

АВОВ ПОЛУПРОВОДНИК  
8-БИТ ОДНОКРИСТАЛЬНЫЕ МИКРОКОНТРОЛЛЕРЫ.

**МС80F0704/0708**

**МС80F0804/0808**

*Руководство пользователя (Ver.  
1.03)*



---

**Версия 1.03**

**Опубликов  
ано  
командой  
FAE**

**©2006 ABOV semiconductor Co., Ltd. Все хорошо зарезервировано.**

---

Дополнительная информация этого руководства может быть предоставлена офисами полупроводников ABOV в Корее или дистрибьюторами и Репрецентаторами.

ABOV semiconductor оставляет за собой право вносить изменения в любую информацию здесь в любое время без уведомления.

Информация, диаграммы и другие данные в данном руководстве верны и достоверны; Однако ABOV semiconductor никоим образом не несет ответственности за какие-либо нарушения патентов или других прав третьей стороны, вызванные использованием данного руководства.

# ИСТОРИЯ ИЗМЕНЕНИЙ

## **ВЕРСИЯ 1.03 (31 октября 2011 г.)**

Логотип изменен.

Размеры контура 28 комплекта СОП фиксированы.

## **ВЕРСИЯ 1.02 (18 августа 2009 г.)**

Порядок устройства переставляют и фиксируют некоторые дефекты.

Цифры флэш-записи были обновлены в "1. ОБЗОР "на стр.1.

Напряжение считывания конфигурации ( $V_{CONFIG}$ ), максимальное напряжение запуска VDD ( $V_{START}$ ) и рис. 7-1 добавлены в «Электрические характеристики 7.4 постоянного тока» на стр. 17.

Рекомендуемые значения нагрузочного конденсатора для цепи главного генератора были добавлены к  $10\text{pF} \sim 30\text{pF}$  в "10. ГЕНЕРАТОРЫ ЧАСОВ - TOP "на стр. 43.

Если напряжение AVREF было меньше напряжения VDD для АЦП, таблица-примечание и примечание были добавлены в "7.3 Аналого-цифровой преобразователь Char-acteristics «на стр. 16 и» 14. АНАЛОГО-ЦИФРОВОЙ ПРЕОБРАЗОВАТЕЛЬ "на стр. 70.

## **ВЕРСИЯ 1.01 (3 ИЮНЯ 2008 Г.)**

Внутренняя спецификация OSC была изменена.

## **ВЕРСИЯ 1.0 (27 ФЕВРАЛЯ 2008 Г.)**

Добавлено минимальное напряжение параметра AVREF в «7.3 Характеристики аналого-цифрового преобразователя» на стр. 16. Добавлены графики характеристик в «7.6 Типичные характеристики» на странице 20.

## **ВЕРСИЯ 0.5 (SEP 28, 2007)**

Исправлена ошибка в описании и схеме 8-разрядного счетчика событий.

## **ВЕРСИЯ 0.4 (5 МАЯ 2007 Г.)**

Добавлено  $2.2\text{V} \sim 5.5\text{V} @ 1 \sim 4\text{MHz}$  в «7.2 Рекомендуемые условия эксплуатации» на стр. 16.

Исправлена ошибка в Рисунок 9-2 на стр. 38: изменены R04, R07 и EC0, EC1 PSR1 на R05, R06 и T00, T20

## **ВЕРСИЯ 0.3 (2 МАЯ 2007 Г.)**

Добавлено 28 пакетов QFN в "3. НАЗНАЧЕНИЕ PIN "на стр.5.

## **ВЕРСИЯ 0.21 (MAR. 2007)**

Изменен 28 чертеж пакета SOP в "4. ЧЕРТЕЖ УПАКОВКИ "на странице 6.

## **ВЕРСИЯ 0.2 (MAR. 2007)**

Добавлена спецификация параметров TVDD.

Добавлено примечание для опции конфигурации и исправлены некоторые ошибки.

## **ВЕРСИЯ 0.1 (AUG. 2006)**

Первое издание (предварительное)



# Оглавление

<b>1. ОБЗОР</b> .....	<b>1</b>	8-битный режим таймера/счетчика .....	53
Описание .....	1	16-битный режим таймера/счетчика .....	58
Особенности .....	1	8-разрядный выход сравнения (16-разрядный)	
Средства разработки .....	2	.....	59
Информация о заказе .....	3	8-битный режим захвата .....	59
<b>2. БЛОК-СХЕМА</b> .....	<b>4</b>	16-разрядный режим захвата .....	64
<b>3. НАЗНАЧЕНИЕ PIN</b> .....	<b>5</b>	Режим PWM .....	66
<b>4. ЧЕРТЕЖ КОМПЛЕКТНОЙ УСТАНОВКИ</b> .....	<b>6</b>	<b>14. АНАЛОГО-ЦИФРОВОЙ ПРЕОБРАЗОВАТЕЛЬ</b>	<b>70.</b>
<b>5. ФУНКЦИЯ PIN</b> .....	<b>10</b>	<b>15. ПОСЛЕДОВАТЕЛЬНЫЙ ВХОД/ВЫХОД (SIO)</b>	<b>73</b>
<b>6. ПОРТОВЫЕ КОНСТРУКЦИИ</b> .....	<b>12</b>	Синхронизация передачи/приема .....	74
<b>7. ЭЛЕКТРИЧЕСКИЕ ХАРАКТЕРИСТИКИ</b> .....	<b>16</b>	Использование последовательного ввода-	
Абсолютные максимальные оценки .....	16	вывода .....	76
Рекомендуемые условия эксплуатации .....	16	<b>16. ФУНКЦИЯ BUZZER</b> .....	<b>77</b>
Характеристики аналого-цифрового		<b>17. ПРЕРЫВАНИЯ</b> .....	<b>79</b>
преобразователя .....	16	Последовательность прерываний .....	82
Электрические характеристики постоянного		Прерывание BRK .....	83
тока .....	17	Множественное прерывание .....	83
Характеристики переменного тока .....	18	Внешнее прерывание .....	85
Типовые характеристики .....	20	<b>18. ЭНЕРГОСБЕРЕЖЕНИЕ</b> .....	<b>87</b>
<b>8. ОРГАНИЗАЦИЯ ПАМЯТИ</b> .....	<b>24</b>	Спящий режим .....	87
Регистры .....	24	Режим остановки .....	88
Память программы .....	27	Режим остановки во внутреннем режиме RC-	
Память данных .....	29	осциллированного сторожевого таймера ...	91
Режим адресации .....	34	Минимизация потребления тока .....	93
<b>9. ПОРТЫ ВВОДА-ВЫВОДА</b> .....	<b>38</b>	<b>19. СБРОС</b> .....	<b>95</b>
R0 и регистр R0IO .....	38	<b>20. ПРОЦЕССОР СБОЯ ПИТАНИЯ</b> .....	<b>97</b>
R1 и R1IO регистр .....	39	<b>21. КОНТРОЛЬ ПО БОРЬБЕ С ШУМОМ</b> .....	<b>99</b>
R2 и регистр R2IO .....	41	Защита от шумов колебаний .....	99
Регистр R3 и R3IO .....	42	Процессор ошибок колебаний .....	100
<b>10. ТАКТОВЫЙ ГЕНЕРАТОР</b> .....	<b>43</b>	<b>22. Область 101 конфигурации устройства.</b>	
Схема колебаний .....	43	<b>23. Эмулятор EVA. Настройка платы</b> .....	<b>102</b>
<b>11. ОСНОВНОЙ ТАЙМЕР ИНТЕРВАЛА</b> .....	<b>45</b>	<b>A. ИНСТРУКЦИЯ</b> .....	<b>ii</b>
<b>12. КОНТРОЛЬНЫЙ ТАЙМЕР</b> .....	<b>47</b>	Терминологический список .....	ii
<b>13. СЧЕТЧИК ТАЙМЕРА/СОБЫТИЙ</b> .....	<b>50</b>	Карта инструкций .....	iii
		Набор инструкций .....	iv



# MC80F0704/0708 MC80F0804/0808

## CMOS ОДНОКРИСТАЛЬНЫЙ 8-БИТНЫЙ МИКРОКОНТРОЛЛЕР С 10-БИТНЫМ КОНВЕРТЕРОМ А/D

### 1. Обзор

#### 1.1 Описание

MC80F0704/0708/0804/0808 представляет собой усовершенствованный 8-битный микроконтроллер CMOS с 4К/8К байтами флэш-памяти (МТР). Это мощный микроконтроллер, который обеспечивает высокую гибкость и экономичность решения для многих встраиваемых приложений управления. Это обеспечивает следующие функции: 8 КБ флэш-памяти, 256 байт ОЗУ, 8/16-битный таймер/счетчик, контрольный таймер, 10-битный аналого-цифровой преобразователь, 8-битный последовательный вход/выход, порт управления зуммером, 10-битный выход PWM и встроенный генератор и тактовая схема. Он также имеет ONP, фильтр шума, PFD для повышения помехоустойчивости. Кроме того, MC80F0704/0708/0804/0808 поддерживает режимы энергосбережения для снижения энергопотребления.

В этом документе **поясняются базовые MC80F0808**, функции, исключенные из других, те же, что и в таблице ниже.

Имя устройства	Размер флэш-памяти	RAM	ADC	ПОРТ ВВОДА-ВЫВОДА	Пакет
MC80F0704G/0708G	4К/8К	256В	16 канал	26 порт	28 SKDIP
MC80F0704D/0708D					28 СОП
MC80F0704U/0708U					28 QFN
MC80F0804K/0808K				30 портов	32 SDIP
MC80F0804D/0808D					32 SOP

#### 1.2 Особенности

- **8К/4К Bytes On-Chip Code FLASH (МТР)**
- **Флэш-память**
  - Выносливость: 100 циклов
  - Время хранения данных: 10 лет
- **256 байт оперативной памяти (встроенная память стека)**
- **Минимальное время выполнения инструкции:**
  - 333 нс при 12MHz (команда NOP)
- **Программируемые контакты ввода/вывода**  
(Прямой привод светодиодов может быть источником и поглотителем)
  - MC80F0804/0808: 29/30
  - MC80F0704/0708: 25/26
- **Один 8-битный таймер основного интервала**
- **Четыре 8-битных таймера/счетчика (или два 16-битных таймера/счетчика)**
- **Один контрольный таймер**
- **Один 8-разрядный последовательный коммуникационный интерфейс:**
  - Один последовательный вход/выход (SIO)
- **Два 10-разрядных высокоскоростных выхода PWM**
- **10-битный аналого-цифровой преобразователь: 16 каналов**

- **Один порт для вождения зуммера**
  - 488Hz ~ 250kHz @ 4MHz
- **Четыре входных порта внешнего прерывания**
- **On-chip POR (питание при сбросе)**
- **Двенадцать источников прерываний**
  - Внешний вход: 4
  - Таймер: 6
  - Аналого-цифровое преобразование: 1
  - SIO: 1
- **Встроенная схема помехоустойчивости**
  - Фильтр шума
  - PFD (датчик отказа питания)
  - ONP (защита от шумов колебаний)
- **Режим отключения питания**
  - Режим остановки
  - Спящий режим
  - Режим внутреннего RC-осциллированного сторожевого таймера
- **Рабочее напряжение и частота**
  - 2.2V ~ 5.5V (при 1 ~ 4MHz)
  - 2.7V ~ 5.5V (при 1 ~ 6MHz)
  - 3.0V ~ 5.5V (при 1 ~ 8MHz)
  - 4.5V ~ 5.5V (при 1 ~ 12MHz)



- **Рабочая температура:** -40 ° C ~ 85 ° C
- **Тип осциллятора**
  - Кристалл
  - Керамический резонатор
  - Внешний RC-генератор (C может быть опущен)

- Внутренний осциллятор (4MHz/2MHz)
- **Пакет**
  - 28SKDIP, 28SOP, 28QFN, 32SKDIP, 32SDIP
  - Доступный бесплатный пакет Pb

### 1.3 Средства разработки

MC80F0704/0708/0804/0808 поддерживается полнофункциональным макросборщиком, компилятором HMS800 C, внутриконтурным эмулятором CHOICE-Dr.™ и программистами OTP/FLASH. Существует два различных типа программистов, таких как однотипный и бандитский. Макросборщик работает под управлением MS-Windows 95 и операционной системы Windows с повышением версии.

Пожалуйста, свяжитесь с продажной частью полупроводника ABOV.

Программное обеспечение	- Ассемблер на базе MS-Windows - Отладчик на базе MS-Windows - компилятор HMS800 C
Аппаратное обеспечение (эмулятор)	- ВЫБОР - Dr. - ВЫБОР - Dr. EVA80C0x B/D
Устройство записи FLASH	- PGM Plus USB (одиночный модуль записи) - Автономный GANG4 USB (автор банды) - ВЫБОР - SIGMA II (одиночный писатель)



Рис. 1-2 Выбор - Dr. (Эмулятор, интерфейс USB)



Рис. 1-1 PGM плюс USB (Single Writer)



Рис. 1-3 Автономный Gang4 USB (Gang Writer)



**1.4 Информация о заказе**

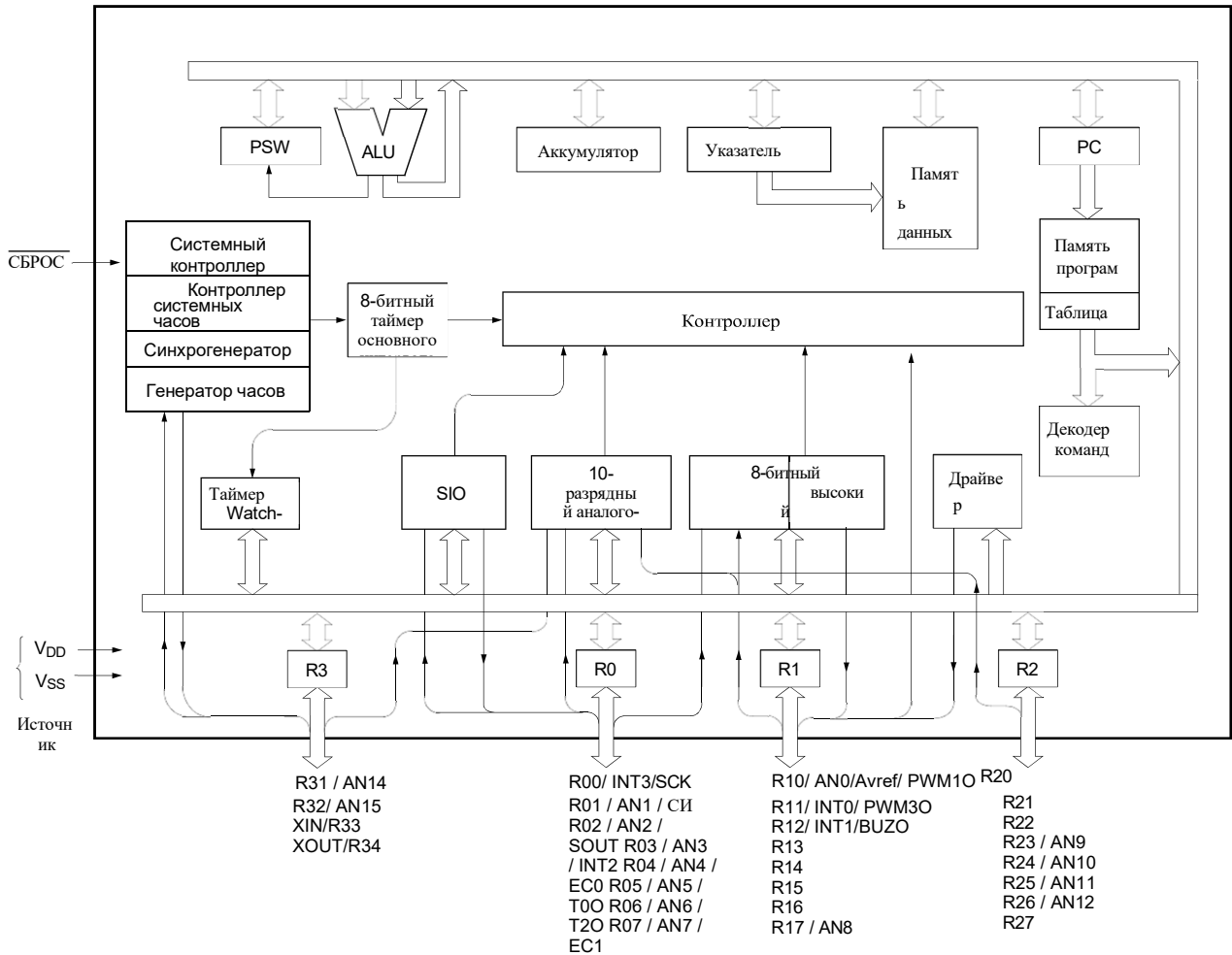
Имя устройства	ФЛЭШ-ПЗУ	RAM	Пакет
MC80F0704G MC80F0704D MC80F0704U MC80F0804K MC80F0804D	4К байт	256 байт	28SKDIP 28SOP 28QFN 32SDIP 32SOP
MC80F0708G MC80F0708D MC80F0708U MC80F0808K MC80F0808D	8К байт	256 байт	28SKDIP 28SOP 28QFN 32SDIP 32SOP

Бесплатный пакет Pb:

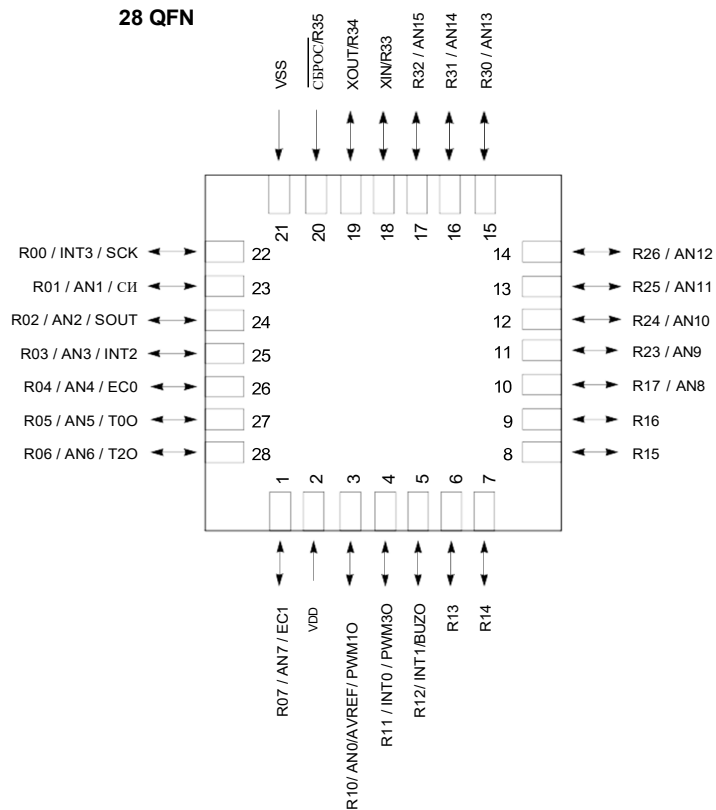
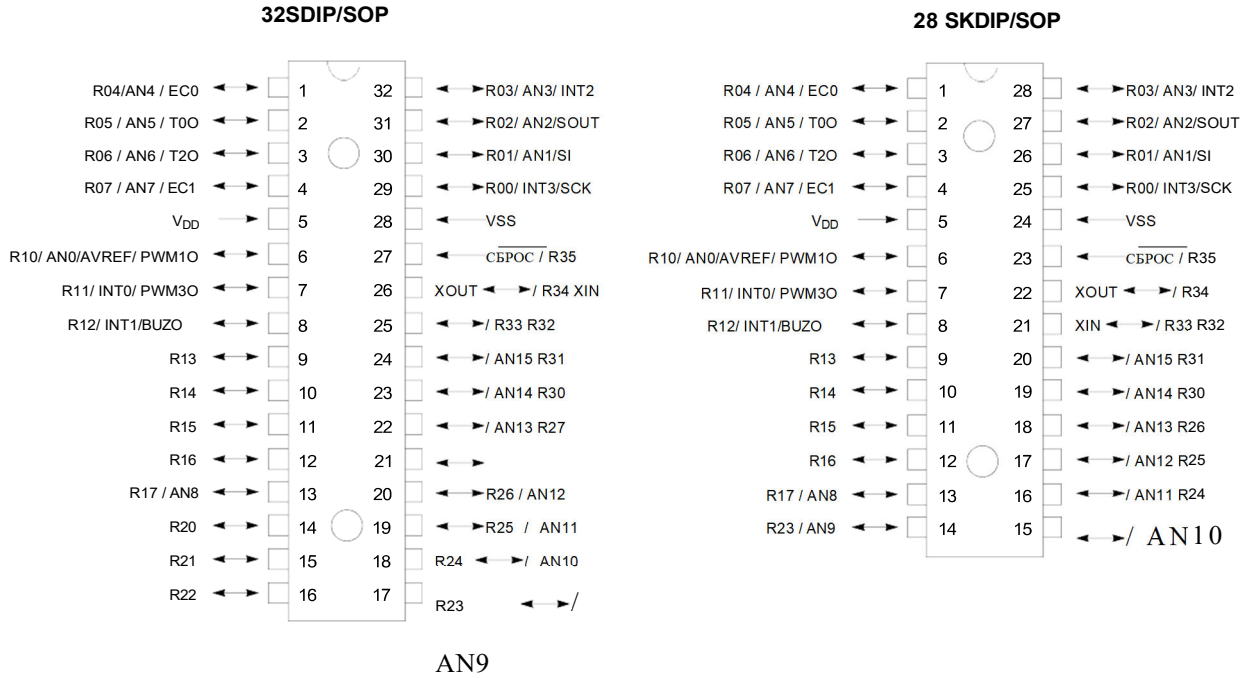
Суффикс «P» будет добавлен по исходному номеру детали.

Например,; MC80F0708G (обычный пакет), MC80F0708G P (бесплатный пакет)

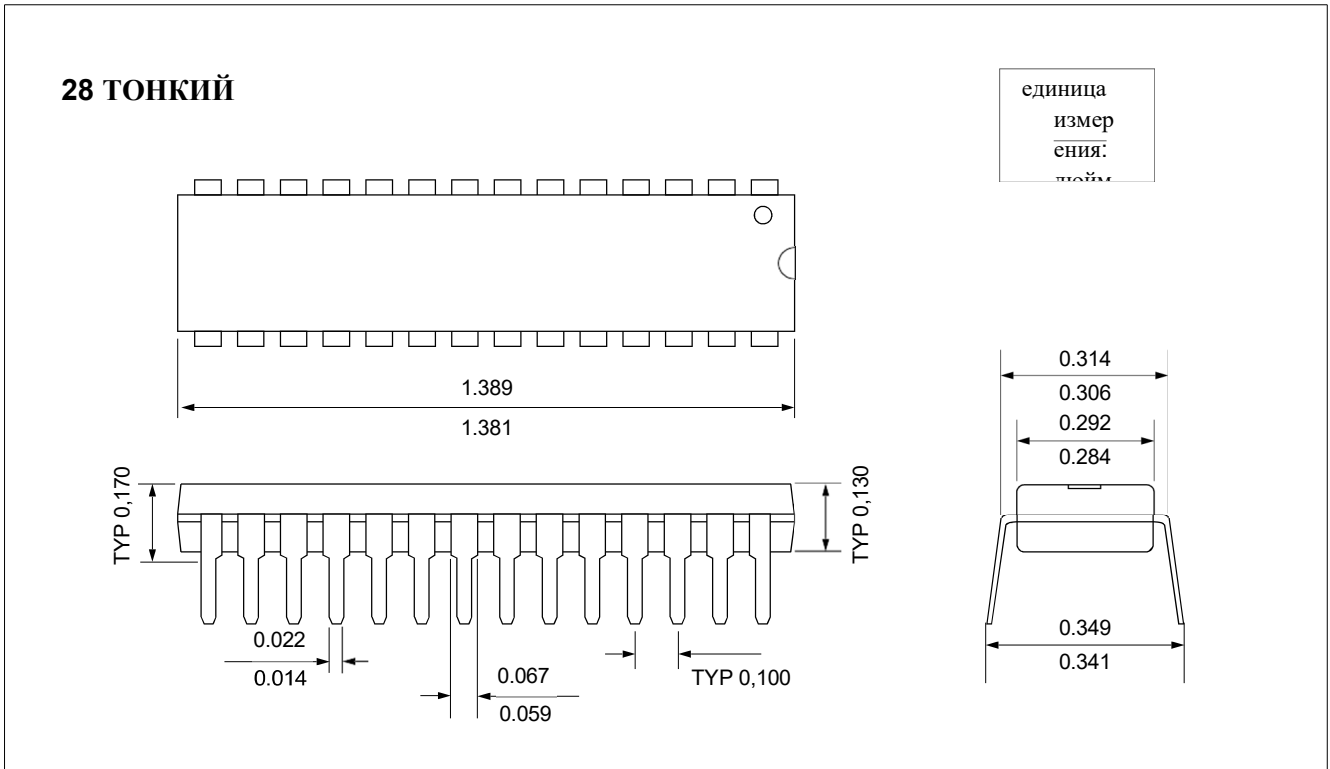
## 2. БЛОК-СХЕМА



### 3. НАЗНАЧЕНИЕ PIN

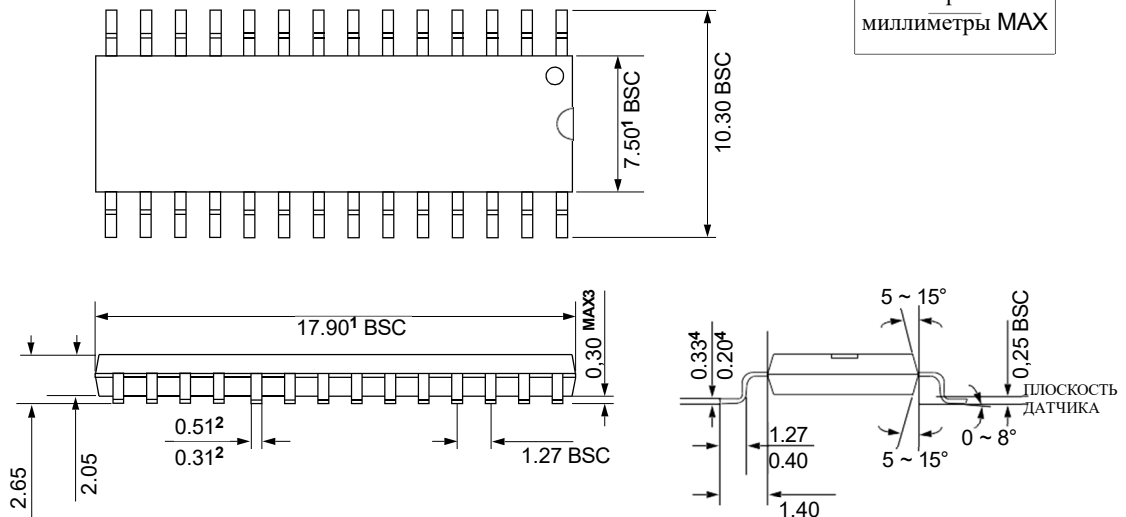


**4. ЧЕРТЕЖ КОМПЛЕКТНОЙ УСТАНОВКИ**



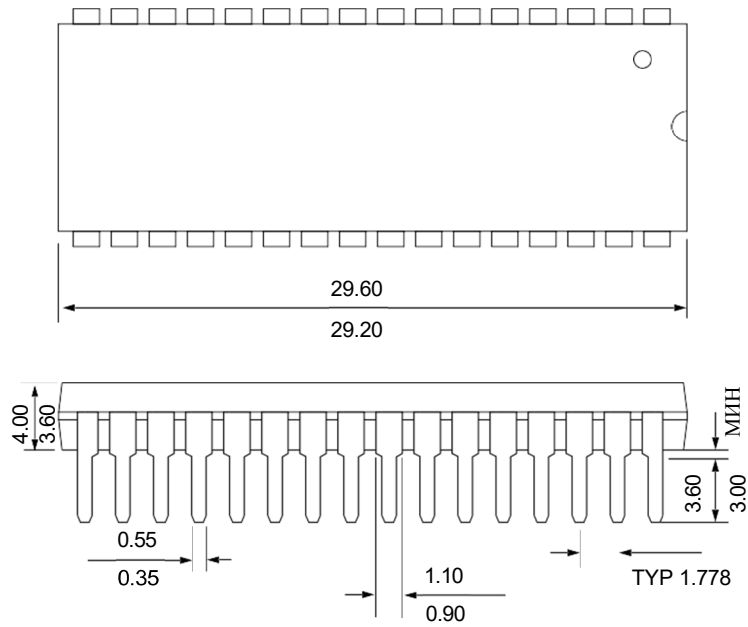
**28 СОП**

единица  
измерения:  
миллиметры MAX

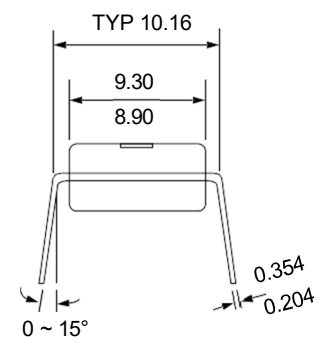


1. Размер 17,90 не включает флэш-форму, выступы или заусенцы затвора.  
 Вспышка пресс-формы, выступы или заусенцы затвора не должны превышать 0,15 мм на конец.  
 Размер 7,50 не включает в себя промежуточный FLASH или выступ.  
 Промежуточная вспышка или выступ не должны превышать 0,25 мм с каждой стороны. Верхняя часть упаковки может быть меньше нижней части упаковки.  
 Размеры 17,90 и 7,50 определяются в крайних крайних точках пластмассового корпуса, за исключением флэш-формы.  
 Обязать заусенцы, заусенцы затвора и промежуточную флэш-память, но с учетом любого несоответствия между верхней и нижней частями пластмассового корпуса.
2. Эти размеры применяются к плоскому сечению вывода от 0,10 до 0,25 мм от наконечника вывода. Размер не включает выступ дамбара.  
 Допустимый выступ демпфера должен составлять 0,10 мм, что превышает максимальный размер материала.  
 Дамбар может не располагаться на нижнем радиусе подложки.
3. Это определяется как вертикальное расстояние от установочной плоскости до нижней точки на корпусе

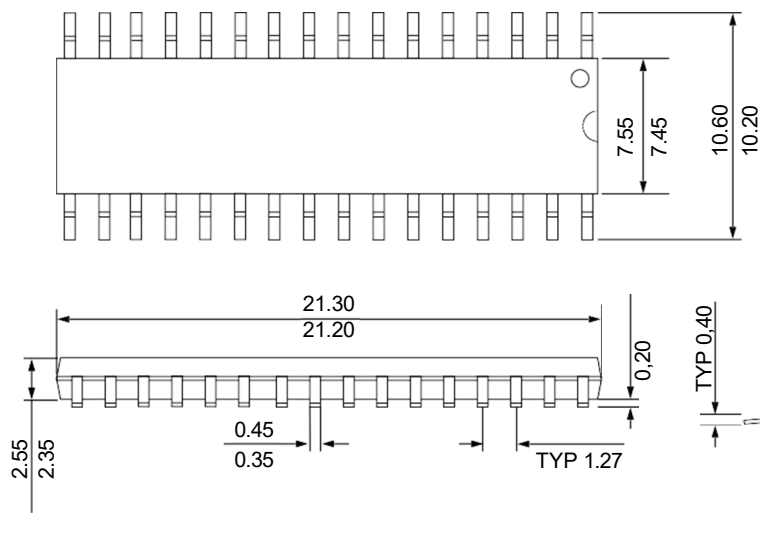
**32 SDIP**



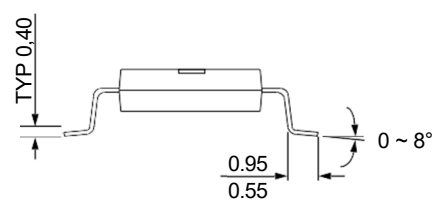
единица  
измерения:  
миллиметр MAX



**32 SOP**



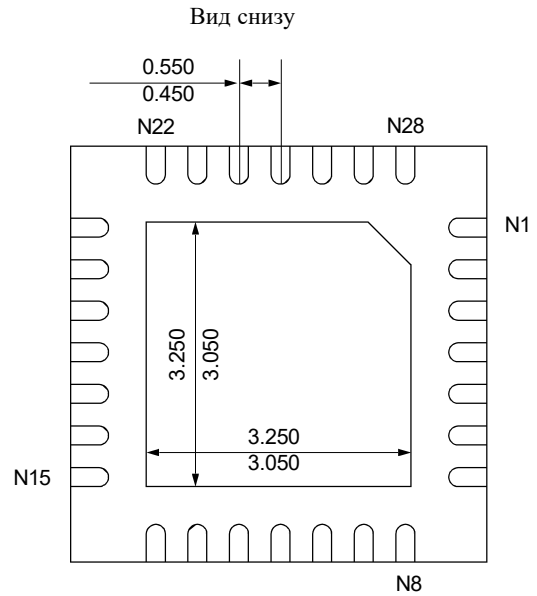
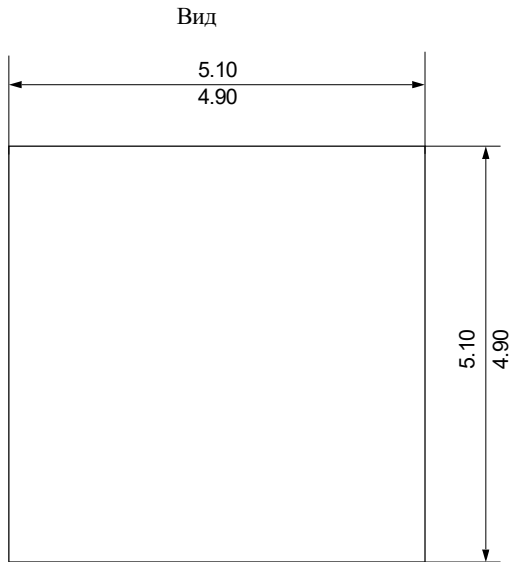
единица  
измерения:  
миллиметр MAX



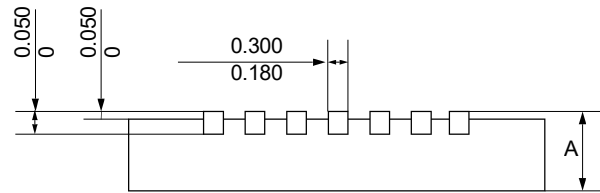


28 QFN

единица  
измерения:  
миллиметры MAX



A	МИН.	НОРМ А.	МАКС.
	0.700	0.750	0.800
0.800	0.850	0.900	



Вид сбоку

## 5. ФУНКЦИЯ PIN

**V<sub>DD</sub>**: Напряжение

питания. **V<sub>SS</sub>**:

Заземление цепи.

**RESET**: Сброс MCU.

**X<sub>IN</sub>**: Вход в усилитель инвертирующего генератора и вход во внутреннюю рабочую схему основного тактового генератора.

Выход усилителя инвертирующего генератора.

**R00 ~ R07**: R0 - 8-разрядный, CMOS, двунаправленный порт ввода-вывода. Контакты R0 могут использоваться в качестве выходов или входов в соответствии с «1» или «0», записанными в их регистре направления порта (R0IO).

Контакт порта	Альтернативная функция
R00	INT3 (Входной Port3 внешнего прерывания) SCK (SPI CLK)
R01	AN1 (порт аналогового входа 1) SI (последовательный ввод данных SPI)
R02	AN2 (порт аналогового входа 2) SOUT (вывод последовательных данных SPI)
R03	AN3 (порт аналогового входа 3) INT2 (Входной Port2 внешнего прерывания)
R04	AN4 (порт аналогового входа 4) EC0 (Источник ввода счетчика событий 0)
R05	AN5 (порт аналогового входа 5) T00 (Timer0 тактовый сигнал)
R06	AN6 (порт аналогового входа 6) T20 (Timer2 тактовый сигнал)
R07	AN7 (порт аналогового входа 7) EC1 (Источник ввода счетчика событий 1)

Таблица 5-1 Порт R0

Кроме того, R0 выполняет функции различных специальных функций в таблице 5-1.

**R10 ~ R17**: R1 является 8-разрядным CMOS двунаправленным портом ввода-вывода. Контакты R1 могут использоваться в качестве выходов или входов в соответствии с «1» или «0», записанными в их регистре направления порта (R1IO).

R1 выполняет функции различных следующих специальных функций

вТаблица 5-2

Контакт порта	Альтернативная функция
R10	AN0 (порт аналогового входа 0) AVref (внешний аналоговый опорный контакт) PWM1O (PWM1 выход)
R11	INT0 (Входной порт внешнего прерывания 0) PWM3O (Выходной сигнал PWM3)
R12	INT1 (входной порт внешнего прерывания 1) BUZ (выходной порт управления зуммером)
R13	-
R14	-
R15	-
R16	-
R17	-
	AN8 (порт аналогового входа 8)

Таблица 5-2 Порт R1

R20 ~ R27: R2 является 8-разрядным CMOS-портом двунаправленного ввода-вывода. Контакты R2 могут использоваться в качестве выходов или входов в соответствии с «1» или «0», записанными в их регистре направления порта (R2IO)

Кроме того, R2 выполняет функции различных специальных функций в таблице 5-3.

Контакт порта	Альтернативная функция
R20	-
R21	-
R22	-
R23	AN9 (аналоговый входной порт 9) AN10 (аналоговый входной порт 10) AN11 (аналоговый входной порт 11)
R24	AN12 (аналоговый входной порт 12)
R25	-
R26	-
R27	-

Таблица 5-3 Порт R2

**R31 ~ R35**: R3 - 6-разрядный CMOS-порт двунаправленного ввода-вывода. Контакты R3 могут использоваться в качестве выходов или входов в соответствии с «1» или «0», записанными в их регистре направления порта (R3IO).

R3 выполняет функции последовательного интерфейса, описанные в таблице 5-4.

Контакт порта	Альтернативная функция
R30	AN13 (порт аналогового входа 13)
R31	AN14 (порт аналогового входа 14)
R32	AN15 (порт аналогового входа 15)

R33	X <sub>IN</sub> (вход колебаний)
R34	X <sub>OUT</sub> (выход колебаний)
R35	RESETB (входной порт сброса)

**Таблица 5-4 Порт R3**

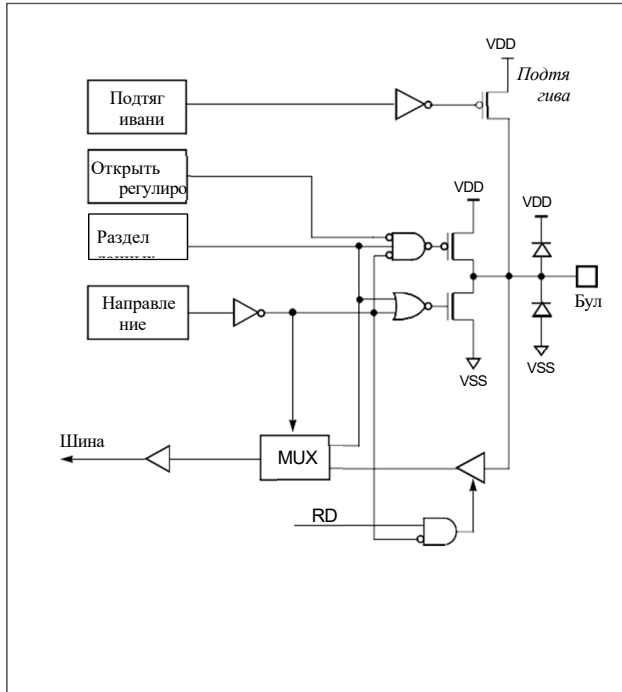
ПИН-КОД	Номер контакта			Вход/Выход	Функция			
	32	28	28 QFN		Сначала	Второй	Третий	Дальше
VDD	5	5	2	-	Напряжение питания			
VSS	28	24	21	-	Заземление цепи			
СБРОС (R35)	27	23	20	Я	Вход сигнала сброса	Порт только для ввода	-	-
X <sub>IN</sub> (R33)	25	21	18	Я	Вход колебаний	Обычный порт ввода-вывода	-	-
X <sub>OUT</sub> (R34)	26	22	19	О	Выход колебаний	Обычный порт ввода-вывода	-	-
R00 (INT3/SCK)	29	25	22	ВВОД-ВЫВОД	Обычные порты ввода-вывода	Внешнее прерывание 3	Вход синхросигнала SPI	-
R01 (AN1/SI)	30	26	23	ВВОД-ВЫВОД		Порт аналогового входа 1	Ввод данных SPI	-
R02 (AN2/SOUT)	31	27	24	ВВОД-ВЫВОД		Порт аналогового входа 2	Вывод данных SPI	-
R03 (AN3/INT2)	32	28	25	ВВОД-ВЫВОД		Порт аналогового входа 3	Внешние Interrupt2	-
R04 (AN4/EC0)	1	1	26	ВВОД-ВЫВОД		Порт аналогового входа 4	Счетчик событий	-
R05 (AN5/T00)	2	2	27	ВВОД-ВЫВОД		Порт аналогового входа 5	Timer0 Выходные данные	-
R06 (AN6/T20)	3	3	28	ВВОД-ВЫВОД		Порт аналогового входа 6	Timer2 Выходные данные	-
R07 (AN7/EC1)	4	4	1	ВВОД-ВЫВОД		Порт аналогового входа 7	Счетчик событий	-
R10 (AN0/AVref/PWM10)	6	6	3	ВВОД-ВЫВОД		Порт аналогового входа 0	Аналоговый эталон	Выход PWM 1
R11 (INT0/PWM30)	7	7	4	ВВОД-ВЫВОД		Внешнее прерывание 0	Выход PWM 3	-
R12 (INT1/BUZO)	8	8	5	ВВОД-ВЫВОД		Внешнее прерывание 1	Выходной сигнал управления зуммером	-
R13	9	9	6	ВВОД-ВЫВОД		-	-	-
R14	10	10	7	ВВОД-ВЫВОД		-	-	-
R15	11	11	8	ВВОД-ВЫВОД		-	-	-
R16	12	12	9	ВВОД-ВЫВОД		-	-	-
R17	13	13	10	ВВОД-ВЫВОД		Порт аналогового входа 8	-	-
R20	14	-	-	ВВОД-ВЫВОД		-	-	-
R21	15	-	-	ВВОД-ВЫВОД		-	-	-
R22	16	-	-	ВВОД-ВЫВОД		-	-	-
R23	17	14	11	ВВОД-ВЫВОД		Порт аналогового входа 9	-	-
R24	18	15	12	ВВОД-ВЫВОД		Порт аналогового входа 10	-	-

R25	19	16	13	ВВОД-ВЫВОД	Порт аналогового входа 11	-	-
R26	20	17	14	ВВОД-ВЫВОД	Порт аналогового входа 12	-	-
R27	21	-	-	ВВОД-ВЫВОД	-	-	-
R30 (AN13)	22	18	15	ВВОД-ВЫВОД	Порт аналогового входа 13	-	-
R31 (AN14)	23	19	16	ВВОД-ВЫВОД	Порт аналогового входа 14	-	-
R32 (AN15)	24	20	17	ВВОД-ВЫВОД	Порт аналогового входа 15	-	-

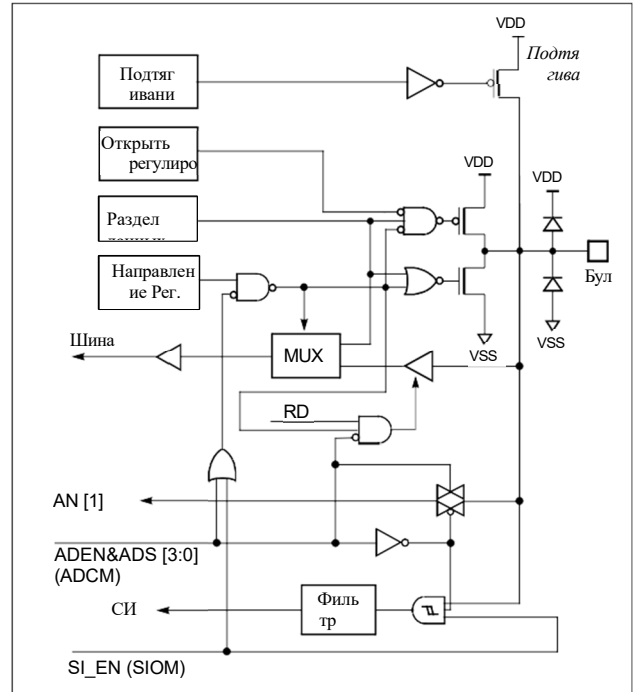
**Таблица 5-5 Описание контактов**

## 6. КОНСТРУКЦИИ ПОРТОВ

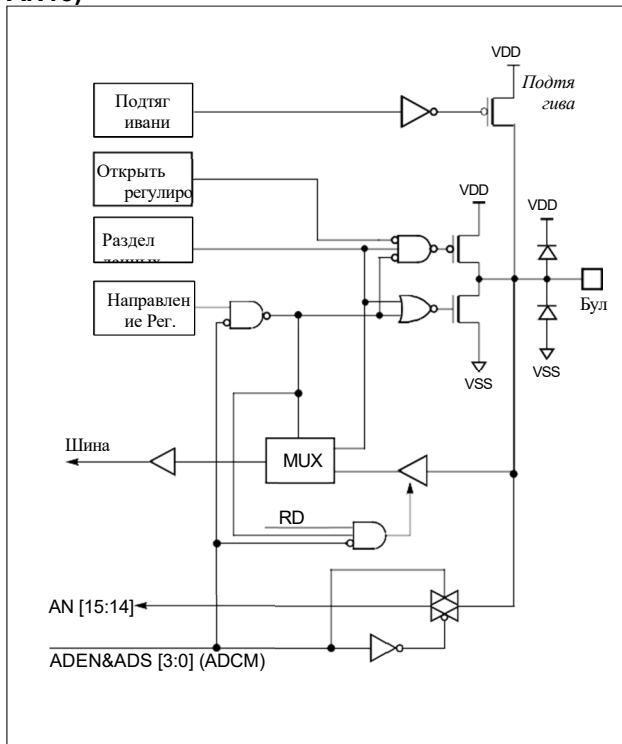
R13~R16, R20~R22, R27



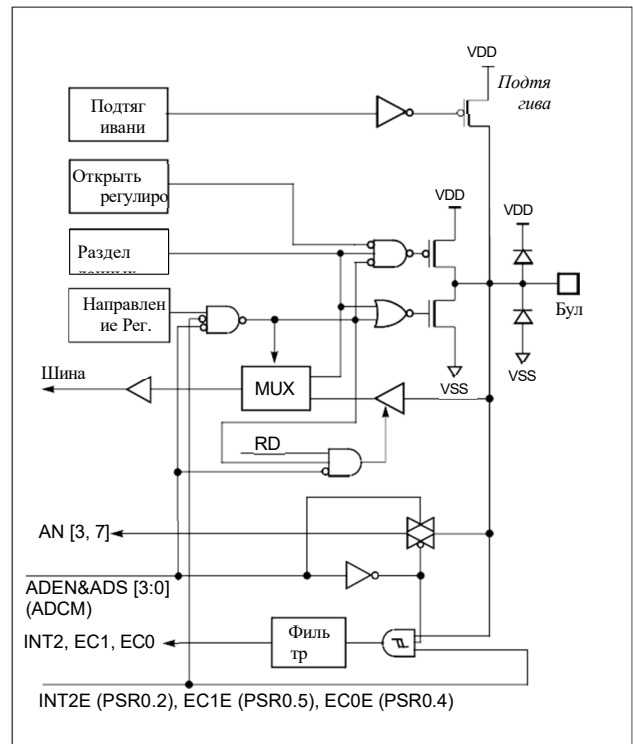
R01 (AN1/SI)



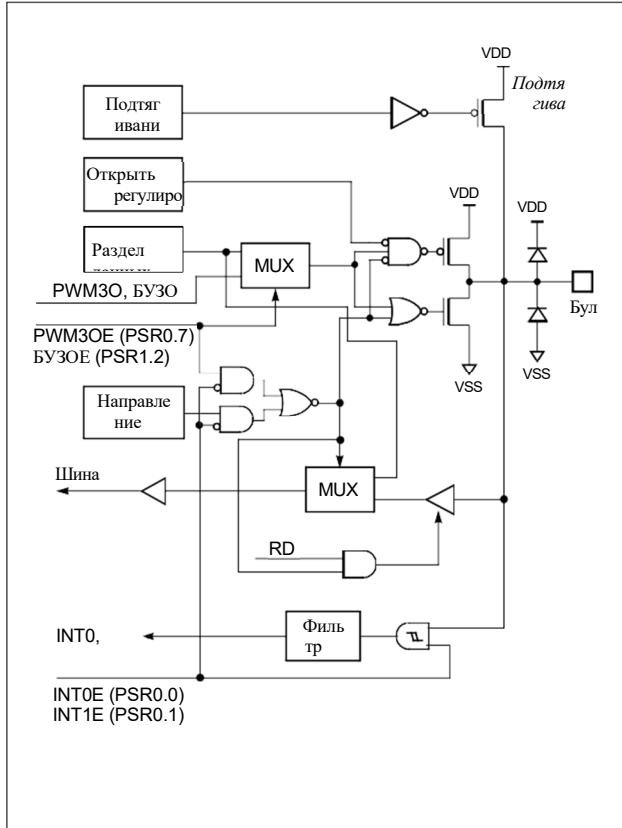
R17, R30~R32, R23~R26 (AN8 ~ AN15)



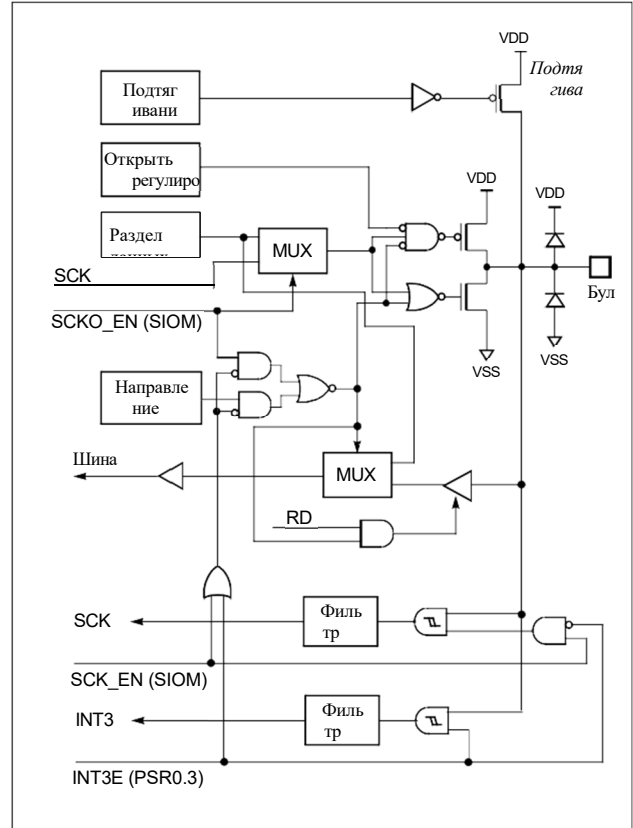
R03 (AN3/INT2), R04 (AN4/EC0), R07 (AN7/EC1)



