

15 Таймеры общего назначения (от TIM2 до TIM5)

Устройства с низкой плотностью это микроконтроллеры STM32F101xx, STM32F102xx и STM32F103xx с плотностью флэш-памяти от 16 до 32 Кбайт.

Устройства средней плотности это микроконтроллеры STM32F101xx, STM32F102xx и STM32F103xx с плотностью флэш-памяти от 64 до 128 Кбайт.

Устройства высокой плотности это микроконтроллеры STM32F101xx и STM32F103xx с плотностью флэш-памяти от 256 до 512 Кбайт.

Устройства XL-плотности это микроконтроллеры STM32F101xx и STM32F103xx, в которых плотность флэш-памяти колеблется от 768 Кбайт до 1 Мбайт.

Устройства линии связи это микроконтроллеры STM32F105xx и STM32F107xx.

Этот раздел относится ко всему семейству STM32F10xxx, если не указано иное.

15.1 Введение от TIM2 до TIM5

Таймеры общего назначения состоят из 16-разрядного счетчика с автоматической перезагрузкой, управляемого программируемым делителем.

Их можно использовать для различных целей, в том числе для измерения длительности импульсов входных сигналов (*входной захват*) или генерация выходных сигналов (*сравнение выхода и ШИМ*).

Длины импульсов и периоды сигналов можно модулировать от нескольких микросекунд до нескольких миллисекунд с помощью предварительного делителя таймера и предварительного делителя контроллера тактовых импульсов RCC.

Таймеры полностью независимы и не используют общие ресурсы. Их можно синхронизировать вместе, как описано в [Раздел 15.3.15](#).

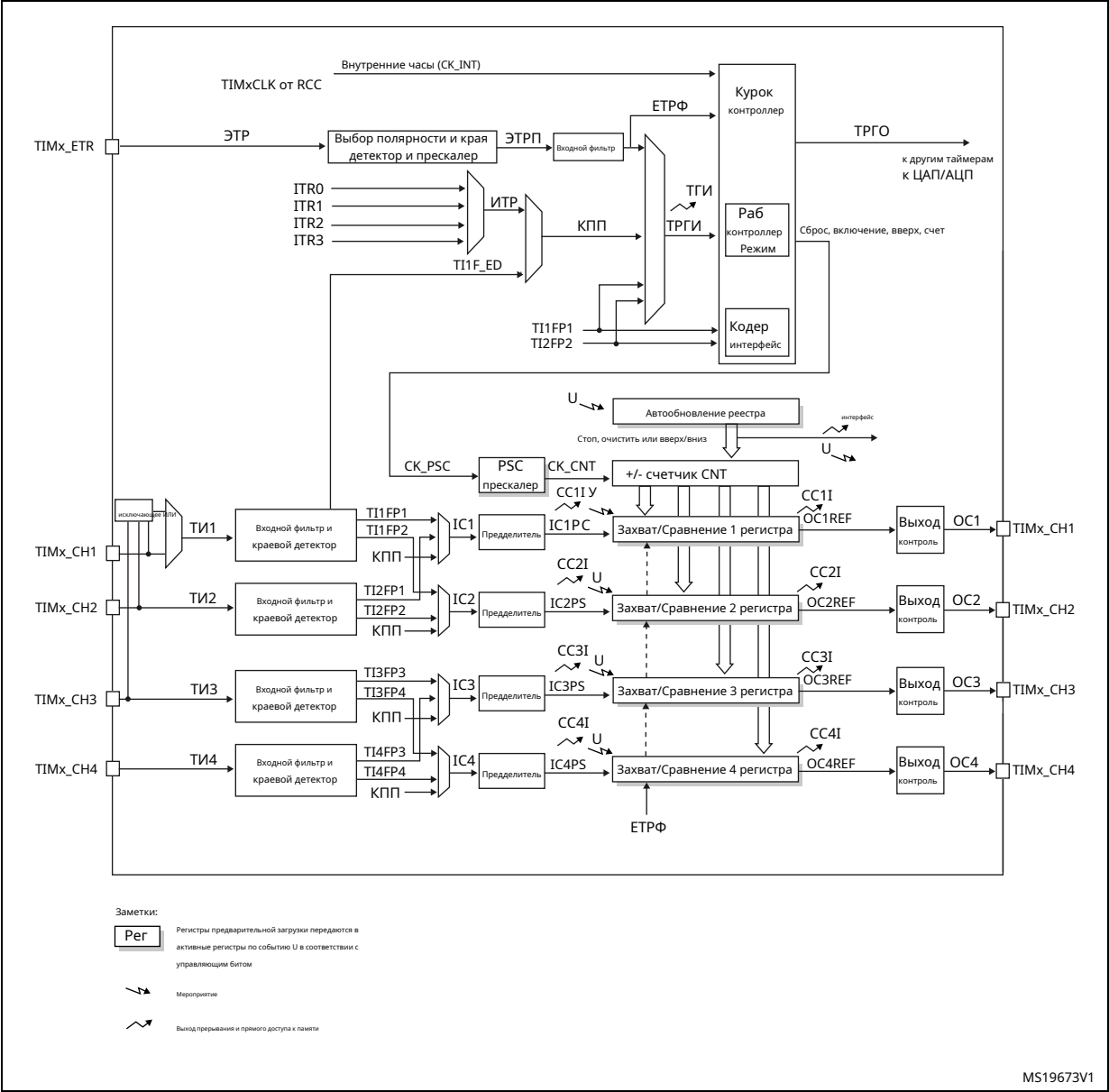
15.2

Основные возможности TIMx

Общие функции таймера TIMx включают в себя:

