты 5-4: T1M1-T1M0: Выбор режима работы Таймера 1.  
 Эти биты определяют режим работы Таймера 1.

T1M1	T1M0	Режим
0	0	Режим 0: 13-разрядный таймер/счетчик
0	1	Режим 1: 16-разрядный таймер/счетчик
1	0	Режим 2: 8-разрядный таймер/счетчик с автоперезагрузкой
1	1	Режим 3: Таймер 1 неактивен

Бит 3: GATE0: Управление блокировкой Таймера 0.  
 0: Таймер 0 включен, если TR0 = 1, независимо от логического уровня на входе /INT0.  
 1: Таймер 0 включен только тогда, когда TR0 = 1 и на входе /INT0 активный логический уровень, определяется битом IN0PL в регистре INT01CF (см. рис.8.13).

Бит 2: C/T0: Выбор режима таймера или счетчика для T/C0.  
 0: T/C0 работает как таймер: Таймер 0 инкрементируется от внутреннего сигнала тактирования, который задается битом T0M (СКCON.3).  
 1: T/C0 работает как счетчик: Таймер 0 инкрементируется под воздействием перехода из 1 в 0 внешнего входного сигнала (T0).

Биты 1-0: T0M1-T0M0: Выбор режима работы Таймера 0.  
 Эти биты определяют режим работы Таймера 0.

T0M1	T0M0	Режим
0	0	Режим 0: 13-разрядный таймер/счетчик
0	1	Режим 1: 16-разрядный таймер/счетчик
1	0	Режим 2: 8-разрядный таймер/счетчик с автоперезагрузкой
1	1	Режим 3: Два 8-разрядных таймера/счетчика

Рисунок 19.6. СКCON: Регистр управления тактированием

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x8E															
T3MH	T3ML	T2MH	T2ML	T1M	T0M	SCA1	SCA0																
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0																
<p><b>Бит 7:</b> T3MH: Выбор источника тактирования для старшего байта Таймера 3. Этот бит определяет, какой тактовый сигнал будет подаваться на старший байт Таймера 3, если Таймер 3 функционирует в раздельном 8-разрядном режиме. Бит T3MH игнорируется, если Таймер 3 функционирует в любом другом режиме. 0: Для тактирования старшего байта Таймера 3 используется сигнал, определяемый битом T3XCLK в регистре TMR3CN. 1: Старший байт Таймера 3 тактируется системным тактовым сигналом.</p>																							
<p><b>Бит 6:</b> T3ML: Выбор источника тактирования для младшего байта Таймера 3. Этот бит определяет, какой тактовый сигнал будет подаваться на Таймер 3. Если Таймер 3 функционирует в раздельном 8-разрядном режиме, то бит T3ML определяет, какой тактовый сигнал будет подаваться на младший 8-разрядный таймер. 0: Для тактирования младшего байта Таймера 3 используется сигнал, определяемый битом T3XCLK в регистре TMR3CN. 1: Младший байт Таймера 3 тактируется системным тактовым сигналом.</p>																							
<p><b>Бит 5:</b> T2MH: Выбор источника тактирования для старшего байта Таймера 2. Этот бит определяет, какой тактовый сигнал будет подаваться на старший байт Таймера 2, если Таймер 2 функционирует в раздельном 8-разрядном режиме. Бит T2MH игнорируется, если Таймер 2 функционирует в любом другом режиме. 0: Для тактирования старшего байта Таймера 2 используется сигнал, определяемый битом T2XCLK в регистре TMR2CN. 1: Старший байт Таймера 2 тактируется системным тактовым сигналом.</p>																							
<p><b>Бит 4:</b> T2ML: Выбор источника тактирования для младшего байта Таймера 2. Этот бит определяет, какой тактовый сигнал будет подаваться на Таймер 2. Если Таймер 2 функционирует в раздельном 8-разрядном режиме, то бит T2ML определяет, какой тактовый сигнал будет подаваться на младший 8-разрядный таймер. 0: Для тактирования младшего байта Таймера 2 используется сигнал, определяемый битом T2XCLK в регистре TMR2CN. 1: Младший байт Таймера 2 тактируется системным тактовым сигналом.</p>																							
<p><b>Бит 3:</b> T1M: Выбор источника тактирования для Таймера 1. Этот бит определяет, какой тактовый сигнал будет подаваться на Таймер 1. Бит T1M игнорируется, если <math>C/T1 = 1</math>. 0: Для тактирования Таймера 1 используется сигнал, определяемый битами настройки предварительного делителя (SCA1 – SCA0). 1: Таймер 1 тактируется системным тактовым сигналом.</p>																							
<p><b>Бит 2:</b> T0M: Выбор источника тактирования для Таймера 0. Этот бит определяет, какой тактовый сигнал будет подаваться на Таймер 0. Бит T0M игнорируется, если <math>C/T0 = 1</math>. 0: Для тактирования Таймера 0 используется сигнал, определяемый битами настройки предварительного делителя (SCA1 – SCA0). 1: Таймер 0 тактируется системным тактовым сигналом.</p>																							
<p><b>Биты 1-0:</b> SCA1–SCA0: Биты выбора коэффициента деления для частоты тактирования Таймеров 0 и 1. Эти биты управляют делением частоты сигнала тактирования, подаваемого на Таймер 0 и/или Таймер 1, если они настроены на использование предварительного делителя.</p>																							
<table border="1"> <thead> <tr> <th>SCA1</th> <th>SCA0</th> <th>Тактовый сигнал</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>SYSCLK/12</td> </tr> <tr> <td>0</td> <td>1</td> <td>SYSCLK/4</td> </tr> <tr> <td>1</td> <td>0</td> <td>SYSCLK/48</td> </tr> <tr> <td>1</td> <td>1</td> <td>EXTCLK/8</td> </tr> </tbody> </table>									SCA1	SCA0	Тактовый сигнал	0	0	SYSCLK/12	0	1	SYSCLK/4	1	0	SYSCLK/48	1	1	EXTCLK/8
SCA1	SCA0	Тактовый сигнал																					
0	0	SYSCLK/12																					
0	1	SYSCLK/4																					
1	0	SYSCLK/48																					
1	1	EXTCLK/8																					
Примечание: Сигнал EXTCLK/8 синхронизирован с SYSCLK.																							

**Рисунок 19.7. TL0: Младший байт Таймера 0**

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x8A

Биты 7-0: TL0: Младший байт Таймера 0.  
Регистр TL0 является младшим байтом 16-разрядного Таймера 0.

**Рисунок 19.8. TL1: Младший байт Таймера 1**

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x8B

Биты 7-0: TL1: Младший байт Таймера 1.  
Регистр TL1 является младшим байтом 16-разрядного Таймера 1.

**Рисунок 19.9. TH0: Старший байт Таймера 0**

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x8C

Биты 7-0: TH0: Старший байт Таймера 0.  
Регистр TH0 является старшим байтом 16-разрядного Таймера 0.

**Рисунок 19.10. TH1: Старший байт Таймера 1**

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x8D

Биты 7-0: TH1: Старший байт Таймера 1.  
Регистр TH1 является старшим байтом 16-разрядного Таймера 1.

## 19.2. Таймер 2

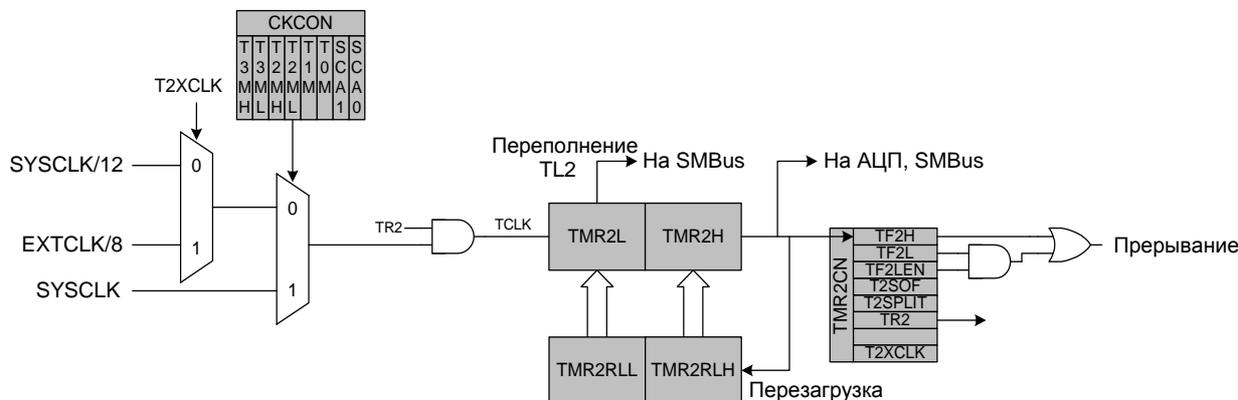
Таймер 2 представляет собой 16-разрядный таймер/счетчик, образованный двумя SFR регистрами: TMR2L (младший байт) и TMR2H (старший байт). Таймер 2 может работать в 16-разрядном режиме с автоперезагрузкой, в раздельном режиме (8-разрядный с автоперезагрузкой) или в режиме захвата признаков Start-Of-Frame (SOF) на шине USB. Режим работы Таймера 2 определяется битами T2SPLIT (TMR2CN.3) и T2SOF (TMR2CN.4).

Таймер 2 может тактироваться либо системным тактовым сигналом, либо системным тактовым сигналом, деленным по частоте на 12, либо сигналом от внешнего источника, деленный по частоте на 8. Режим тактирования от внешнего источника идеален для реализации функций реального времени, когда внутренний генератор генерирует системный тактовый сигнал в то время, как Таймер 2 (и/или ПМС) тактируется внешним прецизионным генератором. Следует иметь в виду, что сигнал от внешнего генератора, деленный по частоте на 8, синхронизируется с системным тактовым сигналом.

### 19.2.1. 16-разрядный таймер с автоперезагрузкой

Если T2SPLIT = 0 и T2SOF = 0, то Таймер 2 функционирует как 16-разрядный таймер с автоперезагрузкой. Для тактирования Таймера 2 можно использовать сигналы SYSCLK, SYSCLK/12 или EXTCLK/8. Таймер 2 считает в прямом направлении. При переполнении Таймера 2 из состояния 0xFFFF в состояние 0x0000 16-разрядное значение регистров перезагрузки Таймера 2 (TMR2RLH:TMR2RLL) загружается в регистр Таймера 2, как показано на рис.19.11, и устанавливается в 1 флаг переполнения старшего байта Таймера 2 (TMR2CN.7). Если прерывания от Таймера 2 разрешены (IE.5 = 1), то прерывание будет генерироваться при каждом переполнении Таймера 2. Кроме этого, если прерывания от Таймера 2 разрешены и бит TF2LEN (TMR2CN.5) установлен в 1, то прерывание будет генерироваться при каждом переполнении младших 8 бит (регистр TMR2L) из состояния 0xFF в состояние 0x00.

Рисунок 19.11. Структурная схема Таймера 2 в 16-разрядном режиме



### 19.2.2. 8-разрядные таймеры с автоперезагрузкой

Если  $T2SPLIT = 1$  и  $T2SOF = 0$ , то Таймер 2 функционирует как два 8-разрядных таймера (TMR2H и TMR2L). Оба 8-разрядных таймера функционируют в режиме с автоперезагрузкой, как показано на рис.19.12. TMR2RLL содержит значение перезагрузки для TMR2L; TMR2RLH содержит значение перезагрузки для TMR2H. Бит TR2 в регистре TMR2CN управляет запуском TMR2H; TMR2L, работающий в 8-разрядном режиме, запущен всегда.

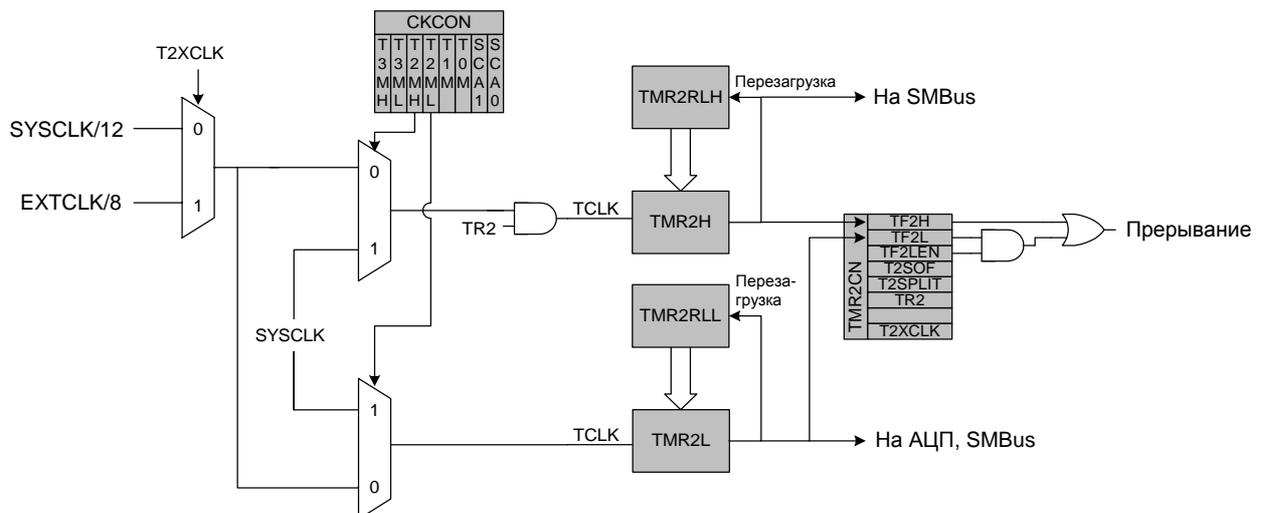
Для тактирования каждого 8-разрядного таймера можно использовать сигналы SYSCLK, SYSCLK/12 или EXTCLK/8. Биты выбора источника тактирования Таймера 2 (T2MH и T2ML в регистре CKCON) выбирают либо SYSCLK, либо сигнал тактирования, определяемый битом выбора внешнего источника тактирования Таймера 2 (T2XCLK в регистре TMR2CN) следующим образом:

T2MH	T2XCLK	Источник тактирования TMR2H
0	0	SYSCLK/12
0	1	EXTCLK/8
1	X	SYSCLK

T2ML	T2XCLK	Источник тактирования TMR2L
0	0	SYSCLK/12
0	1	EXTCLK/8
1	X	SYSCLK

Если TMR2H переполняется из 0xFF в 0x00, то бит TF2H устанавливается в 1; если TMR2L переполняется из 0xFF в 0x00, то бит TF2L устанавливается в 1. Если прерывания от Таймера 2 разрешены (IE.5), то прерывание будет генерироваться при каждом переполнении TMR2H. Если прерывания от Таймера 2 разрешены и бит TF2LEN (TMR2CN.5) установлен в 1, то прерывание будет генерироваться каждый раз при переполнении либо TMR2L, либо TMR2H. Когда TF2LEN разрешено, то программа должна проверять флаги TF2H и TF2L, чтобы определить источник прерывания от Таймера 2. Флаги прерывания TF2H и TF2L не сбрасываются аппаратно и должны сбрасываться программно.