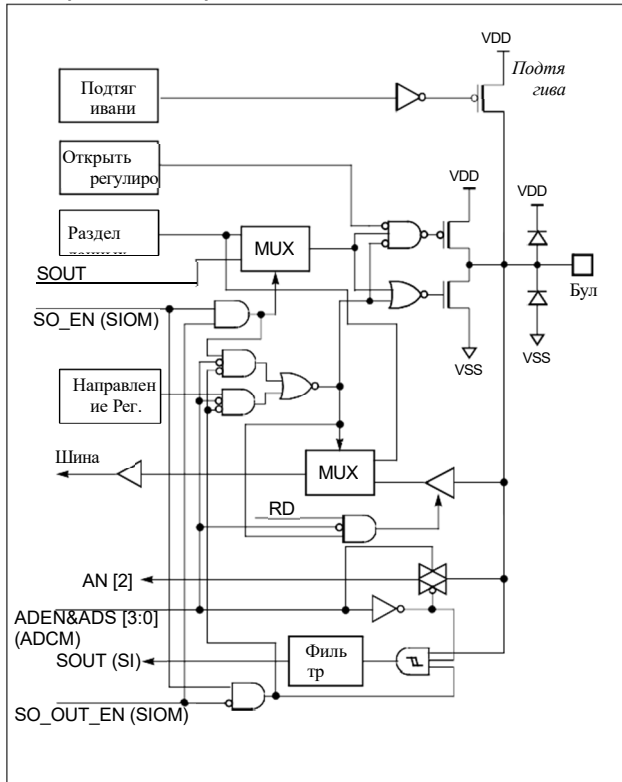
**R11 (INT0/PWM3O), R12 (INT1/BUZO)**



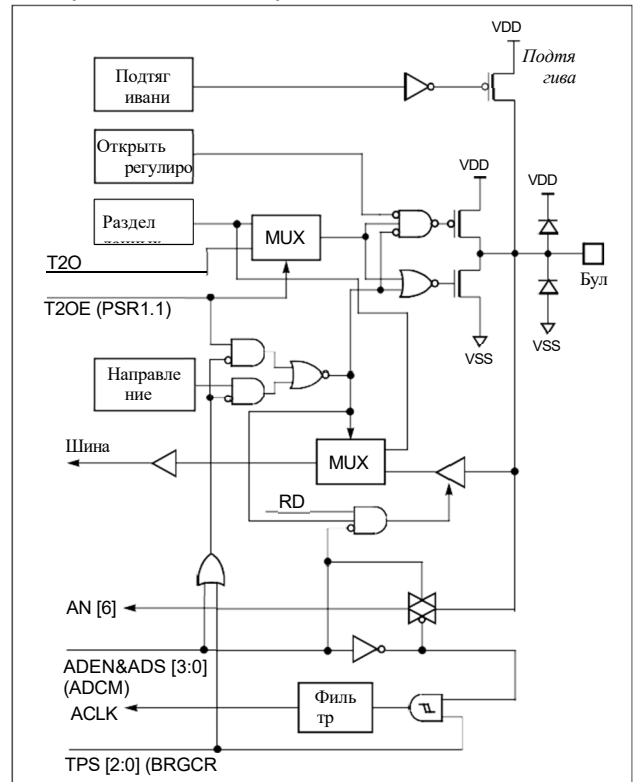
**R00 (INT3/SCK)**



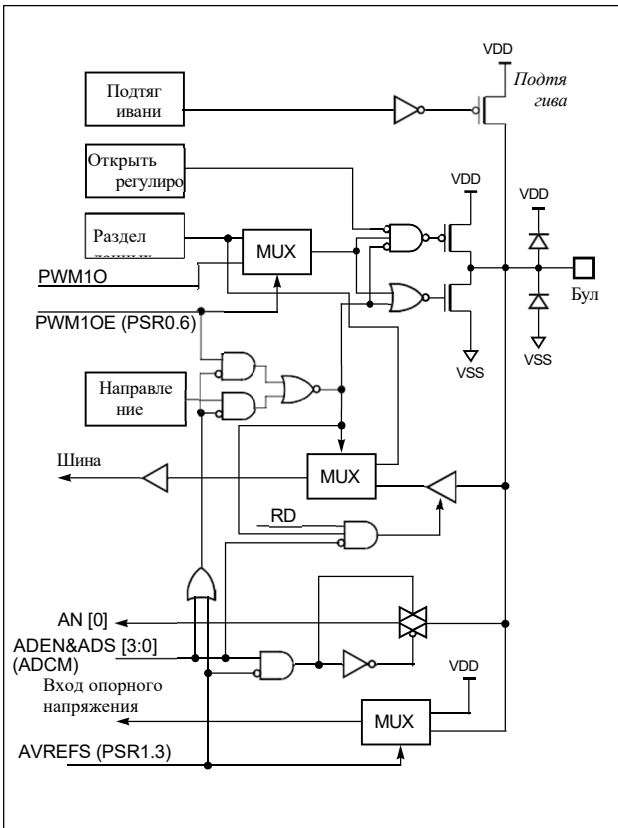
**R02 (AN2/SOUT)**



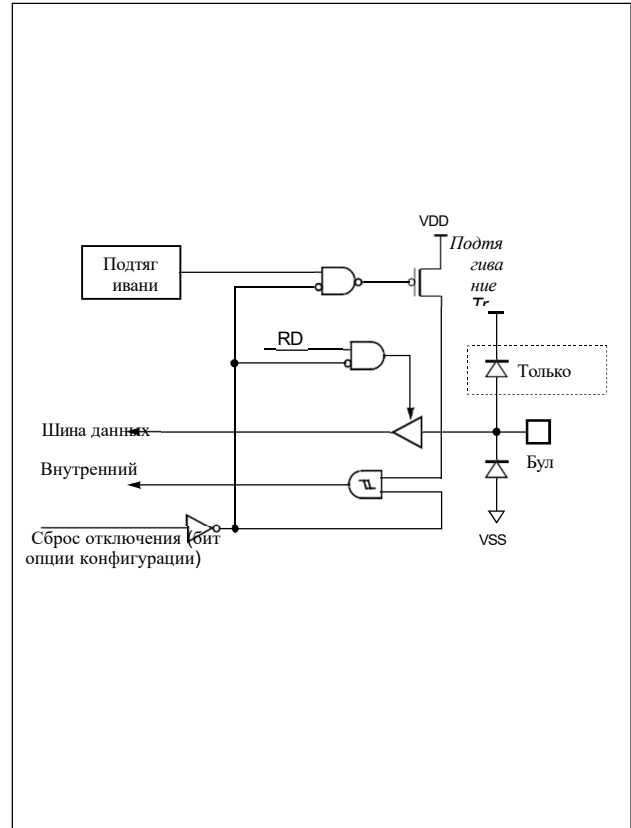
**R06 (AN6/ T2O/ACLK)**



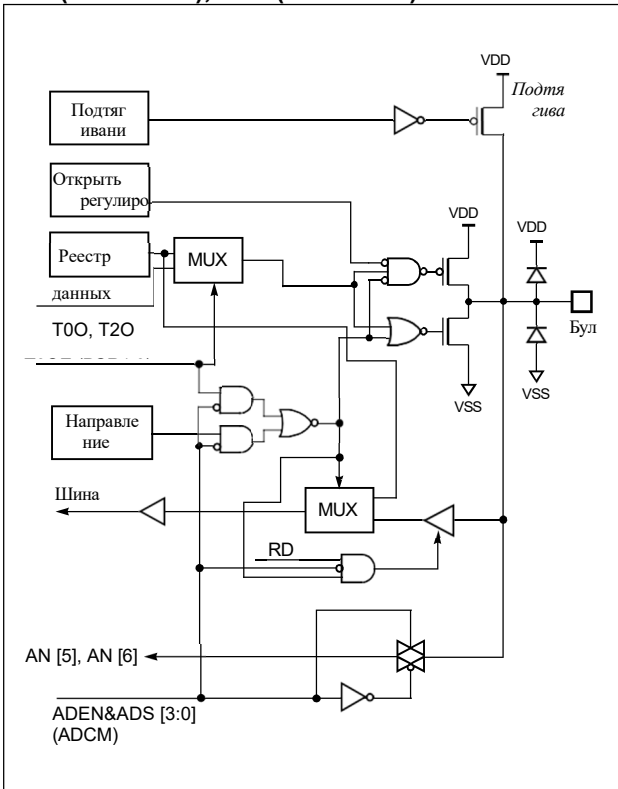
**R10 (AN0/AVREF/ PWM10)**



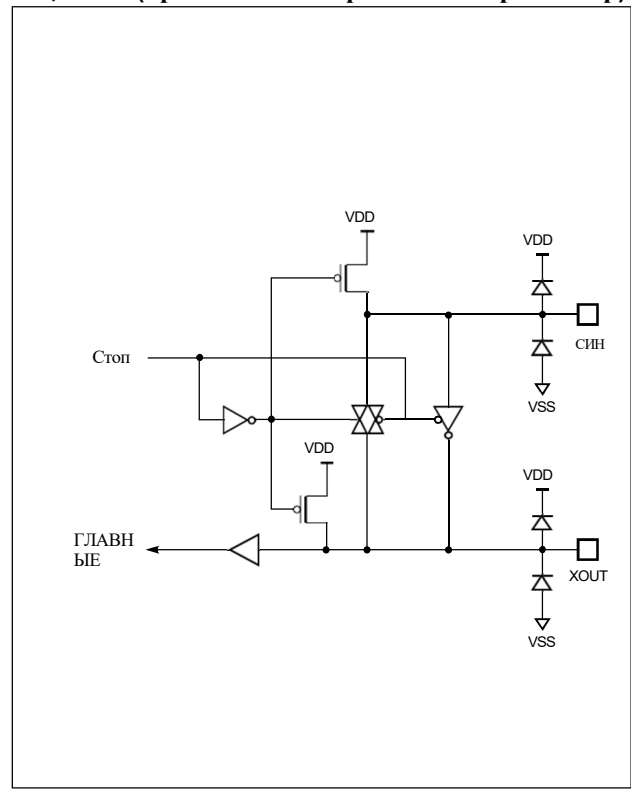
**СБРОС**



**R05 (AN5 / T00), R06 (AN6 / T20)**

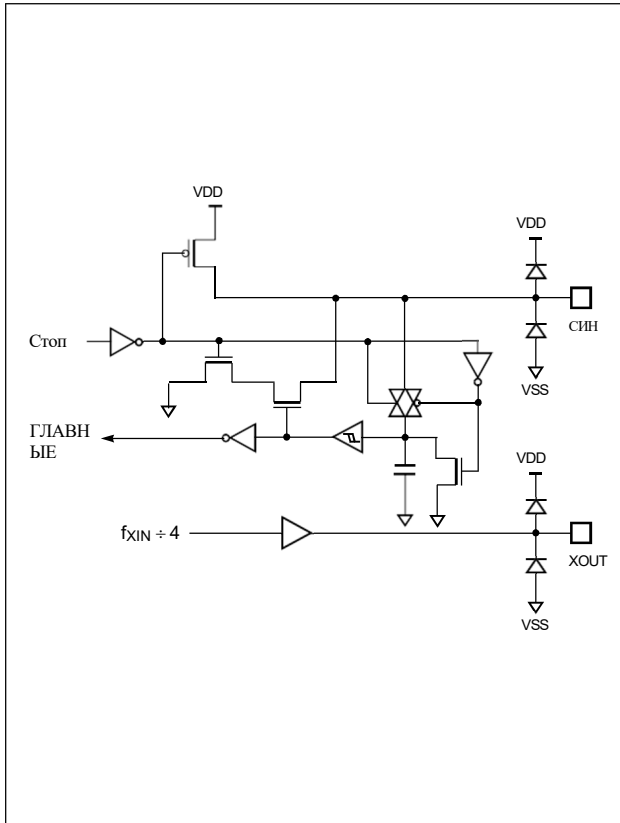


**XIN, XOUT (кристалл или керамический резонатор)**

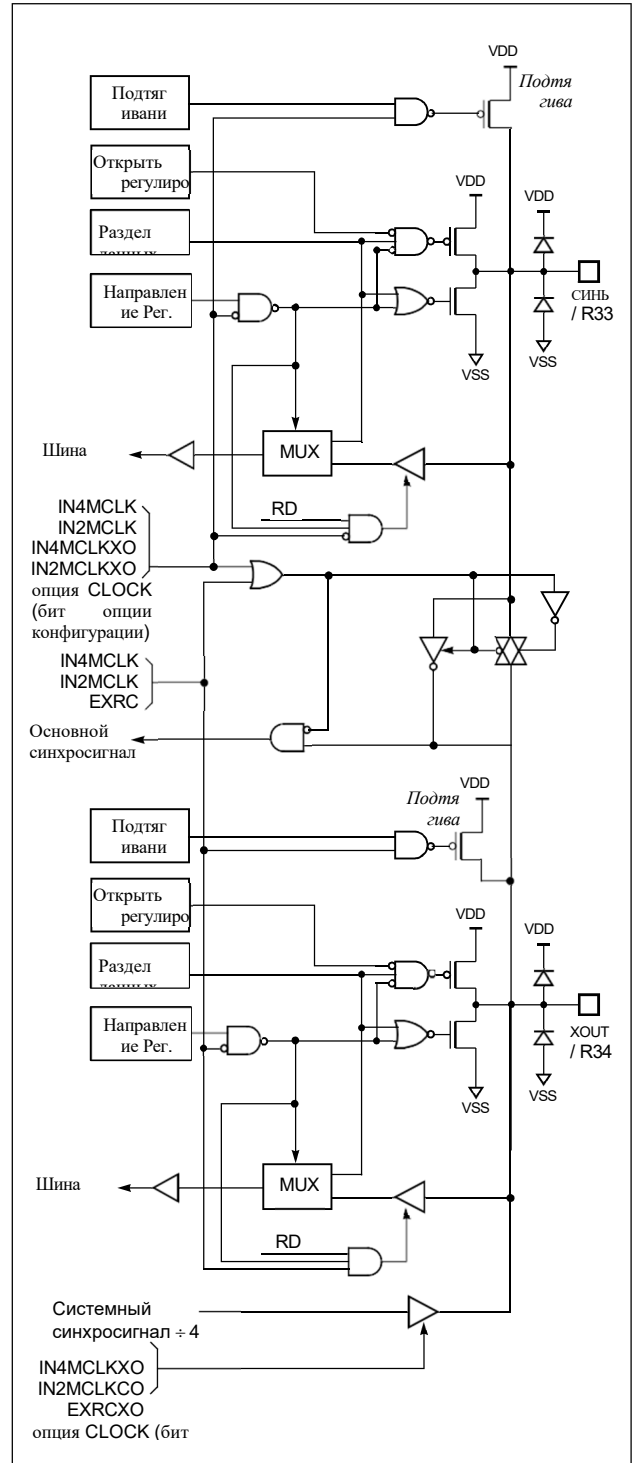




**X<sub>IN</sub>, X<sub>OUT</sub> (внешнее колебание RC или R)**



**R33 (X<sub>IN</sub>), R34 (X<sub>OUT</sub>)**



## 7. ЭЛЕКТРИЧЕСКИЕ ХАРАКТЕРИСТИКИ

### 7.1 Абсолютные максимальные оценки

Напряжение питания ..... до + 6,0 В	..... 10 мА
Температура хранения от ..... -65 до + 125 °С	Максимальный ток ( $\Sigma I_{OL}$ ) ..... 160 мА
Напряжение на любом выводе относительно заземления ( $V_{SS}$ ) ..... -0.3 на $V_{DD} + 0,3$ В	Максимальный ток ( $\Sigma I_{OH}$ ) ..... 80 мА
Максимальный ток из контакта $V_{SS}$ ..... 180 мА	
Максимальный ток в контакте $V_{DD}$ ..... 80 мА	
Максимальный потребляемый ток ( $I_{OL}$ на контакт ввода-вывода) ..... 20 мА	
Максимальный выходной ток ( $I_{OH}$ на контакт ввода-вывода)	

**Примечание:** Напряжения, превышающие значения, указанные в разделе «Абсолютные максимальные значения», могут привести к постоянному повреждению устройства. Это только номинальное напряжение и функциональная работа устройства при любых других условиях, превышающих указанные в эксплуатационных разделах настоящей спецификации, не подразумевается. Воздействие абсолютных максимальных рейтинговых условий в течение длительных периодов может повлиять на надежность устройства.

### 7.2 Рекомендуемые условия эксплуатации

Параметр	Символ	Состояние	Технические требования		Единица
			Мин.	Макс.	
Напряжение питания	VDD	$f_{XIN} = 1 \sim 12\text{MHz}$	4.5	5.5	V
		$f_{XIN} = 1 \sim 8\text{MHz}$	3.0	5.5	
		$f_{XIN} = 1 \sim 6\text{MHz}$	2.7	5.5	
		$f_{XIN} = 1 \sim 4\text{MHz}$	2.2	5.5	
Рабочая частота	$f_{XIN}$	$V_{DD} = 4,5 \sim 5.5\text{V}$	1	12	MHz
		$V_{DD} = 3,0 \sim 5.5\text{V}$	1	8	
		$V_{DD} = 2,7 \sim 5.5\text{V}$	1	6	
		$V_{DD} = 2,2 \sim 5.5\text{V}$	1	4	
Рабочая температура	TOPR	$V_{DD} = 2,7 \sim 5.5\text{V}$	-40	85	°C

### 7.3 Характеристики аналого-цифрового преобразователя

( $T_a = -40 \sim 85^\circ\text{C}$ ,  $V_{SS} = 0\text{V}$ ,  $V_{DD} = AV_{REF} = 5.12\text{V}$ ,  $3.072\text{V}$ )

Параметр	Символ	Условия	Мин.	Тип.	Макс.	Единица
Резолюция		-	-	10	-	БИТ
Общая точность	-	-	-	-	$\pm 3$	LSB
Ошибка нелинейности	NNLE	$f_{XIN} = 4 \text{ МГц}$	-	-	$\pm 3$	LSB
Дифференциальная ошибка НЕлинейности	NDNLE		-	-	$\pm 3$	LSB
Ошибка смещения нуля	NZOE		-	-	$\pm 3$	LSB
Полномасштабная ошибка	NFSE		-	-	$\pm 3$	LSB
Ошибка усиления	NGE		-	-	$\pm 3$	LSB
Время преобразования	TCONV		-	13	-	-
Напряжение аналогового входа	ТЩЕТНЫЙ	-	VSS	-	$V_{DD}$ ( $AV_{REF}$ )	V

Эталонное аналоговое напряжение	AVREF1	-	2.7 В	-	VDD	V
Ток преобразования	СИМВОЛ	$V_{DD} = 5.12V, f_{XIN} = 8 \text{ МГц}$	-	80	200	$\mu A$

1. Если напряжение  $AV_{REF}$  меньше напряжения  $V_{DD}$ , а входные контакты ANX, совместно используемые с различными альтернативными функциями, используются двунаправленными портами ввода/вывода, ток утечки может перетекать через контакт  $V_{DD}$  на контакт  $AV_{REF}$  в режиме высокого выхода или входные контакты ANX на контакт  $AV_{REF}$  в режиме высокого ввода.

#### 7.4 Электрические характеристики постоянного тока

( $T_A = -40 \sim 85^\circ\text{C}$ ,  $V_{DD} = 5,0\text{ В}$ ,  $V_{SS} = 0\text{ В}$ ),

Параметр	Символ	Булавка	Состояние	Технические требования			Единица
				Мин.	Тип.	Макс.	
Входное высокое напряжение	$V_{IH1}$	$X_{IN}, \overline{\text{СБРОС}}$		$0,8 V_{DD}$	-	$V_{DD}$	V
	$V_{IH2}$	Гистерезис Input <sup>1</sup>		$0,8 V_{DD}$	-	$V_{DD}$	
	$V_{IH3}$	Нормальный вход		$0,7 V_{DD}$	-	$V_{DD}$	
Входное низкое напряжение	$V_{IL1}$	$X_{IN}, \overline{\text{СБРОС}}$		0	-	$0,2 V_{DD}$	V
	$V_{IL2}$	Вход гистерезиса <sup>1</sup>		0	-	$0,2 V_{DD}$	
	$V_{IL3}$	Нормальный вход		0	-	$0,3 V_{DD}$	
Выходное высокое напряжение	$V_{OH}$	Все выходные порты	$V_{DD} = 5\text{ В}$ , $I_{OH} = -5\text{ мА}$	$V_{DD} - 1$	-	-	V
Выходное низкое напряжение	$V_{OL}$	Все выходные порты	$V_{DD} = 5\text{ В}$ , $I_{OL} = 10\text{ мА}$	-	-	1	V
Входной ток подтягивания	IP	Нормальный вход	$V_{DD} = 5\text{ В}$	-60	-	-150	$\mu\text{А}$
Ток высокой утечки на входе	I <sub>H1</sub>	Все контакты (за исключением $X_{IN}$ )	$V_{DD} = 5\text{ В}$	-	-	5	$\mu\text{А}$
	I <sub>H2</sub>	СИТЬ	$V_{DD} = 5\text{ В}$	-	-	13	$\mu\text{А}$
Входной низкий ток утечки	I <sub>L1</sub>	Все контакты (за исключением $X_{IN}$ )	$V_{DD} = 5\text{ В}$	-5	-	-	$\mu\text{А}$
	I <sub>L2</sub>	СИТЬ	$V_{DD} = 5\text{ В}$	-13	-	-	$\mu\text{А}$
Гистерезис	$ V_T $	Вход гистерезиса <sup>1</sup>	$V_{DD} = 5\text{ В}$	0.5	-	-	V
Напряжение PFD	$V_{PFD}$	$V_{DD}$		2.0	2.8	3.2	V
Внутренний период WDT RC	TRCWDT	XOUT	$V_{DD} = 5,0\text{ В}$	25	65	95	$\mu\text{С}$
Операционные Current <sup>2</sup>	IDD	$V_{DD}$	$V_{DD} = 5.5\text{ В}$ , $f_{XIN} = 12\text{ МГц}$	-	8.8	15	мама
Текущий спящий режим	ISLEEP	$V_{DD}$	$V_{DD} = 5.5\text{ В}$ , $f_{XIN} = 12\text{ МГц}$	-	1	2	мама
Режим RCWDT - аренда в режиме STOP	IRCWDT	$V_{DD}$	$V_{DD} = 5.5\text{ В}$ , $f_{XIN} = 12\text{ МГц}$	-	20	50	$\mu\text{А}$
Остановить текущий режим	ISTOP	$V_{DD}$	$V_{DD} = 5.5\text{ В}$ , $f_{XIN} = 12\text{ МГц}$	-	4	10	$\mu\text{А}$
Частота внутреннего 4MHz Oscil-	TIN4MCLK	XOUT	$V_{DD} = 5\text{ В}$ , $25^\circ\text{C}$	3.5	4	4.5	MHz
Частота внутреннего 2MHz Oscil-	TIN2MCLK	XOUT	$V_{DD} = 5\text{ В}$ , $25^\circ\text{C}$	1.75	2	2.25	MHz
Время нарастания $V_{DD}$	TVDD3	$V_{DD}$		-	-	40	мс/В

Напряжение запуска V <sub>DD</sub>	V <sub>START</sub> <sup>3</sup>	V <sub>DD</sub>		V <sub>SS</sub>	-	0.7	V
Напряжение считывания конфигурационного элемента	V <sub>CONFIG</sub> <sup>3</sup>	V <sub>DD</sub>	T <sub>VDD</sub> = 40 мс/В, V <sub>START</sub> = V <sub>SS</sub>	1.8	-	-	V
Сброс включения питания	V <sub>POR</sub>	V <sub>DD</sub>		2.4	2.9	3.4	V

Параметр	Символ	Булавка	Состояние	Технические требования			Единица
				Мин.	Тип.	Макс.	
Частота внешнего RC-генератора	$f_{RC-OSC}$	$f_{XOUT} = f_{RC-OSC} \div 4$	$V_{DD} = 5,5 \text{ В}$ $R=30\text{k}\Omega, C = 10 \text{ пФ}$	0.5	1.5	2.5	MHz
	$f_{R-OSC}$	$f_{XOUT} = f_{R-OSC} \div 4$	$V_{DD} = 5,5 \text{ В}, R=30\text{k}\Omega$	1	2	3	MHz

1. Вход гистерезиса: INT0 ~ INT3 (R11, R12, R03, R00), SIO (R00, R01, R02), EC0, EC1 (R04, R07)
2. Этот параметр измеряется во внутреннем режиме работы PROM в режиме ввода, определенном для всех портов ввода/вывода.
3. Эти параметры представлены только для руководства по проектированию и не испытываются и не гарантируются.

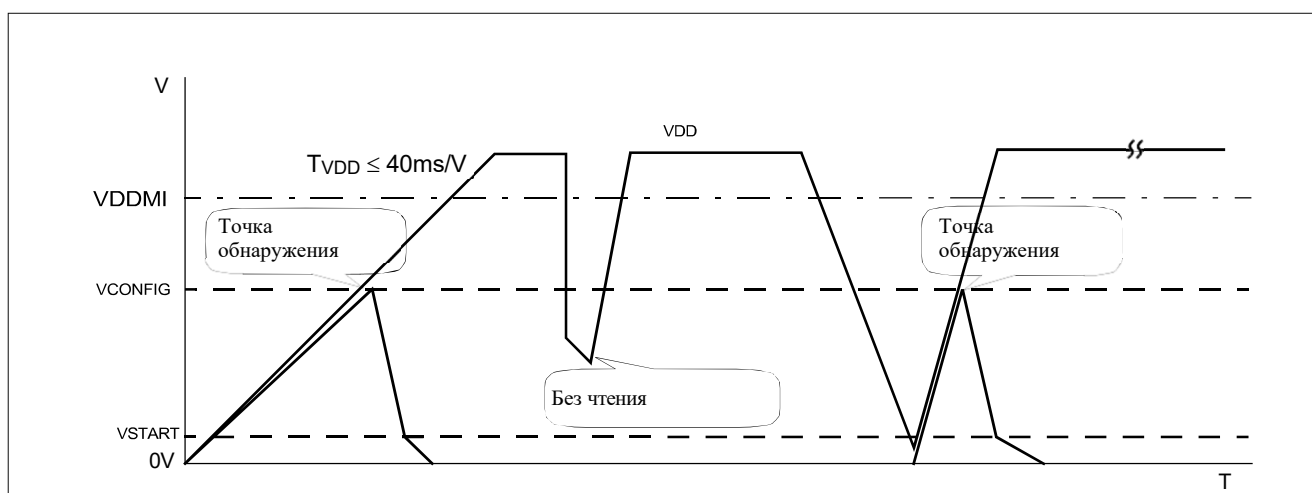


Рис. 7-1 Напряжение считывания конфигурации, включая POR и напряжение питания

### 7.5 Характеристики переменного тока

( $T_A = -40 \sim +85^\circ\text{C}$ ,  $V_{DD} = 5\text{V} \pm 10\%$ ,  $V_{SS} = 0\text{В}$ )

Параметр	Символ	Булавки	Технические требования			Единица
			Мин.	Тип.	Макс.	
Рабочая частота	$f_{CP}$	СИЛЬ	1	-	12	MHz
Ширина внешнего тактового импульса	$t_{CPW}$	СИЛЬ	35	-	-	не уточнено
Время перехода внешнего тактового генератора	$t_{RCP}, t_{FCP}$	СИЛЬ	-	-	20	не уточнено
Время стабилизации колебаний (4MHz)	$t_{ST}$	$X_{IN}, X_{OUT}$	-	-	20	mS
Ширина внешнего входного импульса	$t_{EPW}$	INT0, INT1, INT2, INT3 EC0, EC1	2	-	-	tSYS
Ширина ввода RESET	$t_{RST}$	$\overline{\text{СБРОС}}$	8	-	-	tSYS

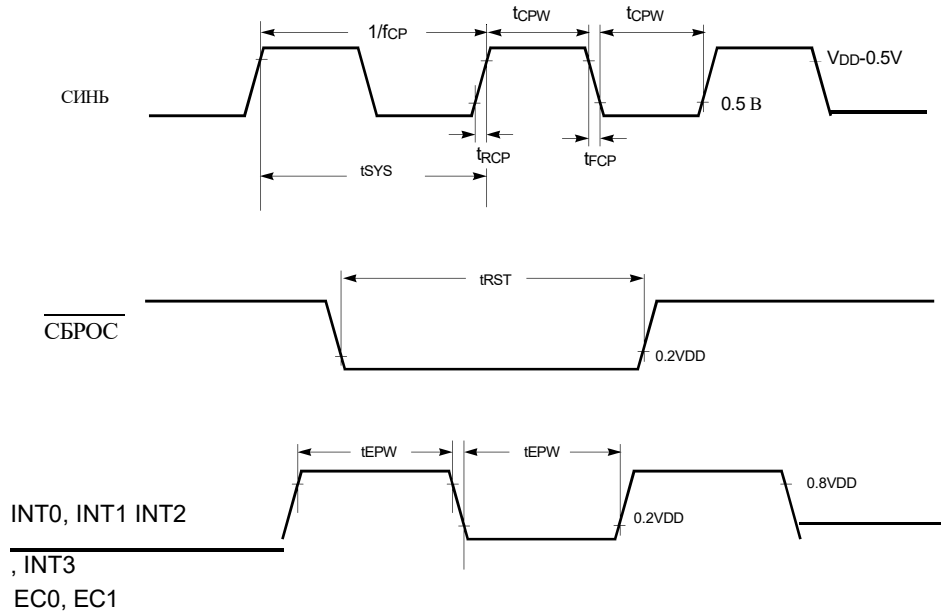


Рис. 7-2 Схема синхронизации

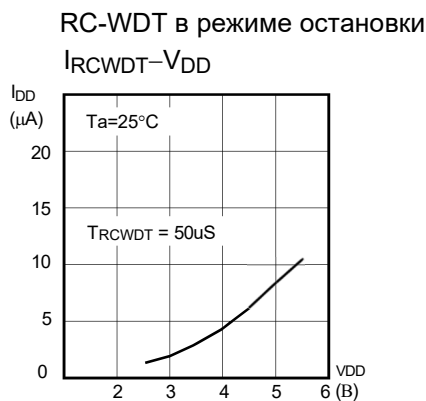
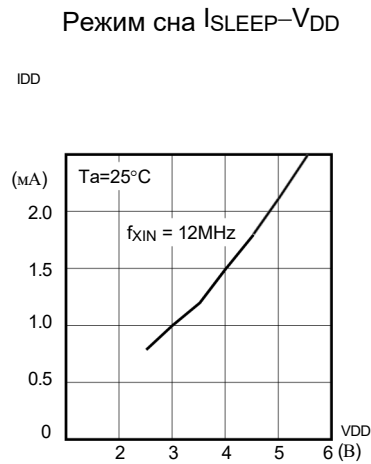
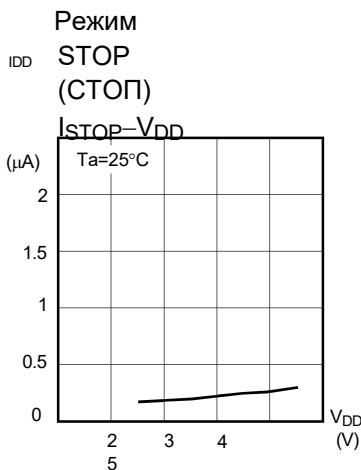
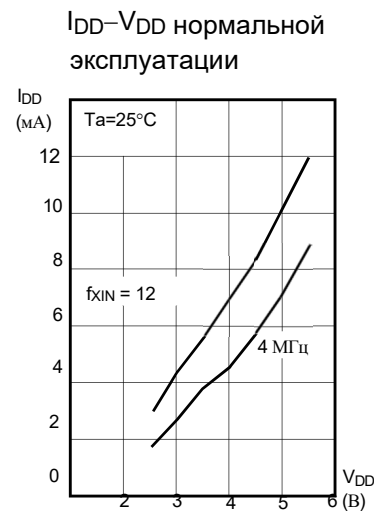
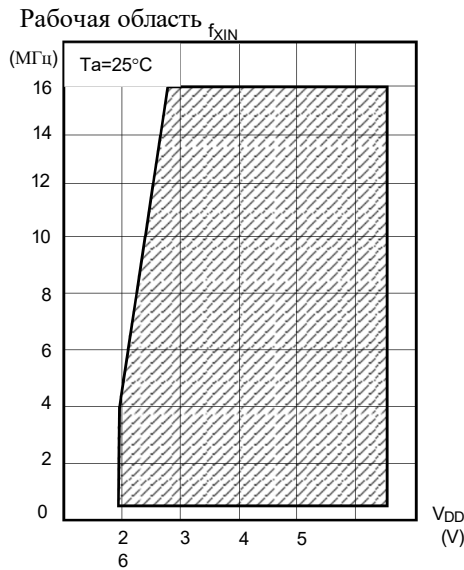
### 7.6 Типовые характеристики

Эти графики и таблицы, представленные в этом разделе, предназначены только для руководства по проектированию и не испытываются и не гарантируются.

**В некоторых графиках или таблицах данные представлены вне указанного рабочего диапазона (например, вне указанного диапазона VDD). Это только для информации, и устройства гарантируют правильную работу только в пределах**

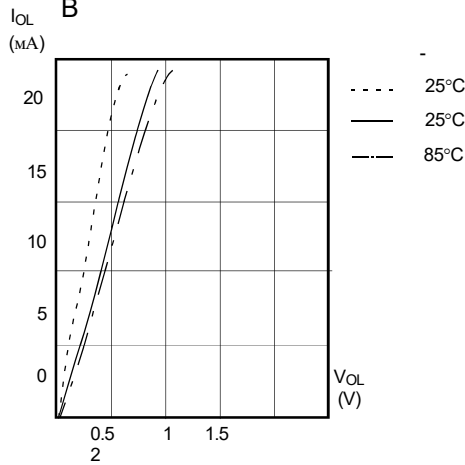
**указанный диапазон.**

Данные, представленные в этом разделе, представляют собой статистическую сводку данных, собранных по единицам из различных партий за определенный период времени. «Typ- ical» представляет среднее распределение, тогда как «max» или «min» представляет (среднее + 3σ) и (среднее - 3σ) соответственно, где σ - стандартное отклонение.

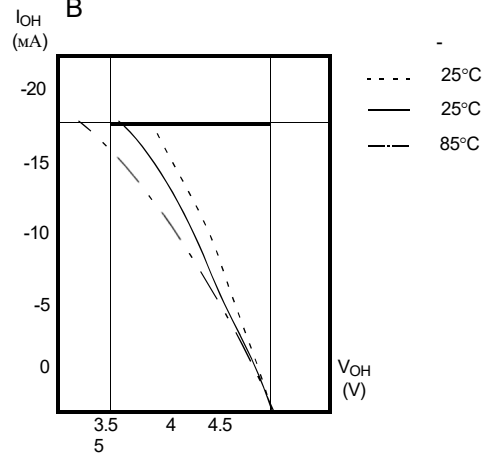




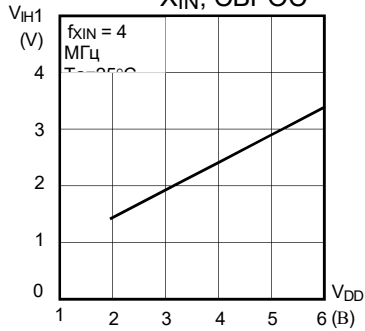
$I_{OL}-V_{OL}, V_{DD} = 5$



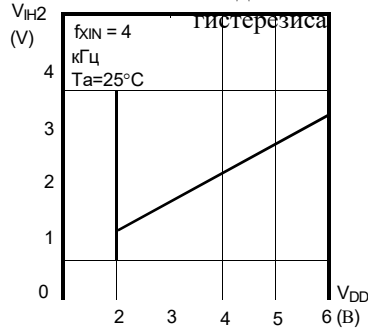
$I_{OH}-V_{OH}, V_{DD} = 5$



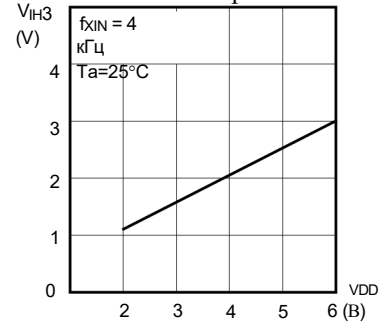
$V_{DD}-V_{IH1}$   
XIN, СБРОС



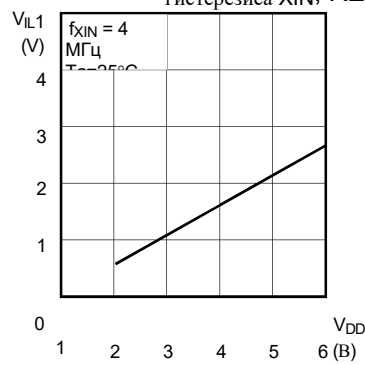
$V_{DD}-V_{IH2}$   
Вход гистерезиса



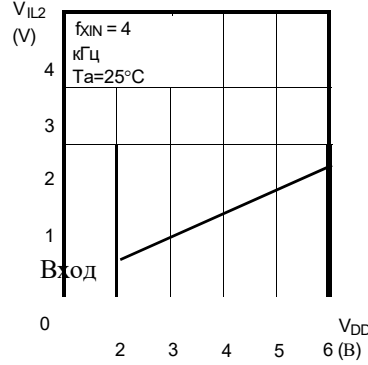
$V_{DD}-V_{IH3}$   
Нормальный вход



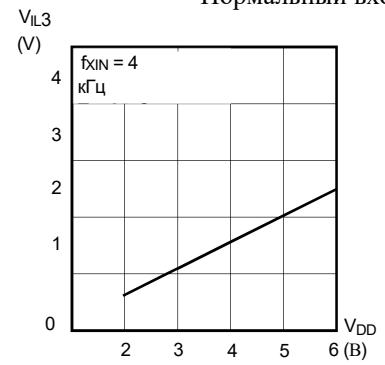
$V_{DD}-V_{IL1}$   
гистерезиса XIN, RESET

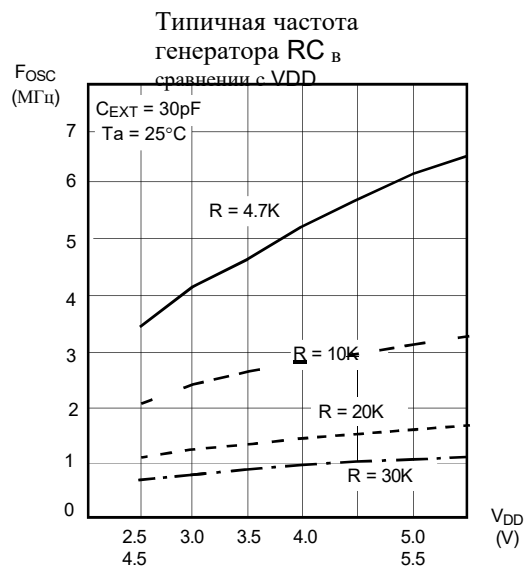
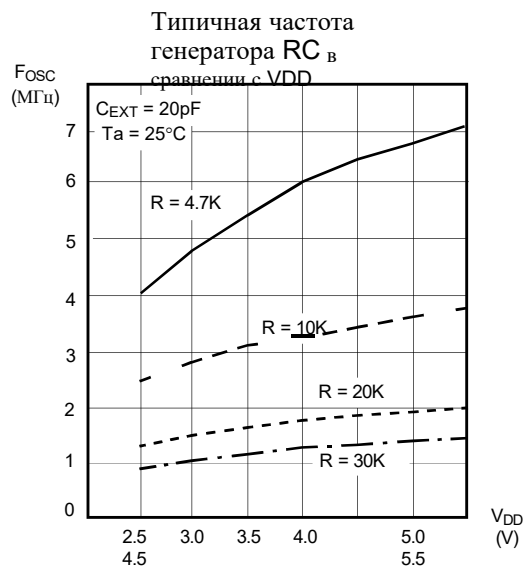
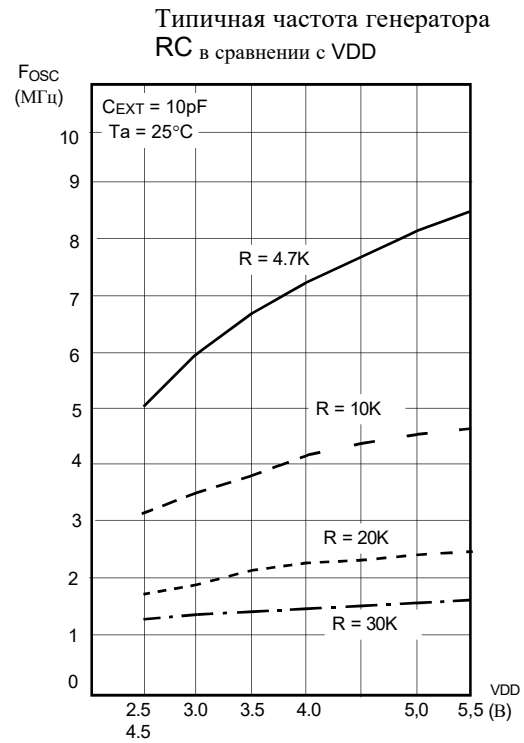
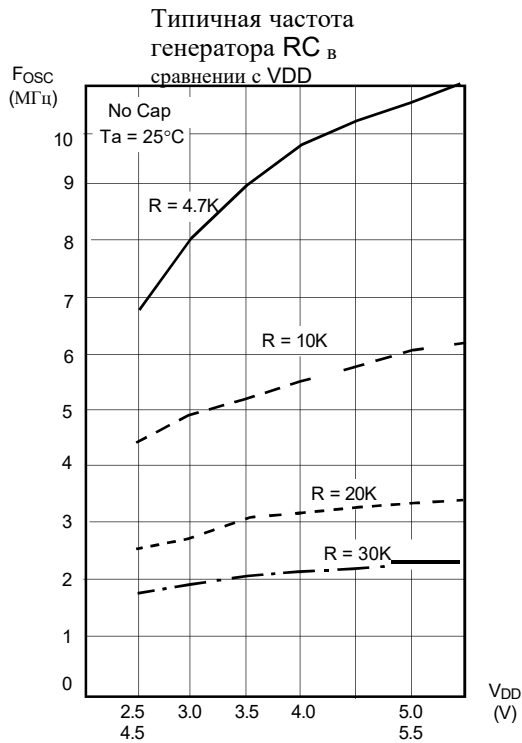


$V_{DD}-V_{IL2}$   
Вход



$V_{DD}-V_{IL3}$   
Нормальный вход

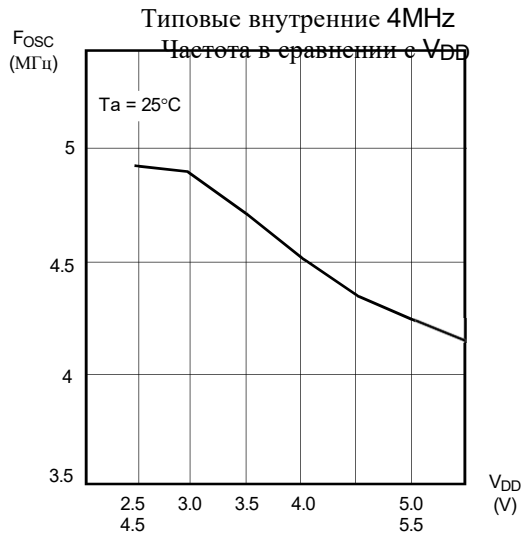




**Примечание:** Внешние частоты колебаний RC, показанные выше, предназначены только для руководства проектом и не испытываются и не гарантируются. Пользователь должен учитывать, что внешние частоты RC-колебаний, генерируемые одной и той же конструкцией схемы, могут быть неодинаковыми. Потому что есть различия в сопротивлении и емкости из-за допуска внешних R и C компонентов. Паразитная разность емкостей, обусловленная различной длиной и компоновкой проводки, может изменять частоту внешних RC-колебаний.

**Примечание:** Могут существовать различия между типами пакетов (PDIP, SOP, TSSOP). Пользователь должен изменить значение компонентов R и C, чтобы получить правильную частоту в MC80F0704/0708/0804/0808 или одном типе упаковки для другого типа упаковки.

по образцу, напряжению и температуре. Внутреннее колебание может быть использовано только при использовании без учета времени.



**Примечание:** Внутренние частоты колебаний 4MHz, показанные выше, предназначены только для руководства конструкцией и не испытываются и не гарантируются. Пользователь должен учитывать, что внутреннее колебание MC80F0704/0708/0804/0808 или MC80F0808 (4) может иметь различную частоту с образцом.

## 8. ОРГАНИЗАЦИЯ ПАМЯТИ

В MC80F0704/0708/0804/0808 имеются отдельные адресные пространства для памяти программ и памяти данных. 4К байт памяти программы - огу можно только читать, но не записывать в.

### 8.1 Регистры

Это устройство имеет шесть регистров, которые являются счетчиком программ (PC), накопителем (A), двумя индексными регистрами (X, Y), указателем стека (SP) и словом состояния программы (PSW). Счетчик программ состоит из 16-разрядного регистра.

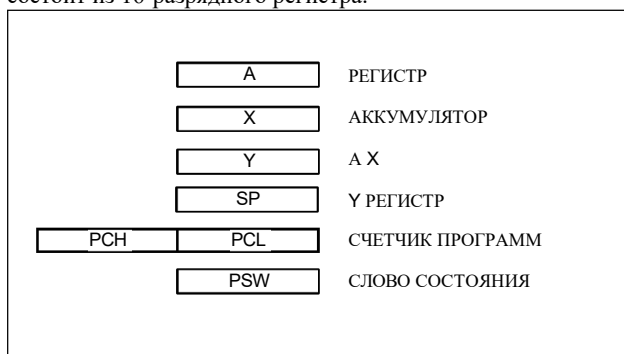


Рис. 8-1 Конфигурация регистров

**Аккумулятор:** Аккумулятор является 8-разрядным регистром общего назначения, используемым для работы с данными, такими как передача, временное сохранение, условное суждение и т.д.

**Накопитель** может использоваться как 16-битный регистр с Y-регистром, как показано ниже.



Рис. 8-2 Конфигурация 16-битного регистра YA

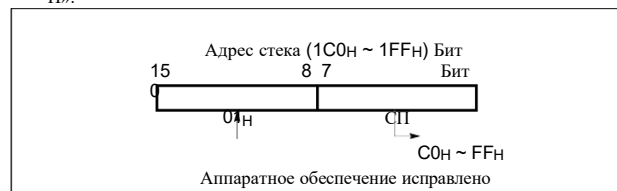
**X, Y Регистры:** В режиме адресации, который использует эти регистры индекса, содержимое регистра добавляется к указанному адресу, который становится фактическим адресом. Эти режимы чрезвычайно эффективны для ссылок на таблицы подпрограмм и таблицы памяти. Индексные регистры также имеют функции приращения, уменьшения, сравнения и передачи данных, и их можно использовать в качестве простых накопителей.

**Указатель стека:** Указатель стека - 8-битный регистр, используемый для ос-сигнпесе прерывы и вызывающие подпрограммы. Указатель стека определяет расположение стека, к которому требуется получить доступ (сохранить или восстановить).

Память данных может быть считана и записана до 256 байт, включая область стека.

Как правило, SP автоматически обновляется при выполнении вызова подпрограммы или принятии прерывания. Однако, если он используется вне области стека, разрешенной конфигурацией распределения памяти данных, обработанные пользователем данные могут быть потеряны.

Стек может быть расположен в любом месте в пределах 1C0H до 1FFH внутренней памяти данных. Процессор СХД не инициализируется аппаратными средствами, что требует записи начального значения (местоположения, с которого начинается использование стека) с помощью процедуры инициализации. Обычно используется начальное значение «FFH».



**Примечание.** Указатель стека должен быть инициализирован программным обеспечением, так как его значение не определено после сброса.

**Пример:** Инициализация процессора СХД

```
LDX # 0FFH
TXSP ; SP ← FFH
```

**Прилавок программы:** Прилавок Программы - 16 битов шириной, который состоит из двух 8-битных регистров, PCN и PCL. Этот счетчик указывает адрес следующей выполняемой команды. В состоянии сброса счетчик программы имеет адрес подпрограммы сброса (PCN:0FFH, PCL:0FEH).

**Program Status Word:** довод «против» Program Status Word (PSW) - оловянные амальгамы несколько битов, которые отражают текущее состояние центрального процессора. Описание PSW приведено в Рис. 8-3. Он содержит флаг «Negative», флаг «Overflow», флаг «Break», флаг «Half Carry» (для операции BCD), флаг «Interrupt enable», флаг «Zero» и флаг «Carry».

[Флаг переноса C]

Этот флаг сохраняет любую передачу или заимствование из ALU CPU после арифметической операции, а также изменяется командой Shift Instruction или Rotate Instruction.

[Флаг нуля Z]

Этот флаг устанавливается, когда результат арифметической операции или передачи данных равен «0» и очищается любым другим результатом.

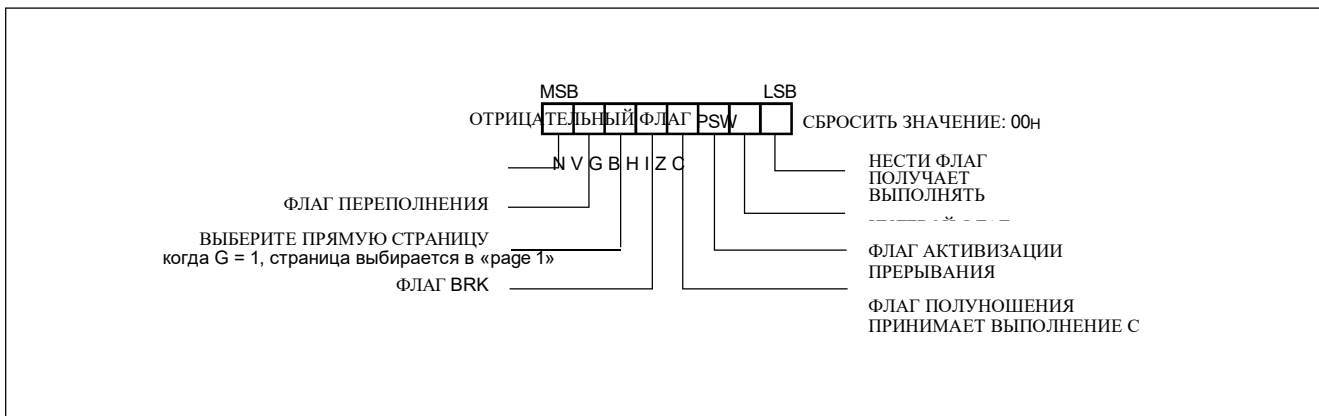


Рис. 8-3 Регистр PSW (Program Status Word)

[Флаг отключения прерывания I]

Этот флаг включает/выключает все прерывания, кроме прерывания, вызванного командой Reset или программной командой BRK. Все прерывания отключаются при сбросе на «0». Этот флаг сразу становится «0», когда обслуживается прерывание. Он устанавливается командой EI и очищается командой DI.

[Флаг H переноса половины]

После выполнения операции этот параметр устанавливается при наличии переноса из бита 3 ALU или при отсутствии заимствования из бита 4 ALU. Этот бит не может быть установлен или сброшен, кроме команды CLR V с флагом переполнения (V).

[Флаг разрыва B]

Этот флаг устанавливается программной командой BRK для отличия BRK от команды TCALL с тем же векторным адресом.

[Флаг прямой страницы G]

регистр

режим

Этот флаг назначает страницу ОЗУ для режима прямой адресации. В ди- адресации, область адресации находится от нуля 00H страницы до 0FFH, когда этот флаг равен «0». Если установлено значение «1», область адресации назначается 100H 1FFH. Устанавливается командой SETG и сбрасывается CLR G.

[Флаг переполнения V]

Этот флаг устанавливается в «1», когда происходит переполнение в результате арифметической операции с использованием знаков. Переполнение происходит, когда результат сложения или вычитания превышает + 127 (7FH) или - 128 (80H). Команда CLR V очищает флаг переполнения. Команда набора отсутствует. Когда команда BIT выполняется, бит 6 памяти копируется в этот флаг.

[Отрицательный флаг N]

Этот флаг устанавливается для соответствия знаковому биту (бит 7) состояния результата операции передачи данных или арифметической операции. Когда команда BIT выполняется, бит 7 памяти копируется в этот флаг.

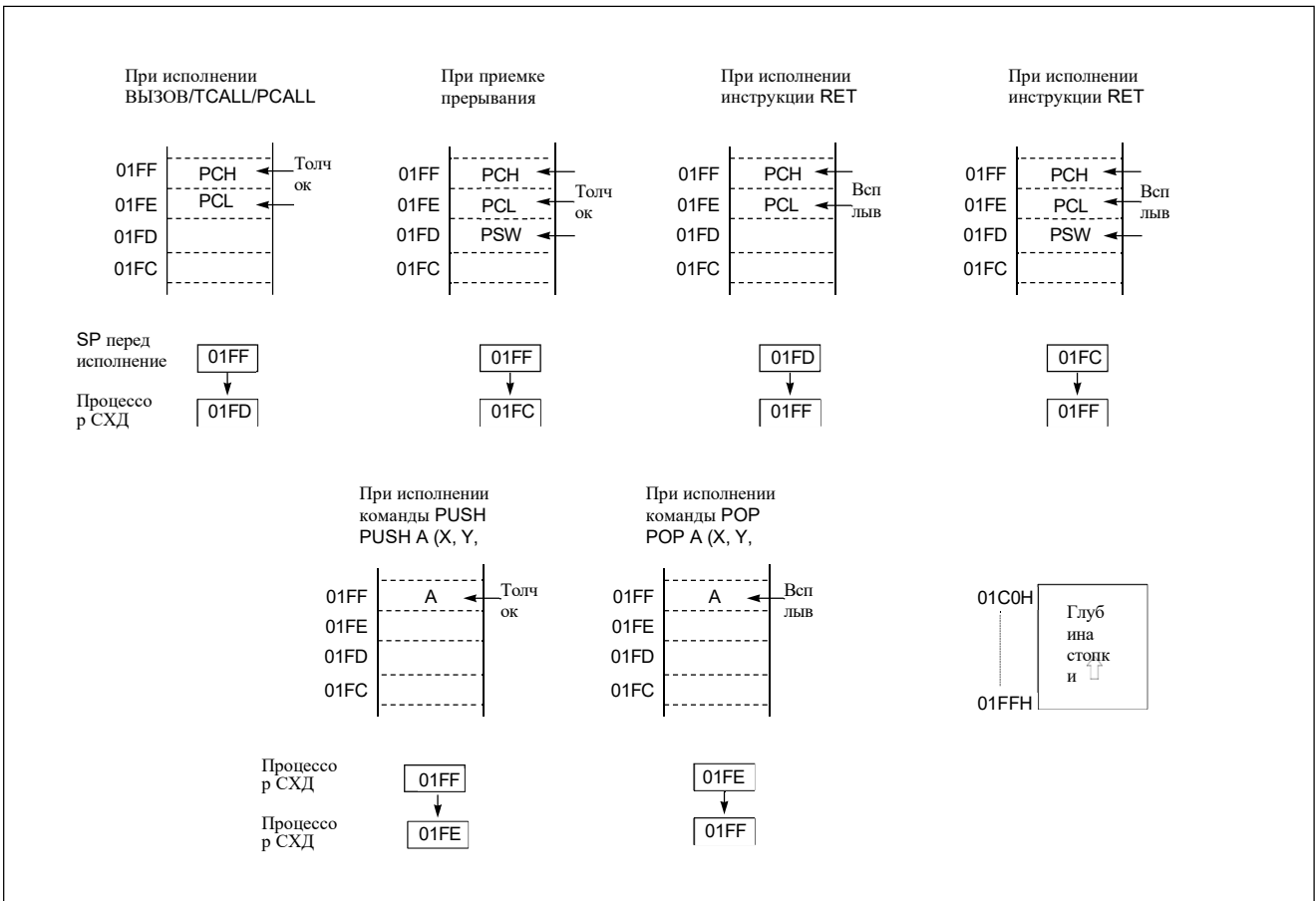


Рис. 8-4 Работа стека



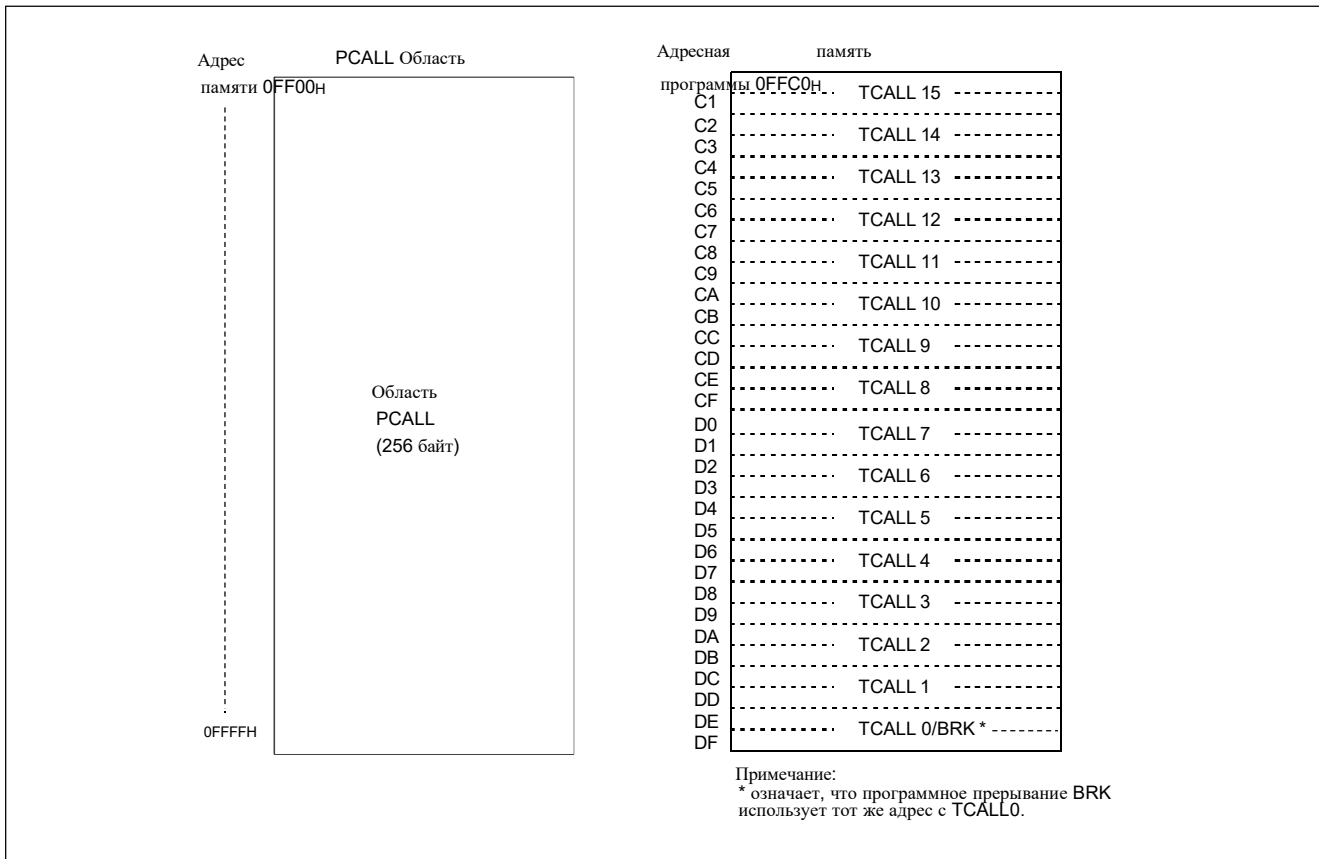
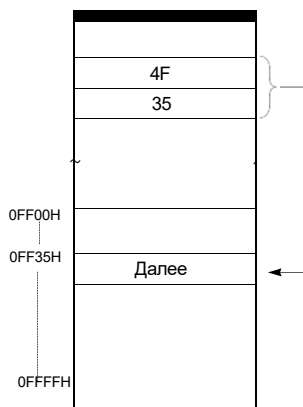


Рис. 8-7 Область памяти PCALL и TCALL

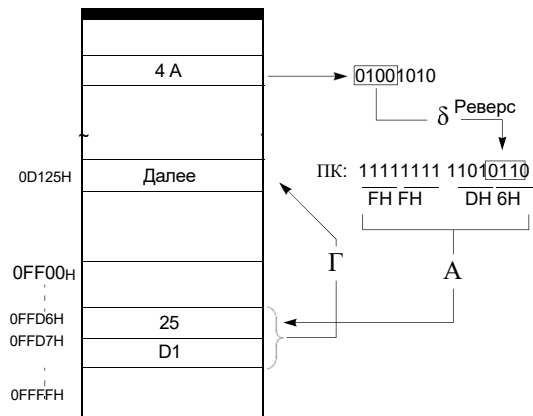
PCALL → rel

4F35 PCALL 35H



TCALL → n

4A TCALL 4



Пример: Пример использования программного обеспечения Vector address



### 8.3 Память данных

На рис. 8-8 показано доступное внутреннее пространство памяти данных. Память данных разделена на три группы: ОЗУ пользователя, регистры управления и память стека.