- 16-битный счетчик автоматической перезагрузки вверх, вниз, вверх/вниз.
- 16-битный программируемый прескалер, используемый для деления (также «на лету») тактовой частоты счетчика на любой коэффициент от 1 до 65536.
- До 4 независимых каналов для:
 - Входной захват
 - Сравнение выходов
 - Генерация ШИМ (режимы с выравниванием по краю и центру)
 - Выход в одноимпульсном режиме
- Схема синхронизации для управления таймером внешними сигналами и для соединения нескольких таймеров между собой.
- Генерация прерывания/DMA по следующим событиям:
 - Обновление: переполнение/недостаточное значение счетчика, инициализация счетчика (программным обеспечением или внутренним/внешним триггером)
 - Триггерное событие (запуск счетчика, остановка, инициализация или подсчет по внутреннему/внешнему триггеру)
 - Входной захват
 - Сравнение выходов
- Поддерживает инкрементный (квадратурный) энкодер и схему датчика Холла для целей позиционирования
- Триггерный вход для внешнего тактирования или поциклового управления током

Рисунок 100. Блок-схема таймера общего назначения



15,3 Функциональное описание TIMx

15.3.1 Базовая единица времени

Основным блоком программируемого таймера является 16-разрядный счетчик с соответствующим ему регистром автоперезагрузки. Счетчик может считать вверх, вниз или одновременно вверх и вниз. Часы счетчика могут быть разделены предварительным делителем.

Счетчик, регистр автоматической перезагрузки и регистр предварительного делителя могут быть записаны или прочитаны программным обеспечением. Это верно даже при работающем счетчике.

Базовая единица времени включает в себя:

- Регистр счетчика (TIMx_CNT) Регистр
- предделителя (TIMx_PSC): Регистр
- автоматической перезагрузки (TIMx_ARR)

Регистр автоматической перезагрузки предварительно загружен. Запись или чтение из регистра автоматической перезагрузки обеспечивает доступ к регистру предварительной загрузки. Содержимое регистра предварительной загрузки передается в теневой регистр постоянно или при каждом событии обновления (UEV), в зависимости от бита разрешения предварительной загрузки автоматической перезагрузки (ARPE) в регистре TIMx_CR1. Событие обновления отправляется, когда счетчик достигает переполнения (или потери значимости при обратном счете) и если бит UDIS равен 0 в регистре TIMx_CR1. Он также может быть сгенерирован программным обеспечением. Генерация события обновления подробно описана для каждой конфигурации.

Счетчик тактируется выходом предделителя CK_CNT, который активируется только тогда, когда бит разрешения счетчика (CEN) в регистре TIMx_CR1 установлен (см. также описание контроллера ведомого режима, чтобы получить более подробную информацию о включении счетчика).

Обратите внимание, что фактический сигнал включения счетчика CNT_EN устанавливается через 1 такт после CEN.

Описание предделителя

Предделитель может делить тактовую частоту счетчика на любой коэффициент от 1 до 65536. Он основан на 16-битном счетчике, управляемом через 16-битный регистр (в регистре TIMx_PSC). Его можно изменить на лету, так как этот регистр управления буферизован. Новое отношение предделителя учитывается при следующем событии обновления.

[Рисунок 101а](#) также [Рисунок 102](#) приведите несколько примеров поведения счетчика при изменении коэффициента предделителя на лету:

Рисунок 101. Временная диаграмма счетчика с изменением деления прескалера с 1 на 2