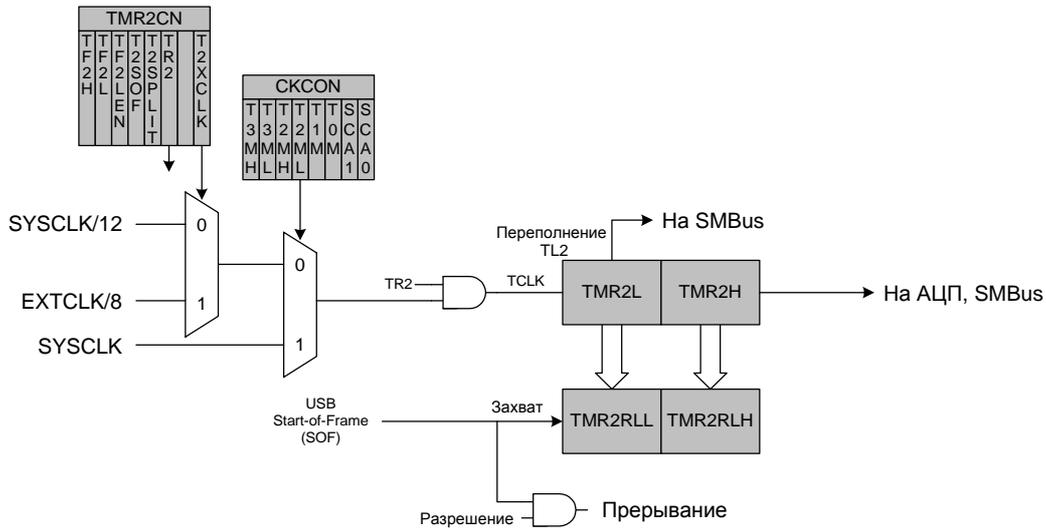
Рисунок 19.12. Структурная схема Таймера 2 в 8-разрядном режиме



## 19.2.3. Режим захвата USB Start-Of-Frame

Если  $T2SOF = 1$ , то Таймер 2 функционирует в режиме захвата признаков Start-Of-Frame (SOF) шины USB. Если  $T2SPLIT = 0$ , то Таймер 2 считает в прямом направлении и переполняется из состояния 0xFFFF в состояние 0x0000. Каждый раз при получении признака SOF USB содержимое регистров Таймера 2 (TMR2H:TMR2L) «защелкивается» в регистры перезагрузки Таймера 2 (TMR2RLH:TMR2RLL). При этом генерируется прерывание от Таймера 2, если оно разрешено. Этот режим можно использовать для калибровки системного или внешнего генератора, используя признаки SOF, передаваемые хостом по шине USB.

**Рисунок 19.13. Таймер 2 в режиме захвата SOF ( $T2SPLIT = '0'$ )**



Если  $T2SOF = 1$ , то регистры Таймера 2 (TMR2H и TMR2L) функционируют как два 8-разрядных счетчика. Каждый счетчик считает в прямом направлении независимо от другого и переполняется из состояния 0xFF в состояние 0x00. Каждый раз при получении признака SOF USB содержимое регистров Таймера 2 (TMR2H:TMR2L) «защелкивается» в регистры перезагрузки Таймера 2 (TMR2RLH:TMR2RLL). При этом генерируется прерывание от Таймера 2, если оно разрешено.

**Рисунок 19.14. Таймер 2 в режиме захвата SOF ( $T2SPLIT = '1'$ )**

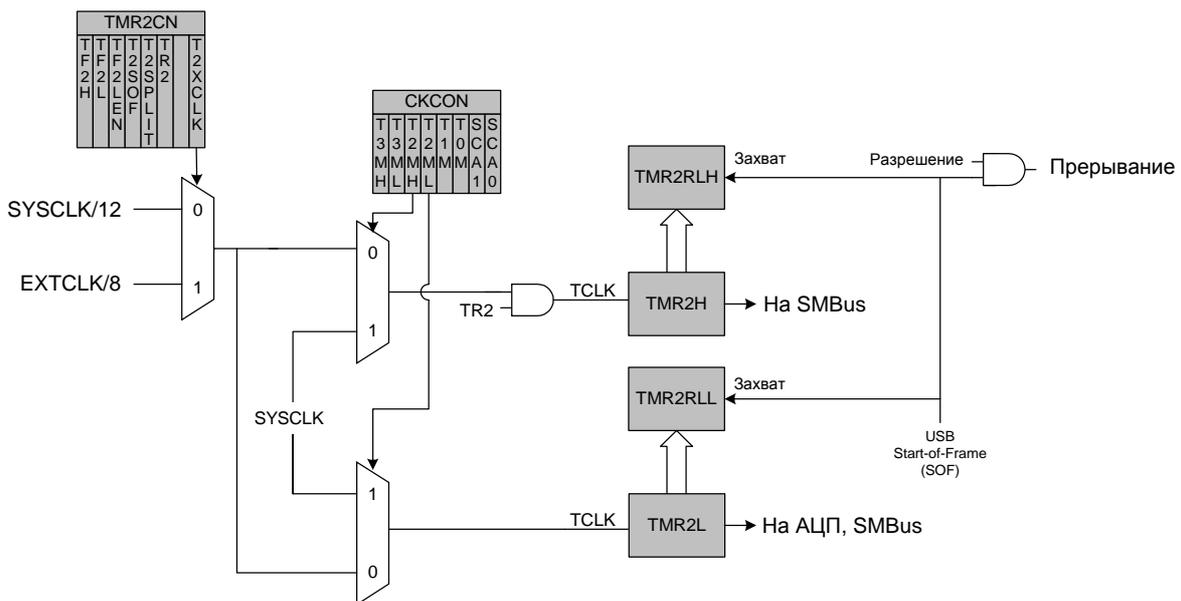


Рисунок 19.15. TMR2CN: Регистр управления Таймера 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
TF2H	TF2L	TF2LEN	T2SOF	T2SPLIT	TR2	-	T2XCLK	SFR Адрес: 0xC8
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 (доступен в битовом режиме адресации)	

**Бит 7: TF2H:** Флаг переполнения старшего байта Таймера 2.  
Этот бит аппаратно устанавливается в 1 при переполнении старшего байта Таймера 2 из состояния 0xFF в состояние 0x00. В 16-разрядном режиме это будет происходить при переполнении Таймера 2 из состояния 0xFFFF в состояние 0x0000. Если прерывание от Таймера 2 разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от Таймера 2. Этот бит не сбрасывается аппаратно, он должен быть сброшен программно.

**Бит 6: TF2L:** Флаг переполнения младшего байта Таймера 2.  
Этот бит аппаратно устанавливается в 1 при переполнении младшего байта Таймера 2 из состояния 0xFF в состояние 0x00. При установке этого бита прерывание будет генерироваться в том случае, если TF2LEN = 1 и прерывания от Таймера 2 разрешены. Бит TF2L устанавливается в 1 при переполнении младшего байта независимо от режима работы Таймера 2. Этот бит не сбрасывается аппаратно.

**Бит 5: TF2LEN:** Флаг разрешения прерывания от переполнения младшего байта Таймера 2.  
Этот бит разрешает/запрещает прерывания от переполнения младшего байта Таймера 2. Если TF2LEN = 1 и прерывания от Таймера 2 разрешены, то при переполнении младшего байта Таймера 2 будет генерироваться прерывание.  
0: Прерывания от переполнения младшего байта Таймера 2 запрещены.  
1: Прерывания от переполнения младшего байта Таймера 2 разрешены.

**Бит 4: T2SOF:** Разрешение захвата признаков SOF Таймером 2.  
0: Захват SOF запрещен.  
1: Захват SOF разрешен. Каждый раз при получении признака SOF USB содержимое регистров Таймера 2 (TMR2H:TMR2L) «защелкивается» в регистры перезагрузки Таймера 2 (TMR2RLH:TMR2RLL). При этом генерируется прерывание от Таймера 2, если оно разрешено.

**Бит 3: T2SPLIT:** Разрешение раздельного режима Таймера 2.  
Если этот бит установлен в 1, то Таймер 2 функционирует как два 8-разрядных таймера с автоперезагрузкой.  
0: Таймер 2 функционирует в 16-разрядном режиме с автоперезагрузкой.  
1: Таймер 2 функционирует как два 8-разрядных таймера с автоперезагрузкой.

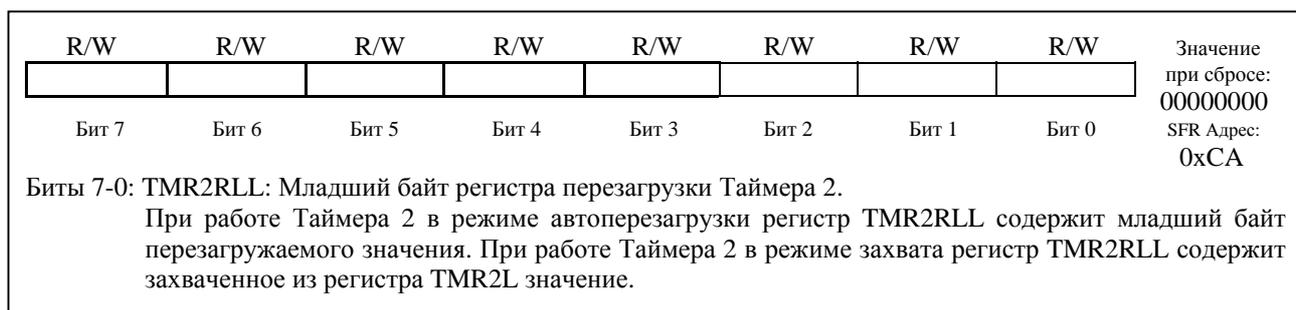
**Бит 2: TR2:** Бит управления запуском Таймера 2.  
Этот бит включает/отключает Таймер 2. В 8-разрядном режиме этот бит включает/отключает только TMR2H; TMR2L в этом режиме включен всегда.  
0: Таймер 2 отключен.  
1: Таймер 2 включен.

**Бит 1:** Не используется. Читается как 0b. Запись не оказывает никакого влияния.

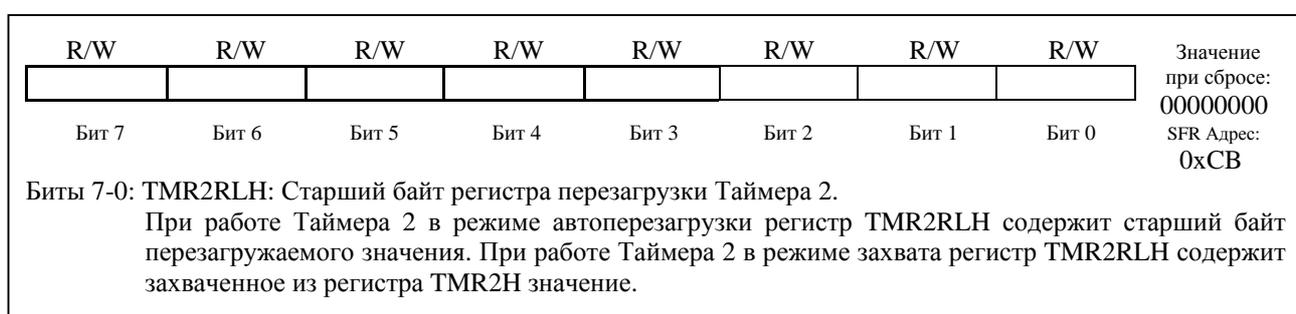
**Бит 0: T2XCLK:** Выбор внешнего источника тактирования для Таймера 2.  
Этот бит выбирает внешний источник тактирования для Таймера 2. В 8-разрядном режиме этот бит выбирает внешний источник тактирования для обоих байт таймера. Однако, все равно можно использовать биты выбора источника тактирования Таймера 2 (TM2H и TM2L в регистре СКCON) для выбора между внешним источником тактирования и системным тактовым сигналом (для любого таймера).  
0: В качестве внешнего источника тактирования для Таймера 2 выбран SYSCLK/12.  
1: В качестве внешнего источника тактирования для Таймера 2 выбран EXTCLK/8. Следует иметь в виду, что EXTCLK/8 синхронизируется с SYSCLK.