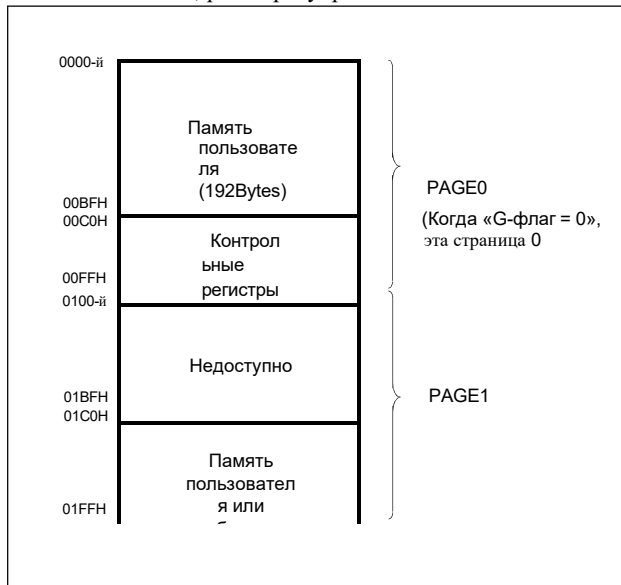


Рис. 8-8 Карта памяти данных

#### Память пользователя

В MC80F0704/0708/0804/0808 имеется 256 × 8 бит для памяти пользователя (ОЗУ). RPR выбирает страницы ОЗУ (смРисунок 8-9.).

**Примечание.** После установки RPR (RAM Page Select Register) обязательно выполните команду SETG. При выполнении команды CLRG выбирать PAGE0 независимо от RPR.

#### Контрольные регистры

Управляющие регистры используются CPU и функций периферийных устройств



Рис. 8-9 RPR (регистр выбора страницы ОЗУ)

блоки для управления требуемой работой устройства. Поэтому эти регистры содержат биты управления и состояния для системы прерываний, таймера/счетчиков, аналого-цифровых преобразователей и портов ввода/вывода. Управляющие регистры находятся в диапазоне адресов от 0C0H до 0FFH.

Следует отметить, что незанятые адреса не могут быть реализованы на кристалле. Доступ для чтения к этим адресам обычно возвращает данные ran-dom, а доступ для записи имеет неопределенный эффект.

Более подробная информация о каждом регистре поясняется в каждом периферийном разделе.

**Примечание.** Доступ к регистрам, предназначенным только для записи, с помощью команды bit manipulation невозможен. Не используйте инструкцию по чтению-изменению-записи. Используйте инструкцию управления байтами, например «LDM».

Пример; Написать в CCCTLR

```
LDM    CCCTLR, # 0AH; Коэффициент
        разделения (+32)
```

#### Область стека

Стек предоставляет область, где сохраняется обратный адрес, прежде чем выполняется переход во время процедуры обработки при выполнении команды вызова подпрограммы или принятии прерывания.

При возврате из подпрограммы обработки выполнение команды [RET] возврата подпрограммы восстанавливает содержимое счетчика pro-грамм из стека; выполнение команды возврата прерывания [RETI] восстанавливает содержимое счетчика программ и флагов.

Местоположения сохранения/восстановления в стеке определяются указателем стека (SP). Процессор СХД автоматически уменьшается после сохранения и увеличивается перед восстановлением. Это означает, что значение SP указывает номер местоположения стека для следующего сохранения. См. рис. 8-4 на стр. 26.

Адрес	Имя регистра	Символ	R/W	Начальное значение								Режим адресации	
				7	6	5	4	3	2	1	0		
00C0	Регистр данных порта R0	R0	R/W	0	0	0	0	0	0	0	0	0	байт, бит <sup>1</sup>
00C1	Регистр направления ввода-вывода порта R0	R0IO	W	0	0	0	0	0	0	0	0	0	byte2
00C2	Регистр данных порта R1	R1	R/W	0	0	0	0	0	0	0	0	0	байт, бит
00C3	Регистр направления ввода/вывода порта R1	R1IO	W	0	0	0	0	0	0	0	0	0	байт
00C4	Регистр данных порта R2	R2	R/W	0	0	0	0	0	0	0	0	0	байт, бит
00C5	Регистр направления ввода/вывода порта R2	R2IO	W	0	0	0	0	0	0	0	0	0	байт
00C6	Регистр данных порта R3	R3	R/W	-	-	0	0	0	0	0	0	0	байт, бит
00C7	Регистр направления ввода-вывода порта R3	R3IO	W	-	-	0	0	0	0	0	0	0	байт
00C8	Порт 0 - Регистр выбора открытого стока	R0OD	W	0	0	0	0	0	0	0	0	0	байт
00C9	Порт 1 Регистр выбора открытого стока	R1OD	W	0	0	0	0	0	0	0	0	0	байт
00CA	Порт 2, регистр выбора открытого стока	R2OD	W	0	0	0	0	0	0	0	0	0	байт
00CB	Порт 3, регистр выбора открытого стока	R3OD	W	-	-	0	0	0	0	0	0	0	байт
00D0	Регистр управления режимом таймера 0	TM0	R/W	-	-	0	0	0	0	0	0	0	байт, бит
00D1	Регистр таймера 0	T0	R	0	0	0	0	0	0	0	0	0	байт
	Регистр данных таймера 0	TDR0	W	1	1	1	1	1	1	1	1	1	
	Регистр данных захвата таймера 0	CDR0	R	0	0	0	0	0	0	0	0	0	
00D2	Регистр управления режимом таймера 1	TM1	R/W	0	0	0	0	0	0	0	0	0	байт, бит
00D3	Регистр данных таймера 1	TDR1	W	1	1	1	1	1	1	1	1	1	байт
	Таймер 1 Регистр периодов PWM	T1PPR	W	1	1	1	1	1	1	1	1	1	байт
00D4	Регистр таймера 1	T1	R	0	0	0	0	0	0	0	0	0	байт
	Регистр данных захвата таймера 1	CDR1	R	0	0	0	0	0	0	0	0	0	
	Регистр работы таймера 1 PWM	T1PDR	R/W	0	0	0	0	0	0	0	0	0	
00D5	Таймер 1 Регистр высокого уровня PWM	T1PWHR	W	-	-	-	-	0	0	0	0	0	бит
00D6	Регистр управления режимом таймера 2	TM2	R/W	-	-	0	0	0	0	0	0	0	байт, бит
00D7	Регистр таймера 2	T2	R	0	0	0	0	0	0	0	0	0	байт
	Регистр данных таймера 2	TDR2	W	1	1	1	1	1	1	1	1	1	
	Регистр данных захвата таймера 2	CDR2	R	0	0	0	0	0	0	0	0	0	
00D8	Регистр управления режимом таймера 3	TM3	R/W	0	0	0	0	0	0	0	0	0	байт, бит
00D9	Регистр данных таймера 3	TDR3	W	1	1	1	1	1	1	1	1	1	байт
	Таймер 3 Регистр периодов PWM	T3PPR	W	1	1	1	1	1	1	1	1	1	
00DA	Регистр таймера 3	T3	R	0	0	0	0	0	0	0	0	0	байт
	Регистр работы таймера 3 PWM	T3PDR	R/W	0	0	0	0	0	0	0	0	0	
	Регистр данных захвата таймера 3	CDR3	R	0	0	0	0	0	0	0	0	0	

Таблица 8-1 Контрольные регистры

Адрес	Имя регистра	Символ	R/W	Начальное значение								Режим адресации
				7	6	5	4	3	2	1	0	
00DB	Таймер 3 Регистр высокого уровня PWM	T3PWHR	W	-	-	-	-	0	0	0	0	байт
00E0	Регистр драйверов зуммера	BUZR	W	1	1	1	1	1	1	1	1	байт
00E1	Регистр выбора страниц RAM	RPR	R/W	-	-	-	-	-	0	0	0	байт, бит
00E2	Регистр управления режимом SIO	SIOM	R/W	0	0	0	0	0	0	0	1	байт, бит
00E3	Регистр сдвига данных SIO	SIOR	R/W	Неопределенный								байт, бит
00EA	Высокий уровень разрешения прерываний в регистре	IENH	R/W	0	0	0	0	0	0	0	0	байт, бит
00EB	Низкий уровень регистра включения прерываний	IENL	R/W	0	0	0	0	0	0	0	0	байт, бит
00EC	Высокий уровень регистра запросов на прерывание	IRQH	R/W	0	0	0	0	0	0	0	0	байт, бит
00ED	Низкий уровень регистра запросов на прерывание	IRQL	R/W	0	0	0	0	0	0	0	0	байт, бит
00EE	Регистр выбора границ прерываний	IEDS	R/W	0	0	0	0	0	0	0	0	байт, бит
00EF	Регистр управления режимом АЦП	ADCM	R/W	0	0	0	0	0	0	0	1	байт, бит
00F0	Высокий регистр результата аналого-цифрового преобразователя	ADCRH	R (W)	0	1	0	Неопределенный					байт
00F1	Низкочастотный регистр результата АЦП	ADCRL	R	Неопределенный								байт
00F2	Регистр таймера основного интервала	BITR	R	Неопределенный								байт
	Регистр управления синхросигналами	CKCTLR	W	0	-	0	1	0	1	1	1	
00F4	Регистр таймера сторожевой собаки	WDTR	W	0	1	1	1	1	1	1	1	байт
	Регистр данных таймера сторожевой собаки	WDTDR	R	Неопределенный								
00F5	Регистр управления остановкой и режимом ожидания	SSCR	W	0	0	0	0	0	0	0	0	байт
00F7	Регистр управления PFD	PFDR	R/W	-	-	-	-	-	0	0	0	байт, бит
00F8	Регистр выбора порта 0	PSR0	W	0	0	0	0	0	0	0	0	байт
00F9	Регистр выбора порта 1	PSR1	W	-	-	-	-	0	0	0	0	байт
00FC	Регистр выбора подтягивания 0	PU0	W	0	0	0	0	0	0	0	0	байт
00FD	Регистр выбора подтягивания 1	PU1	W	0	0	0	0	0	0	0	0	байт
00FD	регистр выбора подтягивания 2	PU2	W	0	0	0	0	0	0	0	0	байт
00FF	Регистр выбора подтягивания 3	PU3	W	-	-	0	0	0	0	0	0	байт

**Таблица 8-1 Контрольные регистры**

- «Байт, бит» означает, что регистры управляются инструкцией манипулирования битами и байтами. Внимание! Регистр R/W, за исключением T1PDR и T3PDR, может обрабатываться байтами и битами.
- «Байтовые» регистры управляются только инструкцией манипулирования байтами. Не используйте инструкции по манипулированию битами, такие как SET1, CLR1 и т.д. Если в этих регистрах используется команда битовой манипуляции, содержание других семи битов может изменяться до нежелательного значения.

\* Метка «-» означает, что это местоположение бита зарезервировано.

Адрес	Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0
0C0H	R0	Регистр данных порта R0							
0C1H	R0IO	Регистр направления порта R0							
0C2H	R1	Регистр данных порта R1							
0C3H	R1IO	Регистр направления порта R1							
0C4H	R2	Регистр данных порта R2							
0C5H	R2IO	R2Port Регистр данных							
0C6H	R3	Регистр данных порта R3							
0C7H	R3IO	Регистр направления порта R3							
0C8H	R0OD	R0 Регистр выбора открытого стока							
0C9H	R1OD	Регистр выбора открытого дренажа R1							
0CAH	R2OD	R2 Регистр выбора открытого стока							
0CBH	R3OD	Регистр выбора открытого дренажа R3							
0D0H	TM0	-	-	CAP0	T0CK2	T0CK1	T0CK0	T0CN	T0ST
0D1H	T0/TDR0/ CDR0	Регистр Timer0 / Регистр Данных Timer0 / Регистр Данных о Захвате Timer0							
0D2H	TM1	T1_POL	T1_16BIT	PWM1E	CAP1	T1CK1	T1CK0	T1CN	T1ST
0D3H	TDR1/T1P PR	Timer1 Регистр данных/ Timer1 Регистр периодов PWM							
0D4H	T1/CDR1	Регистр Timer1 / Регистр Данных о Захвате Timer1							
0D5H	PWM1HR	-	-	-	-	Timer1 Высокий регистр PWM			
0D6H	TM2	-	-	CAP2	T2CK2	T2CK1	T2CK0	T2CN	T2ST
0D7H	T2/TDR2/ CDR2	Регистр Timer2 / Регистр Данных Timer2 / Регистр Данных о Захвате Timer2							
0D8H	TM3	T3_POL	T3_16BIT	PWM3E	CAP3	T3CK1	T3CK0	T3CN	T3ST
0D9H	TDR3/T3P PR	Timer3 Регистр данных/ Timer3 Регистр периодов PWM							
0DAH	T3/CDR3/ T3PDR	Регистр Timer3 / Регистр Данных о Захвате Timer3 / Регистр Обязанности Timer3 PWM							
0DBH	PWM3HR	-	-	-	-	Timer3 Высокий регистр PWM			
0E0H	BUZR	BUCK1	BUCK0	BUR5	BUR4	BUR3	BUR2	BUR1	BUR0
0E1H	RPR	-	-	-	-	-	RPR2	RPR1	RPR0
0E2H	SIOM	ПОЛИТИК	IOSW	SM1	SM0	SCK1	SCK0	SIOST	SIOF
0E3H	SIOR	Регистр сдвига данных SIO							
0EAH	IENH	INT0E	INT1E	INT2E	INT3E	-	-	SIOE	T0E
0EBH	IENL	T1E	T2E	T3E	-	ADCE	WDTE	WTE	УКУС
0ECH	IRQH	INT0IF	INT1IF	INT2IF	INT3IF	-	-	SIOIF	T0IF
0EDH	IRQL	T1IF	T2IF	T3IF	T4IF	ADCIF	WDTIF	WTIF	BITIF

Таблица 8-2 Описание функции регистра управления

Адрес	Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0
0EEH	IEDS	IED3H	IED3L	IED2H	IED2L	IED1H	IED1L	IED0H	IED0L
0EFH	ADCM	ADEN	ADCK	ADS3	ADS2	ADS1	ADS0	ADST	ADSF
0F0H	ADCRH	PSSEL1	PSSEL0	ADC8	-	-	-	Высокий результат ADC	
0F1H	ADCRL	Низкий уровень регистра результатов АЦП							
0F2H	BITR1	Регистр данных таймера основного интервала							
	CCCTLR1	ADRST	-	RCWDT	WDTON	BTCL	BTS2	BTS1	BTS0
0F4H	WDTR	WDTCL	7-битный регистр сторожевого таймера						
	WDTDR	Регистр данных контрольного таймера (регистр счетчиков)							
0F5H	SSCR	Регистр управления остановом и режимом сна							
0F7H	PFDR	-	-	-	-	-	PFDEN	PFDM	PFDS
0F8H	PSR0	PWM3O	PWM1O	EC1E	EC0E	INT3E	INT2E	INT1E	INT0E
0F9H	PSR1	-	-	-	-	XTEN	BUZO	T2O	T0O
0FCH	PU0	R0 Регистр выбора при развертывании							
0FDH	PU1	R1 Регистр выбора подтягивания							
0FEH	PU2	R2 Регистр выбора подтягивания							
0FFH	PU3	R3 Регистр выбора вытягивания							

**Таблица 8-2 Описание функции регистра управления**

1. Регистр *BITR* и *CCCTLR* расположены по одному адресу. Адрес *ECH* считывается как *BITR*, записывается в *CCCTLR*.

Внимание) Регистры темной-затененной области не могут быть доступны с помощью команды манипулирования битами, такой как «*SET1*», «*CLR1*», но должны быть доступны с помощью команды операции регистра, такой как «*LDM dp, # imm*».

### 8.4 Режим адресации

MCU серии MC8 использует шесть режимов адресации;

- Адресация регистров
- Немедленная адресация
- Прямая адресация страницы
- Абсолютная адресация
- Индексированная адресация
- Адресация без учета регистров

#### Регистрация адресации

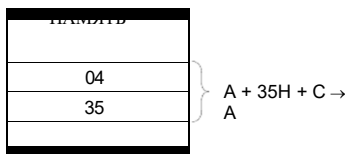
Адресация регистров обеспечивает доступ к A, X, Y, C и PSW.

#### Немедленная адресация → # imm

В этом режиме осуществляется немедленный доступ ко второму байту (операнду) в качестве данных.

Пример:

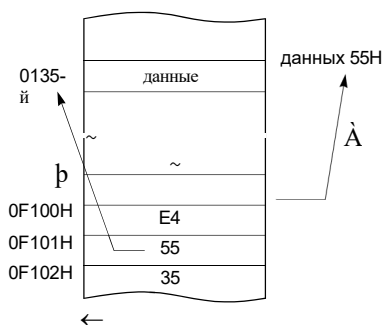
```
0435   ADC   #
35H
```



Когда G-флаг равен 1, то адрес ОЗУ определяется 16-битным адресом, который состоит из 8-битного регистра поискового вызова ОЗУ (RPR) и 8-битных непосредственных данных.

Пример: G = 1

```
E45535 LDM 35H, # 55H
```

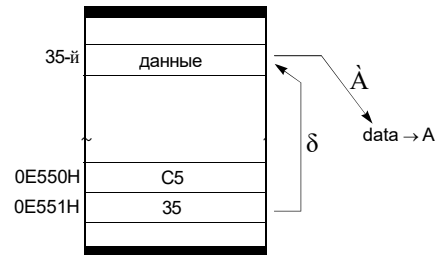


#### Прямая адресация страницы → dp

В этом режиме адрес указывается на прямой странице.

Пример; G = 0

```
C535   LDA   35H; A ←RAM[35H]
```



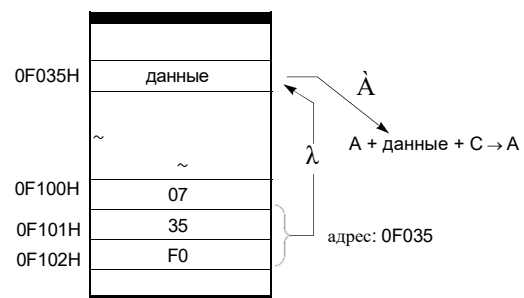
#### Абсолютная адресация →! abs

Абсолютная адресация устанавливает соответствующие Data данные памяти, т.е. второй байт (операнд I) команды становится адресом нижнего уровня, а третий байт (операнд II) становится адресом верхнего уровня. С помощью команды 3 байта можно получить доступ ко всей области памяти.

ADC, И, CMP, CMPX, CMPY, EOR, LDA, LDX, LDY, OREGON, SBC, СТАНЦИЯ, STX, СВИНАРНИК

Пример;

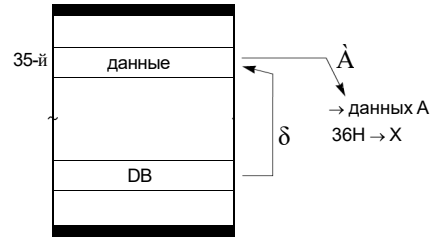
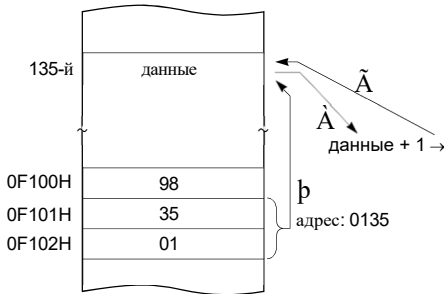
```
0735F0 ADC   ! 0F035H ; A ←ROM[0F035H]
```



Работа в памяти данных (RAM) ASL, BIT, DEC, INC, LSR, ROL, ROR

Пример; Адресация обращается к 0135H адресов независимо от G-флага.

983501 INC ! 0135H ; A ← ROM[135H]



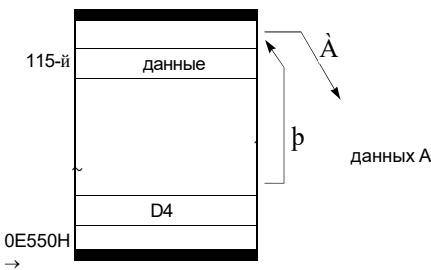
**Индексированная адресация**

**X индексированная прямая страница (без смещения) → {X}**

В этом режиме адрес задается X-регистром. ADC, AND, CMP, EOR, LDA, OR, SBC, STA, XMA

Пример; X = 15H, G = 1

D4 LDA {X} ;  
ACC ← RAM[X].



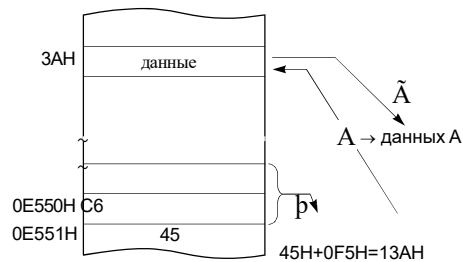
**X индексированная прямая страница (смещение 8 бит) → dp + X**

Это значение адреса является вторым байтом (Operand) команды плюс данные X-регистра. И он назначает память на странице Direct.

ADC, AND, CMP, EOR, LDA, LDY, OR, SBC, STA, STY, XMA, ASL, DEC, INC, LSR, ROL, ROR

Пример; G = 0, X = 0F5H

C645 LDA 45H + X



**X индексированная прямая страница, авто increment → {X} +**

В этом режиме адрес задается в прямой странице X-регистром, и содержание X увеличивается на 1.

LDA, STA

Пример; G = 0, X = 35H

БД LDA {X} +

**Y-индексированная прямая страница (смещение 8 бит) → dp + Y**

Это значение адреса является вторым байтом (Operand) команды плюс данные Y-регистра, который назначает память на странице Direct.

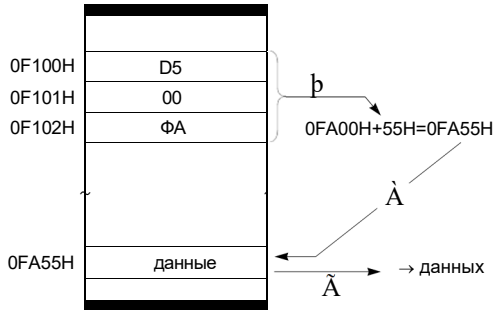
Это то же самое, что и выше (2). Используйте регистр Y вместо X.

**Y индексированный абсолютный → ! abs + Y**

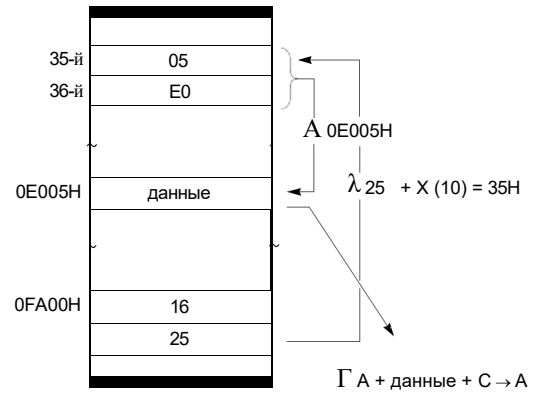
Устанавливает значение 16-битного абсолютного адреса плюс данные Y-регистра как Memory. Этот режим адресации может задавать память в целом.

Пример; Y = 55H

D500FA LDA ! 0FA00H + Y



1625 ADC [25H + X]



**Косвенная адресация**

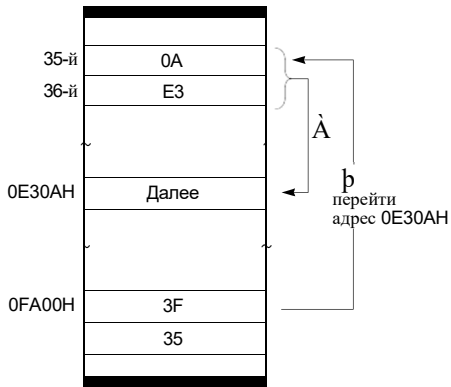
**Прямая непрямая → страницы [dp]**

Назначает адрес данных, используемый для выполнения команды, которая устанавливает данные памяти (или парную память) операндом.

Также индекс может использоваться с индексным регистром X, Y. JMP, ВЫЗОВ

Пример; G = 0

3F35 JMP [35H]



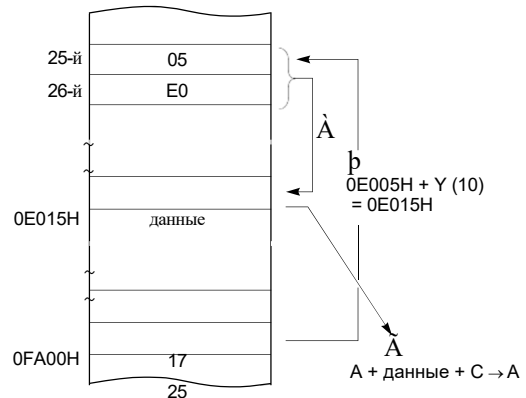
**Y индексированный косвенный → [dp] + Y**

Обрабатывает данные памяти как данные, назначенные данными [dp + 1] [dp] 16-битной парной памяти, спаренной операндом на странице Direct, плюс данные регистра Y.

ADC, AND, CMP, EOR, LDA, OR, SBC, STA

Пример; G = 0, Y = 10H

1725 ADC [25H] + Y



**X индексированный косвенный → [dp + X]**

Обрабатывает данные памяти как данные, назначенные 16-битной парной памятью, которая определяется парными данными [dp + X + 1] [dp + X] Операнд плюс данные X-регистра на странице Direct.

ADC, AND, CMP, EOR, LDA, OR, SBC, STA

Пример; G = 0, X = 10H

**Абсолютная косвенная → [! abs]**

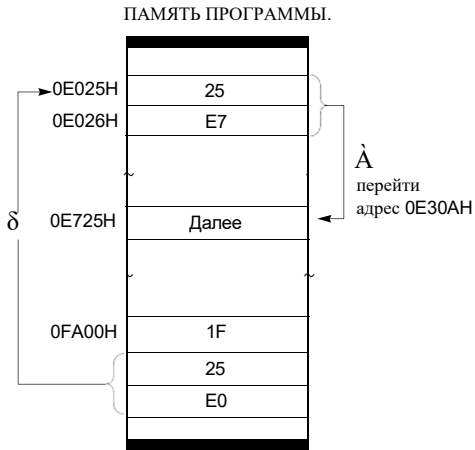
Программа прыгает по адресу, указанному 16-битным абсолютным значением.

JMP

Пример; G = 0



1F25E0 JMP [! 0C025H]



### 9. ПОРТЫ ВВОДА-ВЫВОДА

В MC80F0704/0708/0804/0808 имеется три порта (R0, R1 и R3). Эти контакты портов могут быть мультиплексированы с альтернативной функцией для периферийных функций устройства. Весь порт может возбуждать максимальную 20mA высокого тока в выходном низком состоянии, поэтому он может непосредственно возбуждать светодиодное устройство.

Все контакты имеют регистры направления данных, которые могут определять эти порты как выходные или входные. «1» в регистре направления порта конфигурирует соответствующий вывод порта в качестве выходного. И наоборот, запишите «0» в соответствующий бит, чтобы указать его как входной вывод. Например, чтобы использовать четный бит R0 в качестве выходных портов и нечетные числовые биты в качестве входных портов, запишите «55H» в адрес 0C1H (регистр направления порта R0) во время начальной установки, как показано на рис. 9-1.

Все регистры направления порта в MC80F0704/0708/0804/0808 0 записаны в них функцией сброса. С другой стороны, его начальный статус является входным.

#### 9.1 Регистр R0 и R0IO

R0 является 8-битным двунаправленным портом ввода-вывода CMOS (адрес 0C0H). Каждый вывод ввода/вывода может независимо использоваться в качестве входа или выхода через регистр R0IO (адрес 0C1H). Когда в качестве входных портов используются R00 через R07 штырьковые контакты, может быть подключен встроенный резистор -

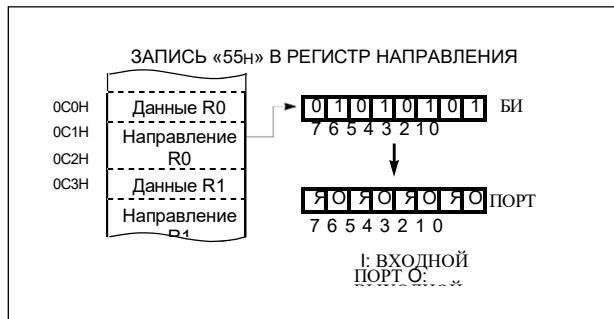


Рис. 9-1 Пример назначения портов ввода-вывода

соединен с ними в 1-разрядных блоках с регистром 0 выбора подтягивания (PU0). Каждый вывод ввода/вывода порта R0 может использоваться для открытия выходного порта стока путем установки соответствующего бита регистра выбора открытого стока 0 (R0OD).

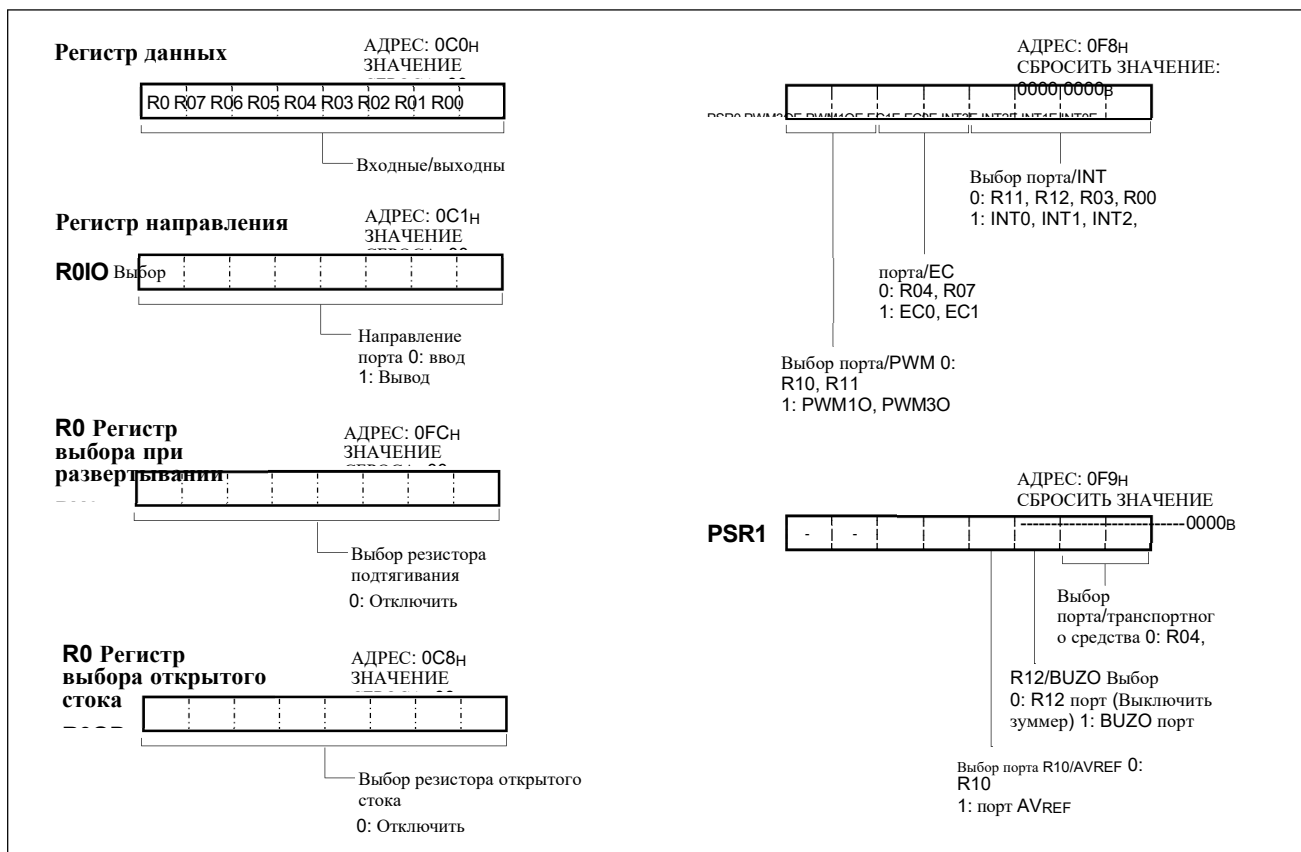


Рис. 9-2 Регистр портов R0

Кроме того, порт R0 мультиплексируется с различными альтернативными функциями. Регистр выбора порта PSR0 (адрес 0F8H) и PSR1 (адрес 0F9H) управляют выбором альтернативных функций, таких как

внешнее прерывание 3 (INT3), внешнее прерывание 2 (INT2), вход счетчика событий 0 (EC0), выход таймера 0 (T0O), выход таймера 2 (T2O) и вход счетчика событий 1 (EC1). Когда альтернативная функция -

tion выбирается путем записи «1» в соответствующий бит PSR0 или PSR1, контакт порта может использоваться в качестве соответствующих альтернативных функций независимо от регистра направления R0IO.

Входной канал АЦП 1 ~ 7 (AN1 ~ AN7) и вход данных SIO (SI), выход данных SIO (SOUT), вход/выход синхросигнала SIO (SCK) могут быть выбраны путем установки регистра ADCM (00EF<sub>H</sub>) и SIOM (00E2<sub>H</sub>) для активизации соответствующей периферийной операции и выбора режима работы.

Контакт порта	Альтернативная функция
R00-	INT3 (Внешнее прерывание 3)
R01	SCK (вход/выход синхросигнала SIO) AN1 (входной канал АЦП 1) SI (вход данных SIO)
R02	AN2 (входной канал АЦП 2) SOUT (вывод данных SIO)
R03	AN3 (входной канал АЦП 3)
R04	INT2 (Внешнее прерывание 2) AN4 (входной канал АЦП 4)
R05	EC0 (вход счетчика событий 0) AN5 (входной канал 5 АЦП) T00 (Выход таймера 0)
R06	AN6 (входной канал АЦП 6) T20 (Выход таймера 2)
R07	AN7 (входной канал АЦП 7) EC1 (Вход счетчика событий 1)

## 9.2 Регистр R1 и R1IO

R1 является 5-битным двунаправленным портом ввода-вывода CMOS (адрес 0C2<sub>H</sub>). Каждый вывод ввода/вывода может независимо использоваться в качестве входа или выхода через регистр R1IO (адрес 0C3<sub>H</sub>). Когда контакты R10-R17 используются в качестве входных портов, к ним может быть подключен встроенный резистор с 1-разрядными блоками с регистром 1 выбора подтягивания (PU1). Каждый вывод ввода/вывода порта R1 может использоваться для открытия выходного порта стока путем установки соответствующего бита регистра выбора открытого стока 1 (R1OD).

Кроме того, порт R1 мультиплексируется с различными альтернативными функциями. Регистр выбора порта PSR0 (адрес 0F8<sub>H</sub>) и PSR1 (адрес 0F9<sub>H</sub>) управляют выбором альтернативных функций, таких как вход аналогового опорного напряжения (AVREF), внешнее прерывание 0 (INT0), внешнее прерывание 1 (INT1), выход PWM 1 (PWM1O), выход PWM 3 (PWM3O) и выход зуммера (BUZO). Когда альтернативная функция выбирается путем записи «1» в соответствующий бит PSR0 или PSR1, вывод порта может использоваться в качестве соответствующих дополнительных признаков независимо от R1IO регистра направления.

Входной канал АЦП 0 (AN0) и канал 8 (AN8) могут быть выбраны путем установки регистра АЦП (00EF<sub>H</sub>) для включения АЦП и

выберите канал 0 и канал 8.

Контакт порта	Альтернативная функция
R10	AN0 (входной канал АЦП 0)
R11	AVREF (аналоговое опорное напряжение) PWM1O (выход ШИМ 1)
R12	INT0 (Внешнее прерывание 0) PWM3O (Выход PWM 3) INT1 (Внешнее прерывание 1) BUZO (Выход Buzzer)
R13	-
R14	-
R15	-
R16	-
R17	-
	AN8

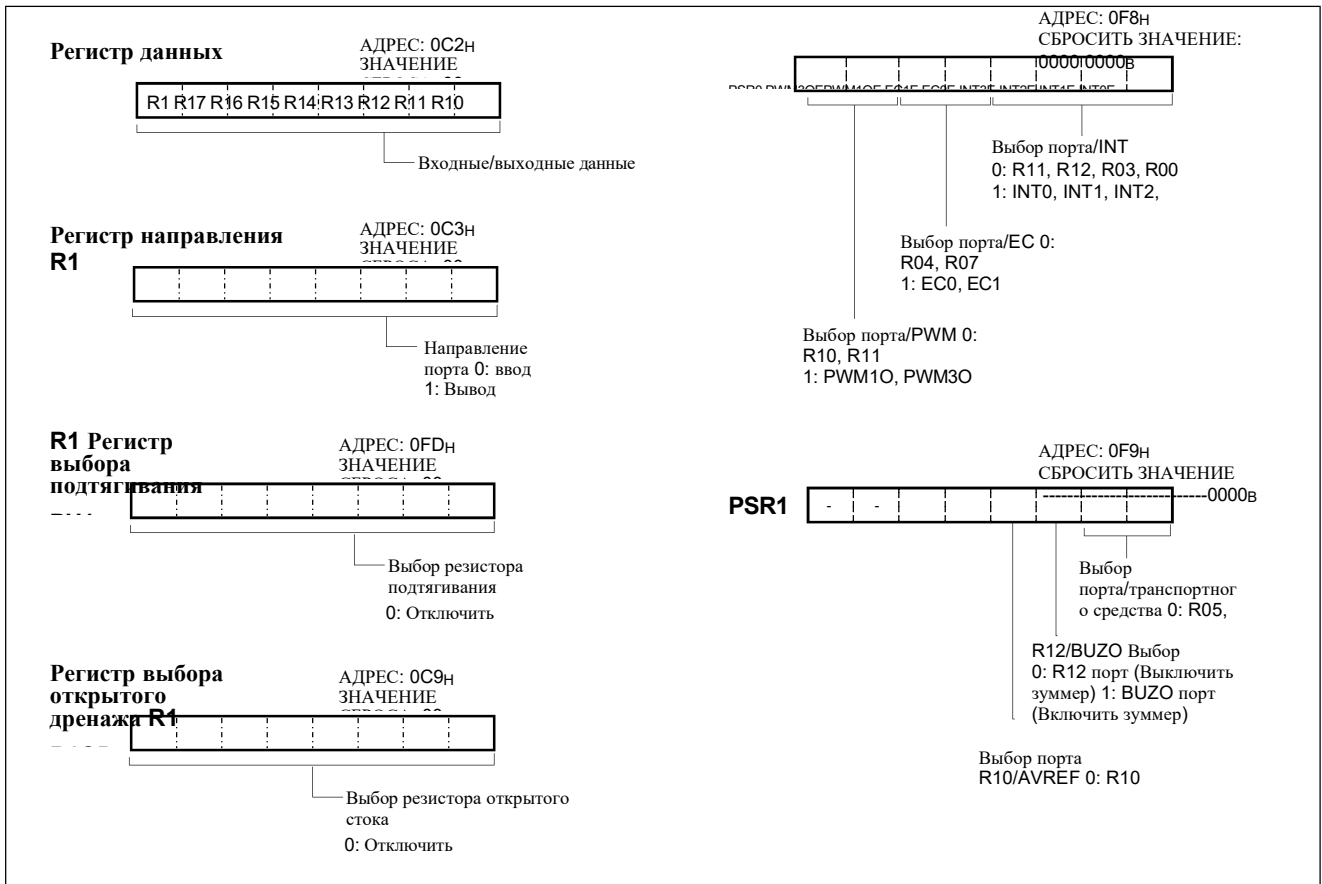


Рис. 9-3 Регистр портов R1

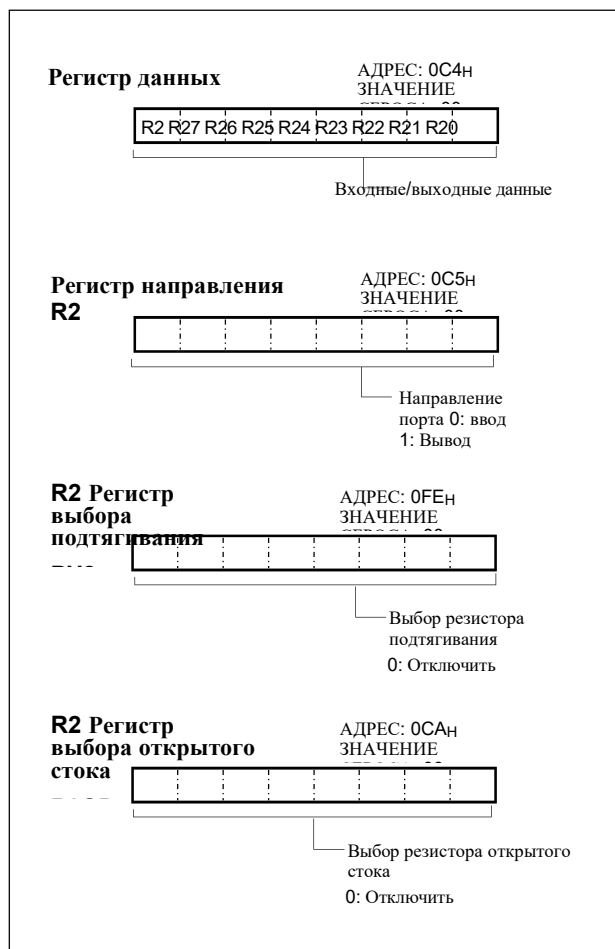
### 9.3 Регистр R2 и R2IO

R2 является 8-битным двунаправленным портом ввода-вывода CMOS (адрес 0C4H). Каждый вывод ввода/вывода может независимо использоваться в качестве входа или выхода через

регистр R3IO (адрес 0C5H). Когда в качестве входных портов используются контакты R20- R27, к ним может быть подсоединен встроенный резистор с 1-разрядным блоком с регистром 2 выбора подтягивания (PU2). Контакты R20-R27 могут использоваться для открытия выходного порта стока путем установки соответствующего бита регистра выбора открытого стока 2 (R2OD).

Кроме того, порт R2 мультиплексируется с альтернативными функциями. R23 R24,R25, and R26 может использоваться в качестве входного канала АЦП с 9 по 12 путем установки АЦП на включение АЦП и выбора канала с 9 по 12.

Контакт порта	Альтернативная функция
R20	-
R21	-
R22	-
R23	AN9 (входной канал АЦП 9)
R24	AN10 (входной канал АЦП 10)
R25	AN11 (входной канал АЦП 11)
R26	AN12 (входной канал АЦП 12)
R27	-



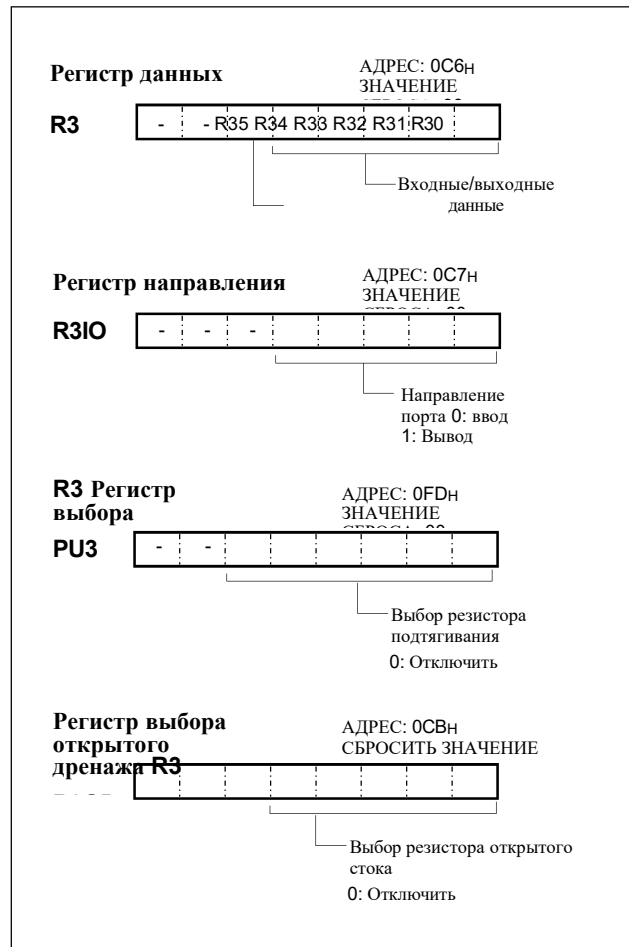
### 9.4 Регистр R3 и R3IO

R3 является 6-битным двунаправленным портом ввода-вывода CMOS (адрес 0C6H). Каждый вывод ввода/вывода (за исключением R35) может независимо использоваться в качестве входа или выхода через регистр R3IO (адрес 0C7H). R35 является портом только для входа. Когда контакты R30-R35 используются в качестве входных портов, к ним может быть подключен встроенный резистор подтягивания в 1-разрядных блоках с регистром 3 выбора подтягивания (PU3). Контакты R30-R34 могут использоваться для открытия выходного порта стока путем установки соответствующего бита регистра выбора открытого стока 1 (R3OD).

Кроме того, порт R3 мультиплексируется с альтернативными функциями. R30, R31, and R32 может использоваться в качестве входного канала АЦП 13, 14 и 15 путем установки АЦП для включения АЦП и выбора канала 13, 14 и 15.

Контакт порта	Альтернативная функция
R30	AN13 (входной канал АЦП 13)
R31	AN14 (входной канал АЦП 14)
R32	AN15 (входной канал АЦП 15)

R33, R34 и R35 мультиплексируются с выводом  $X_{IN}$ ,  $X_{OUT}$  и RESET.



### 10. ТАКОВЫЙ ГЕНЕРАТОР.

Как показано на Рис. 10-1, генератор синхросигналов вырабатывает базовые синхроимпульсы, которые обеспечивают подачу системных синхросигналов в ЦП и периферийные аппаратные средства. Содержит генератор основного тактового сигнала. Работа системного тактового генератора может быть легко обеспечена путем присоединения кристалла или керамического резонатора между X<sub>IN</sub> и X<sub>OUT</sub> контактами соответственно. Системный синхросигнал также может быть получен от внешнего генератора. В этом случае необходимо ввести внешний тактовый сигнал на вывод X<sub>IN</sub> и открыть вывод X<sub>OUT</sub>.

через разделенный на два триггер, но должны соблюдаться минимальные и максимальные высокие и низкие значения времени, указанные в листе технических данных.

Для периферийного блока может быть предусмотрен тактовый сигнал среди не разделенных исходных тактовых импульсов, разделенных на 1, 2, 4,..., до 4096. Тактовый сигнал периферийного устройства включается или отключается командой STOP. Управление периферийными тактовыми импульсами осуществляется посредством регистра управления тактовыми импульсами (CCCTLR). См. "П. ОСНОВНОЙ ТАЙМЕР ИНТЕРВАЛА" на стр. 45 для получения подробной информации.

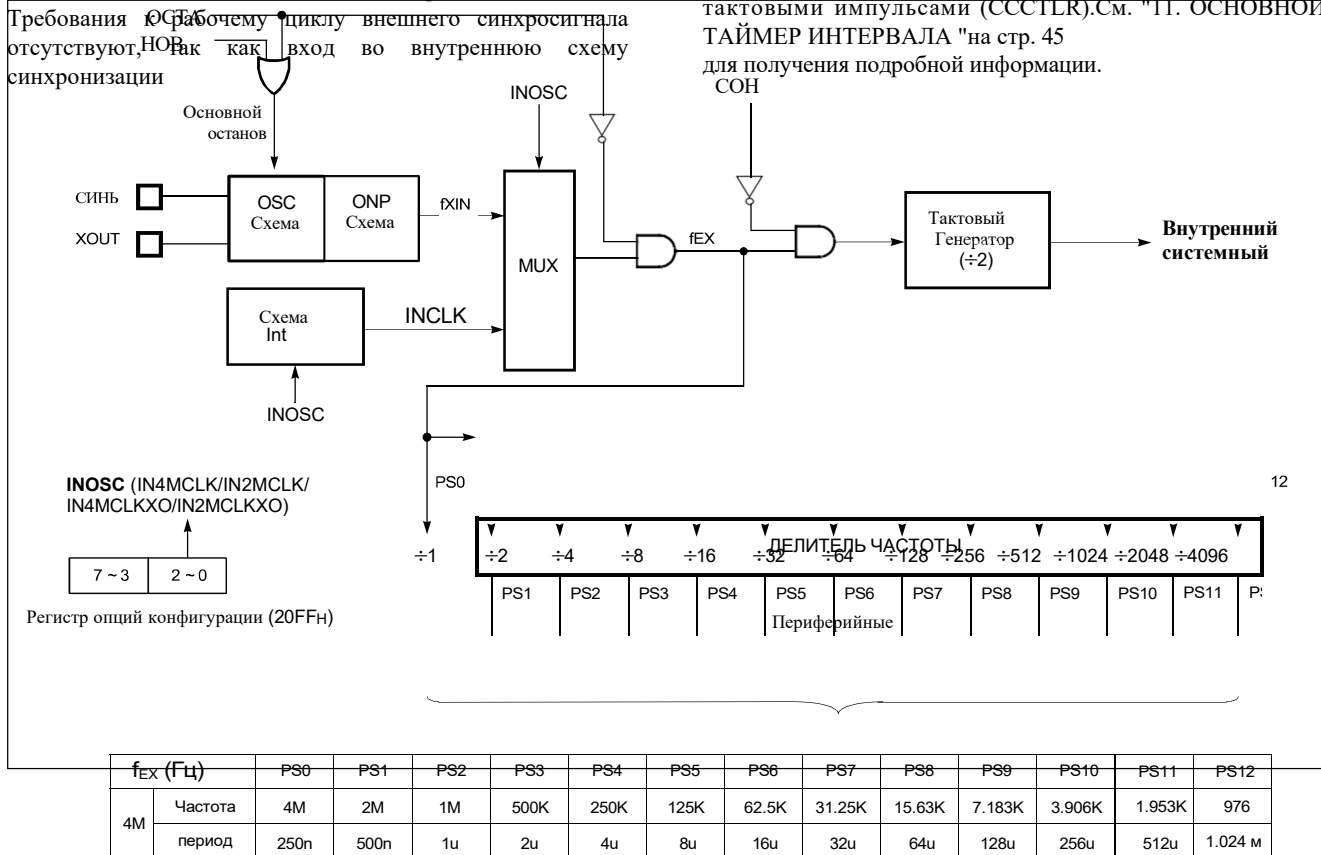


Рис. 10-1 Блок-схема генератора синхросигналов

#### 10.1 Схема колебаний

X<sub>IN</sub> и X<sub>OUT</sub> являются входным и выходным, соответственно, инвертирующим усилителем, который может быть установлен для использования в качестве интегрального генератора, как

показано на рис. 10-2.

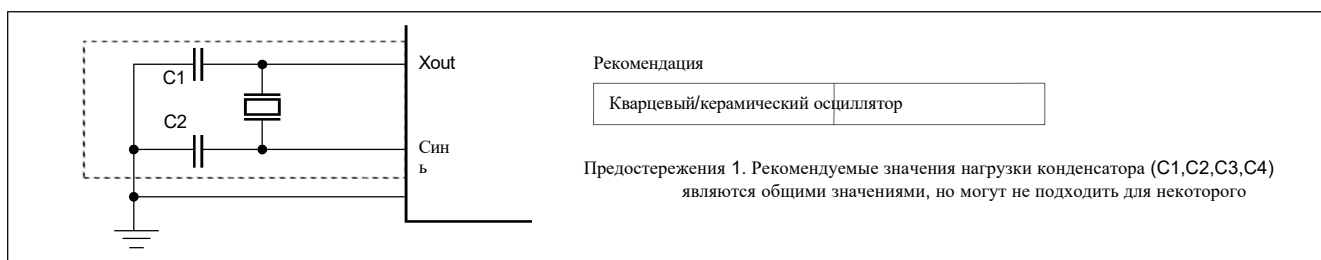


Рис. 10-2. Подключения осцилляторов



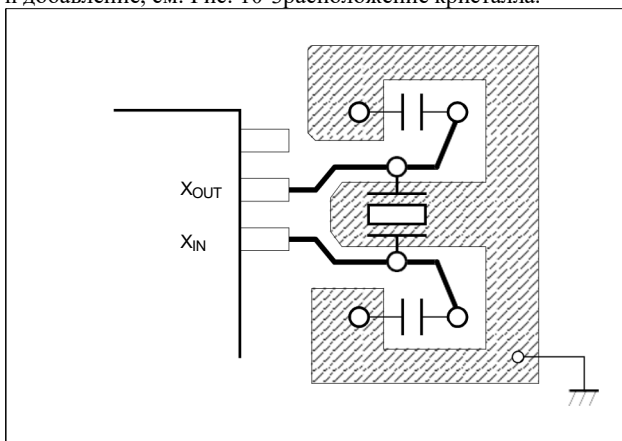
*Примечание. При использовании генератора системных синхросигналов выполните проводку в области пунктирной линии, показанной на рис. 10-2, для предотвращения каких-либо эффектов*

динамические мощности.

- Минимизируйте длину проводки.
- Не допускайте пересечения проводки с другими сигнальными проводниками.
- Не допускайте приближения проводки к изменению высокого тока.
- Установите потенциал положения заземления конденсатора генератора на потенциал Vss. Не заземляйте до любой модели заземления, где присутствует высокий ток.
- Не получайте сигналы от генератора.

генератора RC является функцией напряжения питания, значений внешнего резистора (R<sub>EXT</sub>) и конденсатора (C<sub>EXT</sub>), а также рабочей температуры.

п добавление, см. Рис. 10-3-расположение кристалла.

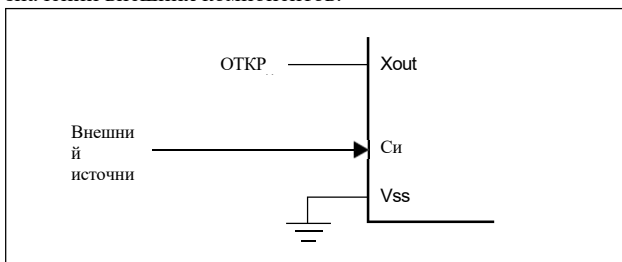


**Рис. 10-3** Схема схемы ПП осциллятора

Для управления устройством от внешнего источника синхросигнала необходимо оставить X<sub>out</sub> неподключенным, пока X<sub>in</sub> включен, как показано на Рис. 10-4

. Требования к рабочему циклу внешнего синхросигнала отсутствуют, так как вход в схему внутреннего синхросигнала осуществляется через триггер с разделением на два, но необходимо соблюдать минимальное и максимальное высокое и низкое значения времени, указанные в паспорте.