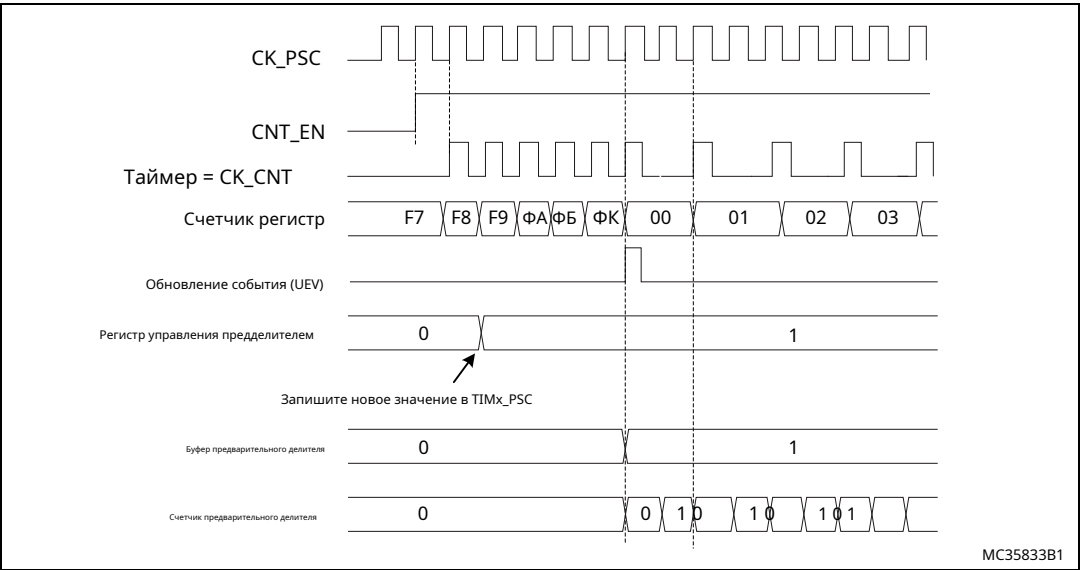
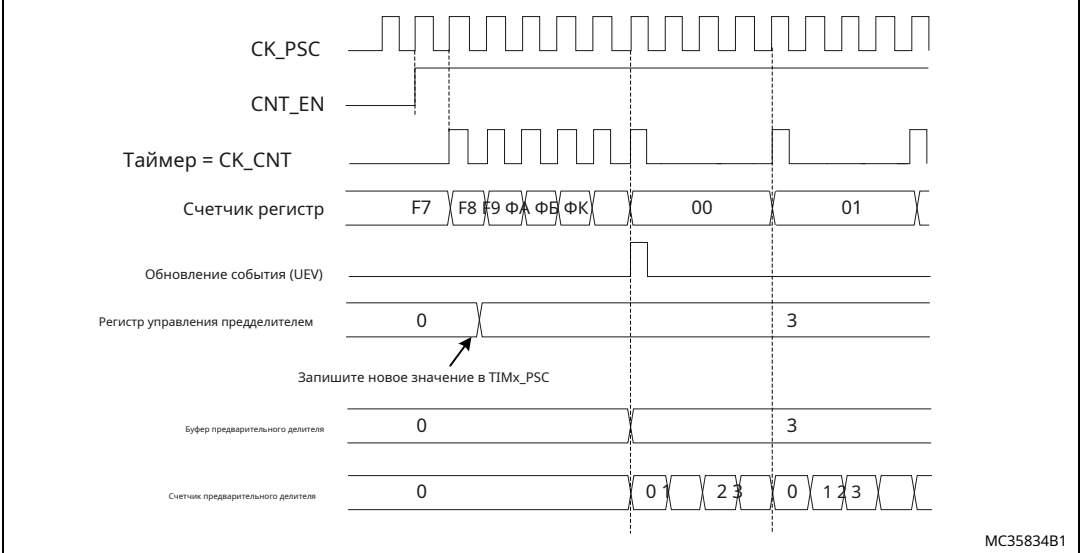


Рисунок 102. Временная диаграмма счетчика с изменением деления прескалера с 1 на 4



15.3.2 Режимы счетчика

Режим прямого счета

В режиме обратного счета счетчик считает от 0 до значения автоперезагрузки (содержимое регистра TIMx_ARR), затем перезапускается с 0 и генерирует событие переполнения счетчика.

Событие Update может генерироваться при каждом переполнении счетчика или установкой бита UG в регистре TIMx_EGR (с помощью программного обеспечения или с помощью контроллера ведомого режима).

Событие UEV можно отключить программно, установив бит UDIS в регистре TIMx_CR1. Это делается для того, чтобы избежать обновления теневых регистров при записи новых значений в регистры предварительной загрузки. Затем событие обновления не происходит до тех пор, пока бит UDIS не будет записан в 0. Однако счетчик перезапускается с 0, как и счетчик предварительного делителя (но скорость предварительного масштабирования)

не меняется). Кроме того, если установлен бит URS (выбор запроса на обновление) в регистре TIMx_CR1, установка бита UG генерирует событие обновления UEV, но без установки флага UIF (таким образом, не отправляется запрос прерывания или DMA). Это делается для того, чтобы избежать генерации прерываний обновления и захвата при очистке счетчика события захвата.

Когда происходит событие обновления, обновляются все регистры и устанавливается флаг обновления (бит UIF в регистре TIMx_SR) (в зависимости от бита URS):

- Буфер прескалера перезагружается значением предварительной загрузки (содержимое регистра TIMx_PSC)
- Теневой регистр автоматической перезагрузки обновляется значением предварительной загрузки (TIMx_ARR).

На следующих рисунках показаны некоторые примеры поведения счетчика для разных тактовых частот, когда TIMx_ARR=0x36.