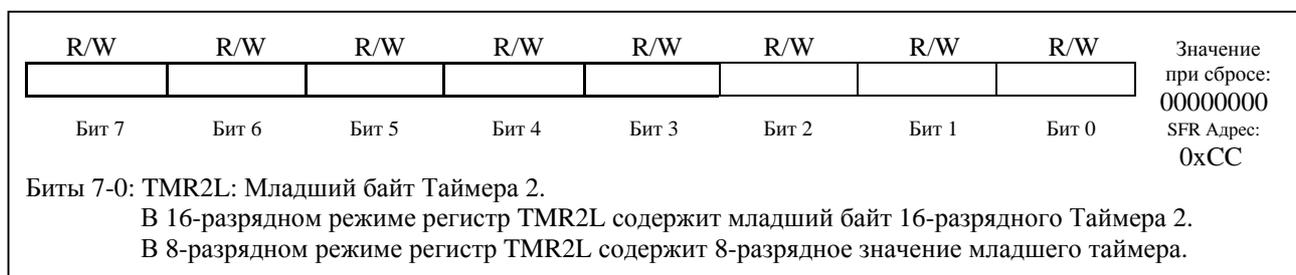
**Рисунок 19.16. TMR2RLL: Младший байт регистра перезагрузки Таймера 2**



**Рисунок 19.17. TMR2RLH: Старший байт регистра перезагрузки Таймера 2**



**Рисунок 19.18. TMR2L: Младший байт Таймера 2**



**Рисунок 19.19. TMR2H: Старший байт Таймера 2**



### 19.3. Таймер 3

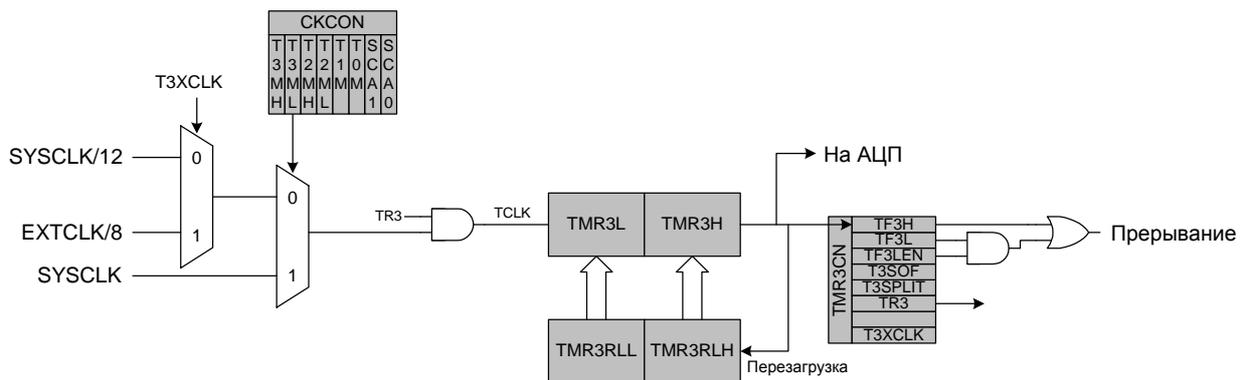
Таймер 3 представляет собой 16-разрядный таймер/счетчик, образованный двумя SFR регистрами: TMR3L (младший байт) и TMR3H (старший байт). Таймер 3 может работать в 16-разрядном режиме с автоперезагрузкой, в раздельном режиме (8-разрядный с автоперезагрузкой) или в режиме захвата признаков Start-Of-Frame (SOF) на шине USB. Режим работы Таймера 3 определяется битами T3SPLIT (TMR3CN.3) и T3SOF (TMR3CN.4).

Таймер 3 может тактироваться либо системным тактовым сигналом, либо системным тактовым сигналом, деленным по частоте на 12, либо сигналом от внешнего источника, деленный по частоте на 8. Режим тактирования от внешнего источника идеален для реализации функций реального времени, когда внутренний генератор генерирует системный тактовый сигнал в то время, как Таймер 3 (и/или ПМС) тактируется внешним прецизионным генератором. Следует иметь в виду, что сигнал от внешнего генератора, деленный по частоте на 8, синхронизируется с системным тактовым сигналом.

#### 19.3.1. 16-разрядный таймер с автоперезагрузкой

Если T3SPLIT (TMR3CN.3) = 0, то Таймер 3 функционирует как 16-разрядный таймер с автоперезагрузкой. Для тактирования Таймера 3 можно использовать сигналы SYSCLK, SYSCLK/12 или EXTCLK/8. Таймер 3 считает в прямом направлении. При переполнении Таймера 3 из состояния 0xFFFF в состояние 0x0000 16-разрядное значение регистров перезагрузки Таймера 3 (TMR3RLH:TMR3RLL) загружается в регистр Таймера 3, как показано на рис.19.20, и устанавливается в 1 флаг переполнения старшего байта Таймера 3 (TMR3CN.7). Если прерывания от Таймера 3 разрешены (IE.5 = 1), то прерывание будет генерироваться при каждом переполнении Таймера 3. Кроме этого, если прерывания от Таймера 3 разрешены и бит TF3LEN (TMR3CN.5) установлен в 1, то прерывание будет генерироваться при каждом переполнении младших 8 бит (регистр TMR3L) из состояния 0xFF в состояние 0x00.

Рисунок 19.20. Структурная схема Таймера 3 в 16-разрядном режиме



## 19.3.2. 8-разрядные таймеры с автоперезагрузкой

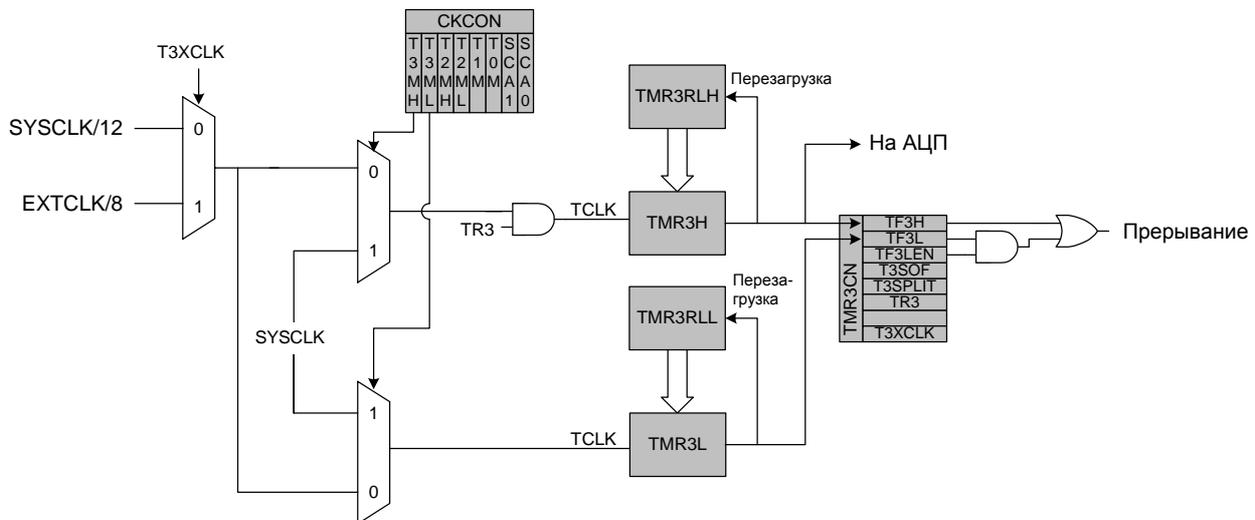
Если T3SPLIT = 1, то Таймер 3 функционирует как два 8-разрядных таймера (TMR3H и TMR3L). Оба 8-разрядных таймера функционируют в режиме с автоперезагрузкой, как показано на рис.19.21. TMR3RLL содержит значение перезагрузки для TMR3L; TMR3RLH содержит значение перезагрузки для TMR3H. Бит TR3 в регистре TMR3CN управляет запуском TMR3H; TMR3L, работающий в 8-разрядном режиме, запущен всегда.

Для тактирования каждого 8-разрядного таймера можно использовать сигналы SYSCLK, SYSCLK/12 или EXTCLK/8. Биты выбора источника тактирования Таймера 3 (T3MH и T3ML в регистре CKCON) выбирают либо SYSCLK, либо сигнал тактирования, определяемый битом выбора внешнего источника тактирования Таймера 3 (T3XCLK в регистре TMR3CN) следующим образом:

T3MH	T3XCLK	Источник тактирования TMR3H	T3ML	T3XCLK	Источник тактирования TMR3L
0	0	SYSCLK/12	0	0	SYSCLK/12
0	1	EXTCLK/8	0	1	EXTCLK/8
1	X	SYSCLK	1	X	SYSCLK

Если TMR3H переполняется из 0xFF в 0x00, то бит TF3H устанавливается в 1; если TMR3L переполняется из 0xFF в 0x00, то бит TF3L устанавливается в 1. Если прерывания от Таймера 3 разрешены (IE.5), то прерывание будет генерироваться при каждом переполнении TMR3H. Если прерывания от Таймера 3 разрешены и бит TF3LEN (TMR3CN.5) установлен в 1, то прерывание будет генерироваться каждый раз при переполнении либо TMR3L, либо TMR3H. Когда TF3LEN разрешено, то программа должна проверять флаги TF3H и TF3L, чтобы определить источник прерывания от Таймера 3. Флаги прерывания TF3H и TF3L не сбрасываются аппаратно и должны сбрасываться программно.

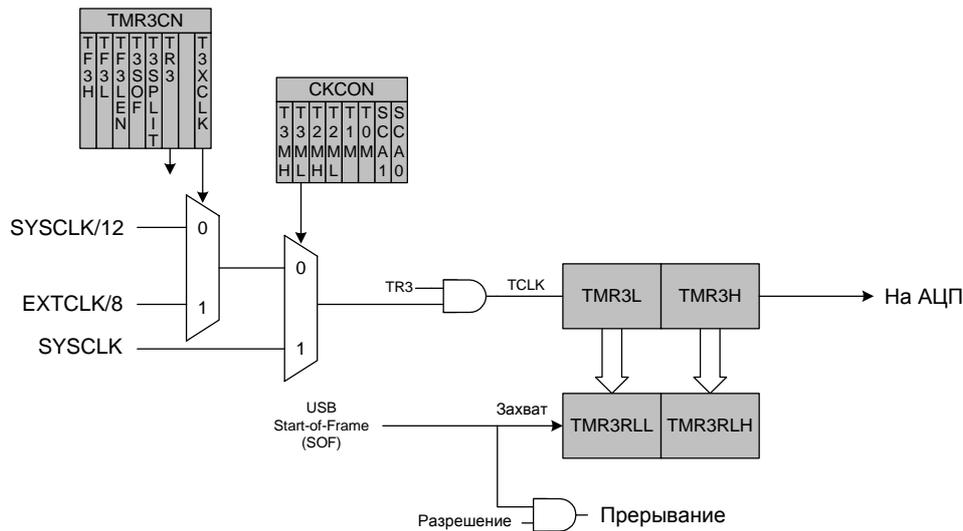
Рисунок 19.21. Структурная схема Таймера 3 в 8-разрядном режиме



### 19.3.3. Режим захвата USB Start-Of-Frame

Если  $T3SOF = 1$ , то Таймер 3 функционирует в режиме захвата признаков Start-Of-Frame (SOF) шины USB. Если  $T3SPLIT = 0$ , то Таймер 3 считает в прямом направлении и переполняется из состояния  $0xFFFF$  в состояние  $0x0000$ . Каждый раз при получении признака SOF USB содержимое регистров Таймера 3 (TMR3H:TMR3L) «защелкивается» в регистры перезагрузки Таймера 3 (TMR3RLH:TMR3RLL). При этом генерируется прерывание от Таймера 3, если оно разрешено. Этот режим можно использовать для калибровки системного или внешнего генератора, используя признаки SOF, передаваемые хостом по шине USB.

Рисунок 19.22. Таймер 3 в режиме захвата SOF ( $T3SPLIT = '0'$ )



Если  $T3SOF = 1$ , то регистры Таймера 3 (TMR3H и TMR3L) функционируют как два 8-разрядных счетчика. Каждый счетчик считает в прямом направлении независимо от другого и переполняется из состояния  $0xFF$  в состояние  $0x00$ . Каждый раз при получении признака SOF USB содержимое регистров Таймера 3 (TMR3H:TMR3L) «защелкивается» в регистры перезагрузки Таймера 3 (TMR3RLH:TMR3RLL). При этом генерируется прерывание от Таймера 3, если оно разрешено.

Рисунок 19.23. Таймер 3 в режиме захвата SOF ( $T3SPLIT = '1'$ )

