

Главы 3, 4, 8, 9, 11, 14, фирменного описания, в которых описаны основные отличительные особенности контроллера PIC16F690.

Черновой перевод на русский осуществил Валентин Володин.

### 3.0 Модуль генератора (с монитором защиты при аварии тактирования)

#### 3.1 Обзор

Генератор имеет широкий ряд источников тактовых частот и особенности выбора, которые позволяют использовать его в широком диапазоне прикладных программ при максимальной эффективности и минимизации потребляемой мощности. На рис. 3-1 показана блок-схема модуля осциллятора.

Тактовый источник может быть сконфигурирован от внешних генераторов, кварцевого кристаллического резонатора, керамического резонатора и RC схемы. Вдобавок, системный тактовый источник может быть сконфигурирован от одного из двух внутренних генераторов, с выбором скоростей через программное обеспечение. Дополнительный тактовый генератор включает характеристики:

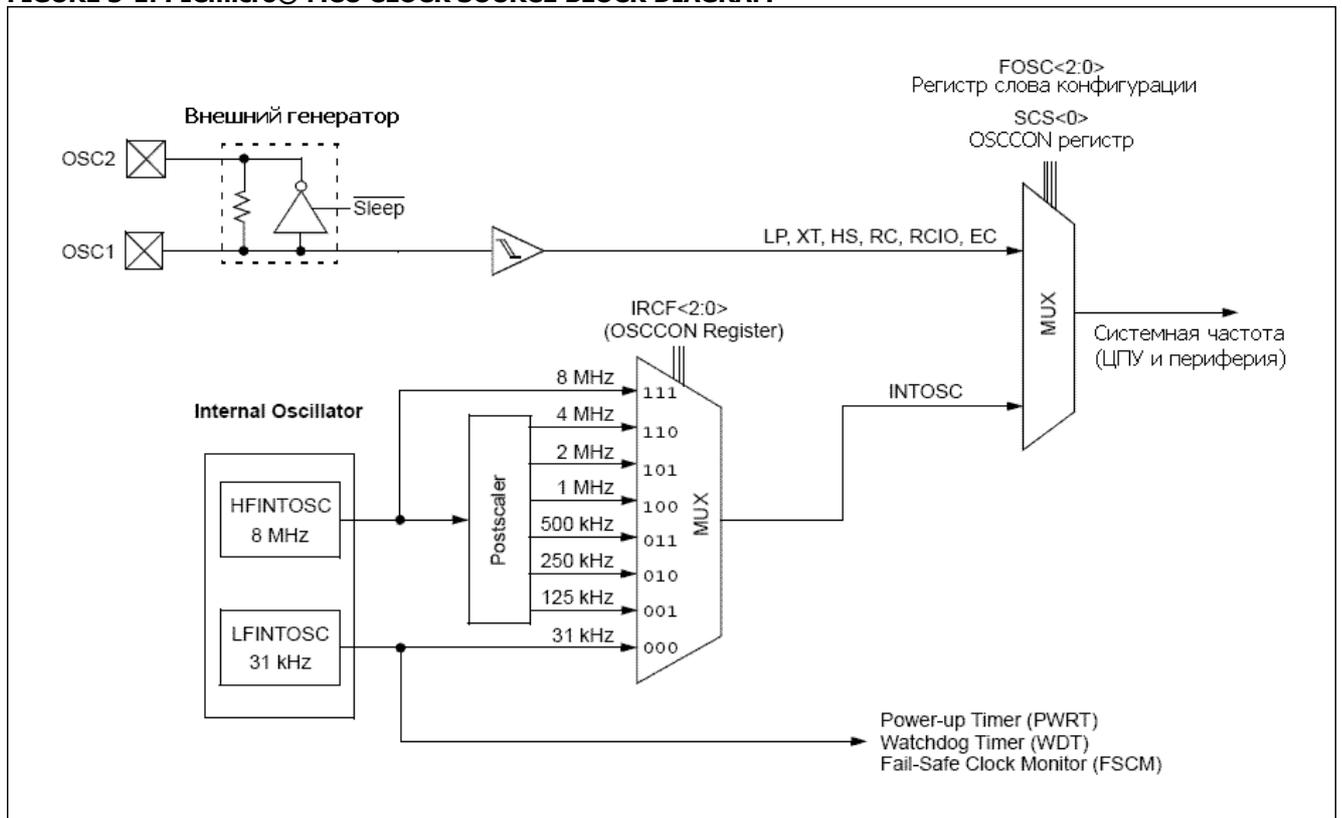
- Программный выбор системного тактового источника между внешним и внутренним.
- Двухскоростной режим старта, что минимизирует время ожидания между стартом внешнего генератора и выполнением кода.
- Fail-Safe Clock Monitor (FSCM) – монитор защиты при аварии тактирования разработанный для обнаружения аварии внешнего тактового источника (LP, XT, HS, EC и RC режимы) и автоматическое переключение на внутренний генератор.

Модуль генератора может быть сконфигурирован в один из восьми режимов тактирования.

1. EC – Внешнее тактирование с I/O на OSC2/CLKOUT.
2. LP – 32 kHz режим мало потребления.
3. XT – Режим генератора для среднечастотного кристалла или керамического резонатора.
4. HS – Режим высокочастотного кристалла или керамического резонатора.
5. RC – Внешние резистор-конденсатор (RC) с FOSC/4 выходом на OSC2/CLKOUT.
6. RCIO – Внешний резистор-конденсатор (RC) с I/O на OSC2/CLKOUT.
7. INTOSC – Внутренний генератор с FOSC/4 выходом на OSC1/CLKIN.
8. INTOSCIO – Внутренний генератор с I/O на OSC1/CLKIN и OSC2/CLKOUT.

Режимы тактового источника конфигурируются битами FOSC<2:0> в регистре слова конфигурации (CONFIG). Внутреннее тактирование может генерироваться двумя внутренними тактовыми источниками. HFINTOSC – это калибруемый высокочастотный генератор. LFINTOSC – это не калибруемый низкочастотный генератор.

**FIGURE 3-1: PICmicro® MCU CLOCK SOURCE BLOCK DIAGRAM**



### 3.2 Управление генератором

Регистр управления генератором (OSCCON), управляет системным тактированием и опциями выбора частоты. Регистр OSCCON содержит следующие биты:

- Биты выбора частоты (IRCF)
- Биты состояния частоты (HTS, LTS)
- Биты управления системным тактированием (OSTS, SCS)

#### Регистр 3-1: OSCCON: Регистр управления генератором

U-0	R/W-1	R/W-1	R/W-0	R-1	R-0	R-0	R/W-0
-	IRCF2	IRCF1	IRCF0	OSTS <sup>(1)</sup>	HTS	LTS	SCS
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b> R=бит читается -n=Значение при POR	W=бит записывается '1'=бит установлен	U=не задействованный бит, читается как '0' '0'=бит очищен	X=бит не определён
---	--	--	--------------------

**Бит 7** **Незадействован:** Читается как '0'

**Биты 6-4** **IRCF<2:0>:** Биты выбора частоты внутреннего генератора

111 = 8MHz  
110 = 4 MHz (по умолчанию)  
101 = 2MHz  
100 = 1MHz  
011 = 500 kHz  
010 = 250 kHz  
001 = 125 kHz  
000 = 31 kHz (LFINTOSC)

**Бит 3** **OSTS:** Бит<sup>(1)</sup> состояния превышения времени запуска генератора

1 = Устройство запущено с внешним тактовым источником определённым битами FOSC<2:0> регистра CONFIG

0 = Устройство запущено с внутренним генератором (HFINTOSC или LFINTOSC)

**Бит 2** **HTS:** Бит состояния HFINTOSC (Высокая частота – от 8 MHz до 125 kHz)

1 = HFINTOSC стабильна

0 = HFINTOSC не стабильна

**Бит 1** **LTS:** Бит стабильности LFINTOSC (Низкая частота – 31 kHz)

1 = LFINTOSC стабильна

0 = LFINTOSC не стабильна

**Бит 0** **SCS:** Бит выбора системного тактового источника

1 = Внутренний генератор используется для системного тактирования

0 = Тактовый источник определён битами FOSC<2:0> регистра CONFIG

**Примечание 1:** Бит сброшен в '0' с двухскоростным запуском и выбранным LP, XT или HS режимом генератора или разрешён отказоустойчивый режим.

### 3.3 Режимы источника тактирования

Режимы тактирования могут быть классифицированы как внешние или внутренние.

- Режимы внешнего тактирования полагаются на внешнюю схему тактового источника. Это могут быть: Генераторные модули (EC режим), кварцевые резонаторы или керамические резонаторы (LP, XT и HS режимы) и резистор-конденсатор (RC) режим схемы.
- Внутренние тактовые источники находятся непосредственно в пределах модуля генератора. Модуль генератора содержит два внутренних генератора: the 8 MHz высокочастотный внутренний генератор (HFINTOSC) и 31 kHz низкочастотный внутренний генератор (LFINTOSC).

Системное тактирование может быть выбрано между внешним или внутренним тактовыми источниками, через бит выбора системного тактирования (SCS) регистра OSCCON. Смотрите **Часть 3.6 "Переключение тактирования"** для дополнительной информации.

### 3.4 Режимы внешнего тактирования

#### 3.4.1 Таймер запуска генератор (OST)

Если модуль генератора сконфигурирован для режимов LP, XT или HS, таймер запуска генератора (OST) считает 1024 генерации от OSC1. Это происходит после сброса после включения (POR) и когда таймер включения питания (PWRT) истёк (если конфигурирован), или после пробуждения из спящего режима. В течении этого времени программный счётчик не увеличивается и программа приостановлена. OST гарантирует, что схема генератора, использующая кварцевый или керамический резонатор, стартует и обеспечит стабильное системное тактирование для модуля генератора. Когда происходит переключение между двумя тактовыми источниками, требуется задержка позволяющая стабилизироваться новому тактовому источнику. Эти задержки генератора показаны в таблице 3-1.

В порядке минимизации ожидания между запуском генератора и выполнением кода, может быть выбран двухскоростной режим запуска (смотрите **Часть 3.7 "Двухскоростной режим запуска"**).

**TABLE 3-1: OSCILLATOR DELAY EXAMPLES**

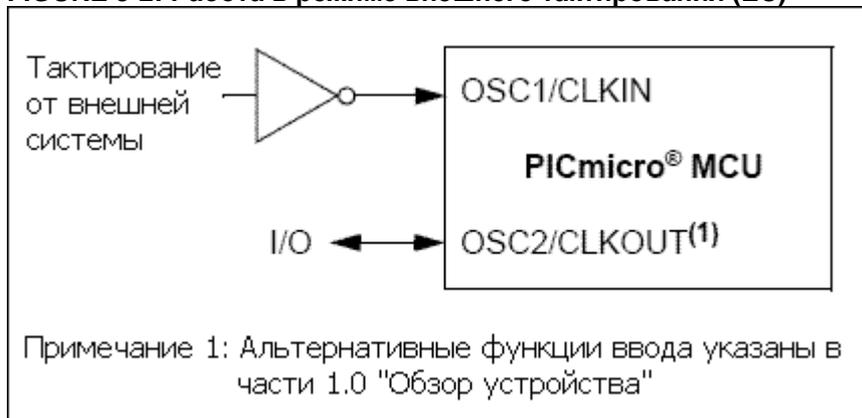
Включен от	Включен в	Частота	Задержка генератора
Sleep/POR	LFINTOSC HFINTOSC	31 kHz 125 kHz to 8 MHz	Генератора тёплая задержка (TWARM)
Sleep/POR	EC, RC	DC – 20 MHz	2 цикла
LFINTOSC (31 kHz)	EC, RC	DC – 20 MHz	1 цикл каждого
Sleep/POR	LP, XT, HS	32 kHz to 20 MHz	1024 тактовых цикла (OST)
LFINTOSC (31 kHz)	HFINTOSC	125 kHz to 8 MHz	1 $\mu$ s (примерно.)

#### 3.4.2 EC MODE

Режим внешнего тактирования (EC) позволяет внешнему генерировать логический уровень как источник системного тактирования. Когда действует этот режим, внешний источник тактирования подключен к OSC1 входу и OSC2 доступен для общего I/O. Рисунок 3-2 показывает подключение ножек для EC режима.

Таймер запуска генератора (OST) заблокирован, когда выбран режим EC. Следовательно, отсутствует задержка в работе после сброса по включению питания или пробуждения из спящего режима. Поскольку ЦПУ PICmicro® полностью статический, остановка внешнего тактирования может иметь эффект остановки устройства, при этом все данные остаются неповреждёнными. С восстановлением внешнего тактирования устройство возобновит работу, как ни в чём не бывало.

**FIGURE 3-2: Работа в режиме внешнего тактирования (EC)**



### 3.4.3 Режимы LP, XT, HS

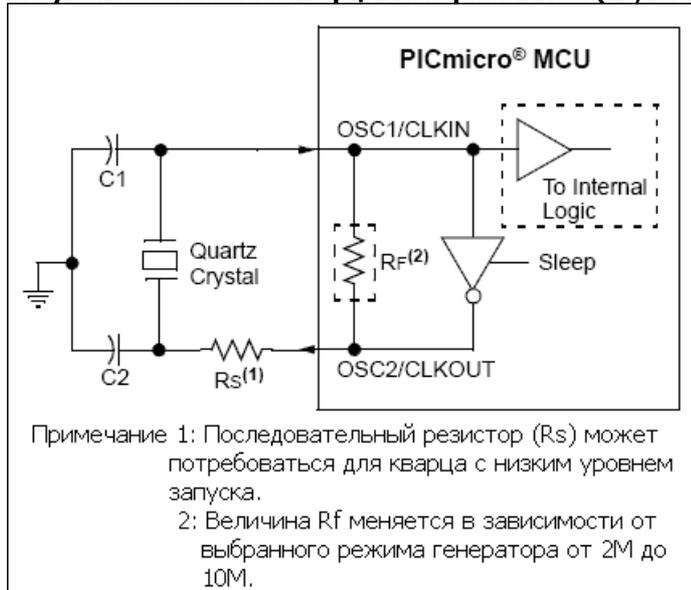
Режимы LP, XT и HS поддерживают использование кварцевых кристаллических резонаторов или керамических резонаторов подключенных к выводам OSC1 и OSC2 (рисунок 3-3). Режим выбирает низкое, среднее или высокое усиление внутреннего инвертирующего усилителя для поддержки различных типов и скоростей резонаторов.

**LP** режим генератора выбирает низкое усиление внутреннего инвертирующего усилителя. LP режим имеет наименьшее потребление тока из всех трёх режимов. Этот режим разрабатывался только для 32.768 kHz часовых кристаллов.

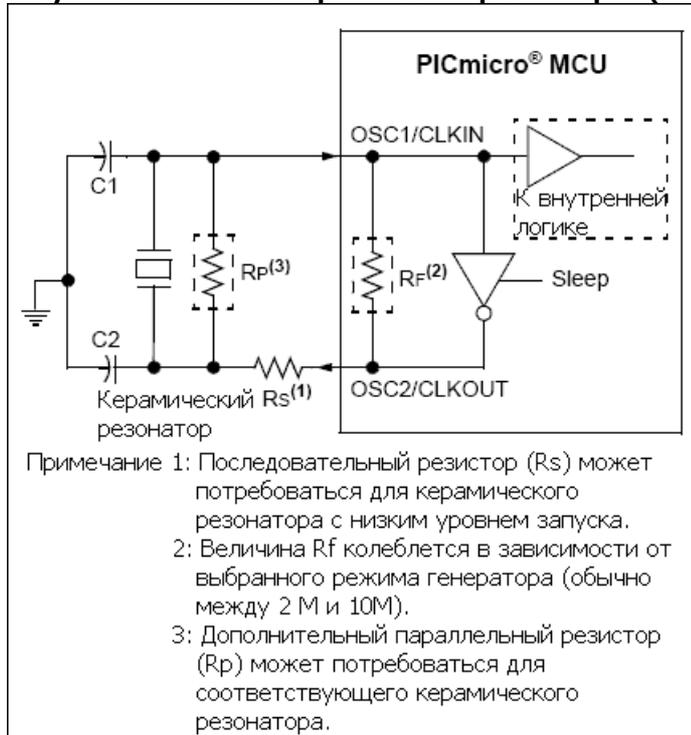
**XT** режим генератора выбирает промежуточное усиление внутреннего инвертирующего усилителя. XT режим имеет среднее потребление тока из всех трёх режимов. Этот режим лучше подходит для резонаторов со средним уровнем запуска.

**HS** режим генератора выбирает высокое усиление внутреннего инвертирующего усилителя. HS режим имеет максимальное потребление тока среди трёх режимов. Этот режим подходит для резонаторов, которые требуют высокие установки запуска. На рис. 3-3 и рис. 3-4 показаны типичные схемы для кварцевых кристаллов и керамических резонаторов соответственно.

**Рисунок 3-3: Работа с кварцевым кристаллом (LP, XT или HS режим)**



**Рисунок 3-4: Работа с керамическим резонатором (XT или HS режим)**



**Примечание**

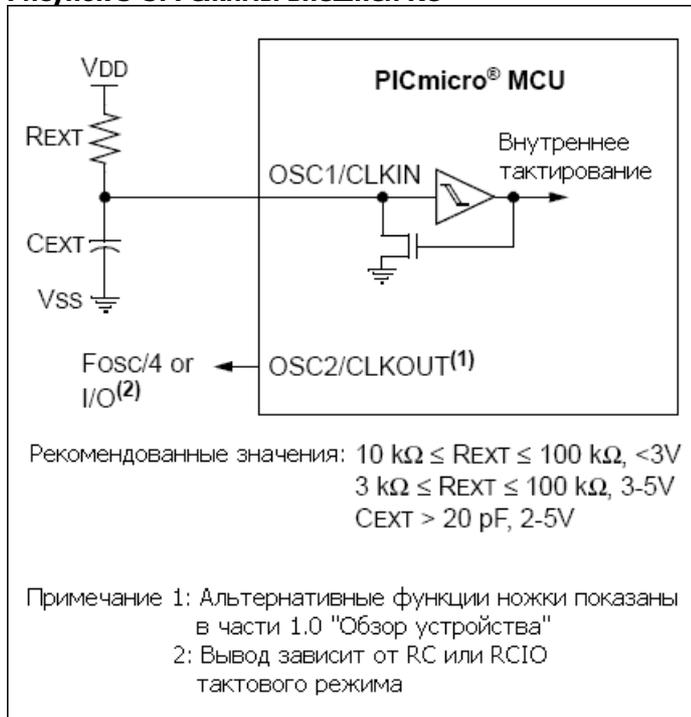
- 1: Характеристики кварцевого кристалла меняются в зависимости от типа, упаковки и производителя. Пользователь должен руководствоваться рекомендациями по применению от производителя.
- 2: Всегда проверять исполнение генератора по температурному диапазону и напряжению, которые ожидаются в приложении.
- 3: Для помощи при разработке генератора обращайтесь к следующим руководствам от Microchip:
  - AN826, "Основы кристаллического генератора и выбор кристалла для устройств *µPIC®* и *PICmicro®*" (DS00826)
  - AN849, "Основа дизайна генератора *PICmicro®*" (DS00849)
  - AN943, "Анализ и разработка практического *PICmicro®* генератора" (DS00943)
  - AN949, "Сделайте ваш генератор рабочим" (DS00949)

### 3.4.4 Режимы внешней RC

Режимы с внешними резистор-конденсатор (RC) поддерживают использование внешней RC схемы. Это позволяет разработку максимально гибкие в выборе частоты при минимальной стоимости, когда не требуется высокая точность тактирования. Там есть два режима: RC и RCIO.

В режиме RC, RC схема подключается к OSC1. На OSC2/CLKOUT выходе присутствует делённая на 4 частота RC генератора. Этот сигнал может использоваться для тактирования внешних схем, синхронизации, калибровки, тестирования и других прикладных функций. На рис. 3-5 показан режим подключенной внешней RC цепочки.

**Рисунок 3-5: Режимы внешней RC**



В режиме RCIO RC схема подключается к OSC1. OSC2 становится дополнительной ножкой I/O.

Частота RC генератора есть функция напряжения питания, величин сопротивления ( $R_{EXT}$ ) и конденсатора ( $C_{EXT}$ ) и рабочей температуры. Другими факторами, влияющими на частоту генератора, являются:

- изменение порогового напряжения
- разбросы компонентов
- упаковочные изменения в ёмкости

Пользователю так же необходимо принять во внимание изменения, вызванные разбросом используемых внешних RC компонентов.

## 3.5 Режимы внутреннего тактирования

Модуль генератора имеет два независимых внутренних генератора, которые могут быть конфигурированы или выбраны как источники системного тактирования.

1. **HFINTOSC** (высокочастотный внутренний генератор) калиброван при производстве и работает на частоте 8 MHz. Частота HFINTOSC может быть программно подстроена пользователем с использованием регистра OSCTUNE (регистр 3-2).

2. **LFINTOSC** (низкочастотный внутренний генератор) не калибруется и работает на частоте 31 kHz.

Скорость системного тактирования может быть выбрана программно, используя биты выбора частоты внутреннего генератора IRCF<2:0> регистра OSCCON.

Системное тактирование может быть выбрано между внешним или внутренним источником тактирования через бит выбора системного тактирования (SCS) регистра OSCCON. Смотреть **Часть 3.6 "Переключение тактирования"** для большей информации.

### 3.5.1 INTOSC и INTOSCIO режимы

INTOSC и INTOSCIO режимы конфигурации внутренних генераторов, как источника системного тактирования, когда устройство запрограммировано, используя выбор генератора или биты FOSC<2:0> в регистре слова конфигурации (CONFIG).

В режиме **INTOSC**, OSC1/CLKIN доступен для I/O. OSC2/CLKOUT выходы выбраны делителем на 4 частоты внутреннего генератора. Сигнал CLKOUT может быть использован для обеспечения тактирования для внешних схем, синхронизации, калибровки, тестирования и других прикладных требований.

В режиме **INTOSCIO**, OSC1/CLKIN и OSC2/CLKOUT доступны для общего I/O.

### 3.5.2 HFINTOSC

Высокочастотный внутренний генератор (HFINTOSC), настроенный производителем на частоту 8 MHz, является внутренним тактовым источником. Частота HFINTOSC может быть изменена программой, используя регистр OSCTUNE (регистр 3-2).

Выход HFINTOSC подключен к делителю и мультиплексору (смотреть рисунок 3-1). Одна из семи частот может быть выбрана программно, используя биты IRCF<2:0> регистра OSCCON. Смотрите **Часть 3.5.4 "Биты выбора частоты (IRCF)"** для большей информации.

HFINTOSC разрешается выбором любой частоты между 8 MHz и 125 kHz установкой битов IRCF<2:0> регистра OSCCON ≠ 000. Затем установить в '1' бит источника системного тактирования (SCS) регистра OSCCON или разрешить двухскоростной запуск установив в '1' бит IESO в регистре слова конфигурации.

Бит HF внутреннего осциллятора (HTS) регистра OSCCON показывает, HFINTOSC стабилен или нет.

### 3.5.2.1 Регистр OSCTUNE

HFINTOSC калиброван при производстве, но может быть подстроен в программе, через запись в регистр OSCTUNE (регистр 3-2). По умолчанию значение регистра OSCTUNE равно '0'. Значение есть 5-ти битное двоичное число в дополнительном коде.

#### REGISTER 3-2: OSCTUNE: OSCILLATOR TUNING REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
-	-	-	TUN4	TUN3	TUN2	TUN1	TUN0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-5 Не задействованы:** Читаются как '0'

**Биты 4-0 TUN<4:0>:** Биты подстройки частоты

01111 = Максимальная частота

01110 =

•

•

•

00001 =

00000 = Модуль генератора запущен на калиброванной частоте.

11111 =

•

•

•

10000 = Минимальная частота

Когда регистр OSCTUNE модифицирован, частота HFINTOSC может начать сдвигаться на новую частоту. Выполнение кода продолжается в течении этого сдвига. Там не указывается, что происходит сдвиг.

OSCTUNE не влияет на частоту LFINTOSC. От тактовой частоты LFINTOSC зависит работа таймера включения питания (PWRT), сторожевого таймера (WDT), монитора защиты от ошибки тактирования (FSCM) и периферии, которые не зависят от изменения частоты.

### 3.5.3 LFINTOSC

Низкочастотный внутренний генератор (LFINTOSC) есть не калиброванный внутренний тактовый источник частотой 31kHz. Выход LFINTOSC подключен к постделителю и мультиплексору (смотреть рисунок 3-1). Выбирается генератор 31 kHz программно, используя биты IRCF<2:0> регистра OSCCON. Смотрите **Часть 3.5.4 "Биты выбора частоты (IRCF)"** для большей информации. LFINTOSC так же формирует частоту для таймера включения питания (PWRT), сторожевого таймера (WDT) и монитора защиты от аварии тактового генератора (FSCM).

LFINTOSC разрешен выбором частоты 31 kHz битами (IRCF<2:0> = 000 регистра OSCCON) как системный источник тактирования (SCS бит регистра OSCCON = 1), или когда любое из следующего разрешено:

- Бит IESO = 1 двухчастотного запуска регистра слова конфигурации и биты IRCF<2:0> = 000 регистра OSCCON
- Таймер включения питания (PWRT)
- Сторожевой таймер (WDT)
- Монитор защиты от аварии тактирования (FSCM)

Бит LF внутреннего генератора (LTS) регистра OSCCON показывает независимо, LFINTOSC стабилен или нет.

### 3.5.4 Биты выбора частоты (IRCF)

Выходы 8 MHz HFINTOSC и 31 kHz LFINTOSC подключены к делителю и мультиплексору (смотреть рисунок 3-1). Биты IRCF<2:0> выбора частоты внутреннего генератора регистра OSCCON выбирают выходную частоту внутреннего генератора. Одна из восьми частот может быть выбрана программно:

- 8 MHz
- 4 MHz (по умолчанию после сброса)
- 2 MHz
- 1 MHz
- 500 kHz
- 250 kHz
- 125 kHz
- 31 kHz (LFINTOSC)

**Примечание:** Любой последующий сброс устанавливает биты IRCF<2:0> регистра OSCCON в '110' и выбирается частота 4 MHz. Пользователь может изменить биты IRCF для выбора другой частоты.

### 3.5.5 Синхронизация переключения HFINTOSC и LFINTOSC генераторов

Когда происходит переключение между LFINTOSC и HFINTOSC, новый генератор уже может быть выключен для сохранения энергии (смотреть рисунок 3-6). Если дело обстоит так, там имеется задержка после модификации битов IRCF<2:0> регистра OSCCON, прежде чем изменение частоты будет иметь место. Биты LTS и HTS регистра OSCCON отражают текущее состояние активности генераторов LFINTOSC и HFINTOSC. Переключение частоты происходит в последовательности:

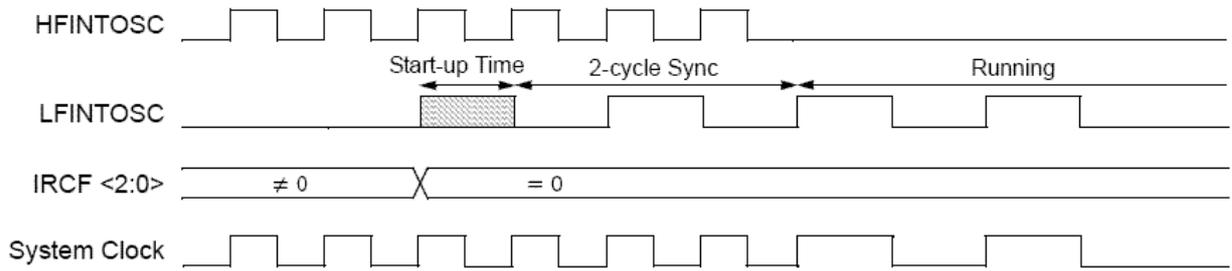
1. Модифицировать биты IRCF<2:0> регистра OSCCON.
2. Если новая тактовая частота была отключена, тактирование стартует с задержкой.
3. Схема переключения тактирования ожидает срез текущего тактового импульса.
4. CLKOUT удерживается на низком уровне и схема переключения ждёт нарастающего фронта нового тактового сигнала.
5. CLKOUT теперь подключен к новому тактированию. Биты LTS и HTS регистра OSCCON будут добавлены, как требуется.
6. Переключение тактирования завершено.

Смотреть рисунок 3-1 для больших подробностей.

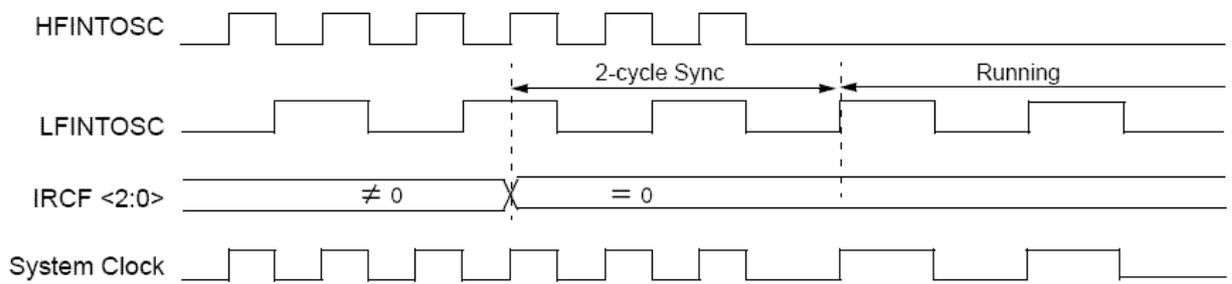
Если внутренний генератор выбран между 8 MHz и 125 kHz, то не будет задержки между выбором новой частоты. Это потому, что старая и новые частоты генерируются одним генератором HFINTOSC через делитель и мультиплексор. Спецификации задержки запуска расположены в таблицах генератора в **Части 17.0 "Электрических спецификаций"**.

**Рисунок 3-6: Переключение внутренних генераторов**

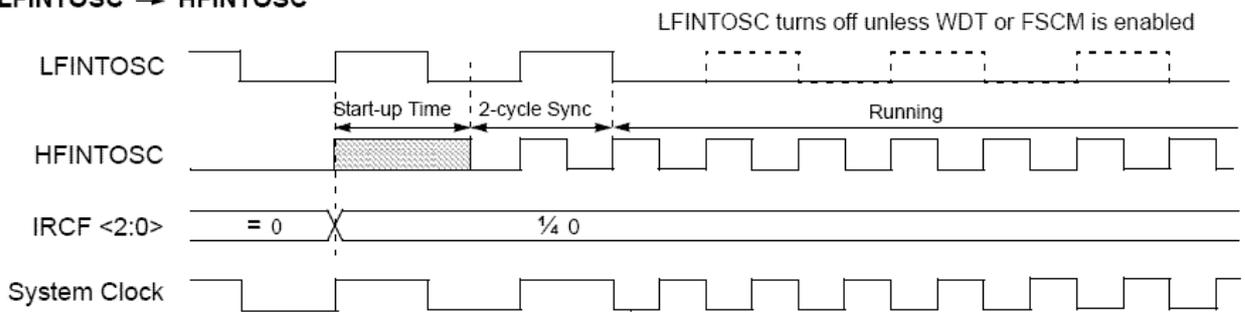
**HFINTOSC → LFINTOSC (FSCM and WDT disabled)**



**HFINTOSC → LFINTOSC (Either FSCM or WDT enabled)**



**LFINTOSC → HFINTOSC**



## 3.6 Clock Switching

Источник системного тактирования может быть переключен между внешним и внутренним тактовыми источниками через программу, используя бит выбора системного тактирования (SCS) регистра OSCCON.

### 3.6.1 Бит выбора системного тактирования (SCS)

Бит выбора системного тактирования (SCS) регистра OSCCON выбирает источник системного тактирования который будет использован для ЦПУ и периферии.

- Когда бит SCS регистра OSCCON равен 0, системный тактовый источник определяется конфигурацией битов FOSC<2:0> в регистре слова конфигурации (CONFIG).
- Когда бит SCS регистра OSCCON равен 1, источник системного тактирования выбран внутренний генератор частоты выбранной битами IRCF<2:0> регистра OSCCON. После сброса бит SCS регистра OSCCON всегда сброшен.

**Примечание:** Любое автоматическое переключение тактирования, что может происходить из двухскоростного запуска или монитор защиты по аварии тактирования, не корректируют бит SCS регистра OSCCON. Пользователь может проверить бит OSTS регистра OSCCON для определения текущего системного тактового источника.

### 3.6.2 Бит состояния превышения времени запуска генератора (OSTS)

Бит превышения времени запуска генератора (OSTS) регистра OSCCON показывает системное тактирование запущено от внешнего источника тактирования, как это определено битами FOSC<2:0> в регистре слова конфигурации (CONFIG), или от внутреннего тактового источника. В частности, OSTS показывает, что таймер запуска генератора (OST) засёк превышение времени для режимов LP, XT или HS.

## 3.7 Режим двухскоростного запуска

Режим двухскоростного запуска обеспечивает дополнительное сбережение энергии для минимизации времени ожидания между запуском внешнего генератора и выполнением кода. В приложениях которые используют спящий режим, двухскоростной запуск удаляет время запуска внешнего генератора из времени, затраченного на пробуждение и может уменьшить мощность потребляемую устройством. Этот режим позволяет приложению, просыпаясь из спящего режима, выполнить несколько инструкций, используя INTOSC как тактовый источник, и идти назад в спящий режим без ожидания, когда первичный генератор стабилизируется.

**Примечание:** Выполнение инструкции SLEEP обрывает запуск генератора и заставляет бит OSTS регистра OSCCON оставаться открытым.

Когда модуль осциллятора сконфигурирован для режимов LP, XT или HS, таймер запуска генератора (OST) разрешён (смотреть **Часть 3.4.1 "Таймер запуска генератора (OST)"**). OST может приостанавливать выполнение программы, пока не сосчитает 1024 осцилляций. Двухскоростной режим запуска минимизирует задержку в выполнении кода, работая от внешнего генератора пока OST считает. Когда OST счётчик достигнет 1024 и бит OSTS регистра OSCCON будет установлен, программа выполнит переключение на внешний генератор.

### 3.7.1 TWO-SPEED START-UP MODE CONFIGURATION

Двухскоростной режим запуска конфигурируется следующими установками:

- IESO (регистра слова конфигурации) = 1; бит переключения Внутренний/Внешний (двухскоростной режим запуска разрешён).
- SCS (регистра OSCCONr) = 0.
- FOSC<2:0> биты в регистре слова конфигурации (CONFIG) конфигурированы для LP, XT или HS режима.

Двухскоростной режим запуска вводится после:

- Сброса по включению питания (POR) и, если разрешено, после истечения таймера включения питания (PWRT), или
- Пробуждения из спящего режима.

Если внешний тактовый генератор сконфигурирован быть отличным от LP, XT или HS режима, затем двухскоростной режим запуска заблокирован. Это потому, что внешний тактовый генератор не требует времени стабилизации после POR или выхода из спящего режима.

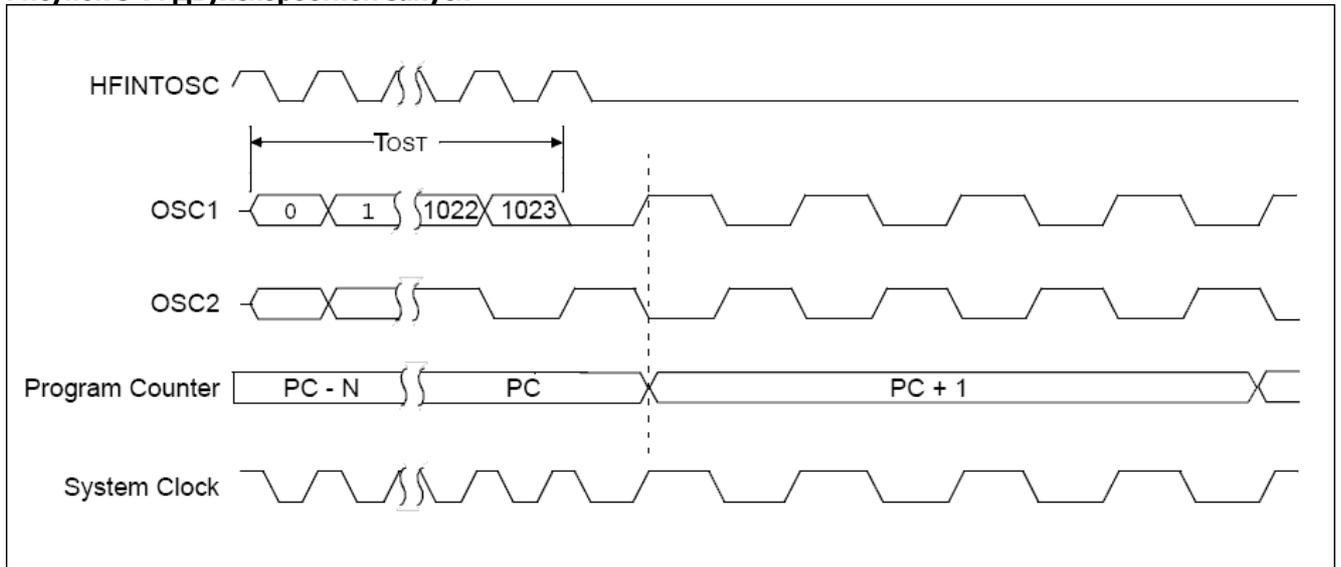
### 3.7.2 Последовательность двухскоростного старта

1. Пробуждение после сброса по включению питания или спящего режима.
2. Начинается выполнение инструкций от внутреннего генератора на частоте установленной битами IRCF<2:0> регистра OSCCON.
3. OST разрешён для подсчёта 1024 тактовых циклов.
4. После окончания времени OST, ожидать срез тактового импульса внутреннего генератора.
5. OSTS установить.
6. Системное тактирование удерживается на низком уровне следующее среза тактового импульса нового тактирования (LP, XT или HS режим).
7. Системное тактирование переключается на внешний тактовый источник.

### 3.7.3 Проверка состояния двухскоростного тактирования

Проверка состояния бита OSTS регистра OSCCON подтверждает, если микроконтроллер запущен из внешнего тактового источника, как определено битами FOSC<2:0> в регистре слова конфигурации (CONFIG), или внешнего генератора.

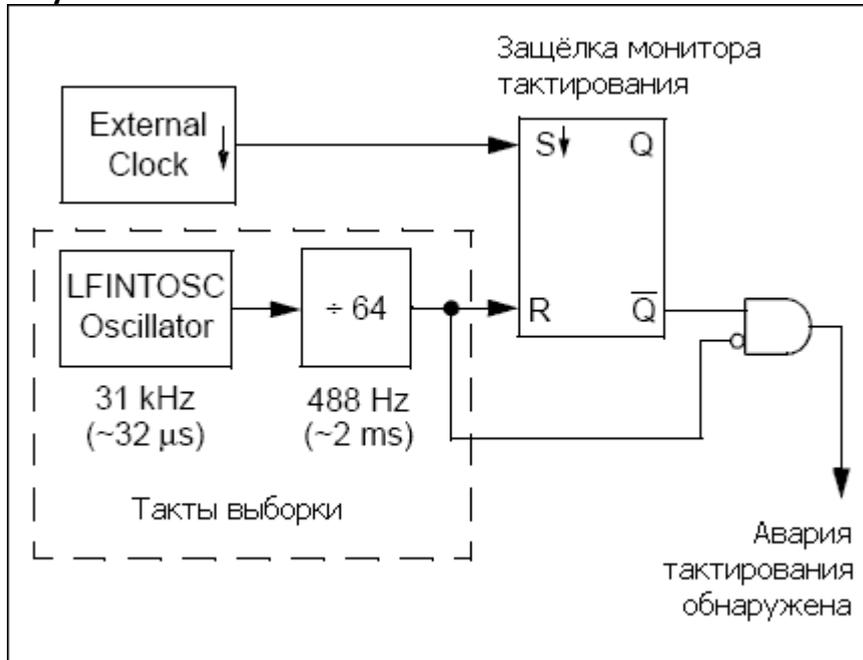
**Рисунок 3-7: Двухскоростной запуск**



### 3.8 Монитор защиты тактирования

Монитор защиты тактирования (FSCM) позволяет устройству продолжать работу при отказе внешнего генератора. FSCM может обнаружить отказ генератора в любой момент после того, как таймер запуска генератора (OST) истёк. FSCM разрешается установкой бита FCMEN регистра слова конфигурации (CONFIG). FSCM применим ко всем режимам внешних генераторов (LP, XT, HS, EC, RC и RCIO).

Рисунок 3-8: Блок-схема FSCM



#### 3.8.1 Определение отказа

Модуль FSCM определяет отказ генератора, сравнивая внешний генератор с FSCM тактами выборки. Такты выборки генерируются делением LFINTOSC на 64. Смотрите рисунок 3-8. Внутри блока детектора отказа есть защёлка. Внешние тактовые импульсы устанавливают защёлку по каждому срезу тактовых импульсов. Такты выборки очищают защёлку по каждому фронту тактов выборки. Отказ обнаруживается, когда целая половина цикла тактов выборки истекает, прежде чем поступит срез первичного такта.