Колесательный контур предназначен для использования либо с керамическим резонатором, либо с кварцевым генератором. Поскольку каждый кристалл и керамический резонатор имеют свои собственные характеристики, пользователь должен проконсультироваться с производителем кристалла для получения соответствующих значений внешних компонентов.



**Рис. 10-4** Подключение внешних синхросигналов

Кроме того, MC80F0704/0708/0804/0808 обладает способностью к внешним колебаниям RC. Это обеспечивает дополнительную экономию затрат для приложений, не чувствительных к времени. Частота

Пользователь должен учитывать вариации из-за допуска используемых внешних компонентов R и C.

На рис. 10-1 показано, как комбинация RC соединена с MC80F0704/0708/0804/0808. Внешний конденсатор (C<sub>EXT</sub>) может быть опущен для большей экономии. Однако характеристики только экстерналичного колебания R более переменны, чем внешнее осциллирование RC.

**Рис. 10-1 Соединения осцилляторов RC**

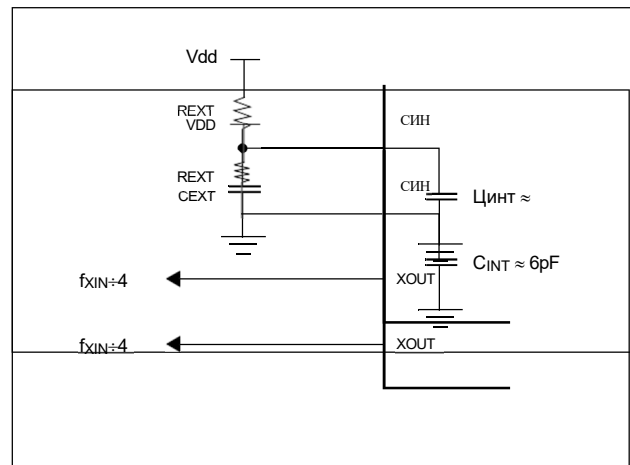
**Рис. 10-2 Соединения осцилляторов R**

Для использования RC-колебаний параметр CLK битов конфигурации (20FF<sub>H</sub>) должен быть установлен в значение «EXRC или EXRCXO».

Частота генератора, деленная на 4, выводится с Xout-вывода и может использоваться для проверки или для синхронизации другой логики.

В дополнение к внешнему кристаллу/резонатору и внешнему RC/R излучению, MC80F0704/0708/0804/0808 обеспечивает внутреннее 4MHz или 2MHz колебание. Внутреннее колебание 4MHz/2MHz не требует внешних частей.

Для использования внутреннего колебания 4MHz/2MHz параметр CLK битов конфигурации должен быть установлен в «IN4MCLK,» «IN2MCLK,» «IN4MCLKXO» или «IN2MCLKXO». Для получения подробной информации описание битов конфигурации см. в разделе "22. Область конфигурации устройства "на стр. 101



## 11. ОСНОВНОЙ ТАЙМЕР ИНТЕРВАЛА

У MC80F0704/0708/0804/0808 есть один 8-битный таймер основного интервала, который свободно запускается и не может останавливаться. Блок-схема показана на рис. 11-1. Кроме того, таймер основного интервала генерирует временную базу для подсчета таймера контрольного сигнала. Он также обеспечивает прерывание таймера интервала Basic (BITIF).

8-битный регистр таймера основного интервала (BITR) увеличивается каждый внутренний импульс подсчета, который делится на prescaler. Поскольку prescaler имеет отношение деления на 8 к 1024, скорость подсчета 1/8 к 1/1024 частоты генератора. Как переполнение счетчика из FFH в 00H, это переполнение вызывает генерацию прерывания.

Управление таймером основного интервала осуществляется регистром управления тактовыми импульсами (СКСТLR), показанным на рис. 11-2. Если бит RCWDT установлен в «1», источник синхронизации BITR изменяется на внутреннее колебание RC.

При записи «1» в бит BTCL СКСТLR регистр BITR сбрасывается в «0» и перезапускается для подсчета. Бит BTCL становится «0» после одного цикла машины аппаратными средствами.

Если команда STOP выполняется после записи «1» в бит RCWDT СКСТLR, то она переходит во внутренний режим колеблющегося сторожевого таймера RC. В этом режиме останавливается весь блок, за исключением внутреннего RC-генератора, основного таймера интервала и контрольного таймера. Более подробная информация приведена в разделе Функция энергосбережения. Битовый WDTON определяет контрольный таймер или обычный 7-битный таймер. Исходный синхросигнал может быть выбран младшими 3 битами СКСТLR.

BITR и СКСТLR расположены по одному адресу, и 0F2H адреса считывается как BITR и записывается в СКСТLR.

**Примечание:** Все управляющие биты таймера основного интервала находятся в регистре СКСТLR, который расположен по одному адресу BITR (адрес EC<sub>n</sub>). Адрес EC<sub>n</sub> читается как BITR, пишется в СКСТLR. Следовательно, доступ к СКСТLR с помощью команды манипулирования битами невозможен.

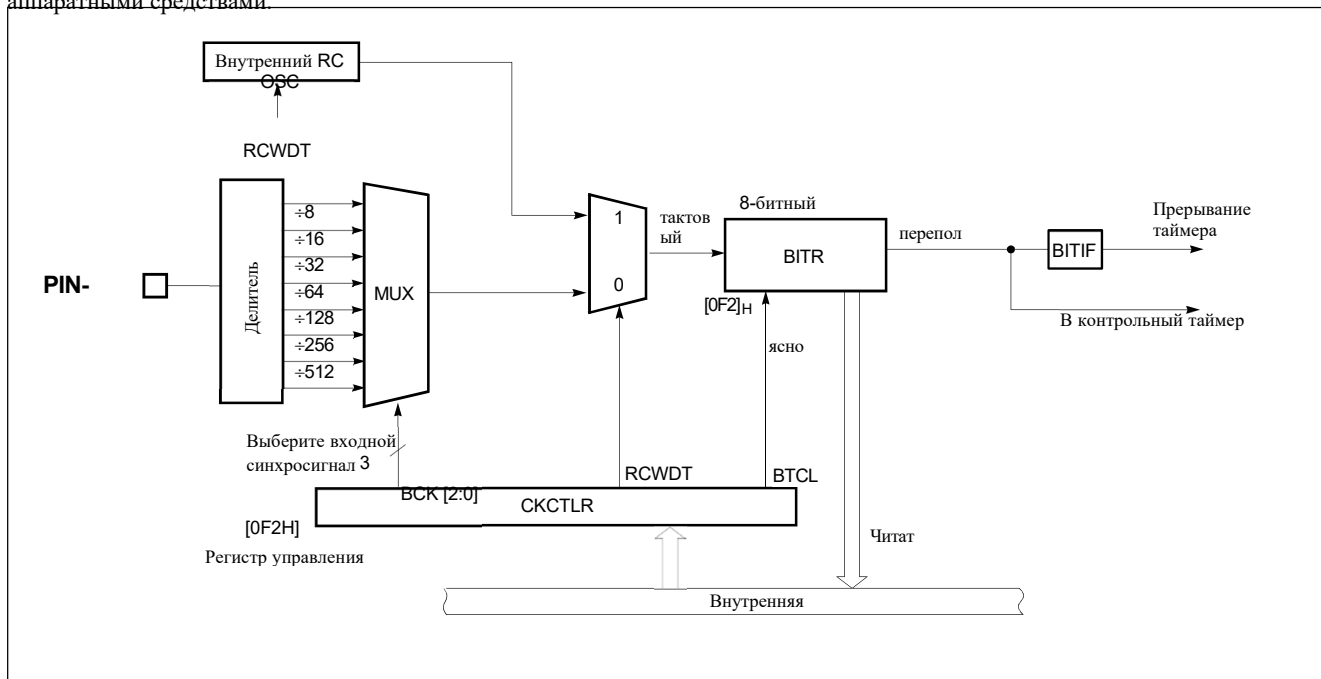


Рис. 11-1 Блок-схема основного таймера интервала

CCCTLR [2:0]	Тактовый генератор источника	Период прерывания (переполнения) (мс) @ f <sub>XIN</sub> = 8MHz
000	f <sub>XIN</sub> ÷ 8	0.256
001	f <sub>XIN</sub> ÷ 16	0.512
010	f <sub>XIN</sub> ÷ 32	1.024
011	f <sub>XIN</sub> ÷ 64	2.048
100	f <sub>XIN</sub> ÷ 128	4.096
101	f <sub>XIN</sub> ÷ 256	8.192
110	f <sub>XIN</sub> ÷ 512	16.384
111	f <sub>XIN</sub> ÷ 1024	32.768

Таблица 11-1 Период прерывания таймера основного интервала

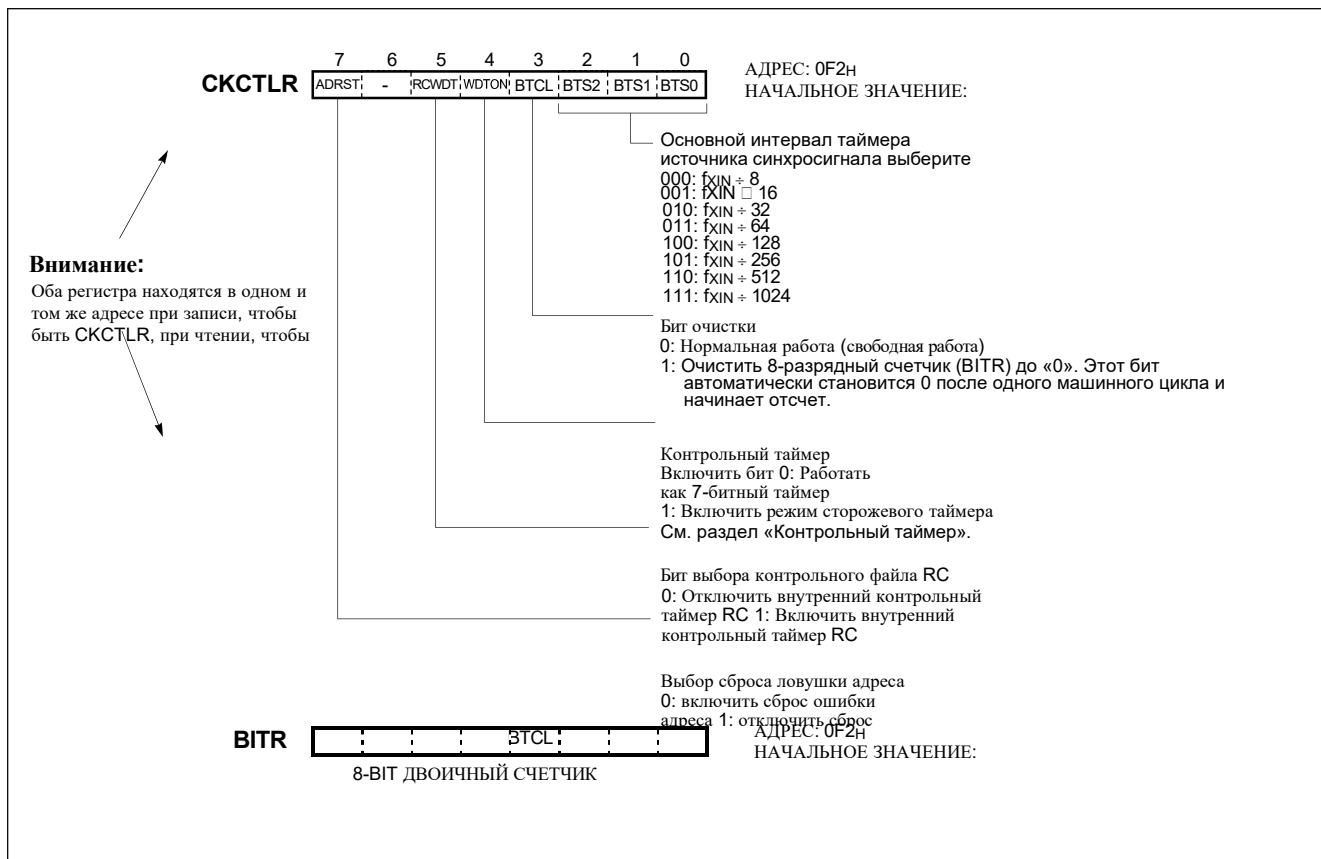


Рис. 11-2 BITR: Регистр режима таймера основного интервала

Пример 1:

Флаг запроса прерывания генерируется каждые 8.192мс при 4MHz.

```

:
LDM CCCTLR, #
1BH SET1 BITE
EI
:
    
```

Флаг запроса

Пример 2:

прерывания генерируется каждые 8.192мс при 8MHz.

```

:
LDM CCCTLR, #
1CH SET1 BITE
EI
:
    
```

## 12. СТОРОЖЕВОЙ ТАЙМЕР

Контрольный таймер быстро обнаруживает неисправность CPU, такую как бесконечное закольцовывание, вызванное шумом или подобным, и возобновляет работу CPU в нормальном состоянии. Сигнал сторожевого таймера для обнаружения неисправности может быть выбран либо как CPU сброса, либо как запрос прерывания.

Когда контрольный таймер не используется для устранения неисправности, он может использоваться в качестве таймера для генерации прерывания через фиксированные интервалы.

Контрольный таймер имеет два типа источника синхросигналов. Первый тип представляет собой встроенный RC-генератор, который не требует каких-либо внешних компонентов. Этот RC-генератор отделен от внешнего осциллятора XIN-контакта. Это означает, что контрольный таймер будет работать, даже если тактовый сигнал XIN-контакта устройства был остановлен, например, путем перехода в режим STOP. Другой тип - это предписанные системные часы.

Контрольный таймер состоит из 7-разрядного двоичного счетчика и регистра данных контрольного таймера. Когда значение 7-разрядного двоичного счетчика равно младшим 7 битам WDTR, генерируется флаг запроса прерывания. Это может использоваться в качестве контрольного таймера для прерывания или сброса ЦП в соответствии с битом WDTON.

**Примечание.** Поскольку счетчик сторожевого таймера активируется после очистки таймера основного интервала, то после установки бита WDTON в значение «1» максимальная ошибка таймера зависит от коэффициента prescaler таймера основного интервала. 7-разрядный двоичный счетчик очищается путем установки WDTCL (bit7 of WDTR), и WDTCL очищается автоматически после 1 машинного цикла.

Контрольный таймер RC активируется путем установки бита

RCWDT, как показано ниже.

```
LDM      CCCTLR, # 3FH; активизация WDT RC-OSC
LDM      WDTR, # 0FFH; установите период
WDT LDM  SSCR, # 5AH; ready для режима
STOP STOP          ; войти в режим STOP
NOP (СТОП)
НОП      ; Работа WDT RC-OSC
:
```

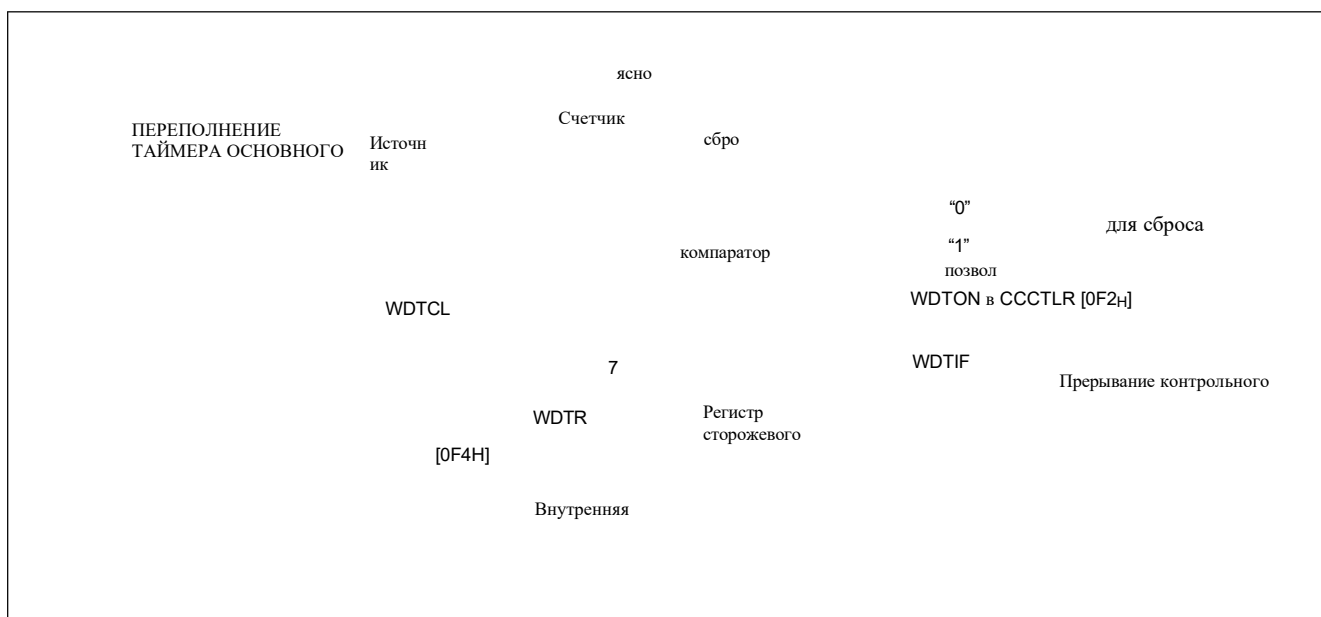
Период колебаний RC-WDT изменяется в зависимости от температуры, V<sub>DD</sub> и изменений процесса от части к части (приблизительно 33 ~ 100uS). Следующее уравнение показывает тайм-аут сторожевого таймера RCWDT.

$$T_{RCWDT} = CLK_{RCWDT} \times 2^8 \times WDTR + (CLK_{RCWDT} \times 2^8) / 2$$

где  $CLK_{RCWDT} = 33 \sim 100 \mu S$

Кроме того, этот контрольный таймер может использоваться как простой 7-битный таймер путем прерывания WDTIF. Интервал прерывания контрольного таймера определяется параметром Basic Interval Timer. Уравнение интервала представлено ниже.

$$T_{WDT} = (WDTR + 1) \times \text{Интервал BIT}$$



**Рис. 12-1** Блок-схема контрольного таймера

**Управление контрольным таймером**

Рис. 12-2 На показан регистр управления сторожевым таймером.

после сброса автоматически отключается

контрольный таймер.

Неисправность CPU обнаруживается при установке функции обнаружения

время, выбор выходного сигнала и очистка двоичного счетчика. Очистка двоичного счетчика повторяется в течение времени обнаружения.

Если неисправность возникает по какой-либо причине, выходной таймер контрольного сигнала становится активным при возрастающем переполнении двоичных счетчиков, если двоичный счетчик не очищен. В это время, когда WDTON = 1, генерируется сброс, который приводит контакт RESET в

для сброса внутреннего аппаратного обеспечения. Когда WDTON = 0, генерируется прерывание контрольного таймера (WDTIF). Бит WDTON находится в регистре CLKCTLR.

Контрольный таймер временно останавливает подсчет в режиме STOP, и когда режим STOP освобождается, он автоматически повторно запускается (продолжает подсчет).

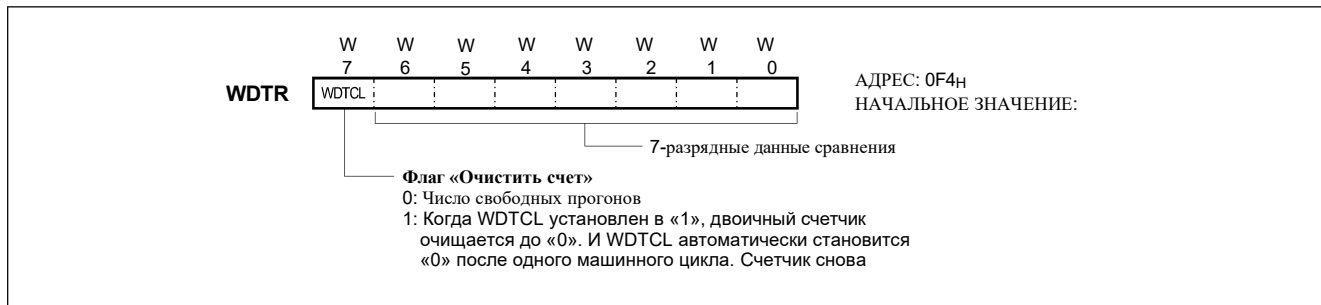


Рис. 12-2 WDTTR: Регистр управления контрольным таймером

Пример: Устанавливает время обнаружения сторожевого таймера 1 сек. в 4.194304MHz

```

LDM CCCTLR, # 3FH; Выбор источника синхросигнала 1/1024, WDTON ← 1, счетчик сброса
LDM WDTTR, # 08FH

LDM WDTTR, # 08FH; Счетчик сброса
:
:
В пределах W: DT detection time
:
LDM WDTTR, # 08FH; Счетчик сброса
:
:
В пределах W: DT detection time
LDM WDTTR, # 08FH; Счетчик сброса
    
```

**Включить и отключить контрольный журнал**

Контрольный таймер включается путем установки для WDTON (бит 4 в CCCTLR) значения «1». WDTON инициализируется в «0» во время сброса и должен быть установлен в «1» для работы после сброса.

Пример: Включение контрольного таймера для сброса

```

:
LDM CCCTLR, # xxx1 _ xxxxB;
WDTON ← 1
:
:
    
```

Контрольный таймер деактивируется посредством сброса бита 4 (WDTON) CCCTLR. Контрольный таймер останавливается в режиме STOP и включается автоматически после выхода из режима STOP.

**Прерывание**

**контрольного таймера**

Контрольный таймер может также использоваться как простой 7-битный таймер путем сброса бита 4 CCCTLR на «0». Интервал прерывания контрольного таймера определяется параметром Basic Interval Timer. Уравнение интервала показано ниже.

$$T_{WDT} = (WDTTR + 1) \times \text{Интервал BIT}$$

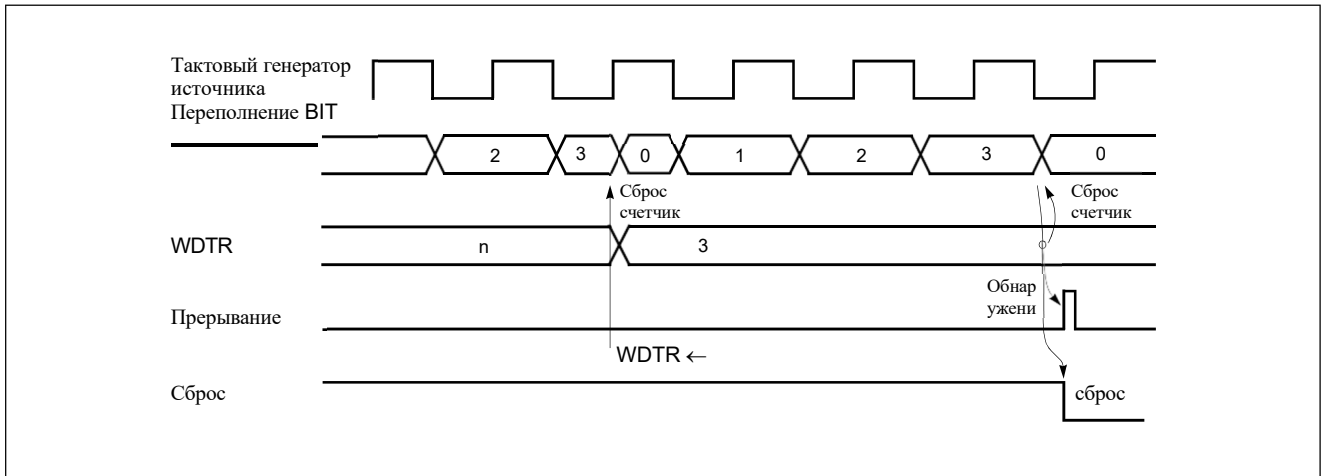
Указатель стека (SP) должен быть инициализирован перед использованием вывода контрольного таймера в качестве источника прерывания.

Пример: установка 7-разрядного прерывания таймера.

```

LDM CCCTLR, # xxx0 _ xxxxB;
WDTON ← 0 LDM WDTTR, # 8FH ;
WDTCL ← 1
    
```





**Рис. 12-3 Синхронизация контрольного таймера**

Если выход контрольного таймера становится активным, генерируется сброс, который приводит в действие вывод RESET low для сброса внутреннего оборудования.

аппарат генерируется в режиме субчасов.

Основной тактовый генератор также включается при повторном включении контрольного таймера

### 13. СЧЕТЧИК ТАЙМЕРОВ/СОБЫТИЙ

В MC80F0704/0708/0804/0808 имеется четыре таймера/счетчика. Каждый модуль может генерировать прерывание для указания того, что произошло событие (т.е. совпадение таймера).

Таймер 0 и таймер 1 могут использоваться либо два 8-битных таймера/счетчика, либо один 16-битный таймер/счетчик с объединением этих таймеров. Также таймер 2 и таймер 3 одинаковы. Таймер 4 - это 16-битный таймер/счетчик.

В функции «таймер» регистр увеличивается при каждом входе внутреннего синхросигнала. Таким образом, его можно рассматривать как подсчет внутреннего входа тактового сигнала. Поскольку наименьший синхросигнал состоит из 2, а наибольший синхросигнал состоит из 2048 периодов генератора, скорость подсчета 1/2 к 1/2048 частоты генератора.

В функции «счетчик» регистр увеличивается в ответ на переход от 0 к 1 (передний фронт) на соответствующем внешнем входном выводе, EC0 или EC1.

Кроме функции «захвата», регистр увеличивается в респонсовых внешних или внутренних тактовых источниках так же, как с таймером или

функция счетчика. При входе внешнего фронта синхросигнала счетчик регистрируется в регистре данных таймера соответственно. При входе внешнего фронта синхросигнала регистр счетчика записывается в регистр CDRx данных capture.

Таймер 0 и таймер 1 совместно используются с функцией «PWM» и функцией «Com- pare output». Он имеет шесть режимов работы: «8-битный таймер/счетчик», «16-битный таймер/счетчик», «8-битный захват», «16-битный capture», «8-битный выход сравнения» и «10-битный ШИМ», которые выбираются по биту в регистре режима таймера TM0 и TM1, как показано в таблице 13-1, рис. 13-1.

Таймер 2 и таймер 3 совместно используются с функцией «PWM» и функцией «Com- pare output». Он имеет шесть режимов работы: «8-битный таймер/счетчик», «16-битный таймер/счетчик», «8-битный захват», «16-битный capture», «8-битный выход сравнения» и «10-битный ШИМ», которые выбираются по биту в регистре режима таймера TM2 и TM3, как показано в таблице 13-2, рис. 13-2.

16 БИТОВ	CAP0	CAP1	PWM1E	T0СК [2:0]	T1СК [1:0]	PWM1O	ТАЙМЕР 0	ТАЙМЕР 1
0	0	0	0	XXX	XX	0	8-битный таймер	8-битный таймер
0	0	1	0	111	XX	0	8-разрядный счетчик событий	8-битный захват
0	1	0	0	XXX	XX	1	8-битный захват (внутренние часы)	8-разрядный вывод сравнения
0	X	0	1	XXX	XX	1	8-битный таймер/счетчик	10-битный PWM
1	0	0	0	XXX	11	0	16-битный таймер	
1	0	0	0	111	11	0	16-разрядный счетчик событий	
1	1	1	0	XXX	11	0	16-битный захват (внутренние часы)	

Таблица 13-1 Режимы работы таймера 0, 1

1. X означает, что значение «0» или «1» соответствует операции пользователя.

16 БИТОВ	CAP2	CAP3	PWM3E	T2СК [2:0]	T3СК [1:0]	PWM3O	ТАЙМЕР 2	ТАЙМЕР 3
0	0	0	0	XXX	XX	0	8-битный таймер	8-битный таймер
0	0	1	0	111	XX	0	8-разрядный счетчик событий	8-битный захват
0	1	0	0	XXX	XX	1	8-битный захват (внутренние часы)	8-разрядный вывод сравнения
0	X	0	1	XXX	XX	1	8-битный таймер/счетчик	10-битный PWM
1	0	0	0	XXX	11	0	16-битный таймер	

1	0	0	0	111	11	0	16-разрядный счетчик событий
1	1	1	0	XXX	11	0	16-битный захват (внутренние часы)

**Таблица 13-2 Режимы работы таймера 2, 3**

**TM0**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
5	4	3	2	1	0		
-	-	CAP0	T0CK2	T0CK1	T0CK0	T0CN	T0ST

АДРЕС: 0D0h  
НАЧАЛЬНОЕ ЗНАЧЕНИЕ: --00 0000<sub>B</sub>

	Описание позиции	бита Bit Name	
CAP0	TM0.5		0: режим таймера/счетчика 1: Флаг выбора режима захвата
T0CK2	TM0.4		000: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 2$ 001: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 4$ 010: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 8$ 011: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 32$ 100: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 128$ 101: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 512$ 110: 8-битный таймер, источник синхросигнала - $f_{XIN} \div 2048$ 111: E00 (Внешний синхросигнал)
T0CN	TM0.1		0: пауза счетчика таймера 1: запуск счетчика таймера
T0ST	TM0.0		0: После очистки остановите подсчет. 1: При установке таймер 0 Count Register сбрасывается и запускается снова.

**TM1**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0
ПОЛ	16	PWM1E	CAP1	T1CK1	T1CK0	T1CN	T1ST

НАЧАЛЬНОЕ ЗНАЧЕНИЕ: 00h

Имя бита	Позиция бита	Описание
ПОЛИТИК	TM1.7	0: Рабочее состояние PWM - активный низкий уровень 1: Функция PWM Active High
16 БИТОВ	TM1.6	0: 8-битный режим 1: 16-битный режим
PWM1E	TM1.5	0: Отключить ШИМ 1: Включить ШИМ
CAP1	TM1.4	0: Режим таймера/счетчика 1: Флаг выбора режима захвата
T1CK1	TM1.3	00: 8-битный таймер, источник синхросигнала $f_{XIN}$
T1CK0	TM1.2	01: 8-битный таймер, источник синхросигнала $f_{XIN} \div 2$ 10: 8-битный таймер, источник синхросигнала $f_{XIN} \div 8$ 11: 8-битный таймер, источник синхросигналов использует таймер 0
T1CN	TM1.1	0: Пауза счетчика таймера 1: Начало подсчета таймеров
T1ST	TM1.0	0: После очистки остановите подсчет. 1: При установке таймер 0 Count Register сбрасывается и запускается снова.

**TDR0**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0		

АДРЕС: 0D1h  
НАЧАЛЬНОЕ ЗНАЧЕНИЕ: 0FFh

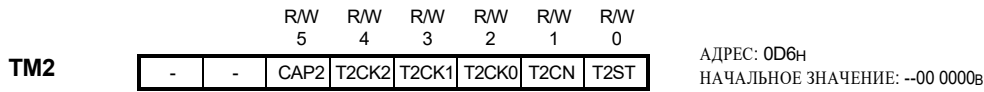
**TDR1**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0		

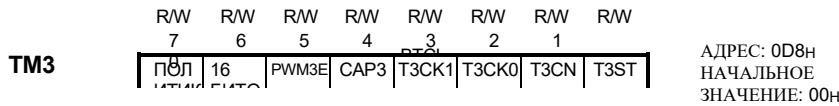
АДРЕС: 0D3h  
НАЧАЛЬНОЕ ЗНАЧЕНИЕ: 0FFh

Read: Count value read  
Write: Compare data write

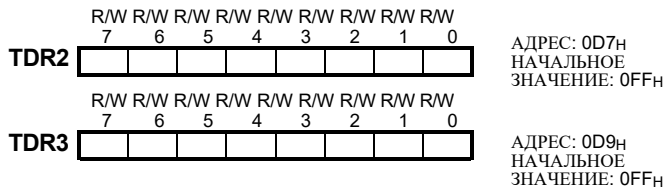
Рис. 13-1 ТМ0, Регистры ТМ1



Имя бита	Позиция бита	Описание
CAP2	TM2.5	0: режим таймера/счетчика 1: Флаг выбора режима захвата
T2CK2	TM2.4	000: 8-битный таймер, источник синхросигнала fXIN ÷ 2
T2CK1	TM2.3	001: 8-битный таймер, источник синхросигнала fXIN ÷ 4
T2CK0	TM2.2	010: 8-битный таймер, источник синхросигнала fXIN ÷ 8 011: 8-битный таймер, источник синхросигнала fXIN ÷ 16 100: 8-битный таймер, источник синхросигнала fXIN ÷ 64 101: 8-битный таймер, источник синхросигнала fXIN ÷ 256 110: 8-битный таймер, источник синхросигнала - fXIN ÷ 1024 111: EC1 (внешние часы)
T2CN	TM2.1	0: Пауза счетчика таймера 1: Начало подсчета таймеров
T2ST	TM2.0	0: После очистки остановите подсчет. 1: При установке таймер 0 Count Register сбрасывается и запускается снова.



Имя бита	Позиция бита	Описание
ПОЛИТИК	TM3.7	0: Рабочее состояние PWM - активный низкий уровень 1: Функция PWM Active High
16 БИТОВ	TM3.6	0: 8-битный режим 1: 16-битный режим
RWМЗЕ	TM3.5	0: Отключить ШИМ 1: Включить ШИМ
CAP3	TM3.4	0: Режим таймера/счетчика 1: Флаг выбора режима захвата
T3CK1	TM3.3	00: 8-битный таймер, источник синхросигнала fXIN
T3CK0	TM3.2	01: 8-битный таймер, источник синхросигнала fXIN ÷ 4 10: 8-битный таймер, источник синхросигнала fXIN ÷ 16 11: 8-битный таймер, источник синхросигнала использует таймер 2
T3CN	TM3.1	0: Пауза счетчика таймера 1: Начало подсчета таймеров
T3ST	TM3.0	0: После очистки остановите подсчет. 1: При установке таймер 0 Count Register сбрасывается и запускается снова.



Read: Count value read  
Write: Compare data write

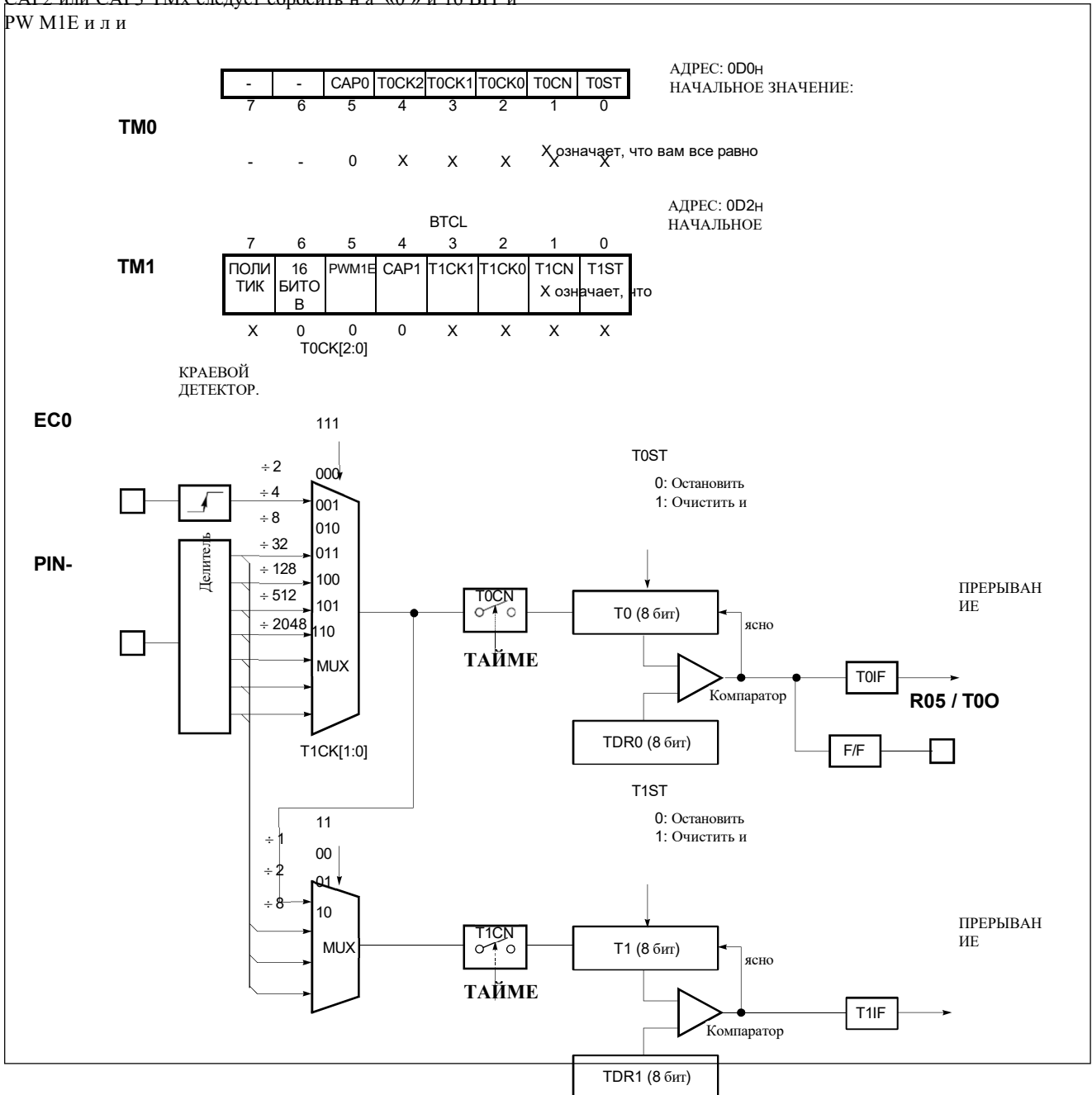
Рис. 13-2 TM2, регистры TM3

### 13.1 8-битный режим таймера/счетчика

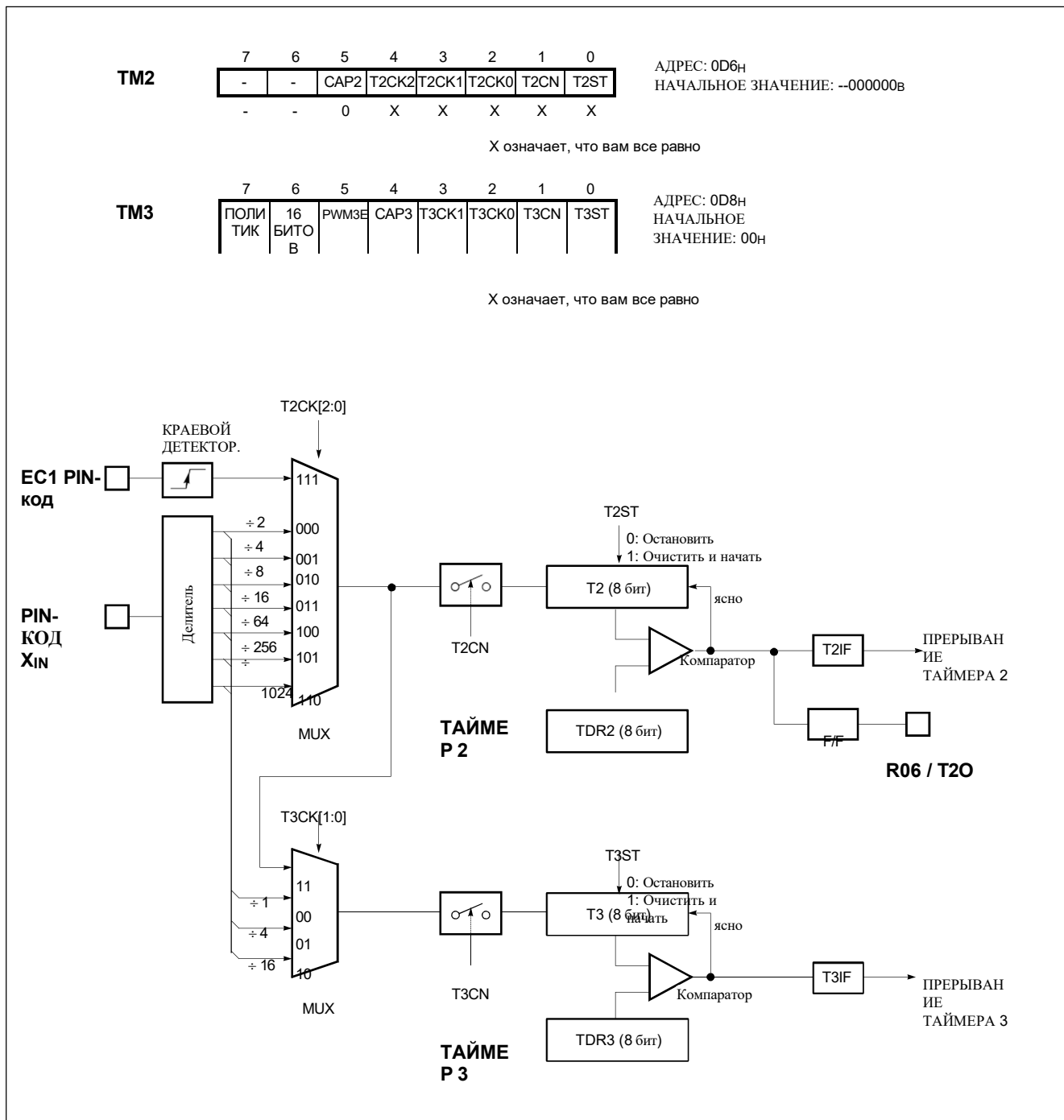
В MC80F0704/0708/0804/0808 имеется четыре 8-битных счетчика таймера, таймера 0, таймера 1, таймера 2, таймера 3. Таймер 0, таймер 1 показаны на, Рис. 13-3а таймер 2, таймер 3 показаны на рис. 13-4.

Функция «таймер» или «счетчик» выбирается контрольными регистрами TM0, TM1, TM2, TM3, как показано на. Рис. 13-1. Для использования в качестве 8-разрядного режима таймера/счетчика бит CAP0, CAP1, CAP2 или CAP3 TMx следует сбросить на «0» и 16 BIT и PW M1E или

PWM3E TM1 или TM3 следует сбросить до «0» (). Рис. 13-3. Эти таймеры имеют каждый 8-битный регистр подсчета и регистр данных. Регистр счетчика увеличивается на каждый внутренний или внешний вход тактового генератора. Внутренний синхросигнал имеет опцию предварительного деления 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048 или внешний синхросигнал (выбираемый управляющими битами TxСК0, TxСК1, TxСК2 регистра TMx).



**Рис. 13-3 8-битный таймер/счетчик 0, 1**





**Пример 1:**

Timer0 = 2 способа таймера 8 битов мс в  
 Timer1 на 4 МГц = 0,5 способа таймера 8  
 битов мс в Timer2 на 4 МГц = 1 способ  
 таймера 8 битов мс в Timer3 на 4 МГц =  
 1 способ таймера 8 битов мс на уровне 4  
 МГц

```
TDR0 LDM, # 249
TDR1 LDM, # 249
TDR2 LDM, # 249
TDR3 LDM, # 249
LDM TM0, # 0000
1111B LDM TM1, # 0000
1011B LDM TM2, #
0000 1111B LDM TM3,
# 0000 - 1011B SET1
T0E
SET1 T1E
SET1 T2E
SET1 T3E
EI
```

**Пример 2:**

Timer0 = 8-битный режим счетчика  
 событий Timer1 = 0,5 мс 8-битный  
 режим таймера при 4MHz Timer2 = 8-  
 битный р е ж и м счетчика событий  
 Timer3 = 1 мс 8-битный режим таймера  
 при 4MHz

```
LDM TDR0,
LDM #249
LDM TDR1,
#249
TDR2,
#249
TDR3,
#249
LDM TM0,
#0001 1111B
LDM TM1,
#0000 1011B
LDM TM2,
#0001 1111B
```

Эти таймеры имеют каждый 8-битный регистр подсчета и регистр данных. Регистр счетчика увеличивается на каждый внутренний или внешний вход тактового генератора. Внутренний синхросигнал имеет опцию prescaler divide ratio из 2, 4, 8, 32, 128, 512, 2048, выбранных управляющими битами T0CK [2:0] регистра TM0 или 1, 2, 8, выбранных контрольными битами T1CK [1:0] регистра TM1 или 2, 4, 8, 16, 64, 256, 1024, выбранного контрольными битами T2CK [2:0] регистра TM2, или 1, 4, 16, выбранный управляющими битами T3CK [1:0] регистра TM3. В таймере 0 регистр таймера T0 складывается из 00H до совпадения TDR0 а затем сбрасывается в 00H. Выходной сигнал согласования таймера 0 генерирует прерывание таймера 0 (фиксируется в T0IF бите).

В функции счетчика счетчик увеличивается через каждый переход от 0 к 1 (передний край) EC0 штыря. Для использования функции счетчика битовый EC0 регистра выбора порта (PSR0.4) устанавливается в «1». Таймер 0 может использоваться в качестве счетчика по входу контактного EC0, но таймер 1 не может использоваться. Аналогично, чтобы использовать Timer2 в качестве функции счетчика, битовый EC1 регистра выбора порта (PSR0.5) устанавливается в «1». Таймер 2 может использоваться в качестве счетчика по входу контактного EC1, но таймер 3 не может.

**8-битный режим таймера**

В режиме таймера для подсчета используется внутренний синхросигнал. Таким образом, вы можете думать о нем как о подсчете внутреннего входа тактового сигнала. Содержимое TDRn сравнивается с содержимым счетчика Tn. Если совпадение найдено, генерируется прерывание таймера n (TnIF), и счетчик повышения сбрасывается на 0. Подсчет вверх возобновляется после сброса счетчика up-counter.

Поскольку значение TDRn может изменяться программным обеспечением, интервал времени устанавливается по мере необходимости.

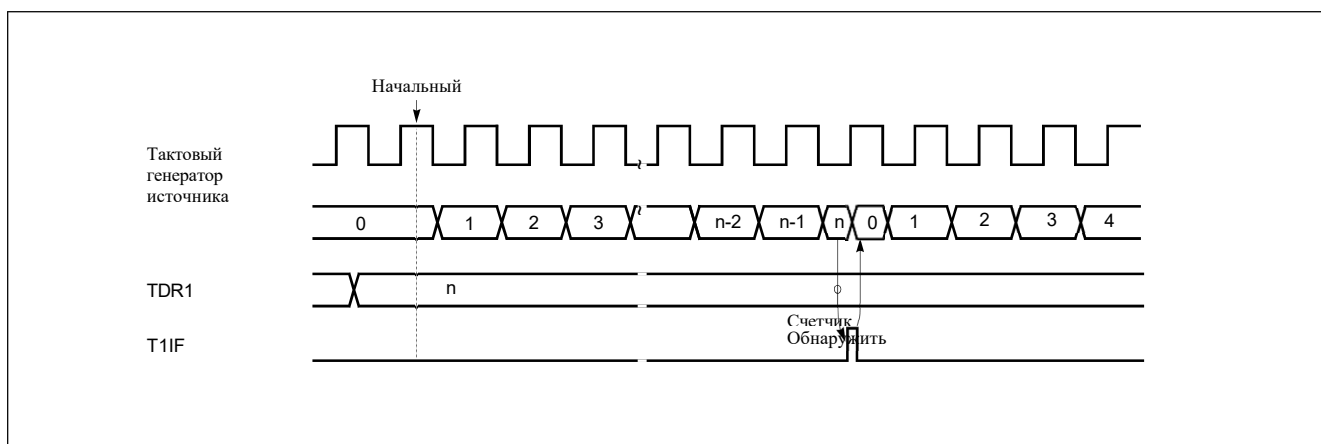


Рис. 13-5 Схема синхронизации  
режима таймера

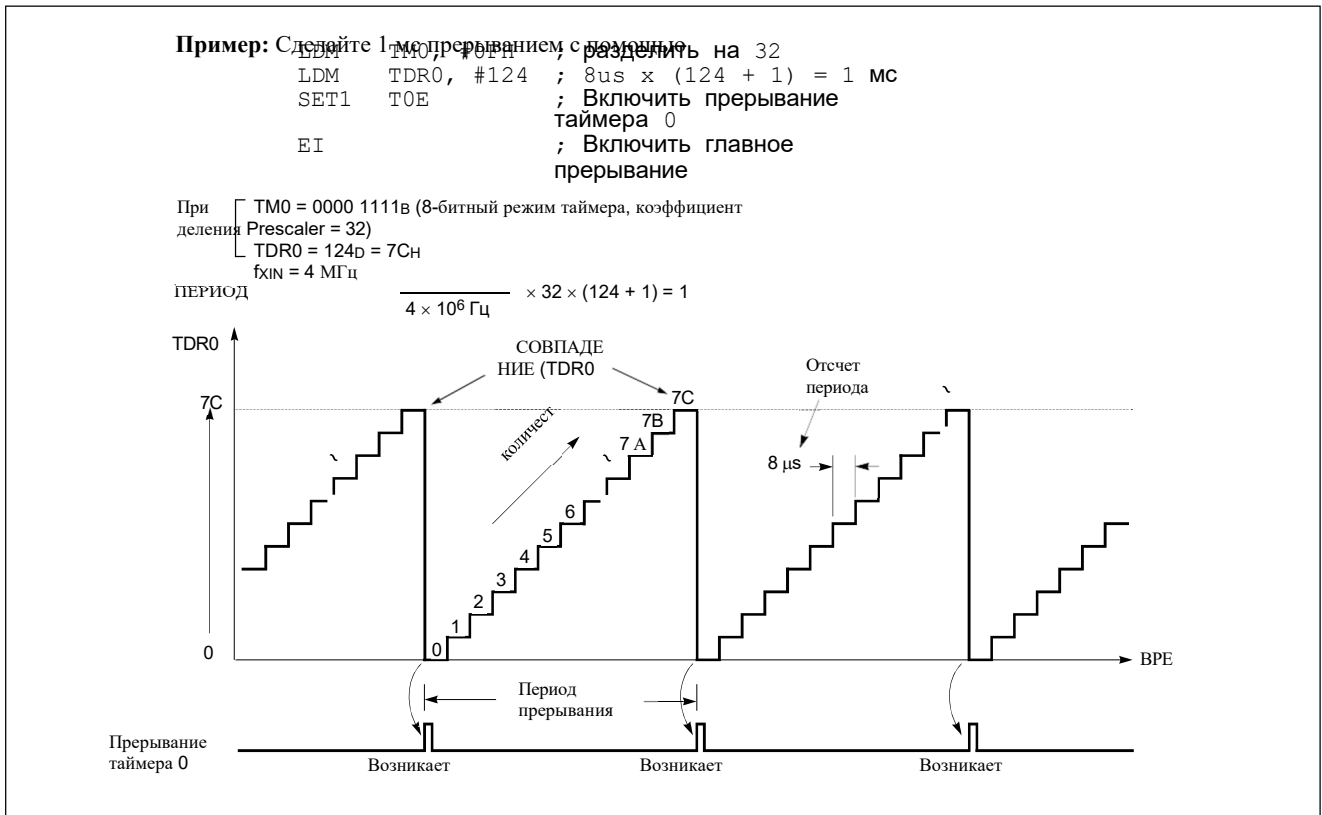


Рис. 13-6 Пример подсчета таймеров

**8-разрядный режим счетчика событий**

В этом режиме подсчет начинается с внешнего триггера. Этот триггер означает передний край входа EC0 или EC1 контакта. Исходный синхросигнал используется в качестве внутреннего синхросигнала, выбранного в режиме таймера register TM0 или TM2. Содержимое регистра данных таймера TDRn (n = 0,1,2,3) сравнивают с содержимым повышающего счетчика Tn. Если совпадение найдено, генерируется флаг TnIF запроса прерывания таймера, и счетчик сбрасывается на «0». Счетчик перезапускается и непрерывно подсчитывается каждым передним фронтом входа EC0 или EC1 вывода. Максимальная частота, применяемая к контакту EC0 или EC1, равна  $f_{XIN}/2$  [Гц].

Для использования функции счетчика событий биты 4, 5 регистра выбора порта PSR0 (адрес 0F8H) должны быть установлены в «1».

После сброса значение регистра TDRn данных таймера инициализируется равным «0».

1

$$\text{Период (сек)} = \frac{1}{f_{XIN}} \times 2 \times \text{Отношение} \times (TDRn + 1) \text{ деления}$$

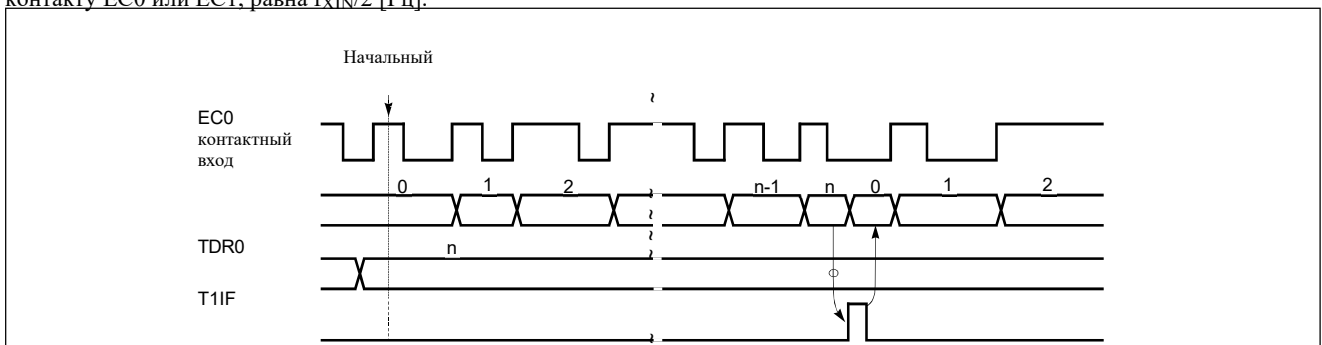


Рис. 13-7 Схема синхронизации режима счетчика событий

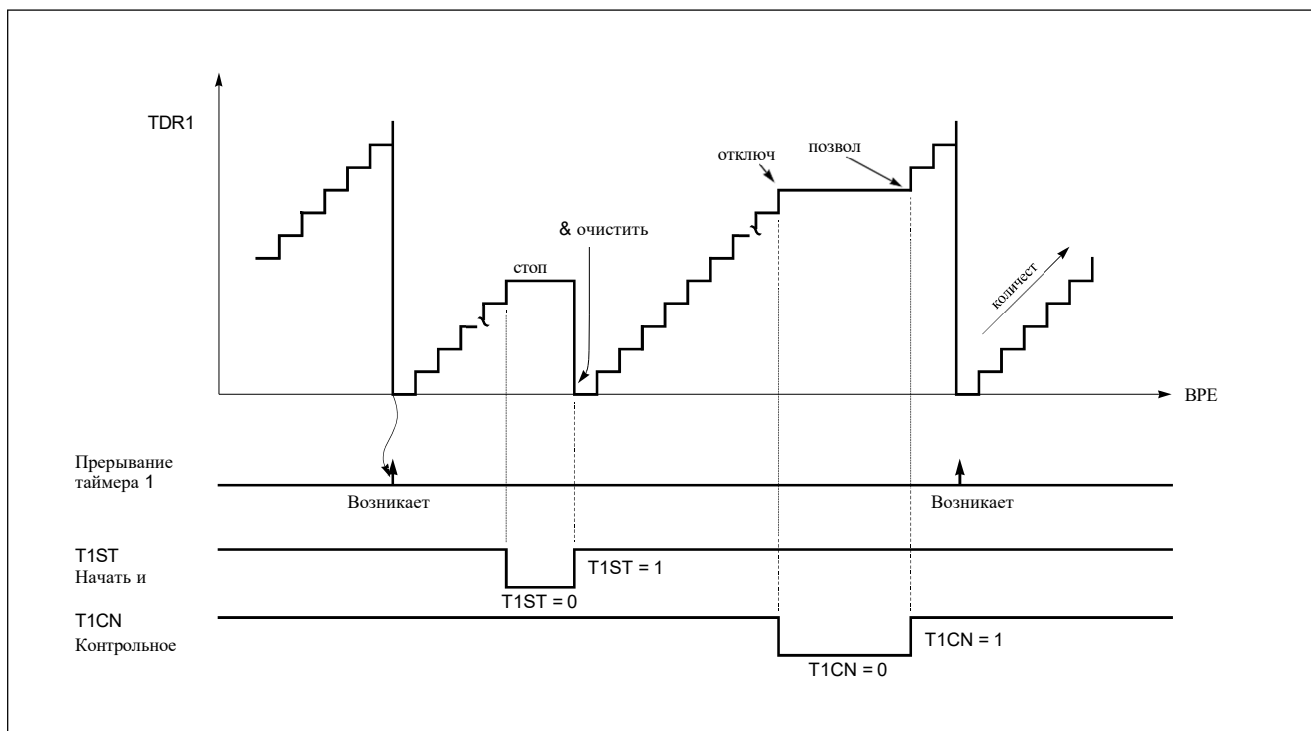


Рис. 13-8 Работа счетчика таймеров/событий

### 13.2 16-битный режим таймера/счетчика

Регистр таймера выполняется со всеми 16 битами. 16-битный регистр таймера/счетчика T0, T1 увеличивается от 0000<sub>H</sub> до тех пор, пока он не совпадет с TDR0, TDR1 и затем сбрасывается до 0000<sub>H</sub>. Выход совпадения генерирует прерывание таймера 0.