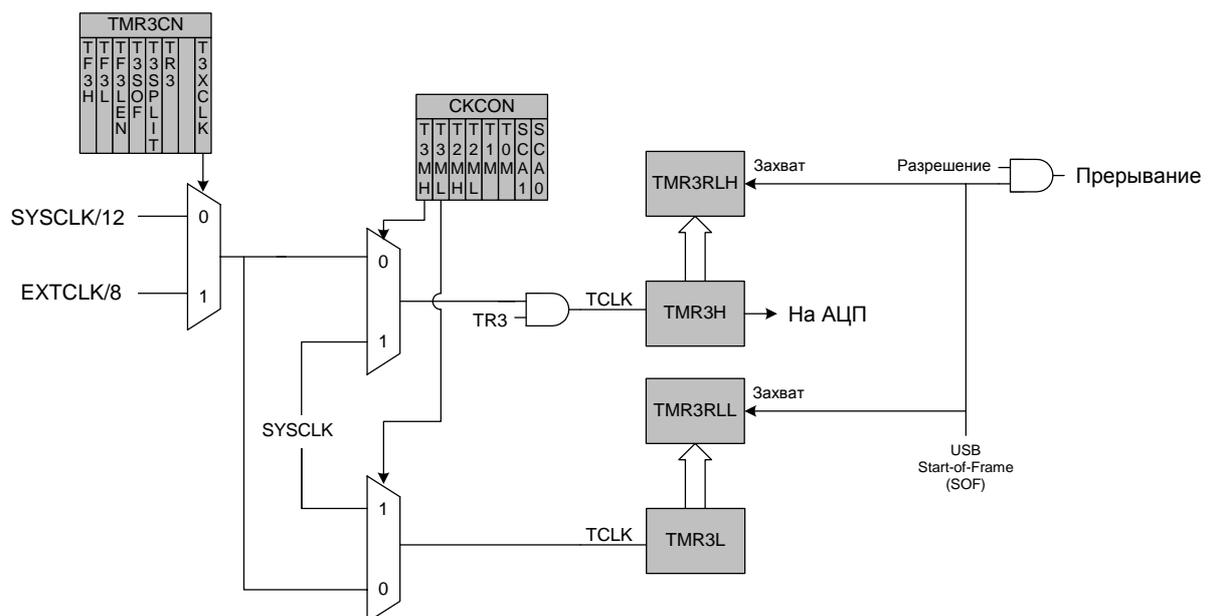


Рисунок 19.24. TMR3CN: Регистр управления Таймера 3

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0x91
TF3H	TF3L	TF3LEN	T3SOF	T3SPLIT	TR3	-	T3XCLK	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p><b>Бит 7: TF3H:</b> Флаг переполнения старшего байта Таймера 3. Этот бит аппаратно устанавливается в 1 при переполнении старшего байта Таймера 3 из состояния 0xFF в состояние 0x00. В 16-разрядном режиме это будет происходить при переполнении Таймера 3 из состояния 0xFFFF в состояние 0x0000. Если прерывание от Таймера 3 разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от Таймера 3. Этот бит не сбрасывается аппаратно, он должен быть сброшен программно.</p>								
<p><b>Бит 6: TF3L:</b> Флаг переполнения младшего байта Таймера 3. Этот бит аппаратно устанавливается в 1 при переполнении младшего байта Таймера 3 из состояния 0xFF в состояние 0x00. При установке этого бита прерывание будет генерироваться в том случае, если TF3LEN = 1 и прерывания от Таймера 3 разрешены. Бит TF3L устанавливается в 1 при переполнении младшего байта независимо от режима работы Таймера 3. Этот бит не сбрасывается аппаратно.</p>								
<p><b>Бит 5: TF3LEN:</b> Флаг разрешения прерывания от переполнения младшего байта Таймера 3. Этот бит разрешает/запрещает прерывания от переполнения младшего байта Таймера 3. Если TF3LEN = 1 и прерывания от Таймера 3 разрешены, то при переполнении младшего байта Таймера 3 будет генерироваться прерывание. 0: Прерывания от переполнения младшего байта Таймера 3 запрещены. 1: Прерывания от переполнения младшего байта Таймера 3 разрешены.</p>								
<p><b>Бит 4: T3SOF:</b> Разрешение захвата признаков SOF Таймером 3. 0: Захват SOF запрещен. 1: Захват SOF разрешен. Каждый раз при получении признака SOF USB содержимое регистров Таймера 3 (TMR3H:TMR3L) «зашелкивается» в регистры перезагрузки Таймера 3 (TMR3RLH:TMR3RLL). При этом генерируется прерывание от Таймера 3, если оно разрешено.</p>								
<p><b>Бит 3: T3SPLIT:</b> Разрешение раздельного режима Таймера 3. Если этот бит установлен в 1, то Таймер 3 функционирует как два 8-разрядных таймера с автоперезагрузкой. 0: Таймер 3 функционирует в 16-разрядном режиме с автоперезагрузкой. 1: Таймер 3 функционирует как два 8-разрядных таймера с автоперезагрузкой.</p>								
<p><b>Бит 2: TR3:</b> Бит управления запуском Таймера 3. Этот бит включает/отключает Таймер 3. В 8-разрядном режиме этот бит включает/отключает только TMR3H; TMR3L в этом режиме включен всегда. 0: Таймер 3 отключен. 1: Таймер 3 включен.</p>								
<p><b>Бит 1:</b> Не используется. Читается как 0b. Запись не оказывает никакого влияния.</p>								
<p><b>Бит 0: T3XCLK:</b> Выбор внешнего источника тактирования для Таймера 3. Этот бит выбирает внешний источник тактирования для Таймера 3. В 8-разрядном режиме этот бит выбирает внешний источник тактирования для обоих байт таймера. Однако, все равно можно использовать биты выбора источника тактирования Таймера 3 (TM3H и TM3L в регистре CKCON) для выбора между внешним источником тактирования и системным тактовым сигналом (для любого таймера). 0: В качестве внешнего источника тактирования для Таймера 3 выбран SYSCLK/12. 1: В качестве внешнего источника тактирования для Таймера 3 выбран EXTCLK/8. Следует иметь в виду, что EXTCLK/8 синхронизируется с SYSCLK.</p>								

Рисунок 19.25. TMR3RLL: Младший байт регистра перезагрузки Таймера 3

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x92

Биты 7-0: TMR3RLL: Младший байт регистра перезагрузки Таймера 3.  
При работе Таймера 3 в режиме автоперезагрузки регистр TMR3RLL содержит младший байт перезагружаемого значения. При работе Таймера 3 в режиме захвата регистр TMR3RLL содержит захваченное из регистра TMR3L значение.

Рисунок 19.26. TMR3RLH: Старший байт регистра перезагрузки Таймера 3

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x93

Биты 7-0: TMR3RLH: Старший байт регистра перезагрузки Таймера 3.  
При работе Таймера 3 в режиме автоперезагрузки регистр TMR3RLH содержит старший байт перезагружаемого значения. При работе Таймера 3 в режиме захвата регистр TMR3RLH содержит захваченное из регистра TMR3H значение.

Рисунок 19.27. TMR3L: Младший байт Таймера 3

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x94

Биты 7-0: TMR3L: Младший байт Таймера 3.  
В 16-разрядном режиме регистр TMR3L содержит младший байт 16-разрядного Таймера 3.  
В 8-разрядном режиме регистр TMR3L содержит 8-разрядное значение младшего таймера.

Рисунок 19.28. TMR2H: Старший байт Таймера 3

R/W	Значение при сбросе:							
								00000000
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	SFR Адрес: 0x95

Биты 7-0: TMR2H: Старший байт Таймера 3.  
В 16-разрядном режиме регистр TMR2H содержит старший байт 16-разрядного Таймера 3.  
В 8-разрядном режиме регистр TMR2H содержит 8-разрядное значение старшего таймера.

## 20. ПРОГРАММИРУЕМЫЙ МАССИВ СЧЕТЧИКОВ

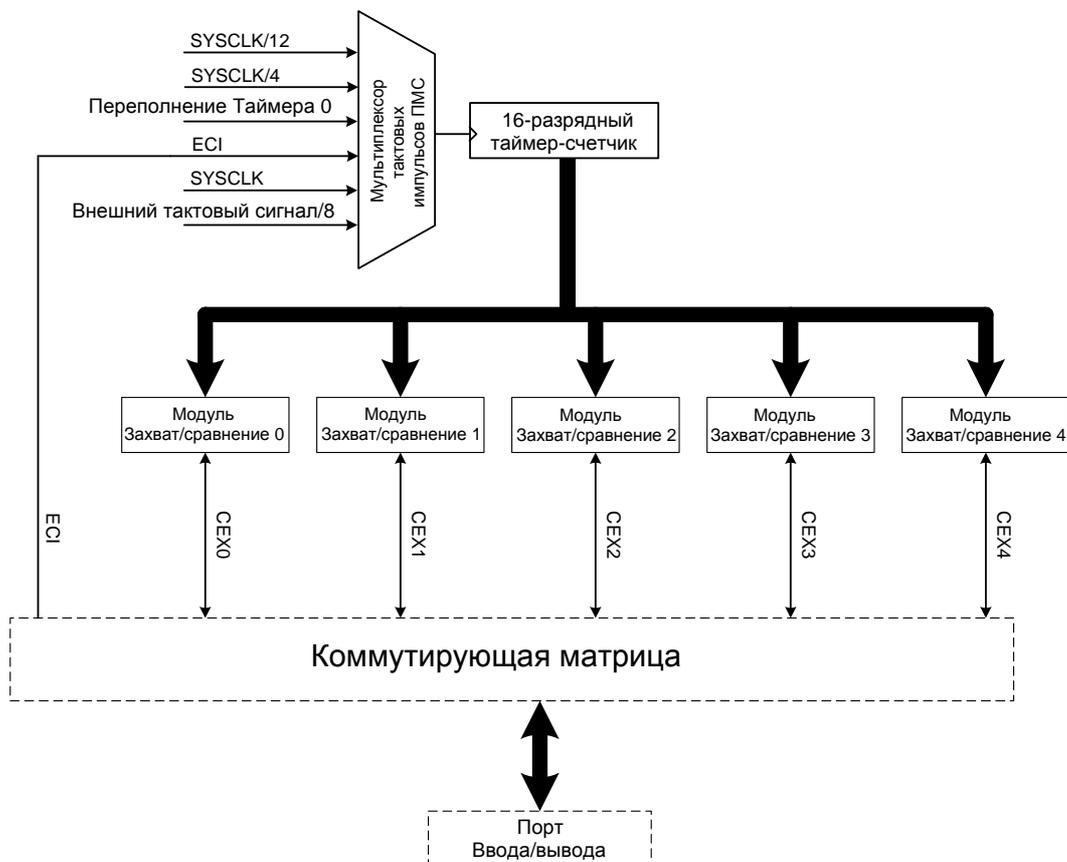
Программируемый массив счетчиков (ПМС) реализует расширенные таймерные функции, при этом требует меньшего вмешательства со стороны процессорного ядра, чем стандартные таймеры/счетчики архитектуры 8051. ПМС состоит из специального 16-разрядного таймера/счетчика и пяти 16-разрядных модулей захват/сравнение. Каждый модуль захват/сравнение имеет свою собственную линию ввода/вывода (СЕХn), которая через матрицу соединяется, если разрешено, с портом ввода/вывода (подробная информация о настройке матрицы приведена в разделе 14.1 «Приоритетный декодер матрицы» на стр.129). Таймер/счетчик тактируется программируемым внутренним сигналом, в качестве которого могут использоваться:

- внутренний сигнал с частотой, равной системной тактовой частоте;
- внутренний сигнал с частотой, равной 1/4 системной тактовой частоты;
- внутренний сигнал с частотой, равной 1/12 системной тактовой частоты;
- сигнал от внешнего генератора, деленный по частоте на 8;
- переполнение Таймера 0;
- входной сигнал на внешнем выводе ECI.

Каждый модуль захвата/сравнения можно независимо настроить для работы в одном из шести режимов: иницируемый по фронту сигнала захват, программный таймер, высокоскоростной выход, выход заданной частоты, 8-разрядный широтно-импульсный модулятор и 16-разрядный широтно-импульсный модулятор (описание каждого режима приведено в разделе 20.2 «Модули захвата/сравнения» на стр.237). Тактирование от внешнего генератора является идеальным решением для реализации функций реального времени, т.к. позволяет тактировать ПМС от прецизионного внешнего генератора в то время, как внутренний генератор используется для системного тактирования. Для управления модулем ПМС и его настройки используются связанные с ним SFR регистры. Структурная схема модуля ПМС показана на рис.20.1.

**Важное примечание:** Модуль 4 захвата/сравнения можно использовать как сторожевой таймер WDT (этот режим включается после системного сброса). Если режим WDT включен, то доступ к некоторым регистрам ПМС ограничен.

Рисунок 20.1. Структурная схема ПМС



## 20.1. Таймер/счетчик модуля ПМС

16-разрядный таймер/счетчик модуля ПМС состоит из двух 8-разрядных SFR регистров: PCA0L и PCA0H. PCA0H является старшим байтом (СЗБ) 16-разрядного таймера/счетчика, а PCA0L образует младший байт (МЗБ). При чтении регистра PCA0L значение регистра PCA0H автоматически фиксируется в регистре-защелке; последующее чтение регистра PCA0H возвратит данные именно из этого регистра-защелки. Таким образом, для обеспечения точности считывания полного 16-разрядного значения таймера/счетчика ПМС необходимо **сначала прочитать регистр PCA0L, а затем регистр PCA0H**. Чтение регистров PCA0H или PCA0L не препятствует функционированию счетчика. Выбор внутреннего сигнала тактирования таймера/счетчика осуществляется битами CPS2 - CPS0 регистра PCA0MD, как показано в табл.20.1.

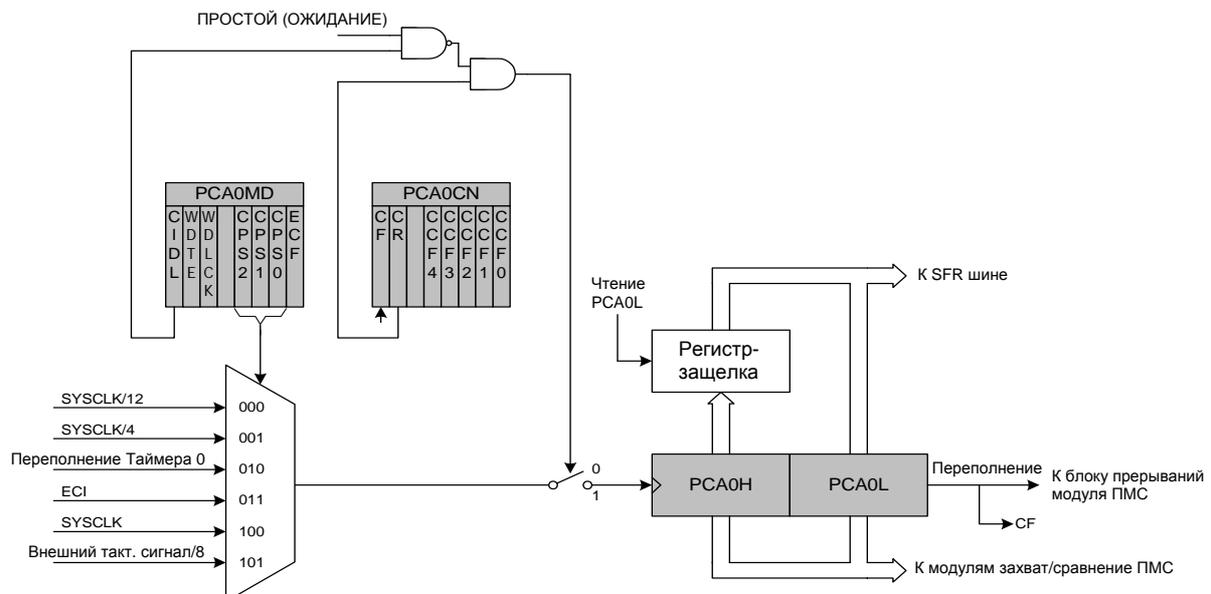
При переполнении таймера/счетчика из состояния 0xFFFF в состояние 0x0000 устанавливается в 1 флаг переполнения счетчика (CF) в регистре PCA0MD и, если прерывание от флага CF разрешено, генерируется запрос прерывания. Установка в 1 бита ECF в регистре PCA0MD разрешает генерацию запроса прерываний при установке флага CF. Бит CF не сбрасывается аппаратно при переходе к процедуре обслуживания прерывания и должен быть сброшен программно. Следует иметь в виду, что прерывания от флага CF распознаются только в том случае, если прерывания от модуля ПМС разрешены глобально. Прерывания от ПМС разрешаются глобально установкой в 1 битов EA (IE.7) и ERCA0 (EIE1.3). Сброс в 0 бита CIDL регистра PCA0MD позволяет ПМС продолжать нормальное функционирование в то время, когда МК переведен в режим ожидания.