#### 3.8.2 Работа отказ-защита

Когда внешний тактовый генератор отказал, FSCM переключает тактирование устройства на внутренний источник тактирования и устанавливает бит флага OSFIF регистра PIR2. Установка этого флага может генерировать прерывание, если бит OSFIE регистра PIE2 был так же установлен. Микропрограмме устройства может затем потребоваться шаги для смягчения проблем, которые могут возникнуть при отказе тактирования. Системное тактирование может продолжать поступать от внутреннего источника тактирования, пока микропрограмма устройства успешно перезапустит внешний генератор и переключится обратно на работу от внешнего источника.

Внутренний источник тактирования, выбираемый FSCM, определён битами IRCF<2:0> регистра OSCCON. Это позволяет конфигурировать внутренний генератор перед отказом.

#### 3.8.3 Очистка состояния отказ-защита

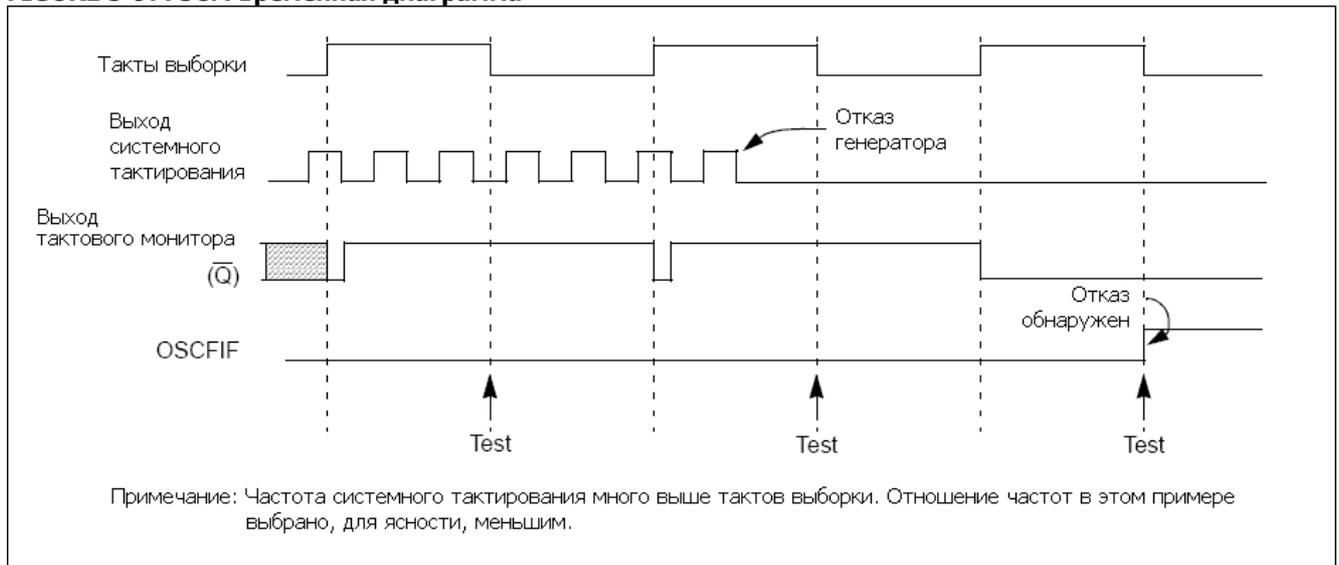
Состояние отказ-защита очищается после сброса, выполнения инструкции SLEEP или переключения бита SCS регистра OSCCON. Когда бит SCS переключен, OST перезапускается. Пока OST работает, устройство продолжает работать от INTOSC выбранного в OSCCON. Когда OST время окончено, состояние отказ-защита очищено и устройство работает от внешнего источника тактовых импульсов. Состояние отказ-защита должно быть очищено, перед тем как флаг OSFIF может быть очищен.

#### 3.8.4 Сброс или пробуждение из спящего режима

FSCM разработан для определения отказа генератора после окончания счёта таймера запуска (OST). OST используется после пробуждения из спящего режима и после любого типа сброса. OST не используется с EC или RC режимами тактирования так что FSCM может быть активным так скоро как сброс или пробуждение завершены. Когда FSCM разрешен, двухскоростной запуск так же разрешен. Следовательно, пока OST работает, устройство выполняет код.

**Примечание:** Из-за широкого диапазона времён запуска, схема отказ-защита не активна в течении запуска генератора (т.е. после сброса или спящего режима). После соответствующего количества времени пользователь должен проверить бит OSTS регистра OSCCON, чтобы проверить запуск генератора, и что резервирование системного тактирования успешно завершено.

**FIGURE 3-9: FSCM временная диаграмма**



**TABLE 3-2: Перечень регистров, связанных с тактовыми источниками**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets <sup>(1)</sup>
CONFIG <sup>(2)</sup>	CPD	CP	MCLRRE	/PWRTE	WDTE	FOSC2	FOSC1	FOSC0	-	-
OSCCON	-	IRCF2	IRCF1	IRCF0	OSTS	HTS	LTS	SCS	-110 x000	-110 x000
OSCTUNE	-	-	-	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	---u uuuu
PIE1	-	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
PIR1	-	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000

**Описание:** x = неизвестный, u = неизменный, - = не задействованный, читается как '0'. Затенённые ячейки не используются генераторами.

**Примечание 1:** Другие (не включение питания) сбросы, включая MCLR сброс и сброс от сторожевого таймера в течении нормальной работы.

**Примечание 2:** Смотреть регистр слова конфигурации (регистр 14-1) для работы всех битов регистра.

## 4.0 I/O Порты

Имеется восемнадцать доступных универсальных выводов I/O. В зависимости от того, какие периферийные устройства разрешены, некоторые или все выводы могут быть не доступны как универсальный I/O. Вообще, когда периферийное устройство разрешено, связанный с ним вывод не может использоваться как универсальный I/O.

### 4.1 PORTA and the TRISA Registers

PORTA 6-ти разрядный двунаправленный порт. TRISA (Регистр 4-2) – соответствующий регистр направления данных. Установка бита TRISA в '1' делает соответствующую ножку PORTA вводом (т.е., отключает выходной драйвер). Очистка бита TRISA делает соответствующую ножку PORTA выводом (т.е., включает выходной драйвер и помещает содержимое выходной защёлки на выбранный вывод). Исключением является RA3, который служит только для ввода и соответствующий бит всегда читается как '1'. Пример 4-1 показывает как инициализировать PORTA.

Чтение регистра PORTA возвращает состояние на выводах порта, а запись производится в защёлку PORTA. Все операции записи в порт выполняются по принципу "чтение-модификация-запись". Следовательно, запись в порт подразумевает, что вывод порта читается, значение модифицируется и затем пишется в защёлку порта. RA3 читается как '0', когда MCLR = 1.

Регистр TRISA управляет выходными драйверами ножек PORTA даже когда они используются как аналоговые входы. Пользователь должен убедиться, что биты регистра TRISA настроены на ввод, когда использует их как аналоговые входы. Ножки I/O, сконфигурированные на аналоговый ввод, всегда читаются '0'.

**Примечание:** Регистры ANSEL и ANSELH должны быть инициализированы, чтобы конфигурировать аналоговый канал, как цифровой ввод. Ножки, конфигурированные как аналоговый ввод, читаются как '0'.

#### Пример 4-1: Инициализация PORTA

```
BCF STATUS,RP0 ;Установить банк 0
BCF STATUS,RP1 ;
CLRF PORTA ;Очистить PORTA
BSF STATUS,RP1 ;Установить 2
CLRF ANSEL ;Цифровой I/O
BSF STATUS,RP0 ;Установить банк 1
BCF STATUS,RP1 ;
MOVLW 0Ch ;Установить RA<3:2> как вводы
MOVWF TRISA ;и установить RA<5:4,1:0>
;как выходы
BCF STATUS,RP0 ;Установить банк 0
```

#### Регистр 4-1: PORTA: Регистр PORTA

U-0	U-0	R/W-x	R/W-x	R-x	R/W-x	R/W-x	R/W-x
-	-	RA5	RA4	RA3	RA2	RA1	RA0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-6 Недействительные:** Читаются как '0'

**Биты 5-0 RA<5:0>:** PORTA I/O Pin bit  
 1 = Port pin is > VIH  
 0 = Port pin is < VIL

#### Регистр 4-2: TRISA: Регистр направления данных PORTA

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-6 Недействительные:** Читаются как '0'

**Биты 5-0 TRISA<5:0>:** Бит направления порта  
 1 = PORTA вывод конфигурирован на ввод  
 0 = PORTA вывод конфигурирован на вывод

**Примечание** **1:** TRISA<3> всегда читается '1'.  
**2:** TRISA<5:4> всегда читаются '1' в XT, HS и LP режимах генератора.

## 4.2 Дополнительные функции ножки

Каждая ножка PORTA на этом семействе устройств имеет опцию прерывания на изменение и опцию медленного нарастания. RA0 так же имеет ультра экономичную функцию пробуждения. Следующие три части описывают эти функции.

### 4.2.1 Регистры ANSEL и ANSELH

Регистры ANSEL и ANSELH используются для отключения буферов ввода I/O ножек, что позволяет аналоговому напряжению прикладываться к этим ножкам без вызывания потребления чрезмерного тока. Установка ANSx бита соответствующей ножки заставляет все цифровые чтения ножки возвращать '0' и так же разрешает корректную работу аналоговых функций.

Состояние бита ANSx не оказывает эффекта на цифровой вывод соответствующей ножки. Ножка с очищенным битом TRISx и установленным битом ANSx работает как цифровой вывод, вместе с аналоговой функцией ввода на этой ножке. Ножки с установленным битом ANSx всегда читаются '0', что может вызывать неожиданное поведение, когда выполняются операции чтения или записи на порт подлежащей последовательности "чтение-изменение-запись" всех таких действий.

### 4.2.2 Подтягивающие резисторы

Каждая из ножек PORTA, за исключением RA3, имеет индивидуальный перестраиваемый внутренний подтягивающий резистор. Биты управления WPUAx разрешают или блокируют каждый подтягивающий резистор. Ссылка на регистр 4-4. Каждый подтягивающий резистор автоматически выключается, когда ножка порта сконфигурирована как выход. Подтягивающие резисторы отключаются при сбросе по включению питания к биту RABPU регистра OPTION. Подтягивающий резистор автоматически включается для RA3, когда сконфигурирован как MCLR и отключается когда RA3 I/O. Там не программный контроль за MCLR подтягиванием.

### 4.2.3 Прерывание на изменение

Каждая ножка PORTA индивидуально конфигурируется как ножка, вызывающая прерывание на применение. Бит управления IOCAx разрешает или запрещает функцию прерывания для каждой ножки. Ссылка на Регистр 4-6. При сбросе по включению питания, прерывание на изменение выключается.

Для разрешения прерывания по изменению ножек, значение сравнивается со старым значением защёлкнутым при последнем чтении PORTA. Выходы 'рассогласования' последнего чтения объединяются схемой ИЛИ вместе для установки флага прерывания по изменению PORTA, бит (RABIF) в регистре INTCON (Регистр 2-6).

Это прерывание может пробудить устройство из спящего режима. Пользователь, в подпрограмме обслуживания прерывания, очищает прерывание:

- а) Любое чтение или запись PORTA. Это завершит состояние несовпадения, затем,
- б) Очистить бит флага RABIF.

Состояние несоответствия можно продолжить установкой бита флага RABIF. Чтение PORTA может закончить состояние несоответствия и позволяет очистить бит флага RABIF. На защёлку, хранящую последнее прочитанное значение не воздействует MCLR или BOR сброс. После этих сбросов, флаг RABIF может оставаться установленным, если несовпадение присутствует.

**Примечание:** Если изменение на ножке I/O происходит, когда выполняется операция чтения (начало цикла Q2), то флаг прерывания RABIF может быть не установленным.

**Регистр 4-3: ANSEL: Регистр аналогового выбора**

| R/W-1 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| ANS7  | ANS6  | ANS5  | ANS4  | ANS3  | ANS2  | ANS1  | ANS0  |
| Бит 7 | Бит 6 | Бит 5 | Бит 4 | Бит 3 | Бит 2 | Бит 1 | Бит 0 |

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-0 ANS<7:0>:** Биты аналогового выбора  
 Выбор аналоговой, между аналоговой и цифровой, функции на ножках AN<7:0>, соответственно.  
 1 = Аналоговый вход. Ножка назначается как аналоговый вход<sup>(1)</sup>.  
 0 = Цифровой I/O. Ножке назначен порт или специальная функция.

**Примечание 1:** Назначение ножки для аналогового ввода, автоматически отключает цифровую схему ввода, подтягивающие резисторы и прерывание на изменение, если доступны. Соответствующий бит TRIS должен быть установлен в режим ввода в порядке позволяющем контроль внешнего напряжения на ножке.

**Регистр 4-4: ANSELH: Старший регистр аналогового выбора<sup>(2)</sup>**

U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
-	-	-	-	ANS11	ANS10	ANS9	ANS8
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-4 Недействительные:** Читаются как '0'

**Биты 3-0 ANS<11:8>:** Биты аналогового выбора  
 Выбор аналоговой, между аналоговой и цифровой, функции на ножках AN<11:8>, соответственно.  
 1 = Аналоговый вход. Ножка назначается как аналоговый вход<sup>(1)</sup>.  
 0 = Цифровой I/O. Ножке назначен порт или специальная функция.

**Note 1:** Назначение ножки для аналогового ввода, автоматически отключает цифровую схему ввода, подтягивающие резисторы и прерывание на изменение, если доступны. Соответствующий бит TRIS должен быть установлен в режим ввода в порядке позволяющем контроль внешнего напряжения на ножке.  
**2:** PIC16F677/PIC16F685/PIC16F687/PIC16F689/PIC16F690 только.

**Регистр 4-5: WPUA: Регистр PORTA**

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
-	-	WPUA5	WPUA4	-	WPUA2	WPUA1	WPUA0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                    W=бит записывается                    U=не задействованный бит, читается как '0'  
-n=Значение при POR            '1'=бит установлен                    '0'=бит очищен                                    X=бит не определён

**Биты 7-6    Недействительные:** Читаются как '0'

**Биты 5-4    WPUA<5:4>:** Бит подтягивающего резистора

1 = Подтягивающий резистор подключен

0 = Подтягивающий резистор отключен

**Бит 3        Недействителен:** Читается как '0'

**Биты 2-0    WPUA<2:0>:** Бит подтягивающего резистора

1 = Подтягивающий резистор подключен

0 = Подтягивающий резистор отключен

**Note 1:** Глобальный бит RABPU регистра OPTION должен быть разрешён для разрешения индивидуальных подтягивающих резисторов.

**2:** Подтягивающие резисторы автоматически отключаются, если ножка в режиме вывода (TRISA = 0).

**3:** RA3 подтягивающий резистор подключен, когда конфигурирован как MCLR и отключен когда конфигурирован как I/O в слове конфигурации.

**4:** WPUA<5:4> всегда читается '1' в XT, HS и LP режимах генератора.

**REGISTER 4-6: IOCA: INTERRUPT-ON-CHANGE PORTA REGISTER**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
-	-	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                    W=бит записывается                    U=не задействованный бит, читается как '0'  
-n=Значение при POR            '1'=бит установлен                    '0'=бит очищен                                    X=бит не определён

**Биты 7-6    Недействительные:** Читаются как '0'

**Биты 5-0    IOCA<5:0>:** Бит управления прерывания по изменению PORTA

1 = Прерывание по изменению включено

0 = Прерывание по изменению выключено

**Примечание    1:** Глобальное разрешение прерывания (GIE) должно быть разрешено для распознавания индивидуальных прерываний.

**2:** IOCA<5:4> всегда читаются '1' в XT, HS и LP режимах генератора.

#### 4.2.4 Пробуждение с ультранизким потреблением

Пробуждение с ультранизким потреблением (ULPWU) на RA0 позволяет при медленном падении напряжения генерировать прерывание по изменению на RA0 без избыточного потребления тока. Режим выбирается установкой бита ULPWUE регистра PCON. Это допускает небольшой втекающий ток, что может использоваться для разрядки конденсатора на RA0.

Следуйте этим шагам, для использования этих характеристик:

- a) Зарядить конденсатор на RA0 сконфигурировав ножку RA0 на вывод (= 1).
- b) Конфигурировать RA0 на ввод.
- c) Разрешить прерывание на изменение для RA0.
- d) Установить бит ULPWUE регистра PCON для начала разрядки конденсатора.
- e) Использовать инструкцию SLEEP.

Когда напряжение на RA0 опустится ниже VIL, будет сгенерировано прерывание, что заставит устройство пробудиться и выполнить следующую инструкцию. Если бит GIE регистра INTCON установлен, устройство затем вызывает подпрограмму по вектору прерывания (0004h). Смотрите **Часть 4.4.2 "Прерывание на изменение"** and **Часть 14.3.3 "Прерывание PORTA/PORTB"** для большей информации.

Эта характеристика обеспечивает технологию низкого потребления для периодического пробуждения устройства из спящего режима. Временные периоды пробуждения зависят от времени разрядки RC цепочки на RA0. Смотрите пример 4-2 инициализации модуля пробуждения с ультранизким потреблением.

Последовательный резистор между RA0 и внешним конденсатором обеспечивает защиту от перегрузки по току для ножки RA0/AN0/C1IN+/ICSPDAT/ULPWU и может допускать программную калибровку временных периодов (смотрите рисунок 4-1). Таймер может быть использован для измерения времени разрядки конденсатора. Время зарядки может затем быть подстроено для обеспечения желаемой задержки прерывания. Эта технология может использоваться для компенсации температурного дрейфа, точности напряжения и компонентов. Пробуждение с ультранизким потреблением может также быть сконфигурированным как простой программируемый низковольтный детектор или температурный датчик.

**Примечание:** Для большей информации, обращайтесь к документу AN879, "Использование *Microchip* модуля пробуждения с ультранизким потреблением" (DS00879).

#### EXAMPLE 4-2: ULTRA LOW-POWER WAKE-UP INITIALIZATION

```
BCF STATUS,RP0 ;Банк 0
BCF STATUS,RP1 ;
BSF PORTA,0 ;Установить RA0 защёлку данных
BSF STATUS,RP1 ;Банк 2
BCF ANSEL,0 ;RA0 цифровой I/O
BSF STATUS,RP0 ;Банк 1
BCF STATUS,RP1 ;
BCF TRISA,0 ;На выход высокий уровень
CALL CapDelay ;для зарядки конденсатора
BSF PCON,ULPWUE ;Разрешить ULP пробуждение
BSF IOCA,0 ;Выбрать RA0 IOC
BSF TRISA,0 ;RA0 на ввод
MOVLW B'10001000' ;Разрешить прерывание
MOVWF INTCON ;и очистить флаг
BCF STATUS,RP0 ;Банк 0
SLEEP ;Ожидать для IOC
```

## 4.2.5 Схемы и описания ножек

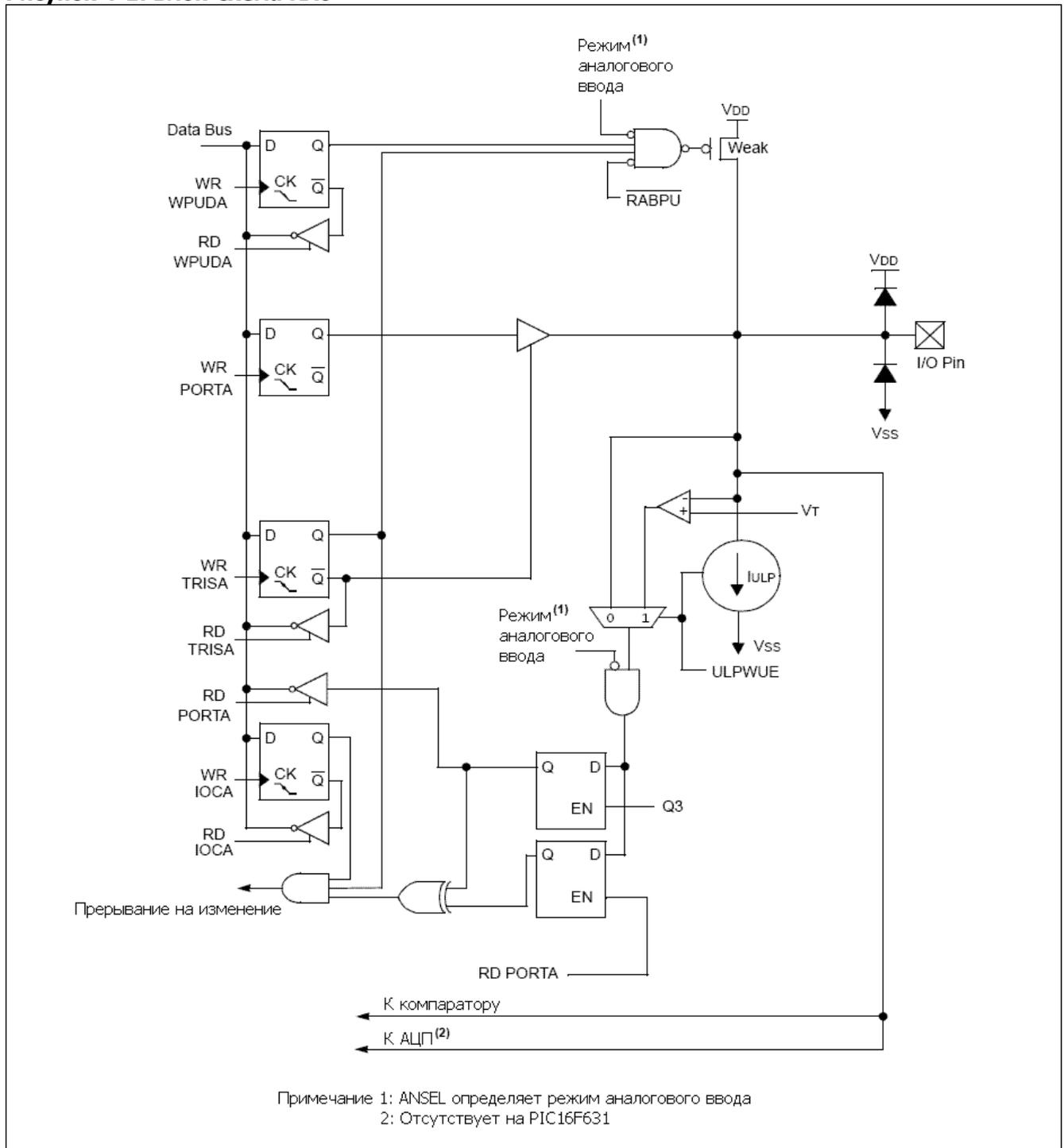
Каждая ножка PORTA мультиплексирована с другими функциями. Ножки и их комбинации функций кратко описаны здесь. Для специфической информации о индивидуальных функциях, такой как компаратор или A/D преобразователь (ADC), обращайтесь к соответствующим частям данного описания.

### 4.2.5.1 RA0/AN0/C1IN+/ICSPDAT/ULPWU

На рисунке 4-2 показана схема для ножки. Ножка RA0/AN0/C1IN+/ICSPDAT/ULPWU конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый вход для ADC (кроме PIC16F631)
- аналоговый вход компаратора C1
- Внутрисхемное последовательное программирование™
- аналоговый вход для пробуждения с ультранизким потреблением

**Рисунок 4-1: Блок-схема RA0**



#### 4.2.5.2 RA1/AN1/C12IN0-/VREF/ICSPCLK

На рисунке 4-2 показана схема для этой ножки. Ножка RA1/AN1/C12IN0-/VREF/ICSPCLK конфигурируется для одной из следующих функций:

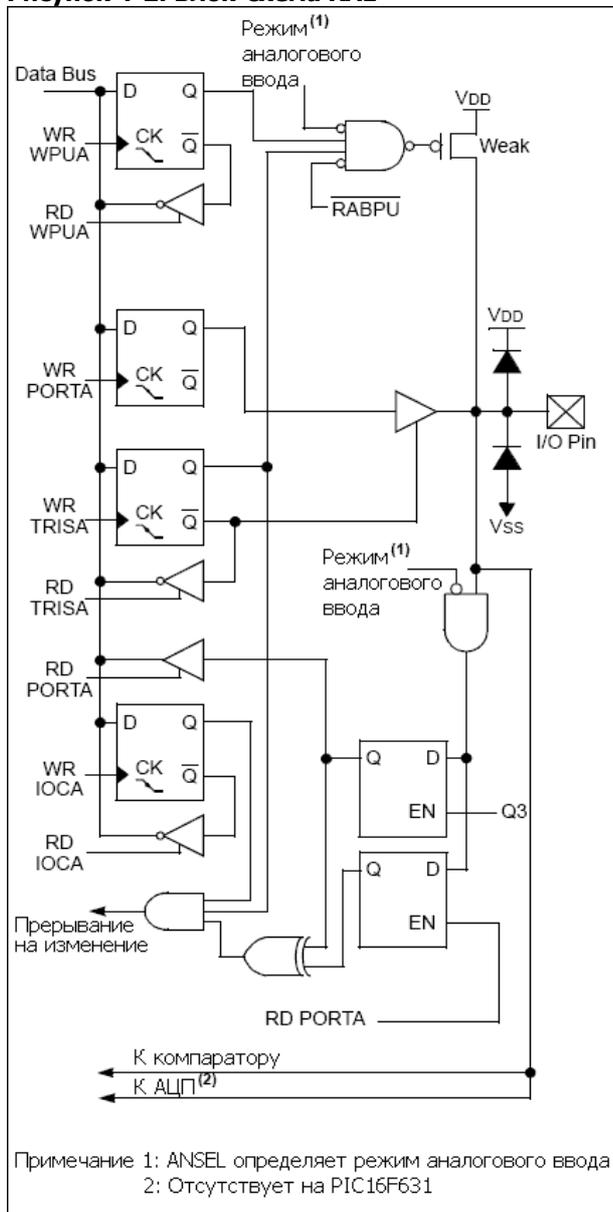
- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- аналоговый вход компаратора C1 или C2
- вход опорного напряжения для АЦП
- Тактирование внутрисхемного последовательного программирования

#### 4.2.5.3 RA2/AN2/T0CKI/INT/C1OUT

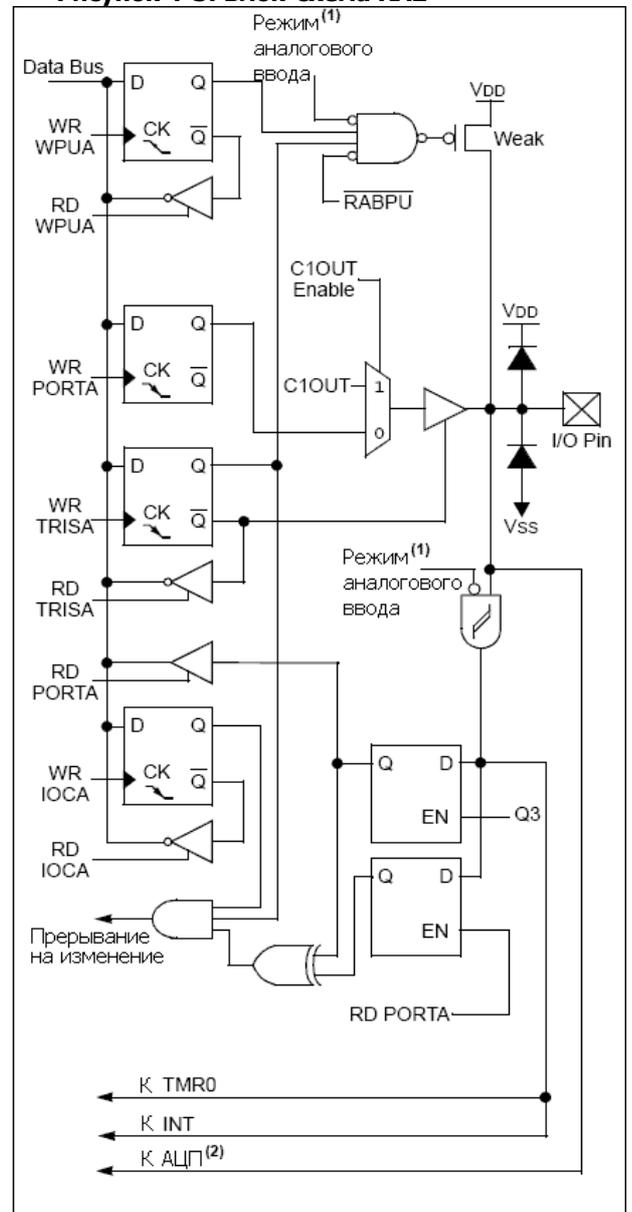
На рисунке 4-3 показана схема для этой ножки. Ножка RA2/AN2/T0CKI/INT/C1OUT конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- тактовый вход для TMR0
- внешнее, запускаемое по перепаду, прерывание
- цифровой выход для компаратора C1

**Рисунок 4-2: Блок-схема RA1**



**Рисунок 4-3: Блок-схема RA2**



#### 4.2.5.4 RA3/MCLR/VPP

На рисунке 4-4 изображены схемы для этой ножки. Ножка RA3/MCLR/VPP может быть перестроена для одной из следующих функций:

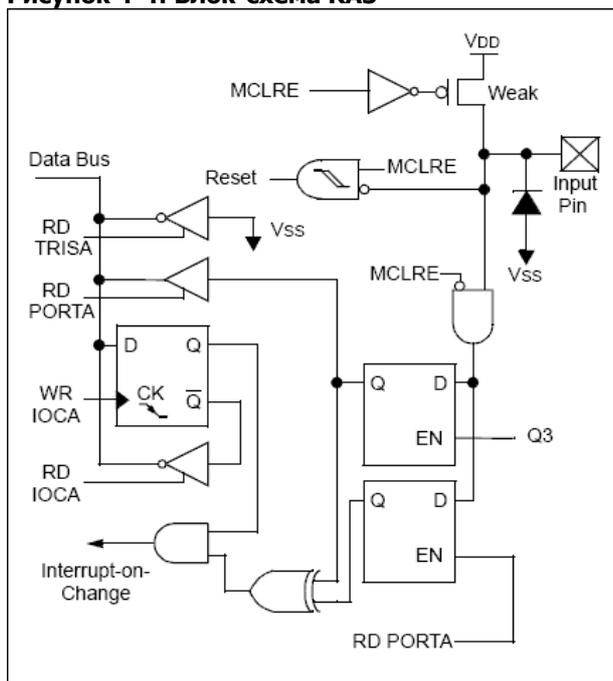
- универсальный ввод
- вход очистки-сброса с подтягивающим резистором

#### 4.2.5.5 RA4/AN3/T1G/OSC2/CLKOUT

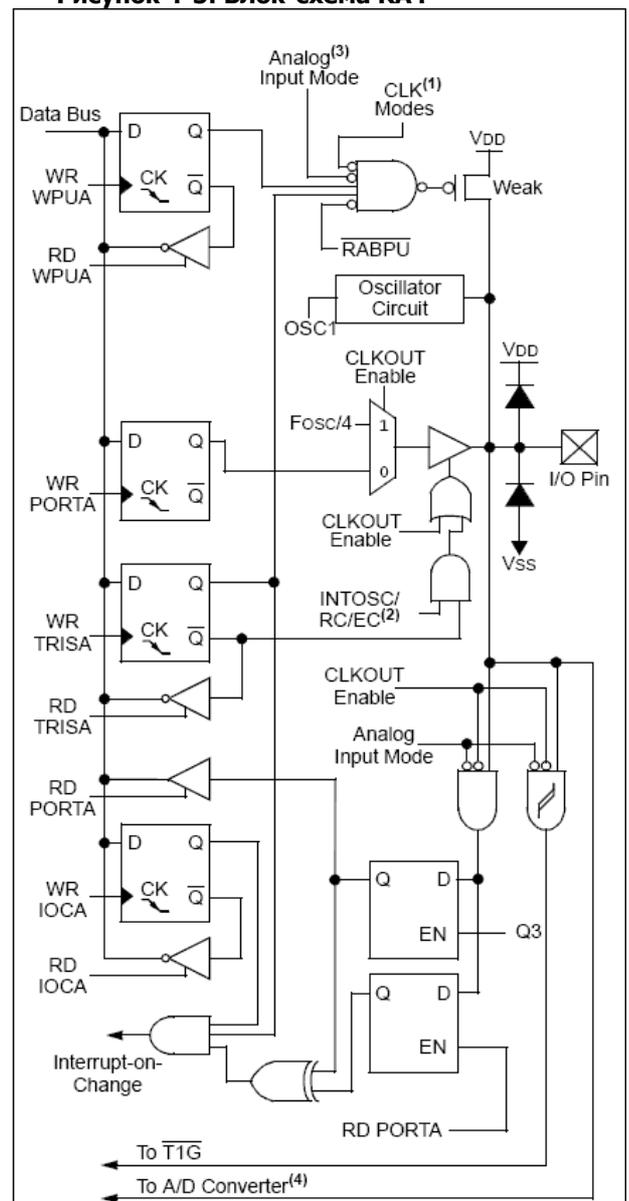
На рисунке 4-5 изображена схема для этой ножки. Ножка RA4/AN3/T1G/OSC2/CLKOUT может быть перестроена для одной из следующих функций:

- универсальный I/O
- аналоговый ввод для АЦП (кроме PIC16F631)
- ворота TMR1
- подключение кристалла резонатора
- тактовый выход

**Рисунок 4-4: Блок-схема RA3**



**Рисунок 4-5: Блок-схема RA4**



Примечание 1: CLK режимы разрешены для XT, HS, LP, LPTMR1 и CLKOUT

2: С опцией CLKOUT

3: ANSEL определяет режим аналогового ввода

4: Не реализовано на PIC16F631



**Таблица 4-1: Список регистров ассоциированных с PORTA**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets <sup>(1)</sup>
ADCON0	ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	0000 0000	0000 0000
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CM1CON0	C1ON	C1OUT	C1OE	C1POL	-	C1R	C1CH1	C1CH0	0000 -000	0000 -000
INTCON	GIE	PEIE	T0IE	INTE	RABIE	T0IF	INTF	RABIF	0000 000x	0000 000x
IOCA	-	-	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	--00 0000	--00 0000
OPTION_REG	/RABPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTA	-	-	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	/T1SYNC	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
WPUA	-	-	WPUA5	WPUA4	-	WPUA2	WPUA1	WPUA0	--11 -111	--11 -111

**Описание:** x = неизвестный, u = неизменный, - = не задействованный, читается как '0'. Затенённые ячейки не используются с PORTA.

## 4.3 Регистры PORTB и TRISB

Двухнаправленный PORTB имеет 4-х битную ширину. Регистр направления передачи данных TRISB (Регистр 4-6). Установка бит TRISB в 1 переводит соответствующие ножки PORTB в режим ввода (т.е., переводит соответствующие выходные драйверы в высокоимпедансное состояние). Очистка бит TRISB (= 0) переводит соответствующие ножки PORTB в режим вывода (т.е., разрешает выходные драйверы и помещает содержимое выходных защёлки на выбранные ножки). Пример 4-3 показывает как инициализировать PORTB. Чтение регистра PORTB (регистр 4-5) приводит к чтению состояния ножек, поскольку запись в него приводит к записи в защёлку порта. Все операции записи являются операциями чтение-модификация-запись. Следовательно запись в порт подразумевает, что ножки порта читаются, прочитанное значение модифицируется и затем записывается в защёлку данных порта.

Регистр TRISB управляет выходными драйверами ножек порта PORTB. Пользователь должен установить биты регистра TRISB, когда использует порт для аналогового ввода. I/O ножки, сконфигурированные как аналоговые входы, всегда читаются как '0'.

### Пример 4-3: Инициализация PORTB

```
BCF STATUS,RP0 ;Банк 0
BCF STATUS,RP1 ;
CLRF PORTB ;очистить PORTB
BSF STATUS,RP0 ;Банк 1
MOVLW FFh ;Установить RB<7:4> как входы
MOVWF TRISB ;
BCF STATUS,RP0 ;Банк 0
```

**Примечание:** Регистры ANSEL и ANSELH должны быть инициализированы, чтобы конфигурировать аналоговый канал, как цифровой ввод. Ножки, сконфигурированные как аналоговый ввод, читаются как '0'.

## 4.4 Additional PORTB Pin Functions

Ножки RB<7:4> порта PORTB имеют опции прерывания на изменение и опции подтягивания. Следующие три подраздела описывают эти функции ножек PORTB.

### 4.4.1 Подтяжки

Каждая из ножек PORTB имеет индивидуально конфигурируемый внутренний подтягивающий резистор. Биты управления WPUB<7:4> разрешают или запрещают подтяжки (смотреть регистр 4-9). Каждый подтягивающий резистор автоматически отключается, если ножка порта конфигурируется на как выход. Все подтяжки выключаются сбросом по включению питания битом RABPU регистра OPTION.

### 4.4.2 Прерывание на изменение

Четыре ножки PORTB индивидуально конфигурируются как ножки прерывания по изменению. Биты управления IOCB<7:4> включают или выключают функцию прерывания для каждой ножки. Обратитесь к регистру 4-10. Характеристика прерывания по изменению выключается сбросом по включению питания.

Для включения ножек прерывания на по изменению, настоящее значение сравнивается со старым значением, защёлкнутом при последнем чтении PORTB для определения какие биты имеют изменения или несовпадения со старым значением. 'Несовпадающие' выходы, объединённые вместе схемой ИЛИ, устанавливают бит флага прерывания (RABIF) по изменению PORTB в регистре INTCON (регистр 2-3).

Это прерывание может пробуждать устройство из спящего режима. Пользователь в подпрограмме обслуживания прерывания очищает прерывание:

- Любое чтение или запись PORTB. Это завершит состояние несоответствия.
- Очистить бит флага RABIF.

Состояние несовпадения можно продолжить установкой бита флага RABIF. Чтение или запись PORTB закончит состояние несовпадения и позволит очистить бит RABIF. Защёлка удерживает последнее прочитанное значение не зависимо от MCLR или сброса по снижению напряжения. После этих сбросов флаг RABIF может оставаться установленным, если несовпадение осталось.

**Примечание:** Если изменение на ножке I/O должно произойти, когда операция чтения выполняется (старт Q2 цикла), то флаг прерывания RABIF может оказаться не установленным. Кроме того, поскольку чтение или запись в порт влияет на все биты порта, нужно проявить осторожность когда многие ножки используются в режиме прерывание на изменение. Изменения на одной ножке можно не увидеть, пока обслуживается изменение на другой ножке.

### Регистр 4-7: PORTB: Регистр PORTB

R/W-x	R/W-x	R/W-x	R/W-x	U-0	U-0	U-0	U-0
RB7	RB6	RB5	RB4	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-4** **RB<7:4>:** PORTB I/O бит ножки  
1 = Уровень на ножке порта > V<sub>IH</sub>  
0 = Уровень на ножке порта < V<sub>IL</sub>

**Биты 3-0** **Незадействованные:** Читаются как '0'

**Регистр 4-8: TRISB: регистр направления PORTB**

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
TRISB7	TRISB6	TRISB5	TRISB4	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                    W=бит записывается                    U=не задействованный бит, читается как '0'  
-n=Значение при POR            '1'=бит установлен                    '0'=бит очищен                                    X=бит не определён

**Биты 7-4**    **TRISB<7:4>**: Бит управления направлением PORTB  
1 = PORTB ножка конфигурирована как ввод  
0 = PORTB ножка конфигурирована как вывод

**Биты 3-0**    **Незадействованные**: Читаются как '0'

**Регистр 4-9: WPUB: Регистр подтяжек**

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
WPUB7	WPUB6	WPUB5	WPUB4	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                    W=бит записывается                    U=не задействованный бит, читается как '0'  
-n=Значение при POR            '1'=бит установлен                    '0'=бит очищен                                    X=бит не определён

**Биты 7-4**    **WPUB<7:4>**: Бит подтяжки  
1 = Подтяжка разрешена  
0 = Подтяжка запрещена

**Биты 3-0**    **Незадействованные**: Читаются как '0'

**Примечание**    **1:** Глобальный бит RABPU регистра OPTION может быть включен для разрешения индивидуальной подтяжки.  
**2:** Устройство подтяжки автоматически отключается, если ножка в режиме вывод (TRISB<7:4> = 0).

**Регистр 4-10: IOCB: Регистр прерывания на изменение PORTB**

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
IOCB7	IOCB6	IOCB5	IOCB4	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                    W=бит записывается                    U=не задействованный бит, читается как '0'  
-n=Значение при POR            '1'=бит установлен                    '0'=бит очищен                                    X=бит не определён

**Биты 7-4**    **IOCB<7:4>**: Управляющие биты прерывания на изменение PORTB  
1 = прерывание на изменение включено  
0 = прерывание на изменение выключено

**Биты 3-0**    **Незадействованные**: Читаются как '0'

### 4.4.3 Описание ножек и диаграмм

Каждая ножка PORTB мультиплексирована с другими функциями. Здесь кратко описаны ножки и их комбинированные функции. Для специфической информации о индивидуальных функциях, таких как SSP, I2C™ или прерывания, обращайтесь к соответствующей части в этом описании.