Рисунок 103. Временная диаграмма счетчика, внутренние часы, разделенные на 1

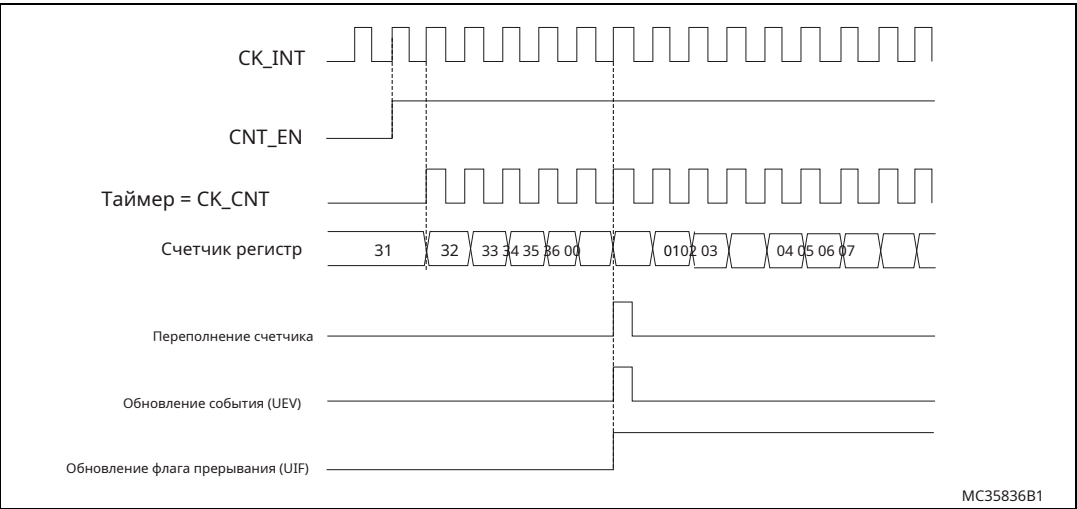


Рисунок 104. Временная диаграмма счетчика, внутренние часы, разделенные на 2

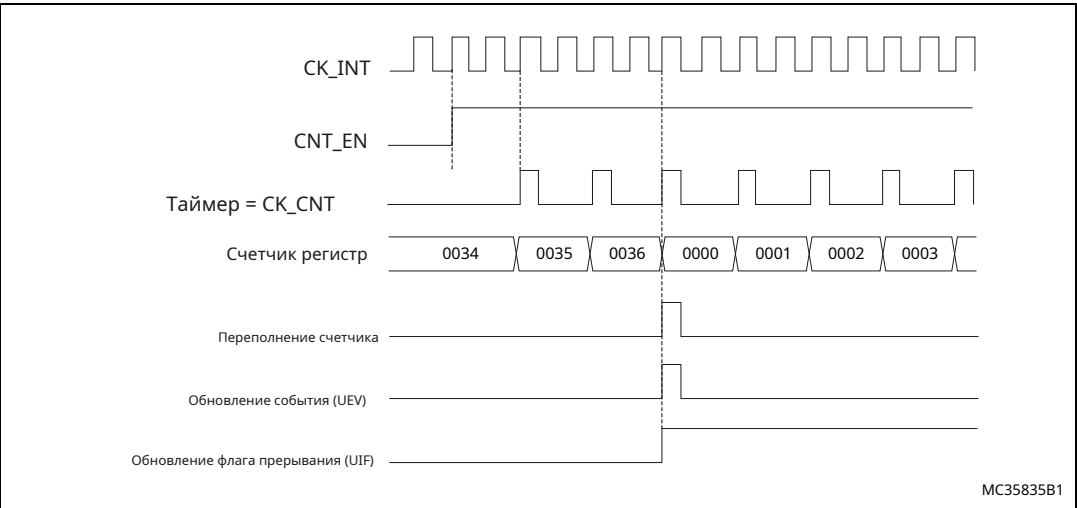


Рисунок 105. Временная диаграмма счетчика, внутренние часы, разделенные на 4