Таблица 20.1. Выбор тактового сигнала для ПМС

CPS2	CPS1	CPS0	Внутренний сигнал тактирования ПМС
0	0	0	SYSCLK/12
0	0	1	SYSCLK/4
0	1	0	Переполнение Таймера 0
0	1	1	Срез (переход из 1 в 0) входного сигнала на внешнем выводе ECI (макс. частота = SYSCLK/4)
1	0	0	SYSCLK
1	0	1	Сигнал от внешнего генератора, деленный по частоте на 8*

\* Сигнал от внешнего генератора, деленный по частоте на 8, синхронизируется с системным тактовым сигналом

Рисунок 20.2. Структурная схема таймера/счетчика модуля ПМС



## 20.2. Модули захват/сравнение

Каждый модуль можно независимо настроить для работы в одном из шести режимов: инициируемый по фронту сигнала захват, программный таймер, высокоскоростной выход, выход заданной частоты, 8-разрядный широтно-импульсный модулятор и 16-разрядный широтно-импульсный модулятор. Каждый модуль имеет связанные с ним регистры специального назначения, которые используются для обмена данными с модулем и для настройки режимов работы.

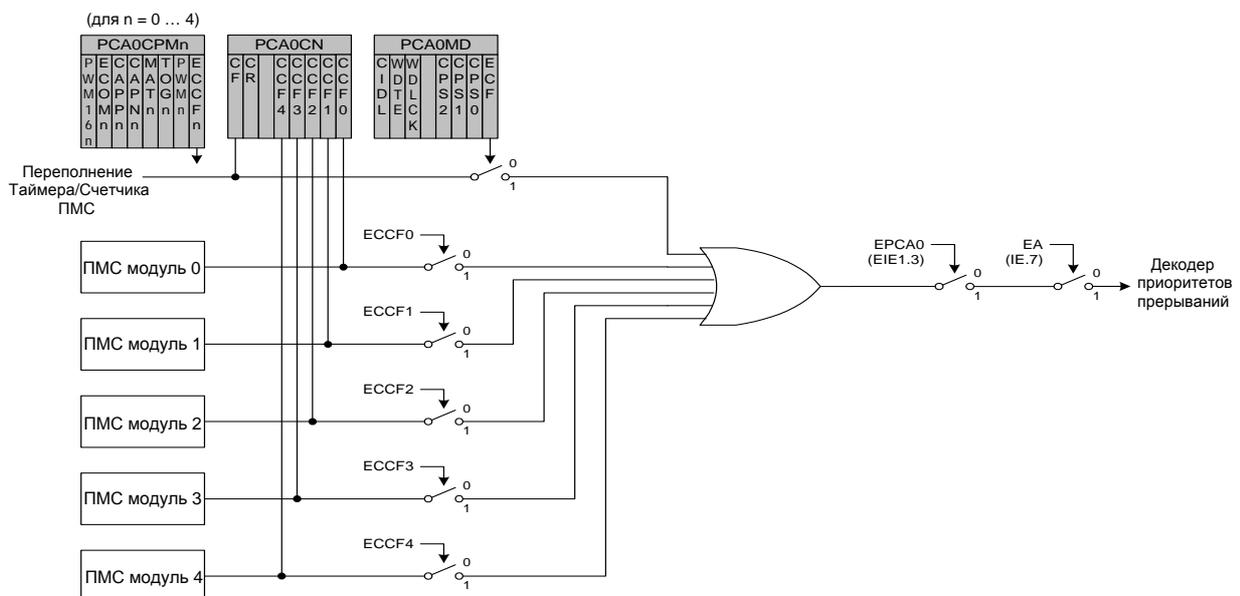
В табл.20.2 приведены комбинации бит в регистрах PCA0CPMn, используемые для перевода модулей захват/сравнение в различные режимы работы. Установка в 1 бит ECCFn в регистрах PCA0CPMn разрешает генерацию прерываний при установке в 1 флагов CCFn регистра PCA0CN. Следует иметь ввиду, что индивидуальные CCFn прерывания распознаются только в том случае, если прерывания от модуля ПМС разрешены глобально. Прерывания от ПМС разрешаются глобально установкой в 1 битов EA (IE.7) и EPCA0 (EIE1.3). Схема формирования прерываний от модуля ПМС приведена на рис.20.3.

**Таблица 20.2. Настройка модулей захват/сравнение в регистре PCA0CPM**

PWM16	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	Operation Mode
x	x	1	0	0	0	0	x	Захват инициируется положительным фронтом сигнала на линии CEXn
x	x	0	1	0	0	0	x	Захват инициируется отрицательным фронтом сигнала на линии CEXn
x	x	1	1	0	0	0	x	Захват инициируется изменением сигнала на линии CEXn
x	1	0	0	1	0	0	x	Программный таймер
x	1	0	0	1	1	0	x	Высокоскоростной выход
x	1	0	0	x	1	1	x	Выход заданной частоты
0	1	0	0	x	0	1	x	8-разр. широтно-импульсный модулятор
1	1	0	0	x	0	1	x	16-разр. широтно-импульсный модулятор

X = не имеет значения

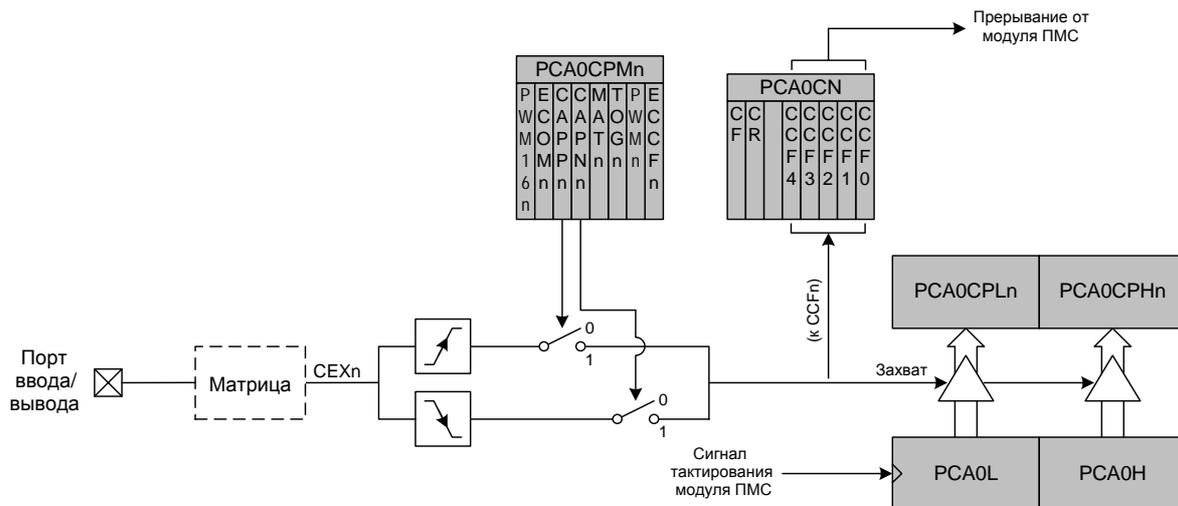
**Рисунок 20.3. Схема формирования прерывания от ПМС**



### 20.2.1. Режим захвата по фронту сигнала

В этом режиме активный фронт сигнала на внешнем выводе CEX<sub>n</sub> приведет к захвату значения таймера/счетчика ПМС и загрузке его в 16-разрядный регистр захвата/сравнения (PCA0CPL<sub>n</sub> и PCA0CPH<sub>n</sub>) соответствующего модуля. Биты CAPP<sub>n</sub> и CAPN<sub>n</sub> регистра PCA0CPM<sub>n</sub> определяют, по какому фронту будет осуществляться захват: по положительному (переход из 0 в 1), по отрицательному (переход из 1 в 0) или по любому фронту. Когда происходит захват, флаг захвата/сравнения (CCF<sub>n</sub>) в регистре PCA0CN устанавливается в 1 и, если CCF прерывание разрешено, генерируется запрос прерывания. Бит CCF<sub>n</sub> не сбрасывается аппаратно при переходе к процедуре обслуживания прерывания и должен быть сброшен программно. Если оба бита CAPP<sub>n</sub> и CAPN<sub>n</sub> установлены в 1, то состояние вывода порта, связанного с CEX<sub>n</sub>, можно прочитать непосредственно, чтобы определить, каким фронтом вызван захват (положительным или отрицательным).

Рисунок 20.4. Структурная схема ПМС в режиме захвата



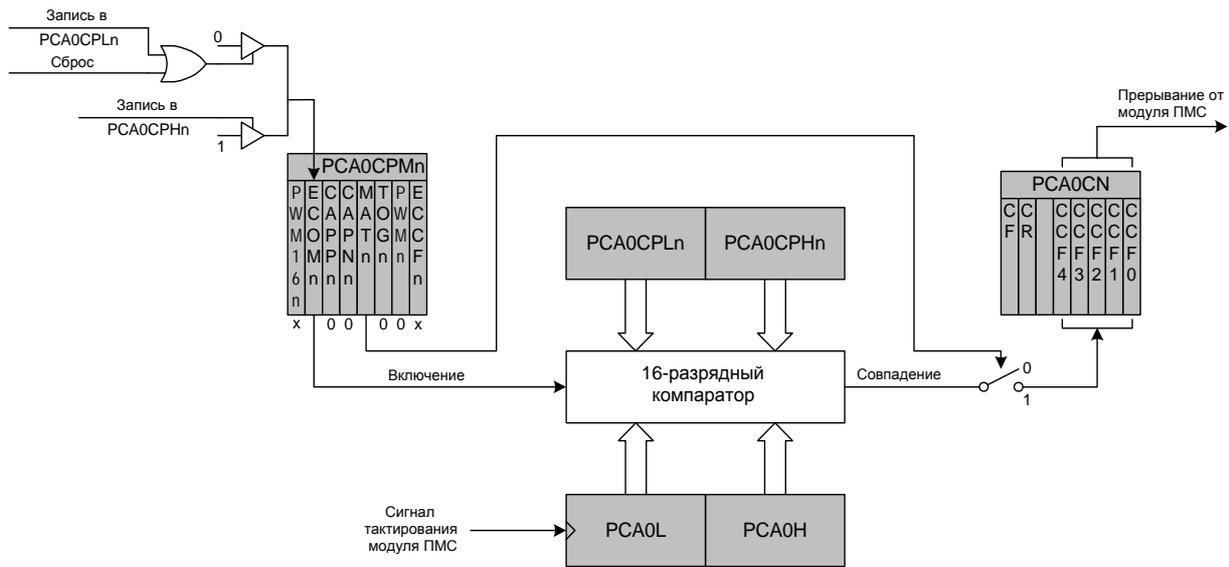
Примечание: Минимальная длительность удержания высокого или низкого уровней сигнала на входе CEX<sub>n</sub> составляет 2 системных тактовых цикла, что необходимо для обеспечения правильного функционирования модуля ПМС.

## 20.2.2. Режим программного таймера (сравнения)

В режиме программного таймера значение таймера/счетчика ПМС сравнивается со значением 16-разрядного регистра захвата/сравнения (PCA0CPHn и PCA0CPLn) соответствующего модуля. Когда происходит совпадение, флаг захвата/сравнения (CCFn) в регистре PCA0CN устанавливается в 1 и, если CCF прерывание разрешено, генерируется запрос прерывания. Бит CCFn не сбрасывается аппаратно при переходе к процедуре обслуживания прерывания и должен быть сброшен программно. Режим программного таймера включается установкой в 1 битов ECOMn и MATn регистра PCA0CPMn.

**Важное замечание относительно регистров захвата/сравнения:** При записи 16-разрядного значения в регистры захвата/сравнения модуля ПМС младший байт всегда необходимо записывать первым. Запись в регистр PCA0CPLn сбрасывает в 0 бит ECOMn; запись в регистр PCA0CPHn устанавливает в 1 бит ECOMn.

**Рисунок 20.5. Структурная схема ПМС в режиме программного таймера**





## 20.2.4. Режим выхода заданной частоты

В режиме выхода заданной частоты на связанном с конкретным модулем выводе CEX<sub>n</sub> генерируется сигнал прямоугольной формы с программируемой частотой. Содержимое старшего байта регистра захвата/сравнения (PCA0CPH<sub>n</sub>) определяет количество циклов тактирования ПМС, отсчитываемых до инвертирования состояния сигнала на выходе CEX<sub>n</sub>. Таким образом, частота прямоугольного сигнала определяется в соответствии со следующим уравнением:

**Уравнение 23.1. Частота прямоугольного сигнала в режиме выхода заданной частоты**

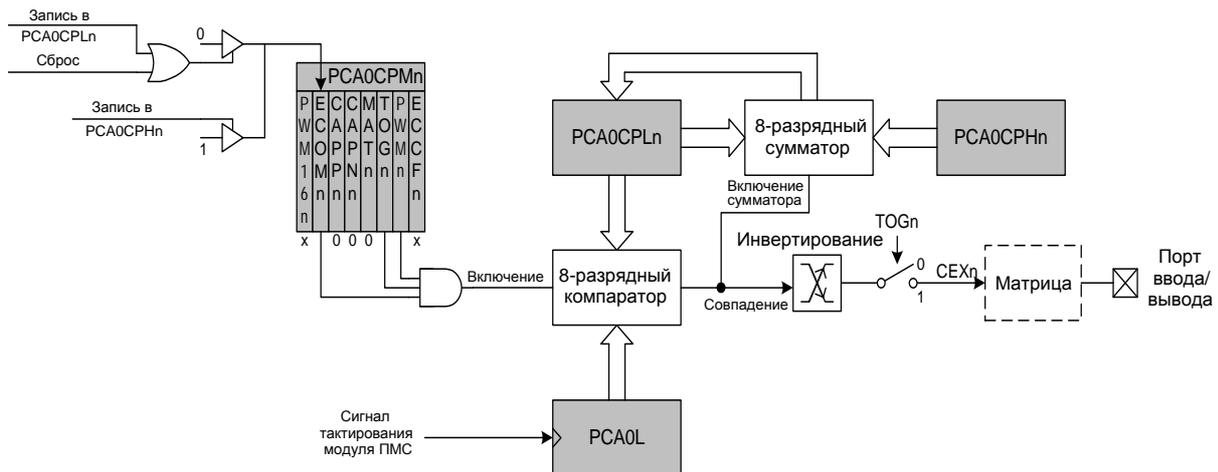
$$F_{CEX_n} = F_{PGA} / (2 * PCA0CPH_n),$$

где  $F_{PGA}$  - частота сигнала тактирования, задаваемая битами CPS2 – 0 регистра режима ПМС (PCA0MD).