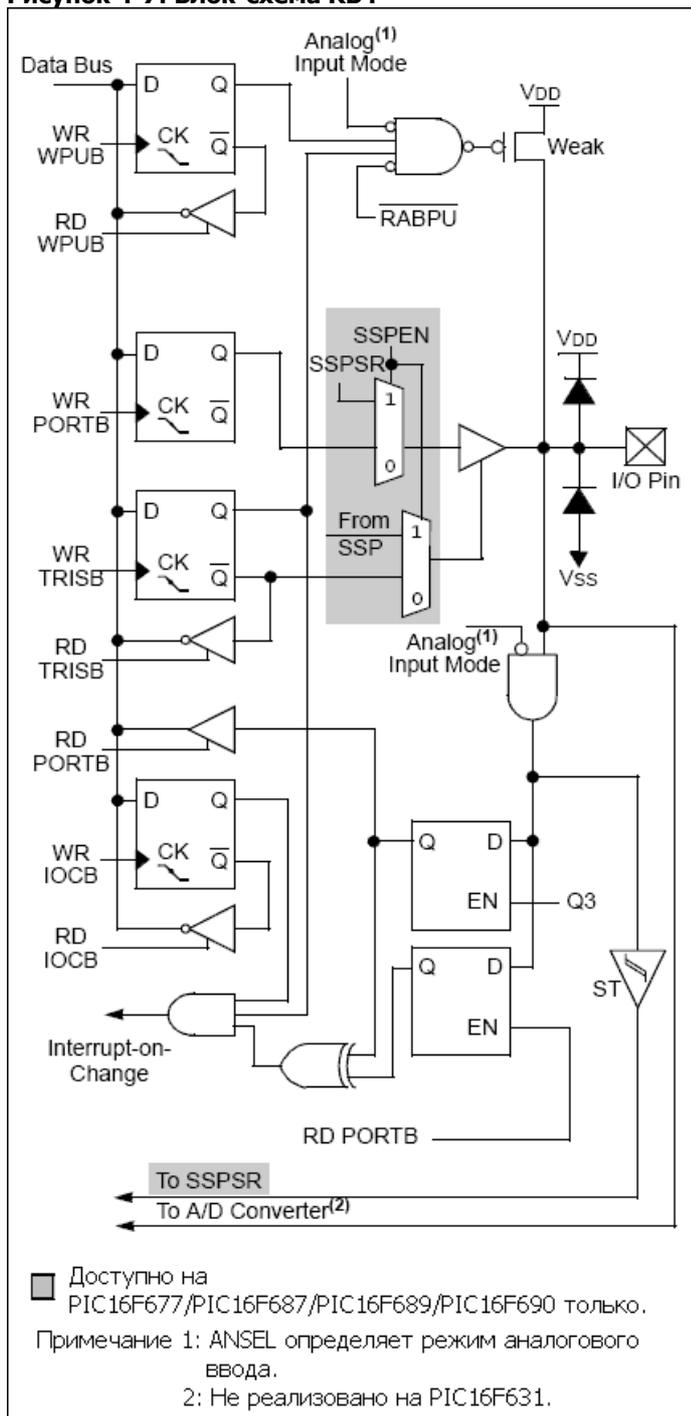
#### 4.4.3.1 RB4/AN10/SDI/SDA

На рисунке 4-7 показана схема для этой ножки. Ножка The RB4/AN10/SDI/SDA<sup>(1)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- I/O данных SPI
- I/O данных I2C

**Note 1:** SDI и SDA доступны только на PIC16F687/PIC16F689/PIC16F690.

**Рисунок 4-7: Блок-схема RB4**



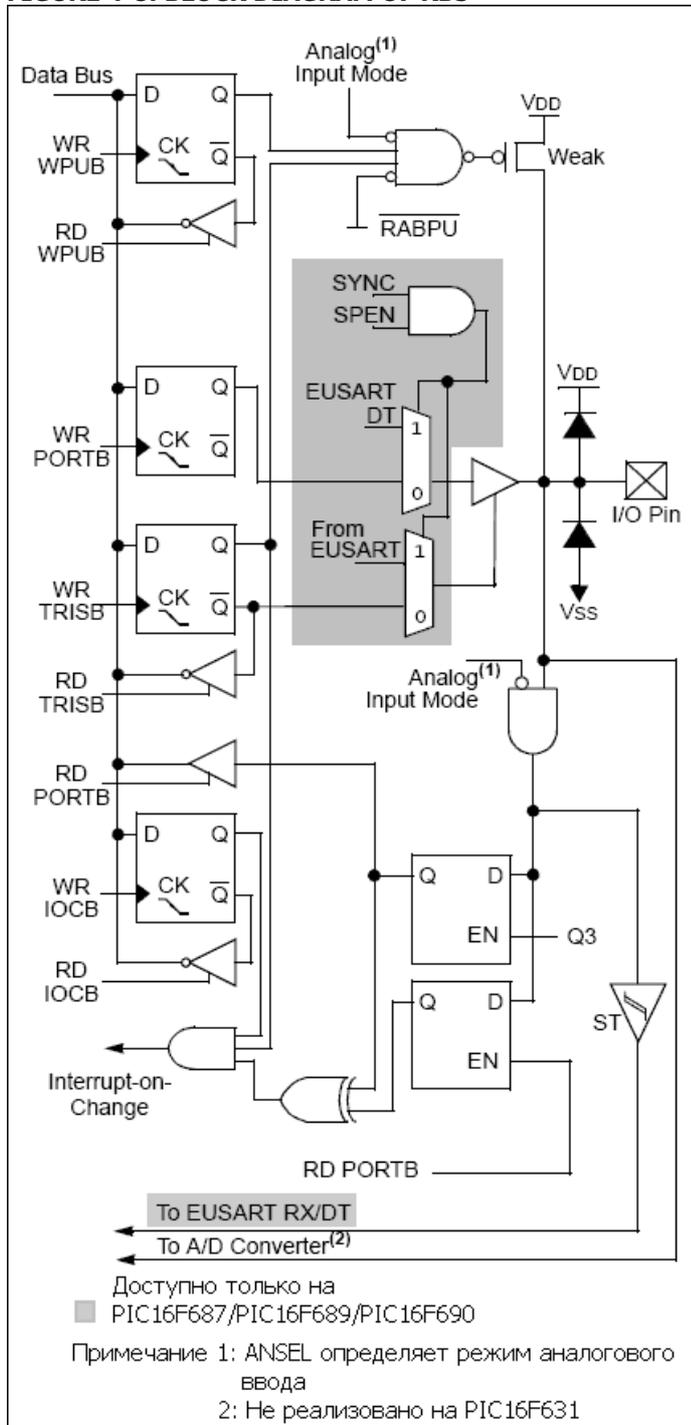
#### 4.4.3.2 RB5/AN11/RX/DT

На рисунке 4-8 показана схема для этой ножки. Ножка RB5/AN11/RX/DT<sup>(1,2)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- асинхронный последовательный ввод
- синхронный последовательный I/O данных

**Примечание** 1: RX и DT доступны только на PIC16F687/PIC16F689/PIC16F690.  
2: AN11 не реализован на PIC16F631.

**FIGURE 4-8: BLOCK DIAGRAM OF RB5**



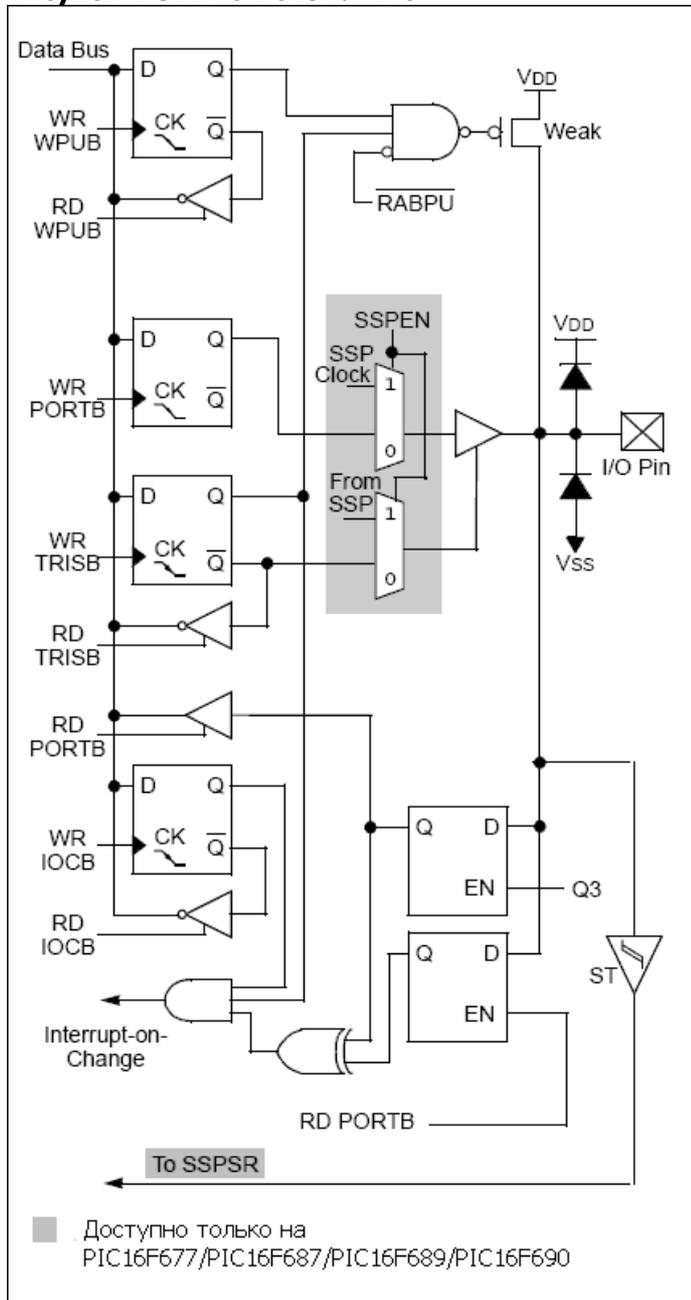
### 4.4.3.3 RB6/SCK/SCL

На рисунке 4-9 изображена схема для этой ножки. Ножка RB6/SCK/SCL<sup>(1)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- тактирование SPI
- тактирование I2C™

**Примечание 1:** SCK и SCL доступны только на PIC16F677/PIC16F687/PIC16F689/PIC16F690.

**Рисунок 4-9: Блок-схема RB6**



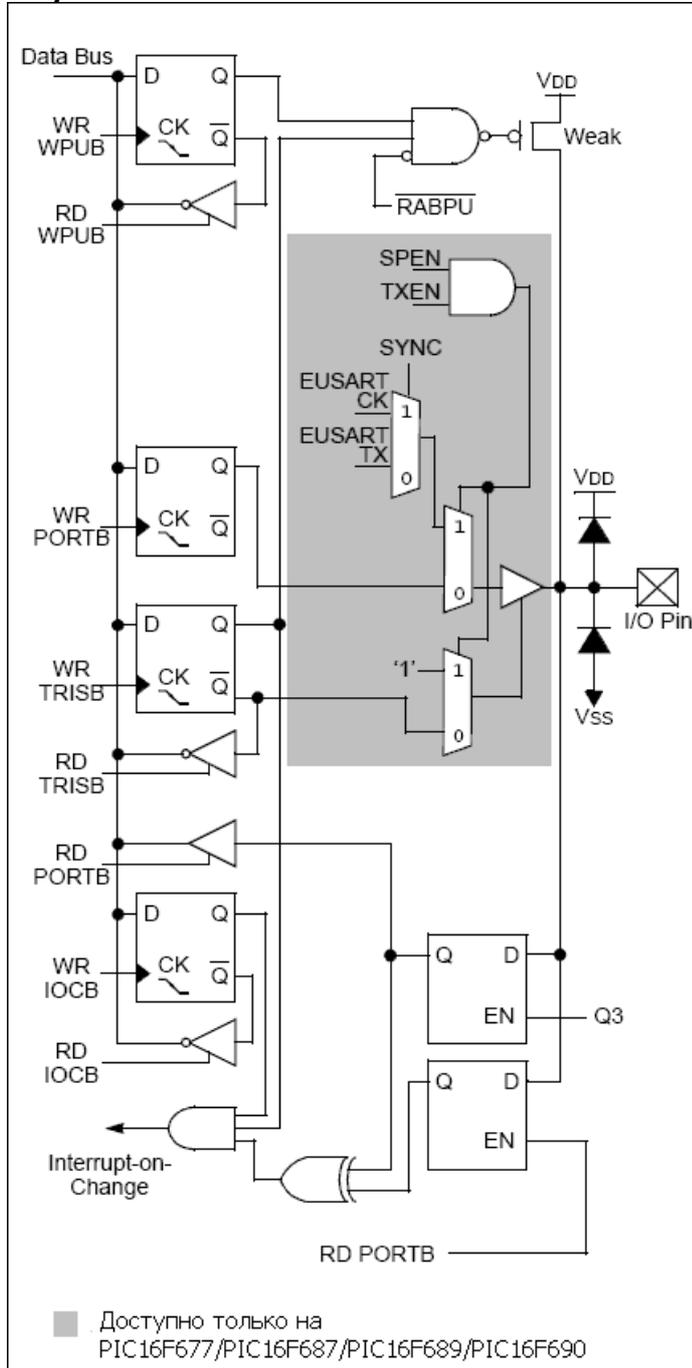
#### 4.4.3.4 RB7/TX/CK

На рисунке 4-10 показана схема для этой ножки. Ножка RB7/TX/CK<sup>(1)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- асинхронный последовательный вывод
- синхронное тактирование I/O

**Note 1:** TX и CK доступны только PIC16F687/PIC16F689/PIC16F690.

**Рисунок 4-10: Блок-схема RB7**



**TABLE 4-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets <sup>(1)</sup>
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	-	-	-	-	0000 ----	0000 ----
INTCON	GIE	PEIE	TOIE	INTE	RABIE	TOIF	INTF	RABIF	0000 000x	0000 000x
PORTB	RB7	RB6	RB5	RB4	-	-	-	-	xxxx ----	uuuu ----
TRISB	GIE	PEIE	TOIE	INTE	-	-	-	-	1111 ----	1111 ----
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	-	-	-	-	1111 ----	1111 ----

**Описание:** x = неизвестный, u = неизменный, – = не задействованный, читается как '0'. Затенённые ячейки не используются с PORTB.

## 4.5 Регистры PORTC и TRISC

PORTC двунаправленный 8-ми разрядный порт. Соответственно регистр направления данных TRISC (регистр 4-10). Установка TRISC бита (= 1) настраивает соответствующую ножку PORTC на ввод (т.е., переводит соответствующий выходной драйвер в высокоимпедансное состояние). Очистка TRISC бита (= 0) настраивает соответствующую ножку PORTC на вывод (т.е., разрешает выходной драйвер и помещает содержимое выходной защёлки на выбранную ножку). Пример 4-4 показывает как инициализировать PORTC. Чтение регистра PORTC (регистр 4-9) читает состояния ножек, а запись в него приводит к записи в защёлку порта. Все операции записи являются операциями чтение-модификация-запись. Следовательно запись в порт подразумевает, что ножка порта читается, это значение модифицируется и затем записывается в защёлку данных порта.

Регистр TRISC управляет выходными драйверами ножек PORTC, даже когда они используются как аналоговые входы. Пользователь должен проверять биты регистра TRISC установлены, когда использует их как аналоговые входы. Ножки I/O сконфигурированные как аналоговые входы всегда читаются '0'.

**Примечание:** Регистры ANSEL и ANSELH должны быть инициализированы, чтобы конфигурировать аналоговый канал, как цифровой ввод. Ножки, конфигурированные как аналоговый ввод, читаются как '0'.

### EXAMPLE 4-4: INITIALIZING PORTC

```
BCF STATUS,RP0 ;Банк 0
BCF STATUS,RP1 ;
CLR PORTC ;Очистить PORTC
BSF STATUS,RP1 ;Банк 2
CLR ANSEL ;цифровой I/O
BSF STATUS,RP0 ;Банк 1
BCF STATUS,RP1 ;
MOVLW 0Ch ;Установить RC<3:2> как входы
MOVWF TRISC ;и установить RC<5:4,1:0>
;как выходы
BCF STATUS,RP0 ;Банк 0
```

### REGISTER 4-11: PORTC: PORTC REGISTER

R/W-0	R/W-x						
RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-0 RC<7:0>:** бит ножки PORTC универсальный I/O  
 1 = уровень на ножке порта > VIN  
 0 = уровень на ножке порта < VIL

### REGISTER 4-12: TRISC: PORTC TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-0 TRISC<7:0>:** PORTC биты управления  
 1 = ножка PORTC конфигурирована как ввод  
 0 = ножка порта конфигурирована как вывод

#### 4.5.1 RC0/AN4/C2IN+

RC0 конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый ввод для АЦП (кроме PIC16F631)
- аналоговый ввод компаратора C2

#### 4.5.2 RC1/AN5/C12IN1-

RC1 конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый ввод для АЦП
- аналоговый ввод компаратора C1 или C2

#### 4.5.3 RC2/AN6/C12IN2-/P1D

RC2/AN6/P1D<sup>(1)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый ввод АЦП (кроме PIC16F631)
- Выход ШИМ
- аналоговый ввод компаратора C1 или C2

**Note 1:** P1D доступен только на PIC16F685/PIC16F690.

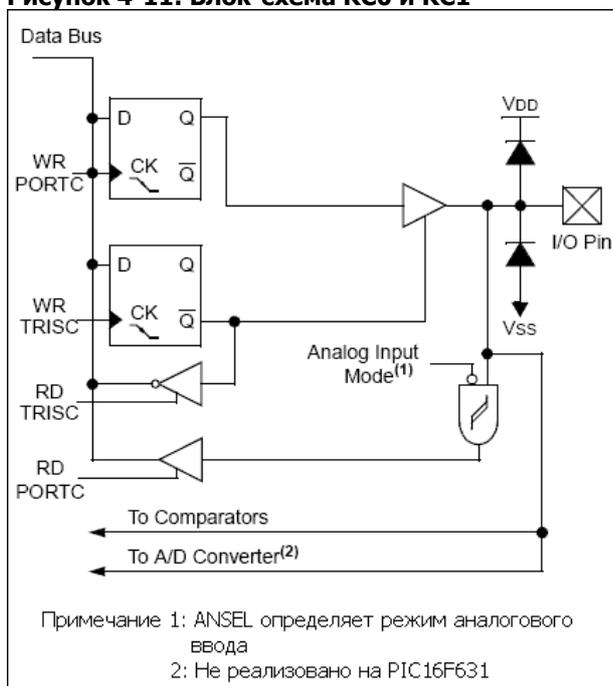
#### 4.5.4 RC3/AN7/C12IN3-/P1C

RC3/AN7/P1C<sup>(1)</sup> конфигурируется для одной из следующих функций:

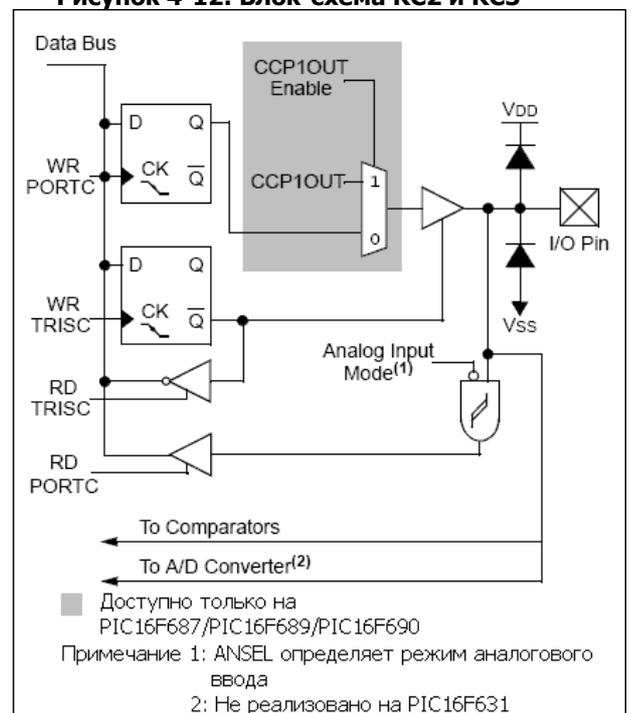
- универсальный I/O
- аналоговый ввод АЦП (кроме PIC16F631)
- Выход ШИМ
- аналоговый ввод компаратора C1 или C2

**Note 1:** P1C доступен только на PIC16F685/PIC16F690.

**Рисунок 4-11: Блок-схема RC0 и RC1**



**Рисунок 4-12: Блок-схема RC2 и RC3**



## 4.5.5 RC4/C2OUT/P1B

RC4/C2OUT/P1B<sup>(1, 2)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- цифровой вывод из компаратора C2
- ШИМ вывод

**Примечание 1:** Одновременное включение C2OUT и P1B причинит конфликт на RC4 и создаст не предсказуемые результаты. Поэтому, если C2OUT разрешен, то ECCP+ не может использоваться в мостовом или полумостовом режимах и vice-versa.

**2:** P1B доступен только на PIC16F685/PIC16F690.

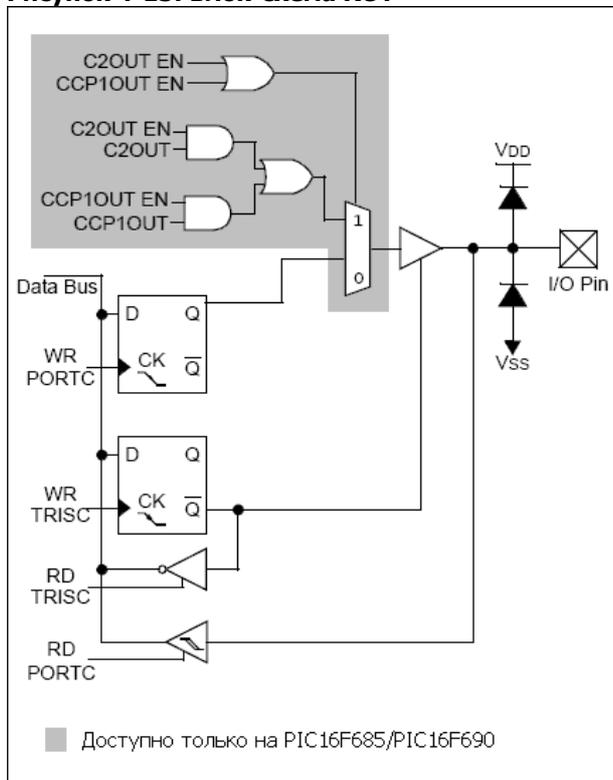
## 4.5.6 RC5/CCP1/P1A

RC5/CCP1/P1A<sup>(1)</sup> конфигурируется для одной из следующих функций:

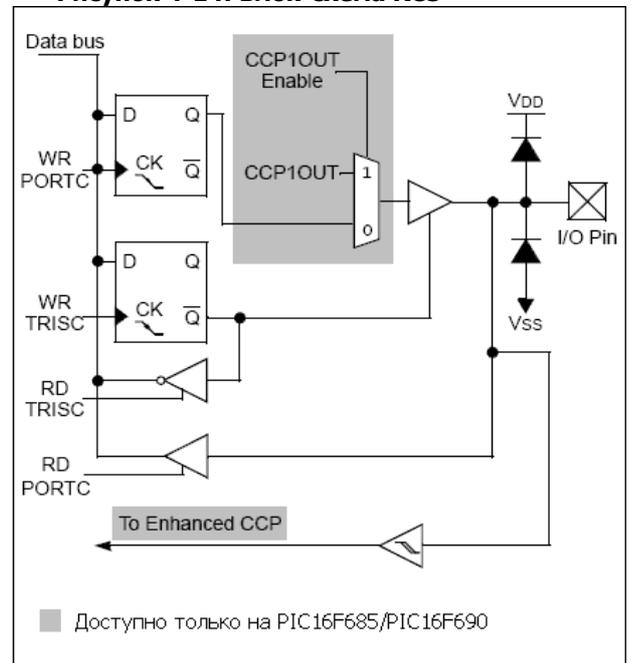
- универсальный I/O
- цифровой ввод/вывод для расширенного CCP
- ШИМ вывод

**Примечание 1:** CCP1 и P1A доступны только на PIC16F685/PIC16F690.

**Рисунок 4-13: Блок-схема RC4**



**Рисунок 4-14: Блок-схема RC5**



#### 4.5.7 RC6/AN8/SS

RC6/AN8/SS<sup>(1,2)</sup> конфигурируется для одной из следующих функций:

- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- подчинённый выбор ввода

**Примечание** 1: SS доступен только PIC16F687/PIC16F689/PIC16F690.  
2: AN8 не реализован на PIC16F631.

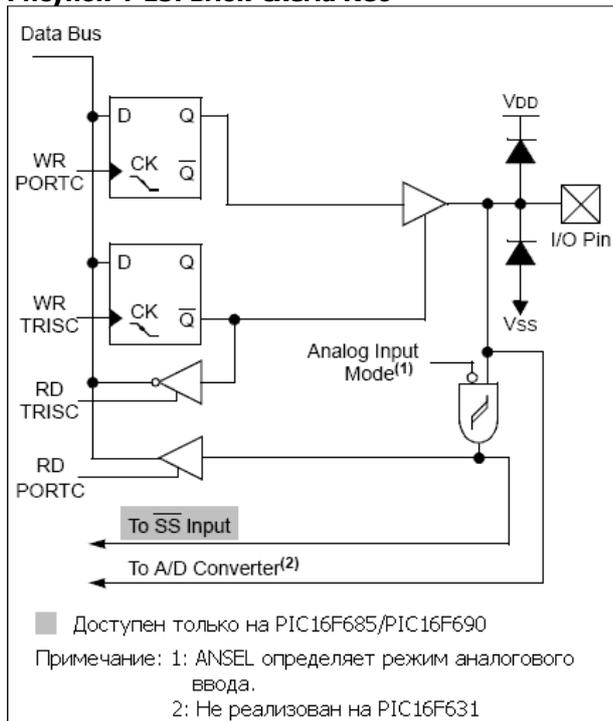
#### 4.5.8 RC7/AN9/SDO

RC7/AN9/SDO<sup>(1,2)</sup> конфигурируется для одной из следующих функций:

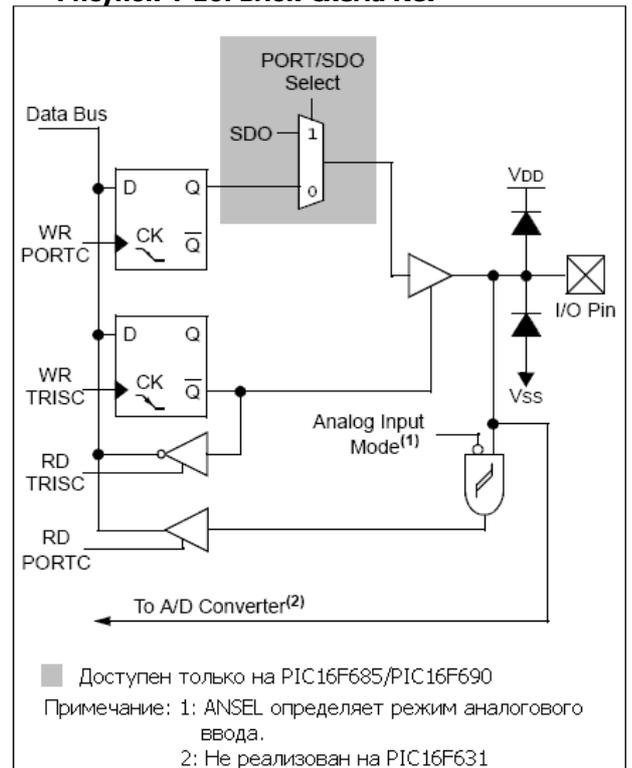
- универсальный I/O
- аналоговый вход для АЦП (кроме PIC16F631)
- вывод последовательных данных

**Примечание** 1: SDO доступен только PIC16F687/PIC16F689/PIC16F690.  
2: AN9 не реализован на PIC16F631.

**Рисунок 4-15: Блок-схема RC6**



**Рисунок 4-16: Блок-схема RC7**



**Таблица 4-3: Перечень регистров ассоциированных с PORTC**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
ANSELH	-	-	-	-	ANS11	ANS10	ANS9	ANS8	---- 1111	---- 1111
CCP1CON <sup>(2)</sup>	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
CM2CON0	C2ON	C2OUT	C2OE	C2POL	-	C2R	C2CH1	C2CH0	0000 -000	0000 -000
CM2CON1	MC1OUT	MC2OUT	-	-	-	-	T1GSS	C2SYNC	00-- --10	00-- --10
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
PSTRCON	-	-	-	STRSYNC	STRD	STRC	STRB	STRA	---0 0001	---0 0001
SRCON	SR1	SR0	C1SEN	C2REN	PULSS	PULSR	-	-	0000 00--	0000 00--
SSPCON <sup>(1)</sup>	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
VRCON	C1VREN	C2VREN	VRR	VP6EN	VR3	VR2	VR1	VR0	0000 0000	0000 0000

**Описание:** x = неизвестный, u = неизменный, - = не задействованный, читается как '0'. Затенённые ячейки не используются с PORTB.

**Примечание**    **1:** Только для PIC16F687/PIC16F689/PIC16F690.  
**2:** Только для PIC16F685/PIC16F690.

## 8.0 Модуль компаратора

Компараторы используются в интерфейсе между аналоговыми схемами и цифровыми схемами, сравнивают два аналоговых напряжения и обеспечивают цифровую индикацию их относительной величины. Компараторы часто используются при построении схем со смешанным сигналом, потому что они обеспечивают аналогово функциональное, независимое от программы выполнение. Модуль аналогового компаратора включает следующие параметры:

- Независимое управление компаратором
- Программируемый выбор входа
- Доступен как внутренний, так и внешний выход компаратора
- Программируется выходная полярность
- Прерывание по нарастанию
- Пробуждения из спящего режима
- Выключение ШИМ
- Ворота Timer1 (разрешение счёта)
- Синхронизация выхода со счётным входом Timer1
- SR защёлка
- Программируемое и фиксированное опорное напряжение

**Примечание:** Только компаратор C2 может быть связан с timer1.

### 8.1 Обзор компаратора

Одиночный компаратор показан на рис.8-1 вместе с графиком отношения между аналоговыми входными уровнями и цифровым выходом. Когда аналоговое напряжение  $V_{IN+}$  меньше чем аналоговое напряжение  $V_{IN-}$ , на выходе компаратора присутствует низкий дискретный уровень. Когда аналоговое напряжение  $V_{IN+}$  больше чем аналоговое напряжение  $V_{IN-}$ , на выходе компаратора присутствует высокий дискретный уровень.

**Рисунок 8-1: Одиночный компаратор**

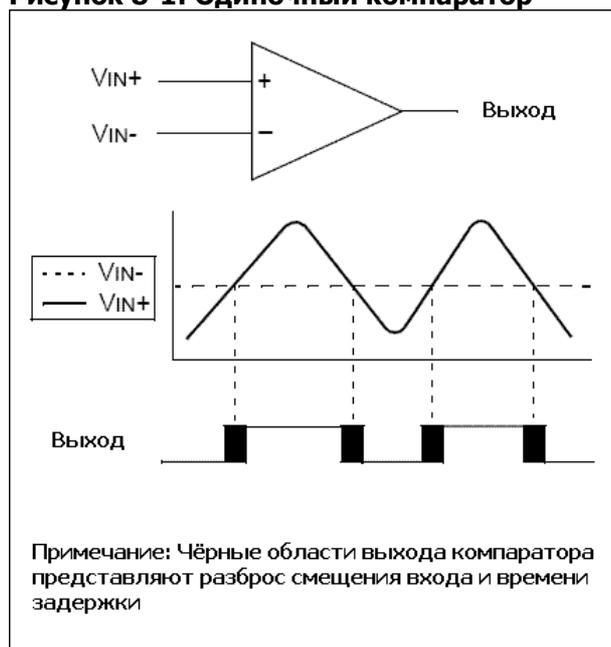


Рисунок 8-2: Упрощённая блок-схема компаратора C1

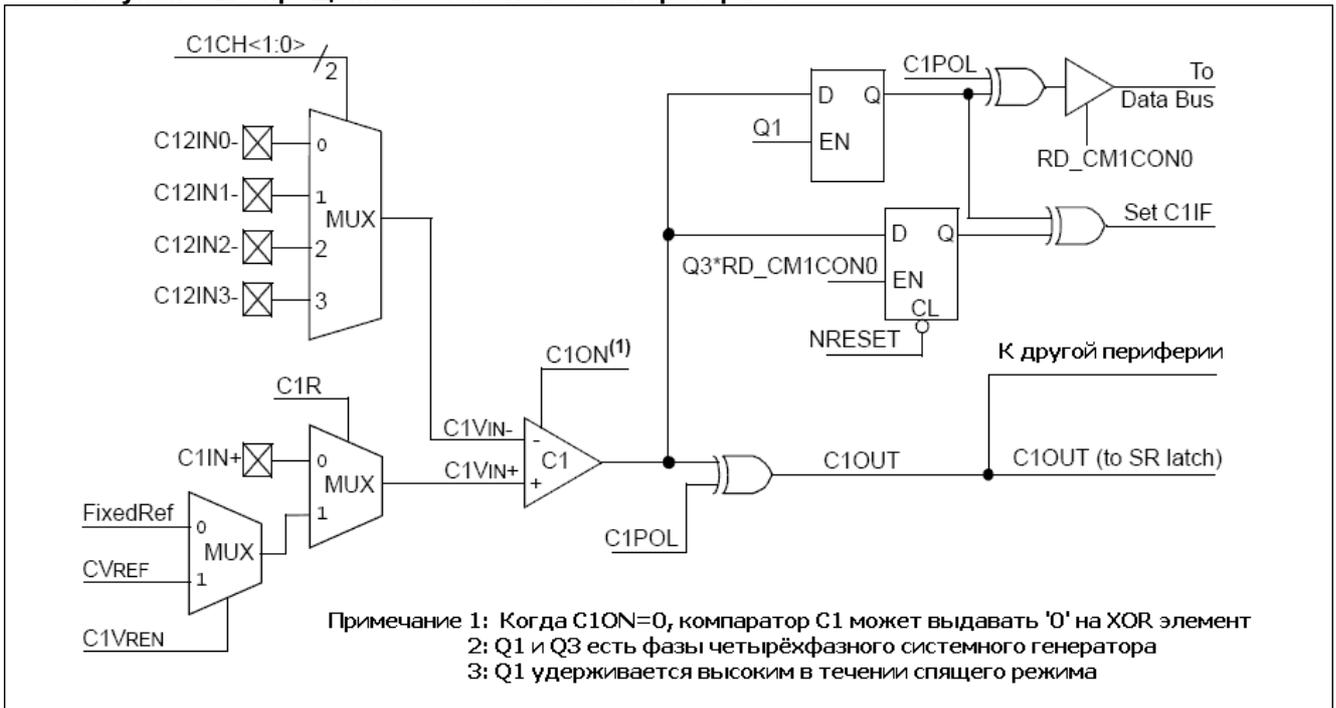
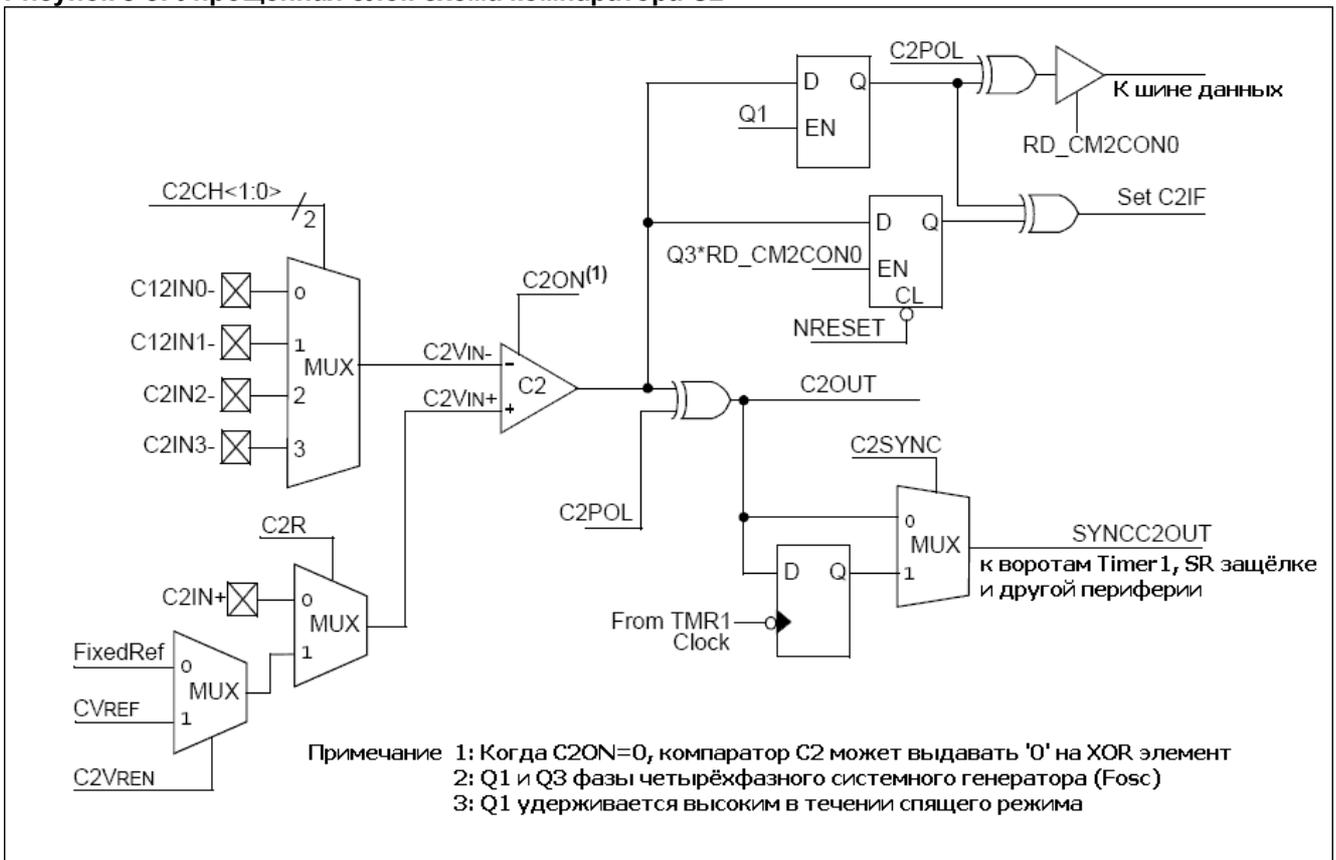


Рисунок 8-3: Упрощённая блок-схема компаратора C2



## 8.2 Управление компаратором

Каждый компаратор имеет отдельные регистры управления и конфигурации: CM1CON0 для компаратора C1 и CM2CON0 для компаратора C2. В дополнение, компаратор C2 имеет второй регистр управления, CM2CON1, для управления взаимодействием с Timer1 и для одновременного чтения обоих выходов компараторов.

Регистры CM1CON0 и CM2CON0 (смотри регистры 8-1 и 8-2, соответственно) содержат следующие биты управления и состояния:

- Разрешить
- Выбор входа
- Выбор опорного напряжения
- Выбор выхода
- Выходная полярность

### 8.2.1 Разрешить компаратор

Установка бита SxON в регистре CMxCON0 разрешает работу компаратора. Очистка бита SxON блокирует компаратор и переводит его в состояние минимального потребления тока.

### 8.2.2 Выбор входа компаратора

Биты SxCH<1:0> регистра CMxCON0 выбирают один из четырёх аналоговых входов для инвертирующего входа компаратора.

**Примечание:** Для использования SxIN+ и C12INx- ножек как аналоговых входов, соответствующие биты должны быть установлены в регистре ANSEL и соответствующие TRIS биты должны быть так же установлены для отключения выходных драйверов.

### 8.2.3 Выбор опорного напряжения компаратора

Установка бита SxR регистра CMxCON0 направляет внутреннее опорное напряжение или ножку аналогового ввода к не инвертирующему входу компаратора. Смотреть **Часть 8.9 “SR защёлка компаратора”** для большей информации на внутренний модуль опорного напряжения.

### 8.2.4 Выбор выхода компаратора

Мониторинг выхода компаратора может осуществляться чтением любого бита SxOUT регистра CMxCON0 или бита MSxOUT регистра CM2CON1. Чтобы сделать выход компаратора доступным для внешнего подключения, нужно выполнить следующие условия:

- Бит SxOE регистра CMxCON0 должен быть установлен
- Соответствующий TRIS бит должен быть очищен
- Бит SxON регистра CMxCON0 должен быть установлен

**Примечание** **1:** Бит SxOE аннулирует PORT защёлку данных. Установка бита SxON не имеет влияния на аннулирование порта.  
**2:** Внутренний выход компаратора защёлкивается с каждым циклом инструкции. Внешние выходы не защёлкиваются, если иначе не определено.

### 8.2.5 Полярность выхода компаратора

Инвертирование выхода компаратора функционально эквивалентно обмену входов компаратора. Полярность выхода компаратора может быть инвертирована через установку бита SxPOL регистра. Результатом очистки бита SxPOL будет не инвертированный выход.

В таблице 8-1 показаны выходные напротив входных условий, включая управление полярностью.