Источник синхросигнала 0 таймера выбирается либо внутренним, либо внешним синхросигналом по биты T0CK [2:0]. В 16-разрядном режиме биты T1CK [1:0] и 16BIT TM1 должны быть установлены в «1» соответственно, как показано на рис. 13-9.

Аналогично, регистр T2, T3 16-битного таймера/счетчика увеличивается от 0000<sub>H</sub> до тех пор, пока он не совпадет с TDR2, TDR3 и затем сбрасывается до 0000<sub>H</sub>. Выходной сигнал сопоставления генерирует прерывание таймера 2.

Источник синхросигнала 2 таймера выбирается либо внутренним, либо внешним синхросигналом по биты T2CK [2:0]. В 16-разрядном режиме биты T3CK [1:0] и 16BIT TM3 должны быть установлены в «1» соответственно, как показано на рис. 13-10.

Даже если таймер 0 (включая таймер 1) используется как 16-битный таймер, таймер 2 и таймер 3 по-прежнему могут использоваться как два 8-битных таймера или один 16-битный таймер путем установки TM3. В обратном направлении, даже если таймер 2 (включая таймер 3) используется в качестве 16-битного таймера, таймер 0 и таймер 1 по-прежнему могут использоваться в качестве 8-битного таймера независимо.

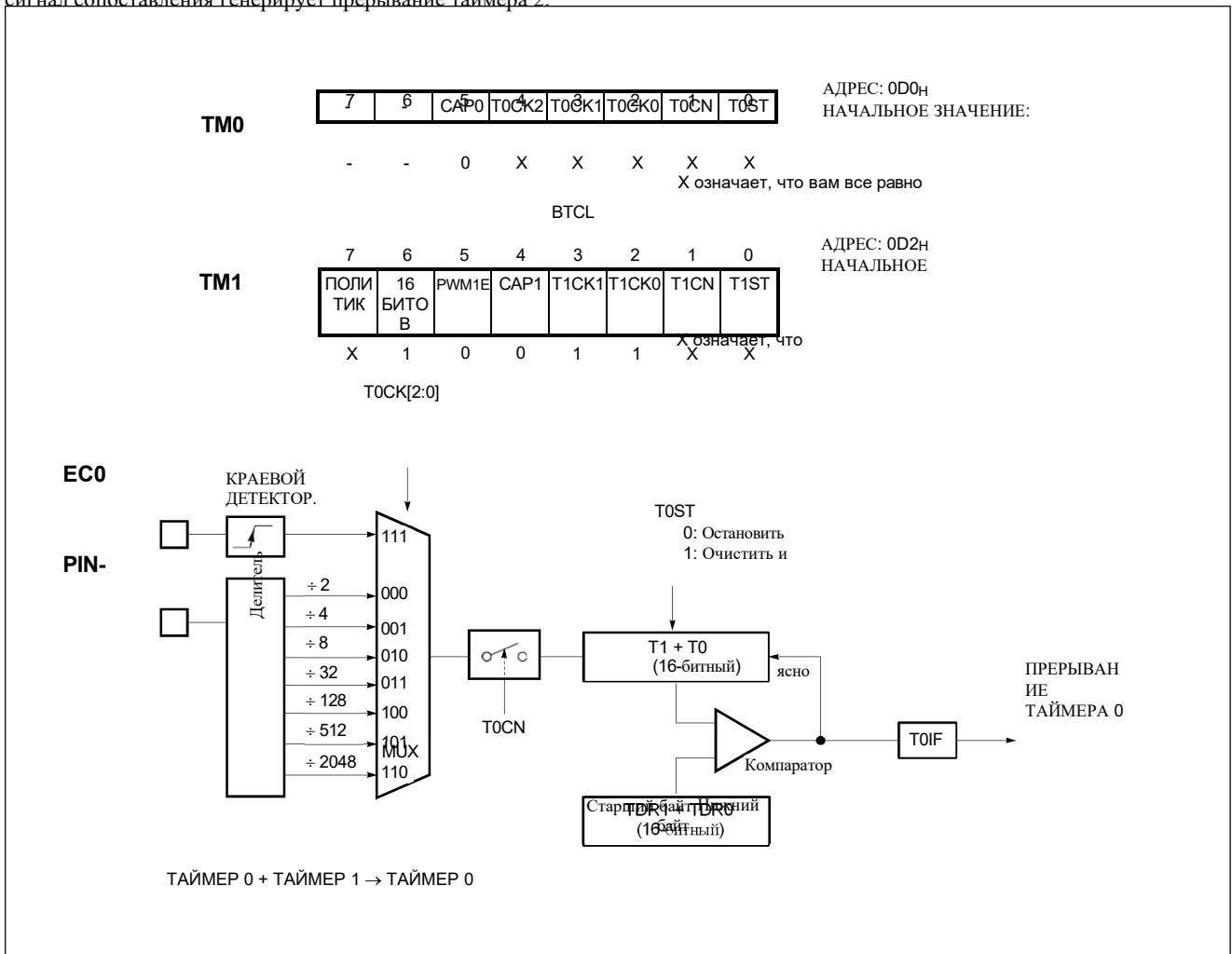


Рис. 13-9 16-битный таймер/счетчик для таймера 0, 1

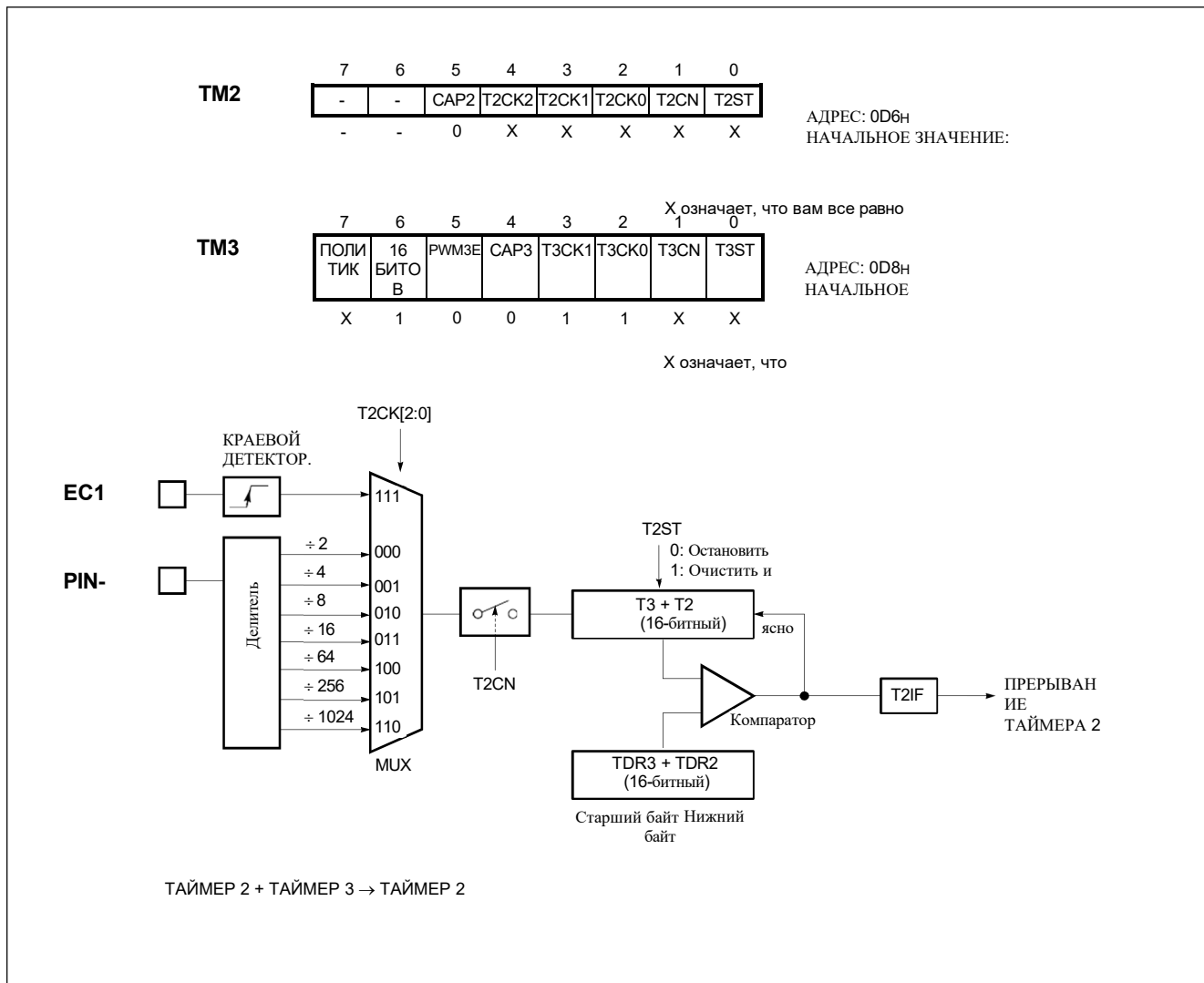


Рис. 13-10 16-битный таймер/счетчик для таймера 2, 3

### 13.3 8-разрядный выход сравнения (16-разрядный)

В MC80F0704/0708/0804/0808 есть функция таймера Compare Output. Чтобы выдать импульс, совпадение таймера может быть передано на контакт порта (T0O или T2O), как показано на Рис. 13-3 или Рис. 13-4. Таким образом, выходной импульс генерируется посредством согласования таймера. Эта операция выполняется для закрепления, R05/AN5//T0O или R06/AN6/T2O.

В этом режиме для бита T0OE или бита T2OE регистра выбора порта (PSR1.0 или PSR1.1) должно быть установлено значение «1». Этот вывод выводит сиг-

таймера (битовый CAP3 регистра режима таймера TM3 для таймера 3), как показано на фиг.13-12.

Регистр таймера/счетчика увеличивается в ответ на внутренний или

### 13.4 8-битный режим захвата

Режим захвата таймера 0 устанавливается битовым CAP0 регистра режима таймера TM0 (битовый CAP1 регистра режима таймера TM1 для таймера 1), как показано на рис. 13-11. Аналогично, режим захвата таймера 2 устанавливается битовым CAP2 регистра TM2 режима

pal, имеющий квадратную волну 50:50, и выходная частота такая же, как в приведенном ниже уравнении.

$$f = \frac{AKKO}{2 \times \text{Значение прескалера} \times (TDR + 1)}$$

внешний вход. Эта функция подсчета аналогична обычному режиму таймера, и прерывание таймера генерируется, когда регистр таймера T0 (T1, T2, T3) увеличивается и соответствует TDR0 (TDR1, TDR2, TDR3).

Это прерывание таймера в режиме захвата очень полезно, когда длительность импульса захваченного сигнала больше максимального периода

Таймера.

Например, в длительность Рисунок 13-14 импульса захваченного сигнала превышает значение данных таймера (FF<sub>n</sub>) более чем в 2 раза. Когда происходит внешнее прерывание, захваченное значение (13<sub>n</sub>) меньше желаемого значения. Это значение можно получить путем подсчета количества случаев переполнения таймера.

Таймер/Счетчик по-прежнему выполняет вышеуказанное, но с добавленной особенностью, что переход края на внешнем входе INT<sub>x</sub> приводит к тому, что текущее значение в регистре таймера x (T0, T1, T2, T3) регистрируется в регистрах CDR<sub>x</sub> (CDR0, CDR1, CDR2, CDR3) соответственно. AF -

тер захвачен, таймер x регистр очищается и перезапускается аппаратными средствами. Он имеет три режима перехода: «падающий фронт», «передний фронт», «оба фронта», которые выбираются регистром выбора края прерывания IEDS. См. раздел «17.4 Внешнее прерывание» на стр. 85. Кроме того, переход на контакте INT<sub>n</sub> генерирует прерывание.

---

**Примечание.** CDR<sub>n</sub> и TDR<sub>n</sub> имеют один и тот же адрес. В режиме захвата операция считывания считывается с CDR<sub>n</sub>, а не TDR<sub>n</sub>, потому что путь открыт к CDR<sub>n</sub>.

---



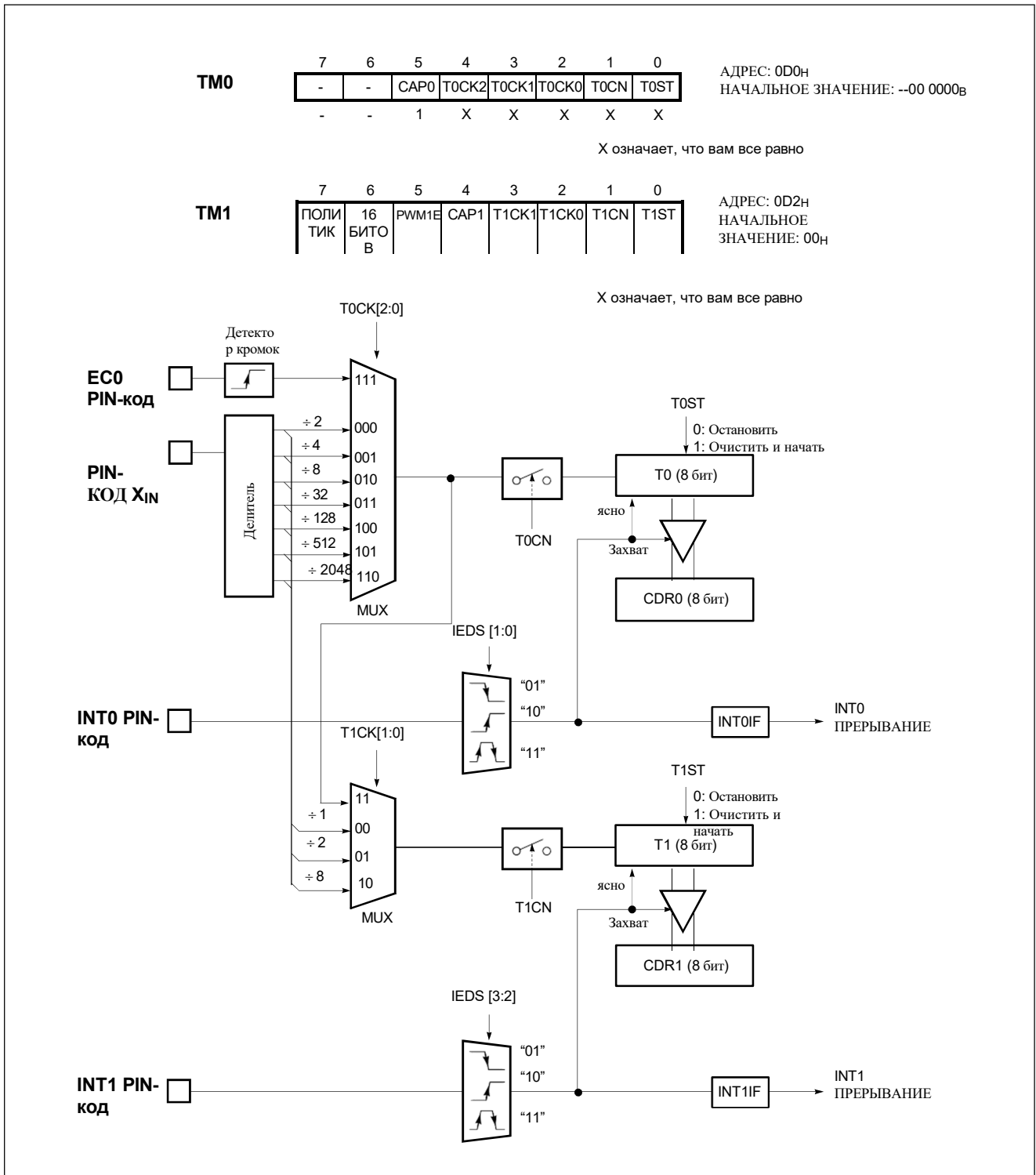


Рис. 13-11 8-битный режим захвата для таймера 0, 1

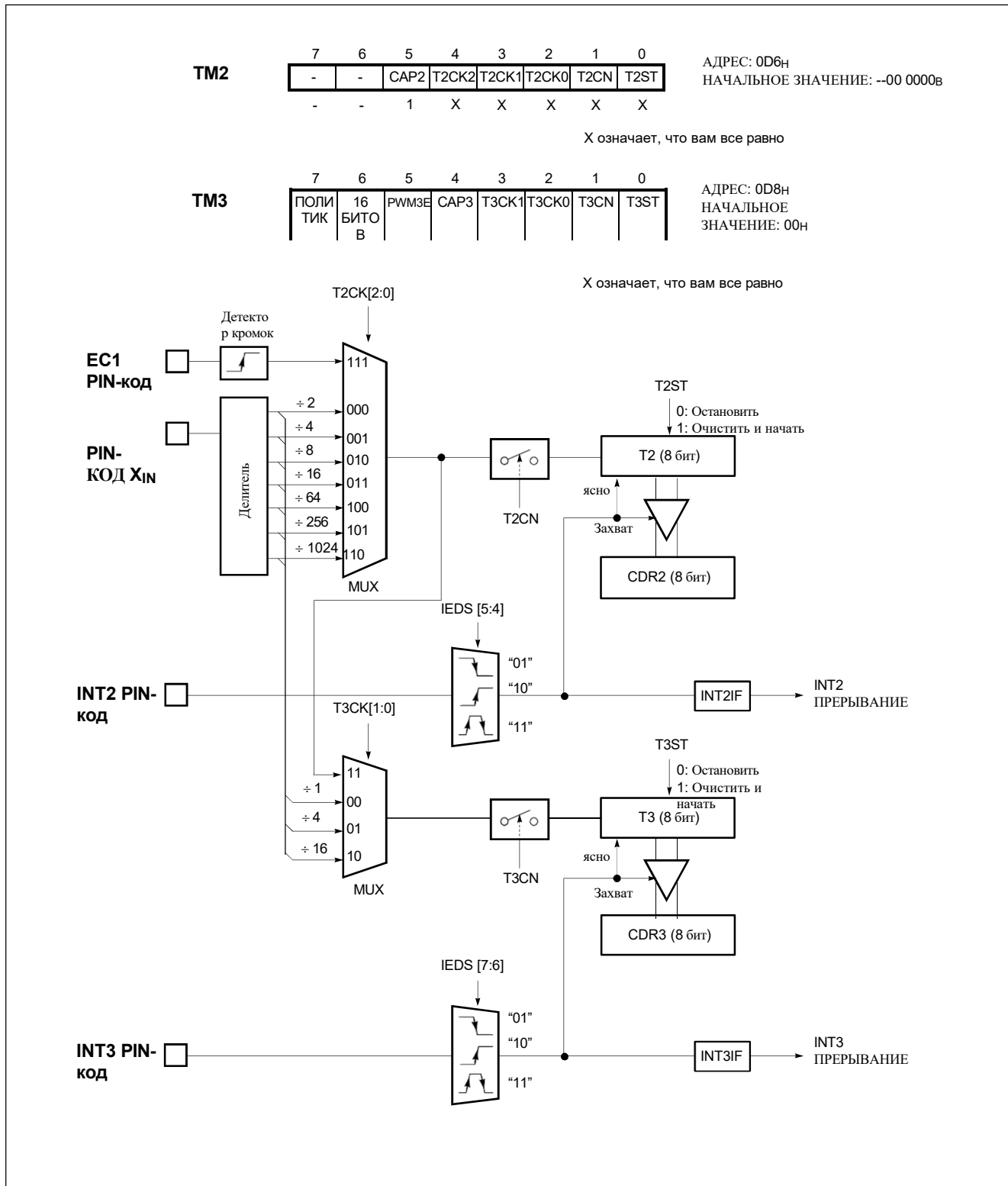
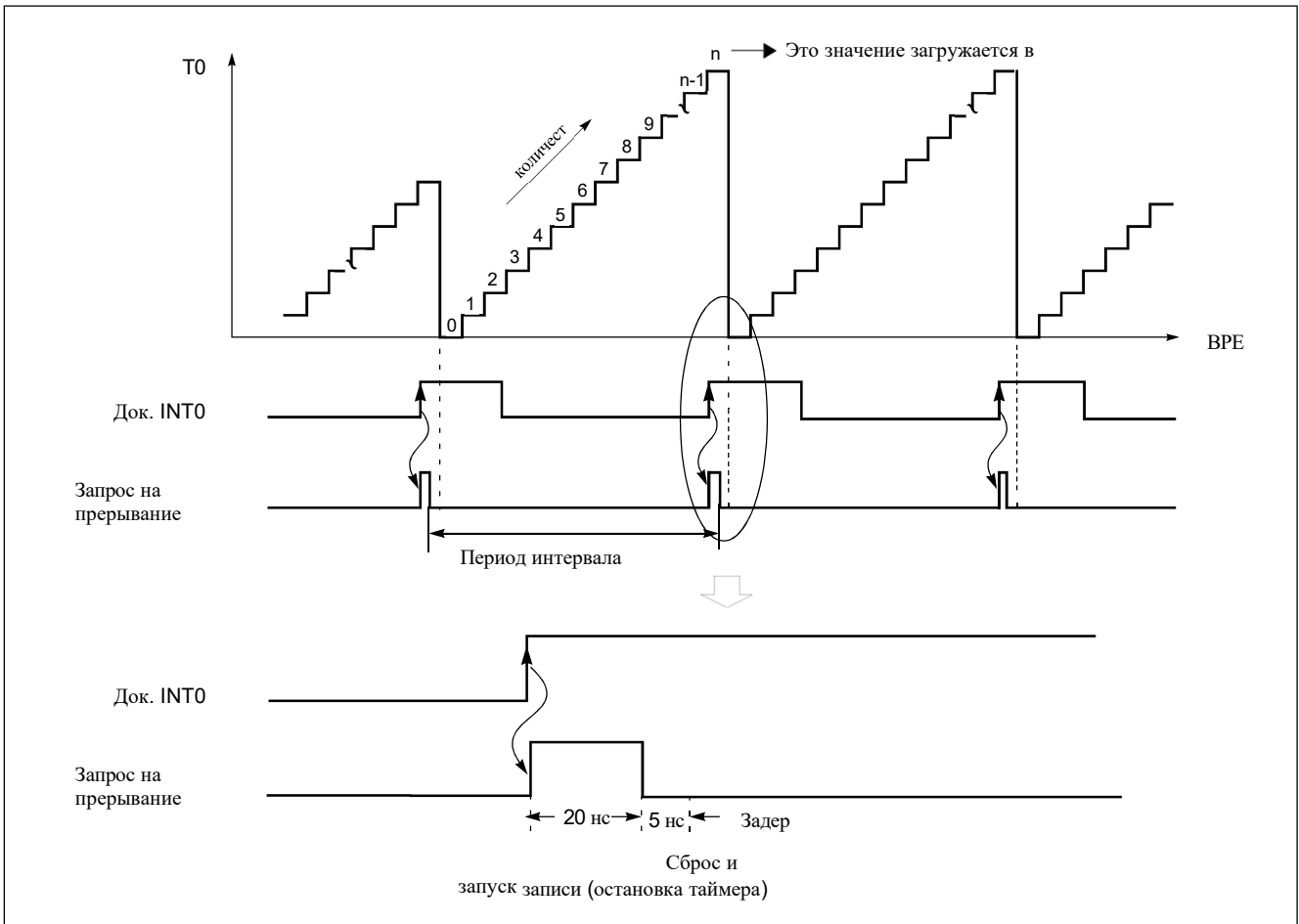
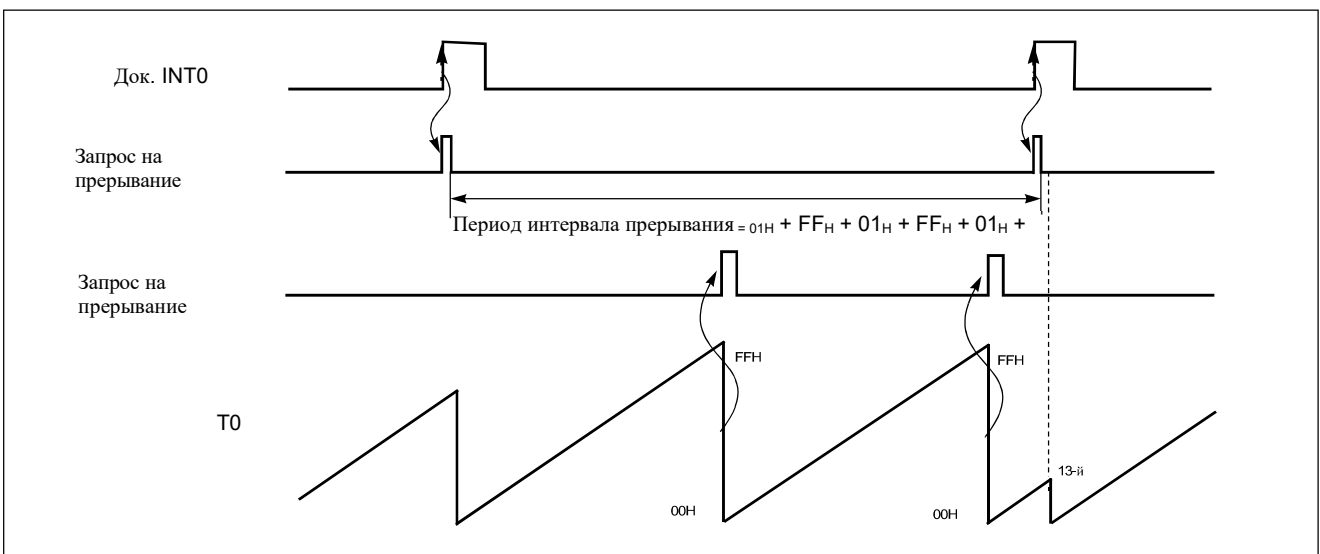


Рис. 13-12 8-битный режим захвата для таймера 2, 3



**Рис. 13-13 Режим захвата входных данных таймера 0**



**Рис. 13-14 Переполнение избыточного таймера в режиме захвата**

### 13.5 16-битный режим захвата

16-битный режим захвата такой же, как и 8-битный, за исключением того, что выполняется регистр таймера, будет 16 бит. Источник синхросигнала 0 таймера выбирается либо внутренним, либо внешним синхросигналом по биту T0СК [2:0]. В 16-битовом режиме биты T1СК1, T1СК0, CAP1 и 16BIT TM1 должны быть установлены в «1» соответственно, как показано на фиг.13-15.

Источник синхросигнала 2 таймера выбирается либо внутренним, либо внешним синхросигналом по биту T2СК [2:0]. В 16-битовом режиме биты T3СК1, T3СК0, CAP3 и 16BIT TM3 должны быть установлены в «1» повторно, как показано на рис. 13-16.

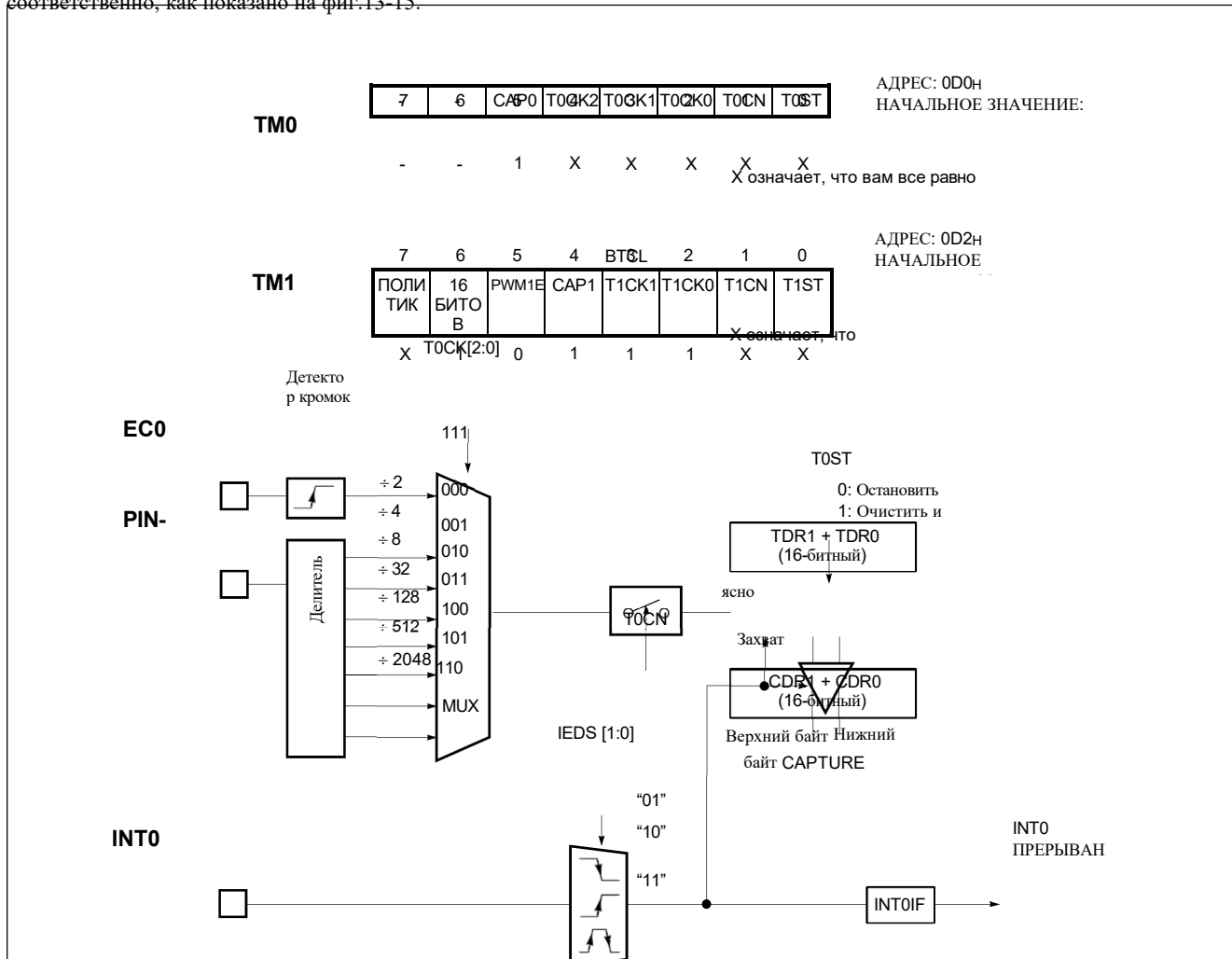


Рис. 13-15 16-битный режим захвата таймера 0, 1

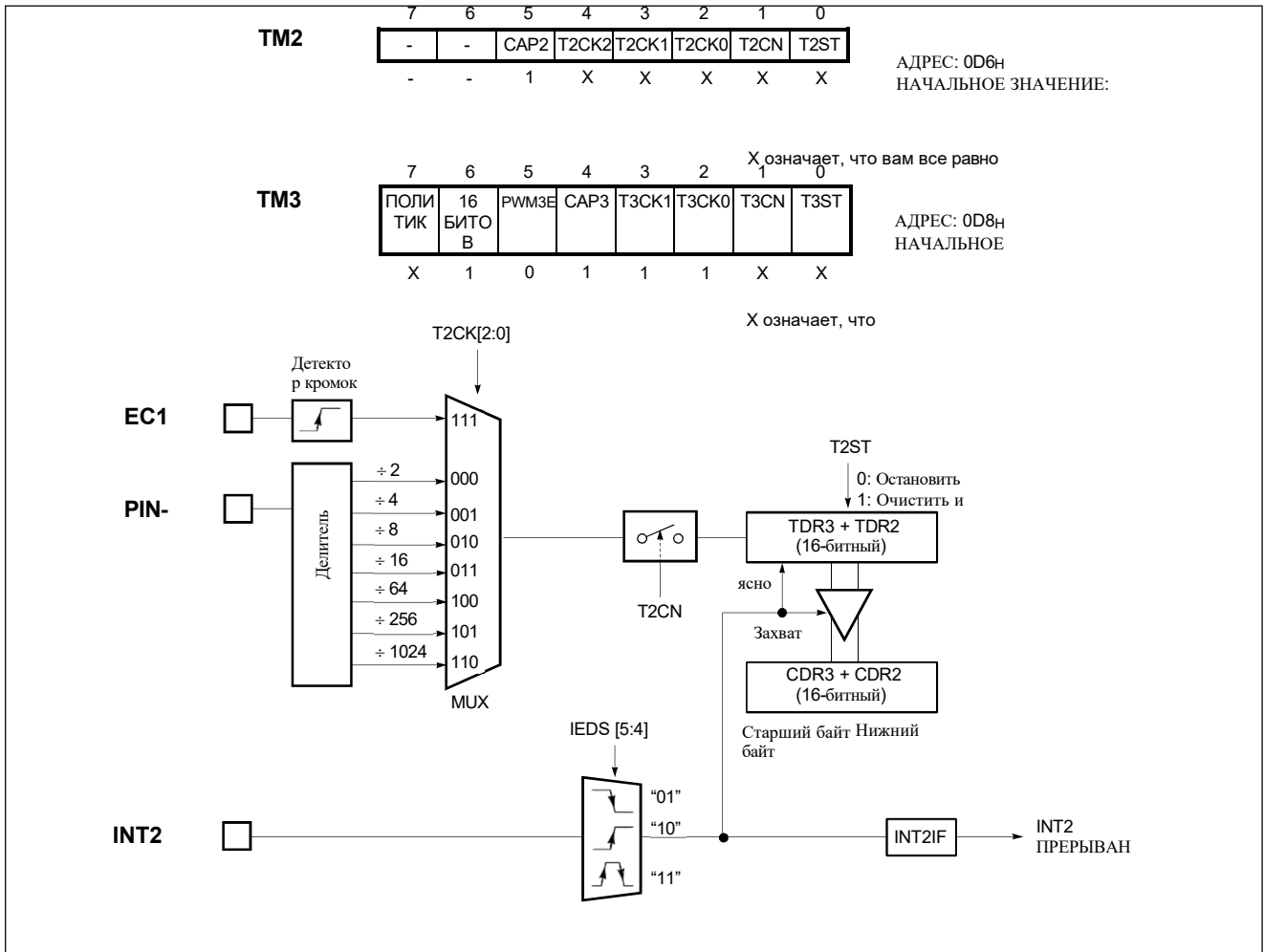


Рис. 13-16 16-битный режим захвата таймера 2, 3

**Пример 1:**

Timer0 = 16-битный режим таймера, 0,5 с при 4MHz

```
LDM  TM0, # 0000 _ 1111B; 8uS
LDM  TM1, # 0100 _ 1100B; 16-
битный режим LDM  TDR0, # <
62499 ; 8uS X 62500 LDM
TDR1, # > 62499; = 0 .5s
SET1 TOE
EI
:
:
```

**Пример 2:**

Timer0 = 16-битный режим счетчика событий

```
LDM  PSR0, # 0001 _ 0000B; EC0
Set
LDM  TM0, # 0001 _ 1111B; режим
счетчика LDM  TM1, # 0100 _
1100B; 16-битный режим LDM
TDR0, # < 0FFH ;
LDM  TDR1, # > 0FFH ;
SET1 TOE
EI
:
```

**Пример 3:**

:

Timer0 = 16-битный режим захвата

```
LDM   PSR0, # 0000 _ 0001B; INTO
аппарат
LDM   TM0, # 0010 _ 1111B; Режим
захвата LDM   TM1, # 0100
_ 1100B; 16-битный режим LDM
   TDRO, # < 0FFH ;
LDM   TDR1, # > 0FFH ;
LDM   IEDS, # 01H; Falling
Edge SET1 T0E
EI
:
:
```

### 13.6 Режим PWM

В MC80F0704/0708/0804/0808 предусмотрены функции высокоскоростной ШИМ (широтно-импульсной модуляции), совместно используемые с Timer1 или Timer3.

В режиме PWM вывод R10/ PWM1O или R11/ PWM3O до выхода PWM с разрешением 10 бит. Эти контакты должны быть сконфигурированы как выходной сигнал PWM путем установки бита «1» PWM1OE и PWM3OE в PSR0 регистре.

Период выхода PWM1 определяется T1PPR (Регистр периода T1 PWM) и T1PWHR [3:2] (bit3,2 Высокого регистра T1 PWM), а режим выхода PWM определяется T1PDR (Регистр пошлыны T1 PWM) и T3PWHR [1:0] (bit1,0 Высокого регистра T1 PWM).

Период вывода PWM3 определяется T3PPR (Регистр периода T3 PWM) и T3PWHR [3:2] (bit3,2 Высокого регистра T3 PWM), а режим вывода PWM определяется T3PDR (Регистр пошлыны T3 PWM) и T3PWHR [1:0] (bit1,0 Высокого регистра T3 PWM).

Пользователь записывает меньшее 8-битовое значение периода в T1 (3) PPR, а более высокое 2-битовое значение периода в T1 (3) PWHR [3:2]. И записывает значение нагрузки в T1 (3) PDR и T1 (3) PWHR [1:0] одинаково.

T1 (3) PDR конфигурируется как двойная буферизация для безбликового выхода PWM. На фиг.13-18 рабочие данные передаются от ведущего устройства ведомому устройству, когда данные периода соответствуют отчитанному значению. (т.е. в начале следующего рабочего цикла)

**PWM1 (3) Период = [PWM1 (3) HR [3:2] T (2) ZPPR] X  
Тактовый генератор источника**

**PWM1 (3) Дежурство = [PWM1 (3) HR [1:0] T3PDR] X  
Источник**

**Часы**

Отношение частоты и разрешения находится в обратной пропорции. Таблица 13-3 показывает отношение частоты ШИМ к разрешению.

Если требуется более высокая частота ШИМ, она должна быть уменьшена

разрешение.

Разолюция	Частота		
	T1CK[1:0] = 00 (250 нс)	T1CK[1:0] = 01 (500 нс)	T1CK[1:0] = 10 (2uS)
10-битный	3.9 кГц	0.98 кГц	0.49 кГц
9-битный	7.8 кГц	1.95 кГц	0.97 кГц
8-битный	15.6 кГц	3.90 кГц	1.95 кГц
7-битный	31.2 кГц	7.81 кГц	3.90 кГц

**Таблица 13-3 Частота PWM в сравнении с разрешением при 4MHz**

Бит POL TM1 или TM3 определяет полярность рабочего цикла.

Если рабочее значение равно значению периода, выходной сигнал ШИМ определяется битом POL (1: High, 0: Low). И если рабочее значение установлено в «00H», выходной сигнал ШИМ определяется битом POL (1: Low, 0: High).

Это значение может быть изменено при выводе ШИМ. Однако измененное значение нагрузки выводится после завершения текущего периода. Кроме того, при изменении значения периода, показанного как, может поддерживаться значение нагрузки на текущем выходе Рис. 13-20. Как бы то ни было, абсолютное время работы не изменяется с различной частотой. Но измененное значение периода должно быть больше, чем значение пошлыны.

*Примечание. При изменении Timer1 на функцию PWM сначала следует остановить тактовый сигнал таймера, а затем установить период и значение регистра заполнения. Если пользователь записывает значения регистров во время работы таймера, эти регистры могут быть установлены с определенными значениями.*

*Пример программы @ 4 МГц 2uS*

*LDM TM1, # 1010 \_ 1000b; Установить часы и PWM3E  
LDM T1PPR, # 199 ; Период: 400uS = 2uSX (199  
+ 1) LDM T1PDR, # 99 ; Дежурство: 200uS = 2uSX  
(99 + 1) LDM PWM1HR, 00H  
LDM TM1, # 1010 \_ 1011b; Время запуска 1*

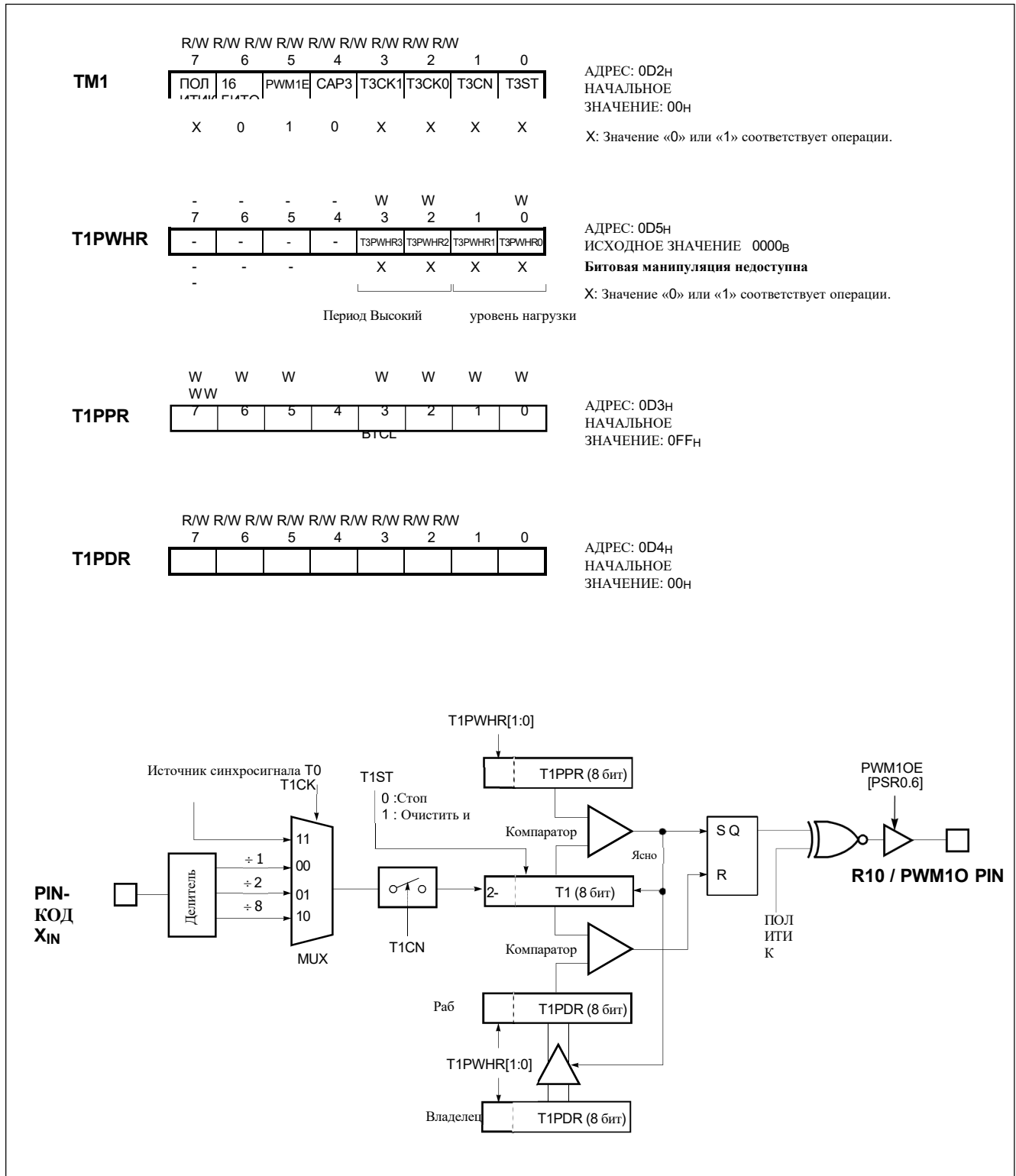
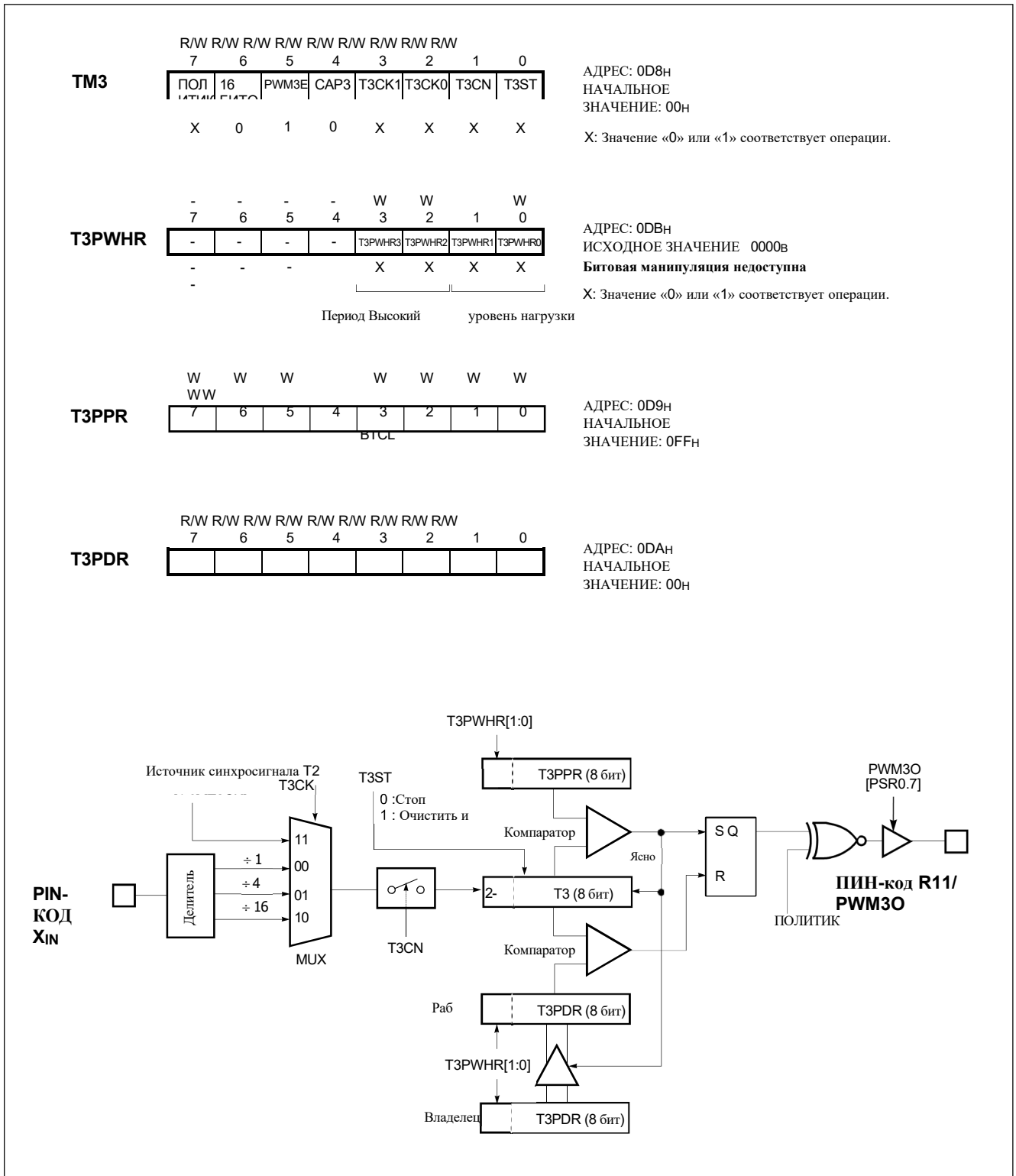


Рис. 13-17 Режим PWM1





**Рис. 13-18 Режим PWM3**

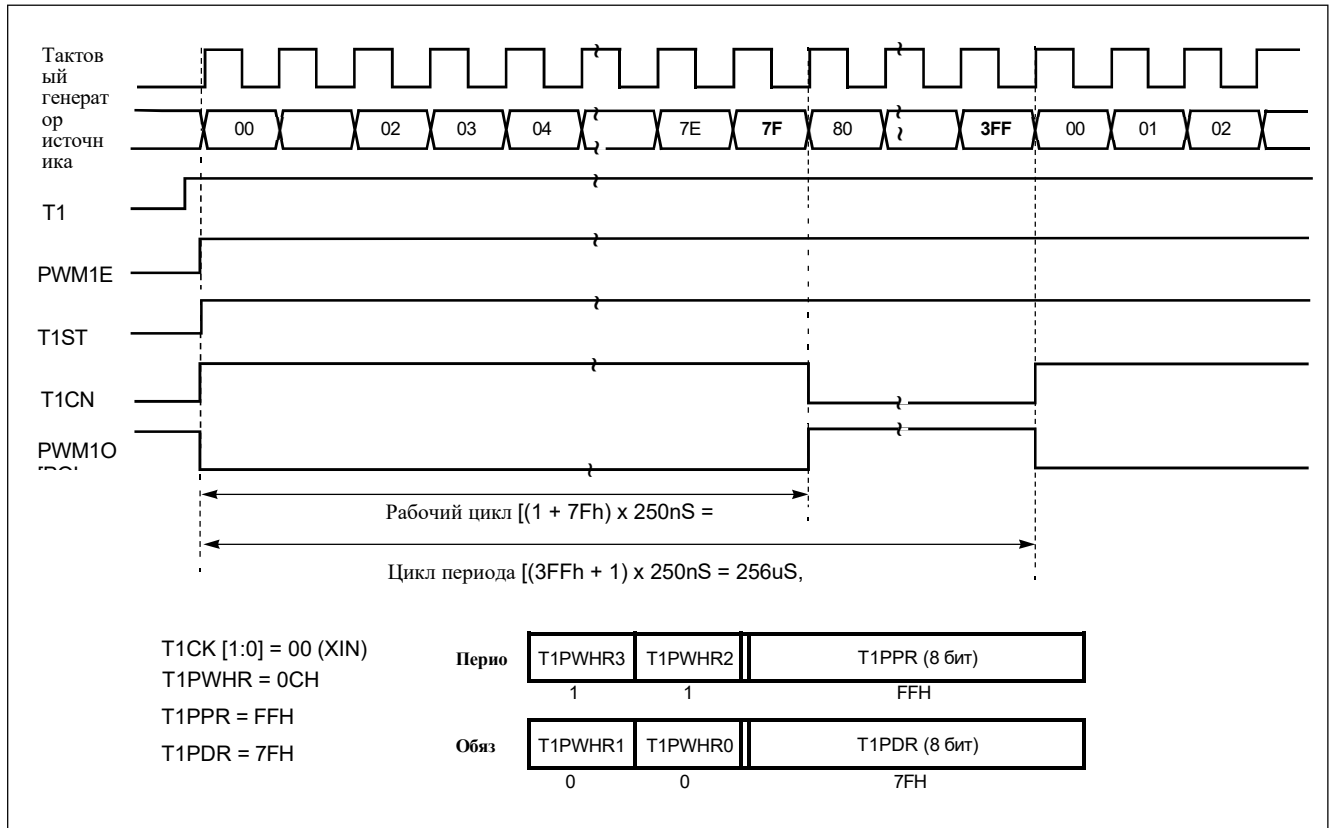


Рис. 13-19 Пример PWM1 при 4MHz

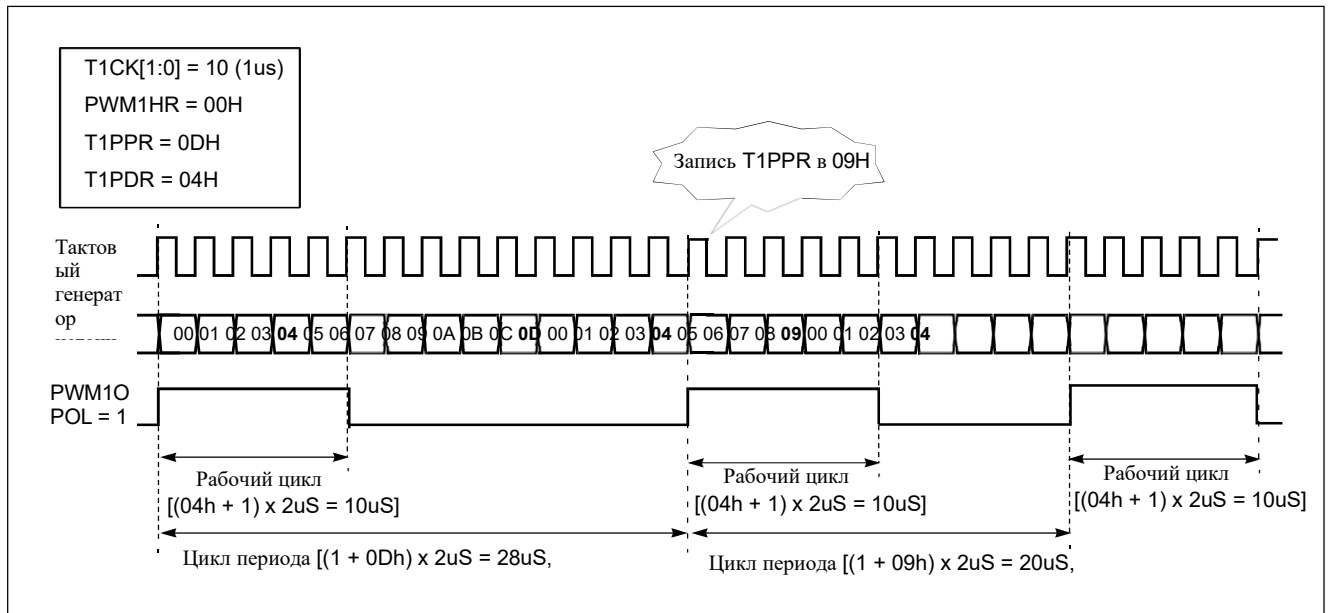


Рис. 13-20 Пример изменения периода PWM1 в абсолютном рабочем цикле (@ 4 МГц)

## 14. АНАЛОГО-ЦИФРОВОЙ ПРЕОБРАЗОВАТЕЛЬ.

Аналого-цифровой преобразователь (АЦП) позволяет преобразовывать аналоговый входной сигнал в соответствующее 10-битовое цифровое значение. АЦП имеет шестнадцать аналоговых входов, которые мультиплексируются в одну выборку и удерживаются. Выход выборки и удержания является входом в преобразователь, который генерирует результат посредством последовательного приближения.