Примечание: значение 0x00 регистра PCA0CPH<sub>n</sub> равно значению 256 для этого уравнения.

Содержимое младшего байта регистра захвата/сравнения (PCA0CPL<sub>n</sub>) сравнивается с младшим байтом счетчика ПМС (PCA0L); при их совпадении сигнал на выводе CEX<sub>n</sub> инвертируется и значение смещения, хранящееся в старшем байте (PCA0CPH<sub>n</sub>), добавляется к значению регистра PCA0CPL<sub>n</sub>. Режим выхода заданной частоты включается установкой в 1 битов ECOM<sub>n</sub>, TOG<sub>n</sub> и PWM<sub>n</sub> и регистра PCA0CPM<sub>n</sub>.

**Рисунок 20.7. Структурная схема ПМС в режиме выхода заданной частоты**



### 20.2.5. Режим 8-разрядного широтно-импульсного модулятора

Каждый модуль захвата/сравнения можно использовать независимо от других для генерации на соответствующем ему выводе СЕХ<sub>n</sub> выходного сигнала с широтно-импульсной модуляцией (ШИМ). Частота этого выходного сигнала зависит от частоты сигнала тактирования таймера/счетчика ПМС. Для изменения коэффициента заполнения (скважности) выходного ШИМ сигнала используется регистр захвата/сравнения PCA0CPL<sub>n</sub> соответствующего модуля. Когда значение младшего байта таймера/счетчика ПМС (PCA0L) становится равным значению регистра PCA0CPL<sub>n</sub>, на внешнем выводе СЕХ<sub>n</sub> устанавливается сигнал высокого уровня. Когда регистр PCA0L переполнится, на выводе СЕХ<sub>n</sub> установится сигнал низкого уровня (см. рис.20.8). Кроме этого, при переполнении младшего байта таймера/счетчика (PCA0L) из состояния 0xFF в состояние 0x00 регистр PCA0CPL<sub>n</sub> автоматически перезагружается значением, хранящимся в регистре PCA0CPH<sub>n</sub>, без вмешательства со стороны программы. Во избежание сбоев в работе цифрового компаратора рекомендуется осуществлять запись в регистр PCA0CPH<sub>n</sub>, а не в регистр PCA0CPL<sub>n</sub>. Режим 8-разрядного широтно-импульсного модулятора включается установкой в 1 бит ECOM<sub>n</sub> и PWM<sub>n</sub> регистра PCA0CPM<sub>n</sub>. Скважность выходного сигнала в режиме 8-разрядного ШИМ определяется уравнением 20.2.

**Важное замечание относительно регистров захвата/сравнения:** При записи 16-разрядного значения в регистры захвата/сравнения модуля ПМС младший байт всегда необходимо записывать первым. Запись в регистр PCA0CPL<sub>n</sub> сбрасывает в 0 бит ECOM<sub>n</sub>; запись в регистр PCA0CPH<sub>n</sub> устанавливает в 1 бит ECOM<sub>n</sub>.

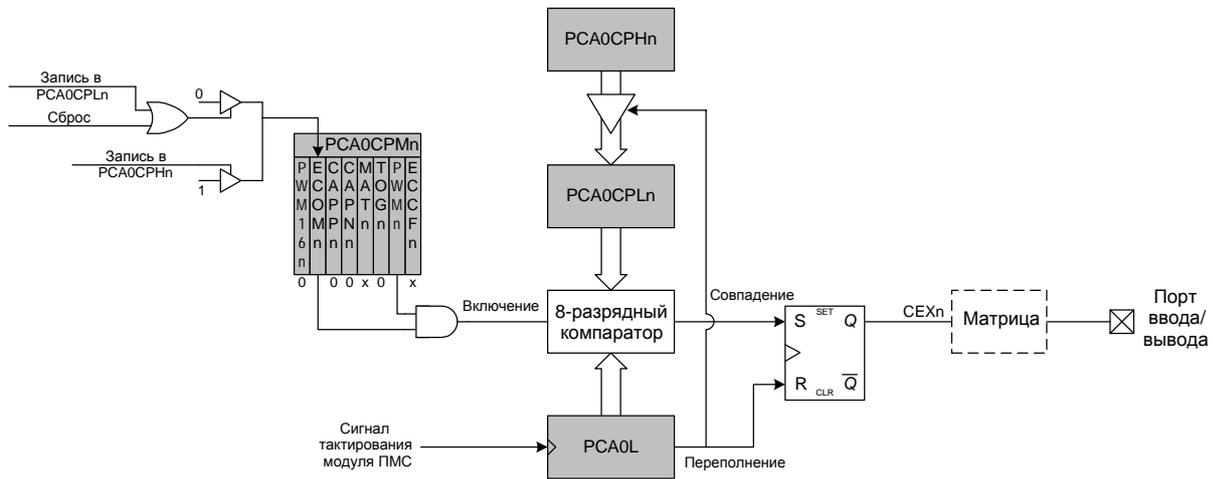
#### Уравнение 20.2. Скважность выходного сигнала в режиме 8-разр. ШИМ

$$DutyCycle = (256 - PCA0CPH_n) / 256$$



В соответствии с уравнением 20.2 максимальная скважность составляет 100% ( $PCA0CPH_n = 0$ ), а минимальная скважность составляет 0,39% ( $PCA0CPH_n = 0xFF$ ). Сигнал со скважностью, равной 0%, можно получить, сбросив в 0 бит  $ECOM_n$ .

**Рисунок 20.8. Структурная схема ПМС в режиме 8-разр. ШИМ.**



### 20.2.6. Режим 16-разрядного широтно-импульсного модулятора

Каждый модуль захвата/сравнения можно также использовать в режиме 16-разрядного ШИМ. В этом режиме 16-разрядное значение регистров захвата/сравнения (PCA0CPHn : PCA0CPLn) определяет количество циклов тактирования ПМС, в течение которых выходной сигнал ШИМ удерживается на низком логическом уровне. Когда значение счетчика ПМС сравнивается с содержимым регистров захвата/сравнения (PCA0CPHn : PCA0CPLn), на выходе CEXn устанавливается сигнал высокого уровня; когда счетчик ПМС переполнится, на выходе CEXn установится сигнал низкого уровня. Чтобы выводить сигнал с изменяемой скважностью, запись новых значений необходимо синхронизировать с прерываниями от флага CCFn модуля ПМС. Режим 16-разрядного широтно-импульсного модулятора включается установкой в 1 бит ECOMn, PWMn и PWM16n регистра PCA0CPMn. Для получения сигнала с изменяемой скважностью следует разрешить прерывания (ECCFn = 1 и MATn = 1), чтобы обеспечить возможность синхронизации операций записи регистра захвата/сравнения. Скважность выходного сигнала в режиме 16-разрядного ШИМ определяется уравнением 20.3.

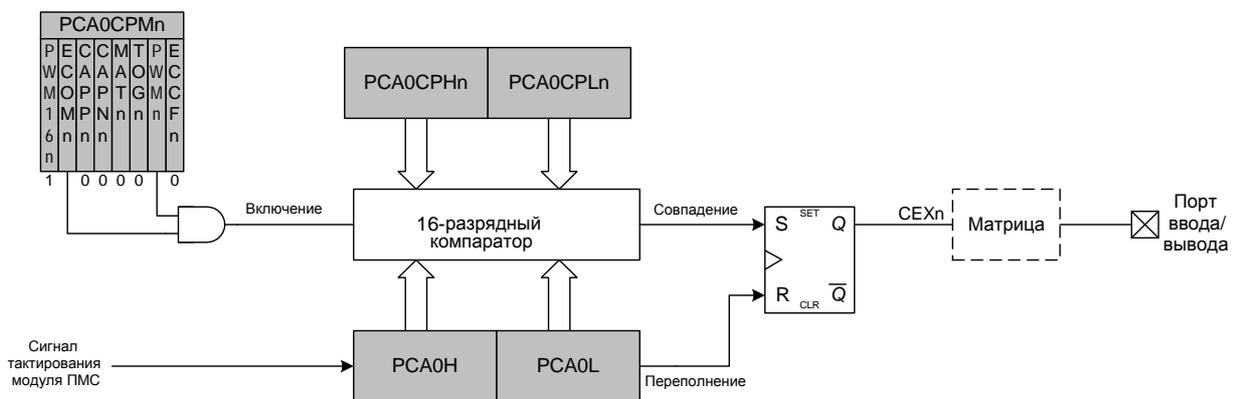
**Важное замечание относительно регистров захвата/сравнения:** При записи 16-разрядного значения в регистры захвата/сравнения модуля ПМС младший байт всегда необходимо записывать первым. Запись в регистр PCA0CPLn сбрасывает в 0 бит ECOMn; запись в регистр PCA0CPHn устанавливает в 1 бит ECOMn.

### Уравнение 20.3. Скважность выходного сигнала в режиме 16-разр. ШИМ

$$DutyCycle = (65536 - PCA0CPn) / 65536$$

В соответствии с уравнением 20.3 максимальная скважность составляет 100% (PCA0CPHn = 0), а минимальная скважность составляет 0,0015% (PCA0CPHn = 0xFFFF). Сигнал со скважностью, равной 0%, можно получить, сбросив в 0 бит ECOMn.

Рисунок 20.9. Структурная схема ПМС в режиме 16-разр. ШИМ.





### 20.3. Режим сторожевого таймера

Модуль 4 ПМС можно использовать в режиме программируемого сторожевого таймера (WDT). WDT используется для генерации системного сброса в случае, если время между операциями записи в регистр обновления WDT (PCA0CPH4) превышает заданное значение. WDT можно программно настраивать, а также включать/отключать при необходимости.

Если биты WDTE и WDLCK в регистре PCA0MD установлены в 1, то модуль 4 функционирует как сторожевой таймер (WDT). Старший байт модуля 4 сравнивается со старшим байтом счетчика ПМС; младший байт модуля 4 содержит смещение, которое используется при обновлении WDT. **При сбросе сторожевой таймер WDT включается. Пока WDT включен, доступ к некоторым регистрам ПМС ограничен.**

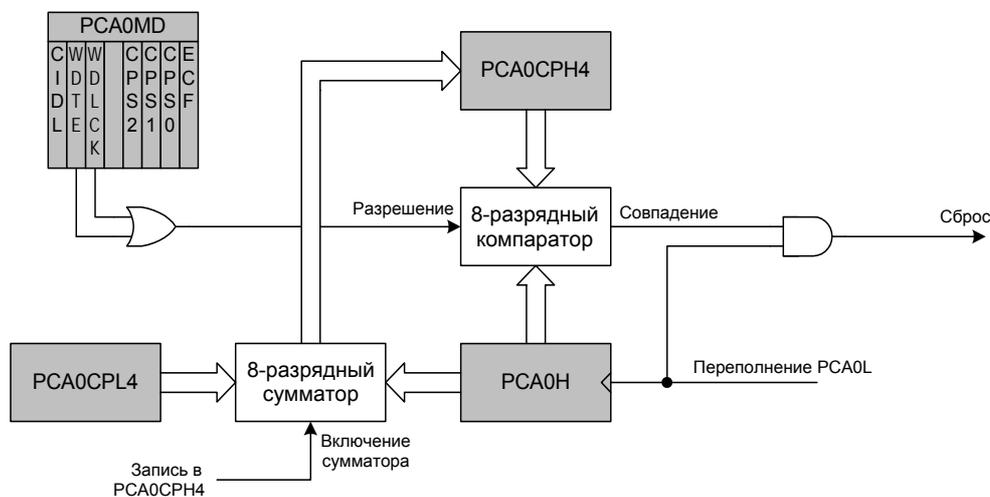
#### 20.3.1. Функционирование сторожевого таймера

Если WDT включен, то:

- Счетчик ПМС считает.
- Запись в регистры PCA0L и PCA0H запрещена.
- Биты выбора источника тактирования ПМС «заморожены».
- Бит управления состоянием ожидания ПМС «заморожен».
- Модуль 4 переведен в режим сторожевого таймера.
- Операции записи в регистр режима модуля 4 (PCA0CPM4) запрещены.

Если WDT включен, то запись бита CR не изменит состояния счетчика ПМС; счетчик будет считать до тех пор, пока WDT не будет запрещен. Бит управления запуском счетчика ПМС (CR) будет читаться как '0', если WDT включен, но не программа пользователя включает счетчик ПМС. Если произойдет совпадение значений регистров PCA0CPH4 и PCA0H, когда WDT включен, то будет сгенерирован системный сброс. Если требуется предотвратить сброс от WDT, то WDT можно обновить путем записи любого значения в регистр PCA0CPH4. При записи PCA0CPH4 в регистр PCA0CPH4 будет загружаться значение регистра PCA0L плюс смещение, содержащееся в регистре PCA0CPL4 (см. рис.20.10).