**TABLE 8-1: Состояния выхода компаратора через входные условия**

Входное состояние	SxPOL	SxOUT
SxVIN- > SxVIN+	0	0
SxVIN- < SxVIN+	0	1
SxVIN- > SxVIN+	1	1
SxVIN- < SxVIN+	1	0

## 8.3 Время ответа компаратора

Выход компаратора не определён для периода времени после изменения входного источника или выбора нового опорного напряжения. Этот период соответствует времени ответа. Время ответа компаратора отличается от времени установки опорного напряжения. Следовательно, оба этих времени определяют общее время ответа на изменение входа компаратора. Смотреть Спецификации компаратора и источника опорного напряжения в **Части 17.0 “Электрические спецификации”** для большей информации.

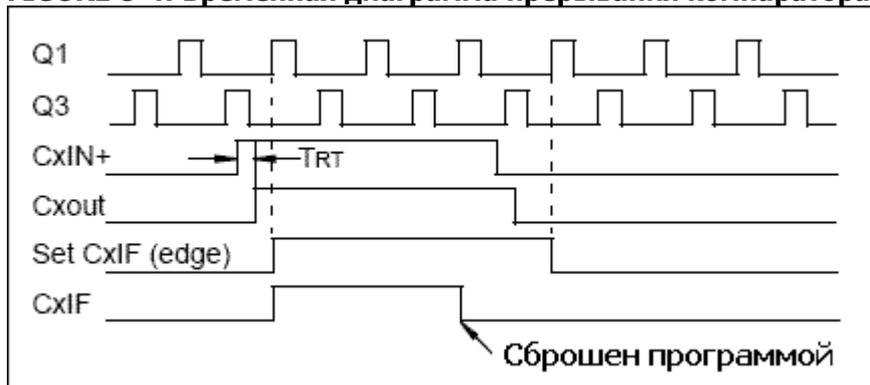
## 8.4 Управление прерыванием компаратора

Флаг прерывания компаратора может быть установлен каждый раз когда происходит изменение выходной величины компаратора. Изменение распознаётся посредством схемы несовпадений которая состоит из двух защёлки и сумматора по модулю два (смотреть рис. 8-2 и рис. 8-3). Одна защёлка корректируется в соответствии с выходным уровнем компаратора, когда читается регистр CMxCON0. Эта защёлка удерживает значение, пока не случится следующее чтение регистра CMxCON0 или сброс. Другая защёлка схемы несовпадения обновляется на каждый системный такт Q1. Условие несовпадения может происходить, когда изменение выхода компаратора синхронно через вторую защёлку на тактовом цикле Q1. В этот момент две защёлки несовпадения имеют противоположные выходные уровни которые обнаруживаются схемой исключающее или и подаётся на схему прерывания. Состояние несовпадения остаётся, пока регистр CMxCON0 не будет прочитан или выход компаратора вернётся в предыдущее состояние.

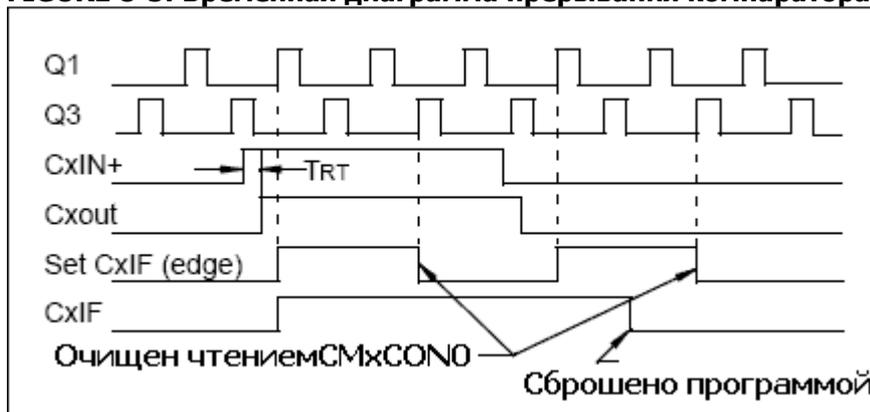
**Note 1:** Запись в регистр CMxCON0 может так же очистить состояние несовпадения, потому что все записи включают операцию чтения в начале цикла записи.  
**2:** Прерывания компаратора действуют корректно независимо от состояния CxOE.

Прерывания компаратора устанавливаются по фронту несовпадения и не уровню несовпадения. Это означает, что флаг прерывания может быть сброшен без дополнительного шага чтения или записи регистра CMxCON0 для очистки регистров несовпадения. Когда регистры несовпадения очищены, прерывание может происходить при возврате компаратора в предыдущее состояние, в противном случае прерывание не генерируется. Программа может нуждаться в дополнительной информации о состоянии выхода компаратора, как чтение регистров CMxCON0 или CM2CON1, для определения текущего изменения, которое происходит. Бит CxIF регистра PIR1 является флагом прерывания компаратора. Этот бит должен быть сброшен в программе как очистка в '0'. Поскольку возможна запись '1' в этот регистр, это может использоваться для генерации прерывания. Бит CxIE регистра PIE1 и биты PEIE и GIE регистра INTCON должны быть установлены для разрешения прерывания компаратора. Если любой из этих битов очищен, прерывание не разрешено, хотя бит CxIF регистра PIR1 устанавливается, если состояние прерывания возникает.

**FIGURE 8-4: Временная диаграмма прерывания компаратора без чтения W/O**



**FIGURE 8-5: Временная диаграмма прерывания компаратора с чтением CMxCON0**



**Note 1:** Если изменения в регистре CMxCON0 (CxOUT) происходят, когда операция чтения выполнена (начало цикла Q2), тогда флаг прерывания CxIF регистра PIR1 может не получить установку.  
**2:** When Так же, когда компаратор разрешён впервые, схема смещения в модуле компаратора может вызывать не верный выход из компаратора, пока схема смещения стабильна. Необходимо позволить около 1  $\mu$ s для урегулирования смещения, затем очистить состояние несовпадения и флаги прерывания, прежде чем разрешать прерывания компаратора.

## 8.5 Работа в течении спящего режима (Sleep)

Компаратор, если он разрешён перед входом в спящий режим, остаётся активным в течении этого режима. Дополнительный ток, потребляемый компаратором, показан в **Части 17.0 "Электрических характеристик"**. Если компаратор не используется в спящем режиме, потребление мощности может быть минимизировано, если перед входом в спящий режим компаратор выключить. Каждый компаратор выключается очисткой бита SxON в регистре CMxCON0.

Изменение на выходе компаратора может пробудить устройство из спящего режима. Для разрешения компаратору пробуждения устройства из спящего режима, бит SxIE регистра PIE1 и бит PEIE регистра INTCON должны быть установлены. Инструкция следующая за инструкцией Sleep всегда выполняется первой после выхода из спящего режима. Если бит GIE регистра INTCON так же установлен, устройство может затем выполнить подпрограмму обслуживания прерывания.

## 8.6 Эффект от сброса

Сброс устройства переводит регистры CMxCON0 и CM2CON1 в их состояние после сброса. Это переводит оба компаратора и источник опорного напряжения в выключенное состояние.

**Регистр 8-1: CM1CON0: Компаратора C1 регистр управления 0**

R/W-0	R-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
C1ON	C1OUT	C1OE	C1POL	-	C1R	C1CH1	C1CH0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается

W=бит записывается

U=не задействованный бит,  
читается как '0'

-n=Значение при POR

'1'=бит установлен

'0'=бит очищен

X=бит не определён

- Бит 7**      **C1ON:** Бит разрешения компаратора C1  
1 = Компаратор C1 разрешён  
0 = Компаратор C1 запрещён
- Бит 6**      **C1OUT:** Бит выхода компаратора C1  
Если C1POL = 1 (обратная полярность):  
C1OUT = 0 когда C1VIN+ > C1VIN-  
C1OUT = 1 когда C1VIN+ < C1VIN-  
Если C1POL = 0 (прямая полярность):  
C1OUT = 1 когда C1VIN+ > C1VIN-  
C1OUT = 0 когда C1VIN+ < C1VIN-
- Бит 5**      **C1OE:** Бит разрешения выхода компаратора C1  
1 = C1OUT представлен на C1OUT ножке<sup>(1)</sup>  
0 = C1OUT только внутренний
- Бит 4**      **C1POL:** Бит выбора полярности выхода компаратора C1  
1 = C1OUT инвертирующая логика  
0 = C1OUT не инвертирующая логика
- Бит 3**      **Не задействован:** Читается как '0'
- Бит 2**      **C1R:** Бит выбора опорного напряжения компаратора C1 (для не инвертирующего входа)  
1 = C1VIN+ подключен к выходу C1VREF  
0 = C1VIN+ подключен к C1IN+ ножке
- Биты 1-0**      **C1CH<1:0>:** Биты выбора канала компаратора C1  
00 = C1VIN- компаратор C1 подключен к ножке C12IN0-  
01 = C1VIN- компаратор C1 подключен к ножке C12IN1-  
10 = C1VIN- компаратор C1 подключен к ножке C12IN2-  
11 = C1VIN- компаратор C1 подключен к ножке C12IN3-

**Note 1:** Для выхода компаратора три следующих условия: C1OE = 1, C1ON = 1 и соответствующий PORT TRIS бит = 0.

**Регистр 8-2: CM2CON0: Компаратора C2 регистр управления 0**

R/W-0	R-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
C2ON	C2OUT	C2OE	C2POL	-	C2R	C2CH1	C2CH0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается

W=бит записывается

U=не задействованный бит,  
читается как '0'

-n=Значение при POR

'1'=бит установлен

'0'=бит очищен

X=бит не определён

- Бит 7**      **C2ON:** Бит разрешения компаратора C2  
1 = Компаратор C2 разрешён  
0 = Компаратор C2 запрещён
- Бит 6**      **C2OUT:** Бит выхода компаратора C2  
Если C2POL = 1 (обратная полярность):  
C2OUT = 0 когда C2VIN+ > C2VIN-  
C2OUT = 1 когда C2VIN+ < C2VIN-  
Если C2POL = 0 (прямая полярность):  
C2OUT = 1 когда C2VIN+ > C2VIN-  
C2OUT = 0 когда C2VIN+ < C2VIN-
- Бит 5**      **C2OE:** Бит разрешения выхода компаратора C2  
1 = C2OUT представлен на C2OUT ножке<sup>(1)</sup>  
0 = C2OUT только внутренний
- Бит 4**      **C2POL:** Бит выбора полярности выхода компаратора C2  
1 = C2OUT инвертирующая логика  
0 = C2OUT не инвертирующая логика
- Бит 3**      **Не задействован:** Читается как '0'
- Бит 2**      **C2R:** Бит выбора опорного напряжения компаратора C2 (для не инвертирующего входа)  
1 = C2VIN+ подключен к выходу C2VREF  
0 = C2VIN+ подключен к C2IN+ ножке
- Биты 1-0**      **C2CH<1:0>:** Биты выбора канала компаратора C2  
00 = C2VIN- компаратор C2 подключен к ножке C12IN0-  
01 = C2VIN- компаратор C2 подключен к ножке C12IN1-  
10 = C2VIN- компаратор C2 подключен к ножке C12IN2-  
11 = C2VIN- компаратор C2 подключен к ножке C12IN3-

**Note 1:** Для выхода компаратора три следующих условия: C2OE = 1, C2ON = 1 и соответствующий PORT TRIS бит = 0.

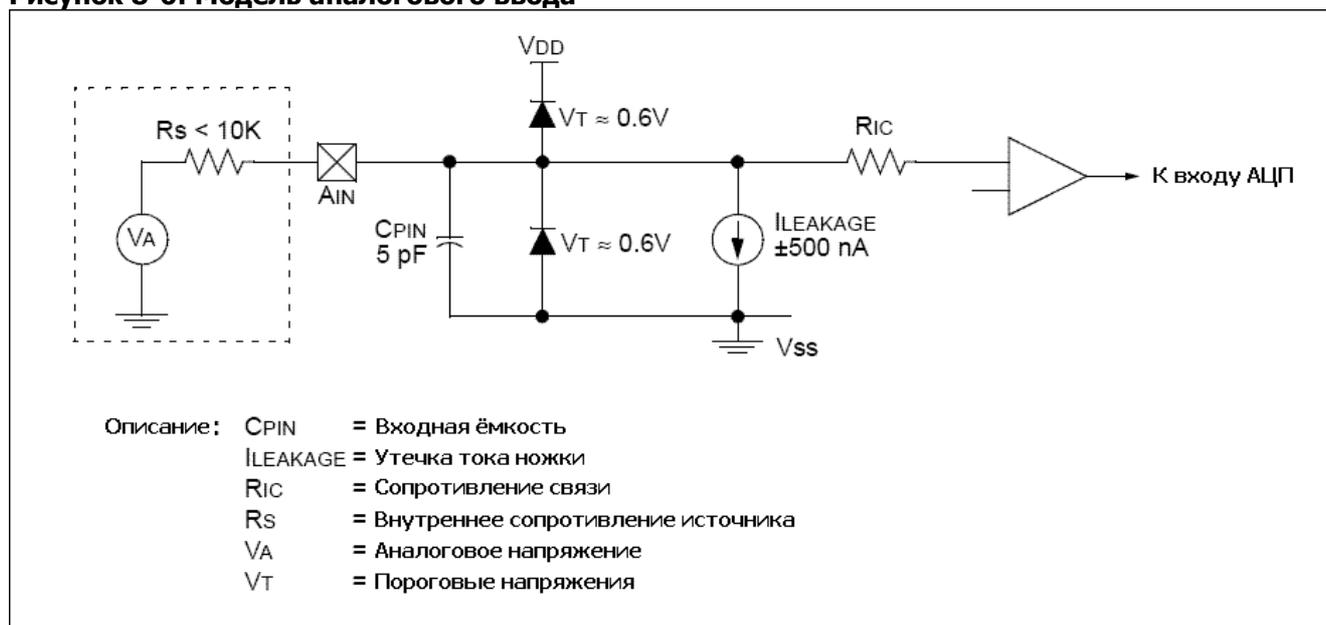
## 8.7 Соображения по поводу подключения аналогового ввода

Упрощённая схема аналогового ввода показана на рис. 8-6. Поскольку ножки аналогового ввода совмещены с цифровым вводом, они имеют обратно смещённые защитные диоды подключённые VDD и VSS. Следовательно, диапазон аналогового напряжения должен находиться между VSS и VDD. Если входное напряжение выходит из этого диапазона более чем на 0.6V в любом направлении, один из диодов открывается и происходит блокировка.

Максимальное внутреннее сопротивление источника 10 kΩ рекомендовано для аналоговых источников. Так же, любые внешние компоненты, подключённые к ножке аналогового ввода, такие как ёмкость или стабилитрон, должны иметь очень маленькую утечку тока, для минимизации погрешности.

- Примечание**
- 1: Когда читается регистр PORT, все ножки, сконфигурированные как аналоговые входы, будут читаться как '0'. Ножки, сконфигурированные как цифровые входы, преобразовывают аналоговые сигналы в соответствии со спецификацией ввода.
  - 2: Аналоговые уровни на любых ножках определённых как цифровой ввод, могут вызывать увеличенное поглощение тока буфером, чем это определено спецификацией.

**Рисунок 8-6: Модель аналогового ввода**



## 8.8 Дополнительные характеристики компаратора

Есть три дополнительные характеристики компаратора:

- Разрешение счёта Timer1 (ворота)
- Синхронизация выхода с Timer1
- Одновременное чтение выходов компараторов

### 8.8.1 Запуск TIMER1 компаратором C2

Эта возможность может быть использована для измерения длительности или интервала аналоговых событий. Очистка бита T1GSS регистра CM2CON1 может разрешить Timer1 увеличивать основание на выход компаратора C2. Это требует, чтобы таймер был включен и действие разрешено. Смотрите **Часть 6.0 "Модуль Timer1 с Gate Control"** для деталей.

Рекомендуется для синхронизации компаратора с Timer1 установкой бита C2SYNC, когда компаратор используется как источник управления для Timer1. Это гарантирует Timer1 не делать приращение если компаратор изменится в течении приращения.

### 8.8.2 Синхронизация выхода компаратора C2 к TIMER1

Выход компаратора C2 может быть синхронизирован с Timer1 установкой бита C2SYNC регистра CM2CON1. Когда синхронизация разрешена, выход C2 защёлкивается по падающему фронту тактового сигнала Timer1. Если с Timer1 используется предделитель, выход компаратора защёлкивается после функции предделения. Для предотвращения гонки состояний, выход компаратора защёлкивается по падающему фронту тактового сигнала Timer1, а Timer1 приращается на нарастающем фронте этого тактового сигнала. Смотрите блок-схему компаратора (Рисунок 8-3) и блок-схему Timer1 (Рисунок 6-1) для большей информации.

### 8.8.3 Одновременное чтение выхода компаратора

Биты MC1OUT и MC2OUT регистра CM2CON1 есть зеркальная копия обоих выходов компараторов. Способность одновременного чтения выходов из одного регистра устраняет временной перекос возникающий в случае чтения отдельных регистров.

**Note 1:** Получение состояния C1OUT или C2OUT через чтение CM2CON1 делают не влиять, то что прерывания от компараторов расположены в различных регистрах.

**Регистр 8-3: CM2CON1: Регистр управления 1 компаратором C2**

R-0	R-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0
MC1OUT	MC2OUT	-	-	-	-	T1GSS	C2SYNC
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен
		X=бит не определён

**Бит 7**      **MC1OUT:** Зеркальная копия бита C1OUT

**Бит 6**      **MC2OUT:** Зеркальная копия бита C2OUT

**Бит 5-2**    **Не задействованы:** Читается как '0'

**Бит 1**      **T1GSS:** Бит выбора источника управления Timer1<sup>(1)</sup>

1 = T1G источник управления Timer1

0 = SYNCC2OUT источник управления Timer1.

**Бит 0**      **C2SYNC:** Бит синхронизации выхода компаратора C2<sup>(2)</sup>

1 = Выход синхронизируется по падающему фронту тактового сигнала Timer1

0 = Выход не синхронизируется

**Примечание 1:** Ссылка на **Часть 6.6 "Управление Timer1"**.

**2:** Ссылка на рисунок 8-3.

## 8.9 SR защёлка компаратора

Модуль SR защёлки обеспечивает дополнительное управление выходами компаратора. Модуль состоит из одной SR защёлки и выходных мультиплексоров. SR защёлка может быть установлена, сброшена или переключена выходами компаратора. SR защёлка может так же установлена или сброшена независимо от выхода компаратора, через биты управления в регистре управления SRCON. Выходной мультиплексор выбирает независимо защёлки выходы или выходы компаратора быть направлены в логику порта I/O port для возможного вывода в ножку.

### 8.9.1 Работа защёлки

Защёлка представляет собой RS-триггер, который делает не зависимый включить тактовый источник. Каждый S и R входы активно высокие. Каждый вход защёлки подключен к выходу компаратора и программно управляемому генератору тактовых импульсов. Защёлка может быть установлена битами C1OUT или PULSS регистра SRCON. Защёлка может быть сброшена битами C2OUT или PULSR регистра SRCON. В защёлке сброс преобладает, следовательно, если оба S и R входы будут высокими, защёлка перейдет в сброшенное состояние. Оба бита PULSS и PULSR само сбрасываемые, что означает, что единственная запись в любой из битов, всё, что необходимо для операции полной установки или сброса защёлки.

### 8.9.2 Выход защёлки

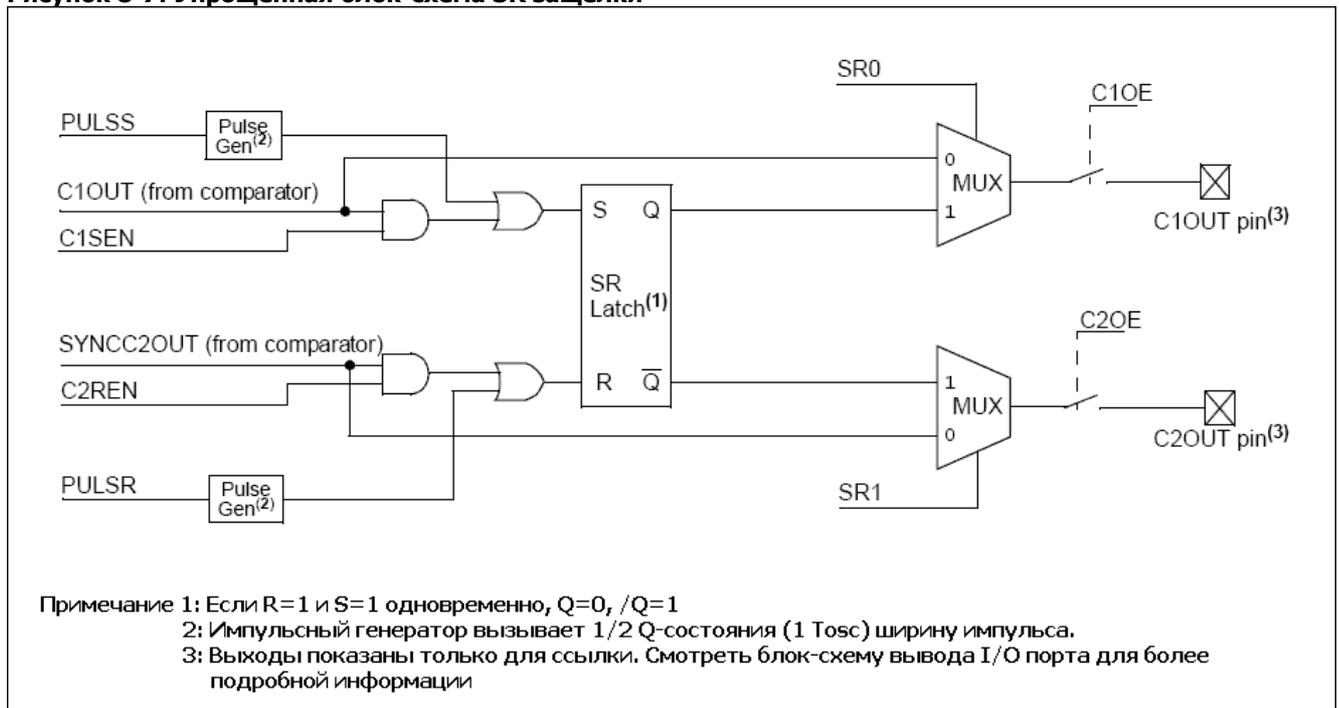
Биты SR<1:0> регистра SRCON управляют выходным мультиплексором защёлки и определяют четыре возможные выходные конфигурации. В этих четырёх конфигурациях порт I/O CxOUT логически подключен к:

- C1OUT и C2OUT
- C1OUT и SR защёлка /Q
- C2OUT и SR защёлка Q
- SR защёлка Q и /Q

После любого сброса, выходная конфигурация по умолчанию есть открытый C1OUT и C2OUT режим. Это поддерживает совместимость с устройствами, которые не имеют SR защёлки.

Прилагаемые биты TRIS соответствующие портам должны быть очищены для разрешения выходных драйверов вывода порта. Дополнительно, бит разрешения выхода компаратора SxOE регистра CMxCON0 может быть установлен в заказывать компаратора или защёлки выходы будут доступны на выходных ножках. Конфигурация защёлки разрешает состояния полной независимости разрешённых состояний для компараторов.

**Рисунок 8-7: Упрощённая блок-схема SR защёлки**



**Регистр 8-4: SRCON: Регистр управления SR защёлкой**

R/W-0	R/W-0	R/W-0	R/W-0	R/S-0	R/S-0	U-0	U-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>	S = бит только устанавливается		
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

- Бит 7**      **SR1:** Бит конфигурации SR защёлки<sup>(2)</sup>  
1 = C2OUT на ножке выход /Q защёлки  
0 = C2OUT на ножке выход компаратора C2
- Бит 6**      **SR0:** Бит конфигурации SR защёлки<sup>(2)</sup>  
1 = C1OUT на ножке выход Q защёлки  
0 = C1OUT на ножке выход компаратора C1
- Бит 5**      **C1SEN:** Бит разрешения установки от C1  
1 = Выход компаратора C1 устанавливает SR защёлку  
0 = Выход компаратора C1 не оказывает влияния на SR защёлку
- Бит 4**      **C2REN:** Бит разрешения сброса от C2  
1 = Выход компаратора C2 сбрасывает SR защёлку  
0 = Выход компаратора C2 не оказывает влияния на SR защёлку
- Бит 3**      **PULSS:** Бит импульса установки SR защёлки  
1 = Запускает генератор импульса установки SR защёлки. Бит немедленно сбрасывается аппаратно.  
0 = Не запускает генератор импульса
- Бит 2**      **PULSR:** Бит импульса сброса SR защёлки  
1 = Запускает генератор импульса сброса SR защёлки. Бит немедленно сбрасывается аппаратно.  
0 = Не запускает генератор импульса
- Бит 1-0**    **Не используется:** Читается как '0'

**Примечание**    **1:** Бит SxOUT в регистре CMxCON0 всегда отражает актуальный выход компаратора (не уровень на ножке), независимо от операции SR защёлки.  
**2:** Разрешает SR защёлки выход на ножку, соответствующие SxOE и TRIS биты должны быть правильно конфигурированы.

## 8.10 Опорное напряжение компаратора

Модуль опорного напряжения непосредственно обеспечивает генерацию опорного напряжения для компаратора.

Следующие характеристики доступны:

- Независимость для операций компаратора
- Два диапазона на 16 уровней напряжения
- Выход привязан к VSS
- Ratiometric с VDD
- Фиксированная ссылка (0.6)

Регистр VRCON (Register 8-5) управления модулем опорного напряжения показан на рисунке 8-8.

### 8.10.1 Независимые операции

Опорное напряжение компаратора не зависит от конфигурации компаратора. Установка бита VREN регистра VRCON может разрешить опорное напряжение.

### 8.10.2 Выбор выходных напряжений

CVREF опорное напряжение имеет 2 области, с 16 уровнями напряжения в каждой области. Выбор области управляется битом VRR регистра VRCON. 16 уровней устанавливаются битами VR<3:0> регистра VRCON.

CVREF выходное напряжение определяется следующим уравнением:

#### Уравнение 8-1: CVREF выходное напряжение

$VRR = 1$  (нижняя область):

$$CVREF = (VR<3:0>/24) \times VDD$$

$VRR = 0$  (верхняя область):

$$CVREF = (VDD/4) + (VR<3:0> \times VDD/32)$$

Полная область от Vss до Vdd не может быть реализована due к конструкции модуля. Смотреть рисунок 8-8.

### 8.10.3 Выход привязан к VSS

Выходное напряжение CVREF может быть установлено к VSS с небольшим потреблением, конфигурируя VRCON, как следует:

- VREN=0
- VRR=1
- VR<3:0>=0000

Это позволяет компаратору обнаруживать пересечение с нулём пока CVREF модуль не поглощает дополнительный ток.

### 8.10.4 Выход RATIOMETRIC к VDD

Опорное напряжение компаратора быть производной от VDD и следовательно, в выходном напряжении CVREF могут присутствовать колебания VDD. Тест абсолютной точности опорного напряжения компаратора можно найти в **Части 17.0 "Электрических характеристик"**.

### 8.10.5 Фиксированное опорное напряжение

Независимый от VDD, фиксированный источник опорного напряжения с номинальным выходным напряжением 0.6V. Эта опора может быть разрешена установкой бита VP6EN регистра VRCON в '1'. Эта опора всегда разрешена, когда активен осциллятор HFINTOSC.

### 8.10.6 Период стабилизации источника фиксированного опорного напряжения.

После того, как модуль фиксированного опорного напряжения разрешён, потребуется некоторое время, чтобы опора и схема усилителя стабилизировались. Программа пользователя должна включать небольшую программную задержку, позволяющую модулю выйти в режим. Смотреть раздел Электрических характеристик для требований минимальной задержки.

### 8.10.7 Выбор опорного напряжения

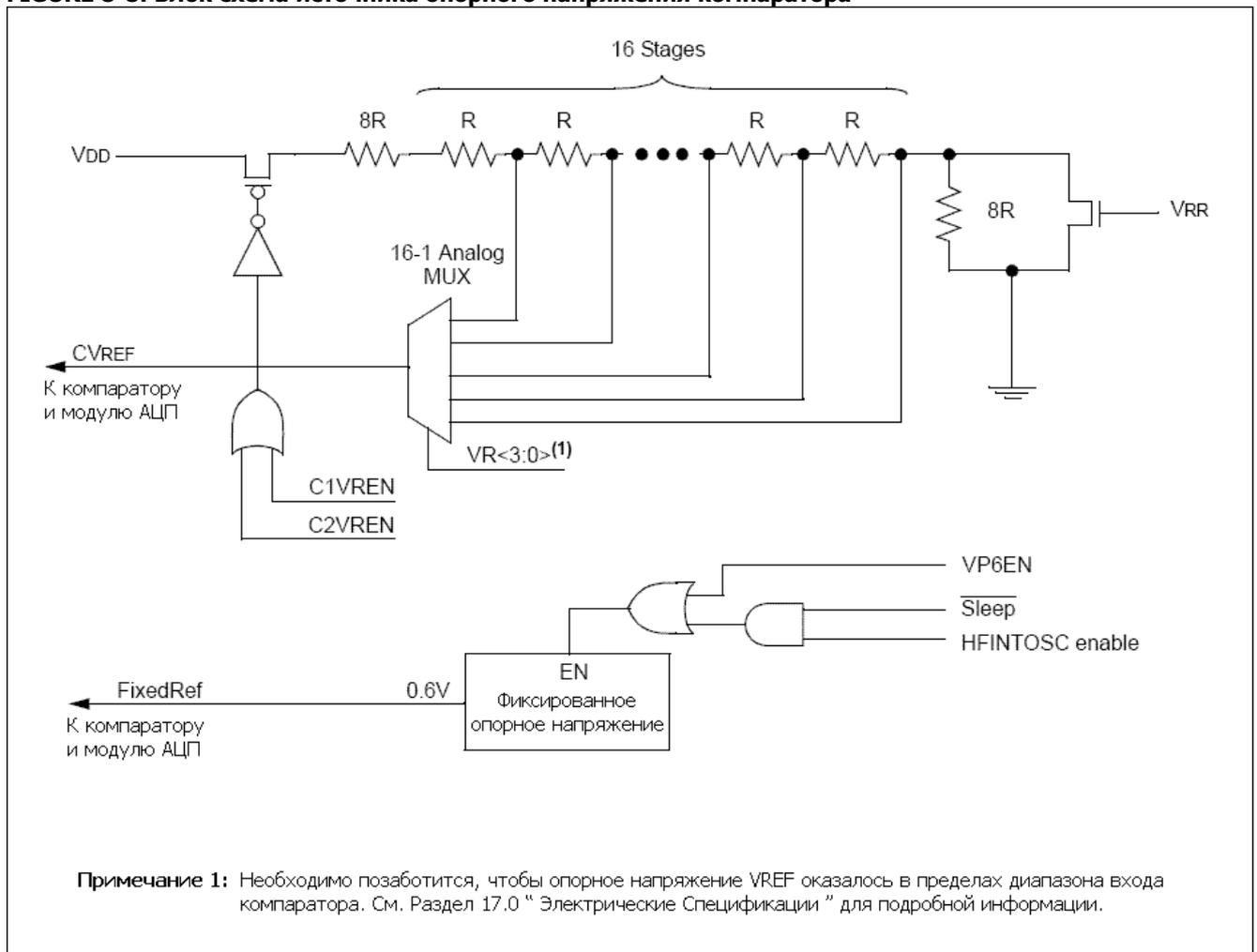
Мультиплексор на выходе модуля опорного напряжения позволяет выбрать любое CVREF или фиксированное опорное напряжение для использования с компаратором.

Установка бита C1VREN регистра VRCON разрешает подачу тока в CVREF делитель напряжения и выбор CVREF напряжения для компаратора C1. Очистка бита C1VREN выбирает фиксированное напряжение для использования с компаратором C1.

Установка бита C2VREN регистра разрешает подачу тока в CVREF делитель напряжения и выбор CVREF напряжения для компаратора C2. Очистка бита C2VREN выбирает фиксированное напряжение для использования с компаратором C2.

Когда оба бита C1VREN и C2VREN очищены, ток в CVREF делитель напряжения не подаётся, что минимизирует потребляемую мощность периферией опорного напряжения.

**FIGURE 8-8: Блок схема источника опорного напряжения компаратора**



**Регистр 8-5: VRCON: Регистр управления опорным напряжением**

R/W-0	R/W-0	R/W-0	R/W-0	R/S-0	R/S-0	U-0	U-0
C1VREN	C2VREN	VRR	VP6EN	VR3	VR2	VR1	VR0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>	S =бит только устанавливается	U=не задействованный бит, читается как '0'
R=бит читается	W=бит записывается	'0'=бит очищен
-n=Значение при POR	'1'=бит установлен	X=бит не определён

- Бит 7** **C1VREN:** Бит разрешения опорного напряжения для компаратора C1  
 1 = Схема CVREF включена и подключена к C1VREF входу компаратора C1  
 0 = Фиксированное напряжение 0.6 V подключено к C1VREF входу компаратора C1
- Бит 6** **C2VREN:** Бит разрешения опорного напряжения для компаратора C2  
 1 = Схема CVREF включена и подключена к C2VREF входу компаратора C2  
 0 = Фиксированное напряжение 0.6 V подключено к C2VREF входу компаратора C2
- Бит 5** **VRR:** Бит CVREF выбора области опорного напряжения  
 1 = Нижняя область  
 0 = Верхняя область
- Бит 4** **VP6EN:** Бит разрешения опоры 0.6V  
 1 = Разрешить  
 0 = Запретить
- Биты 3-0** **VR<3:0>:** Выбор значения опорного напряжения CVREF  $0 \leq VR<3:0> \leq 15$   
 Когда VRR = 1: CVREF =  $(VR<3:0>/24) * VDD$   
 Когда VRR = 0: CVREF =  $VDD/4 + (VR<3:0>/32) * VDD$

**Таблица 8-2: Перечень регистров ассоциированных с компаратором и модулем опорного напряжения**

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Значение на POR, BOR	Значение на все другие сбросы
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CM1CON0	C1ON	C1OUT	C1OE	C1POL	-	C1R	C1CH1	C1CH0	0000 -000	0000 0000
CM2CON0	C2ON	C2OUT	C2OE	C2POL	-	C2R	C2CH1	C2CH0	0000 -000	0000 0000
CM2CON1	MC1OUT	MC2OUT	-	-	-	-	T1GSS	C2SYNC	00-- --10	00-- --10
INTCON	GIE	PEIE	T0IE	INTE	RABIE	T0IF	INTF	RABIF	0000 000x	0000 000x
PIE2	OSFIE	C2IE	C1IE	EEIE	-	-	-	-	0000 ----	0000 ----
PIR2	OSFIF	C2IF	C1IF	EEIF	-	-	-	-	0000 ----	0000 ----
PORTA	-	-	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
REFCON	-	-	BGST	VRBB	VREN	VROE	CVROE	-	--00 000-	--00 000-
SRCON	SR1	SR0	C1SEN	C2REN	PULSS	PULSR	-	-	0000 00--	0000 00--
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
VRCON	C1VREN	C2VREN	VRR	VP6EN	VR3	VR2	VR1	VR0	0000 0000	0000 0000

**Описание:** x не определён, u = неизменный, - = не задействован и читается как '0'. Затенённые клетки не используются для компаратора.

## 9.0 Модуль аналогово-цифрового преобразователя (ADC)

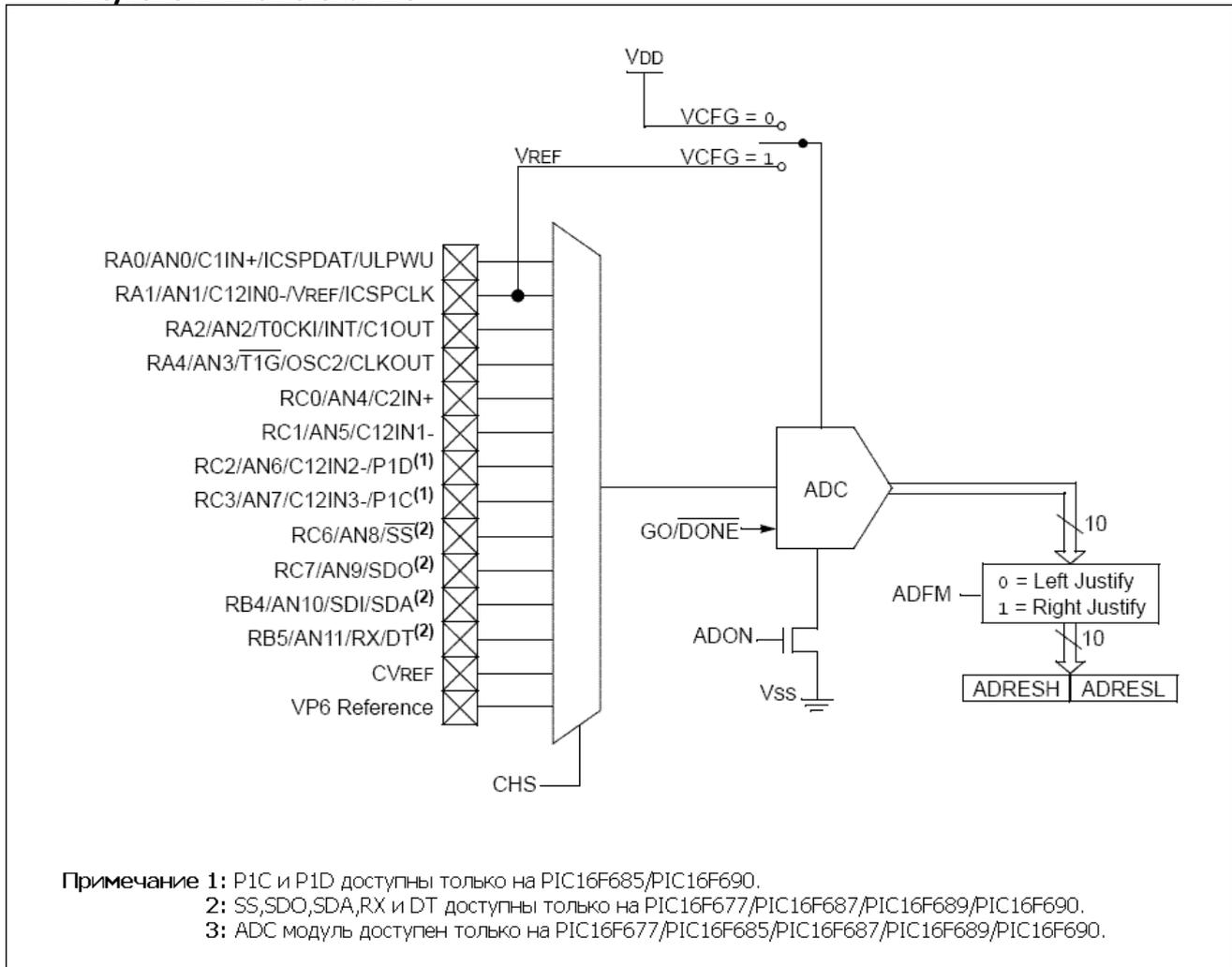
Аналогово-цифровой преобразователь (ADC) позволяет преобразовать аналоговый входной сигнал в 10-ти разрядный двоичный код. Это устройство использует аналоговые входы, которые с помощью мультиплексора подключаются к единственной схеме выборки и хранения. Выход схемы выборки и хранения подключен к входу конвертера. Конвертер выдаёт 20-ти разрядный двоичный результат путём последующих аппроксимаций и запоминает результат преобразования в ADC регистре результата (ADRESH и ADRESL).

**Примечание:** Модуль ADC имеется только в устройствах PIC16F677/PIC16F685/PIC16F687/PIC16F689/PIC16F690.

В качестве опорного напряжения ADC можно выбирать программно VDD или внешнее опорное напряжение, приложенное к ножкам. ADC может генерировать прерывание после завершения преобразования. Это прерывание может использоваться для пробуждения контроллера из спящего режима.

На рис. 9-1 показана блок-схема ADC.

**Рисунок 9-1 Блок-схема ADC**



### 9.1 Конфигурация ADC

При конфигурации и использовании ADC должны быть решены следующие функции:

- Конфигурация порта
- Выбор канала
- Выбор источника опорного напряжения ADC
- Тактовый источник преобразования ADC
- Управление прерыванием
- Форматирование результата

#### 9.1.1 Конфигурация порта

ADC может быть использован для преобразования аналоговых и дискретных сигналов. Когда преобразуется аналоговый сигнал, ножка I/O должна быть сконфигурирована как аналоговая установкой соответствующих TRIS и ANSEL битов. Смотреть соответствующую секцию порта для более подробной информации.

**Примечание:** Аналоговое напряжение на любой ножке, которая определена как цифровой ввод, может привести к повышенному потреблению тока буфером.

#### 9.1.2 Выбор канала

Биты CHS регистра ADCON0 определяют какой канал подключен к схеме выборки и хранения. Когда меняются канала, требуется задержка перед началом следующего преобразования. Смотреть часть 9.2 "Работа ADC" для большей информации.

### 9.1.3 Опорное напряжение ADC

Бит VCFG регистра ADCON0 обеспечивает управление позитивным источником опорного напряжения. В качестве позитивного источника опорного напряжения можно использовать VDD или внешний источник опорного напряжения. Отрицательный источник опорного напряжения всегда связан с потенциалом земли.