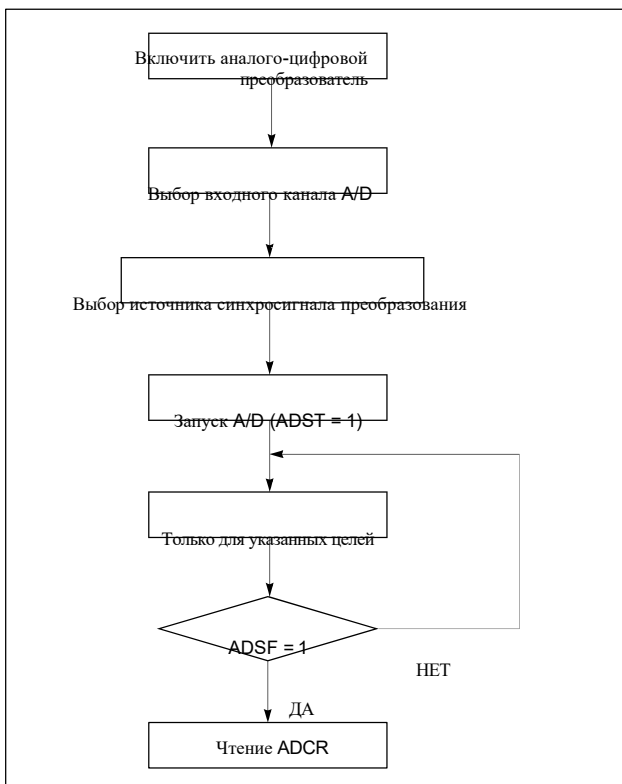
Аналоговое опорное напряжение выбирается для  $V_{DD}$  или  $AV_{ref}$  путем установки бита  $AVREFS$  в  $PSR1$  регистре. Если выбрана внешняя аналоговая ссылка  $AV_{ref}$ , то для использования не следует выбирать аналоговый входной канал 0 ( $AN0$ ). Поскольку этот контакт используется для аналогового опорного сигнала аналого-цифрового преобразователя.

АЦП имеет три регистра - регистр управления  $ADCM$  и регистр  $A/D$  результата  $ADCRH$  и  $ADCRL$ .  $AD- CRH$  [7:6] также используется в качестве битов выбора источника тактовых сигналов  $ADC$ . Регистр  $ADCM$ , показанный на, Рис. 14-4 управляет работой модуля аналого-цифрового преобразователя. Контакты порта могут быть сконфигурированы как аналоговые входы или цифровые входы/выходы.

Он выбирается для преобразования соответствующего канала путем установки  $ADS$  [3:0].  $A/D$ -порт устанавливается  $ADEN$  и  $ADS$  [3:0] в аналоговый входной порт независимо от регистра направления ввода/вывода порта. Порт, не выбранный  $ADS$  [3:0], работает как обычный порт.

Рис. 14-1 Рабочий поток АЦП

**Использование аналого-цифрового преобразователя**  
Обработка преобразования начинается, когда начальный бит  $ADST$  равен



установите значение «1». После одного цикла он очищается аппаратными средствами. Регистр ADCRH и ADCRL содержит результаты аналого-цифрового преобразования. Когда преобразование завершено, результат загружается в ADCRH и ADCRL, бит состояния A/D преобразования ADSF устанавливается в «1», и устанавливается флаг A/D прерывания ADCIF. Рис. 14-1 Для получения информации о потоке операций см. раздел.

Блок-схема модуля A/D показана на рис. 14-3. Бит АЦП состояния устанавливается автоматически при завершении АЦП и сбрасывается при выполнении АЦП. Время преобразования занимает 13 раз тактового сигнала источника преобразования. Синхросигнал источника преобразования должен выбираться для времени преобразования, превышающего 25μs.

### Меры предосторожности для аналого-цифрового преобразователя

#### (1) Диапазон входных значений AN0 ~ AN15

Входное напряжение входов АЦП должно находиться в пределах заданного диапазона. В частности, если на вход подается напряжение выше  $V_{DD}$  (или  $AV_{ref}$ ) или ниже  $V_{SS}$  (даже если оно находится в пределах абсолютного максимального диапазона номинальных значений), значение преобразования для этого канала не может быть индетерминированным. Также могут быть затронуты значения преобразования других каналов.

#### (2) Меры противодействия шуму

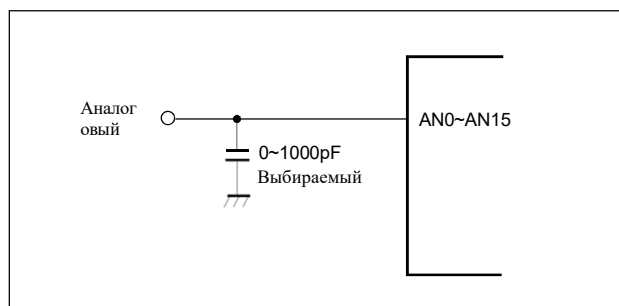
Для поддержания 10-разрядного разрешения необходимо обратить внимание на шум контактов  $V_{DD}$  (или  $AV_{ref}$ ) и контактов аналогового ввода (AN0 ~ AN15). Поскольку эффект увеличивается пропорционально выходу источника аналогового ввода, в некоторых случаях рекомендуется, чтобы конденсатор был подключен снаружи, как показано на рисунке, 14-2 для уменьшения шума. Емкость выбирается пользователем и соответствующим образом определяется в соответствии с целевой системой.

**Рис. 14-2 Аналоговый входной контакт для подключения конденсатора**

#### (3) Работа ввода-вывода

Контакты аналогового ввода AN0 ~ AN15 также функционируют как контакты порта ввода/вывода. При выполнении аналого-цифрового преобразования с любым выводом не выполняйте команду PORT input с выбранным выводом во время выполнения преобразования, поскольку это может снизить разрешение преобразования.

Кроме того, если цифровые импульсы подаются на контакт рядом с выводом в процессе аналого-цифрового преобразования, ожидаемое значение аналого-цифрового преобразования

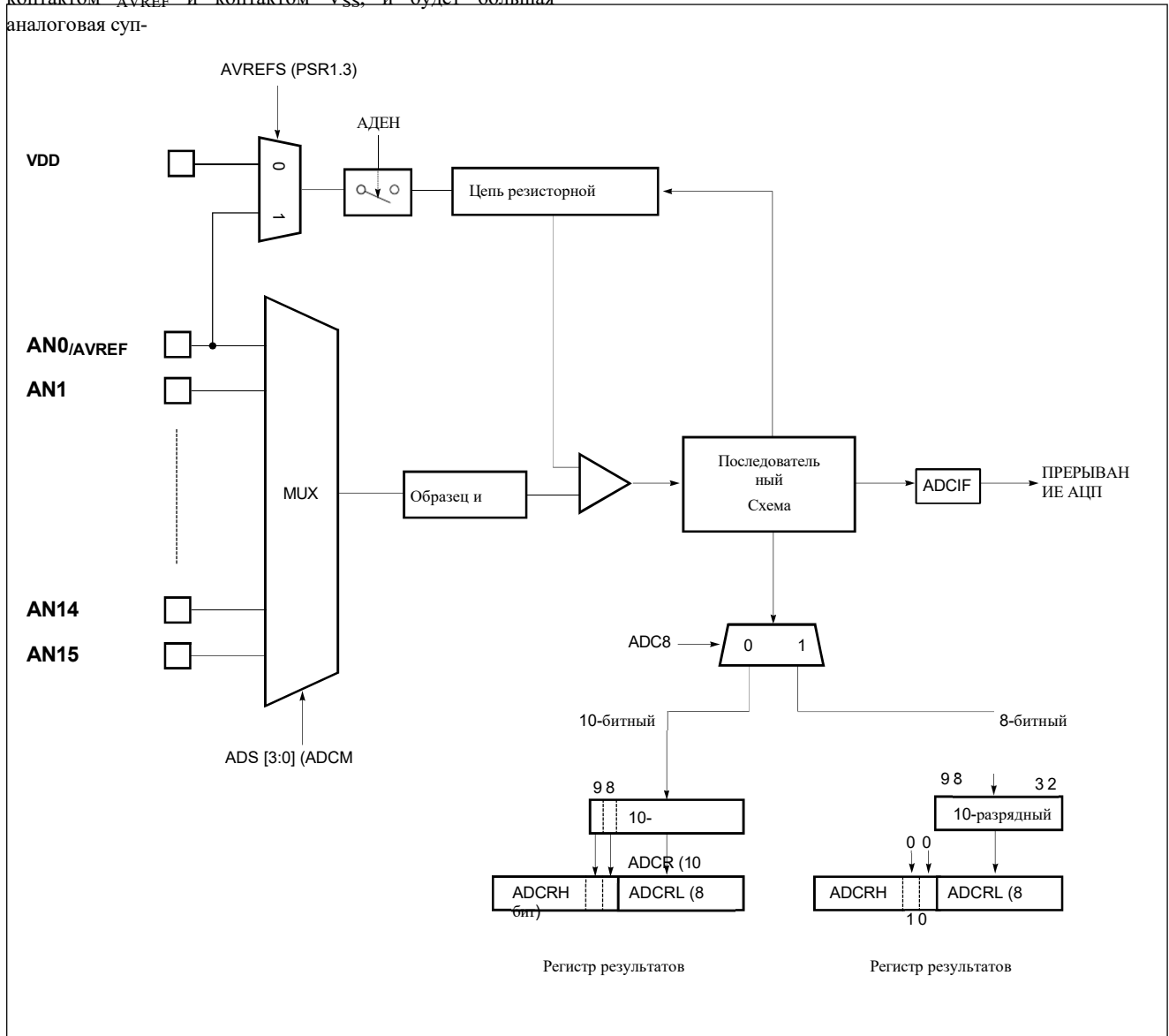


может быть недоступно из-за шума соединения. Поэтому избегайте погрешности по напряжению прикладываю импульсы к контактам, смежным с контактом, подвергающимся аналого-цифровому преобразованию.

**(4) Входной импеданс контактов AV<sub>DD</sub>**

Последовательная последовательность резисторов приблизительно 5KΩ подключается к контактам AV<sub>REF</sub> и V<sub>SS</sub>. Таким образом, если выходной импульс аналогового источника питания высокий, это приведет к параллельному подключению к последовательному резистору между контактом AV<sub>REF</sub> и контактом V<sub>SS</sub>, и будет большая аналоговая суп-

***Примечание.** Если напряжение AV<sub>REF</sub> меньше напряжения V<sub>DD</sub>, а входные контакты an-*lalog* (ANX), совместно используемые с различными альтернативными функциями, используются двунаправленными портами ввода/вывода, ток утечки может протекать через контакт V<sub>DD</sub> на контакт AV<sub>REF</sub> в режиме высокого выхода или входные контакты an-*log* (ANX) в режим высокого входа AV<sub>RE</sub>.*



**Рис. 14-3 Блок-схема A/D**



---

PS: часы  
преобразования

**Рис. 14-4** Регистр управления и результатов АЦП

### 15. ПОСЛЕДОВАТЕЛЬНЫЙ ВХОД/ВЫХОД (SIO)

Последовательный вход/выход используется для последовательной передачи/приема 8-битных данных. Модуль последовательного ввода/вывода (SIO) является последовательным интерфейсом, используемым для связи с другими периферийными устройствами микроконтроллера. Эти периферийные устройства могут быть последовательными EEPROM, сдвиговыми регистрами, драйверами дисплея, аналого-цифровыми преобразователями и т.д. Этот SIO является 8-разрядным синхронным сигналом и состоит из регистра данных последовательного ввода-вывода, регистра режима последовательного ввода-вывода, схемы выбора синхросигнала, восьмеричного счетчика и

схема управления, показанная на рис. 15-1. Контакт SO подписывается на вход и выход. Таким образом, Serial I/O (SIO) может работать как минимум с двумя контактами. Контакты R00/SCK, R01/SI и R02/SO управляются регистром последовательного режима. Содержимое регистра данных последовательного ввода-вывода может быть записано или считано программным обеспечением. Данные в регистре последовательных данных могут быть сдвинуты синхронно с тактовым сигналом передачи.

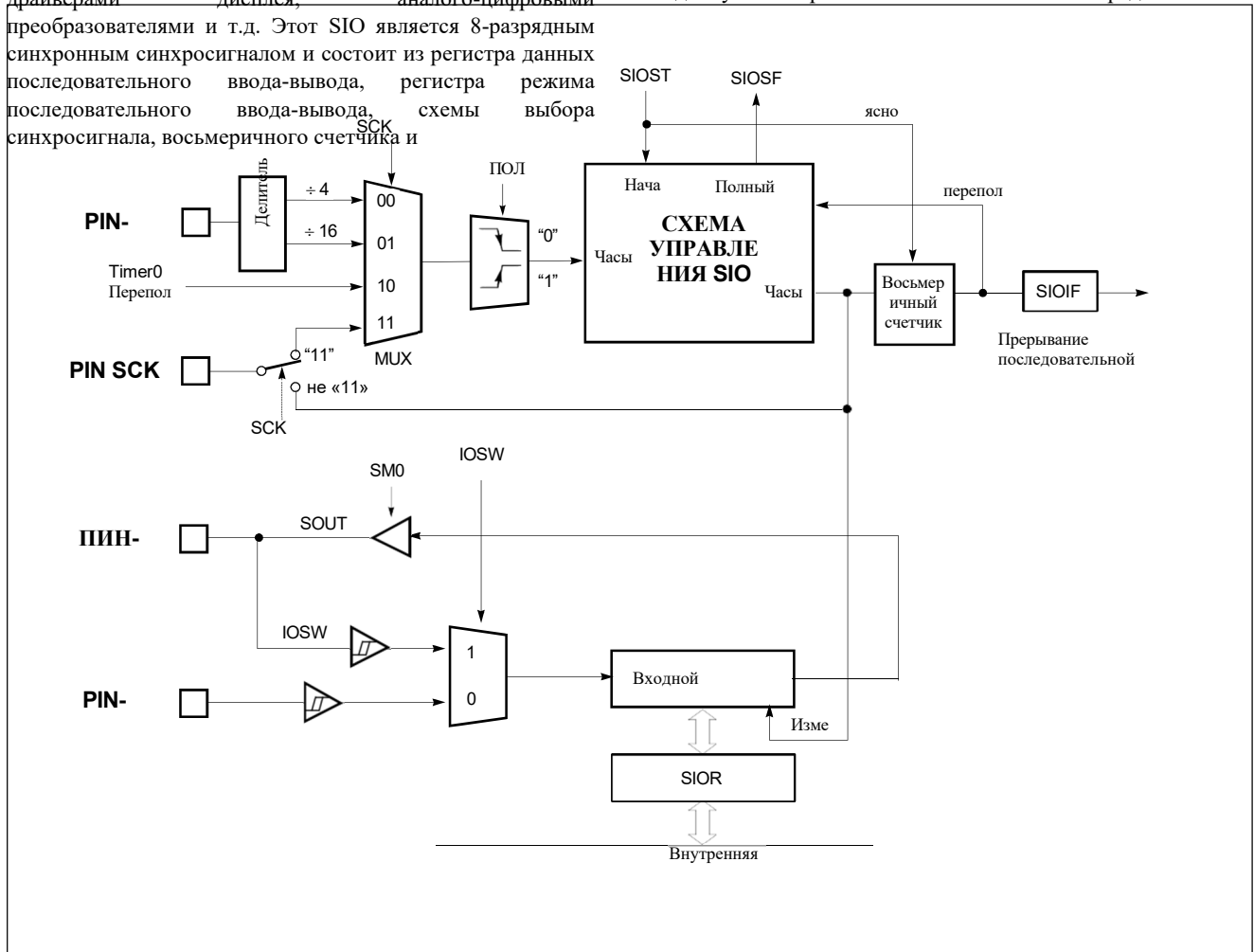


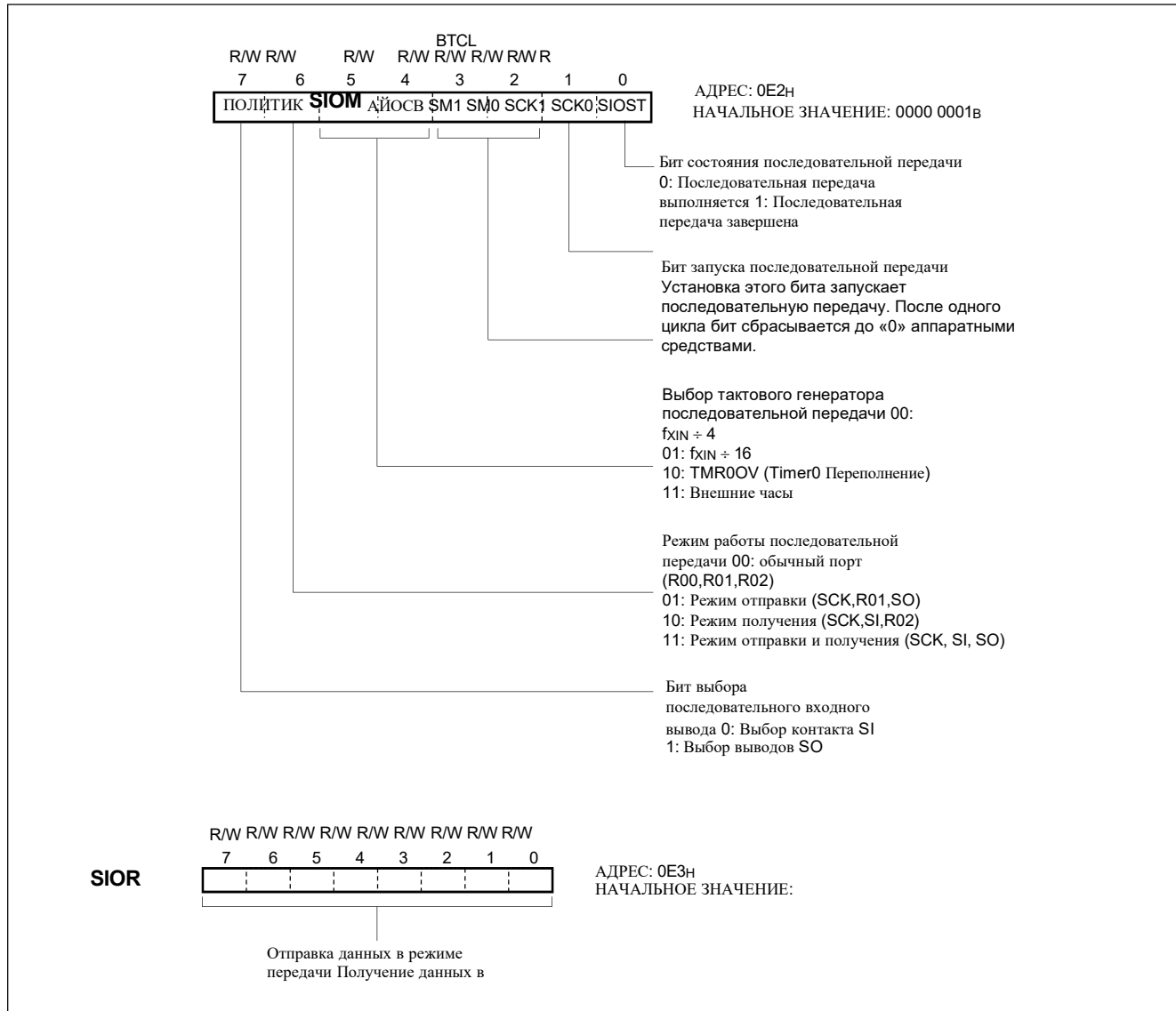
Рис. 15-1 Блок-схема SIO



Регистр режима последовательного ввода-вывода (SIOM) управляет функцией последовательного ввода-вывода. В соответствии с SCK1 и SCK0 можно выбрать внутренний или внешний синхросигнал.

последовательного ввода-вывода (SIOR) представляет собой 8-битный сдвиговый регистр. Первый LSB посылается или принимается первым.

Регистр данных



**Рис. 15-2 Регистр управления SIO**

### 15.1 Синхронизация передачи/приема

Последовательная передача запускается путем установки параметра SIOST (bit1 для SIOM) в значение «1». После одного цикла SCK, SIOST и SIOSF (бит 0 SI-OM) автоматически сбрасывается на «0». В состоянии сброса бита POL по умолчанию последовательные выходные данные из 8-разрядного сдвигового регистра выводятся на заднем крае SCLK, а входные данные фиксируются на переднем крае

контакта SCLK (см.). Рис. 15-3 Когда тактовый сигнал передачи подсчитывается 8 раз, счетчик последовательного ввода-вывода сбрасывается как «0». Тактовый сигнал передачи останавливается в состоянии «H» и выполняется последовательное прерывание ввода-вывода (SIOIF). Для SIOSF автоматически устанавливается значение «1».

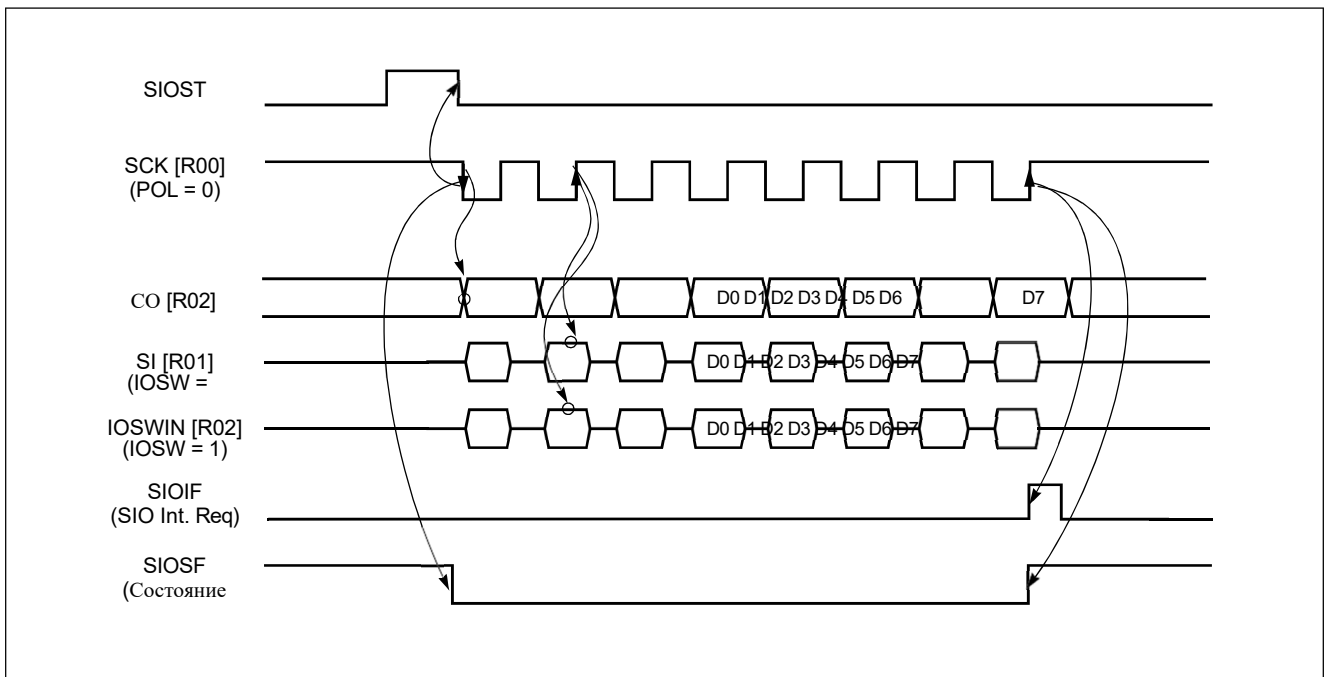


Рис. 15-3 Схема синхронизации последовательного ввода-вывода в POL = 0

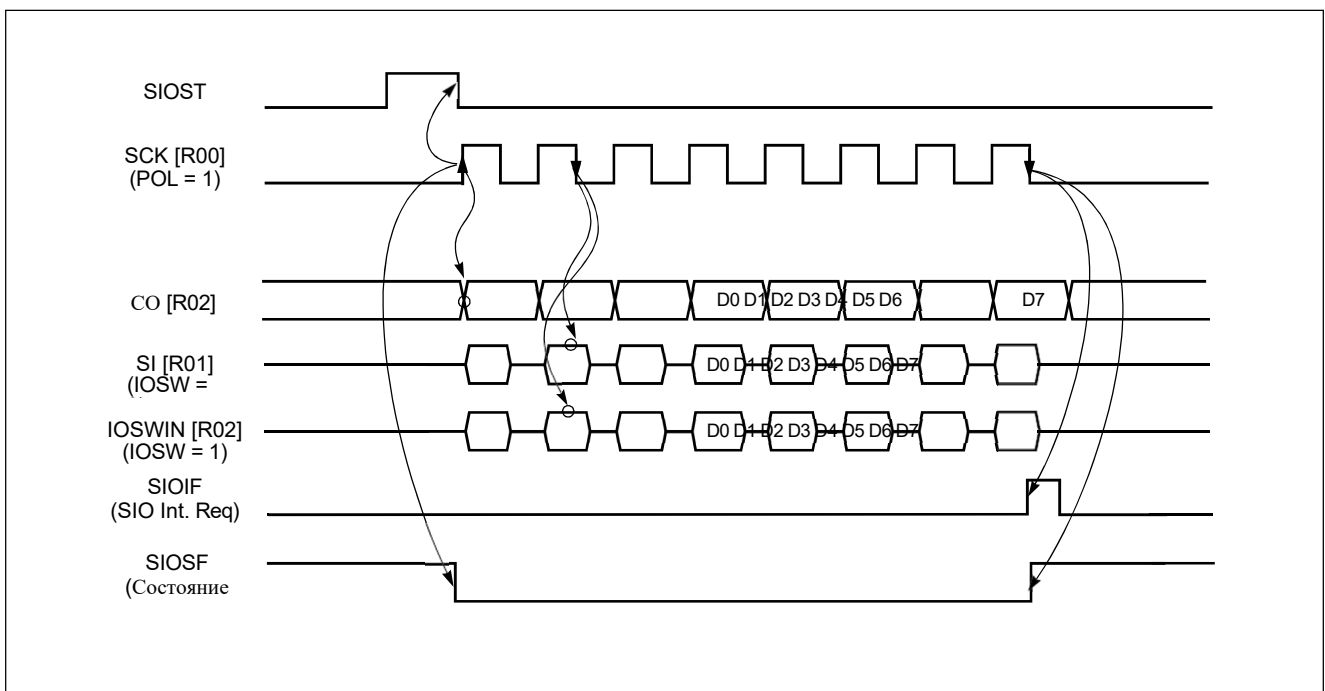


Рис. 15-4 Схема синхронизации последовательного ввода-вывода в POL = 1

## 15.2 Использование последовательного ввода-вывода

1. Выберите режим передачи/приема.
2. В режиме передачи запишите данные, подлежащие отправке в SIOR.
3. Установите SIOST в значение «1» для запуска последовательной передачи.
4. Прерывание SIO генерируется по завершении SIO, и SIOIF устанавливается в «1».
5. В случае режима приема полученные данные получают считыванием SIOR.
6. При использовании метода опроса завершение 1-байтовой последовательной связи можно проверить путем считывания SIOST и SIOSF. Как показано в примере кода, ожидание до изменения SIOST на «0», а затем ожидание изменения SIOSF на «1» для проверки завершения.

```

LDM SIOR, #          ; установка
LDM SIOM, # 0011 _ 1100b; установка
NOP  режима SIO
LDM
Тол
SIO WAIT:
НП
ББС SIOST,SIO_WAIT;wait первое ребро
Би-би-SIOSF,SIO_WAIT;wait завершено
    
```

---

**Примечание:** При использовании внешнего синхросигнала частота должна быть меньше 1MHz, а рекомендуемая нагрузка - 50%. Если выбран оба режима передачи и передача выполняется одновременно, может возникнуть ошибка.

---

## 16. ФУНКЦИЯ BUZZER

Блок драйвера зуммера состоит из 6-разрядного двоичного счетчика, регистра зуммера BUZR и селектора источника тактовых импульсов. Он генерирует квадратичную волну, имеющую очень широкую частоту диапазона (488Hz ~ 250kHz при  $f_{XIN} = 4MHz$ ) пользовательским программным обеспечением.

На контакт R12/BUZO может выводиться 50% рабочий импульс, используемый для привода электрозуммера. Контакт R12 назначен для выходного порта Драйвер зуммера, установив бит 2 PSR1 (адрес 0F9H) в «1». PSR1 регистр приведен на рис. 16-2.

Пример: вывод 5kHz при 4MHz.

```
LDM BUZR, # 0011
0001B LDM PSR1, # XXXX
_ X1XXB
```

X означает, что вам все равно

Разряды от 0 до 5 БУЗР определяют выходную частоту зуммера

вождение.

Ниже приведено уравнение вычисления частоты.

$$f_{BUZ} = \frac{f_{XIN}}{2 \times \text{Отношение} \times (BUR + 1)}$$

$f_{BUZ}$ : Частота зуммера  $f_{XIN}$ :  
Частота генератора  
Разделите Отношение: Делитель частоты делит отношение на ДОЛЛАР [1:0] КОЛЮЧКА: Повысьте 6 битовых значений BUZR. Значение периода зуммера.

Частота выходного сигнала регулируется регистром управления зуммером BUZR. Разряды от 0 до 5 БУЗР определяют частоту вывода для возбуждения зуммера.

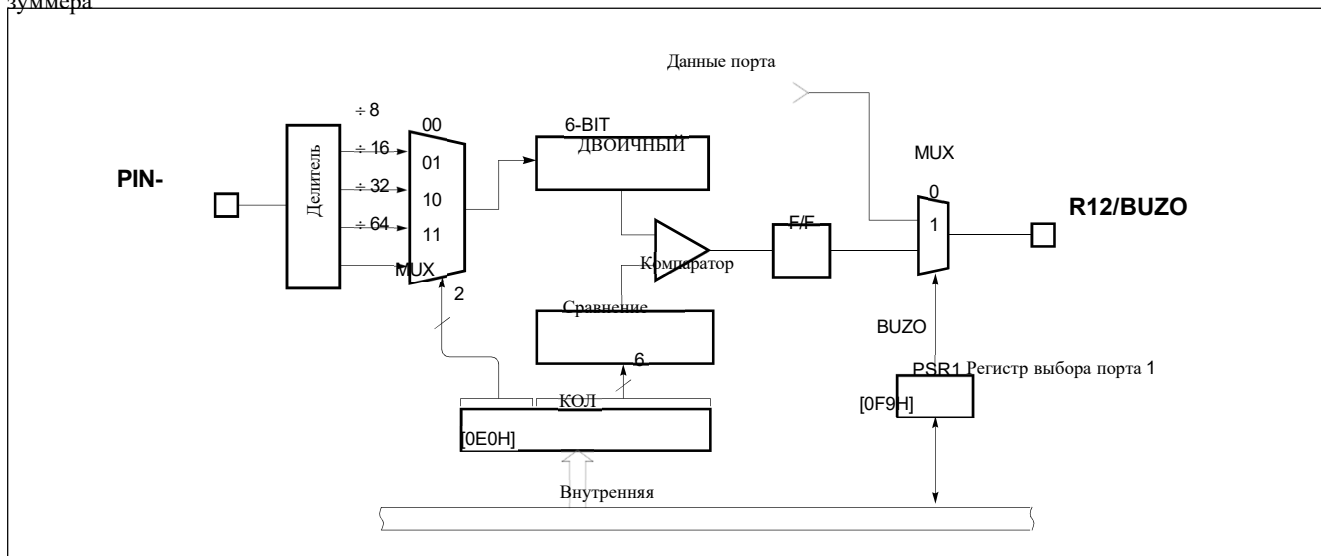


Рис. 16-1 Блок-схема драйвера зуммера

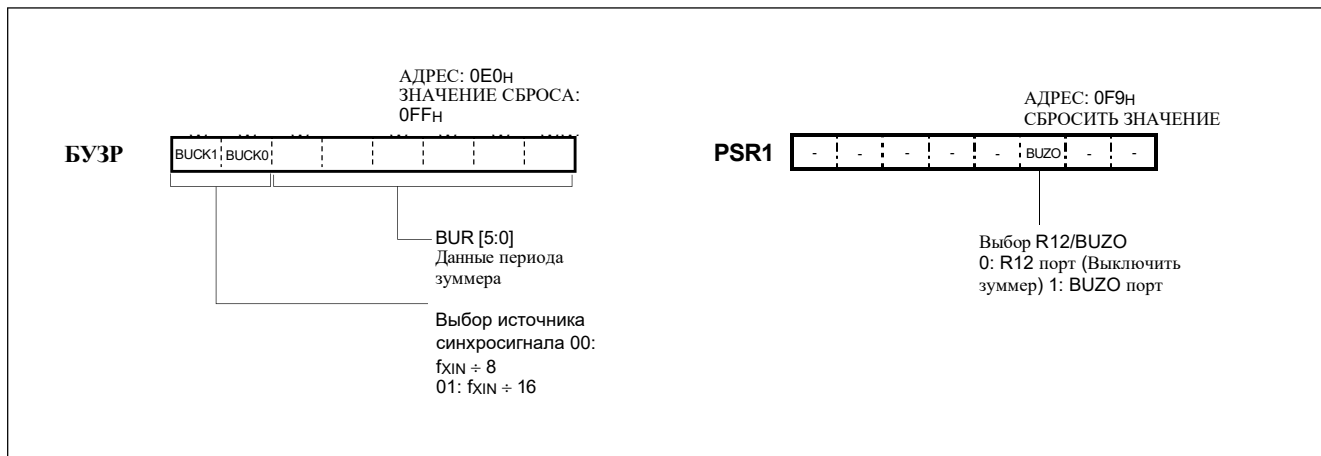


Рис. 16-2 Регистр и PSR1 зуммера

6-разрядный счетчик очищается и начинает подсчет, записывая сигнал в регистр BUZR. Он является инкрементным от 00<sub>H</sub> до тех пор, пока не будет соответствовать 6-битному значению BUR.

При 4MHz основной частоты частота зуммера показана в таблице 16-1.

BUR [5:0]	BUR [7:6]			
	00	01	10	11
00	250.000	125.000	62.500	31.250
01	125.000	62.500	31.250	15.625
02	83.333	41.667	20.833	10.417
03	62.500	31.250	15.625	7.813
04	50.000	25.000	12.500	6.250
05	41.667	20.833	10.417	5.208
06	35.714	17.857	8.929	4.464
07	31.250	15.625	7.813	3.906
08	27.778	13.889	6.944	3.472
09	25.000	12.500	6.250	3.125
0A	22.727	11.364	5.682	2.841
0B	20.833	10.417	5.208	2.604
0C	19.231	9.615	4.808	2.404
0D	17.857	8.929	4.464	2.232
0E	16.667	8.333	4.167	2.083
0F	15.625	7.813	3.906	1.953
10	14.706	7.353	3.676	1.838
11	13.889	6.944	3.472	1.736
12	13.158	6.579	3.289	1.645
13	12.500	6.250	3.125	1.563
14	11.905	5.952	2.976	1.488
15	11.364	5.682	2.841	1.420
16	10.870	5.435	2.717	1.359
17	10.417	5.208	2.604	1.302
18	10.000	5.000	2.500	1.250
19	9.615	4.808	2.404	1.202
1A	9.259	4.630	2.315	1.157
1B	8.929	4.464	2.232	1.116
1C	8.621	4.310	2.155	1.078
1D	8.333	4.167	2.083	1.042
1E	8.065	4.032	2.016	1.008
1F	7.813	3.906	1.953	0.977

BUR [5:0]	BUR [7:6]			
	00	01	10	11
20	7.576	3.788	1.894	0.947
21	7.353	3.676	1.838	0.919
22	7.143	3.571	1.786	0.893
23	6.944	3.472	1.736	0.868
24	6.757	3.378	1.689	0.845
25	6.579	3.289	1.645	0.822
26	6.410	3.205	1.603	0.801
27	6.250	3.125	1.563	0.781
28	6.098	3.049	1.524	0.762
29	5.952	2.976	1.488	0.744
2A	5.814	2.907	1.453	0.727
2B	5.682	2.841	1.420	0.710
2C	5.556	2.778	1.389	0.694
2D	5.435	2.717	1.359	0.679
2E	5.319	2.660	1.330	0.665
2F	5.208	2.604	1.302	0.651
30	5.102	2.551	1.276	0.638
31	5.000	2.500	1.250	0.625
32	4.902	2.451	1.225	0.613
33	4.808	2.404	1.202	0.601
34	4.717	2.358	1.179	0.590
35	4.630	2.315	1.157	0.579
36	4.545	2.273	1.136	0.568
37	4.464	2.232	1.116	0.558
38	4.386	2.193	1.096	0.548
39	4.310	2.155	1.078	0.539
3A	4.237	2.119	1.059	0.530
3B	4.167	2.083	1.042	0.521
3C	4.098	2.049	1.025	0.512
3D	4.032	2.016	1.008	0.504
3E	3.968	1.984	0.992	0.496
3F	3.907	1.953	0.977	0.488

Таблица 16-1 Частота зуммера (блок кГц)

## 17. ПЕРЕРЫВЫ

Каналы прерываний MC80F0704/0708/0804/0808 состоят из регистра разрешения прерываний (IENH, IENL), флагов запроса прерываний IRQH, IRQL, приоритетного канала и флага разрешения Master (флаг «I» PSW). Предусмотрено пятнадцать источников прерываний. Конфигурация канала прерывания показана в, Рисунок 17-1а приоритет прерывания показан в таблице 17-1.

Внешние прерывания (External Interrupts) INT0 ~ INT3 каждый могут быть активированы (переход 1 в 0 или 0 в 1) посредством выбора регистра IEDS.

Флаги, которые фактически генерируют эти прерывания, являются битовыми INT0IF, INT1IF, INT2IF и INT3IF в регистре IRQH. Когда генерируется внешнее прерывание, генерируемый флаг сбрасывается аппаратным обеспечением, когда подпрограмма обслуживания активируется только в том случае, если прерывание было активировано при переходе.

Прерывания таймера 0 – таймера 3 генерируются посредством T0IF, T1IF, T2IF и T3IF, которые устанавливаются посредством совпадения в соответствующем регистре таймера/счетчика.

Прерывание таймера основного интервала генерируется BITIF, которое устанавливается переполнением в регистре таймера.

Прерывание преобразователя AD генерируется ADCIF, который устанавливается путем завершения аналого-цифрового преобразования.

Сторожевой таймер генерируется WDTIF и WTIF, которые устанавливаются посредством совпадения в регистре сторожевого таймера.

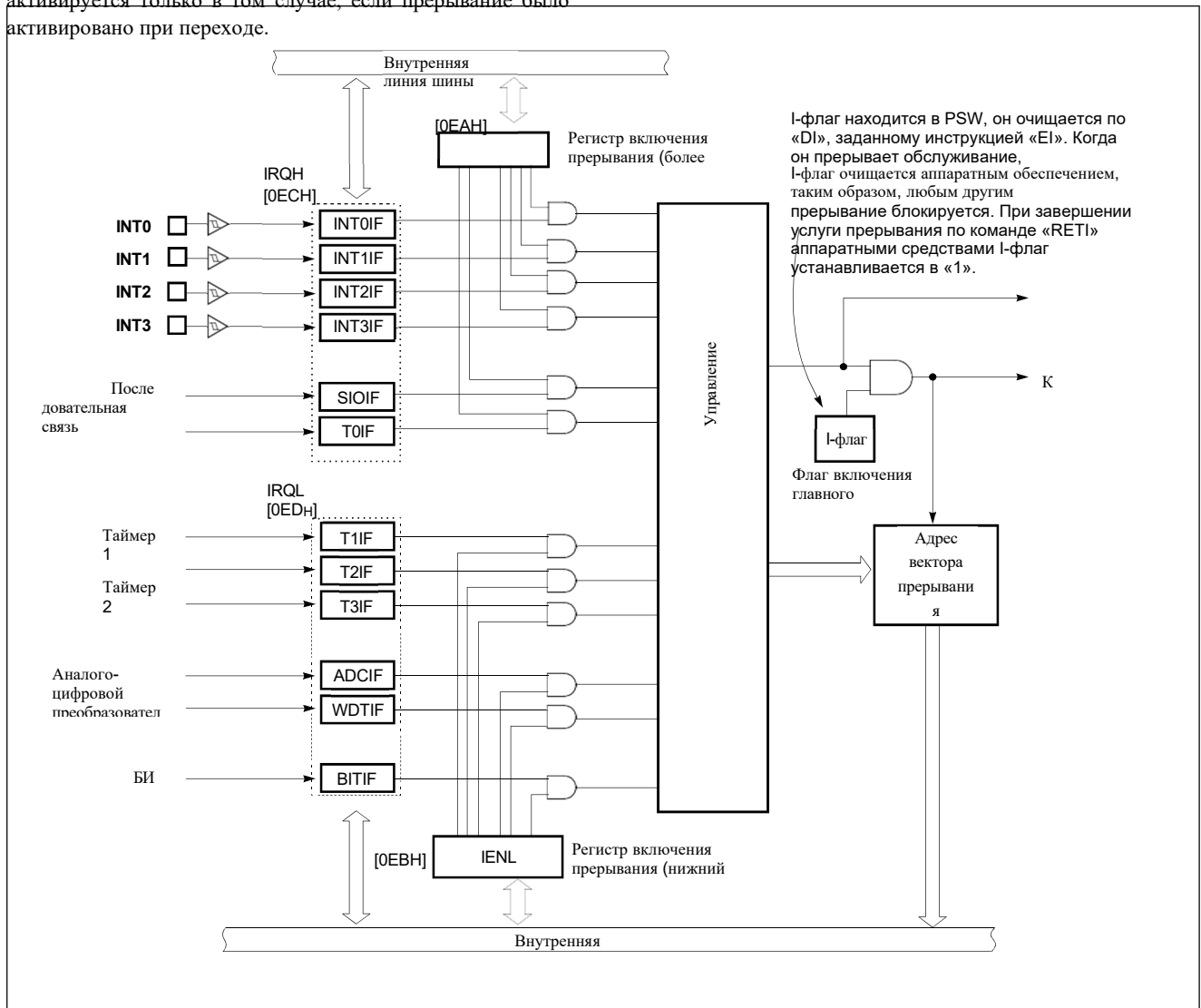


Рис. 17-1 Блок-схема прерывания

Прерывание таймера основного интервала генерируется BITIF, которое устанавливается переполнением в регистре

счетчика таймера.



Прерывание SIO генерируется SIOIF, который устанавливается посредством завершения приема или передачи данных SIO.

Прерывания управляются флагом I-flag включения главного прерывания (бит 2 включенного PSW), регистром разрешения Рис. 8-3 прерывания (IENH, IENL) и флагами запроса прерывания (в IRQH и IRQL), за исключением сброса питания и прерывания программного BRK. В таблице 17-1 показан приоритет прерывания.

Векторные адреса показаны на рис. 8-6. Параметры активизации прерываний показаны в Рисунок 17-2 Эти регистры состоят из флагов разрешения прерываний каждого источника прерываний, и эти флаги определяют, будет ли прерывание принято или нет. Когда флаг enable равен «0», соответствующий источник прерывания запрещен. Следует отметить, что PSW также содержит главный бит включения, I-флаг, который отключает все прерывания одновременно.

Сброс/прерывание	Символ	Приоритет
Аппаратный сброс	СБРОС	1
Внешнее прерывание 0	INT0	2
Внешнее прерывание 1	INT1	3
Внешнее прерывание 2	INT2	4
Внешнее прерывание 3	INT3	5
Последовательный ввод/вывод	SIO	6
Таймер/счетчик 0	Таймер 0	7
Таймер/счетчик 1	Таймер 1	8
Таймер/счетчик 2	Таймер 2	9
Таймер/счетчик 3	Таймер 3	10
Прерывание АЦП	ADC	11
Контрольный таймер	WDT	12
Системный таймер	SYST	13

**Таблица 17-1 Приоритет прерывания**

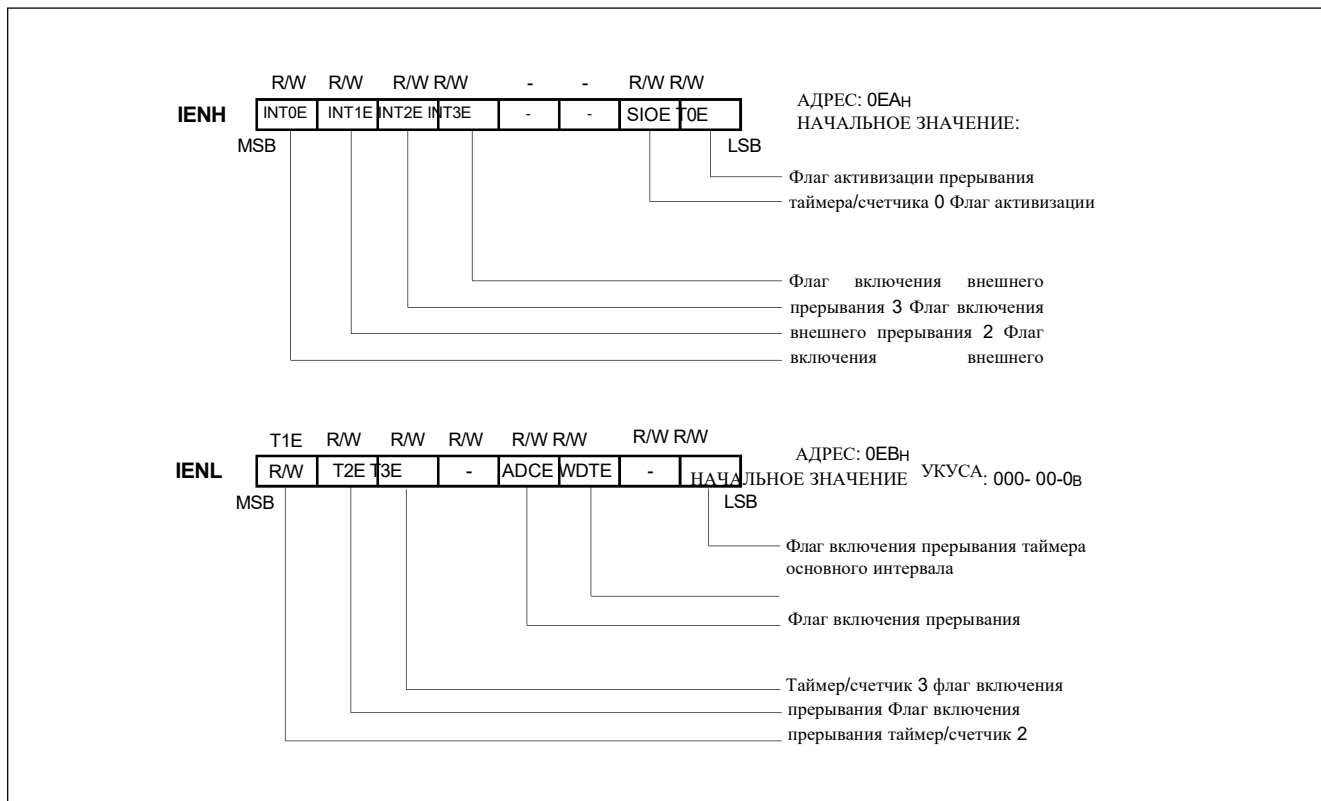


Рис. 17-2 Регистр флагов включения прерываний

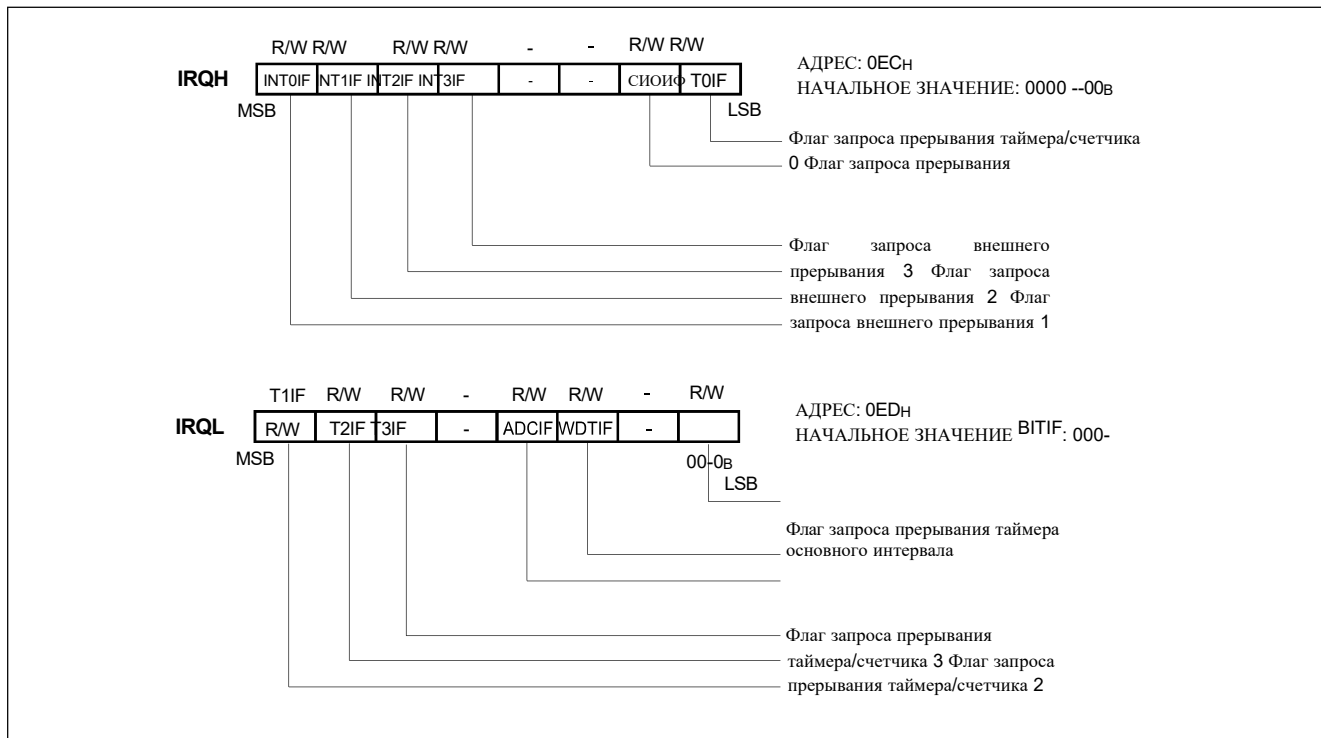


Рис. 17-3 Регистр флага запроса на прерывание

### 17.1 Последовательность прерываний

Запрос на прерывание удерживается до тех пор, пока прерывание не будет принято или защелка прерывания не будет сброшена на «0» посредством сброса или команды. Последовательность приемки между разрывами требует 8 циклов  $f_{XIN}$  ( $2\mu s$  при  $f_x$ ).

#### Прерывание приемки

1. Флаг включения главного прерывания (I-флаг) сбрасывается на «0», чтобы временно отключить принятие любых последующих маскируемых прерываний. Когда принимается немаскируемое прерывание, прием любых последующих прерываний временно деактивируется.
2. Содержимое счетчика программ (обратный адрес) и слово состояния программы сохраняются (толкаются) на

$IN = 4$  МГц) после завершения текущего выполнения команды. Служебная задача прерывания завершается после выполнения команды возврата прерывания [RETI].

3. Адрес входа программы обслуживания прерываний считывается из адреса таблицы векторов, и адрес входа загружается в счетчик программ.
4. Выполняется команда, сохраненная по адресу входа программы обслуживания между разрывами.

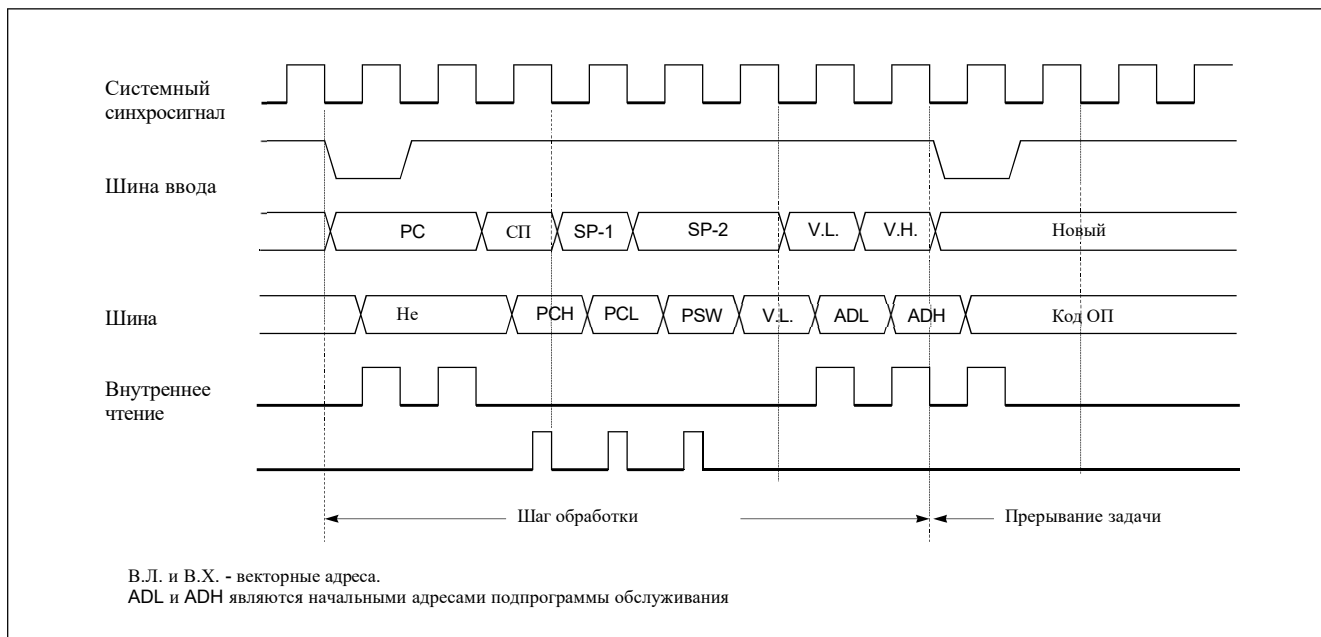
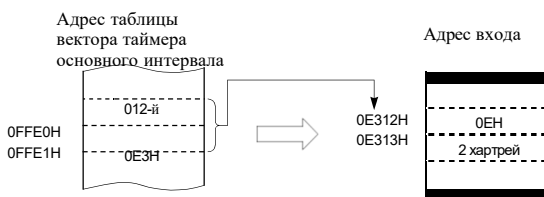


Рис. 17-4 Временная диаграмма инструкции по приемке и возврату прерывания



Соответствие между адресом таблицы векторов для прерывания ВIT и адресом входа программы обслуживания прерываний.

Запрос на прерывание не принимается до тех пор, пока флаг I не будет установлен в «1» даже если запрашиваемое прерывание имеет более высокий приоритет, чем у текущего обслуживаемого прерывания.

Когда требуется вложенная услуга прерывания, I-флаг должен быть установлен в «1» командой «EI» в программе услуги прерывания. В этом случае допустимые источники прерываний избирательно активируются отдельными флагами разрешения прерываний.

**Флаг запроса на сброс прерывания**

Флаг «Запрос на прерывание» не может быть сброшен во время обработки приема прерываний. После прерывания приемки она должна быть

очищено, как показано в подпрограмме обслуживания прерываний.

**Примечание.** Функция MC80F0708 и HMS87C1416B аналогична, но метод обработки прерываний отличается. При замене HMS87C1416B на MC80F0708 необходимо добавить флаг запроса на сброс прерывания.

**Пример: Флаг запроса на сброс прерывания**

```
T1_INT:   CLR1   T1IF; CLEAR T1 ЗАПРОС
          [обработка прерываний]
          RETI   ; ВОЗВРАТ
```

**Сохранение/восстановление регистра общего назначения**

Во время обработки приема прерываний счетчик программ и слово состояния программы автоматически сохраняются в стеке, но накопитель и другие регистры не сохраняются сами по себе. При необходимости эти регистры сохраняются программным обеспечением. Кроме того, при вложении нескольких служб прерываний необходимо избегать использования одной и той же области памяти данных для сохранения регистров.