**Рисунок 20.10. Модуль 4 захвата/сравнения ПМС с включенным сторожевым таймером**



Следует иметь в виду, что 8-разрядное смещение, хранимое в регистре PCA0CPL4, сравнивается со старшим байтом 16-разрядного счетчика ПМС. Это значение представляет собой количество переполнений PCA0L до сброса. Перед первым переполнением PCA0L может пройти до 256 циклов ПМС, в зависимости от значения PCA0L при выполнении обновления. Таким образом, общее смещение определяется (в циклах ПМС) уравнением 20.4, где PCA0L – значение регистра PCA0L в момент обновления.

## Уравнение 20.4. Смещение WDT в циклах ПМС

$$\text{Смещение} = (256 \times \text{PCA0CPL4}) + (256 - \text{PCA0L})$$

Сброс от WDT генерируется, если PCA0L переполняется, когда значения регистров PCA0CPH4 и PCA0CN совпадают. Программа может вызвать сброс от WDT, установив в 1 бит CCF4 (PCA0CN.4), когда WDT включен.

### 20.3.2. Использование сторожевого таймера

Чтобы настроить WDT, необходимо выполнить следующую последовательность действий:

1. Отключить WDT, сбросив в 0 бит WDTE.
2. Выбрать необходимый источник тактирования ПМС (с помощью бит CPS2 – CPS0).
3. Загрузить в регистр PCA0CPL4 необходимое значение смещения, используемое при обновлении WDT.
4. Перевести ПМС в режим ожидания (установить бит CIDL в 1, если WDT должен останавливаться при переходе CPU в режим ожидания).
5. Включить WDT, установив в 1 бит WDTE.
6. (При необходимости) Заблокировать WDT (запретить отключение WDT до следующего системного сброса), установив в 1 бит WDLCK.

Источник тактирования ПМС и выбор режима ожидания нельзя изменить, пока WDT включен. Сторожевой таймер включается установкой в 1 битов WDTE или WDLCK в регистре PCA0MD. Если WDLCK = 1, то WDT нельзя отключить до следующего системного сброса. Если WDLCK = 0, то WDT можно отключить, сбросив в 0 бит WDTE.

WDT включается после любого сброса. По умолчанию счетчик ПМС тактируется сигналом SYSCLK/12, PCA0L = 0x00, PCA0CPL4 = 0x00. Отсюда следует, что по умолчанию таймаут WDT составляет 256 системных тактовых циклов (см. уравнение 20.4). В таблице 20.3 приведены некоторые значения таймаутов WDT для типичных частот системного тактового сигнала.

**Таблица 20.3. Значения таймаута сторожевого таймера\***

Системный тактовый сигнал (МГц)	PCA0CPL4	Таймаут WDT (мс)
12,000,000	255	65.5
12,000,000	128	33.0
12,000,000	32	8.4
18,432,000	255	42.7
18,432,000	128	21.5
18,432,000	32	5.5
11,059,200	255	71.1
11,059,200	128	35.8
11,059,200	32	9.2
4,000,000	255	196.6
4,000,000	128	99.1
4,000,000	32	25.3
32,000	255	24,576.0
32,000	128	12,384.0
32,000	32	3,168.0

\* Предполагается использование SYSCLK/12 в качестве сигнала тактирования ПМС и PCA0L = 0x00 в момент обновления.

\*\* Частота сброса внутреннего генератора.

## 20.4. Описание регистров модуля ПМС

Ниже приводится описание регистров специального назначения, связанных с работой модуля ПМС.

**Рисунок 20.11. PCA0CN: Регистр управления ПМС**

R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD8
CF	CR	-	CCF4	CCF3	CCF2	CCF1	CCF0	
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0 <small>(доступен в битовом режиме адресации)</small>	
<p><b>Бит 7: CF:</b> Флаг переполнения Таймера/Счетчика ПМС. Устанавливается в 1 аппаратно, когда Таймер/Счетчик ПМС переполняется из состояния 0xFFFF в состояние 0x0000. Если прерывание от Таймера/Счетчика ПМС (от флага CF) разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								
<p><b>Бит 6: CR:</b> Управление запуском Таймера/Счетчика ПМС. Этот бит включает/отключает Таймер/Счетчик ПМС. 0: Таймер/Счетчик ПМС отключен. 1: Таймер/Счетчик ПМС включен.</p>								
<p><b>Бит 5:</b> Не используется. Читается как 0b. Запись не оказывает никакого влияния.</p>								
<p><b>Бит 4: CCF4:</b> Флаг захвата/сравнения модуля 4 ПМС. Этот бит устанавливается в 1 аппаратно, если происходит захват или совпадение сравниваемых значений. Если прерывание от флага CCF разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CCF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								
<p><b>Бит 3: CCF3:</b> Флаг захвата/сравнения модуля 3 ПМС. Этот бит устанавливается в 1 аппаратно, если происходит захват или совпадение сравниваемых значений. Если прерывание от флага CCF разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CCF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								
<p><b>Бит 2: CCF2:</b> Флаг захвата/сравнения модуля 2 ПМС. Этот бит устанавливается в 1 аппаратно, если происходит захват или совпадение сравниваемых значений. Если прерывание от флага CCF разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CCF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								
<p><b>Бит 1: CCF1:</b> Флаг захвата/сравнения модуля 1 ПМС. Этот бит устанавливается в 1 аппаратно, если происходит захват или совпадение сравниваемых значений. Если прерывание от флага CCF разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CCF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								
<p><b>Бит 0: CCF0:</b> Флаг захвата/сравнения модуля 0 ПМС. Этот бит устанавливается в 1 аппаратно, если происходит захват или совпадение сравниваемых значений. Если прерывание от флага CCF разрешено, то установка этого бита приведет к переходу на процедуру обслуживания прерывания от флага CCF. Этот бит не сбрасывается аппаратно и должен быть сброшен программно.</p>								



Рисунок 20.12. PCA0MD: Регистр режима ПМС

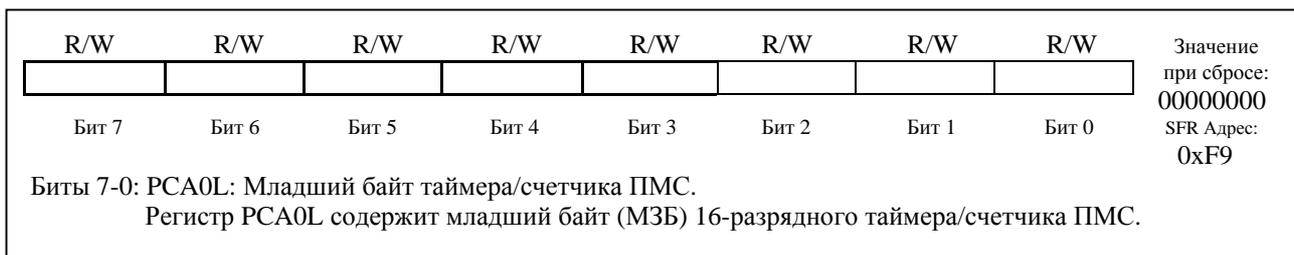
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000 SFR Адрес: 0xD9																																				
CIDL	WDTE	WDLCK	-	CPS2	CPS1	CPS0	ECF																																					
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0																																					
<p><b>Бит 7: CIDL:</b> Управление режимом простоя (ожидания) Таймера/Счетчика ПМС.          Это бит определяет поведение ПМС в то время, когда CPU находится в режиме простоя (ожидания).          0: ПМС продолжает нормально функционировать в то время, когда МК находится в режиме простоя (ожидания).          1: Работа ПМС приостанавливается в то время, когда МК находится в режиме простоя (ожидания).</p> <p><b>Бит 6: WDTE:</b> Включение сторожевого таймера.          Если WDTE = 1, то Модуль 4 ПМС используется как сторожевой таймер.          0: Сторожевой таймер отключен.          1: Модуль 4 ПМС включен в режиме сторожевого таймера.</p> <p><b>Бит 5: WDLCK:</b> Блокировка сторожевого таймера.          Этот бит включает и блокирует сторожевой таймер. Если WDLCK = 1, то сторожевой таймер нельзя отключить до следующего системного сброса.          0: Сторожевой таймер разблокирован.          1: Сторожевой таймер включен и заблокирован.</p> <p><b>Бит 4:</b> Не используется. Читается как 0b. Запись не оказывает никакого влияния.</p> <p><b>Биты 3-1: CPS2-CPS0:</b> Выбор сигнала тактирования Таймера/Счетчика ПМС.          Эти биты определяют, какой сигнал будет использоваться для тактирования Таймера/Счетчика ПМС.</p> <table border="1"> <thead> <tr> <th>CPS2</th> <th>CPS1</th> <th>CPS0</th> <th>Внутренний сигнал тактирования ПМС</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>SYSClk/12</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>SYSClk/4</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>Переполнение Таймера 0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>Срез (переход из 1 в 0) входного сигнала на внешнем выводе ECI (максимальная частота = SYSClk/4)</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>SYSClk</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>Сигнал от внешнего источника, деленный по частоте на 8*</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>Зарезервировано</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>Зарезервировано</td> </tr> </tbody> </table> <p>* Сигнал от внешнего генератора, деленный по частоте на 8, синхронизируется с системным тактовым сигналом.</p> <p><b>Бит 0: ECF:</b> Разрешение прерываний от переполнения Таймера/Счетчика ПМС.          Этот бит разрешает/запрещает прерывания от переполнения Таймера/Счетчика ПМС (от флага CF).          0: Прерывания от флага CF (PCA0CN.7) запрещены.          1: Прерывания от флага CF (PCA0CN.7) разрешены.</p> <p><b>Примечание:</b> Если WDTE = 1, то значение регистра PCA0MD нельзя модифицировать. Чтобы изменить значение регистра PCA0MD, необходимо сначала отключить сторожевой таймер.</p>									CPS2	CPS1	CPS0	Внутренний сигнал тактирования ПМС	0	0	0	SYSClk/12	0	0	1	SYSClk/4	0	1	0	Переполнение Таймера 0	0	1	1	Срез (переход из 1 в 0) входного сигнала на внешнем выводе ECI (максимальная частота = SYSClk/4)	1	0	0	SYSClk	1	0	1	Сигнал от внешнего источника, деленный по частоте на 8*	1	1	0	Зарезервировано	1	1	1	Зарезервировано
CPS2	CPS1	CPS0	Внутренний сигнал тактирования ПМС																																									
0	0	0	SYSClk/12																																									
0	0	1	SYSClk/4																																									
0	1	0	Переполнение Таймера 0																																									
0	1	1	Срез (переход из 1 в 0) входного сигнала на внешнем выводе ECI (максимальная частота = SYSClk/4)																																									
1	0	0	SYSClk																																									
1	0	1	Сигнал от внешнего источника, деленный по частоте на 8*																																									
1	1	0	Зарезервировано																																									
1	1	1	Зарезервировано																																									

Рисунок 20.13. PCA0CPMn: Регистры управления модулями захват/сравнение

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Значение при сбросе: 00000000
PWM16n	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	SFR Адрес: 0xDA-0xDE
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	
<p>Адреса регистров PCA0CPMn: PCA0CPM0 = 0xDA (n = 0)  PCA0CPM1 = 0xDB (n = 1)  PCA0CPM2 = 0xDC (n = 2)  PCA0CPM3 = 0xDD (n = 3)  PCA0CPM4 = 0xDE (n = 4)</p>								
<p><b>Бит 7: PWM16n:</b> Включение режима 16-разрядного ШИМ.  Этот бит выбирает 16-разрядный режим, если режим ШИМ включен (PWMn = 1).  0: Выбран режим 8-разр. ШИМ.  1: Выбран режим 16-разр. ШИМ.</p>								
<p><b>Бит 6: ECOMn:</b> Разрешение функции компаратора.  Этот бит включает/отключает функцию компаратора модуля n ПМС.  0: Компаратор отключен.  1: Компаратор включен.</p>								
<p><b>Бит 5: CAPPn:</b> Разрешение функции захвата по положительному фронту.  Этот бит разрешает/запрещает захват по положительному фронту для модуля n ПМС.  0: Захват по положительному фронту запрещен.  1: Захват по положительному фронту разрешен.</p>								
<p><b>Бит 4: CAPNn:</b> Разрешение функции захвата по отрицательному фронту.  Этот бит разрешает/запрещает захват по отрицательному фронту для модуля n ПМС.  0: Захват по отрицательному фронту запрещен.  1: Захват по отрицательному фронту разрешен.</p>								
<p><b>Бит 3: MATn:</b> Разрешение функции определения совпадения.  Этот бит включает/отключает функцию определения совпадения для модуля n ПМС. Если MATn = 1, то совпадение значения счетчика ПМС со значением регистра захвата/сравнения соответствующего модуля приведет к установке в 1 бита CCFn в регистре PCA0MD.  0: Функция определения совпадения отключена.  1: Функция определения совпадения включена.</p>								
<p><b>Бит 2: TOGn:</b> Разрешение функции инвертирования выхода.  Этот бит включает/отключает функцию инвертирования выходного сигнала для модуля n ПМС. Если TOGn = 1, то совпадение значения счетчика ПМС со значением регистра захвата/сравнения соответствующего модуля приведет к инвертированию логического уровня выходного сигнала на внешнем выводе CEXn. Если также PWMn = 1, то модуль функционирует в режиме выхода заданной частоты.  0: Функция инвертирования выхода отключена.  1: Функция инвертирования выхода включена.</p>								
<p><b>Бит 1: PWMn:</b> Включение режима ШИМ.  Этот бит включает/отключает функцию ШИМ для модуля n ПМС. Если PWMn = 1, то выходной ШИМ-сигнал появляется на внешнем выводе CEXn. Если PWM16n = 0, то используется режим 8-разр. ШИМ; если PWM16n = 1, то используется режим 16-разр. ШИМ. Если TOGn = 1, то модуль работает в режиме выхода заданной частоты.  0: Функция ШИМ отключена.  1: Функция ШИМ включена.</p>								
<p><b>Бит 0: ECCFn:</b> Разрешение прерываний от флага захвата/сравнения (CCFn).  Этот бит разрешает/запрещает прерывания от флага захвата/сравнения (CCFn).  0: Прерывания от флага CCFn запрещены.  1: Прерывания от флага CCFn разрешены.</p>								