### 9.1.4 Тактовый источник преобразования ADC

Тактовый источник преобразования выбирается программно через биты ADCS регистра ADCON1. Там возможно семь вариантов частоты:

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC (внутренний RC-генератор)

Полное время преобразования одного бита называется TAD. Одно полное 10-битное преобразование требует 11 TAD периодов, как показано на рис. 9-2. Для корректного преобразования, должно удовлетворять требованиям. Смотреть требования A/D преобразования в части 17.0 "Электрические спецификации" для большей информации. В таблице 9-1 приводятся примеры подходящего выбора частоты ADC.

**Примечание:** Если используется FRC, любые изменения системной тактовой частоты могут изменить тактовую частоту ADC, что может повлиять на результат ADC.

**Таблица 9-1: Период частоты ADC (TAD) в зависимости от рабочей частоты устройства (VDD > 3.0V)**

Период частоты ADC (TAD)		Рабочая частота устройства (Fosc)			
Источник частоты ADC	ADCS<2:0>	20МГц	8МГц	4МГц	1МГц
Fosc/2	000	100 ns <sup>(2)</sup>	250 ns <sup>(2)</sup>	100 ns <sup>(2)</sup>	2.0 us
Fosc/4	100	200 ns <sup>(2)</sup>	500 ns <sup>(2)</sup>	100 ns <sup>(2)</sup>	4.0 us
Fosc/8	001	400 ns <sup>(2)</sup>	1.0 us <sup>(2)</sup>	2.0 us	8.0 us <sup>(3)</sup>
Fosc/16	101	800 ns <sup>(2)</sup>	2.0 us	4.0 us	16.0 us <sup>(3)</sup>
Fosc/32	010	1.6 us	4.0 us	8.0 us <sup>(3)</sup>	32.0 us <sup>(3)</sup>
Fosc/64	110	3.2 us	8.0 us <sup>(3)</sup>	16.0 us <sup>(3)</sup>	64.0 us <sup>(3)</sup>
FRC	X11	2-6 us <sup>(1,4)</sup>	2-6 us <sup>(1,4)</sup>	2-6 us <sup>(1,4)</sup>	2-6 us <sup>(1,4)</sup>

**Описание:** Затемнённые ячейки находятся за пределами рекомендованного диапазона.

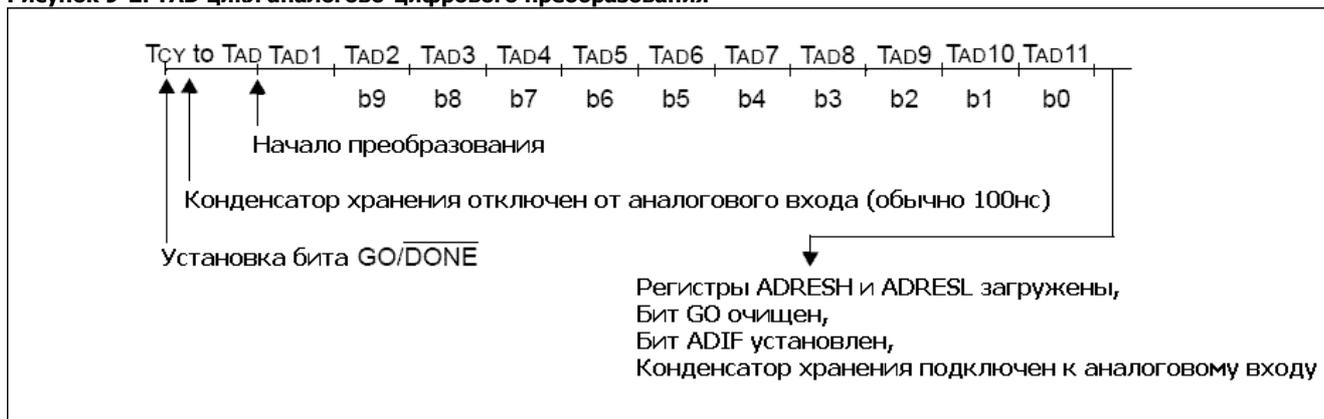
**Примечание 1:** Источник FRC обеспечивает типовую длительность TAD равную 4 us для VDD > 3.0V.

2: Эти значения меньше минимально возможной длительности TAD.

3: Рекомендуется выбрать другой тактовый источник для получения меньшего времени преобразования.

4: Когда тактовая частота устройства выше 1МГц, тактовый источник FRC рекомендуется, если только предполагается использовать ADC в спящем режиме.

**Рисунок 9-2: TAD цикл аналогово-цифрового преобразования**



### 9.1.5 Прерывания

Модуль ADC может генерировать прерывание по окончании аналогово-цифрового преобразования. Бит ADIF в регистре PIR1 является Флагом прерывания ADC. При возникновении прерывания от ADC этот бит устанавливается. Бит ADIF должен быть очищен программно.

**Примечание:** Бит ADIF устанавливается при завершении каждого преобразования, не зависимо от того разрешено прерывание или нет.

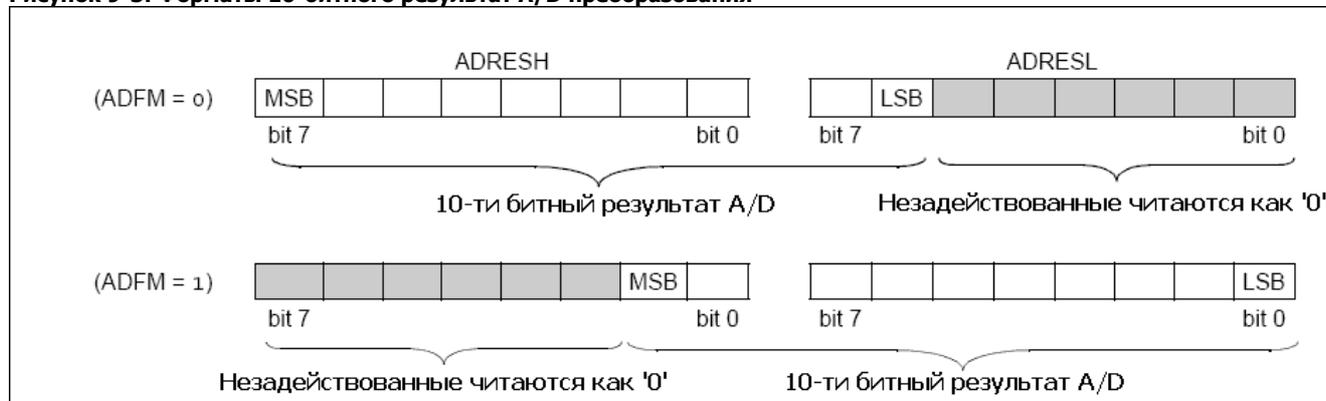
Прерывание может генерироваться как в нормальном, так и в спящем режимах функционирования устройства. Если устройство находится в спящем режиме, то прерывание может пробудить его. После пробуждения выполняется инструкция, следующая за SLEEP в программном коде. Глобальное прерывание должно быть запрещено, если пользователь хочет продолжать

выполнение кода после пробуждения. Если прерывание не будет запрещено, то управление будет передано программе обработки прерываний. Смотрите часть 9.1.5 "Прерывания" для большей информации.

### 9.1.6 Форматирование результата

10-битный результат A/D преобразования может выдаваться в двух форматах, левое выравнивание и правое выравнивание. Бит ADFM регистра ADCON0 управляет выходным форматом. На рис. 9-3 показаны два выходных формата.

**Рисунок 9-3: Форматы 10-битного результат A/D преобразования**



## 9.2 Функционирование ADC

### 9.2.1 Начало преобразования

Для разрешения модуля ADC, бит ADON регистра ADCON0 должен быть установлен в '1'. Установка в "1" бита GO/DONE регистра ADCON0 начинает аналогово-цифровое преобразование.

**Примечание:** Бит GO/DONE не должен быть установлен в той же инструкции, которая включает ADC. Обратитесь к части 9.2.6 "Процедура A/D преобразования".

### 9.2.2 Завершение преобразования

Когда завершится преобразование, модуль ADC:

- Очистит бит GO/DONE
- Установит бит флага ADI
- Загрузит в регистр ADRESH:ADRESL новый результат преобразования

### 9.2.3 Остановка преобразования

Если преобразование нужно остановить до его завершения, то бит GO/DONE можно очистить программно. В регистр ADRESH:ADRESL не будет загружен результат частичного преобразования выборки. В этом случае этот регистр будет содержать значение предыдущего преобразования. В добавок потребуется задержка в 2 TAD перед началом следующего преобразования. В течении этой задержки вход автоматически начинает выбор канала.

**Примечание:** Сброс устройства переводит все регистры в их состояние после сброса. В этом случае модуль ADC будет выключен все незаконченные преобразования будут остановлены.

### 9.2.4 Работа ADC в течении SLEEP

Модуль ADC может работать в течении спящего режима. Для этого требуется в качестве источника тактовых импульсов выбрать FRC. Когда выбран тактовый источник FRC, ADC ожидает одну дополнительную инструкцию перед началом преобразования. Это позволяет выполнить инструкцию SLEEP, что уменьшит системный шум в течении преобразования. Если прерывание от ADC разрешено, устройство пробудится после завершения преобразования. Если прерывание ADC запрещено, модуль ADC просто выключится после завершения преобразования, хотя бит ADON останется установленным.

Когда в качестве тактового источника ADC выбран не FRC, инструкция SLEEP вызывает прекращение преобразование и модуль ADC выключается, хотя бит ADON остаётся установленным.

### 9.2.5 Специальный триггер(запуск) события

Специальный триггер(запуск) события ECCP позволяет периодически производить измерения с ADC без вмешательства программы. Когда событие(триггер) происходит, бит GO/DONE устанавливается аппаратно и счётчик Timer1 сбрасывается в ноль. Использование специального триггера(запуска) события не гарантирует соответствующей синхронизации ADC. В этом случае пользователь должен убедиться, что синхронизация ADC соответствует требованиям. Смотрите часть 11.0 "Раширенный модуль захвата/сравнения/PWM" для большей информации.

### 9.2.6 Процедура A/D преобразования

Это пример процедуры использования ADC для выполнения аналогово-цифрового преобразования:

1. Конфигурировать порт:
  - Отключить ножку выходного драйвера (смотреть регистр TRIS)
  - Конфигурировать ножку, как аналоговую
2. Конфигурировать модуль ADC:
  - Выбрать источник тактирования ADC преобразователя
  - Конфигурировать источник опорного напряжения
  - Выбрать входной канал ADC
  - Выбрать формат результата

- Включить модуль ADC
3. Конфигурировать прерывание от ADC (опционально):
    - Очистить флаг прерывания ADC
    - Разрешить прерывание ADC
    - Разрешить периферийные прерывания
    - Разрешить глобальные прерывания<sup>(1)</sup>
  4. Ожидать время требуемое для выборки<sup>(2)</sup>.
  5. Начать преобразование, установив бит GO/DONE.
  6. Ожидать завершения преобразования ADC одним из следующих способов:
    - Опрашивая бит GO/DONE
    - Ожидая прерывания ADC (прерывания разрешены)
  7. Read ADC Result
  8. Очистить флаг прерывания ADC (необходимо, если прерывания разрешены).

**Примечание** 1: Глобальные прерывания могут быть запрещены, если пользователь пробудиться из спящего режима и продолжить выполнение встроенного кода.

2: Смотреть часть 9.3 "A/D требования выборки".

### Пример 9-1: A/D преобразование

```

; Этот кодовый блок конфигурирует ADC
; для опроса, Vdd опорное напряжение, источник тактирования Frc
; и AN0 вход.
;
; Старт преобразования & опрос для завершения
; включены.
;
    BANKSEL ADCON1          ;
    MOVLW B'01110000'      ; ADC Frc clock
    MOVWF ADCON1           ;
    BANKSEL TRISA          ;
    BSF TRISA,0            ; Настроить RA0 на ввод
    BANKSEL ANSEL          ;
    BSF ANSEL,0            ; Настроить RA0 как аналоговый
    BANKSEL ADCON0         ;
    MOVLW B'10000001'      ; Правое выравнивание,
    MOVWF ADCON0           ; Vdd Vref, AN0, On
    CALL SampleTime        ; Задержка выборки
    BSF ADCON0,GO          ; Старт преобразования
    BTFSC ADCON0,GO        ; Преобразование завершено?
    GOTO $-1               ; Нет, тестировать снова
    BANKSEL ADRESH         ;
    MOVF ADRESH,W          ; Читать два верхних бита
    MOVWF RESULTHI         ; запомнить в области GPR
    BANKSEL ADRESL         ;
    MOVF ADRESL,W          ; Читать нижних 8 бит
    MOVWF RESULTLO        ; Запомнить в области GPR
  
```

### 9.2.7 Определение регистров ADC

Следующие регистры используются для управления работой ADC.

#### Регистр 9-1: ADCON0: 0-й регистр управления A/D

R/W-0	R/W-0						
ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/-DONE	ADON
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

#### Описание:

R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'
-n=Значение при POR	'1'=бит установлен	'0'=бит очищен
		X=бит не определён

**Бит 7**     **ADFM:** Бит выбора формата результата A/D преобразования

1 = правое выравнивание

0 = левое выравнивание

**Бит 6**     **VCFG:** Бит источника опорного напряжения

1 = Vref ножка

0 = Vdd

**Биты 5-2**   **CHS<3:0>:** Биты выбора аналогового канала

0000 = AN0

0001 = AN1

0010 = AN2

0011 = AN3

0100 = AN4

0101 = AN5

0110 = AN6

0111 = AN7

1000 = AN8  
 1001 = AN9  
 1010 = AN10  
 1011 = AN11  
 1100 = CVref  
 1101 = Опорное напряжение 0.6В  
 1110 = Зарезервировано. Не используется  
 1111 = Зарезервировано. Не используется

- Бит 1** **GO-DONE:** Бит состояния A/D преобразователя  
 1 = A/D преобразование в процессе. Установка этого бита запускает цикл A/D преобразования. Этот бит автоматически аппаратно сбрасывается, когда A/D преобразование завершено.  
 2 = A/D преобразование завершено/не в процессе
- Бит 0** **ADON:** Бит разрешения ADC  
 1 = ADC разрешён  
 2 = ADC запрещён и не потребляет ток

**Регистр 9-2: ADCON1: 1-й регистр управления A/D**

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
-	ADCS2	ADCS1	ADCS0	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                      W=бит записывается                      U=не задействованный бит, читается как '0'  
 -n=Значение при POR                      '1'=бит установлен                      '0'=бит очищен                      X=бит не определён

- Бит 7** **Не задействован:** Читается как '0'
- Биты 6-4** **ADCS<2:0>:** Биты выбора тактирования A/D преобразователя  
 000 = Fosc/2  
 001 = Fosc/8  
 010 = Fosc/32  
 x11 = Frc (тактирование происходит от внутреннего генератора = 500 кГц)  
 100 = Fosc/4  
 101 = Fosc/16  
 110 = Fosc/64
- Биты 3-0** **Не задействован:** Читается как '0'

**Регистр 9-3: ADRESH: Старший регистр результата ADC (ADRESH) ADFM = 0**

| R/W-x  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| ADRES9 | ADRES8 | ADRES7 | ADRES6 | ADRES5 | ADRES4 | ADRES3 | ADRES2 |
| Бит 7  | Бит 6  | Бит 5  | Бит 4  | Бит 3  | Бит 2  | Бит 1  | Бит 0  |

**Описание:**

R=бит читается                      W=бит записывается                      U=не задействованный бит, читается как '0'  
 -n=Значение при POR                      '1'=бит установлен                      '0'=бит очищен                      X=бит не определён

- Биты 7-0** **ADRES<9:2>:** Биты результата ADC  
 Верхние 8 бит 10-ти битного результата преобразования

**Регистр 9-4: ADRESL: Младший регистр результата ADC (ADRESL) ADFM = 0**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES1	ADRES0	-	-	-	-	-	-
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                      W=бит записывается                      U=не задействованный бит, читается как '0'  
 -n=Значение при POR                      '1'=бит установлен                      '0'=бит очищен                      X=бит не определён

- Биты 7-6** **ADRES<1:0>:** Биты результата ADC  
 Нижние 2 бита 10-ти битного результата преобразования
- Биты 5-0** **Зарезервировано:** Не используются

**Регистр 9-5: ADRESH: Старший регистр результата ADC (ADRESH) ADFM = 1**

R/W-x	R/W-x						
-	-	-	-	-	-	ADRES9	ADRES8
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                      W=бит записывается                      U=не задействованный бит,  
читается как '0'  
-n=Значение при POR                      '1'=бит установлен                      '0'=бит очищен                      X=бит не определён

**Биты 7-2**      **Зарезервированы:** Не используются.

**Биты 1-0**      **ADRES<9:0>:** Биты результата ADC  
Верхние 2 бита 10-ти битного результата преобразования.

**Регистр 9-6: ADRESL: Младший регистр результата ADC (ADRESL) ADFM = 1**

R/W-x							
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается                      W=бит записывается                      U=не задействованный бит,  
читается как '0'  
-n=Значение при POR                      '1'=бит установлен                      '0'=бит очищен                      X=бит не определён

**Биты 7-0**      **ADRES<7:0>:** Биты результата ADC

Нижние 8 бит 10-ти битного результата преобразования

**9.3 Требования выборки A/D**

Для получения требуемой точности преобразования необходимо обеспечить чтобы конденсатор хранения (CHOLD) полностью зарядился до уровня напряжения на входе канала. Модель аналогового ввода показана на рис. 9-4. Внутреннее сопротивление источника (RS) и внутреннее сопротивление ключа выборки (RSS) непосредственно влияют на время требуемое для зарядки конденсатора CHOLD. Зависимость сопротивления ключа выборки (RSS), в зависимости от напряжения питания (VDD), можно увидеть на рис. 9-4. Максимальное рекомендуемое сопротивление для аналогового источника 10 кΩ. Уменьшая внутреннее сопротивление источника, можно уменьшить необходимое время выборки. После того как аналоговый входной канал выбран (или изменён), выборка A/D должна быть сделана перед тем как стартует преобразование. Для вычисления минимального времени выборки можно использовать уравнения 9-1. Эти уравнения допускают ошибку 1/2 LSB (1024 шагов ADC). Ошибка 1/2 LSB является максимальной ошибкой допустимой для определённой разрешающей способности ADC.

**9-1: Пример вычисления времени выборки**

*Предположения: Температура = 50°C и внешнее сопротивление 10ком Vdd=5В*

$T_{ASQ} = \text{Время установки усилителя} + \text{Время зарядки конденсатора хранения} + \text{Температурный коэффициент}$   
 $= T_{amp} + T_c + T_{coff} = 5\text{мкс} + T_c + [(Температура - 25°C)(0.05\text{мкс}/°C)]$

*Значение Tc может быть приближённо вычислено при помощи следующих уравнений:*

$$V_{APPLIED} \left( 1 - \frac{1}{2047} \right) = V_{CHOLD} \quad ;[1] V_{CHOLD} \text{ заряжено с точностью } 0.5\text{-LSB}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_c}{RC}} \right) = V_{CHOLD} \quad ;[2] V_{CHOLD} \text{ заряд в зависимости от } V_{APPLIED}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_c}{RC}} \right) = V_{APPLIED} \left( 1 - \frac{1}{2047} \right) \quad ;\text{комбинация [1] и [2]}$$

*Решение для Tc:*

$$T_c = -C_{HOLD}(R_{IC} + R_{SS} + R_s) \ln(1/2047) = -10\text{пф}(1\text{ком} + 7\text{ком} + 10\text{ком}) \ln(0.0004885) = 1.37 \text{ мкс}$$

*Следовательно:*

$$T_{ACQ} = 5\text{мкс} + 1.37\text{мкс} + [(50°C - 25°C)(0.05\text{мкс}/°C)] = 7.67 \text{ мкс}$$

**Примечание** 1: Опорное напряжение не влияет на уравнение.

2: Заряженный конденсатор хранения (CHOLD) не разряжается после каждого преобразования.

3: Рекомендуется чтобы внутреннее сопротивление источника аналогового сигнала не превышало 10ком.

Это требуется выполнять из-за наличия утечки ножки.

Рисунок 9-4: Модель аналогового входа

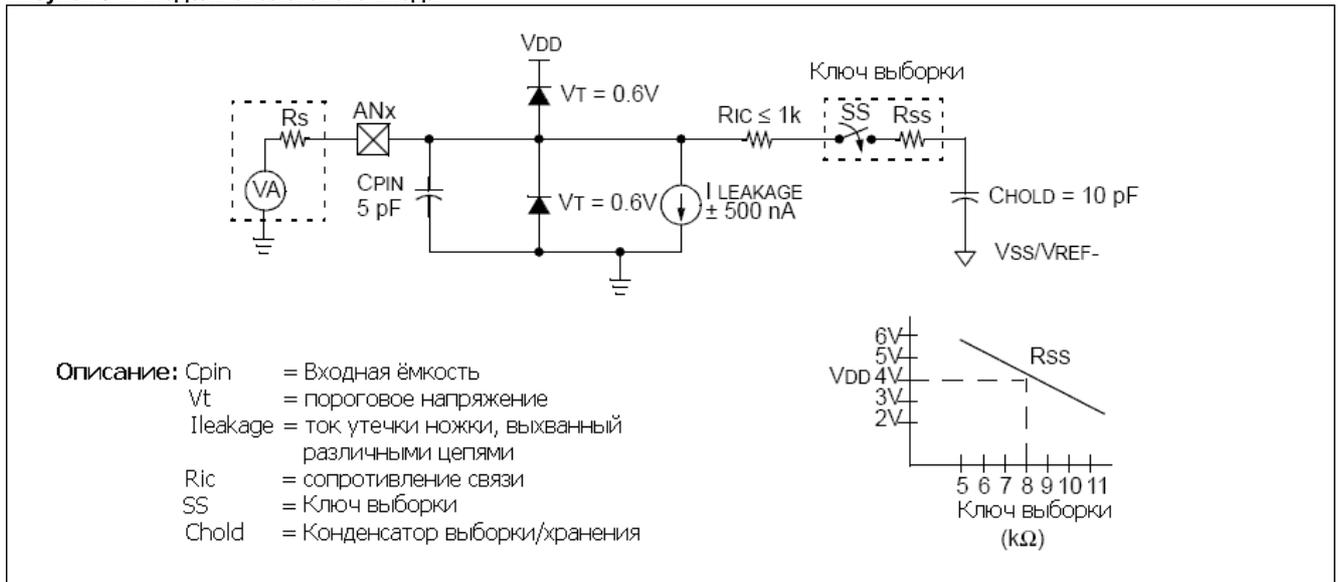
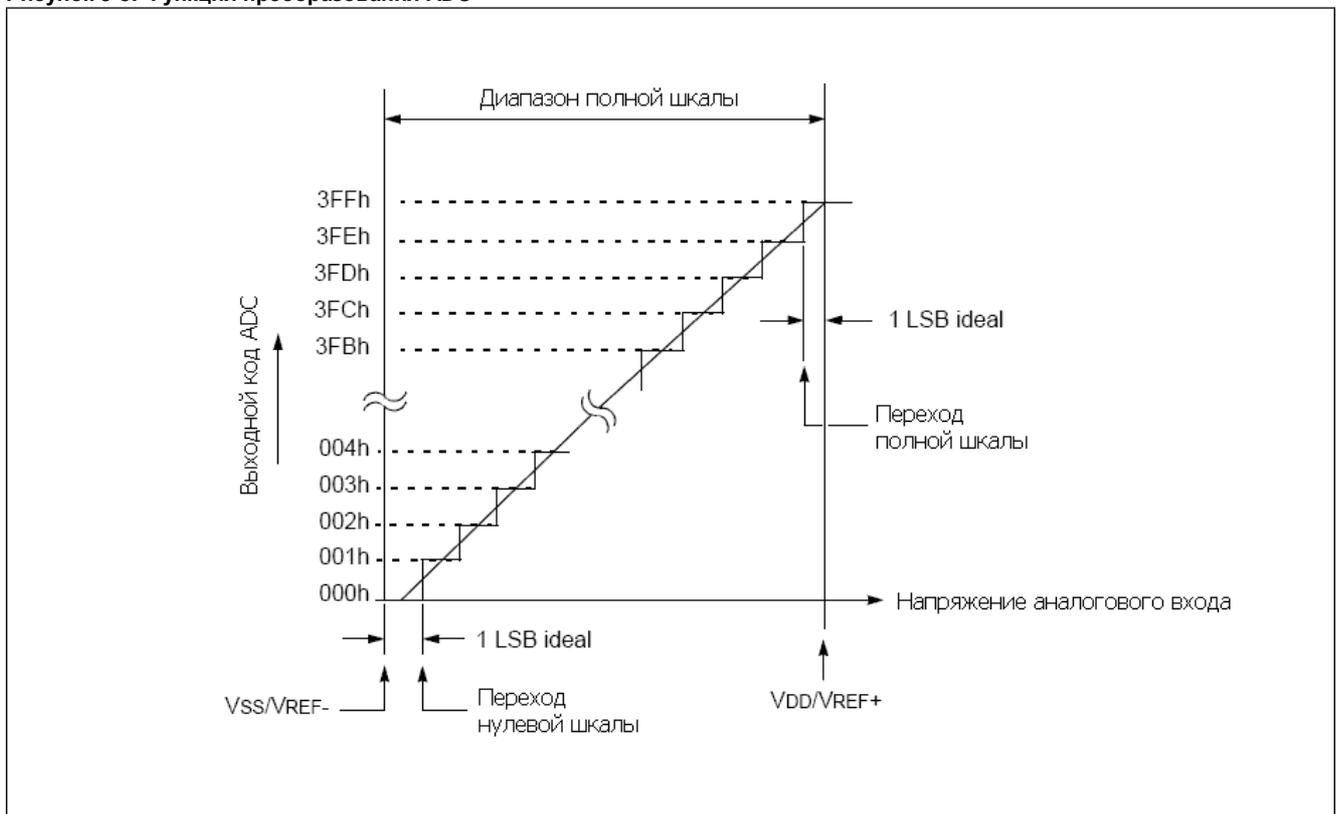


Рисунок 9-5: Функция преобразования ADC



**Таблица 9-2: Регистры связанные с ADC**

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Значение после POR,BOR	Значение после остальных сбросов
ADCON0	ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/-DONE	ADON	0000 0000	0000 0000
ADCON1	-	ADCS2	ADCS1	ADCS0	-	-	-	-	-000 ----	-000 ----
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
ANSELH	-	-	-	-	ANS11	ANS10	ANS9	ANS8	---- 1111	---- 1111
ADRESH	Регистр старшего байта результата преобразования A/D								xxxx xxxx	uuuu uuuu
ADRESL	Регистр младшего байта результата преобразования A/D								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	TOIE	INTE	RABIE	TOIF	INTF	RABIF	0000 000x	0000 000x
PIE1	-	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
PIR1	-	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
PORTA	-	-	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
PORTB	RB7	RB6	RB5	RB4	-	-	-	-	xxxx ----	uuuu ----
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	-	-	-	-	1111 ----	1111 ----
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISCO	1111 1111	1111 1111

**Описание:** x = неизвестный, u = неизменный, — = не задействован, читается как '0'. Серые клетки не используются для модуля ADC.

## 11.0 РАСШИРЕННЫЙ МОДУЛЬ ЗАХВАТА/СРАВНЕНИЯ/ШИМ

Расширенный модуль захвата/сравнения/ШИМ является периферийным устройством, которое можно использовать для подсчёта и контроля различных событий. В режиме захвата модуль позволяет подсчитывать длительность события. Режим сравнения позволяет использовать переключение внешний случай, когда predetermined сумма времени истекает. ШИМ режим позволяет генерировать ШИМ сигнал с переменной частотой и заполнением периода.

В таблице 11-1 показаны ресурсы таймера необходимые для ECSP модуля.

**Таблица 11-1: Режим ECSP - Используемые ресурсы таймера**

ECSP режим	Ресурсы таймера
Захвата	Таймер 1
Сравнения	Таймер 1
ШИМ	Таймер 2

**Регистр 11-1: CCP1CON: Регистр расширенного управления CCP1**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

**Описание:**

R=бит читается

W=бит записывается

U=не задействованный бит,  
читается как '0'

-n=Значение при POR

'1'=бит установлен

'0'=бит очищен

X=бит не определён

**Биты 7-6**

**P1M<1:0>**: Бит конфигурации выхода ШИМ

если CCP1M<3:2> = 00, 01, 10:

xx = P1A назначен как вход захвата/сравнения;

P1B, P1C, P1D назначены как выводы порта

если CCP1M<3:2> = 11:

00 = единственный выход; P1A модулирован; P1B, P1C, P1D назначены как выводы порта

01 = Мостовой выход вперёд; P1D модулирован; P1A активный; P1B, P1C не активные.

10 = Полу-мостовой выход; P1A, P1B модулированы с управлением мёртвым временем; P1C, P1D определены как выводы порта

11 = Мостовой выход назад; P1B модулирован; P1C активный; P1A, P1D не активные.

**Биты 5-4**

**DC1B<1:0>**: Младшие биты заполнения периода ШИМ

Режим захвата: Не используется.

Режим сравнения: Не используется

Режим ШИМ: Эти биты есть два LSbs заполнения ШИМ. Восемь MSbs находятся в CCP1L.

**Биты 3-0**

**CCP1M<3:0>**: Биты выбора режима ECSP

0000 = Захват/Сравнение/ШИМ выключены (сброшен ECSP модуль)

0001 = Не используется (зарезервировано)

0010 = Режим сравнения, переключение выхода на соответствие (CCP1IF бит устанавливается)

0011 = Не используется (зарезервировано)

0100 = Режим захвата, спад каждого импульса

0101 = Режим захвата, нарастание каждого импульса

0110 = Режим захвата, нарастание каждого 4-го импульса

0111 = Режим захвата, нарастание каждого 16-го импульса

1000 = Режим сравнения, устанавливается выход на соответствие (CCP1IF бит устанавливается)

1001 = Режим сравнения, очищается выход на соответствие (CCP1IF бит устанавливается)

1010 = Режим сравнения, генерирует программное прерывание на соответствие (CCP1IF бит устанавливается, CCP1 вывод не затрагивается)

1011 = Режим сравнения, запускает специальное событие (CCP1IF бит устанавливается; CCP1 сбрасываются TMR1 или TMR2, и стартует A/D преобразование, если ADC модуль разрешён)

1100 = Режим ШИМ; P1A, P1C активный высокий уровень; P1B, P1D активный высокий уровень

1101 = Режим ШИМ; P1A, P1C активный высокий уровень; P1B, P1D активный низкий уровень

1110 = Режим ШИМ; P1A, P1C активный низкий уровень; P1B, P1D активный высокий уровень

1111 = Режим ШИМ; P1A, P1C активный низкий уровень; P1B, P1D активный низкий уровень

### 11.3 Режим ШИМ

Режим ШИМ позволяет генерировать широтно-импульсный модулированный сигнал на выводе CCP1. Заполнение, период и разрешение определяются следующими регистрами:

- PR2
- T2CON

- CCP1L
- CCP1CON

В режиме Широтно-Импульсной Модуляции (ШИМ), CCP модуль производит 10-ти разрядный ШИМ сигнал, который поступает на вывод CCP1. Поскольку ножка CCP1 мультиплексирована с защёлкой данных PORT, TRIS для этого ножки должен быть очищен для разрешения работы в режиме вывода.

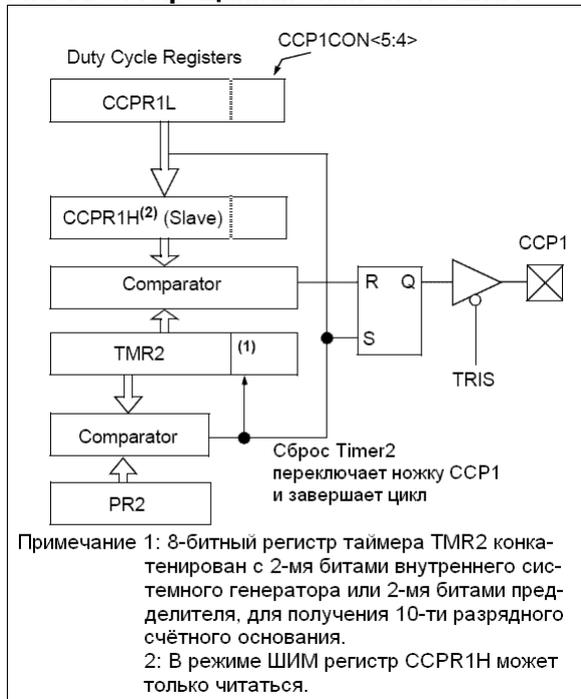
**Примечание:** Очистка регистра CCP1CON освобождает контроль CCP1 ножкой CCP1.

На рис. 11-3 показана упрощённая блок-схема формирования ШИМ.

На рис. 11-4 показана типовая форма ШИМ сигнала.

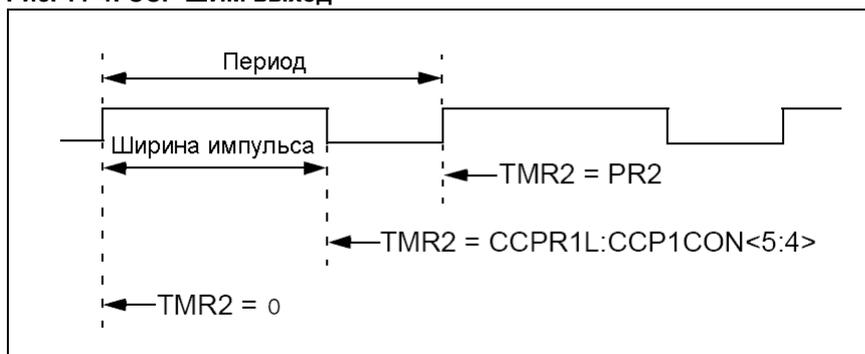
Для пошаговой инструкции настройки модуля CCP для формирования ШИМ, смотреть секцию 11.3.7 "Настройка для формирования ШИМ".

**Рис. 11-3: Упрощённая блок-схема ШИМ**



ШИМ выход (рис. 11-4) имеет базовое время (период) и время, когда на выходе высокий уровень (заполнение).

**Рис. 11-4: CCP ШИМ выход**



### 11.3.1 ПЕРИОД ШИМ

Период ШИМ определяется через регистр PR2 таймера Timer2. Период ШИМ может быть посчитан при помощи формулы 11-1.

#### Формула 11-1: Период ШИМ

$$\text{ШИМ период} = [(PR2) + 1] * 4 * T_{osc} * (\text{TMR2 Значение предделителя})$$

Когда TMR2 равен PR2, в следующем цикле приращения происходят три события:

- TMR2 очищается
- Ножка CCP1 устанавливается в единицу. (Исключение: Если заполнение ШИМ = 0%, то ножка не устанавливается в единицу.)
- Заполнение ШИМ защёлкивается из CCP1L в CCPR1H.

**Примечание:** Постделитель Timer2 (смотреть **Раздел 7.1 "Работа Timer2"**) не используется для определения частоты ШИМ.

### 11.3.2 ЗАПОЛНЕНИЕ ШИМ

Заполнение ШИМ определяется содержимым 10-битного значения составного регистра: CCP1L регистр и DC1B<1:0> биты CCP1CON регистра. CCP1L содержит восемь MSBs и DC1B<1:0> биты регистра CCP1CON содержат два LSBs. CCP1L и DC1B<1:0> биты регистра CCP1CON могут быть записаны в любое время. Значение заполнения не защёлкивается в CCP1H, пока период не завершён (т.е., пока не произойдёт соответствие между регистрами PR2 и TMR2). Пока используется ШИМ, регистр CCP1H можно только читать.  
Формула 11-2 используется для подсчёта ширины импульса.  
Формула 11-3 используется для подсчёта коэффициента заполнения ШИМ.

#### Формула 11-2: Ширина импульса

Ширина импульса =  $(CCP1L:CCP1CON<5:4>) * T_{osc} * (\text{Значение делителя TMR2})$

#### Формула 11-3: Коэффициент заполнения

Коэффициент заполнения =  $(CCP1L:CCP1CON<5:4>)/(4*(PR2 + 1))$

Регистр CCP1H и 2-битная внутренняя защёлка используются для двойного буфера заполнения ШИМ. Это двойное буферирование необходимо для уменьшения ошибки формирования ШИМ.  
8-битный регистр таймера TMR2 конкатенирован с 2-мя битами внутреннего системного генератора (FOSC), или 2-мя битами делителя, для создания 10-ти разрядного счётного основания. Системный генератор используется если делитель Timer2 установлен в 1:1.  
Когда 10-ти битное счётное основание соответствует CCP1H и 2-х битному защёлка, затем CCP1 ножка будет очищена (смотреть рис. 11-3).

### 11.3.3 Разрешение ШИМ

Разрешение определяет число доступных рабочих циклов для данного периода. Для примера 10-ти битное разрешение позволяет получить 1024 дискретных рабочих циклов, поскольку 8-ми битное разрешение позволяет получить 256 дискретных рабочих циклов.

Максимальное разрешение ШИМ достигает 10 бит, когда PR2 равен 255. Разрешение является функцией значения регистра PR2, согласно формулы 11-4.

#### Формула 11-4: Разрешение ШИМ

Разрешение =  $\log[4(PR2 + 1)]/\log(2)$  бит

**Примечание:** Если значение ширины импульса будет больше чем период, назначенная ножка(и) ШИМ будут находится в неизменном состоянии.

TABLE 11-2: Пример частоты и разрешения ШИМ для FOSC = 20 MHz

Частота ШИМ	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Делитель таймера (1,4,16)	16	4	1	1	1	1
Значение PR2	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Максимальное разрешение (бит)	10	10	10	8	7	6.6

TABLE 11-3: Пример частоты и разрешения ШИМ для FOSC = 8 MHz

Частота ШИМ	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Делитель таймера (1,4,16)	16	4	1	1	1	1
Значение PR2	0x65	0x65	0x65	0x19	0x0C	0x09
Максимальное разрешение (бит)	8	8	8	6	5	5

### 11.3.4 Работа в спящем режиме SLEEP

В спящем режиме регистр TMR2 не инкрементируется и состояние модуля не меняется. Если на ножке CCP1 есть значение, то это значение продолжит удерживаться. Когда устройство проснётся, TMR2 продолжит работу с предыдущего состояния.

### 11.3.5 Изменение частоты системного генератора

Частота ШИМ зависит от частоты системного генератора. Любые изменения частоты системного генератора повлекут за собой изменение частоты ШИМ. Смотрите **Часть 3.0 "Модуль Генератора (With Fail-Safe Clock Monitor)"** для дополнительных деталей.

### 11.3.6 Эффекты сброса

Любой сброс переводит все порты в режим ввода и регистры CCP переводит в их состояние после сброса.

### 11.3.7 Установка для режима ШИМ

Следующие шаги могут потребоваться, когда CCP модуль конфигурируется для ШИМ:

1. Блокировать вывод через ШИМ ножку (CCP1), установив соответствующий бит TRIS.
2. Установить период ШИМ, загрузив регистр PR2.
3. Конфигурировать модуль CCP для режима ШИМ, загрузив в регистр CCP1CON соответствующее значение.
4. Установить рабочий цикл ШИМ, загрузив регистр CCP1L и DC1B<1:0> биты регистра CCP1CON.
5. Сконфигурировать и запустить Timer2:
  - Очистить бит флага прерывания TMR2IF регистра PIR1.
  - Установить значение делителя Timer2, загрузив биты T2CKPS регистра T2CON.
  - Разрешить Timer2, установив бит TMR2ON регистра T2CON.
6. Разрешить выход ШИМ после старта нового цикла ШИМ:
  - Ждать пока переполнится Timer2 (будет установлен бит TMR2IF регистра PIR1).
  - Разрешить вывод на ножку CCP1, очистив соответствующий бит TRIS.

### 11.4 PWM (Enhanced Mode)

Расширенный режим ШИМ может генерировать ШИМ сигнал на до четырёх различных выходных ножки с разрешением 10 бит. Это может делать через четыре различных режима вывода:

- Одиночный ШИМ
- Полумостовой ШИМ
- Мостовой ШИМ, прямой режим
- Мостовой ШИМ, обратный режим

Для выбора расширенного режима ШИМ нужно соответственно установить биты P1M регистра CCP1CON.