Следующий метод используется для сохранения/восстановления регистров общего назначения.

**Пример: Зарегистрируйтесь, используя инструкции push и pop**

```
INTxx:   CLR1   INTxxIF ; ОЧИСТИТЬ ЗАПРОС.
          PUSH  A       ; СОХРАНИТЬ ACC.
          PUSH  X       ; СОХРАНИТЬ X REG.
          PUSH  Y       ; СОХРАНИТЬ Y REG.
```

```
[обработка прерываний]
POP      Y       ; ВОССТАНОВИТЬ Y
POP      X       REG.
POP      A       ; ВОССТАНОВИТЬ X
RETI     REG.
          ; ВОССТАНОВИТЬ
          ACC.
          ; Вернуть
```

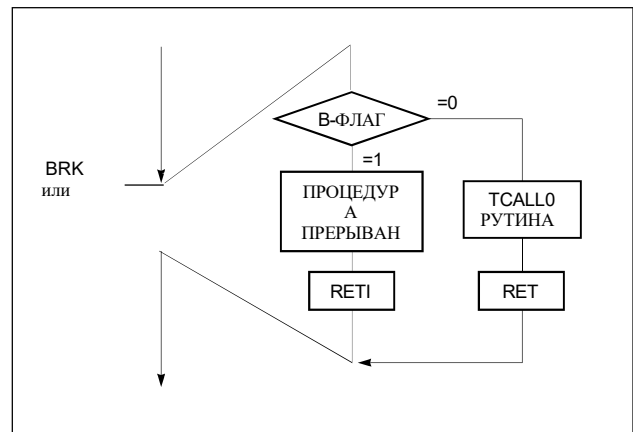
Сохранение/восстановление регистров общего назначения с использованием команд push и pop;

**17.2 Прерывание BRK**

Прерывание программного обеспечения может быть вызвано командой BRK, которая имеет самый низкий порядок приоритета.

Адрес вектора прерывания BRK совместно используется с вектором TCALL 0 (см. раздел памяти программы). При формировании межразрывного соединения BRK В-флаг PSW устанавливается для отличия BRK от TCALL 0.

Каждый этап обработки определяется В-флагом, как показано на фиг17-5.



**Рис. 17-5** Выполнение BRK/TCALL0

**17.3 Множественное прерывание**

Если одновременно принимаются два запроса различных уровней приоритета, то обслуживается запрос более высокого уровня приоритета. Если повторные запросы прерывания принимаются одновременно, то внутренняя последовательность опроса определяется по аппаратным средствам

программа, запрос которой обслуживается. Однако возможна многократная обработка с помощью программного обеспечения для специальных функций. Обычно, когда принимается прерывание, I-флаг сбрасывается, чтобы отключить любое дальнейшее прерывание. Но как пользователь устанавливает I-флаг в

подпрограмме прерываний, некоторые дальше

прерывание может обслуживаться, даже если выполняется определенное прерывание.

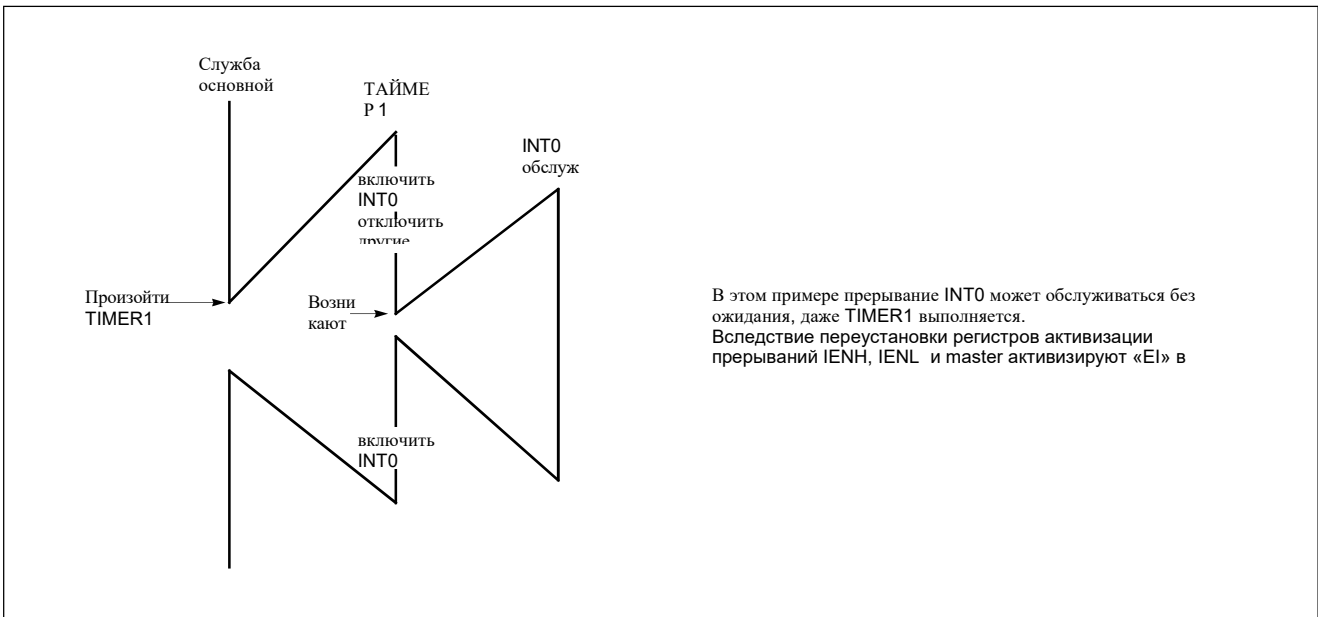


Рис. 17-6 Выполнение нескольких прерываний

**Пример:** Во время Timer1 прерывания выполняется, INT0 прерывание обслуживается без приостановки.

```
TIMER1: CLR1 T1IF; Clear Timer1 Запрос
        PUSH A
        PUSH X
        PUSH Y
        LDM IENH, # 80H ; Включить только INT0
        LDM IENL, # 0 ; Отключить другие int.
        EI ; Включить прерывание
        :
```

```
:
:
:
:
LDM IENH, # 0FFH; Активизация всех прерываний
LDM IENL, # 0FFH
POP X
POP A
RETI
```

### 17.4 Внешнее прерывание

Внешнее прерывание на контактах INT0, INT1, INT2 и INT3 инициируется краями в зависимости от регистра выбора краев IEDS (address 0EEH), как показано на рис. 17-7.

Обнаружение края внешнего прерывания имеет три режима действия перехода: передний край, задний край и оба края.

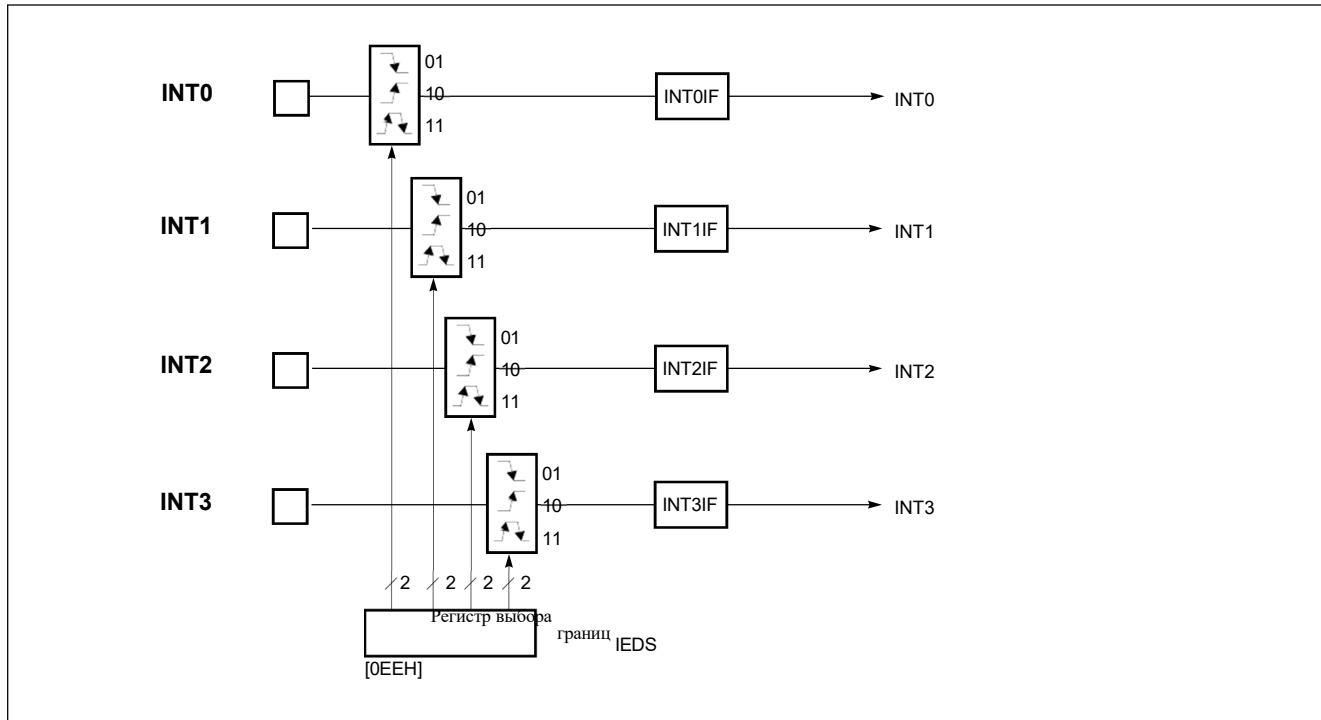


Рис. 17-7 Блок-схема внешних прерываний

INT0 ~ INT3 мультиплексируются с общими портами ввода-вывода (R11, R12, R03, R00). Для использования в качестве внешнего вывода прерывания бит регистра выбора порта PSR0 должен быть установлен в «1» соответственно.

**Пример:** Использование в качестве INT0 и INT2

```

:
; **** Установите внешний порт прерывания в состояние
; подтягивания.
LDM  PU1, # 0000 _ 0101B
;
; **** Установить порт в качестве внешнего порта
; прерывания
LDM  PSR0, # 0000 _ 0101B
;
; **** Установка обнаружения падающих краев
LDM  IEDS, # 0001 _ 0001B
:
    
```

#### ответа

Края INT0 ~ INT3 фиксируются в INT0IF ~ INT3IF при каждом цикле машины. Эти значения фактически не опрашиваются схемой до следующего цикла машины. Если запрос активен и условия являются правильными для его подтверждения, аппаратный вызов подпрограммы запрашиваемого обслуживания будет следующей выполняемой командой. Сам DIV занимает двенадцать циклов. Таким образом, между активацией запроса внешнего прерывания и началом выполнения первой команды служебной подпрограммы проходит минимум двенадцать полных машинных циклов.

Рисунок 17-8 показывает тайминги ответа на прерывание.

#### Время

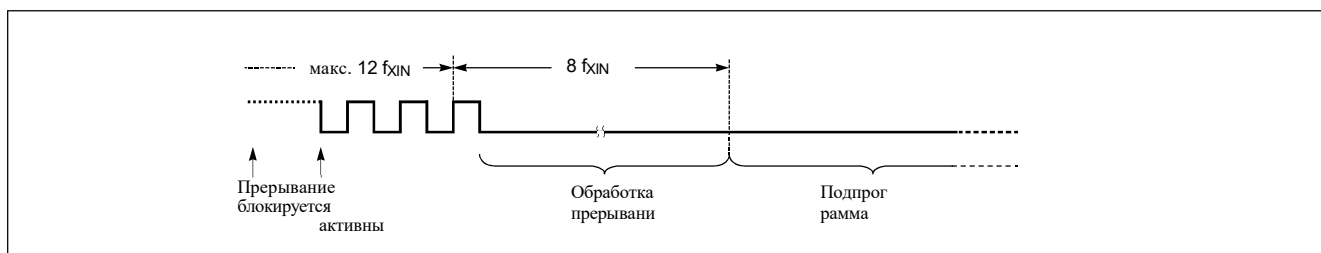


Рис. 17-8 Схема синхронизации ответа на прерывание



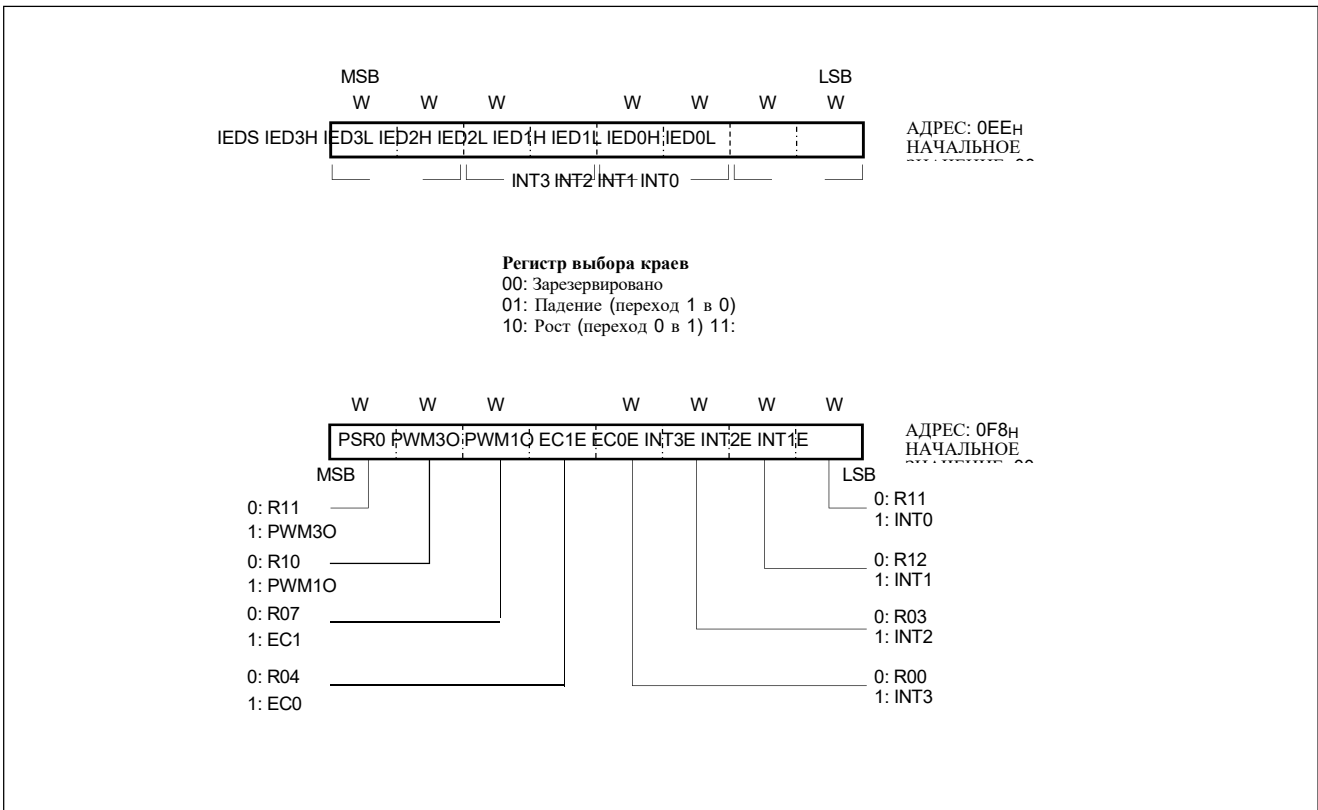


Рис. 17-9 Регистр IEDS и регистр выбора портов PSR0

## 18. РАБОТА ПО ЭНЕРГОСБЕРЕЖЕНИЮ

В MC80F0704/0708/0804/0808 имеется два режима отключения питания. В режиме отключения питания энергопотребление значительно снижается. Для приложений, в которых энергопотребление является критическим фактором, устройство обеспечивает два вида функции энергосбережения -

### 18.1 Спящий режим

В этом режиме цепи внутренних колебаний остаются активными. Колебания продолжаются, и периферийные устройства работают нормально, но ЦП останавливается. Движение всех периферийных устройств индицируется в режиме Таблица 18-1. SLEEP вводится установкой регистра SSCR в «0Fh.» Это

освобождается сбросом или прерыванием. Для разъединения прерыванием перед режимом сна необходимо включить прерывание.

освобождается сбросом или прерыванием. Для разъединения прерыванием перед режимом сна необходимо включить прерывание.

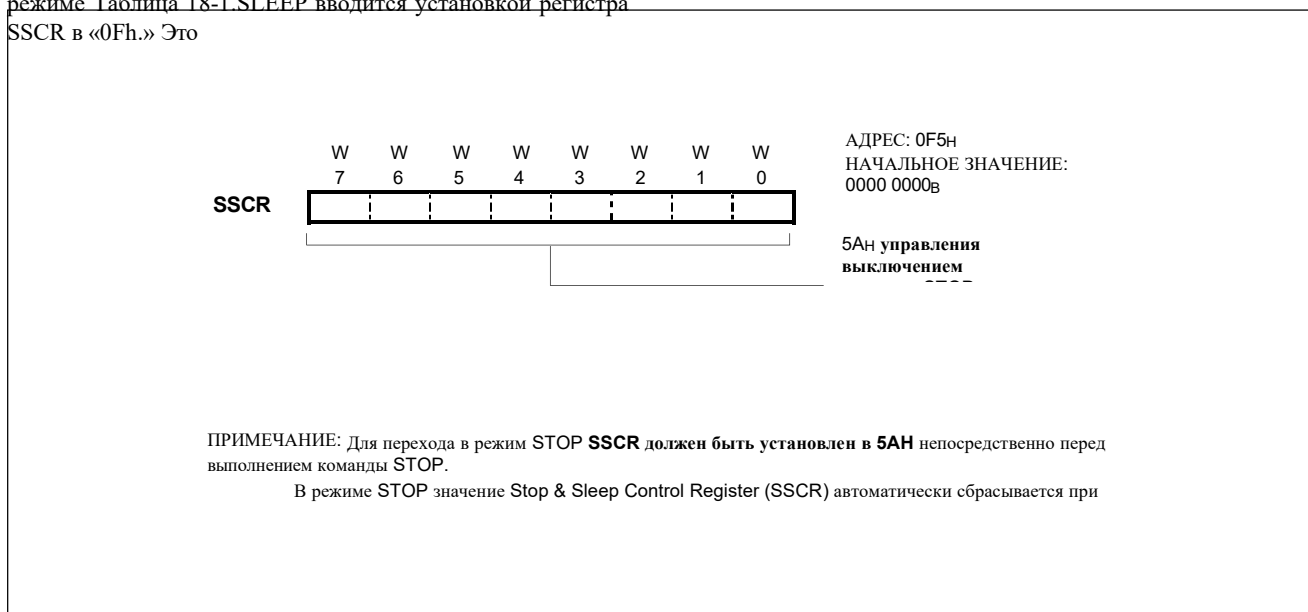


Рис. 18-1. Регистр управления STOP и SLEEP

### Отпустите режим SLEEP

Выход из режима сна - аппаратный сброс или все прерывания. Сброс повторно определяет все регистры управления, но не изменяет ОЗУ на кристалле. Прерывания позволяют как встроенным RAM, так и контрольным регистрам сохранять свои значения.

Если I-флаг = 1, происходит нормальный ответ на прерывание. Если I-флаг = 0, микросхема возобновит выполнение, начиная с команды, следующей за командой SLEEP. Он не будет вектором прерывать последовательную процедуру. (см. рисунок 18-4)

При выходе из режима сна путем сброса для нормальной работы требуется достаточное время стабилизации колебаний. На рис. 18-3 показана временная диаграмма. При выходе из режима сна таймер основного интервала активируется при пробуждении. Она увеличивается от 00н до FFh. Переполнение счетчика устанавливается для запуска нормальной работы. Следовательно, перед инструкцией SLEEP пользователь должен установить соответствующий коэффициент деления prescaler, чтобы иметь достаточно длительное время (более 20 мс). Это гарантирует запуск и остановку осциллятора. По прерываниям выход из режима сна показан на рисунке 18-2. При сбросе выход из режима сна показан в Рис. 18-3.

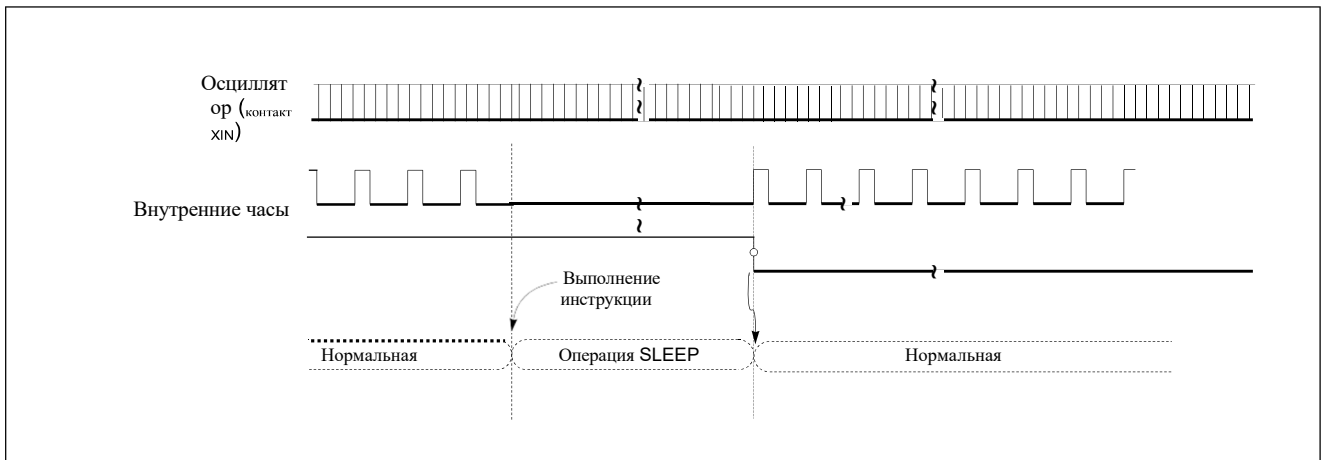


Рис. 18-2. Синхронизация освобождения режима сна при внешнем прерывании

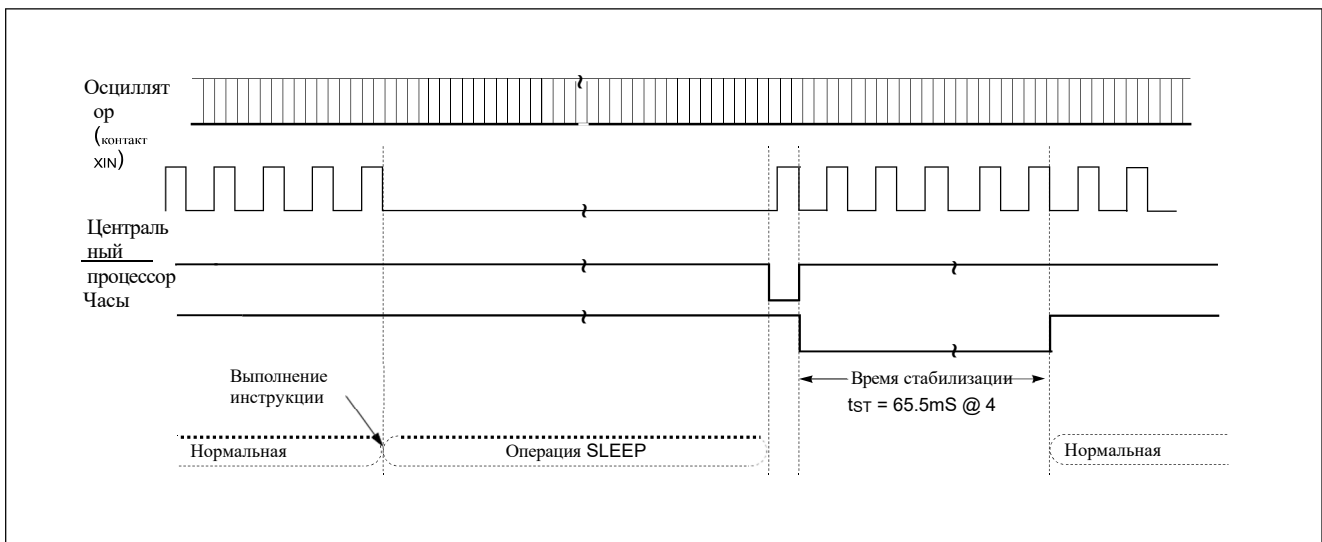


Рис. 18-3 Синхронизация освобождения режима сна путем сброса

## 18.2 Режим остановки

В режиме Stop основной осциллятор, системный тактовый сигнал и периферийный тактовый сигнал останавливаются, но RC-колебательный сторожевой таймер продолжает работать. При замораживании тактового сигнала все функции останавливаются, но регистры ОЗУ и управления сохраняются. Порт устанавливает контакты для значений, удерживаемых соответствующим регистром данных порта, реестрами направления порта. Осциллятор останавливается, и внутренние операции системы задерживаются.

- Состояния ОЗУ, регистров и защелок, действительные непосредственно перед переводом системы в состояние STOP, сохраняются.
- Программный счетчик останавливает адрес выполняемой команды после команды «STOP», которая запускает режим работы

STOP.

---

**Примечание:** Режим Stop активизируется выполнением команды STOP после установки SSCR в «5A<sub>H</sub>». (Этот регистр должен быть записан по байту. Если этот регистр установлен командой битовой манипуляции, например командой «set1» или «clr1», это может быть нежелательной операцией)

В режиме остановки V<sub>DD</sub> может быть снижен до минимума энергопотребления. Однако необходимо следить за тем, чтобы уровень V<sub>DD</sub> не уменьшался до вызова режима остановки и чтобы уровень V<sub>DD</sub> был восстановлен до нормального рабочего уровня до завершения режима остановки.

Сброс не должен быть активирован до восстановления нормального рабочего уровня V<sub>DD</sub> и должен поддерживаться в активном состоянии достаточно долго, чтобы обеспечить перезапуск и стабилизацию генератора.

**Примечание.** После инструкции STOP должно быть записано не менее двух или более инструкций NOP.

Ex) LDM CCTLR, # 0FH; более 20 мс  
LDM SSCR, # 5AH  
Стоп  
NOP; для времени стабилизации NOP ; для времени стабилизации

При операции STOP рассеяние мощности, связанной с генератором и внутренними аппаратными средствами, снижается; однако,

рассеяние

мощности, связанное с контактным интерфейсом (зависящее от внешней схемы и программы), не определяется непосредственно аппаратной операцией функции STOP. Эта точка должна быть малой, когда входной уровень стабилен на уровне напряжения питания ( $V_{DD}/V_{SS}$ ); однако, когда входной уровень становится выше уровня напряжения мощности (приблизительно на 0,3-0,5 В), начинает протекать ток. Следовательно, если отключение выходного транзистора на порте ввода-вывода переводит штыревой сигнал в состояние высокой скорости, ток протекает через входные порты транзистора, требуя фиксировать уровень посредством подтягивания или другими средствами.

Периферия	Режим STOP	СПЯЩИЙ режим
Центральный процессор	Стоп	Стоп
RAM	Сохранить	Сохранить
Основной таймер интервала	Остановленный	Работает непрерывно
Контрольный таймер	Стоп (работает только в режиме RC-WDT)	Стоп
Таймер/счетчик	Остановлено (таймер работает нормально только в том случае, если включен режим счетчика событий)	Работает непрерывно
Зузер, АЦП	Стоп	Стоп
SIO	Работать только с внешним синхросигналом	Работать только с внешним синхросигналом
Генератор	Стоп ( $X_{IN} = L, X_{OUT} = H$ )	Колебание
Порты ввода-вывода	Сохранить	Сохранить
Контрольные регистры	Сохранить	Сохранить
Внутренний контур	Режим остановки	Спящий режим
Делитель частоты	Сохранить	Активный
Шина адресных данных	Сохранить	Сохранить
Источник деблокирования	Сброс, таймер (EC0,1), SIO (внутренний тактовый сигнал), контрольный таймер (режим RC-WDT), внешнее прерывание	Сброс, все прерывания

**Таблица 18-1 Работа периферийных устройств в режиме энергосбережения**

### Отпустите режим STOP (СТОП)

Источником выхода из режима STOP является аппаратный сброс, внешнее прерывание, таймер (EC0,1), WDT, SIO. При освобождении из режима STOP посредством SIO (ext.clock) SIOR имеет фиктивные данные, и бит SIOST должен быть сброшен после выхода из режима STOP.

Сброс повторно определяет все регистры управления, но не изменяет ОЗУ на кристалле. Внешние прерывания позволяют как встроенным RAM, так и контрольным регистрам сохранять свои значения.

Если I-флаг = 1, происходит нормальный ответ на прерывание. Если I-флаг = 0, микросхема возобновит

выполнение, начиная с команды, следующей за командой STOP. Он не будет вектором прерывать обслуживание

рутина. (смРис. 18-4.)

При выходе из режима «Стоп» по внешнему прерыванию для нормальной работы требуется достаточное время стабилизации колебаний. Рисунок 18-5 На показана временная диаграмма. При выходе из режима Stop таймер Basic interval активируется при пробуждении. Она увеличивается от 00<sub>H</sub> до FF<sub>H</sub>. Переполнение счетчика настроено на запуск нормальной работы. Следовательно, перед инструкцией STOP пользователь должен установить соответствующий коэффициент деления prescaler, чтобы иметь достаточно длительное время (более 20 мс). Это гарантирует запуск и остановку осциллятора.

При сбросе выход из режима остановки отображается вРисунок 18-6.

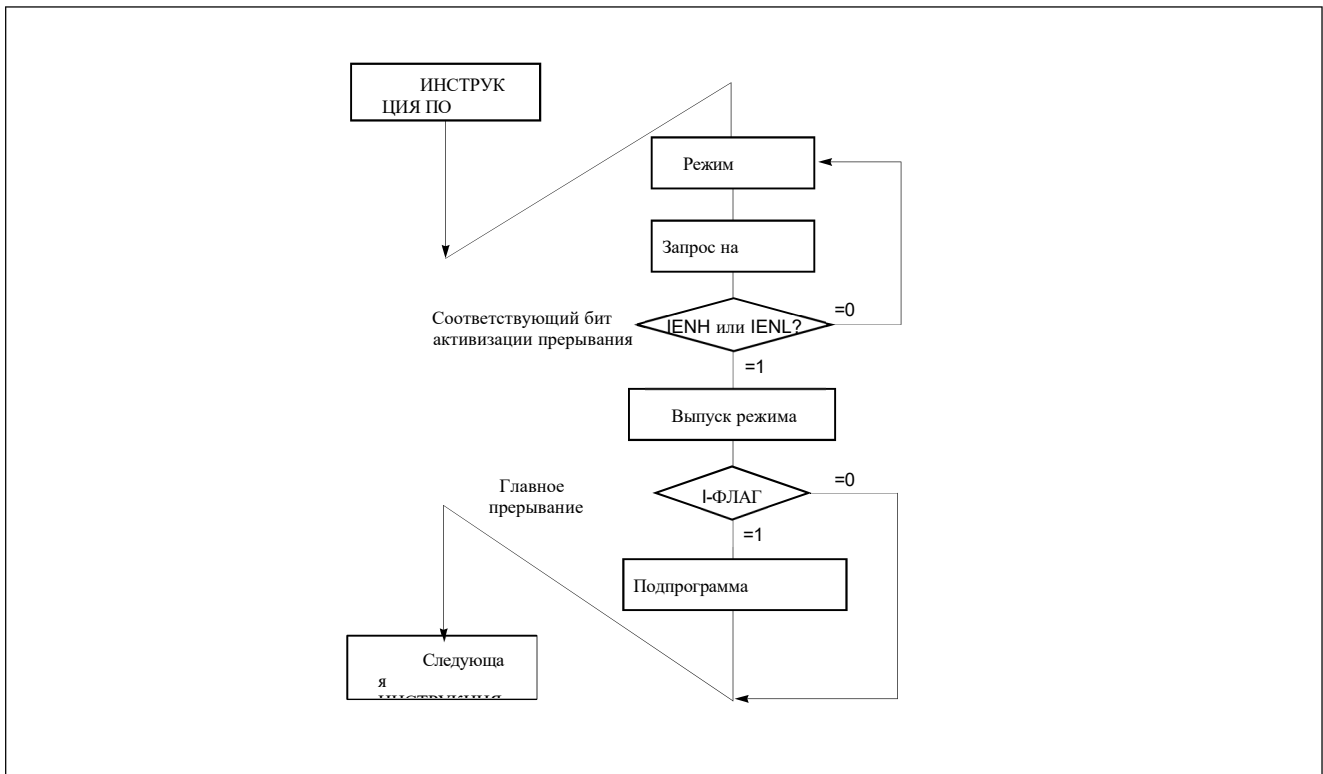


Рис. 18-4. Прекращение отпуская потока прерываниями

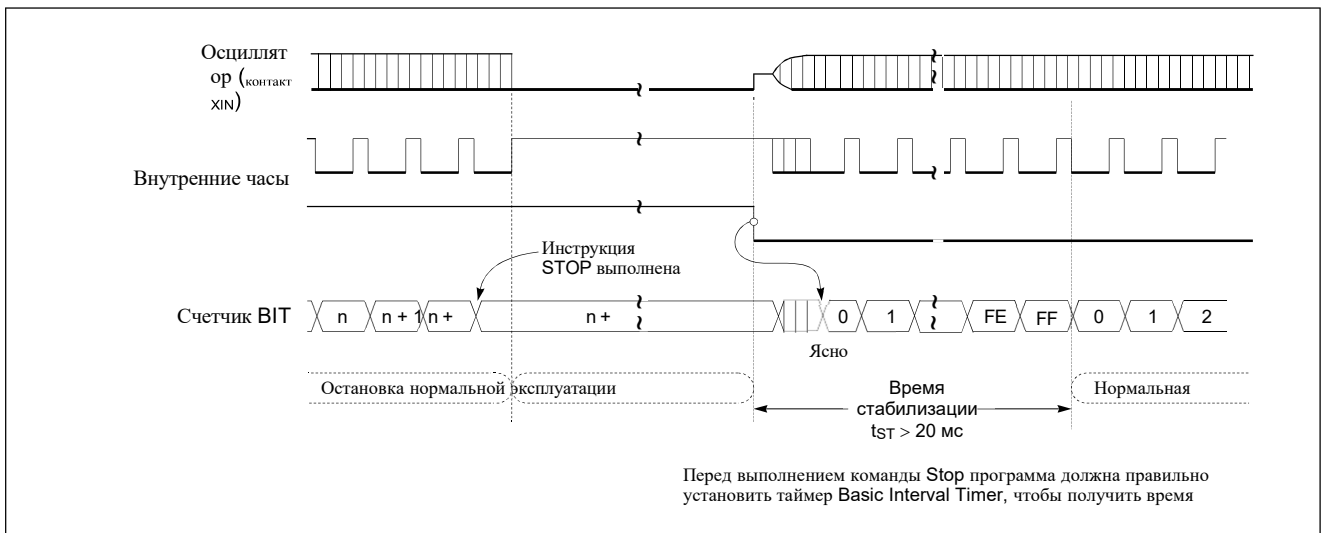


Рис. 18-5 Время освобождения режима STOP при внешнем прерывании

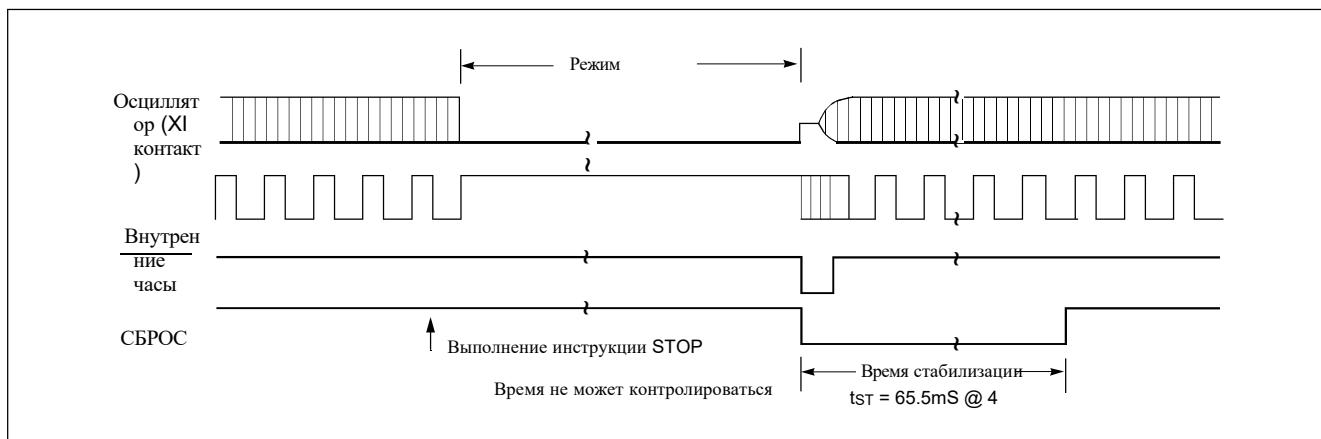


Рис. 18-6 Синхронизация сброса режима STOP посредством сброса

### 18.3 Режим остановки во внутреннем режиме RC-осциллированного сторожевого таймера

В режиме внутреннего RC-осциллированного сторожевого таймера генератор на кристалле останавливается. Но внутренний RC колебательный контур в этом режиме облучается. Регистры ОЗУ и управления хранятся на кристалле. Порт устанавливает контакты для значений, удерживаемых соответствующим регистром данных порта, реестрами направления порта.

Режим внутреннего RC-осциллированного сторожевого таймера активируется выполнением команды STOP после установки бита RCWDT CCTLR в «1». (Этот регистр должен быть записан байтовой операцией. Если этот регистр установлен командой битовой манипуляции, например, командой «set1» или «clr1», это может быть нежелательной операцией)

**Примечание:** Внимание: После инструкции STOP следует написать не менее двух или более инструкций NOP

Например,  
 LDM WDTR, # 1111  
 \_ 1111B LDM CCTLR, #  
 0010 \_ 1110B LDM SSCR, #  
 0101 \_ 1010B STOP  
 NOP ; для времени  
 стабилизации NOP ;  
 для времени стабилизации

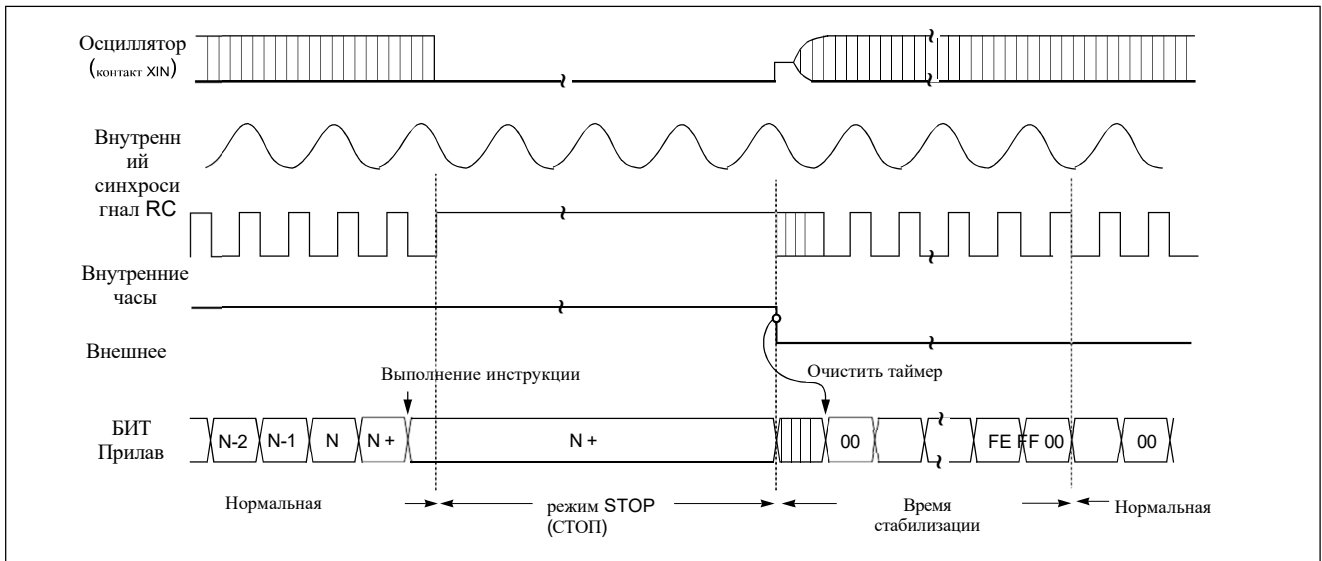
Выход из режима внутреннего RC-осциллированного контрольного таймера - аппаратный сброс или внешнее прерывание или прерывание контрольного таймера

(в режиме RC-watchdog timer). Сброс повторно определяет все регистры управления, но не изменяет ОЗУ на кристалле. Внешние межразрывы позволяют как встроенным RAM, так и контрольным регистрам сохранять свои значения.

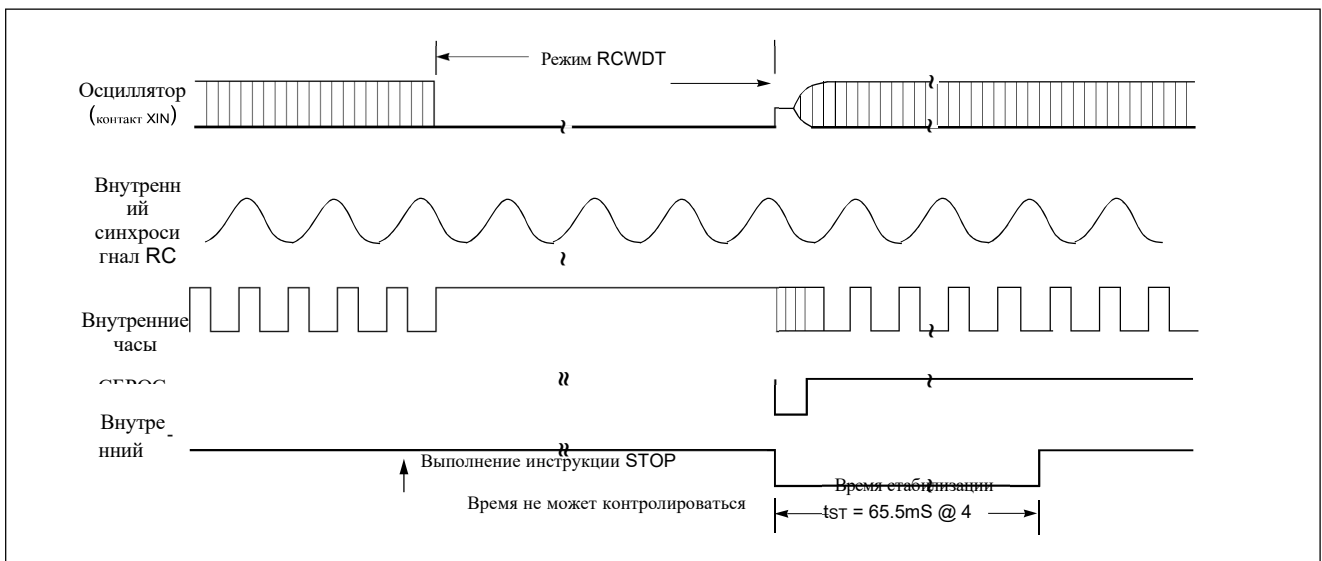
Если I-флаг = 1, происходит нормальный ответ на прерывание. В этом случае, если бит WDTON CCCTLR установлен в «0», а бит WDTE IENH установлен в «1», то устройство будет выполнять подпрограмму обслуживания прерываний сторожевого таймера (рис. 8-6). Однако, если бит WDTON CCCTLR установлен в «1», устройство генерирует внутренний сигнал сброса и выполняет обработку сброса (рис. 18-8). Если I-флаг = 0, микросхема возобновит выполнение, начиная с команды, следующей за командой STOP. Он не будет вектором прерывать процедуру обслуживания (см. рис. 18-4).

При выходе из режима Stop в режиме Internal RC-Sombed Watchdog Timer по внешнему прерыванию для нормальной работы требуется время стабилизации колебаний. Рисунок 18-7 На показана диаграмма времени. При освобождении режима внутреннего RC-осциллированного сторожевого таймера при пробуждении активируется основной таймер интервала. Она увеличивается от  $00_H$  до  $FF_H$ . Переполнение счетчика устанавливается для запуска нормальной работы. Следовательно, перед инструкцией STOP пользователь должен установить соответствующий коэффициент деления prescaler, чтобы иметь достаточно длительное время (более 20 мс). Это гарантирует запуск и стабилизацию генератора. При сбросе выход из внутреннего режима RC-осциллированного сторожевого таймера показан на рис. 18-8.





**Рис. 18-7 Освобождение режима остановки во внутреннем режиме RC-WDT по внешнему прерыванию или прерыванию WDT**



**Рис. 18-8 Освобождение режима внутреннего RC-WDT путем сброса**

### 18.4 Минимизация потребления тока

Режим остановки предназначен для снижения энергопотребления. Чтобы минимизировать ток, вытянутый во время режима остановки, пользователь должен возвращаться

выключенные выходные драйверы, которые являются источником или потоком тока, если это практично.

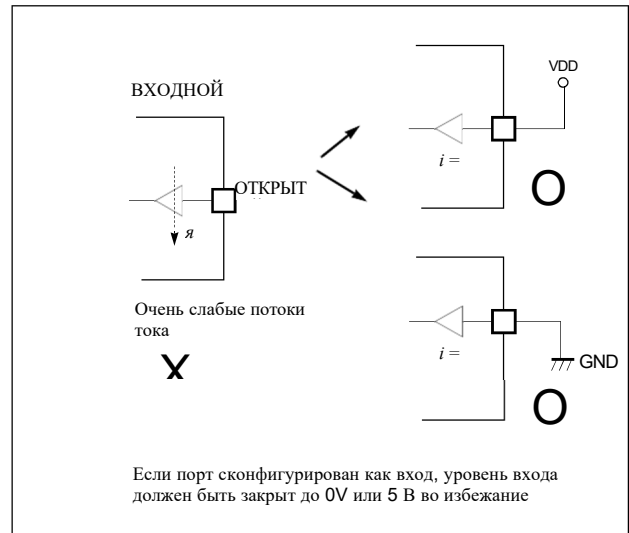
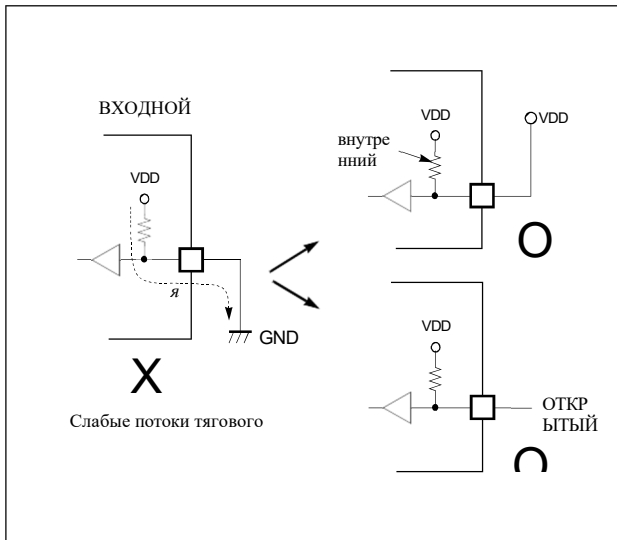


Рис. 18-9 Пример применения неиспользуемого порта ввода

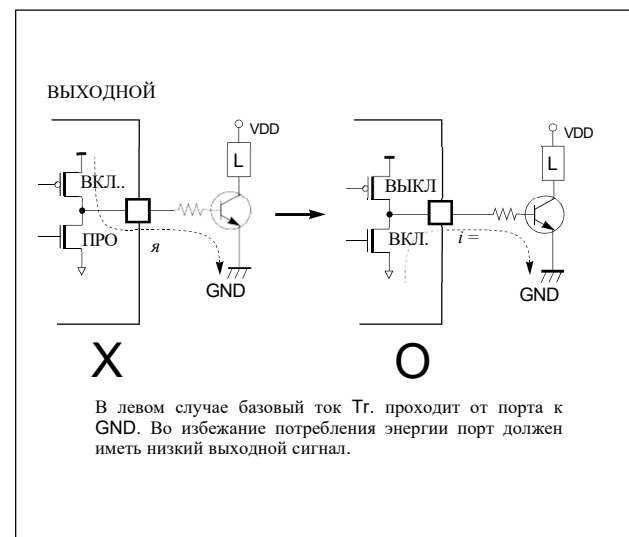
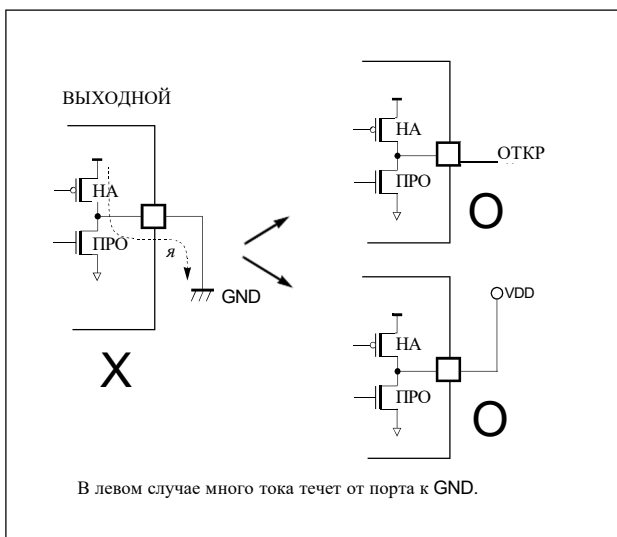


Рис. 18-10 Пример применения неиспользуемого выходного порта

**Примечание.** В операции STOP связано рассеяние мощности с генератором и опускают внутреннее оборудование; однако рассеяние мощности, связанное с контактным интерфейсом (в зависимости от внешней схемы и программы), не определяется непосредственно аппаратной операцией функции STOP. Эта точка должна быть малой, когда входной уровень стабилен на уровне напряжения питания ( $V_{DD}/V_{SS}$ ); однако, когда входной уровень становится выше, чем уровень напряжения мощности (приблизительно на 0,3 В), начинает протекать ток. Следовательно, если отключение выходного транзистора на порте ввода/вывода переводит штыревой сигнал в состояние высокого импеданса, ток протекает через входные транзисторы

подтягивание или другие средства.

портов, требуя, чтобы он фиксировал уровень на

Он должен быть установлен правильно, чтобы текущий поток через порт не существовал.

Сначала рассмотрите установку порта в режим ввода. Убедитесь в отсутствии тока после рассмотрения его взаимосвязи с внешним контуром. В режиме ввода полное сопротивление контактов от внешнего MCU очень велико, чтобы ток не протекал.

Но уровень входного напряжения должен быть  $V_{SS}$  или  $V_{DD}$ . Будьте осторожны, если не указано напряжение, т.е. если неопределенный уровень напряжения (не  $V_{SS}$  или

V<sub>DD</sub>) подается на входной вывод, ток может быть небольшим (макс. 1mA при токе около 2 В).

Если неуместно устанавливать в качестве режима ввода, то установить в режим вывода, учитывая отсутствие потока тока. Установка порта в положение High (высокий) или Low (низкий) определяется с учетом его взаимосвязи с внешним

nal цепь. Например, если имеется внешний повышающий резистор, то он устанавливается в выходной режим, т.е. в высокий, и если имеется внешний понижающий регистр, он устанавливается в низкий.

## 19. СБРОС

Как показано ниже, MC80F0704/0708/0804/0808 поддерживает различные виды сброса.

- Сброс при включении питания (POR)
- $\overline{\text{RESET}}$  (внешняя схема сброса)

- Сброс времени ожидания контрольного таймера
- Сброс функции обнаружения сбоя питания (PFD)
- Не удалось сбросить адрес

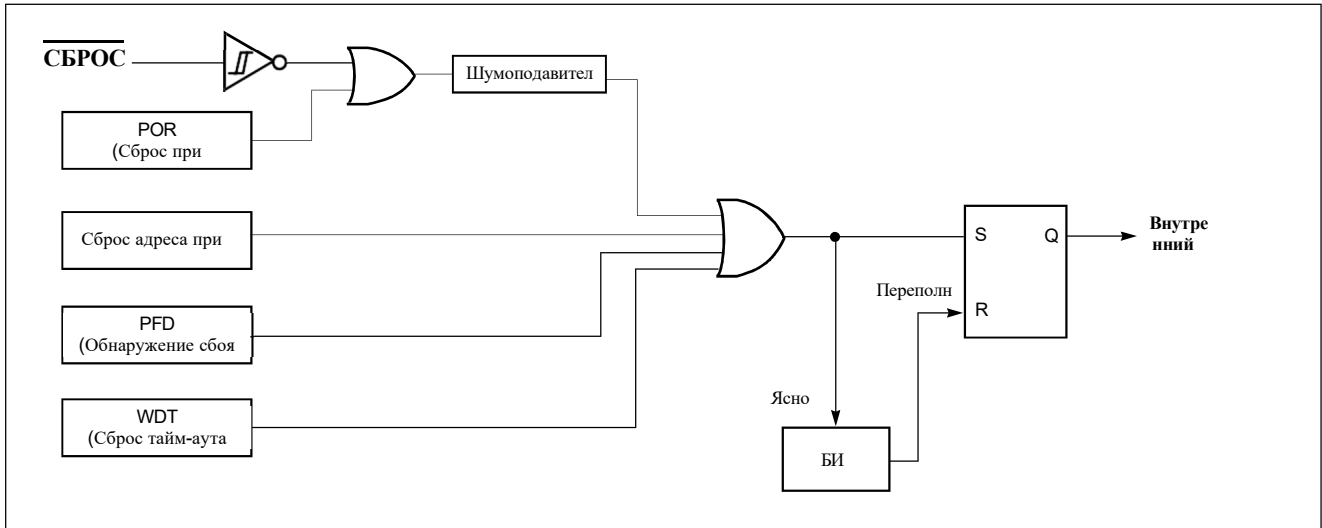


Рис. 19-1 Блок-схема RESET

Схема POR на кристалле удерживает устройство в состоянии RESET до тех пор, пока VDD не достигнет достаточно высокого уровня для правильной работы. Он устранит внешние компоненты, такие как микросхема сброса или внешний резистор и конденсатор для внешней цепи сброса. Кроме того, контакт RE-SET можно использовать для подключения к нормальному входному порту R35 путем установки параметра «POR».

и «R35EN» bit Configuration Area (20FFH) во флэш-памяти. При включении устройства в нормальную работу должны быть соблюдены его рабочие параметры (напряжение, частота, температура и т.д.).

Таблица 19-1 показывает аппаратную инициализацию на кристалле посредством операции сброса.

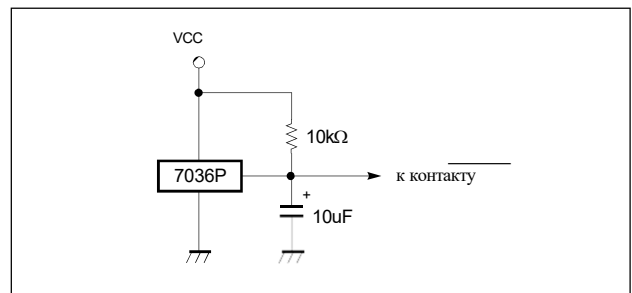
Аппаратное обеспечение на кристалле	Начальное значение	Аппаратное обеспечение на кристалле	Начальное значение
Счетчик программ (PC)	(FFFFH) - (FFFEH)	Периферийные часы	Прочь
Регистр страниц RAM (RPR)	0	Контрольный таймер	Отключить
		Контрольные регистры	См.Таблица 8-1 на стр. 30

Таблица 19-1 Инициализация внутреннего состояния посредством действия сброса

Вход сброса - это вывод RESET, который является входом триггера Шмитта. Сброс выполняется путем удержания контакта RESET на низком уровне в течение, по меньшей мере, 8 периодов генератора в пределах диапазона рабочего напряжения и стабильности колебаний, и внутреннее состояние инициализируется. После сброса необходимо добавить 65,5 мс (при частоте 4 МГц) с 7 периодами работы генератора, как показано на рис. 19-3.