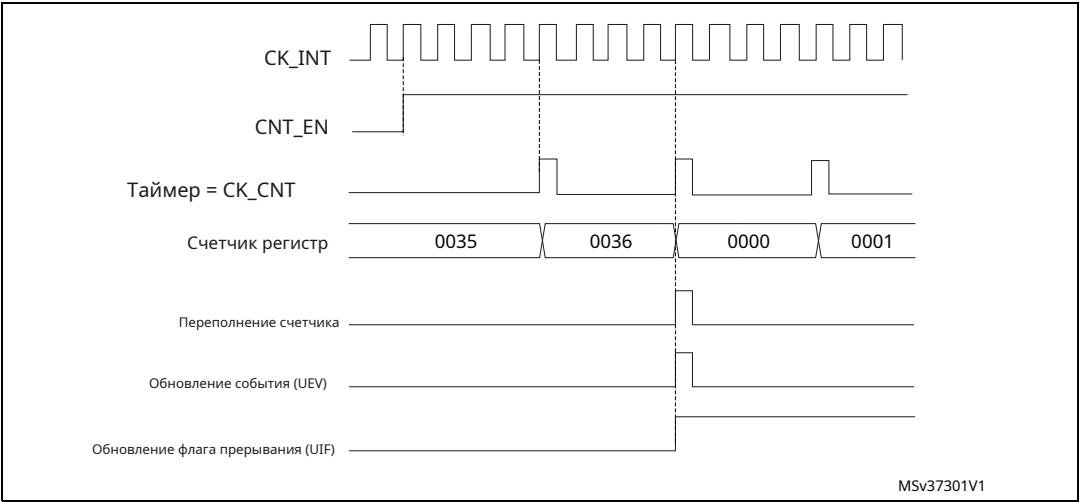


Рисунок 106. Временная диаграмма счетчика, внутренние часы, разделенные на N

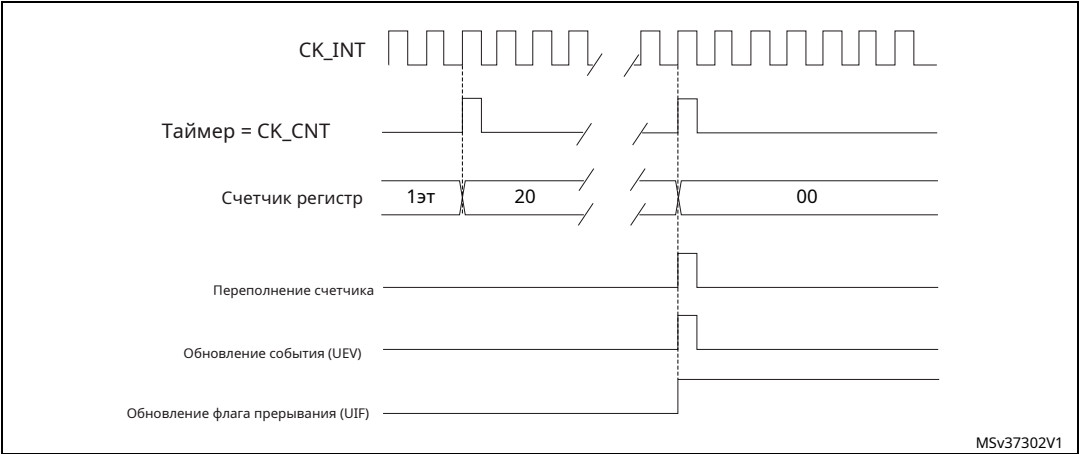


Рисунок 107. Временная диаграмма счетчика, событие обновления, когда ARPE=0 (TIMx_ARR не

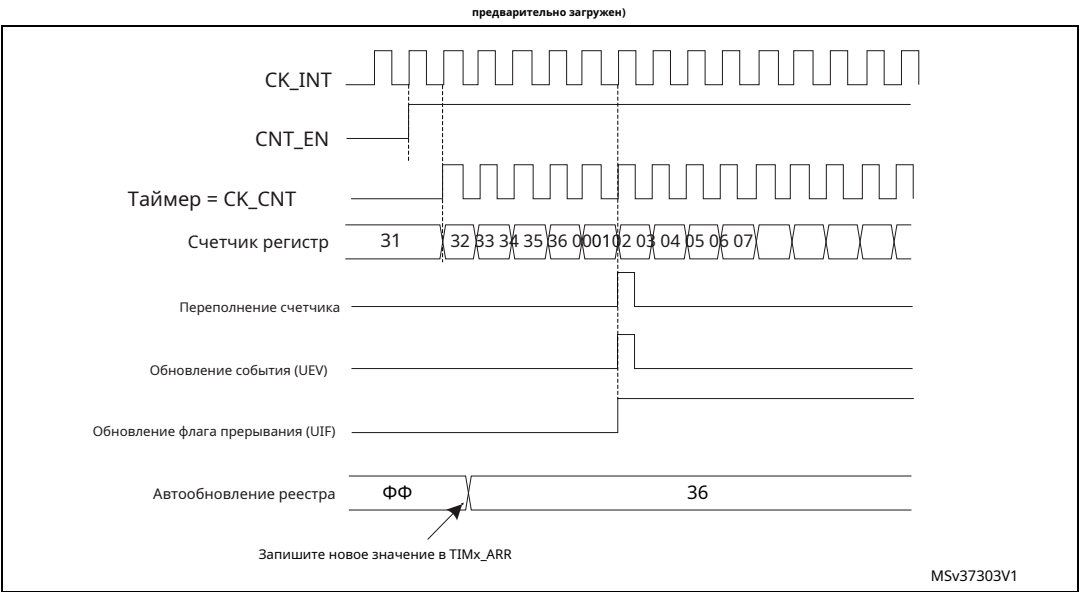
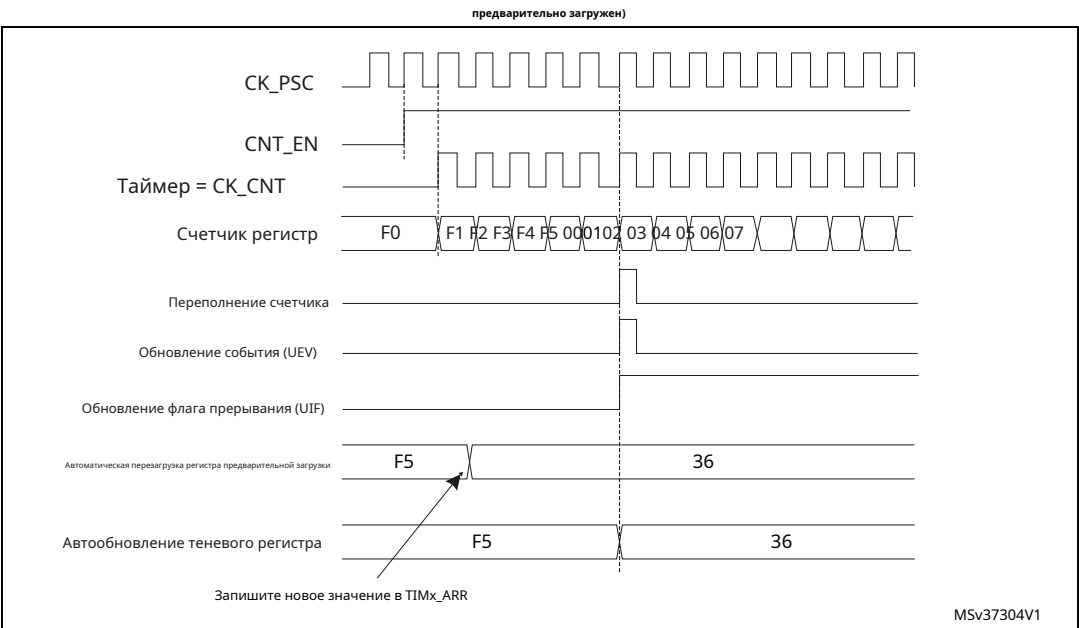