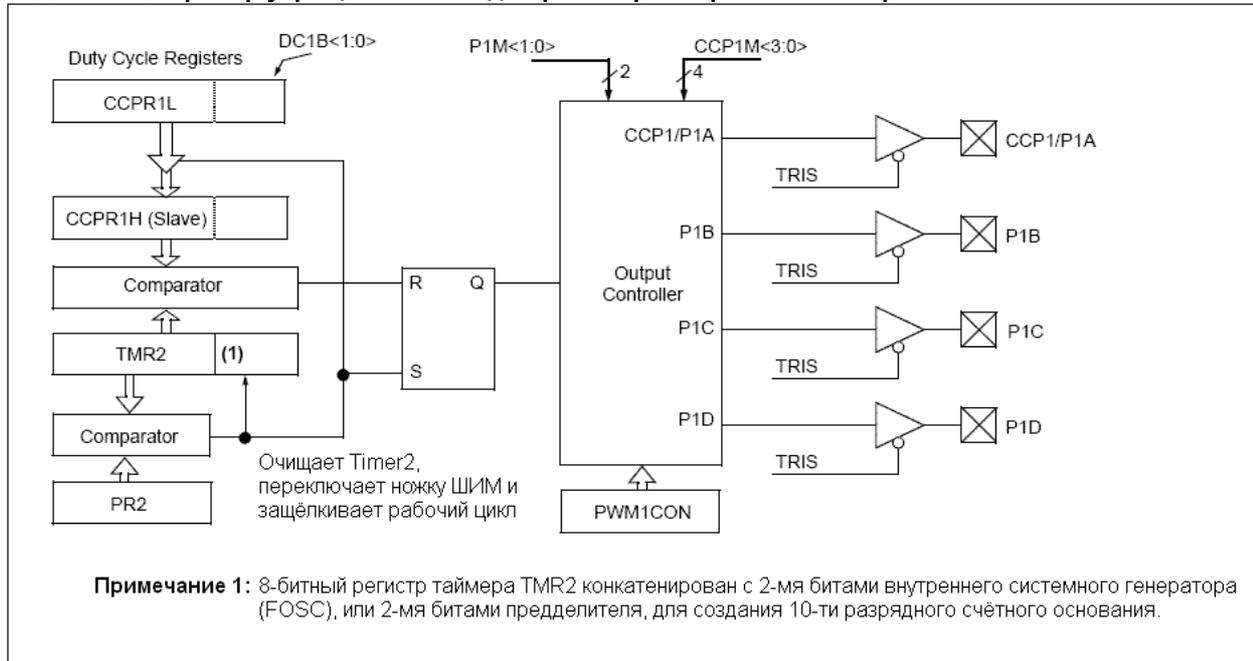
Выходы ШИМ мультиплексированы с ножками Ввода/вывода и определены P1A, P1B, P1C и P1D. Полярность ножек ШИМ конфигурируется и выбирается через установку соответствующих бит CCP1M в регистре CCP1CON.

В таблице 11-4 показано назначение ножек для каждого режима расширенного режима ШИМ.

На рис. 11-5 показана упрощённая блок-схема модуля расширенного ШИМ.

**Примечание:** Для предотвращения генерации неполного сигнала, когда ШИМ впервые включен, модуль ЕССР ждёт начала нового ШИМ периода прежде чем начать генерирование ШИМ сигнала.

**FIGURE 11-5: Пример упрощённой блок-диаграммы расширенного ШИМ режима**



**Примечание 1:** Значение регистра TRIS для каждого выхода ШИМ соответственно конфигурируется.

**2:** Очистка регистра CCP1CON может освободить контроль ЕССР над всеми выходными ножками ШИМ.

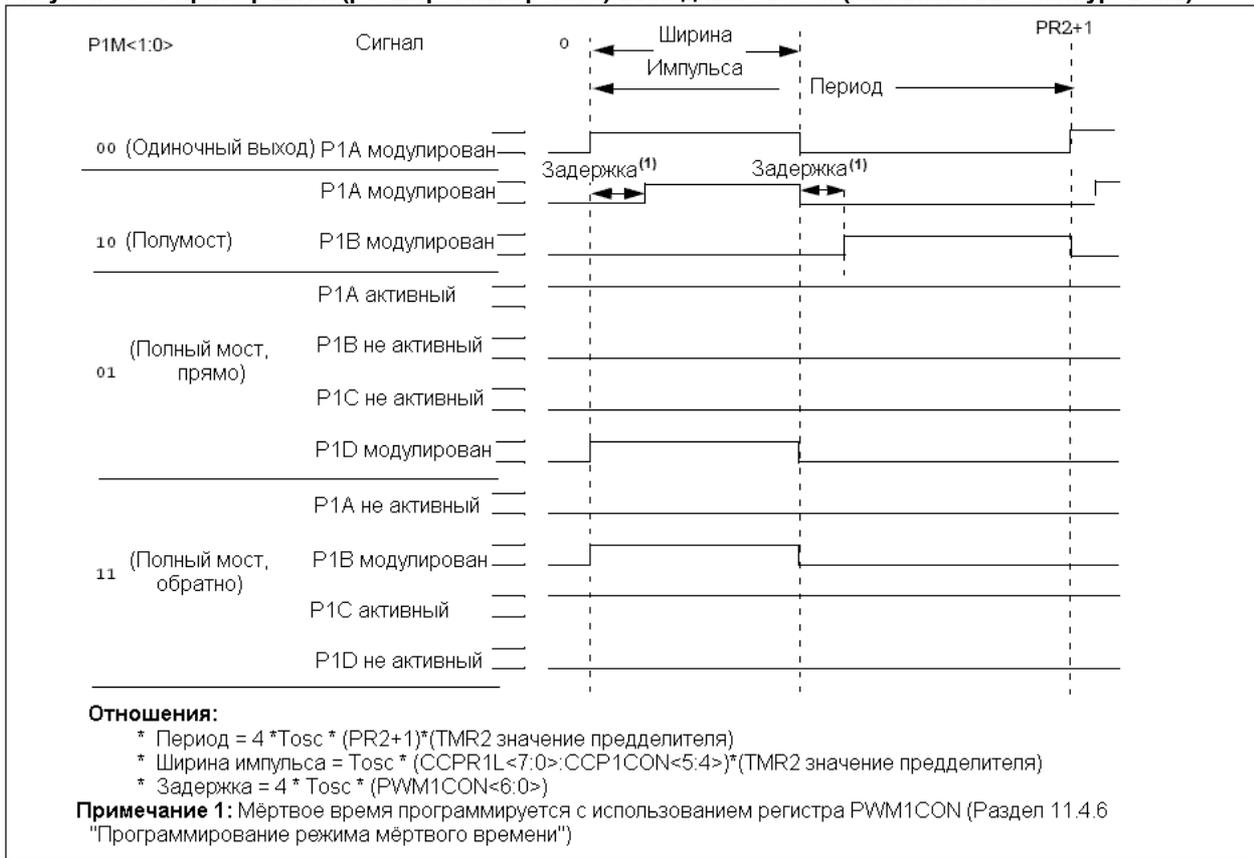
**3:** Любая ножка не используемая в расширенном режиме ШИМ доступна для альтернативных функций ножек

**TABLE 11-4: Пример назначения ножек для различных расширенных режимов ШИМ**

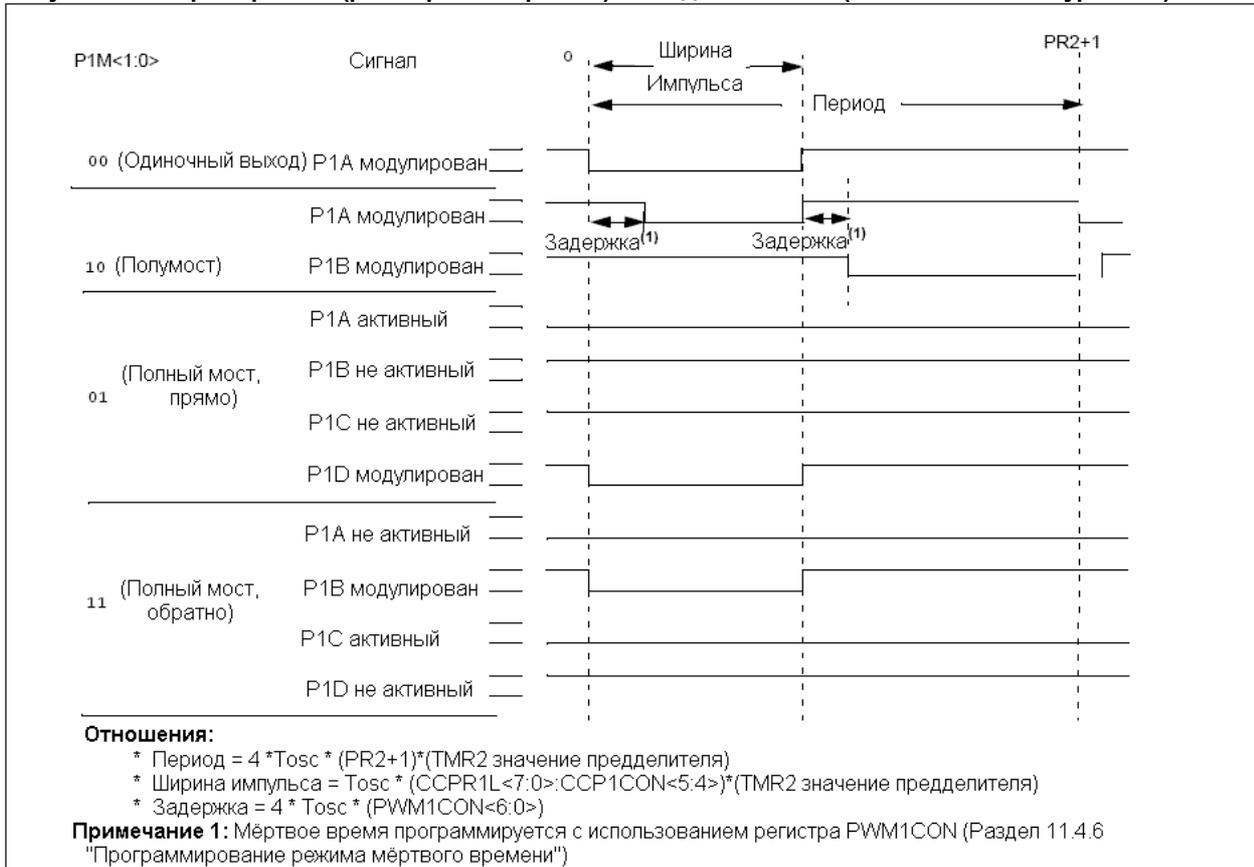
Режим ЕССР	P1M<1:0>	CCP1/P1A	P1B	P1C	P1D
Одиночный	00	Да <sup>(1)</sup>	Да <sup>(1)</sup>	Да <sup>(1)</sup>	Да <sup>(1)</sup>
Полумостовой	10	Да	Да	Нет	Нет
Мостовой, прямой	01	Да	Да	Да	Да
Мостовой, обратный	11	Да	Да	Да	Да

**Примечание 1:** Импульсное управление разрешёнными выходами в одиночном (Single) режиме.

**Рисунок 11-6: Пример ШИМ (расширенный режим) выходных связей (активный высокий уровень)**



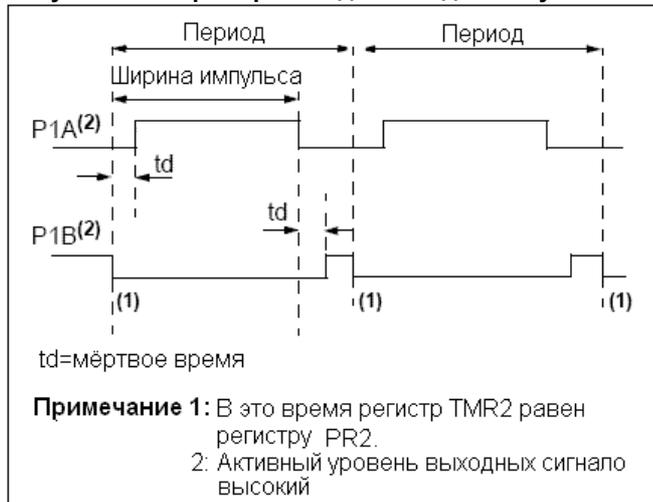
**Рисунок 11-7: Пример ШИМ (расширенный режим) выходных связей (активный низкий уровень)**



### 11.4.1 HALF-BRIDGE MODE

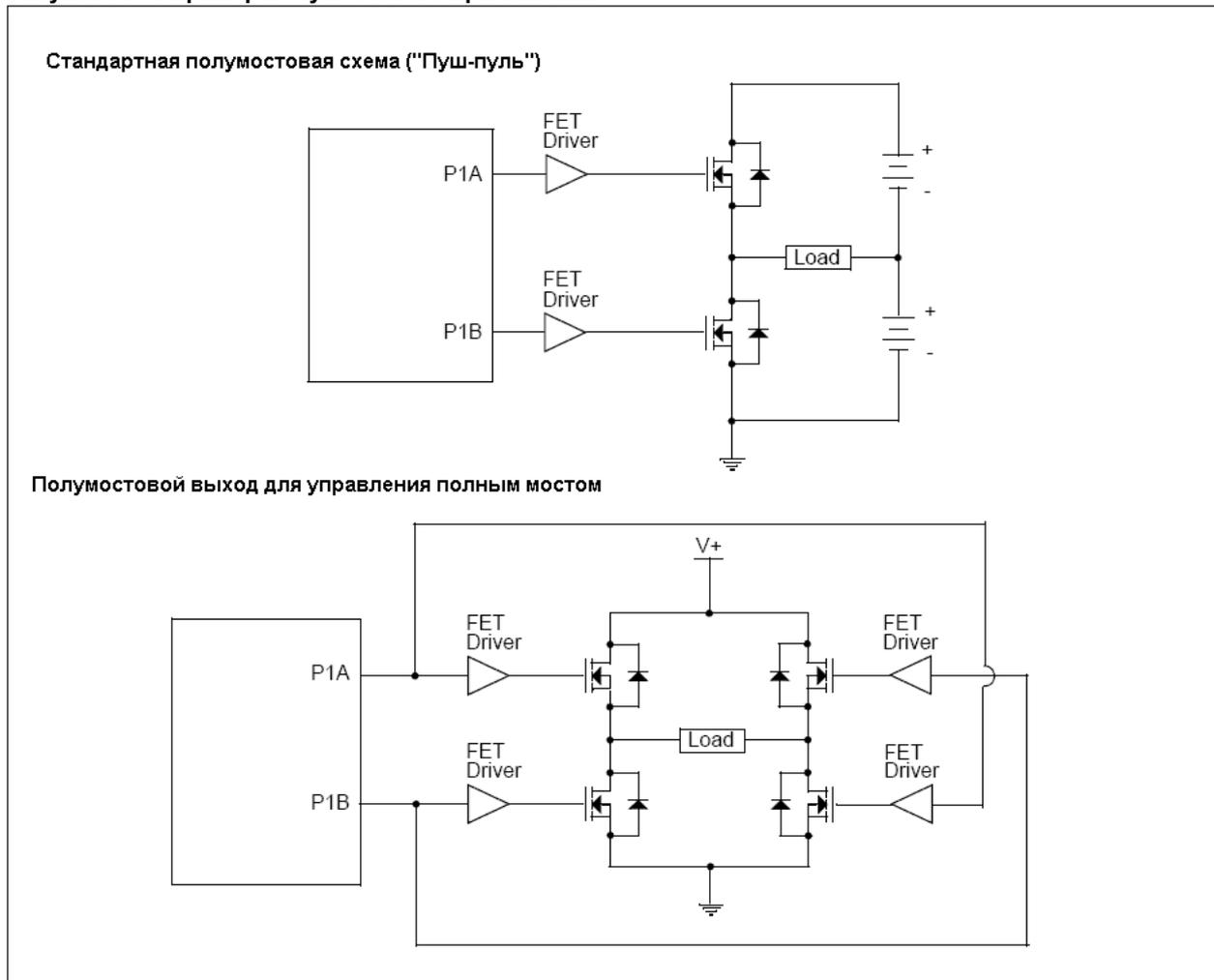
В полумостовом режиме две ножки используются как выходы драйвера двухтактной нагрузки. Выходной сигнал ШИМ выводится на ССР1/P1A ножку, пока комплиментарный ШИМ выходной сигнал выводится на P1B ножку (смотреть рис. 11-6). Этот режим может быть использован для полумостовых приложений, как показано на рис.11-9, или для мостовых приложений, где четыре силовых ключа модулируются двумя ШИМ сигналами.

**Рисунок 11-8: Пример выхода ШИМ для полумоста**



В полумостовом режиме программируется мёртвое время, которое может использоваться для предотвращения сквозного тока в полумостовых мощных устройствах. Значение битов PDC<6:0> регистра PWM1CON устанавливает число циклов инструкции перед тем, как выход драйвера будет активным. Если значение больше рабочего цикла, соответствующий выход остаётся не активным в течении целого цикла. Смотрите **Часть 11.4.6 “Программирование режима мёртвого времени”** для больших подробностей функционирования задержки мёртвого времени.

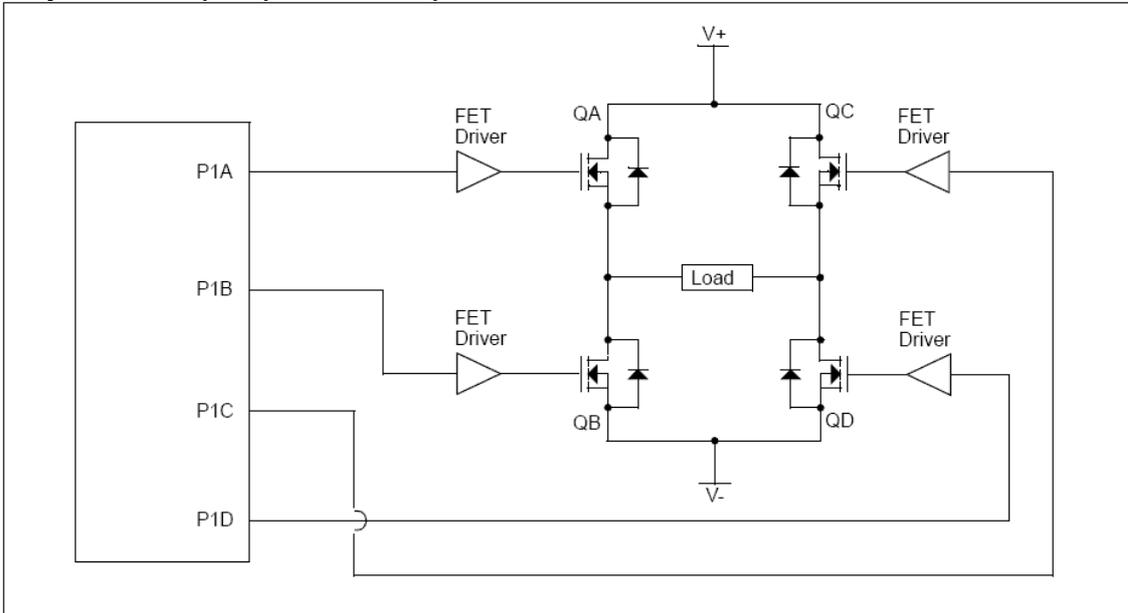
**Рисунок 11-9: Пример полумостового приложения**



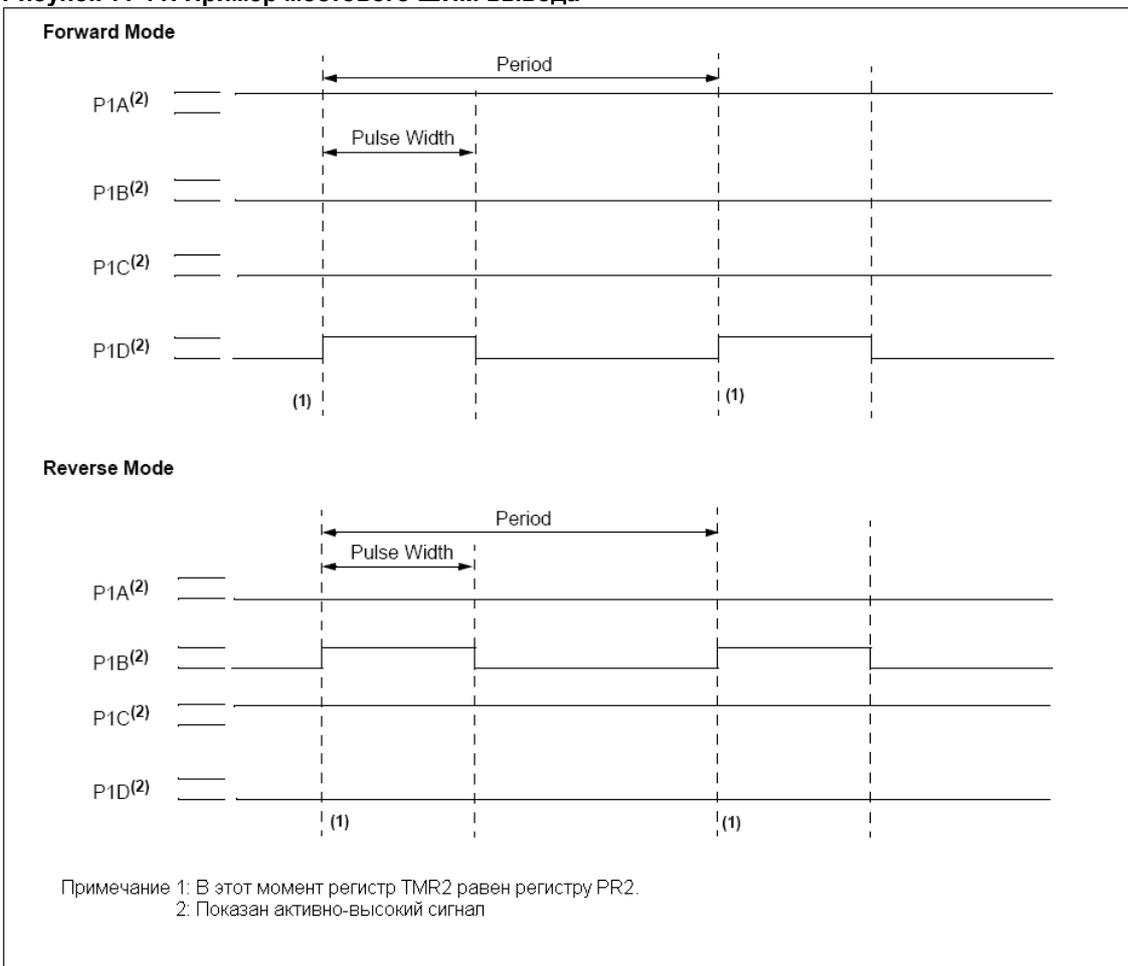
### 11.4.2 Мостовой режим

В мостовом режиме все четыре ножки используются как выходы. Например, мостовое приложение, показанное на рис. 11-10. В прямом режиме ножка CCP1/P1A находится в активном состоянии, ножка P1D модулирована, пока P1B и P1C находятся в не активном состоянии, как показано на рис. 11-11. В обратном режиме, P1C находится в активном состоянии, ножка P1B модулирована, пока P1A и P1D находятся в не активном состоянии, как показано на рис. 11-11. Выходы P1A, P1B, P1C и P1D мультиплексируются с защёлками данных порта. Связанные биты TRIS могут быть очищены для конфигурирования ножек P1A, P1B, P1C и P1D как выходов.

**Рисунок 11-10: Пример мостового приложения**



**Рисунок 11-11: Пример мостового ШИМ вывода**



### 11.4.2.1 Изменение направления в мостовом режиме

В мостовом режиме бит P1M1 в регистре CCP1CON можно использовать для управления прямым/обратным направлением. Когда программа приложения меняет этот бит направления, модуль меняет новое направление в следующем ШИМ цикле.

Изменение направления инициируется в программе изменением бита P1M1 регистра CCP1CON. Следующая последовательность происходит четыре Timer2 цикла, предшествуя концу текущего периода ШИМ:

- Модулированные выходы (P1B и P1D) устанавливаются в их неактивное состояние.
- Связанные немодулированные выходы (P1A и P1C) переключают привод в противоположное направление.
- ШИМ модулятор начинает следующий период.

Смотреть рис. 11-12 для иллюстрации этой последовательности.

Мостовой режим не обеспечивает формирования мёртвого времени. Когда модулирован только один выход, мёртвое время обычно не требуется. Иногда возможна ситуация, когда требуется мёртвое время. Эта ситуация происходит, когда справедливы оба следующих условия:

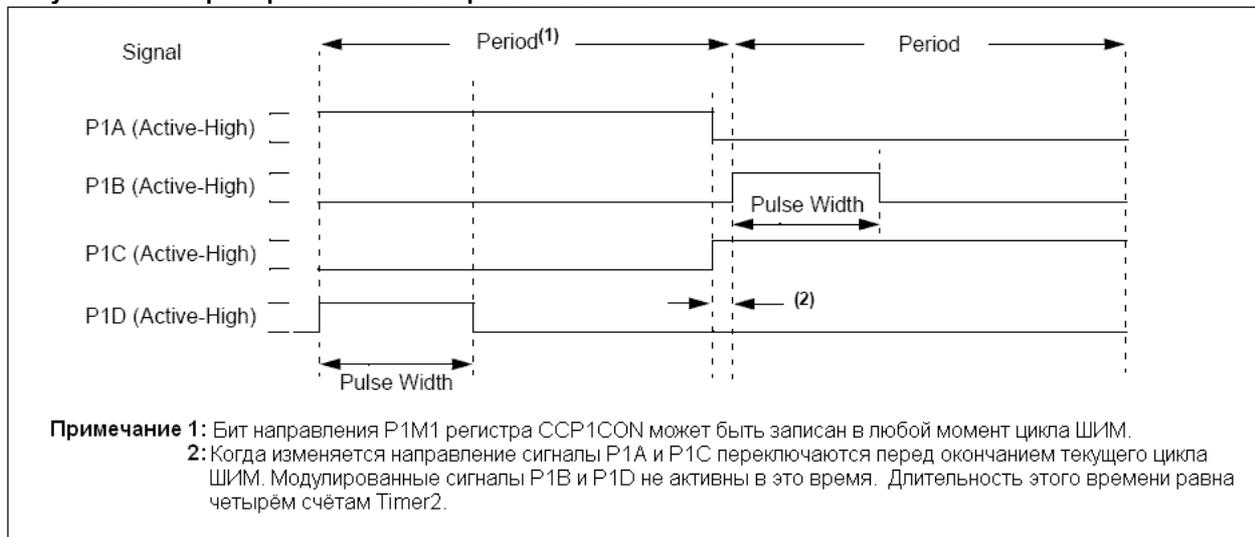
1. Направление изменяется, когда рабочий цикл выхода близок к 100%.
2. Время выключения мощного ключа, включая мощное устройство и схему драйвера, превышает время включения.

На рис. 11-13 показан пример изменения направления ШИМ с прямого на обратное при 100% рабочем цикле. В этом примере, в момент t1, выходы P1A и P1D становятся неактивными, пока выход P1C становится активным. Поскольку время выключения мощного устройства длиннее времени включения, через мощные устройства QC и QD будет протекать сквозной ток (смотреть рис. 11-10) в течении времени T. Этот феномен будет происходить с мощными устройствами QA и QB при изменении направления ШИМ с обратного на прямое.

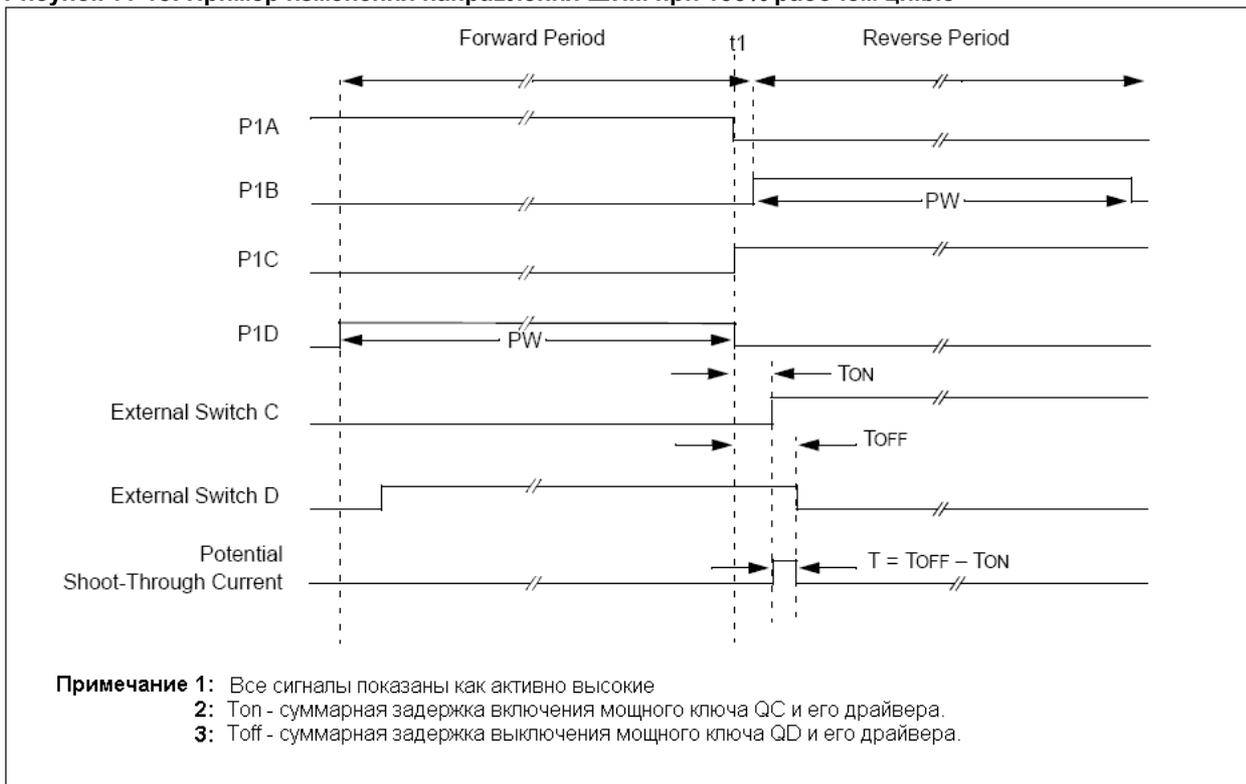
Если для приложения возможно изменение направления ШИМ при высоком рабочем цикле, то возможны два решения для устранения сквозного тока:

1. Уменьшить рабочий цикл ШИМ для одного периода ШИМ перед изменением направления.
  2. Использовать ключе, которые выключаются быстрее, чем включаются.
- Могут существовать другие варианты устранения сквозного тока.

**Рисунок 11-12: Пример изменения направления ШИМ**



**Рисунок 11-13: Пример изменения направления ШИМ при 100% рабочем цикле**



### 11.4.3 Стартовые соображения

Когда используется любой режим ШИМ, прикладные аппаратные средства должны использовать соответствующие внешние подтягивающие вверх и/или подтягивающие вниз резисторы на выходных ножках ШИМ.

**Примечание:** После сброса микроконтроллера все его ножки ввода/вывода находятся в высокоимпедансном состоянии. Внешние схемы должны удерживать мощные ключи в выключенном состоянии пока микроконтроллер не установит на ножках ввода/вывода соответствующие сигналы или активизируются ШИМ выходы.

Биты  $ССР1M<1:0>$  регистра  $ССР1CON$  позволяют пользователю выбирать независимо, какие выходные ШИМ сигналы будут иметь активно высокий или активно низкий уровень, для каждой пары выходных ножек ШИМ (P1A/P1C и P1B/P1D). Полярность ШИМ выхода должна быть выбрана перед разрешением драйверов выходных ножек ШИМ. Изменять полярность при разрешённых драйверах выходных ножек ШИМ не рекомендуется, т.к. это может привести к аварии прикладной схемы. P1A, P1B, P1C и P1D выходные защёлки могут не быть в соответствующих состояниях, пока ШИМ модуль инициализируется. Разрешение ножки ШИМ выходных драйверов в момент времени как Расширенный режим ШИМ может вызывать повреждение прикладной схемы. Режим выхода должен соответствовать разрешённому расширенному режиму и завершить полный ШИМ цикл перед разрешением ножки ШИМ выходных драйверов. Завершение полного цикла ШИМ индицируется через бит  $TMR2IF$  регистра  $PIR1$ , который устанавливается в начале второго периода ШИМ.

### 11.4.4 Режим авто-выключения расширенного ШИМ

Режим ШИМ поддерживает режим авто-выключения, который может выключить выходы ШИМ, когда происходит случай внешнего выключения. Режим авто-выключения устанавливает выходные ножки ШИМ в predetermined состояния. Это режим позволяет предохранять ШИМ от аварии приложения. Источник авто-выключения выбирается с помощью битов  $ECCPASx$  регистра  $ECCPAS$ . Случай выключения может генерироваться через:

- Логический '0' на ножке INT
- Компаратор C1
- Компаратор C2
- Программной установкой бита  $ECCPASE$

Состояние выключения индицируется через бит  $ECCPASE$  (Состояние авто-выключения) регистра  $ECCPAS$ . Если этот бит равен '0', то ШИМ ножки функционируют нормально. Если бит равен '1', выходы ШИМ находятся в состоянии выключения.

Когда происходит случай выключения, случаются две вещи:

Бит  $ECCPASE$  устанавливается в '1'.  $ECCPASE$  остаётся установленным, пока не будет очищен программно или не произойдёт авто-рестарт (смотреть **Раздел 11.4.5 "Режим авто-рестарта"**).

Разрешённые ножки ШИМ асинхронно устанавливаются в их выключенные состояния. Выходные ножки ШИМ сгруппированы парами [P1A/P1C] и [P1B/P1D]. Состояние каждой ножки пары определено битами  $PSSAC$  и  $PSSBD$  регистра  $ECCPAS$ . Каждая ножка пары может быть установлена в одно из трёх состояний:

- Логическая '1'
- Логический '0'
- Третье состояние (Высокоимпедансное)

### Регистр 11-2: ECCPAS: Регистр управления расширенным захват/сравнение/ШИМ авто-выключение

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

#### Легенда:

R = Читаемый бит  
 W = Записываемый бит  
 U = Недействительный бит, читается как '0'  
 -n = Значение при POR  
 '1' = Бит установлен  
 '0' = Бит сброшен  
 x = Бит не определён

- Бит 7 **ECCPASE:** ECCP Бит статуса события авто-выключения  
 1 = Произошло событие авто-выключения; ECCP выходы находятся в выключенном состоянии  
 0 = ECCP выходы нормально функционируют
- Биты 6-4 **ECCPAS<2:0>:** ECCP Биты выбора источника авто-выключения  
 000 = Авто-выключение отключено  
 001 = Изменение выхода компаратора C1  
 010 = Изменение(1) выхода компаратора C2  
 011 = Изменение на выходе любого C1 или C2 компараторов  
 100 = VIL на INT ножке  
 101 = VIL на INT ножке или изменение компаратора C1  
 110 = VIL на INT ножке или изменение компаратора C2  
 111 = VIL на INT ножке или изменение на выходе любого C1 или C2 компараторов
- Биты 3-2 **PSSACn:** Биты управления выключенным состоянием ножек P1A и P1C  
 00 = Перевести ножки P1A и P1C в '0'  
 01 = Перевести ножки P1A и P1C в '1'  
 1x = Ножки P1A и P1C в третье состояние
- Биты 1-0 **PSSBDn:** Биты управления выключенным состоянием ножек P1B и P1D  
 00 = Перевести ножки P1B и P1D в '0'  
 01 = Перевести ножки P1B и P1D в '1'  
 1x = Ножки P1B и P1D в третье состояние

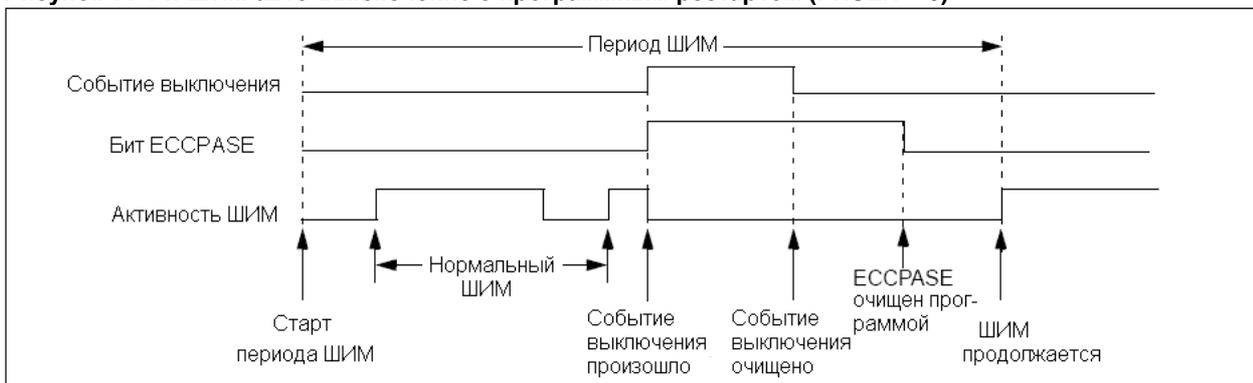
**Примечание 1:** Если C2SYNC разрешен, выключение может быть задержано к Timer1.

**Примечание 1:** Условием авто-выключения является сигнал базового уровня, не сигнал перепада. Как долго уровень присутствует, так долго присутствует авто-выключение.

**2:** Запись бита ECCPASE блокируется, пока условие авто-выключения присутствует.

**3:** Как только условие авто-выключения будет удалено и ШИМ заново стартует (так же через программу или авто-рестарт) ШИМ сигнал всегда повторно стартует в начале следующего периода ШИМ.

**Рисунок 11-14: ШИМ авто-выключение с программным рестартом (PRSEN = 0)**



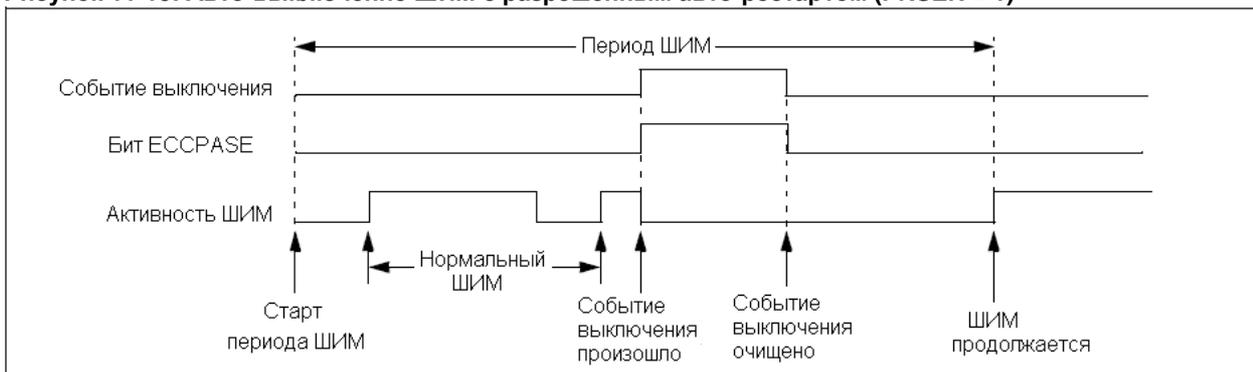
#### 11.4.5 Режим авто-рестарта

Расширенный ШИМ может быть конфигурирован для автоматического рестарта ШИМ сигнала как только условие авто-выключения будет устранено. Авто-рестарт разрешается установкой бита PRSEN в регистре PWM1CON.

Если авто-рестарт включен, как долго бит ECCPASE установлен, так долго активно состояние авто-выключения.

Когда условие авто-выключения удалено, бит ECCPASE должен быть очищен через аппаратное средство и нормальное функционирование будет продолжено.

**Рисунок 11-15: Авто-выключение ШИМ с разрешённым авто-рестартом (PRSEN = 1)**



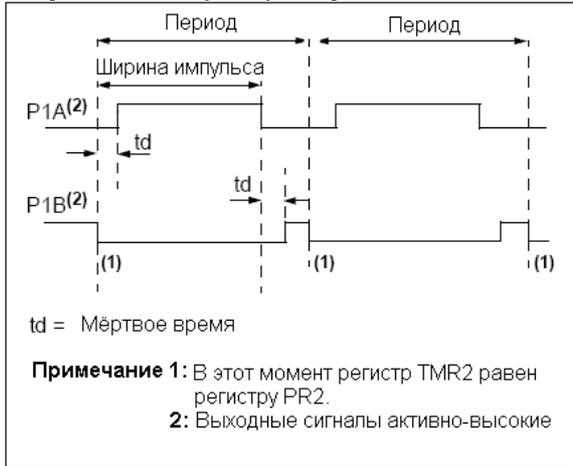
### 11.4.6 Программируемый режим мёртвого времени

В полумостовых приложениях, когда все мощные ключи модулируются частотой ШИМ, мощные ключи обычно требуют больше времени на выключение, чем на включение. Если оба, верхний и нижний, мощные ключи одновременно открыты (один открывается, а другой закрывается), оба ключа оказываются открытыми на короткий промежуток времени, пока один ключ полностью закроется. В течении этого короткого интервала очень большой ток (сквозной ток) протекает через оба мощных ключа, закорачивая питание моста. Для избегания этого разрушительного сквозного тока, включение мощных ключей обычно задерживается, позволяя другим ключам полностью закрыться.