**Рисунок 20.14. PCA0L: Младший байт таймера/счетчика ПМС**



**Рисунок 20.15. PCA0H: Старший байт таймера/счетчика ПМС**

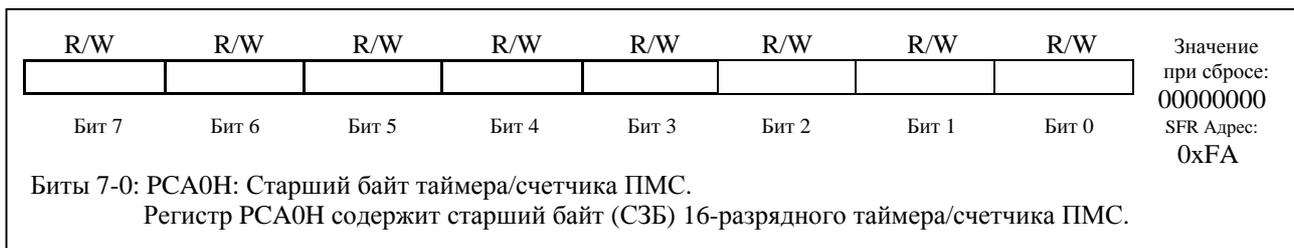


Рисунок 20.16. PCA0CPLn: Младший байт модуля захвата ПМС

| R/W   |
|-------|-------|-------|-------|-------|-------|-------|-------|
|       |       |       |       |       |       |       |       |
| Бит 7 | Бит 6 | Бит 5 | Бит 4 | Бит 3 | Бит 2 | Бит 1 | Бит 0 |

Значение при сбросе: 00000000  
SFR Адрес: 0xFB, 0xE9, 0xEB, 0xED, 0xFD

Адреса регистров PCA0CPLn: PCA0CPL0 = 0xFB (n = 0)  
PCA0CPL1 = 0xE9 (n = 1)  
PCA0CPL2 = 0xEB (n = 2)  
PCA0CPL3 = 0xED (n = 3)  
PCA0CPL4 = 0xFD (n = 4)

Биты 7-6: PCA0CPLn: Младший байт модуля захвата ПМС.  
Регистр PCA0CPLn содержит младший байт (МЗБ) 16-разрядного модуля захвата n.

Рисунок 20.17. PCA0CPHn: Старший байт модуля захвата ПМС

| R/W   |
|-------|-------|-------|-------|-------|-------|-------|-------|
|       |       |       |       |       |       |       |       |
| Бит 7 | Бит 6 | Бит 5 | Бит 4 | Бит 3 | Бит 2 | Бит 1 | Бит 0 |

Значение при сбросе: 00000000  
SFR Адрес: 0xFC, 0xEA, 0xEC, 0xEE, 0xFE

Адреса регистров PCA0CPHn: PCA0CPH0 = 0xFC (n = 0)  
PCA0CPH1 = 0xEA (n = 1)  
PCA0CPH2 = 0xEC (n = 2)  
PCA0CPH3 = 0xEE (n = 3)  
PCA0CPH4 = 0xFE (n = 4)

Биты 7-0: PCA0CPHn: Старший байт модуля захвата ПМС.  
Регистр PCA0CPHn содержит старший байт (СЗБ) 16-разрядного модуля захвата n.



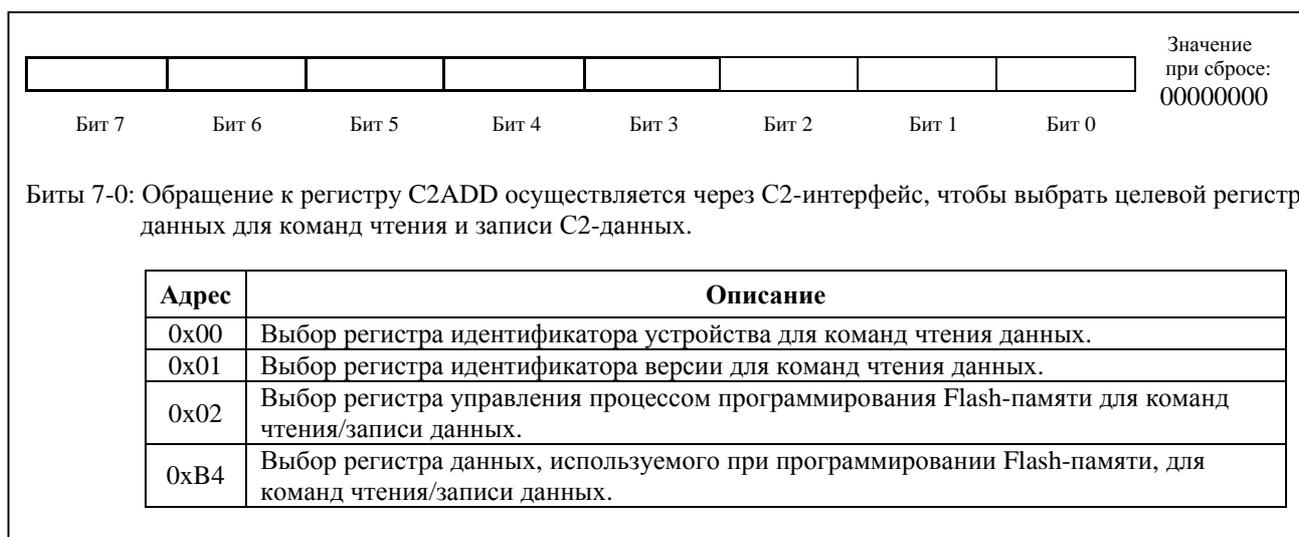
## 21. ИНТЕРФЕЙС C2

МК C8051F320/1 имеет встроенный интерфейс отладки Silicon Labs 2-Wire (C2), предназначенный для программирования Flash-памяти, выполнения граничного сканирования, а также для внутрисистемной отладки с помощью МК, установленного в конечное изделие. Интерфейс C2 использует сигнал тактирования (C2СК) и двунаправленный сигнал C2-данных (C2D) для передачи информации между МК и хост-системой. Подробная информация о протоколе C2 приведена в спецификации на интерфейс C2.

### 21.1. Регистры интерфейса C2

Ниже описаны C2-регистры, необходимые для выполнения программирования Flash-памяти и функций граничного сканирования. Все C2-регистры доступны через C2-интерфейс, как описано в спецификации на интерфейс C2.

**Рисунок 21.1. C2ADD: Регистр адреса C2**



**Рисунок 21.2. DEVICEID: Регистр идентификатора C2-устройства**

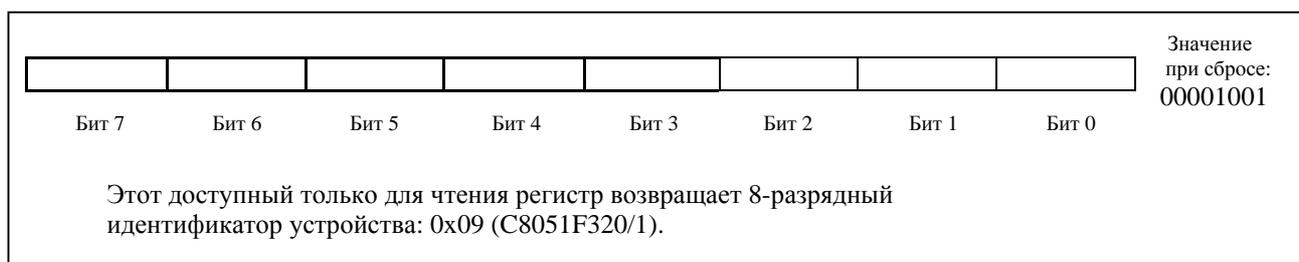


Рисунок 21.3. REVID: Регистр идентификатора версии C2

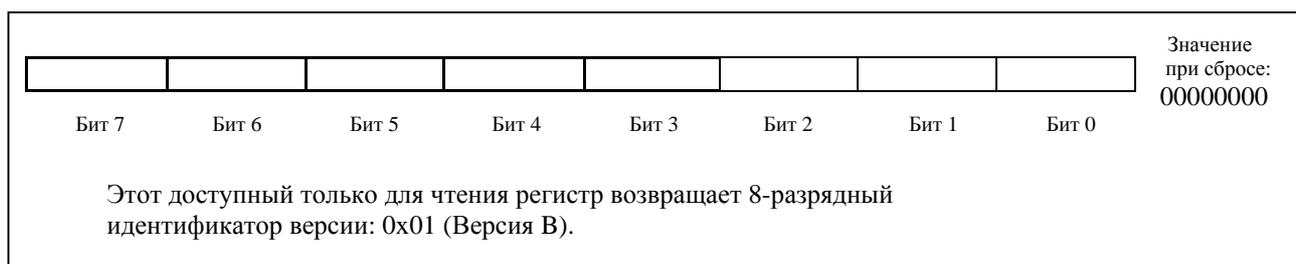


Рисунок 21.4. FPCTL: Регистр управления программированием Flash-памяти посредством интерфейса C2

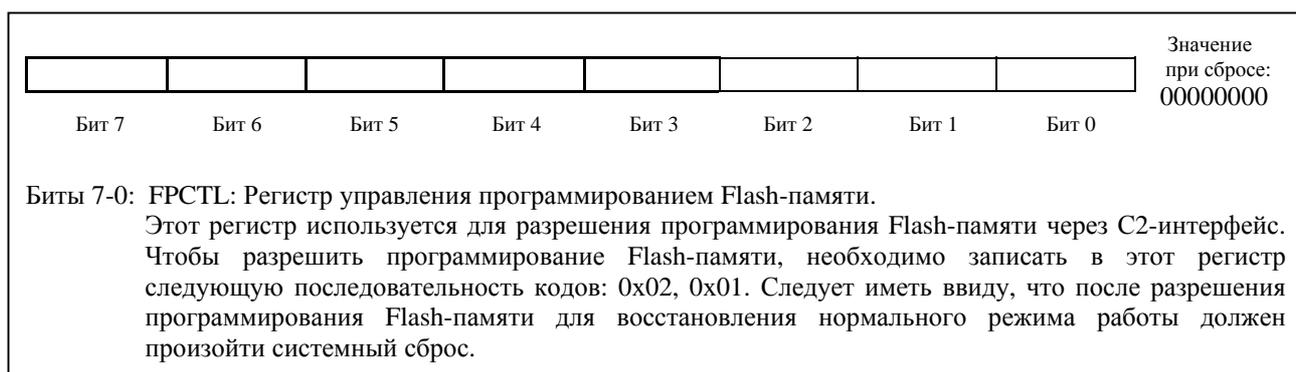


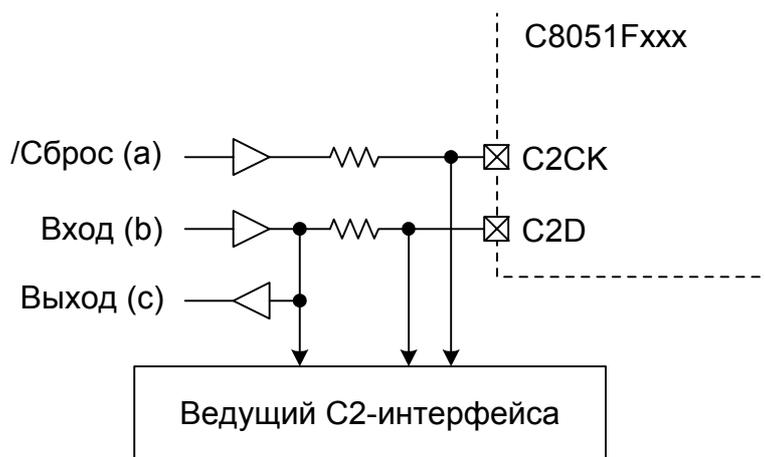
Рисунок 21.5. FPDAT: Регистр данных интерфейса C2, используемый при программировании Flash-памяти



## 21.2. Разделение выводов интерфейса C2

Протокол C2 допускает разделение выводов интерфейса C2 с другими пользовательскими функциями таким образом, чтобы можно было выполнять внутрисистемную отладку, программирование Flash-памяти и граничное сканирование. Это возможно, т.к. взаимодействие с помощью C2-интерфейса обычно осуществляется тогда, когда МК находится в состоянии останова, при котором все встроенные периферийные модули и программа пользователя останавливаются. В этом остановленном состоянии C2-интерфейс может, не мешая системе, «занять» выводы C2CK (/RST) и C2D (P3.0). В большинстве приложений требуются внешние резисторы, чтобы изолировать трафик C2-интерфейса от приложения пользователя. Типичная схема совместного использования выводов C2-интерфейса приведена на рис.21.6.

**Рисунок 21.6. Разделение выводов интерфейса C2**



Конфигурация, приведенная на рис.21.6, предполагает следующее:

1. Пользовательский вход (b) не может изменять свое состояние, пока целевой МК остановлен.
2. Вывод /RST целевого МК используется только как вход.

В зависимости от конкретного приложения могут потребоваться дополнительные резисторы.

---

## Contact Information

### Silicon Laboratories Inc.

4635 Boston Lane

Austin, TX 78735

Tel: 1+(512) 416-8500

Fax: 1+(512) 416-9669

Toll Free: 1+(877) 444-3032

Email: [productinfo@silabs.com](mailto:productinfo@silabs.com)

Internet: [www.silabs.com](http://www.silabs.com)

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Silicon Laboratories assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Silicon Laboratories assumes no responsibility for the functioning of undescribed features or parameters. Silicon Laboratories reserves the right to make changes without further notice. Silicon Laboratories makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Silicon Laboratories assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Silicon Laboratories products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Silicon Laboratories product could create a situation where personal injury or death may occur. Should Buyer purchase or use Silicon Laboratories products for any such unintended or unauthorized application, Buyer shall indemnify and hold Silicon Laboratories harmless against all claims and damages.

Silicon Laboratories and Silicon Labs are trademarks of Silicon Laboratories Inc. Other products or brandnames mentioned herein are trademarks or registered trademarks of their respective holder.