Рисунок 108. Временная диаграмма счетчика, событие обновления при ARPE=1 (TIMx_ARR



Режим обратного отсчета

В режиме обратного отсчета счетчик ведет отсчет от значения автоматической перезагрузки (содержимого регистра TIMx_ARR) до 0, затем перезапускается со значения автоматической перезагрузки и генерирует событие опустошения счетчика.

Событие обновления может генерироваться при каждом опустошении счетчика или путем установки бита UG в регистре TIMx_EGR (с помощью программного обеспечения или с помощью контроллера ведомого режима).

Событие обновления UEV можно отключить программно, установив бит UDIS в регистре TIMx_CR1. Это делается для того, чтобы избежать обновления теневых регистров при записи новых значений в



предварительная загрузка регистров. Затем событие обновления не происходит до тех пор, пока бит UDIS не будет записан в 0. Однако счетчик перезапускается с текущего значения автоматической перезагрузки, тогда как счетчик предварительного делителя перезапускается с 0 (но скорость предварительного масштабирования не изменяется).

Кроме того, если установлен бит URS (выбор запроса на обновление) в регистре TIMx_CR1, установка бита UG генерирует событие обновления UEV, но без установки флага UIF (таким образом, запрос прерывания или DMA не отправляется). Это делается для того, чтобы избежать генерации прерываний обновления и захвата при очистке счетчика события захвата.

Когда происходит событие обновления, обновляются все регистры и устанавливается флаг обновления (бит UIF в регистре TIMx_SR) (в зависимости от бита URS):

- Буфер предварительного делителя перезагружается значением предварительной загрузки (содержимое регистра TIMx_PSC).
- Активный регистр автоматической перезагрузки обновляется значением предварительной загрузки (содержимое регистра TIMx_ARR). Обратите внимание, что автоматическая перезагрузка обновляется до перезагрузки счетчика, поэтому следующий период является ожидаемым.

На следующих рисунках показаны некоторые примеры поведения счетчика для разных тактовых частот, когда TIMx_ARR=0x36.