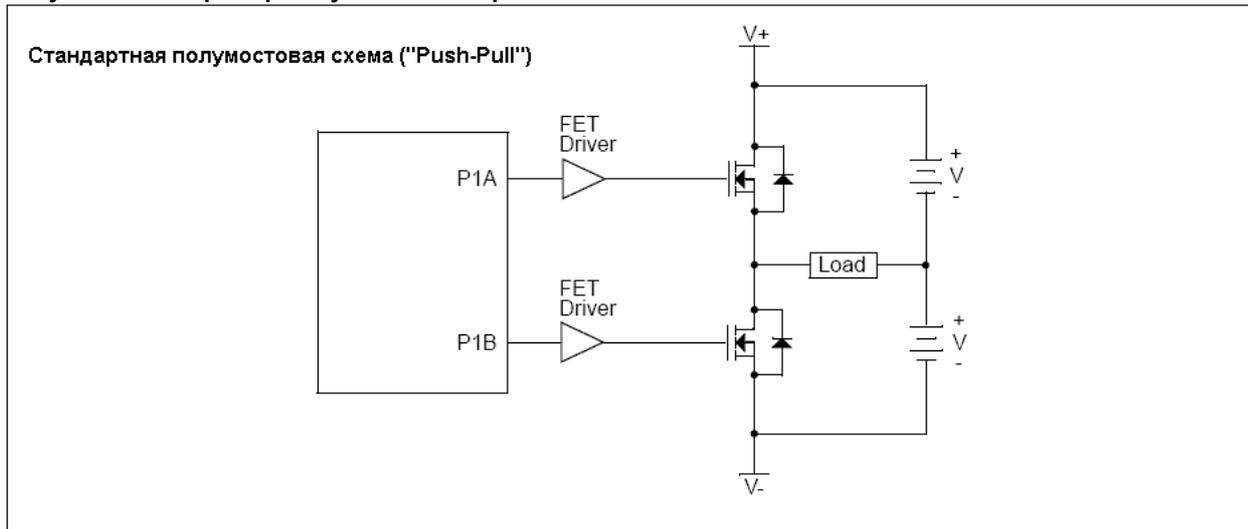
В полу-мостовом режиме, доступно цифровое программирование величины мёртвого времени, позволяющего избежать сквозного тока разрушающего мощные ключи. Задержка происходит при переходе сигнала из не активного состояния в активное состояние. Смотрите рисунок 11-8 для иллюстрации.

Младшие семь бит связанного регистра PWM1CON (регистр 11-3) устанавливают период задержки в циклах инструкций микроконтроллера (TCY или 4 TOSC).

**Рисунок 11-16: Пример полумостового выхода ШИМ**



**Рисунок 11-17: Пример полумостового приложения**



**Регистр 11-3: PWM1CON: Регистр управления расширенным ШИМ**

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PRSEN | PDC6  | PDC5  | PDC4  | PDC3  | PDC2  | PDC1  | PDC0  |
| Бит 7 | Бит 6 | Бит 5 | Бит 4 | Бит 3 | Бит 2 | Бит 1 | Бит 0 |

**Легенда:**

R = Читаемый бит                      W = Записываемый бит                      U = Неадействованный бит, читается как '0'  
 -n = Значение при POR                      '1' = Бит установлен                      '0' = Бит сброшен                      x = Бит не определён

Бит 7	<b>PRSEN:</b> Бит разрешения рестарта ШИМ 1 = С авто-выключением, бит ECCPASE очищается автоматически как только уходит прочь событие отключения; автоматический рестарт ШИМ 0 = С авто-выключением, ECCPASE необходимо очистить программно для рестарта ШИМ
Биты 6-0	<b>PDC&lt;6:0&gt;:</b> PWM Биты счётчика задержки PDCn = Число циклов FOSC/4 (4 * TOSC) между запланированным моментом когда сигнал ШИМ <b>должен</b> становится активным и <b>реальным</b> моментом когда это произойдёт

**Note 1:** Бит сбрасывается в '0' с двухскоростной стартуп и LP, XT или HS выбраны как режим генератора или разрешён режим защиты от ошибки.

### 11.4.7 PULSE STEERING MODE

В режиме одиночного выхода, на любую из ШИМ ножек может быть выведен модулированный сигнал. В добавок один и тот же ШИМ сигнал можно одновременно вывести на несколько ножек.

Как только выбран режим одиночного выхода ( $CCP1M<3:2> = 11$  и  $P1M<1:0> = 00$  в регистре CCP1CON), программа пользователя может переносить тот же ШИМ сигнал на одну, две, три или четыре выходных ножки, установив подходящие биты STR<D:A> регистра PSTRCON, как показано на рисунке 11-18.

**Note:** Соответствующие биты TRIS должны быть установлены на вывод ('0') разрешив для ножки драйвер выхода чтобы увидеть сигнал ШИМ на ножке.

Пока активен режим управления ШИМ, биты CCP1M<1:0> регистра CCP1CON выбирают выходную полярность ШИМ для ножек P1<D:A>.

Операция авто-выключения ШИМ так же относится к режиму управления ШИМ как описано в Части 11.4.4 "Режим Авто-выключения расширенного ШИМ". Событие авто-выключения может влиять только на ножки, которые сконфигурированы как выходы ШИМ.

**Регистр 11-4: PSTRCON: PULSE STEERING CONTROL REGISTER<sup>(1)</sup>**

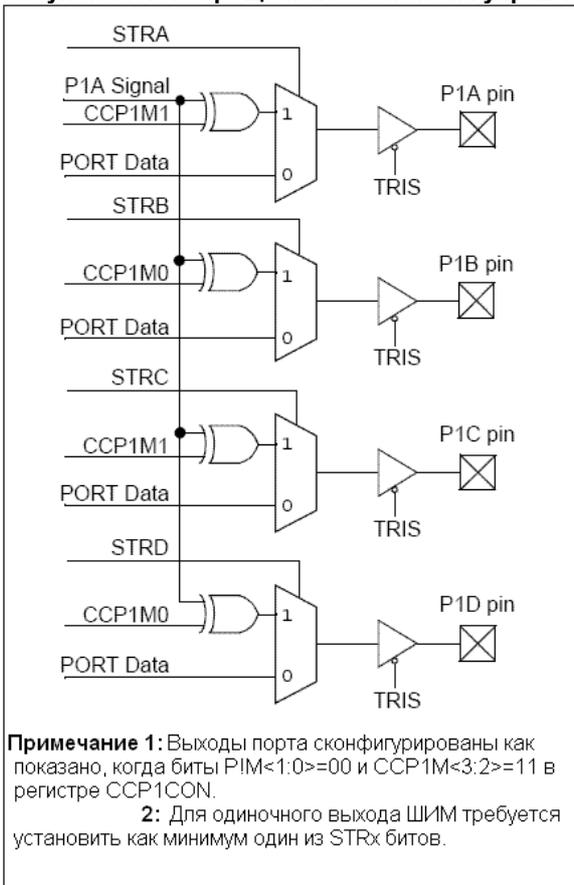
U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
—	—	—	STRSYNC	STRD	STRC	STRB	STRA
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Легенда:</b>			
R = Читаемый бит	W = Записываемый бит	U = Недействительный бит, читается как '0'	
-n = Значение при POR	'1' = Бит установлен	'0' = Бит сброшен	x = Бит не определён

- Биты 7-5 **Не задействованные:** Читаются как '0'
- Бит 4 **STRSYNC:** Бит синхронизации управления
  - 1 = Управление коррекции выхода происходит в следующем периоде ШИМ
  - 0 = Управление коррекции выхода происходит на границе инструкций
- Бит 3 **STRD:** Бит D разрешения управления
  - 1 = на ножке P1D ШИМ сигнал с полярностью определяемой CCP1M<1:0>
  - 0 = ножка P1D назначена как ножка порта
- Бит 2 **STRC:** Бит C разрешения управления
  - 1 = на ножке P1C ШИМ сигнал с полярностью определяемой CCP1M<1:0>
  - 0 = ножка P1C назначена как ножка порта
- Бит 1 **STRB:** Бит B разрешения управления
  - 1 = на ножке P1B ШИМ сигнал с полярностью определяемой CCP1M<1:0>
  - 0 = ножка P1B назначена как ножка порта
- Бит 0 **STRA:** Бит A разрешения управления
  - 1 = на ножке P1A ШИМ сигнал с полярностью определяемой CCP1M<1:0>
  - 0 = ножка P1A назначена как ножка порта

**Примечание 1:** Режим ШИМ управления доступен только когда в регистре CCP1CON биты CCP1M<3:2> = 11 и P1M<1:0> = 00.

**Рисунок 11-18: Упрощённая блок схема управления**



#### 11.4.7.1 Управление синхронизацией

Бит STRSYNC регистра PSTRCON даёт пользователю два выбора, когда событие управления может случиться. Когда бит STRSYNC равен '0', событие управления может случиться при конце инструкции которая записывается в регистр PSTRCON. В этом случае выходной сигнал на ножках P1<D:A> может иметь не полную форму ШИМ. Это действие оказывается полезным, когда программа пользователя нуждается в немедленном удалении ШИМ сигнала с ножки.

Когда бит STRSYNC установлен в '1', эффективное управление коррекцией может случиться в начале следующего периода ШИМ. В этом случае управление вкл/выкл выхода ШИМ может производить полный ШИМ сигнал.

**Рисунок 11-19: Пример события управления - конец инструкции (STRSYNC = 0)**

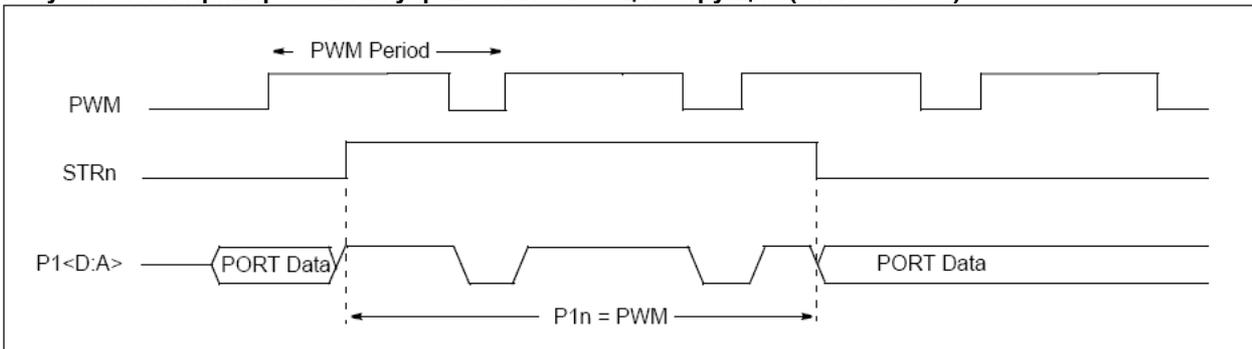


Рисунок 11-20: Пример события управления – начало инструкции (STRSYNC = 1)

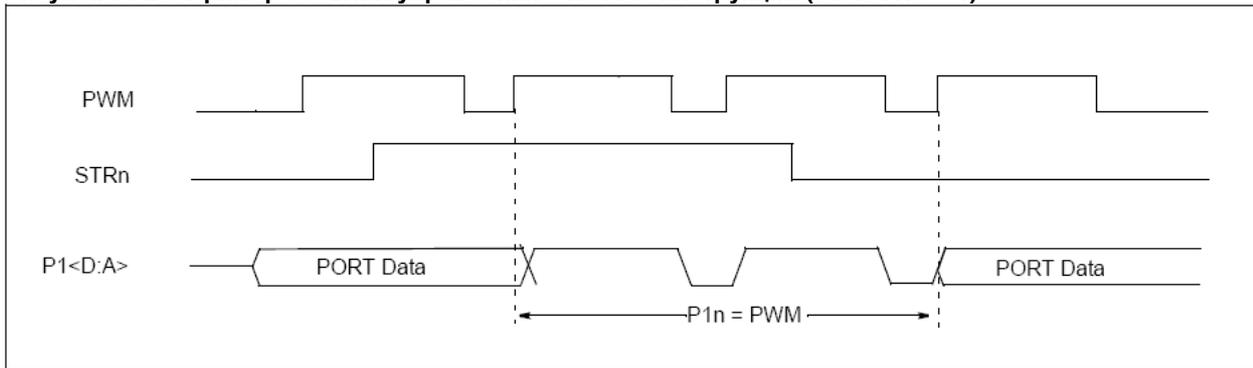


TABLE 11-5: Регистры связанные с модулем захват/сравнение/ШИМ

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Value on POR, BOR	Value on all other Resets
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1R	C1CH1	C1CH0	0000 -000	0000 -000
CM2CON0	C2ON	C2OUT	C2OE	C2POL	—	C2R	C2CH1	C2CH0	0000 -000	0000 -000
CM2CON1	MC1OUT	MC2OUT	—	—	—	—	T1GSS	C2SYNC	00-- --10	00-- --10
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	uuuu uuuu
CCPR1H	Capture/Compare/PWM Register 1 High Byte								xxxx xxxx	uuuu uuuu
ECCPAS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000	0000 0000
INTCON	GIE	PEIE	T0IE	INTE	RABIE	T0IF	INTF	RABIF	0000 0000	0000 0000
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
PSTRCON	—	—	—	STRSYNC	STRD	STRC	STRB	STRA	---0 0001	---0 0001
PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	0000 0000
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR2	Timer2 Module Register								0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Легенда: — = Недействующие позиции, читаются как '0', u = неизменно, x = неопределенно. Затенённые ячейки не используются в захвате, сравнении и ШИМ.

## 14.0 Специальные характеристики ЦПУ

PIC16F631/677/685/687/689/690 имеет главные особенности, предполагающие максимизацию системной надёжности, минимизацию стоимости через устранение навесных компонентов, экономия энергии и защита кода.

Эти характеристики есть:

- Сброс
- Сброс по включению питания (POR)
- Таймер включения питания (PWRT)
- Таймер запуска генератора (OST)
- Сброс по понижению напряжения питания (BOR)
- Прерывания
- сторожевой таймер (WDT)
- Выбор генератора
- Спящий режим
- Защита кода
- ID позиция
- Внутрисхемное последовательное программирование

PIC16F631/677/685/687/689/690 имеет два таймера, которые предлагают необходимые задержки по включению питания. Один есть таймер запуска генератора (OST), удерживающий чип в состоянии сброса, пока кристаллический генератор не стабилизируется. Другой таймер включения питания (PWRT), который обеспечивает фиксированную задержку 64 ms (номинал) на включение питания только, разработан удерживать часть в состоянии сброса, пока напряжение питания не стабилизируется. Имеется так же схема сброса устройства, если понижение питания происходит, что может использовать таймер включения питания для обеспечения минимальной длительности сброса в 64 ms. С этими тремя встроенными функциями большинство приложений не нуждаются во внешней схеме сброса.

Спящий режим был разработан для обеспечения режима очень низкого потребления энергии. Контроллер может пробуждаться из спящего режима используя:

- Внешний сброс
- Пробуждение от сторожевого таймера
- Прерывание

Различные опции генератора сделаны доступными, чтобы лучше приспособиваться под прикладную задачу. Опция INTOSC сохраняет системную стоимость, в то время как опция кристалла LP сохраняет мощность. Установка битов конфигурации используется, чтобы выбрать различные параметры (смотреть регистр 14-2).

### 14.1 Биты конфигурации

Биты конфигурации могут быть запрограммированы (читаются как '0'), или оставленными не запрограммированными (читаются как '1') для выбора различных конфигураций устройства, как показано в регистре 14-2. Эти биты отображаются в программной памяти по адресу 2007h.

**Примечание:** Адрес 2007h находится вне пространства памяти программы пользователя. Он принадлежит специальной конфигурационной области памяти (2000h-3FFFh), которая может быть доступна только в течении программирования. See "*PIC12F6XX/16F6XX спецификация программирования памяти*" (DS41204) для большей информации.

## Регистр 14-1: CONFIG: Регистр слова конфигурации

Зарезервирован	Зарезервирован	FCMEN	IESO	BOREN1 <sup>(1)</sup>	BOREN0 <sup>(1)</sup>	CPD <sup>(2)</sup>
Бит 13						Бит 7

CP <sup>(3)</sup>	MCLRE <sup>(4)</sup>	/PWRT <sup>E</sup>	WDTE	FOSC2	FOSC1	FOSC0
Бит 6						Бит 0

<b>Описание:</b>			
R = читаемый бит	W = записываемый бит	P = программируемый	U = не задействованный бит, читается как '0'
-n = значение POR	'1' = бит установлен	'0' = бит очищен	x = бит не определен

Биты 13-12

**Зарезервированы:** Резервные биты. Не используются.

Бит 11

**FCMEN:** Бит разрешения монитора защиты от аварии тактирования

1 = монитор включен

0 = монитор выключен

Бит 10

**IESO:** Бит режима переключателя внутренний-внешний

1 = режим включен

0 = режим выключен

Биты 9-8

**BOREN<1:0>:** Биты выбора сброса по снижению питания<sup>(1)</sup>

11 = BOR включен

10 = BOR включен в работе и выключен в спящем режиме

01 = BOR управляется битом SBOREN регистра PCON

00 = BOR выключен

Бит 7

**/CPD:** Бит защиты данных<sup>(2)</sup>

1 = защита памяти данных выключена

0 = защита памяти данных включена

Бит 6

**/CP:** Бит защиты кода<sup>(2)</sup>

1 = защита кода выключена

0 = защита кода включена

Бит 5

**MCLRE:** Бит выбора функции ножки MCLR<sup>(3)</sup>

1 = функция ножки /MCLR, есть /MCLR

0 = функция ножки /MCLR есть цифровой вход, /MCLR непосредственно привязан к VDD

Бит 4

**/PWRT<sup>E</sup>:** Бит таймера включения питания

1 = PWRT выключен

0 = PWRT включен

Бит 3

**WDTE:** Бит разрешения сторожевого таймера

1 = WDT включен

0 = WDT выключен

Бит 2-0

**FOSC<2:0>:** Биты выбора генератора

111 = RC генератор: Функция CLKOUT на ножке RA4/OSC2/CLKOUT, RC на RA5/OSC1/CLKIN

110 = RCIO генератор: Функция I/O на ножке RA4/OSC2/CLKOUT, RC на RA5/OSC1/CLKIN

101 = INTOSC генератор: Функция CLKOUT на ножке RA4/OSC2/CLKOUT, функция I/O на ножке RA5/OSC1/CLKIN

100 = INTOSCIO генератор: Функция I/O на ножке RA4/OSC2/CLKOUT, функция I/O на ножке RA5/OSC1/CLKIN

011 = EC: Функция I/O на ножке RA4/OSC2/CLKOUT, CLKIN на RA5/OSC1/CLKIN

010 = HS генератор: Высокоскоростной кристалл/резонатор на RA4/OSC2/CLKOUT и RA5/OSC1/CLKIN

001 = XT генератор: Кристалл/резонатор на RA4/OSC2/CLKOUT и RA5/OSC1/CLKIN

000 = LP генератор: Мало потребляющий кристалл на RA4/OSC2/CLKOUT и RA5/OSC1/CLKIN

### Примечание

**1:** Включение сброса по снижению питания не делает автоматического включения таймера включения питания.

**2:** Данные EEPROM могут быть стёрты, когда выключается защита данных.

**3:** Программная память может быть стёрта, когда выключается защита кода.

**4:** Когда MCLR утверждён в режимах INTOSC или RC режиме, внутренний тактовый генератор выключен.

## 14.2 Сброс

Для PIC16F631/677/685/687/689/690 имеются различные типы сброса:

- a) Сброс по включению питания (POR)
- b) WDT сброс в течении нормальной работы
- c) WDT сброс в течении спящего режима
- d) MCLR сброс в течении нормальной работы
- e) MCLR сброс в течении спящего режима
- f) Сброс по снижению питания (BOR)

На некоторые регистры сброс не воздействует; их состояние неизвестно после POR и не изменяется после другого сброса.

Большинство других регистров сбрасываются в "Состояние сброса" на:

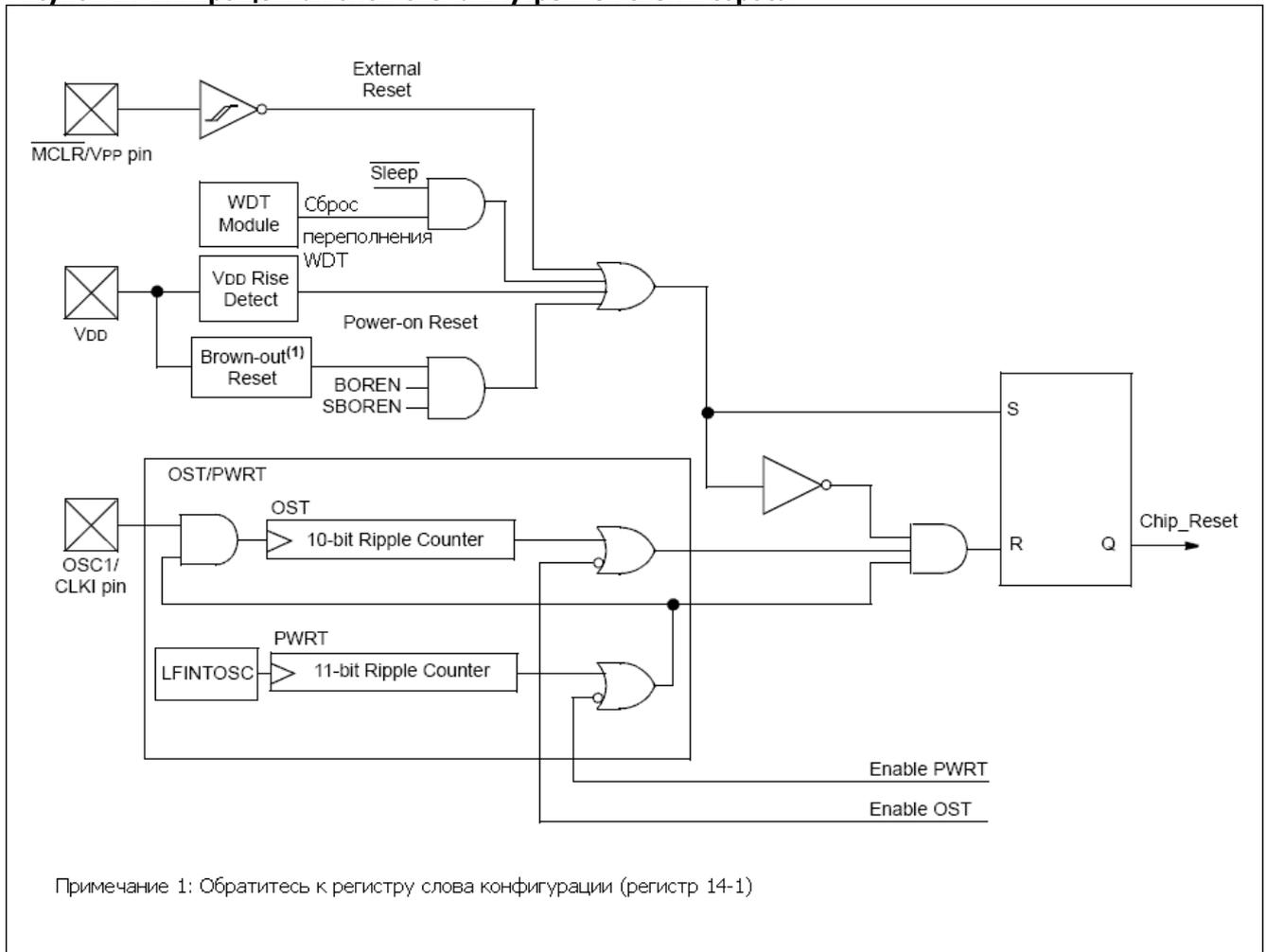
- Сброс по включению питания
- MCLR сброс
- MCLR сброс в течении спящего режима
- WDT сброс
- Сброс по снижению напряжения питания (BOR)

На них не воздействует WDT пробуждение, поскольку это рассматривается как возобновление нормальной работы. TO и PD биты устанавливаются или сбрасываются по разному в разных ситуациях сброса, как показано в таблице 14-2. Эти биты используются в программе для определения характера сброса. Смотреть таблицу 14-4 для полного описания состояний всех регистров после сброса.

Упрощённая блок-схема внутренней схемы сброса показана на рис. 14-1.

Сброс MCLR проходит через фильтр шума, игнорирующий малые пульсации. Смотреть **Часть 17.0 "Электрические спецификации"** для спецификаций ширины импульса.

**Рисунок 14-1: Упрощённая блок-схема внутренней схемы сброса**



### 14.2.1 Сброс по включению питания (POR)

Внутренняя POR схема удерживает чип в состоянии сброса, пока VDD достигнет высокого уровня, достаточного для нормальной работы. Требуется максимальная скорость нарастания VDD. Смотрите **Часть 17.0 "Электрических спецификаций" для деталей**. Если BOR включен, максимальная скорость нарастания не актуальна. Схема BOR будет удерживать устройство в состоянии сброса, пока VDD достигнет VBOR (смотрите **Часть 14.2.4 "Сброс по понижению питания (BOR)"**).

**Примечание:** POR схема не генерирует внутренний сброс, когда VDD уменьшается. Для перезапуска POR, VDD должно достигать Vss за минимум 100  $\mu$ s.

Когда устройство стартует для нормальной работы (выходит из состояния сброса), рабочие параметры устройства (т.е., напряжение, частота, температура и т.д.) должны удовлетворять нормальной работе. Если эти состояния не удовлетворяют, устройство должно удерживаться в состоянии сброса, пока параметры работы не станут удовлетворительными. Для дополнительной информации обратиться к документу AN607, "Проблемы включения" (DS00607).

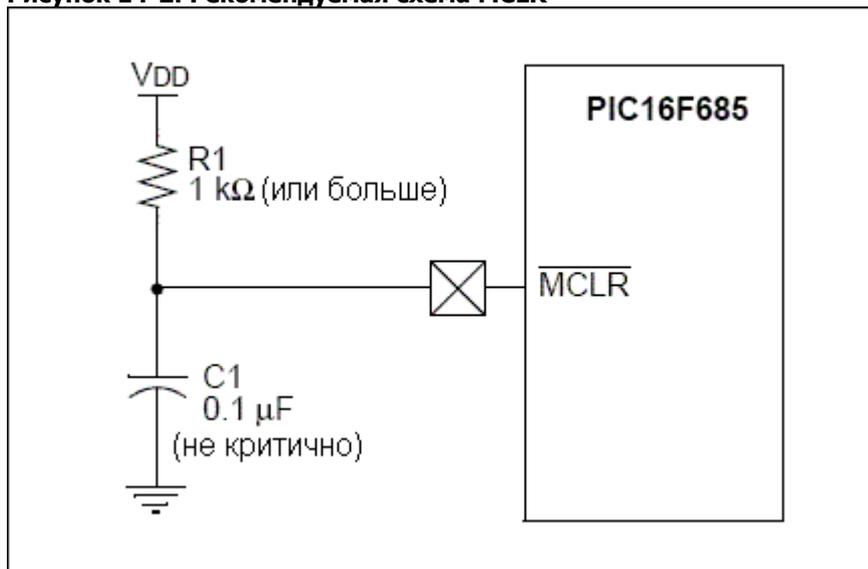
### 14.2.2 MCLR

PIC16F631/677/685/687/689/690 имеют фильтр шума в MCLR пути сброса. Фильтр отсеивает маленькие пульсации. Это стоит отметить, что WDT сброс не переводит MCLR ножку в низкое состояние.

Поведение ESD (диодной) защиты на ножке MCLR может отличаться от ранних устройств этого семейства. Напряжение, прикладываемое к ножке, превышающее эту спецификацию, может привести к сбросам и чрезмерному току в течении ESD события. По этой причине, Microchip рекомендует не делать непосредственной связи MCLR с VDD. Используйте RC цепочку, как показано на рисунке 14-2.

Внутренняя опция MCLR разрешается очисткой бита MCLRE в регистре слова конфигурации. Когда MCLRE = 0, сигнал сброса чипа генерируется внутри. Когда MCLRE = 1, ножка RA3/MCLR становится вводом внешнего сброса. В этом режиме, ножка RA3/MCLR имеет подтяжку к VDD.

**Рисунок 14-2: Рекомендуемая схема MCLR**



### 14.2.3 Таймер включения питания (PWRT)

Таймер включения питания обеспечивает фиксированное переполнение 64 ms (номинал) на включение питания, от POR или BOR. Таймер включения работает от 31 kHz генератора LFINTOSC. Для большей информации смотрите **Часть 3.5 "Режимы внутреннего тактирования"**. Чип удерживается сбросом так долго, как активен PWRT. Задержка PWRT позволяет VDD подняться до приемлемого уровня. Бит конфигурации PWRT $\bar{E}$  может выключить (если установлен) или выключить (если очищен или программируем) таймер включения питания. Таймер включения питания должен быть разрешен, когда разрешен сброс по понижению питания, хотя это не обязательно.

Таймер включения питания может изменяться от чипа к чипу в зависимости от:

- Изменения VDD
- Изменения температуры
- Изменения процесса

Смотреть DC параметры для подробностей (**Часть 17.0 "Электрических спецификаций"**).

## 14.2.4 Сброс по снижению напряжения (BOR)

Биты BOREN0 и BOREN1 в регистре слова конфигурации выбирают один из четырёх режимов BOR. Два режима, которые были добавлены, допускают программный или аппаратный контроль включения BOR. Когда BOREN<1:0> = 01, бит SBOREN (PCON<4>) разрешает/запрещает программе контролировать BOR. Выбор BOREN<1:0>, автоматически выключает BOR в спящем режиме, для сохранения энергии, и включает его при пробуждении. В этом режиме бит SBOREN выключен. Смотрите регистр 14-2 для определений слова конфигурации.

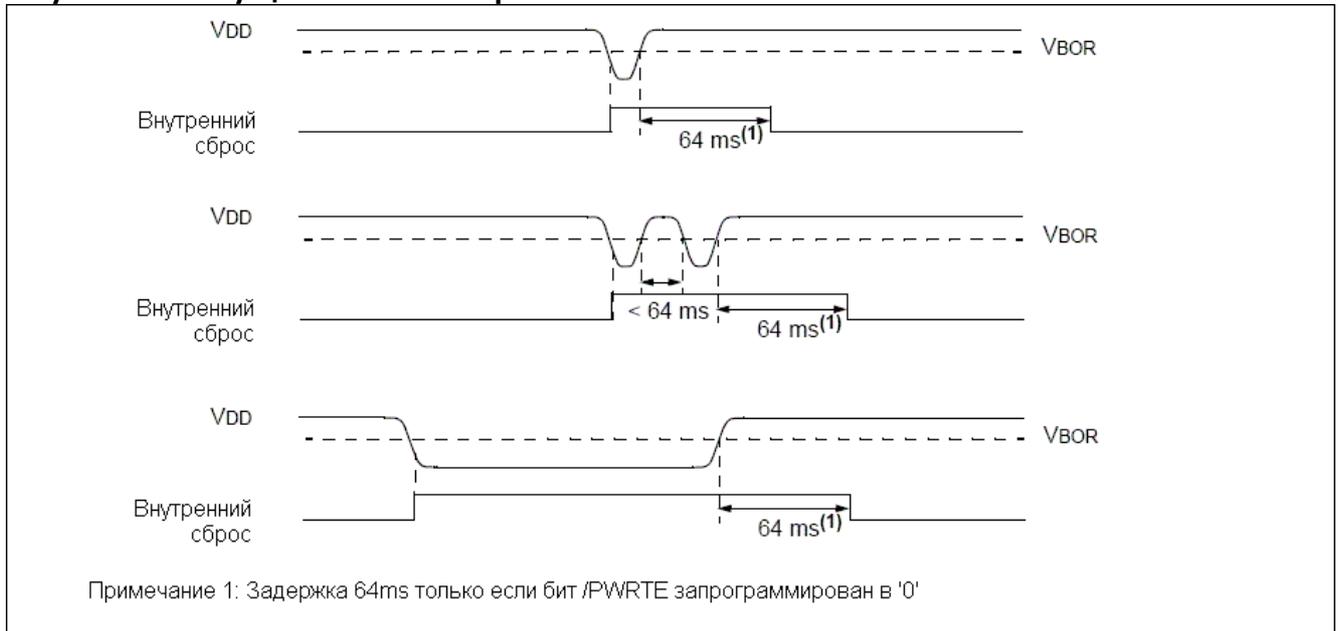
Если VDD упало ниже VBOR более чем параметр (TBOR) (смотрите **Часть 17.0 "Электрические спецификации"**), ситуация снижения напряжения сбросит устройство. Это будет происходить независимо от скорости снижения VDD. Сброс может не произойти, если VDD опустится ниже VBOR меньше чем параметр (TBOR).

На любой сброс (POR, BOR, от сторожевого таймера и т.д.), чип будет оставаться сброшенным, пока VDD не вырастит выше VBOR (смотрите рисунок 14-3). После этого будет запущен таймер включения питания, если он разрешен, и продолжит удержание чипа в состоянии сброс ещё 64 ms.

**Примечание:** Таймер включения питания разрешается битом PWRTS в регистре слова конфигурации.

Если VDD упадёт ниже VBOR во время работы таймера включения питания, чип перейдёт обратно в сброс по понижению напряжения и таймер включения питания будет перезапущен. Как только VDD превысит VBOR, таймер включения питания вновь будет выполнять сброс в течении 64.

**Рисунок 14-3: Ситуации снижения напряжения**



## 14.2.5 Последовательность завершения счёта

На подачу питания следующая последовательность: сначала, переполняется PWRT, после того как POR истёк, затем, после переполнения PWRT, активизируется OST. Общая задержка может изменяться в зависимости от конфигурации генератора и бита состояния PWRT. Например, в режиме EC, с очищенным битом PWRT (PWRT выключен), переполнения не будет вообще. Рисунки 14-4, 14-5 и 14-6 изображают последовательности счёта. Устройство может выполнять код от INTOSC пока OST активен, при включенном двухскоростном запуске или мониторе защиты от аварии тактирования (смотреть **Часть 3.7.2**

**“Последовательность двухскоростного старта” and Часть 3.8 “Монитор защиты от аварии тактирования”**).

Поскольку счёт запускается импульсом POR, если на MCLR достаточно долго удерживается низкий уровень, таймеры будут исчерпаны. Затем, появление высокого уровня на MCLR начнёт исполнение немедленно (смотреть рисунок 14-5). Это полезно исключительно для испытания или синхронизации нескольких устройств PIC16F631/677/685/ 687/689/690 при их параллельной работе.

В таблице 14-5 показаны последовательности сброса для некоторых специальных регистров, пока таблица 14-4 показывает последовательности сброса для всех регистров.

## 14.2.6 Регистр управления мощностью (PCON)

Регистр управления мощностью PCON (адрес 8Eh) имеет два бита состояния, которые показывают тип сброса, произошедший последним.

Бит 0 есть BOR (Сброс по понижению напряжения питания). BOR не узнаётся на сброс по подаче питания. Пользователю необходимо установить его и проверять на последующих сбросах, если BOR = 0, то это показывает, что произошёл сброс по снижению питания. Бит состояния BOR безразличен и не предсказуем, если BOR схема выключена (BOREN<1:0> = 00 в регистре слова конфигурации).

Бит 1 есть POR (Сброс по включению питания). Устанавливается в '0' на сброс от подачи питания и безразличен к другим сбросам. Пользователь должен записать '1' в этот бит. После последующего сброса, если POR равен '0', это означает, что произошёл сброс по включению питания (т.е., VDD значительно снижалось).

Для большей информации смотрите **Часть 4.2.4 “Ультра низкое потребление пробуждения” и Част 14.2.4 “Сброс по снижению напряжения питания (BOR)”**.

**Таблица 14-1: Завершение счёта в различных ситуациях**

Конфигурация генератора	Подача питания		Сброс по снижению		Пробуждение из спящего режима
	/PWRT = 0	/PWRT = 1	/PWRT = 0	/PWRT = 1	
XT, HS, LP	TPWRT + 1024 • TOSC	1024 • TOSC	TPWRT + 1024 • TOSC	1024 • TOSC	1024 • TOSC
LP, T1OSCIN = 1	TPWRT	-	TPWRT	-	-
RC, EC, INTOSC	TPWRT	-	TPWRT	-	-

**Таблица 14-2: Биты STATUS/PCON и их значение**

POR	BOR	/TO	/PD	Состояние
0	X	1	1	POR
U	0	1	1	BOR
U	U	0	U	WDT сброс
U	U	0	0	WDT пробуждение
U	U	U	U	/MCLR сброс в течении нормальной работы
U	u	1	0	/MCLR сброс в течении спящего режима

Описание: u=неизменные, X=неизвестные

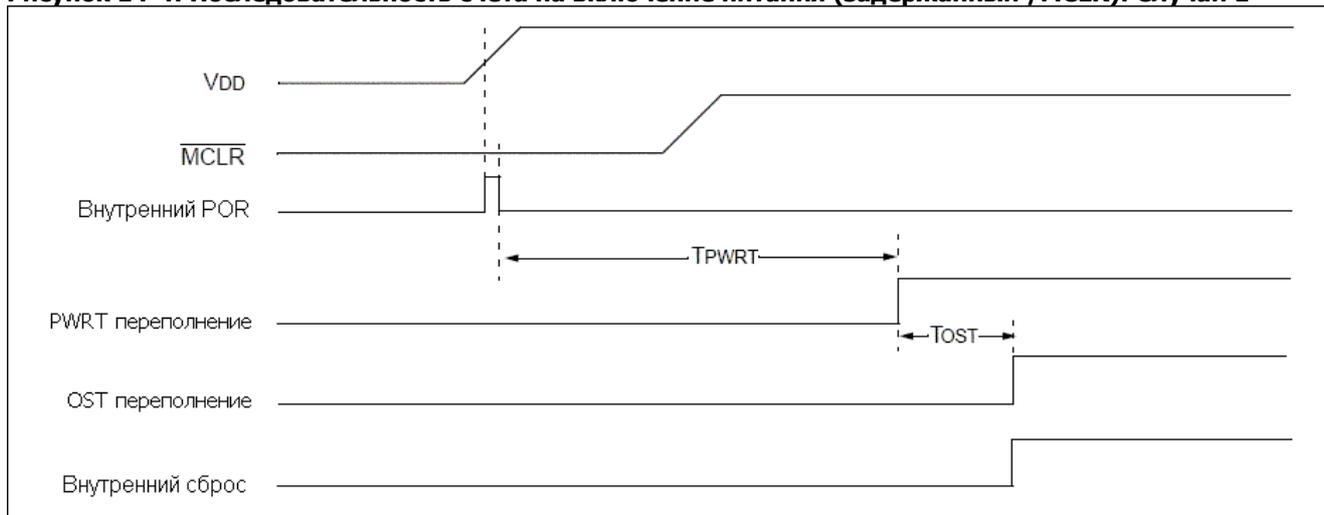
**TABLE 14-3: Список регистров ассоциированных со снижением питания**

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Значение на POR, BOR	Значение на все остальные сбросы
PCON	-	-	ULPWUE	SBOREN	-	-	/POR	/BOR	--01 --qq	--0u --uu
STATUS	IRP	RP1	RPO	/TO	/PD	Z	DC	C	0001 1xxx	000q quuu

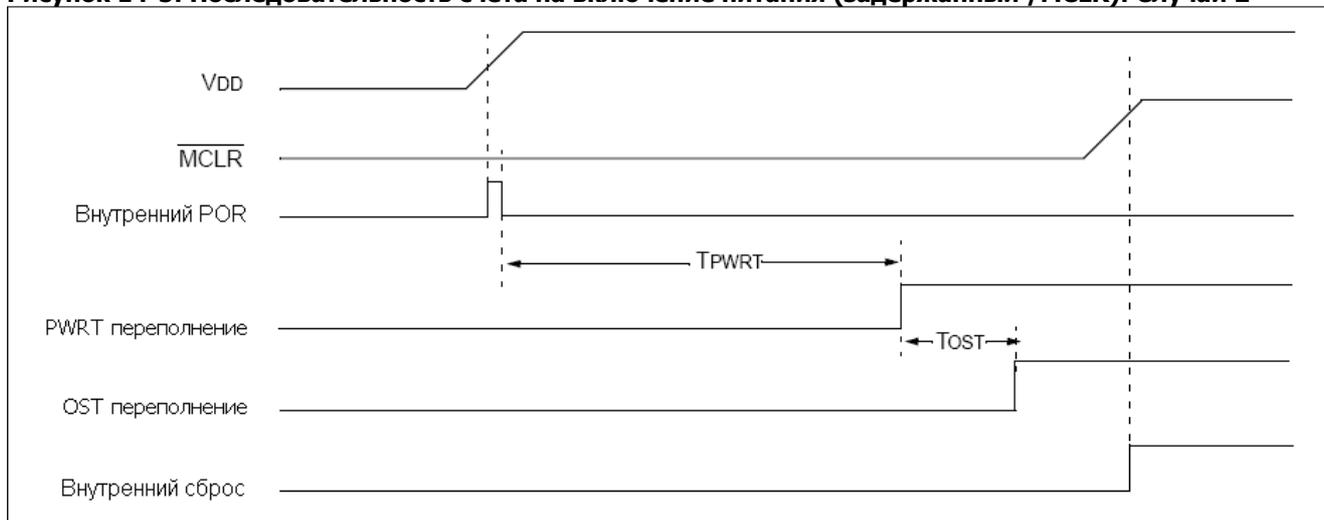
**Описание:** u = неизменный, x = неизвестный, – = незадействованный тип, читается как '0', q = значение зависимое от условия. Затенённые ячейки не используются для BOR.

**Примечание 1:** Другие (не на повышение напряжения) сбросы, включая MCLR сброс и сброс от сторожевого в течении нормальной работы.