Сброс не влияет на внутреннюю ОЗУ. Когда VDD включен, содержимое ОЗУ является неопределенным. Поэтому эта ОЗУ должна быть инициализирована перед прочтением или тестированием.

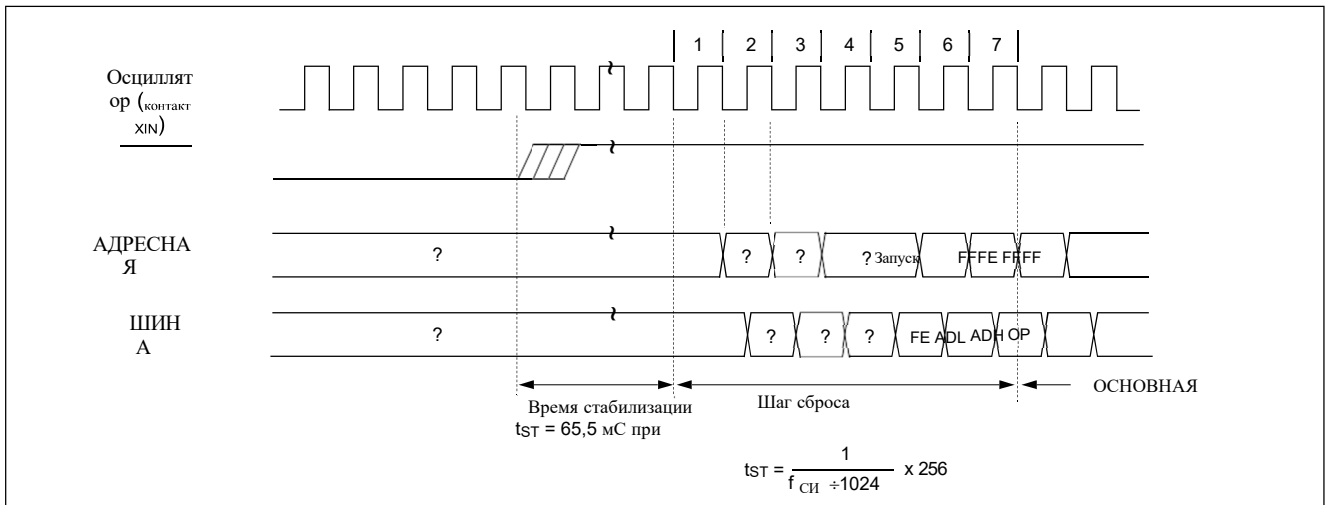
Когда вход контакта RESET становится высоким, операция сброса повторно освобождается, и выполнение программы



начинается по векторному адресу  
хранится по адресам FFFE<sub>H</sub> - FFFF<sub>H</sub>.

Подключение для простого включения и сброса показано  
на Рисунок 19-2.

**Рис. 19-2. Простая схема включения и сброса  
питания**



**Рис. 19-3** Схема синхронизации после сброса

Reset Address Fail - это функция сброса системы путем проверки доступа кода к ненормальному и незаполненному адресу, вызванному ошибочным программным кодом или внешним шумом, который не может быть возвращен в нормальное рабочее состояние и может стать неисправным. Если CPU пытается получить команду из неэффективной

кодированная область или область ОЗУ, происходит сброс адреса. Для получения информации о настройке параметра «address fail» см. рис. 11-2.

## 20. ПРОЦЕССОР СБОЯ ПИТАНИЯ

Для защиты от шума питания MC80F0704/0708/0804/0808 имеет встроенную схему подавления отказа питания. Регистр конфигурации, PFDR, может включать или отключать схему обнаружения сбоя питания. Всякий раз, когда напряжение отказа питания  $V_{DD}$  падает близко к или ниже значения напряжения отказа питания в течение 100 нс, ситуация отказа питания может быть сброшена или зафиксирована MCU в соответствии с битом PFDM PFDR. См. "Рис. 20-1 Мощность

Регистр детектора отказавшего напряжения "на стр. 97.

В внутриконтурном эмуляторе функция отказа питания не реализована и пользователь не может экспериментировать с ней. Поэтому после окончательной разработки пользовательской программы эту функцию можно экспериментировать или оценивать.

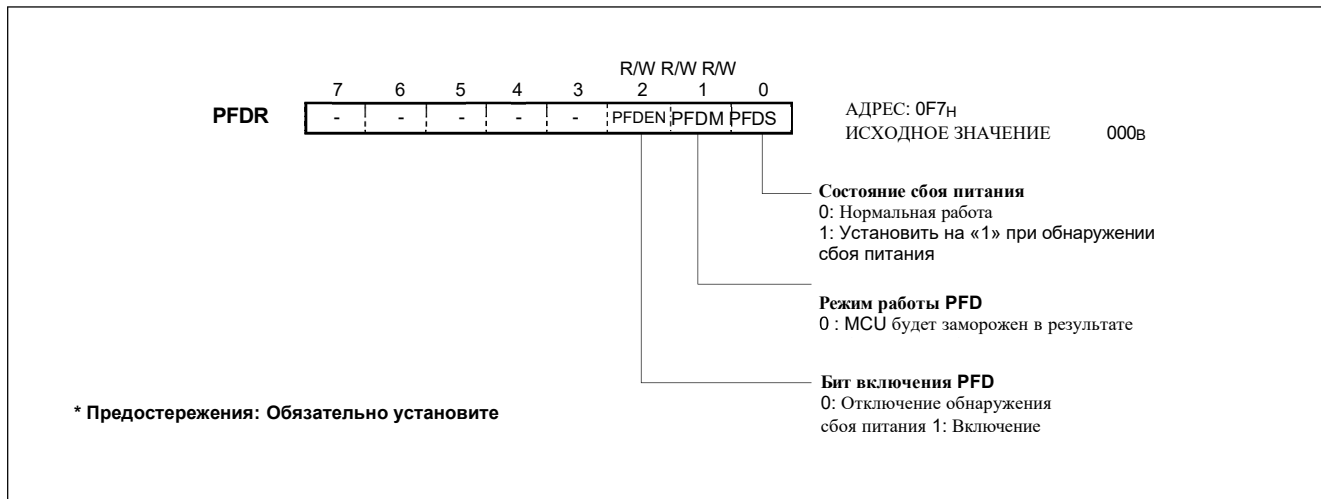


Рис. 20-1 Регистр детектора напряжения сбоя питания

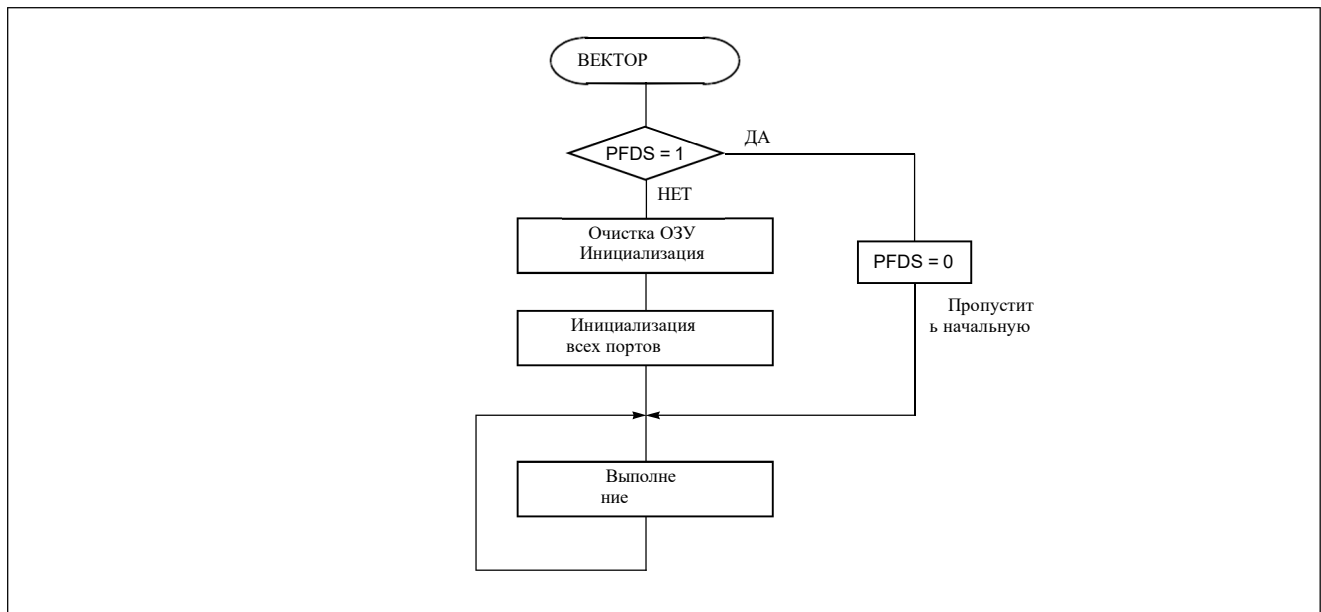


Рис. 20-2 Пример S/W потока сброса по отказу питания



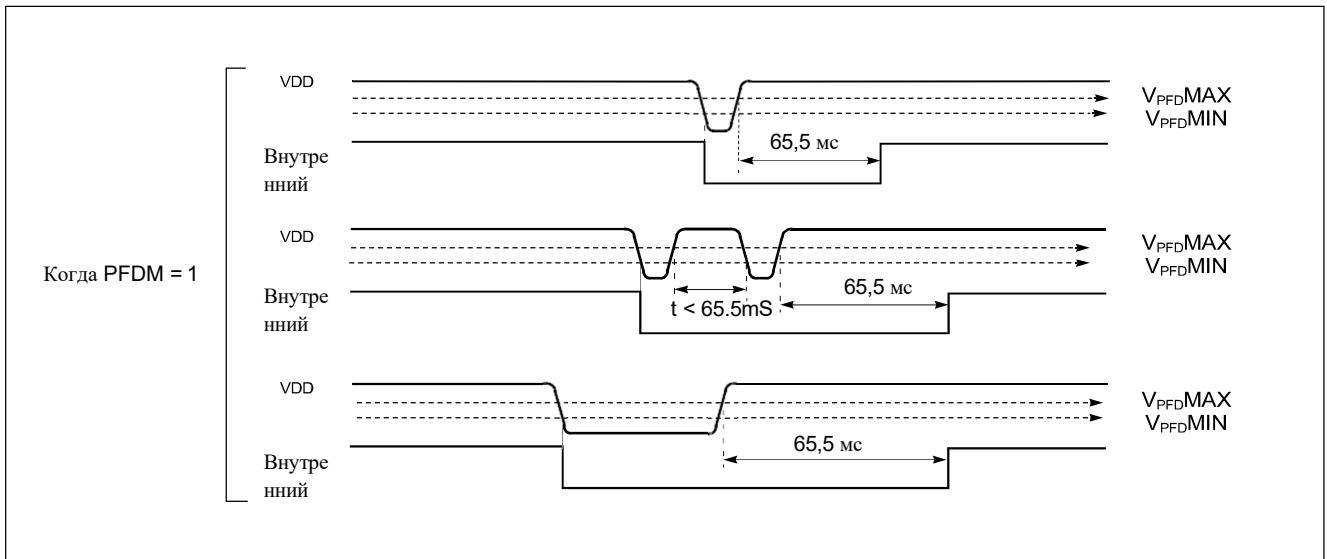


Рис. 20-3 Ситуации сбоя питания процессора (при 4MHz работе)

## 21. КОНТРОЛЬ ПО БОРЬБЕ С ШУМОМ

### 21.1 Защита от шумов колебаний

Предохранитель осцилляционного шума (ONP) используется для подачи стабильного внутреннего системного тактового сигнала путем исключения шума, который может быть введен в осциллятор, и восстановления отказа осцилляции. Эта функция может быть активизирована или деактивирована битом «ONP» области конфигурации устройства (20FFH) для MC80F0704/0708/0804/ 0808, «ONP» опция bits MASK.

Функция ONP аналогична приведенной ниже.

- Восстановление разрушенной или вызванной потерей колебательной волны

высокочастотным шумом.

- При продолжении высокочастотного шума измените тактовый сигнал системы на тактовый сигнал внутреннего колебания.
- При коротком замыкании или размыкании сигнала XIN/XOUT, остановке основного колебания, за исключением команды stop, и вводе низкочастотного шума измените тактовый сигнал системы на синхросигнал внутренних колебаний.

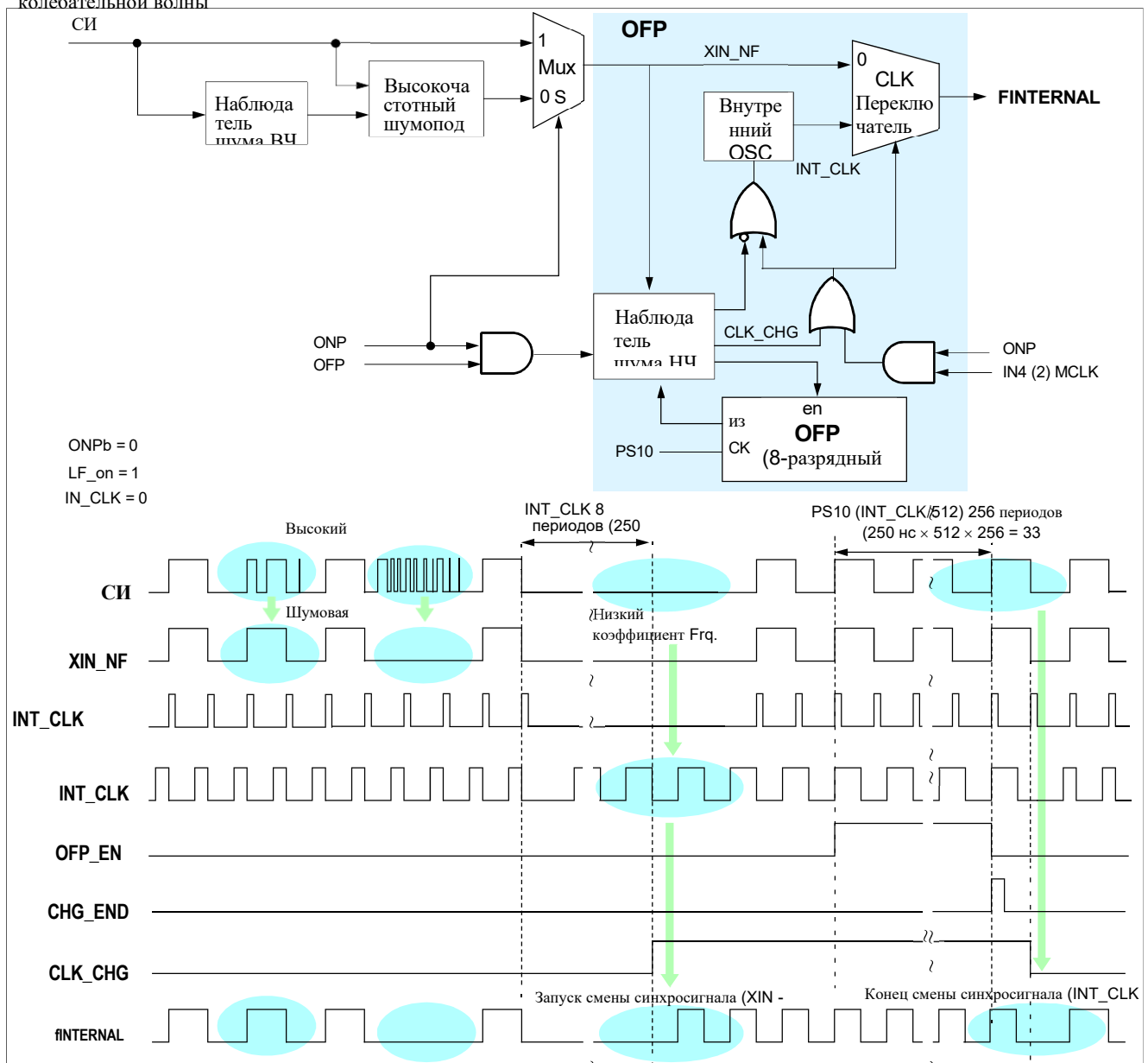


Рисунок 21-1 Блок-схема ONP и OFF и соответствующих форм волн

## 21.2 Процессор ошибок колебаний

Процессор отказов колебаний (OFP) может переключать источник синхросигнала с внешнего на внутренний генератор, когда сбой колебаний происходит. Эта функция может быть включена или отключена битом «OFP» области конфигурации устройства (опция MASK для MC80F0704/0708/0804/0808).

И эта функция может восстанавливать внешний источник синхросигналов, когда внешний синхросигнал восстанавливается до нормального состояния.

### IN4 (2) Опция MCLK/CLKXO (XO)

Бит «IN4MCLK (XO)», «IN2MCLK (XO)» устройства

Configuration Area (опция MASK для MC80F0704/0708/0804/0808 позволяет функции управления устройством с использованием синхросигнала внутреннего генератора в блоке ONP в качестве системного синхросигнала. Нет необходимости соединять X-tal, резонатор, RC и R внешне. Пользователь должен подключить XIN-контакт только к V<sub>DD</sub>. После выбора этой опции период синхросигнала внутреннего генератора можно проверить с помощью выходного синхросигнала XOUT, разделенного на 4.

## 22. Область конфигурации устройства

Область конфигурации устройства может быть запрограммирована или оставлена незапрограммированной для выбора конфигурации устройства, такой как POR, ONP, опция CLK и бит защиты. Эта область недоступна во время обычного выполнения, но доступна для чтения и записи в режиме программы FLASH/проверки.

*Примечание. Параметр Configuration Option может не считываться точно, когда время нарастания VDD очень медленно. Рекомендуется регулировать время нарастания VDD быстрее 40ms/V (200 мс с 0V до 5 В).*

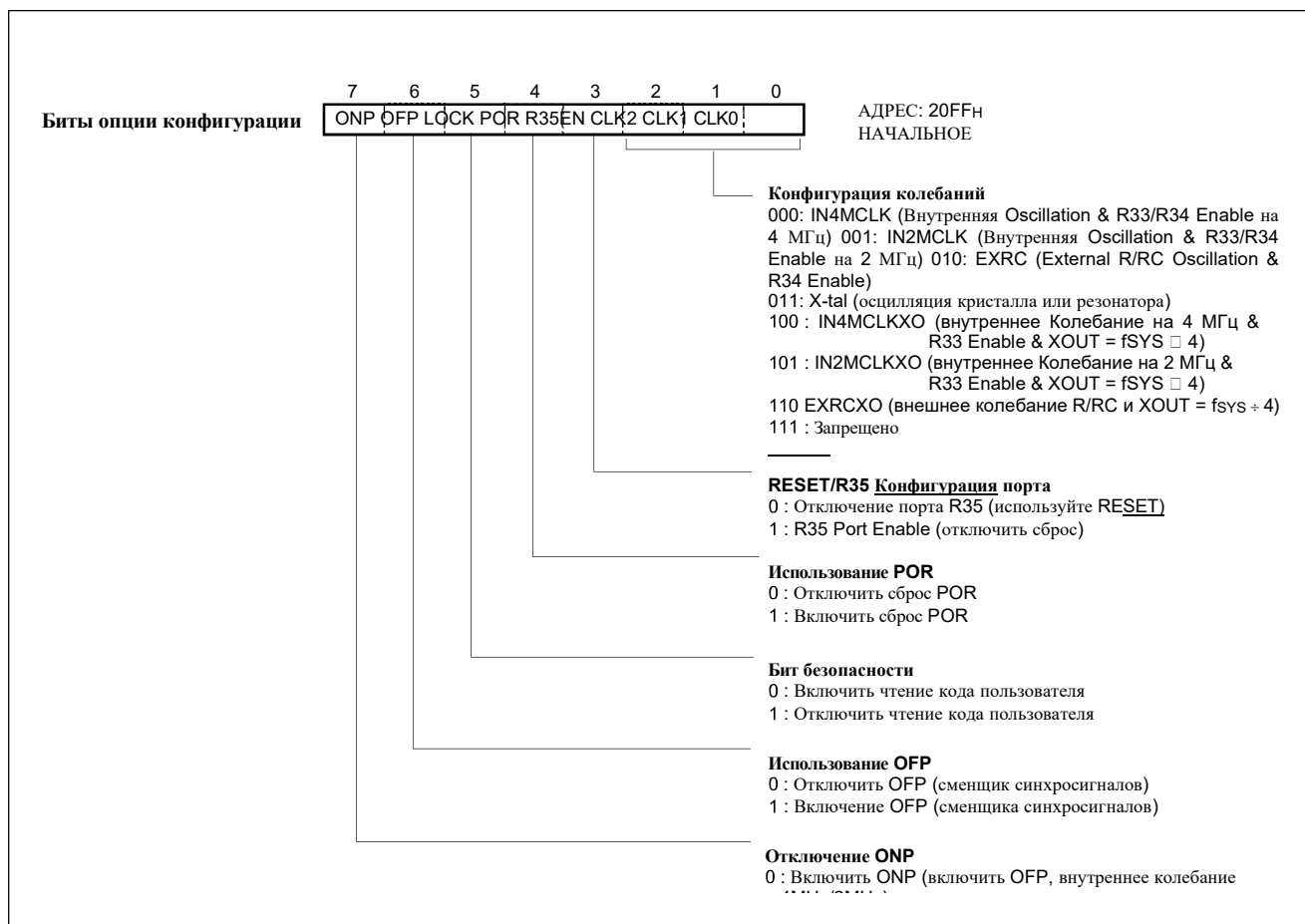
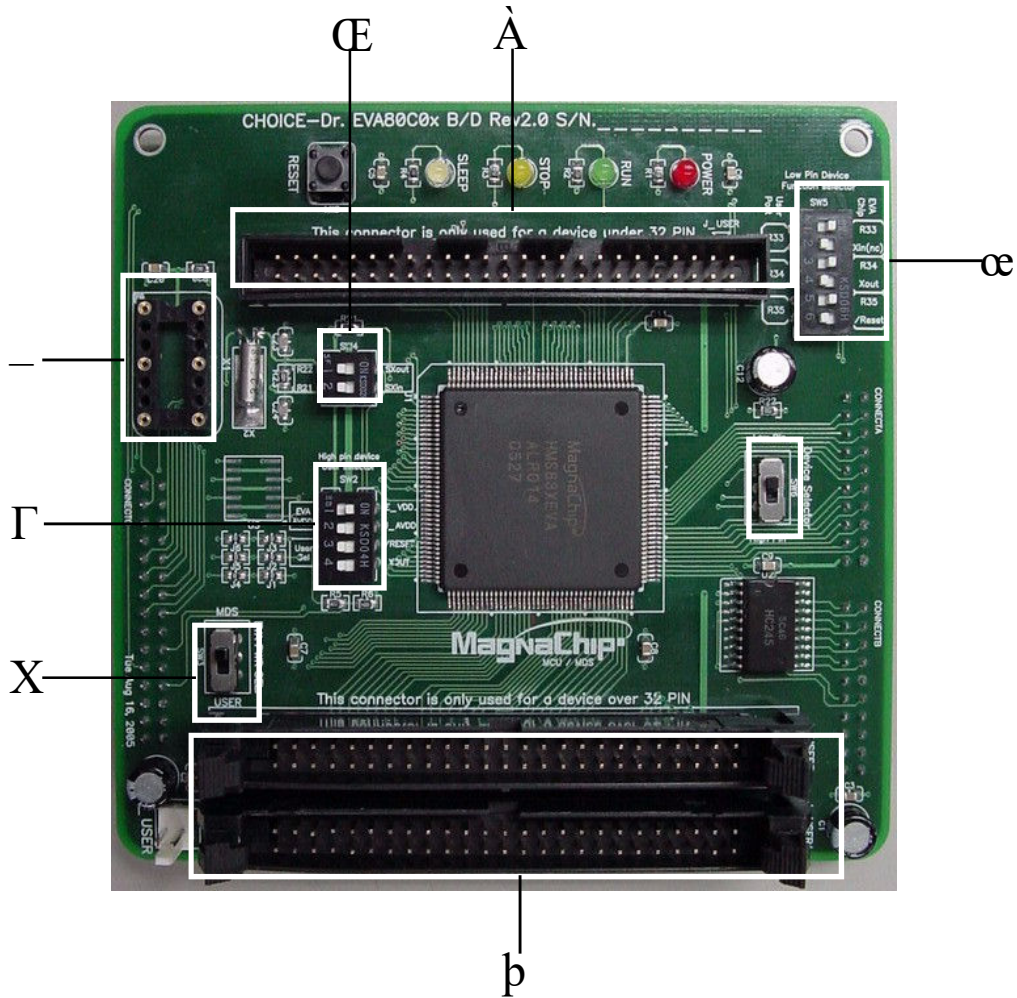


Рис. 22-1 Область конфигурации устройства





### 23. Эмулятор EVA. Настройка платы



## DIP-переключатель и настройка VR

конфигурация

Перед выполнением пользовательской программы помните следующее.

DIP SW	Описание	Установка ВКЛ/ВЫКЛ.
<b>Р</b>	- Этот разъем используется только для устройства с 32-контактным кодом.	Для MC80F0224/MC80F0448
<b>А</b>	- Этот разъем используется только для устройства с 32-контактным кодом.	Для MC80F0708/0808.
<b>Ã</b> SW2	1  НА Eva. выбрать переключатель	Должно быть <b>выключено</b> .  Для MC80F0224/MC80F0448. Для MC80F0708/0808.
	2  <b>ВКЛ.</b> 3  <b>ВЫКЛ.</b>  Используй Eву. VDD пользователя Использовать AVDD	Эти переключатели выбирают источник AVDD.  <b>ВКЛ. И ВЫКЛ.:</b> Использовать Eву. VDD <b>OFF&amp;ON:</b> Использование пользователя AVDD
	4 Этот переключатель выбирает источник/Reset.	Обычно <b>ВЫКЛ.</b> ЭВА. микросхема может быть сброшена внешней платой пользователя tar- get. <b>ON:</b> Reset доступен либо для целевой системной платы пользователя, <b>либо для переключателя Emulator RESET.</b> <b>ВЫКЛ.:</b> Сброс MCU с помощью переключателя RESET эмулятора. Не работает с целевой платы пользователя.
	5 Этот переключатель включает/выключает сигнал Xout.	Обычно <b>ВЫКЛ.</b> Контакт MCU XOUT отключен внутри эмулятора. При некоторых обстоятельствах пользователь может подключить эту схему. <b>ON:</b> Выходной сигнал XOUT <b>OFF:</b> Разъединить цепь
<b>Ö</b> SW3	1  MDS ↑ ПОЛЬЗОВАТЕЛЬ ПОЛЬЗОВАТЕЛЯ  Использование питания MDS	Обычно <b>MDS</b> . Этот переключатель выбирает Eva. B/D Источник питания.
<b>Є</b> SW4	1 Этот переключатель выбирает R22	Эти переключатели выбирают порт нормального ввода-вывода (выкл.) или Sub-Clock (вкл.). Он зарезервирован для MC80F0448. <b>ВКЛ.:</b> SXOUT, SXIN <b>ВЫКЛ.:</b> R22, R21 Не волнуйтесь (MC80F0224/MC80F0448).
	2 Этот переключатель выбирает R21 или SXIN.	

DIP SW		Описание	Установка ВКЛ/ВЫКЛ.
 SW5	1 2	Эти переключатели выбирают R33 или X <sub>IN</sub>	Этот коммутатор выбирает обычный порт ввода/вывода (on & off) или специальную функцию (off & on). Он зарезервирован для MC80F0708/0808. <b>ВКЛ . /ВЫКЛ.:</b> Выбран порт <b>R33, R34, R35. ВЫКЛ. &amp; ВКЛ.:</b> Выбрано <b>X<sub>OUT</sub>, X<sub>IN</sub>, /Reset.</b> Не волнуйтесь (MC80F0224/MC80F0448).
	3 4	Эти переключатели выбирают R34 или X <sub>OUT</sub>	
	5 6	Эти переключатели выбирают R35 или/Reset	
—	-	Это гнездо внешних колебаний (CAN Type. OSC)	Это для внешнего синхросигнала (тип CAN. OSC).

# Приложени

е

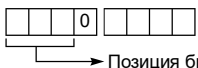
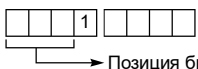
---





## А. ИНСТРУКЦИЯ

### А.1 Терминологический список

Терминология	Описание
A	Аккумулятор
X	X - регистр
Y	Y - регистр
PSW	Слово состояния программы
# imm	8-разрядные немедленные данные
разность потенциалов	Адрес прямого смещения страницы
! абс	Абсолютный адрес
[]	Косвенное выражение
{}	Регистр Косвенное выражение
{ }+	Register Indirect expression, после этого - Register auto-increment.
.bit	Позиция бита
A.bit	Разрядное положение аккумулятора
dp.bit	Битовая позиция памяти прямой страницы
M.bit	Битовая позиция данных памяти (000H ~ 0FFFH)
рэл	Относительная адресация данных
urage	U-page (0FF00H ~ 0FFFFH) Адрес смещения
n	Таблица Номер CALL (0 ~ 15)
+	Дополнение
x	 <p>Верхнее выражение Nibble в Opcode</p>
y	 <p>Верхнее выражение Nibble в Opcode</p>
-	Вычитание
x	Умножение
/	Подразделение
()	Выражение содержимого
^	И
∨	ИЛИ
⊕	Исключительное ИЛИ
~	НЕТ
←	Присвоение/перенос/смена слева
→	Сместить вправо
↔	Обмен
=	Равный
≠	Не равно

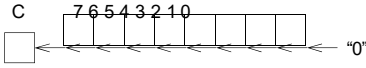
**А.2 Карта инструкций**

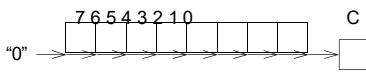
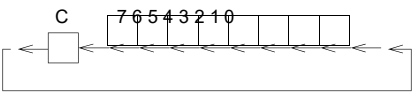
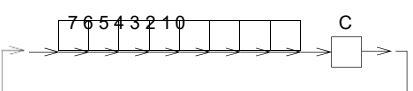
НИЗ КИЙ ВЫСОКИЙ	0000 00	00001 01	00010 02	00011 03	00100 04	00101 05	00110 06	00111 07	01000 08	01001 09	01010 0A	01011 0B	01100 0C	01101 0D	01110 0E	01111 0F
000	-	SET1 dp.bit	BBS A.бит, рель	BBS dp.bit, rel	ADC # imm	ADC разность потенци алов	ADC dp + X	ADC ! абс	ASL A	ASL разность потенци алов	TCALL 0	SETA1 .bit	БИТ разность потенци алов	POP A	PUSH A	КИРПИЧ
001	CLRC	"	"	"	SBC # imm	SBC разность потенци алов	SBC dp + X	SBC ! абс	ROL A	ROL разность потенци алов	TCALL 2	CLRA1 .bit	COM разность потенци алов	POP X	PUSH X	ЛИФЧИК рэл
010	CLRG	"	"	"	CMP # imm	CMP разность потенци алов	CMP dp + X	CMP ! абс	LSR A	LSR разность потенци алов	TCALL 4	NOT1 M.bit	TST разность потенци алов	POP Y	PUSH Y	PCALL Upage
011	DI	"	"	"	ИЛИ # imm	ИЛИ разность потенци алов	ИЛИ dp + X	ИЛИ ! абс	ROR A	ROR разность потенци алов	TCALL 6	OR1 OR1B	CMPX разность потенци алов	POP PSW	PUSH PSW	RET
100	CLRV	"	"	"	И # imm	И разность потенци алов	И dp + X	И ! абс	ИН К А	INC разность потенци алов	TCALL 8	AND1 AND1B	CMPY разность потенци алов	CBNE dp + X	TXSP	INC X
101	SETC	"	"	"	EOR # imm	EOR разность потенци алов	EOR dp + X	EOR ! абс	DEC A	ДЕКАБРЬ разность потенци алов	TCALL 10	EOR1 EOR1B	DBNE разность потенци алов	XMA dp + X	TSPX	DEC X
110	SETG	"	"	"	LDA # imm	LDA разность потенци алов	LDA dp + X	LDA ! абс	TXA	LDY разность потенци алов	TCALL 12	HPC LDCB	LDX разность потенци алов	LDX dp + Y	XCN	DAS (H/D)
111	EI	"	"	"	LDM dp, # imm	СТАНЦИЯ разность потенци алов	СТАНЦИЯ dp + X	СТАНЦИЯ ! абс	НАЛОГ	СВИНАРНИК разность потенци алов	TCALL 14	STC M.bit	STX разность потенци алов	STX dp + Y	XAX	Стоп

НИЗ КИЙ ВЫСОКИЙ	10000 10	10001 11	10010 12	10011 13	10100 14	10101 15	10110 16	10111 17	11000 18	11001 19	11010 1A	11011 1B	11100 1C	11101 1D	11110 1E	11111 1F
000	BPL рэл	CLR1 dp.bit	Би-би- си A.бит, рель	Би-би-си dp.bit, rel	ADC {X}	ADC ! abs + Y	ADC [dp + X]	ADC [dp] + Y	ASL ! абс	ASL dp + X	TCALL 1	JMP ! абс	БИТ ! абс	ADDW разность потенци алов	LDX # imm	JMP [! abs]
001	BVC рэл	"	"	"	SBC {X}	SBC ! abs + Y	SBC [dp + X]	SBC [dp] + Y	ROL ! абс	ROL dp + X	TCALL 3	ЗВОНИТЬ ! абс	ТЕСТ ! абс	SUBW разность потенци алов	LDY # imm	JMP [dp]
010	BCC рэл	"	"	"	CMP {X}	CMP ! abs + Y	CMP [dp + X]	CMP [dp] + Y	LSR ! абс	LSR dp + X	TCALL 5	MUL	TCLR1 ! абс	CMPW разность потенци алов	CMPX # imm	ЗВОНИТЬ [dp]
011	BNE рэл	"	"	"	ИЛИ {X}	ИЛИ ! abs + Y	ИЛИ [dp + X]	ИЛИ [dp] + Y	ROR ! абс	ROR dp + X	TCALL 7	DBNE Y	CMPX ! абс	LDYA разность потенци алов	CMPY # imm	RETI
100	IMT рэл	"	"	"	И {X}	И ! abs + Y	И [dp + X]	И [dp] + Y	INC ! абс	INC dp + X	TCALL 9	ОТДЕЛЕНИЕ	CMPY ! абс	INCW разность потенци алов	ИН К Y	ТЕЙ
101	BVS рэл	"	"	"	EOR {X}	EOR ! abs + Y	EOR [dp + X]	EOR [dp] + Y	ДЕКАБРЬ ! абс	ДЕКАБРЬ dp + X	TCALL 11	XMA {X}	XMA разность потенци алов	DECW разность потенци алов	DEC Y	TYA
110	BSC рэл	"	"	"	LDA {X}	LDA ! abs + Y	LDA [dp + X]	LDA [dp] + Y	LDY ! абс	LDY dp + X	TCALL 13	LDA {X} +	LDX ! абс	STYA разность потенци алов	XAY	ДАА (H/D)
111	BEQ рэл	"	"	"	СТАНЦИЯ {X}	СТАНЦИЯ ! abs + Y	СТАНЦИЯ [dp + X]	СТАНЦИЯ [dp] + Y	СВИНАРНИК ! абс	СВИНАРНИК dp + X	TCALL 15	СТАНЦИЯ {X} +	STX ! абс	CBNE разность потенци алов	XYX	Только для указанн ых целей

### А.3 Набор инструкций

#### Арифметическая/логическая операция

HEX.	МНЕМΟΣХЕМА	OP КОДЕК С	БАЙТ HEX	ЦИКЛ HEX	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	ADC # imm	04	2	2	Добавить с переноской. $A \leftarrow (A) + (M) + C$	NV - H-ZC
2	АЦП dp	05	2	3		
3	АЦП dp + X	06	2	4		
4	ADC! abs	07	3	4		
5	ADC! abs + Y	15	3	5		
6	ADC [dp + X]	16	2	6		
7	ADC [dp] + Y	17	2	6		
8	АЦП {X}	14	1	3		
9	И # имм	84	2	2	ЛОГИЧЕСКОЕ И $A \leftarrow (A) \wedge (M)$	N ---- Z-
10	И dp	85	2	3		
11	И DP + X	86	2	4		
12	И! абы	87	3	4		
13	И! abs + Y	95	3	5		
14	И [dp + X]	96	2	6		
15	И [dp] + Y	97	2	6		
16	И {X}	94	1	3		
17	ASL A	08	1	2	Арифметический сдвиг влево 	N ---- ZC
18	ASL dp	09	2	4		
19	ASL dp + X	19	2	5		
20	ASL! abs	18	3	5		
21	СМР # imm	44	2	2	Сравнение содержимого накопителя с содержимым памяти (A) - (M)	N ---- ZC
22	СМР dp	45	2	3		
23	СМР dp + X	46	2	4		
24	СМР! abs	47	3	4		
25	СМР! abs + Y	55	3	5		
26	СМР [dp + X]	56	2	6		
27	СМР [dp] + Y	57	2	6		
28	СМР {X}	54	1	3		
29	СМРХ # imm	5E	2	2	Сравнение содержимого X с содержимым памяти (X) - (M)	N ---- ZC
30	СМРХ dp	6C	2	3		
31	СМРХ! abs	7C	3	4		
32	СМРУ # imm	7E	2	2	Сравнение содержимого Y с содержимым памяти (Y) - (M)	N ---- ZC
33	СМРУ dp	8C	2	3		
34	СМРУ! abs	9C	3	4		
35	СОМ dp	2C	2	4	1 "S Дополнение: (dp) $\square \sim$ (dp)	N ---- Z-
36	DAA	-	-	-	Неподдержанный	-
37	ДЕСЯТЬ КУБОМЕТРОВ	-	-	-	Неподдержанный	-
38	DEC A	A8	1	2	Декремент $M \leftarrow (M) - 1$	N ---- Z-
39	DEC dp	A9	2	4		
40	DEC dp + X	B9	2	5		
41	DEC! абс	B8	3	5		
42	DEC X	AF	1	2		
43	DEC Y	БЫТЬ	1	2		
44	ОТДЕЛЕНИЕ	9B	1	12		

HEX.	МНЕМΟΣХЕМА	OP КОДЕК С	БАЙТ HEX	ЦИКЛ HEX	ОПЕРАЦИЯ	ФЛАГ NVGBHI ZC
45	EOR # imm	A4	2	2	Исключительное ИЛИ $A \leftarrow (A) \oplus (M)$	N ---- Z-
46	EOR dp	A5	2	3		
47	EOR dp + X	A6	2	4		
48	EOR! abs	A7	3	4		
49	EOR! abs + Y	B5	3	5		
50	EOR [dp + X]	B6	2	6		
51	EOR [dp] + Y	B7	2	6		
52	EOR {X}	B4	1	3		
53	ИНКА	88	1	2	Приращение $M \leftarrow (M) + 1$	N ---- Z-
54	INC dp	89	2	4		
55	INC dp + X	99	2	5		
56	INC! abs	98	3	5		
57	INC X	8F	1	2		
58	ИНК Y	9E	1	2		
59	LSR A	48	1	2	Логическое смещение вправо 	N ---- ZC
60	ЛСП dp	49	2	4		
61	LSR dp + X	59	2	5		
62	LSR! abs	58	3	5		
63	MUL	5B	1	9	Умножение: $YA \leftarrow Y \times A$	N ---- Z-
64	OR # imm	64	2	2	Логическое ИЛИ $A \leftarrow (A) \vee (M)$	N ---- Z-
65	ИЛИ dp	65	2	3		
66	ИЛИ dp + X	66	2	4		
67	ИЛИ! abs	67	3	4		
68	ИЛИ! abs + Y	75	3	5		
69	OR [dp + X]	76	2	6		
70	OR [dp] + Y	77	2	6		
71	ИЛИ {X}	74	1	3		
72	ROL A	28	1	2	Повернуть влево через переноску 	N ---- ZC
73	ROL dp	29	2	4		
74	ROL dp + X	39	2	5		
75	ROL! abs	38	3	5		
76	ROR A	68	1	2	Поворачивать навывлет 	N ---- ZC
77	ROR dp	69	2	4		
78	ROR dp + X	79	2	5		
79	ROR! abs	78	3	5		
80	SBC # imm	24	2	2	Вычитание с переносом $A \leftarrow (A) - (M) - (C)$	NV - HZC
81	СБК dp	25	2	3		
82	SBC dp + X	26	2	4		
83	SBC! abs	27	3	4		
84	SBC! abs + Y	35	3	5		
85	SBC [dp + X]	36	2	6		
86	SBC [dp] + Y	37	2	6		
87	SBC {X}	34	1	3		
88	TST dp	4C	2	3	Проверить содержимое памяти на отрицательное или нулевое значение (dp) - 00 <sub>H</sub>	N ---- Z-
89	XCN	CE	1	5	Обменивайтесь дисками в накопителе $A_7 \sim A_4 \leftrightarrow A_3 \sim A_0$	N ---- Z-

## Регистрация/Работа с памятью

НЕТ.	МНЕМОСХЕМА	ОР КОДЕКС	БАЙТ НЕТ	ЦИКЛ НЕТ	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	LDA # imm	C4	2	2	Аккумулятор нагрузки $A \leftarrow (M)$	N --- Z-
2	LDA dp	C5	2	3		
3	LDA dp + X	C6	2	4		
4	LDA! abs	C7	3	4		
5	LDA! abs + Y	D5	3	5		
6	LDA [dp + X]	D6	2	6		
7	LDA [dp] + Y	D7	2	6		
8	LDA {X}	D4	1	3		
9	LDA {X} +	DB	1	4		
10	LDM dp, # imm	E4	3	5	Загрузить память с немедленными данными: $(M) \leftarrow imm$	-----
11	LDX # imm	1E	2	2	Загрузить X-регистр $X \leftarrow (M)$	N --- Z-
12	LDX dp	Копия	2	3		
13	LDX dp + Y	CD	2	4		
14	LDX! abs	DC	3	4		
15	LDY # imm	3E	2	2	Загрузить Y-регистр $Y \leftarrow (M)$	N --- Z-
16	LDY dp	C9	2	3		
17	LDY dp + X	D9	2	4		
18	LDY! abc	D8	3	4		
19	STA dp	E5	2	4	Хранить содержимое накопителя в памяти $(M) \leftarrow A$	-----
20	STA dp + X	E6	2	5		
21	STA! abs	E7	3	5		
22	STA! abs + Y	F5	3	6		
23	STA [dp + X]	F6	2	7		
24	STA [dp] + Y	F7	2	7		
25	STA {X}	F4	1	4		
26	STA {X} +	FB	1	4	X - автоприращение регистра: $(M) \leftarrow A, X \leftarrow X + 1$	
27	STX dp	EC	2	4	Хранить содержимое X-регистра в памяти $(M) \leftarrow X$	-----
28	STX dp + Y	ED	2	5		
29	STX! abs	ФК	3	5		
30	STY dp	E9	2	4	Сохранить содержимое регистра Y в памяти $(M) \leftarrow Y$	-----
31	STY dp + X	F9	2	5		
32	STY! abs	F8	3	5		
33	НАЛОГ	E8	1	2	Перенос содержимого накопителя в X-регистр: $X \leftarrow A$	N --- Z-
34	ТЕЙ	9F	1	2	Перенос содержимого накопителя в Y-регистр: $Y \leftarrow A$	N --- Z-
35	TSPX	ОДИН	1	2	Передача содержимого стека-указателя в X-регистр: $X \leftarrow sp$	N --- Z-
36	TXA	C8	1	2	Передача содержимого X-регистра в накопитель: $A \leftarrow X$	N --- Z-
37	TXSP	8E	1	2	Передача содержимого X-регистра в указатель стека: $sp \leftarrow X$	N --- Z-
38	TYA	BF	1	2	Передача содержимого регистра Y в накопитель: $A \leftarrow Y$	N --- Z-
39	XAX	EE	1	4	Обмен содержимым X-регистра с накопителем: $X \leftrightarrow A$	-----
40	XAY	DE	1	4	Обмен содержимым Y-регистра с накопителем: $Y \leftrightarrow A$	-----
41	XMA dp	До н.э.	2	5	Обмен содержимым памяти с накопителем $(M) \leftrightarrow A$	N --- Z-
42	XMA dp + X	Н. э.	2	6		
43	XMA {X}	BB	1	5		
44	XYX	FE	1	4		

**16-BIT Эксплуатация**

НЕТ.	МНЕМΟΣХЕМА	ОП КОДЕК С	БАЙТ НЕТ	ЦИКЛ НЕТ	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	ADDW dp	1D	2	5	16-Bits добавить без переноски $YA \leftarrow (YA) + (dp + 1) (dp)$	NV - H-ZC
2	CMPW dp	5D	2	4	Сравните содержимое YA с содержимым пары памяти: $(YA) - (dp + 1) (dp)$	N --- ZC
3	DECW dp	BD	2	6	Уменьшение пары памяти $(dp + 1) (dp) \leftarrow (dp + 1) (dp) - 1$	N --- Z-
4	INCW dp	9D	2	6	Приращение пары памяти $(dp + 1) (dp) \leftarrow (dp + 1) (dp) + 1$	N --- Z-
5	LDYA dp	7D	2	5	Загрузить YA $YA \leftarrow (dp + 1) (dp)$	N --- Z-
6	STYA dp	DD	2	5	Магазин YA $(dp + 1) (dp) \leftarrow YA$	-----
7	SUBW dp	3D	2	5	16-Bits подакт без переноса $YA \leftarrow (YA) - (dp + 1) (dp)$	NV - H-ZC

**Битовая манипуляция**

НЕТ.	МНЕМΟΣХЕМА	ОП КОДЕК С	БАЙТ НЕТ	ЦИКЛ НЕТ	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	AND1 M.bit	8B	3	4	Флаг BIT AND C: $C \leftarrow (C) \wedge (M .bit)$	- ---- C
2	AND1B M.bit	8B	3	4	Флаг BIT AND C и NOT: $C \leftarrow (C) \wedge \sim (M .bit)$	- ---- C
3	BIT dp	0C	2	4	Битовый тест A с памятью: $Z \leftarrow (A) \wedge (M), N \leftarrow (M_7), V \leftarrow (M_6)$	MM -- Z-
4	BIT! abs	1C	3	5		
5	CLR1 dp.bit	y1	2	4	Очистить бит: $(M.bit) \square \langle 0 \rangle$	-----
6	CLRA1 A.bit	2B	2	2	Очистить бит: $(A.bit) \square \langle 0 \rangle$	-----
7	CLRC	20	1	2	Очистить флаг C: $C \square \langle 0 \rangle$	- ---- 0
8	CLRG	40	1	2	Очистить флаг G: $G \square \langle 0 \rangle$	--0-----
9	CLR V	80	1	2	Очистить V-флаг: $V \square \langle 0 \rangle$	-0--0----
10	EOR1 M.bit	AB	3	5	Флаг исключающего ИЛИ бита: $C \leftarrow (C) \oplus (M .bit)$	- ---- C
11	EOR1B M.bit	AB	3	5	Флаг исключающего ИЛИ бита и НЕ: $C \leftarrow (C) \oplus \sim (M .bit)$	- ---- C
12	M.бит HPC	CB	3	4	Флаг загрузки C: $C \leftarrow (M .bit)$	- ---- C
13	M.бит LDCB	CB	3	4	Загрузить флаг C с NOT: $C \leftarrow \sim (M .bit)$	- ---- C
14	NOT1 M.bit	4B	3	5	Битовое дополнение: $(M .bit) \leftarrow \sim (M .bit)$	-----
15	OR1 M.bit	6B	3	5	Флаг BIT OR C: $C \leftarrow (C) \vee (M .bit)$	- ---- C
16	OR1B M.bit	6B	3	5	Флаг BIT OR C и NOT: $C \leftarrow (C) \vee \sim (M .bit)$	- ---- C
17	SET1 dp.bit	x1	2	4	Установить бит: $(M.bit) \square \langle 1 \rangle$	-----
18	SETA1 A.bit	0B	2	2	Установить бит A: $(A.bit) \square \langle 1 \rangle$	-----
19	SETC	A0	1	2	Установить флаг C: $C \square \langle 1 \rangle$	- ---- 1
20	SETG	C0	1	2	Установить флаг G: $G \square \langle 1 \rangle$	--1-----
21	M.bit STC	ЭБ	3	6	Флаг сохранения C: $(M .bit) \leftarrow C$	-----
22	TCLR1! abc	5C	3	6	Тестирование и сброс битов с помощью A: $A - (M), (M) \leftarrow (M) \wedge \sim (A)$	N --- Z-
23	TSET1! abc	3C	3	6	Тестирование и установка битов с помощью A: $A - (M), (M) \leftarrow (M) \vee (A)$	N --- Z-

## Операция перехода/перехода

HEТ.	МНЕМОСХЕМА	OP КОДЕК C	БАЙТ HEТ	ЦИКЛ HEТ	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	Би-би-си A.бит, рель	y2	2	4/6	Ветвь, если бит свободен:	-----
2	BBC dp.bit, rel	y3	3	5/7	если (бит) = 0, то $pc \leftarrow (pc) + rel$	-----
3	BBS A.bit, rel	x2	2	4/6	Ветвь, если установлен бит:	-----
4	BBS dp.bit, rel	x3	3	5/7	если (бит) = 1, то $pc \leftarrow (pc) + rel$	-----
5	BCC rel	50	2	2/4	Ветвь, если бит переноса свободен если (C) = 0, то $pc \leftarrow (pc) + rel$	-----
6	BCS rel	D0	2	2/4	Ветвь, если установлен бит переноса если (C) = 1, то $pc \leftarrow (pc) + rel$	-----
7	BEQ rel	F0	2	2/4	Ветвь, если она равна если (Z) = 1, то $pc \leftarrow (pc) + rel$	-----
8	IMT rel	90	2	2/4	Ветвь, если минус если (N) = 1, то $pc \leftarrow (pc) + rel$	-----
9	BNE rel	70	2	2/4	Ветвь, если она не равна если (Z) = 0, то $pc \leftarrow (pc) + rel$	-----
10	BPL rel	10	2	2/4	Ветвь, если минус если (N) = 0, то $pc \leftarrow (pc) + rel$	-----
11	BRA rel	2F	2	4	Ветвь всегда $pc \leftarrow (pc) + rel$	-----
12	BVC rel	30	2	2/4	Ветвь, если бит переполнения очищен если (V) = 0, то $pc \leftarrow (pc) + rel$	-----
13	BVS rel	B0	2	2/4	Ветвь, если установлен бит переполнения если (V) = 1, то $pc \leftarrow (pc) + rel$	-----
14	CALL! abs	3B	3	8	Вызов подпрограммы	-----
15	ВЫЗОВ [dp]	5F	2	8	$M(sp) \leftarrow (pc_H)$ , $sp \leftarrow sp - 1$ , $M(sp) \leftarrow (pc_L)$ , $sp \leftarrow sp - 1$ , если! abs, $pc \leftarrow abs$ ; если [dp], $pc_L \leftarrow (dp)$ , $pc_H \leftarrow (dp + 1)$ .	-----
16	CBNE dp, rel	FD	3	5/7	Сравнение и ветвление, если они не равны:	-----
17	CBNE dp + X, rel	8D	3	6/8	если (A) $\neq$ (M), то $pc \leftarrow (pc) + rel$ .	-----
18	DBNE dp, rel	AC	3	5/7	Декремент и ответвление если не	-----
19	DBNE Y, rel	7B	2	4/6	равно: если (M) $\neq$ 0, то $pc \leftarrow (pc) + rel$ .	-----
20	JMP! abs	1B	3	3	Безусловный переход $pc \leftarrow$ адрес перехода	-----
21	JMP [! abs]	1F	3	5		
22	JMP [dp]	3F	2	4		
23	Обновление PCALL	4F	2	6	U-пейджинговый вызов $M(sp) \square (pc_H)$ , $sp \leftarrow sp - 1$ , $M(sp) \square (pc_L)$ , $sp \square sp - 1$ , $pc_L \square$ (повышение), $pc_H \square$ «0FFH».	-----
24	TCALL n	nA	1	8	Вызов таблицы: $(sp) \leftarrow (pc_H)$ , $sp \leftarrow sp - 1$ , $M(sp) \leftarrow (pc_L)$ , $sp \leftarrow sp - 1$ , $pc_L \leftarrow$ (вектор таблицы L), $pc_H \leftarrow$ (вектор таблицы H)	-----



**Управление и т.д.**

HEX.	МНЕМΟΣХЕМА	OP КОДЕК С	БАЙТ HEX	ЦИКЛ HEX	ОПЕРАЦИЯ	ФЛАГ NVGBHIZC
1	КИРПИЧ	0F	1	8	Прерывание программного обеспечения: $V \leftarrow \langle 1 \rangle$ , $M(sp) \leftarrow (PC_H)$ , $sp \leftarrow sp - 1$ , $M(s) \leftarrow (PC_L)$ , $sp \leftarrow sp - 1$ , $M(sp) \leftarrow (PSW)$ , $sp \leftarrow sp - 1$ , $PC_L \leftarrow (OFFDE_H)$ , $PC_H \leftarrow (OFFDF_H)$ .	---1-0--
2	DI	60	1	3	Отключить прерывания: $я \leftarrow \langle 0 \rangle$	---- 0--
3	EI	E0	1	3	Включить прерывания: $я \leftarrow \langle 1 \rangle$	---- 1--
4	Только для указанных целей	FF	1	2	Без работы	-----
5	POP A	0D	1	4	$sp \leftarrow sp + 1$ , $A \leftarrow M(sp)$	восстановле нный
6	POP X	2D	1	4	$sp \leftarrow sp + 1$ , $X \leftarrow M(sp)$	
7	POP Y	4D	1	4	$sp \leftarrow sp + 1$ , $Y \leftarrow M(sp)$	
8	POP PSW	6D	1	4	$sp \leftarrow sp + 1$ , $PSW \leftarrow M(sp)$	
9	PUSH A	0E	1	4	$M(sp) \leftarrow A$ , $sp \leftarrow sp - 1$	-----
10	PUSH X	2E	1	4	$M(sp) \leftarrow X$ , $sp \leftarrow sp - 1$	
11	PUSH Y	4E	1	4	$M(sp) \leftarrow Y$ , $sp \leftarrow sp - 1$	
12	PUSH PSW	6E	1	4	$M(sp) \leftarrow PSW$ , $sp \leftarrow sp - 1$	
13	RET	6F	1	5	Возврат из подпрограммы $sp \leftarrow sp + 1$ , $PC_L \leftarrow M(sp)$ , $sp \leftarrow sp + 1$ , $PC_H \leftarrow M(sp)$	-----
14	RETI	7F	1	6	Возврат из прерывания $sp \leftarrow sp + 1$ , $PSW \leftarrow M(sp)$ , $sp \leftarrow sp + 1$ , $PC_L \leftarrow M(sp)$ , $sp \leftarrow sp + 1$ , $PC_H \leftarrow M(sp)$	восстановле нный
15	Стоп	EF	1	3	Режим остановки (остановка CPU, остановка генератора)	-----