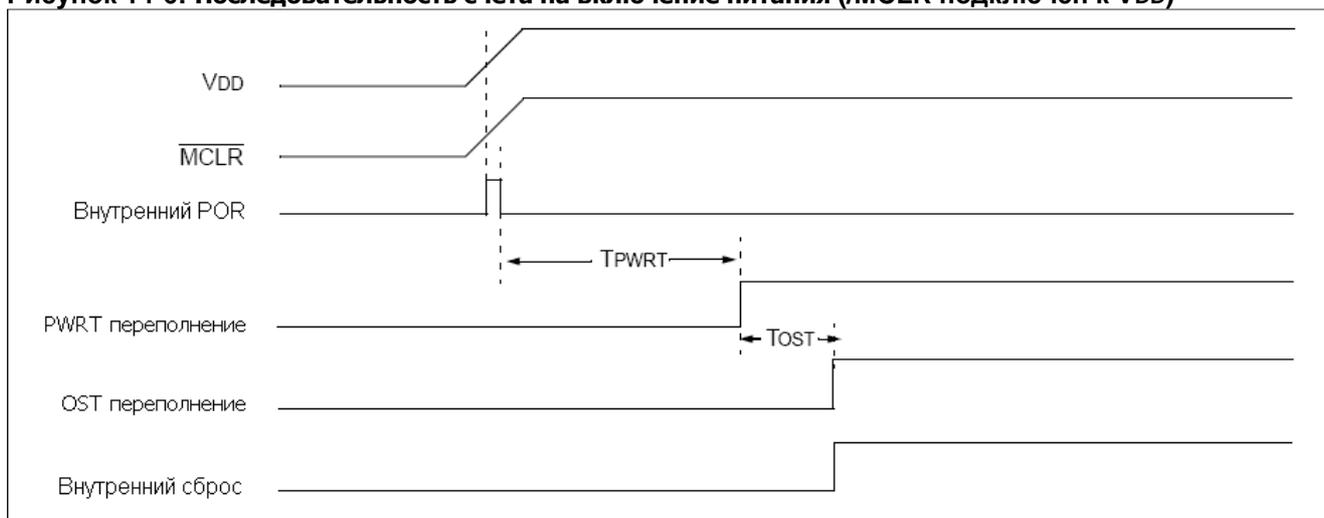
**Рисунок 14-4: Последовательность счёта на включение питания (задержанный /MCLR): Случай 1**



**Рисунок 14-5: Последовательность счёта на включение питания (задержанный /MCLR): Случай 2**



**Рисунок 14-6: Последовательность счёта на включение питания (/MCLR подключен к VDD)**



**Таблица 14-4: Состояние инициализации для регистра**

Регистр	Адрес	POR	MCLR сброс, WDT сброс, BOR <sup>(1)</sup>	Пробуждение из спящего режима Через прерывание. Пробуждение из спящего режима сторожевым таймером.
W	-	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h/100h/180h	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0	01h/101h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h/102h/182h	0000 0000	0000 0000	PC + 1 <sup>(3)</sup>
STATUS	03h/83h/103h/183h	0001 1xxx	000q quuu <sup>(4)</sup>	uuuq quuu <sup>(4)</sup>
FSR	04h/84h/104h/184h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	05h/105h	--xx xxxx	--uu uuuu	--uu uuuu
PORTB	06h/106h	Xxxx ----	Uuuu ----	Uuuu ----
PORTC	07h/107h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCLATH	0Ah/8Ah/10Ah/18Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh/10Bh/18Bh	0000 000x	0000 000u	uuuu uuuu <sup>(2)</sup>
PIR1	0Ch	-000 0000	-000 0000	-uuu uuuu <sup>(2)</sup>
PIR2	0Dh	0000 ----	0000 ----	uuuu ---- <sup>(2)</sup>
TMR1L	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h	0000 0000	uuuu uuuu	uuuu uuuu
TMR2	11h	0000 0000	0000 0000	uuuu uuuu
T2CON	12h	-000 0000	-000 0000	-uuu uuuu
SSPBUF	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	14h	0000 0000	0000 0000	uuuu uuuu
CCPR1L	15h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	16h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	17h	0000 0000	0000 0000	uuuu uuuu
RCSTA	18h	0000 000x	0000 000x	uuuu uuuu
TXREG	19h	0000 0000	0000 0000	uuuu uuuu
RCREG	1Ah	0000 0000	0000 0000	uuuu uuuu
PWM1CON	1Ch	0000 0000	0000 0000	uuuu uuuu
ECCPAS	1Dh	0000 0000	0000 0000	uuuu uuuu
ADRESH	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	1Fh	0000 0000	0000 0000	uuuu uuuu
OPTION_REG	81/181h	1111 1111	1111 1111	uuuu uuuu
TRISA	85h/185h	--11 1111	--11 1111	--uu uuuu

**Обозначения:** u = неизменный, x = неизвестный, — = не задействованный бит, читается как '0', q = значение зависит от состояния.

**Примечание 1:** Если VDD снижается, POR может быть активизирован и регистры будут воздействованы иначе.

**2:** Один или более бит в INTCON и/или PIR1 будут деланный (заставлять пробуждаться).

**3:** Когда пробуждение происходит по прерыванию и бит GIE установлен, в PC загружается вектор прерывания (0004h).

**4:** Смотреть таблицу 14-5 значений сброса для специфических состояний.

**5:** Если сброс был от понижения напряжения, тогда бит 0 = 0. Все другие сбросы вызывают бит 0 = u.

**6:** Доступный только когда SSPM<3:0> = 1001.

**Таблица 14-4: Состояние инициализации для регистра (продолжение)**

Регистр	Адрес	POR	MCLR сброс, WDT сброс, BOR <sup>(1)</sup>	Пробуждение из спящего режима Через прерывание. Пробуждение из спящего режима сторожевым таймером.
TRISB	86h/186h	1111 ----	1111 ----	uuuu ----
TRISC	87h/187h	1111 1111	1111 1111	uuuu uuuu
PIE1	8Ch	-000 0000	-000 0000	-uuu uuuu
PIE2	8Dh	0000 ----	0000 ----	uuuu uuuu
PCON	8Eh	--01 --0x	--0u -uq <sup>1,5)</sup>	--uu -uu
OSCCON	8Fh	-110 q000	-110 q000	-uuu uuuu
OSTUNE	90h	---0 0000	---u uuuu	---u uuuu
PR2	92h	1111 1111	1111 1111	uuuu uuuu
SSPADD	93h	0000 0000	1111 1111	uuuu uuuu
SSPMSK <sup>(6)</sup>	93h	---- ----	1111 1111	uuuu uuuu
SSPSTAT	94h	0000 0000	1111 1111	uuuu uuuu
WPUA	95h	--11 -111	--11 -111	uuuu uuuu
IOCA	96h	--00 0000	--00 0000	--uu uuuu
WDTCON	97h	---0 1000	---0 1000	---u uuuu
TXSTA	98h	0000 0010	0000 0010	uuuu uuuu
SPBRG	99h	0000 0000	0000 0000	uuuu uuuu
SPBRGH	9Ah	0000 0000	0000 0000	uuuu uuuu
BAUDCTL	9Bh	01-0 0-00	01-0 0-00	uu-u u-uu
ADRESL	9Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON1	9Fh	-000 ---	-000 ---	-uuu ----
EEDAT	10Ch	0000 0000	0000 0000	uuuu uuuu
EEADR	10Dh	0000 0000	0000 0000	uuuu uuuu
EEDATH	10Eh	--00 0000	--00 0000	--uu uuuu
EEADRH	10Fh	---- 0000	---- 0000	---- uuuu
WPUB	115h	1111 ----	1111 ----	uuuu ----
IOCB	116h	0000 ----	0000 ----	uuuu ----
VRCON	118h	0000 0000	0000 0000	uuuu uuuu
CM1CON0	119h	0000 -000	0000 -000	uuuu -uuu
CM2CON0	11Ah	0000 -000	0000 -000	uuuu -uuu
CM2CON1	11Bh	00-- --00	00-- --10	uu-- --uu
ANSEL	11Eh	1111 1111	1111 1111	uuuu uuuu
ANSELH	11Fh	---- 1111	---- 1111	---- uuuu
EECON1	18Ch	x--- x000	u--- q000	---- uuuu
EECON2	18Dh	---- ----	---- ----	---- ----
PSTRCON	19Dh	---0 0001	---0 0001	---u uuuu
SRCON	19Eh	0000 00--	0000 00--	uuuu uu--

**Обозначения:** u = неизменный, x = неизвестный, — = не задействованный бит, читается как '0', q = значение зависит от состояния.

**Примечание 1:** Если VDD снижается, POR может быть активизирован и регистры будут воздействованы иначе.

**2:** Один или более бит в INTCON и/или PIR1 будут деланный (заставлять пробуждаться).

**3:** Когда пробуждение происходит по прерыванию и бит GIE установлен, в PC загружается вектор прерывания (0004h).

**4:** Смотреть таблицу 14-5 значений сброса для специфических состояний.

**5:** Если сброс был от понижения напряжения, тогда бит 0 = 0. Все другие сбросы вызывают бит 0 = u.

**6:** Доступный только когда SSPM<3:0> = 1001.

**Таблица 14-5: Состояние инициализации для специальных регистров**

Состояние	Программный счётчик	Регистр статуса	Регистр PCON
POR	000h	0001 1xxx	--01 --0x
Сброс /MCLR в течении нормальной работы	000h	000u uuuu	--0u --uu
Сброс /MCLR в течении спящего режима	000h	0001 0uuu	--0u --uu
Сброс от WDT	000h	0000 uuuu	--0u --uu
Пробуждение от WDT	PC + 1	uuu0 0uuu	--uu --uu
BOR	000h	0001 1uuu	--01 --u0
Пробуждение из спящего режима по прерыванию	PC + 1 <sup>(1)</sup>	uuu1 0uuu	--uu --uu

**Обозначения:** u = неизменный, x = неизвестный, — = не задействованный бит, читается как '0'.

**Примечание 1:** Когда пробуждение происходит по прерыванию и бит GIE установлен, в PC загружается вектор прерывания (0004h) после выполнения инструкции по адресу PC + 1.

## 14.3 Прерывания

PIC16F631/677/685/687/689/690 имеют много источников прерывания:

- Внешнее прерывание RA2/INT
- Прерывание по переполнению TMR0
- Прерывание по изменению уровня на PORTA/PORTB
- Прерывания от 2-х компараторов
- Прерывание от АЦП (кроме PIC16F631)
- Прерывание по переполнению Timer1
- Прерывание по совпадению Timer2 (только PIC16F685/PIC16F690)
- Прерывание записи данных EEPROM
- Прерывание от монитора защиты тактирования
- Прерывание Расширенного (только PIC16F685/PIC16F690)
- Прерывание приёмника и передатчика EUSART (только PIC16F687/PIC16F689/PIC16F690)

Регистр управления прерыванием (INTCON) и регистр запроса периферийных прерываний 1 (PIR1) записывают индивидуальные запросы в биты флагов. Регистр INTCON так же имеет биты разрешения индивидуальных и бит глобального разрешения прерываний.

Бит глобального разрешения прерываний GIE (INTCON<7>), разрешает (если установлен) все немаскируемые прерывания, или запрещает (если очищен) все прерывания. Индивидуальные прерывания могут быть запрещены через их соответствующие биты разрешения в регистрах INTCON, PIE1 и PIE2, соответственно. GIE очищается по сбросу.

Инструкция возврата из прерывания, RETFIE, позволяющая выйти из процедуры обработки прерывания, так же устанавливает бит GIE, вновь разрешая немаскируемые прерывания.

Регистр INTCON содержит следующие флаги прерывания:

- Прерывание от ножки INT
- Прерывания по изменению уровней на PORTA/PORTB
- Прерывание по переполнению TMR0

Регистры PIR1 и PIR2 содержат флаги периферийных прерываний. Регистры PIE1 и PIE2 содержат соответствующие биты разрешения прерываний.

Регистр PIR1 содержит следующие флаги прерываний:

- Прерывание АЦП
- Прерывания приёмника и передатчика EUSART
- Прерывание синхронного последовательного порта (SSP)
- Прерывание от расширенного CCP1
- Прерывание переполнения Timer1
- Прерывание по совпадению Timer2

Регистр PIR2 содержит следующие флаги прерывания:

- Прерывание от монитора защиты тактирования
- Прерывания от 2-х компараторов
- Прерывание записи данных EEPROM

Когда прерывание обслуживается:

- Бит GIE очищен для запрета любых последующих прерываний.
- Адреса возврата сохранены в стеке.
- PC загружен с 0004h.

Для внешних событий прерывания, таких как INT, PORTA/PORTB, время ожидания прерывания может составлять три или четыре цикла инструкций. Точное время ожидания зависит от того, когда происходит событие прерывания (смотреть Рисунок 14-8). Время ожидания такое же для одно или двух циклов команд. Источник(и) прерывания определяются программой обслуживания прерывания по установленным флажкам прерывания. Бит(ы) флагов прерывания должны быть очищены программно перед выходом из прерывания, для предотвращения закливания программы обработки прерывания.

**Примечание 1:** Индивидуальные биты флагов прерывания устанавливаются независимо от соответствующих бит маски или бита GIE.

**2:** When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts, which were ignored, are still pending to be serviced when the GIE bit is set again.

Когда команда, которая очищает GIE бит, выполнена, любое прерывание, которое ожидало выполнение в следующем цикле, игнорируются. Прерывание, которое игнорировалось, будут ожидать обслуживания, когда бит GIE будет установлен вновь.

Для дополнительной информации на модули Timer1, Timer2, компараторы, АЦП, EEPROM, EUSART, SSP или расширенный CCP, обращайтесь к соответствующей части описания.

### 14.3.1 Прерывание RA2/INT

Внешнее прерывание от ножки RA2/INT происходит по перепаду уровней; любое нарастание, если бит INTEDG (OPTION\_REG<6>) установлен, или падение, если бит INTEDG очищен. Когда правильный фронт появляется на ножке RA2/INT, бит INTF(INTCON<1>) устанавливается. Это прерывание может быть выключено очисткой управляющего бита INTE (INTCON<4>). Бит INTF должен быть очищен в программе в процедуре обслуживания прерывания прежде чем разрешить это прерывание. Прерывание от RA2/INT может пробудить процессор из спящего режима, если бит INTE был установлен перед уходом в спячку. Состояние бита определяет, действительно ли процессор, после пробуждения, выполняет переход к вектору прерывания (0004h). Смотрите **Часть 14.6 "Режим снижения потребления (Спящий)"** для деталей на спящий режим и рисунок 14-10 для временных диаграмм пробуждения из спячки через прерывание RA2/INT.

**Примечание:** Регистры ANSEL и CM2CON0 должны быть инициализированы для конфигурации аналогового канала как цифрового ввода. На ножках, сконфигурированных как аналоговые входы, читается '0'.

### 14.3.2 Прерывания TMR0

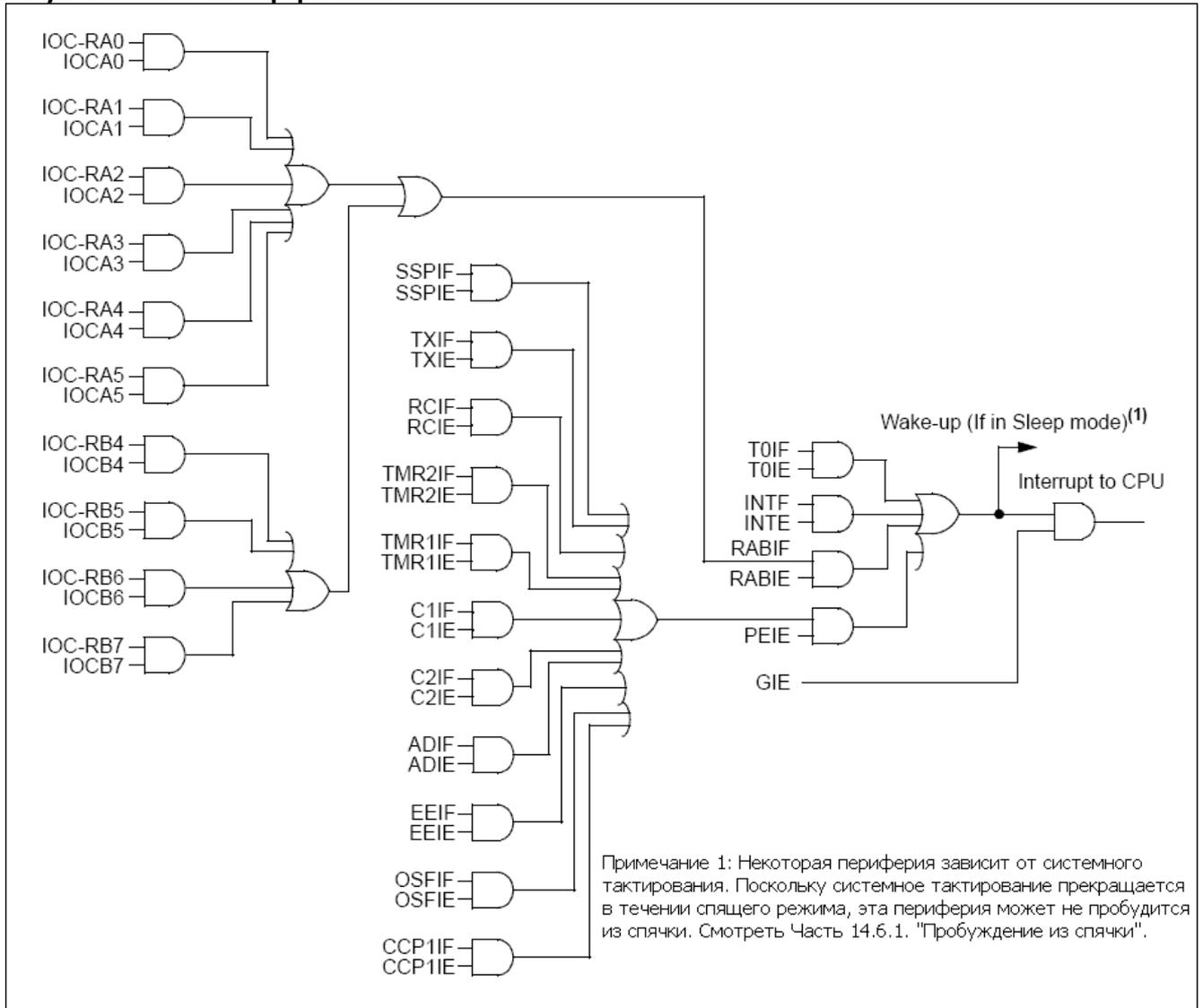
При переполнении (FFh -> 00h) регистра TMR0 устанавливается бит T0IF (INTCON<2>). Прерывание может быть разрешено/запрещено установкой/очисткой бита T0IE (INTCON<5>). Смотреть **Часть 5.0 "Модуль Timer0"** для большей информации о модуле Timer0.

### 14.3.3 Прерывание PORTA/PORTB

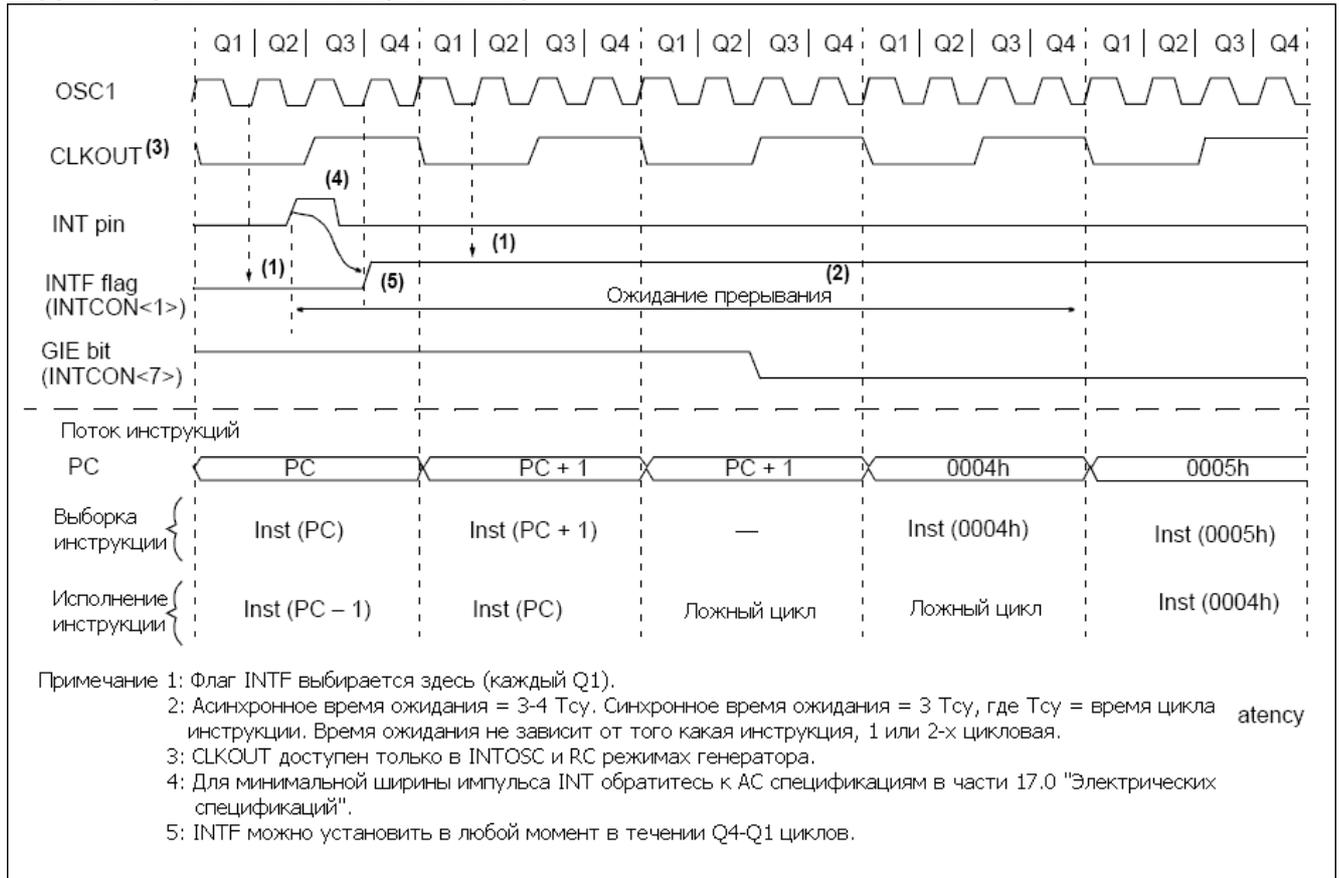
Изменения на вводах PORTA или PORTB изменяют установки бита RABIF (INTCON<0>). Прерывание может быть разрешено/запрещено установкой/очисткой бита RABIE (INTCON<3>). Плюс, индивидуальные ножки могут быть конфигурированы через регистры IOCA или IOCB.

**Примечание:** Если на ножке I/O проводит событие в момент операции чтения (начало цикла Q2), флаг прерывания RABIF может остаться неустановленным. Смотреть **Часть 4.2.3 "Прерывание на изменение"** для большей информации.

**Рисунок 14-7: Логика прерывания**



**FIGURE 14-8: INT PIN INTERRUPT TIMING**



**TABLE 14-6: Перечень регистров прерывания**

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Значение на POR, BOR	Значение на другие сбросы
INTCON	GIE	PEIE	TOIE	INTE	RABIE	TOIF	INTF	RABIF	0000 000x	0000 000x
PIE1	-	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
PIE2	OSFIE	C2IE	C1IE	EEIE	-	-	-	-	0000 ----	0000 ----
PIR1	-	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
PIR2	OSFIF	C2IF	C1IF	EEIF	-	-	-	-	0000 ----	0000 ----

**Пояснения:** x = неизвестный, u = неизменный, — = не задействованный, читается как '0', q = значение зависит от состояния. Затемнённые ячейки не используются в модуле прерывания.

## 14.4 Context Saving During Interrupts

В течении прерывания в стек записывается только значение возврата PC. Обычно пользователи стремятся сохранить ключевые регистры в течении прерывания (такие как W и STATUS регистры). Это можно сделать программно. Поскольку верхние 16 байт всех GPR банков общие в PIC16F631/677/685/687/689/690 (смотреть рисунок 2-2 и 2-3), регистры временного хранения W\_TEMP и STATUS\_TEMP надо располагать там. Эти 16 позиций не требуют манипуляций с банками и следовательно в них проще сохранять, а затем восстанавливать контекст.

Код, показанный в примере 14-1, может быть использован для:

- Хранения регистра W
- Хранения регистра STATUS
- Выполнения ISR кода
- Восстановления регистра статуса (и регистра бита выбора банков)
- Восстановления регистра W

**Примечание:** PIC16F631/677/685/687/689/690 обычно не требуют сохранения PCLATH. Тем не менее, если в вычислениях используются переходы GOTO в ISR и главном коде, PCLATH надо хранить и восстанавливать в ISR.

### Пример 14-1: Сохранение регистров STATUS и W в RAM

```
MOVWF W_TEMP           ;Копировать W в регистр TEMP
SWAPF STATUS,W         ;Обменять статус, который должен быть сохранён, в W
CLRF STATUS            ;банк 0, независимо от текущего банка, очистить IRP,RP1,RP0
MOVWF STATUS_TEMP     ;Сохранить статус в нулевой регистр STATUS_TEMP
:
:(ISR) ;Вставить здесь коде пользователя
:
SWAPF STATUS_TEMP,W   ;Обменять регистр STATUS_TEMP в W
                       ;(установить банк для оригинального состояния)
MOVWF STATUS          ;Переместить W в регистр STATUS
SWAPF W_TEMP,F        ;Обменять W_TEMP
SWAPF W_TEMP,W        ;Обменять W_TEMP в W
```

## 14.5 Сторожевой таймер (WDT)

WDT имеет следующие характеристики:

- Работает от LFINTOSC (31 kHz)
- Содержит 16-битный предделитель
- Общий с Timer0 8-битный предделитель
- Период счёта от 1 ms до 268 секунд
- Бит конфигурации и возможность программного контроля

WDT обнуляется по определённым условиям описанным в таблице 14-7.

### 14.5.1 WDT генератор

WDT получает базовое время от 31 kHz LFINTOSC. Бит LTS регистра OSCCON не отражает, что LFINTOSC разрешён. Значение WDTCON есть '---0 1000' на все сбросы. Это даёт номинальную временную базу 17ms.

**Примечание:** Когда таймер запуска генератора (OST) вызван, WDT удерживается в состоянии сброса, потому что WDT волновой счётчик используется у OST для выполнения счёта задержки генератора. Когда OST счётчик переполнен, WDT может начать счёт (если разрешён).

### 14.5.2 Управление WDT

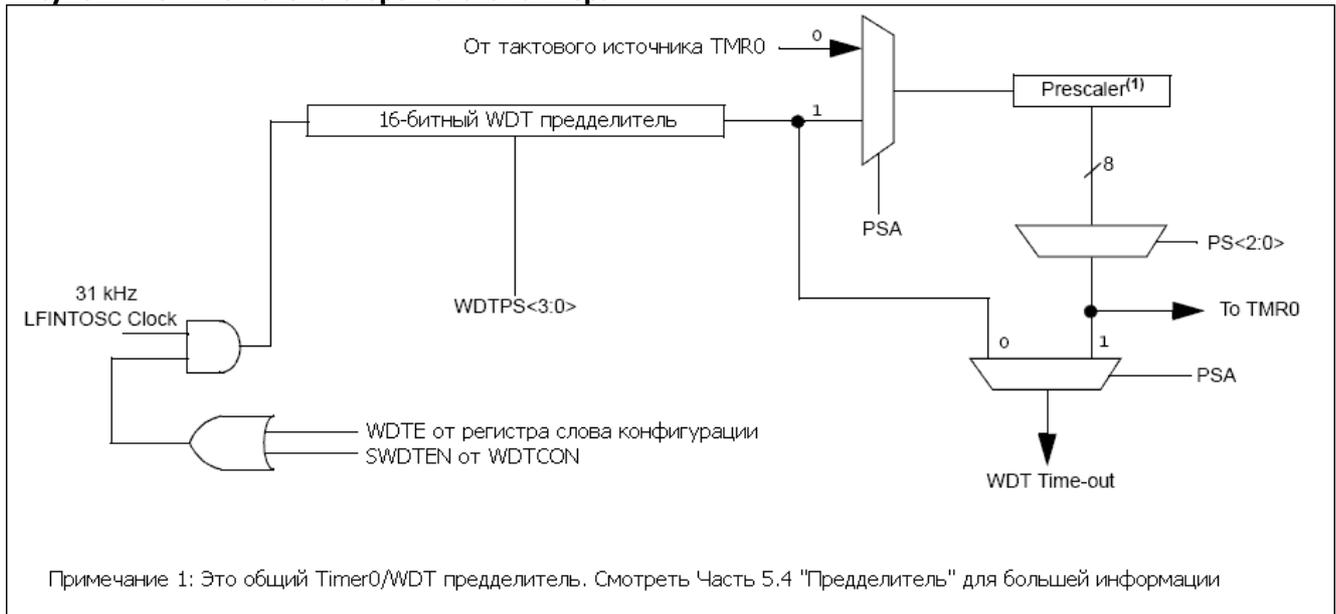
Бит WDTE расположен в регистре слова конфигурации. Когда он установлен, WDT работает непрерывно.

Когда бит WDTE в регистре слова конфигурации установлен, бит SWDTEN регистра WDTCON не оказывает эффекта. Если WDTE очищен, бит SWDTEN может быть использован для включения и выключения WDT. Установка бита включает и очистка выключает его.

Биты PSA и PS<2:0> регистра OPTION имеет эту же функцию в предыдущих версиях семейства микроконтроллеров PIC16F631/677/685/687/689/690.

Смотреть **Часть 5.0 "Модуль Timer0"** для большей информации.

**Рисунок 14-9: Блок схема сторожевого таймера**



**Таблица 14-7: Состояние WDT**

Условие	WDT
WDTE = 0	Очищен
Команда CLRWDT	
Обнаружена авария генератора	
Выход из спячки + системное тактирование = T1OSC, EXTRC, INTOSC, EXTCLK	
Выход из спячки + системное тактирование = XT, HS, LP	Очищен после окончания OST

**Регистр 14-2: WDTCON: Регистр управления сторожевым таймером**

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN <sup>(1)</sup>
Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0

<b>Описание:</b>			
R=бит читается	W=бит записывается	U=не задействованный бит, читается как '0'	
-n=значение при POR	'1'=бит установлен	'0'=бит очищен	X=бит не определён

**Биты 7-5** Не используются: Читаются как '0'

**Биты 4-1** **WDTPS<3:0>**: Биты выбора периода сторожевого таймера

Значение бит = Показатель предделителя

0000 = 1:32

0001 = 1:64

0010 = 1:128

0011 = 1:256

0100 = 1:512 (Значение после сброса)

0101 = 1:1024

0110 = 1:2048

0111 = 1:4096

1000 = 1:8192

1001 = 1:16384

1010 = 1:32768

1011 = 1:65536

1100 = резервировано

1101 = резервировано

1110 = резервировано

1111 = резервировано

**Бит 0** **SWDTEN**: Бит программного включения или выключения сторожевого таймера(1)

1 = WDT включен

0 = WDT выключен (значение после сброса)

**Примечание 1:** Если бит конфигурации WDTE = 1, WDT всегда разрешён, независимо от этого бита управления. Если бит конфигурации WDTE = 0, то можно включать/выключать WDT этим битом управления.

**Таблица 14-8: Перечень регистров сторожевого таймера**

Имя	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Значение на POR, BOR	Значение на другие сбросы
CONFIG <sup>(1)</sup>	/CPD	/CP	MCLRE	/PWRTS	WDTE	FOSC2	FOSC1	FOSC0	-	-
OPTION_REG	/RABPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
WDTCON	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 1000	---0 1000

**Пояснения:** Затемнённые клетки не используются сторожевым таймером.

**Примечание 1:** Смотри регистр 14-1 для пояснения работы всех битов регистра слова конфигурации.

## 14.6 Режим снижения потребления (Спящий)

В режим снижения потребления контроллер входит после выполнения инструкции SLEEP.

Если сторожевой таймер разрешён:

- WDT будет очищен, но продолжает работать.
- Бит PD в регистре STATUS будет очищен.
- Бит TO установлен.
- Драйвер генератора отключен.
- I/O порты поддерживают состояние, которое они имели перед SLEEP (высокий или низкий уровень, или высокоимпедансное состояние).

Для низкого потребления тока в этом режиме, на всех ножках I/O должны быть уровни VDD или VSS, без внешней схемы, потребляющей ток с ножки I/O, а так же должны быть выключены компараторы и CVREF. Ножки I/O, которые находятся в высокоимпедансном состоянии, должны внешне быть подтянуты вниз или вверх, избегая переключения токов вызванных плавающими входами. На ввод T0CKI так же должен быть уровень VDD или VSS для минимального потребления тока. Вклад от внутренних подтягивающих резисторов на PORTA должен быть решающим. На ножке MCLR должен быть логически высокий уровень.

**Примечание:** Надо отметить, что сброс, генерируемый переполнением WDT, не тянет ножку MCLR вниз.

### 14.6.1 Пробуждение из спящего режима

Одно из следующих событий может пробудить устройство из спящего режима:

1. Внешний сброс на ножке MCLR.
2. Пробуждение от сторожевого таймера (если WDT разрешён).
3. Прерывание от ножки RA2/INT, изменение PORTA или периферийное прерывание.

Первый случай вызывает сброс устройства. Два последних события рассматриваются как продолжение выполнения программы. Биты TO и PD в регистре STATUS могут быть использованы для определения причины сброса устройства. Бит PD, который устанавливается при включении питания, сбрасывается, когда вызывается спящий режим. Бит TO очищен, если произошло пробуждение WDT.

Следующие периферийные прерывания могут пробудить устройство из спячки:

1. Прерывание от TMR1. Timer1 может работать, как асинхронный счётчик.
2. Прерывание режима захвата ECCP.
3. АЦП преобразование (когда FRC является тактовым источником АЦП).
4. Завершена операция записи EEPROM.
5. Изменение состояния на выходе компаратора.
6. Прерывание на изменение.
7. Внешнее прерывание от ножки INT.
8. Обнаружена остановка EUSART, I2C подчинённый.

Другая периферия не может генерировать прерывания, поскольку в течении спящего нет внутреннего тактирования. После выполнения инструкции SLEEP выбирается следующая (PC + 1) инструкция. Для пробуждения устройства по прерыванию, соответствующий бит разрешения прерывания должен быть установлен (разрешён). Пробуждение происходит независимо от состояния бита GIE. Если бит GIE очищен (запрещены прерывания), устройство продолжит выполнение инструкции после инструкции SLEEP. Если бит был установлен (разрешён), устройство выполняет инструкцию после инструкции SLEEP, затем идёт по адресу прерывания (0004h). В случае, если выполнение команды после SLEEP не желательно, пользователь должен поставить NOP после инструкции SLEEP.

**Примечание:** Если прерывания запрещены (GIE очищен), но любые источники прерывания имеющий бит разрешения прерывания и соответствующий установленный бит флага прерывания, немедленно пробудит устройство из спящего режима. Инструкция SLEEP будет полностью выполнена.

WDT будет очищен, когда устройство пробудится из спящего режима, независимо от источника пробуждения.

### 14.6.2 Пробуждение использующее прерывания

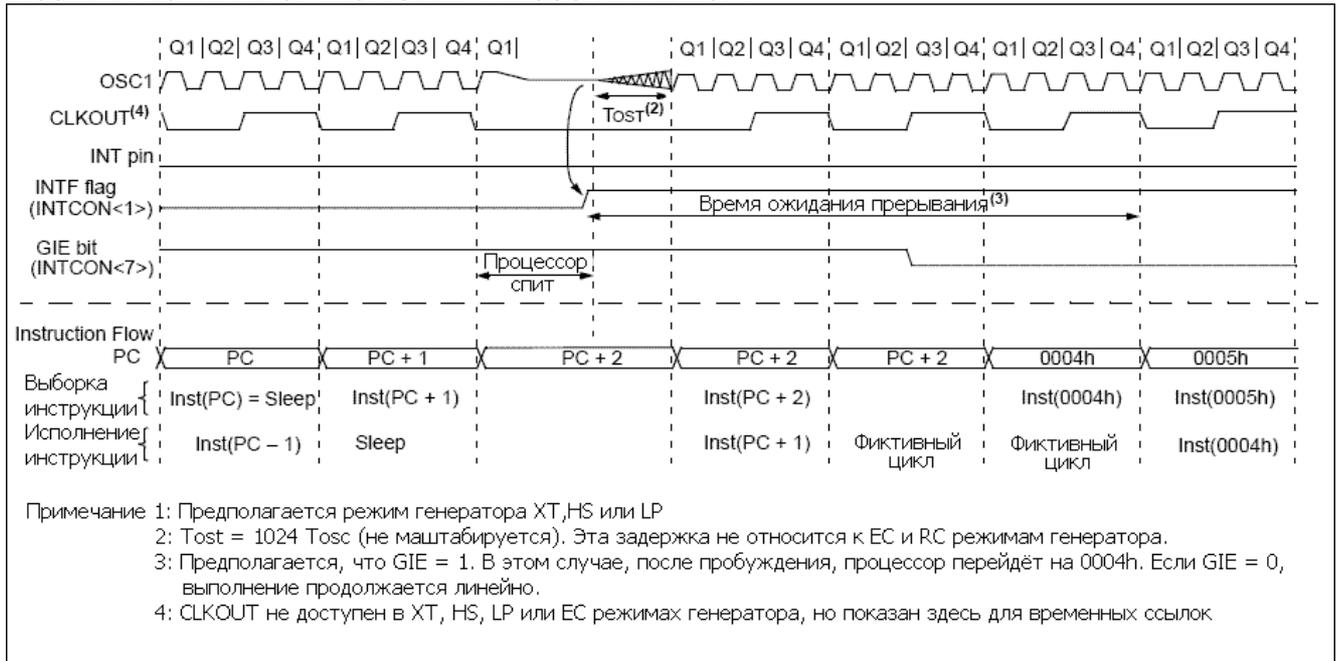
Когда прерывания запрещены (GIE очищен) и любой источник прерывания имеет установленные биты разрешения прерывания и флага прерывания, произойдёт одно из следующего:

- Если прерывание произошло перед выполнением инструкции SLEEP, инструкция SLEEP будет завершена как NOP. Следовательно, WDT и WDT предделитель и постделитель (если разрешены) не будут очищены, бит TO не будет установлен и бит PD не будет очищен.
- Если прерывание произошло в течении или после выполнения инструкции SLEEP, устройство немедленно пробудится из спящего режима. Инструкция SLEEP будет полностью выполнена перед пробуждением. Следовательно, WDT WDT предделитель и постделитель (если разрешены) будут очищены, бит TO будет установлен и бит PD будет очищен.

Даже если биты флажка были проверены перед выполнением команды SLEEP, это может быть возможно для битов флажка, чтобы стать установленным прежде, чем команда SLEEP выполнена. Чтобы определить была ли выполнена команда SLEEP, надо проверить бит PD. Если бит PD установлен, команда SLEEP была выполнена как NOP.

Чтобы гарантировать очистку WDT, необходимо команду CLRWDT выполнить перед командой SLEEP.

**FIGURE 14-10: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



## 14.7 Защита кода

Если бит(ы) защиты кода не запрограммированы, программная память контроллера может быть прочитана, используя ICSP™.

**Примечание:** Данные EEPROM и Flash памяти программ будут стёрты, когда защита кода будет отключена. Смотреть "PIC12F6XX/16F6XX Спецификация программирования памяти" (DS41204) для большей информации.

## 14.8 Расположения ID

Четыре ячейки памяти (2000h-2003h) определены для ID расположений, где пользователь может запомнить контрольную сумму или другие числа идентификации кода. Эти расположения не доступны в течении нормальной работы, но могут читаться и писаться в течении режима записи/проверки. Только наименее значимые 7 бит ID расположений использованы.

## 14.9 Внутрисхемное последовательное программирование

Микроконтроллеры PIC16F631/677/685/687/689/690 могут быть последовательно запрограммированы в схеме конечного приложения. Для это достаточно две линии для тактирования и данных, а так же три другие линии для:

- питания
- земли
- программирующего напряжения

Это позволяет заказчику производить платы с не запрограммированными устройствами и программировать их перед пересылкой продукции. Это позволяет так же производить загрузку последнего программного обеспечения на рабочем оборудовании. Для установки устройства в режим программирования/проверки необходимо удерживать на ножках RA0/AN0/C1IN+/ICSPDAT/ULPWU и RA1/AN1/C12IN-/VREF/ICSPCLK низкий уровень, пока нарастает напряжение ножки MCLR (VPP) от VIL до VIH. Смотреть "PIC12F6XX/16F6XX Спецификация программирования памяти" (DS41204) для большей информации. RA0 становится данными программирования и RA1 становится тактированием программирования. Оба, RA0 и RA1, являются вводом с триггером Шмидта в этом режиме.

После сброса устройства, в режиме программирования/проверки, программный счётчик (PC) устанавливается на 00h. 6-битные команды затем поступают в устройство. Зависящие от этих команд, 14 битные программные данные затем загружаются в устройство или читаются из него. Для полной информации о последовательном программировании, пожалуйста обращайтесь к "PIC12F6XX/16F6XX Спецификации программирования памяти" (DS41204).

Типовая схема подключения для последовательного внутрисхемного программирования показана на рисунке 14-11.

**Рисунок 14-11: Типовое подключение для внутрисхемного последовательного программирования**