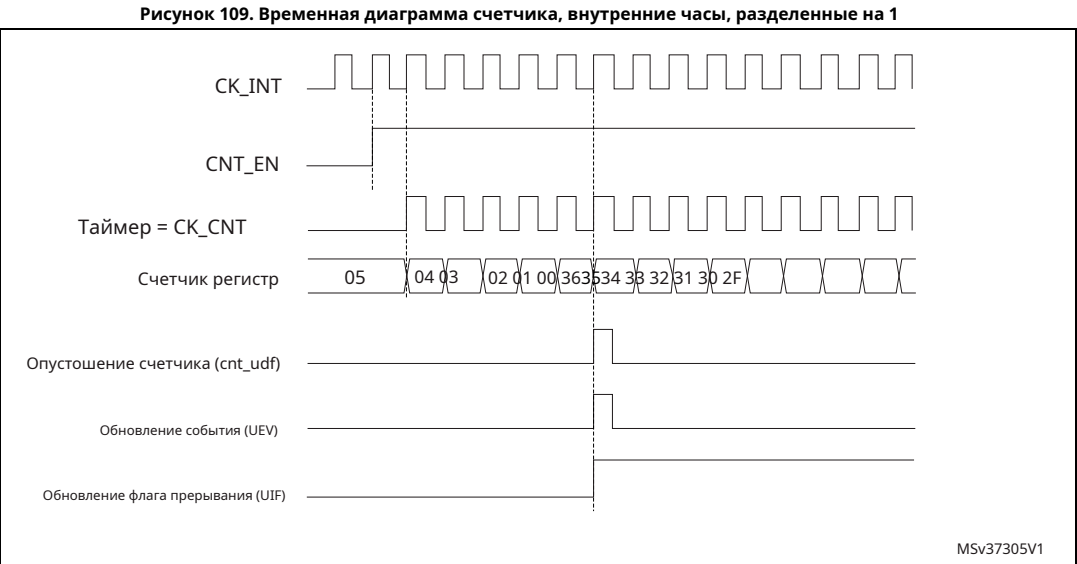


Рисунок 110. Временная диаграмма счетчика, внутренние часы, разделенные на 2

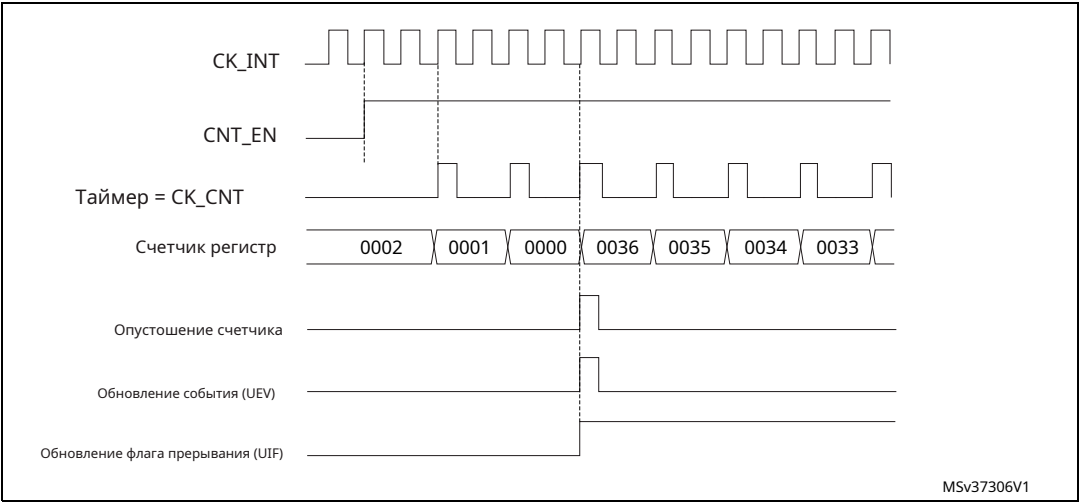


Рисунок 111. Временная диаграмма счетчика, внутренние часы, разделенные на 4

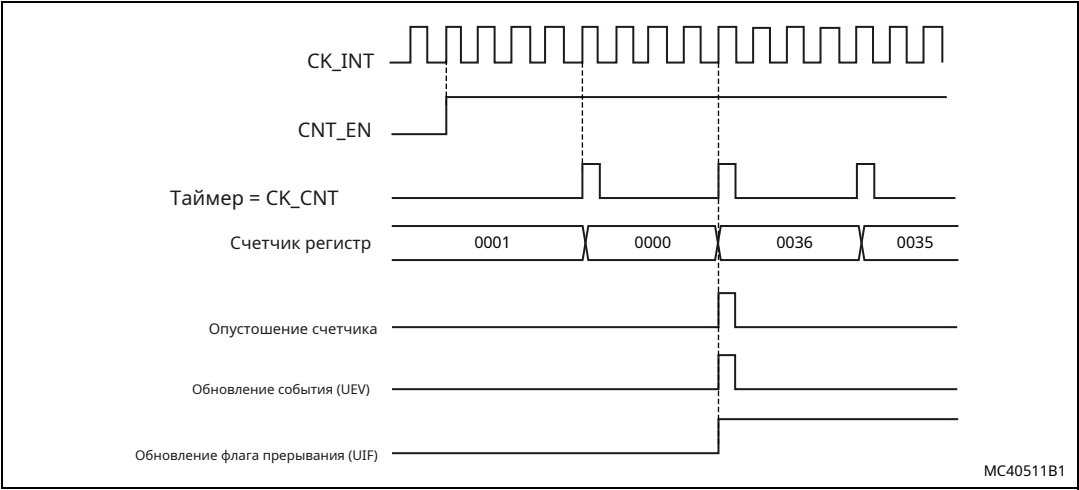


Рисунок 112. Временная диаграмма счетчика, внутренние часы, разделенные на N

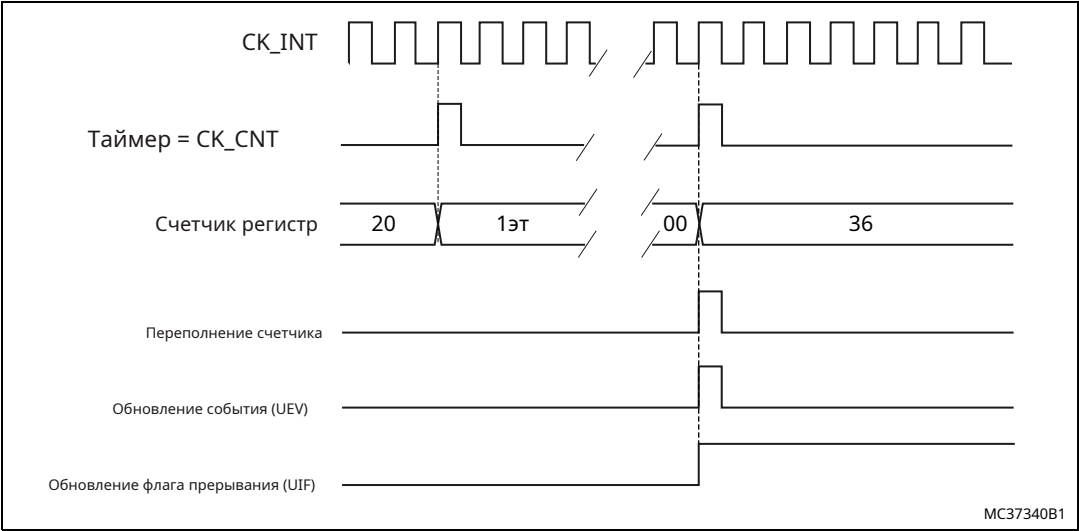
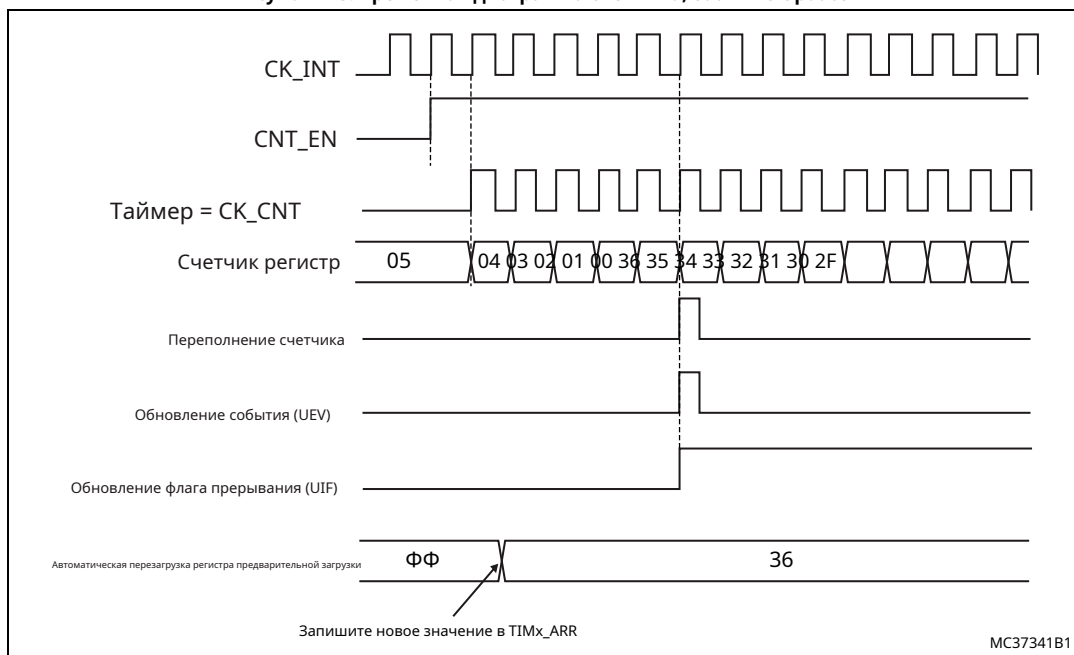


Рисунок 113. Временная диаграмма счетчика, событие Update

**Режим выравнивания по центру (счет вверх/вниз)**

В режиме с выравниванием по центру счетчик считает от 0 до значения автоперезагрузки (содержимое регистра TIMx_ARR) – 1, генерирует событие переполнения счетчика, затем ведет счет от значения автоперезагрузки до 1 и генерирует событие опустошения счетчика. Затем он перезапускает счет с 0.

Выровненный по центру режим активен, когда биты CMS в регистре TIMx_CR1 не равны '00'. Флаг прерывания сравнения вывода каналов, сконфигурированных на выходе, устанавливается, когда: счетчик ведет обратный отсчет (режим выравнивания по центру 1, CMS = "01"), счетчик ведет прямой счет (режим выравнивания по центру 2, CMS = "10") счетчик считает вверх и вниз (режим выравнивания по центру 3, CMS = "11").

В этом режиме бит направления (DIR из регистра TIMx_CR1) не может быть записан. Он обновляется аппаратно и выдает текущее направление счетчика.

Событие обновления может генерироваться при каждом переполнении счетчика и при каждом опустошении счетчика или путем установки бита UG в регистре TIMx_EGR (с помощью программного обеспечения или с помощью контроллера ведомого режима) также генерирует событие обновления. В этом случае счетчик перезапускает счет с 0, как и счетчик предделителя.

Событие обновления UEV можно отключить программно, установив бит UDIS в регистре TIMx_CR1. Это делается для того, чтобы избежать обновления теневых регистров при записи новых значений в регистры предварительной загрузки. Затем событие обновления не происходит до тех пор, пока бит UDIS не будет записан в 0. Однако счетчик продолжает считать вверх и вниз на основе текущего значения автоматической перезагрузки.

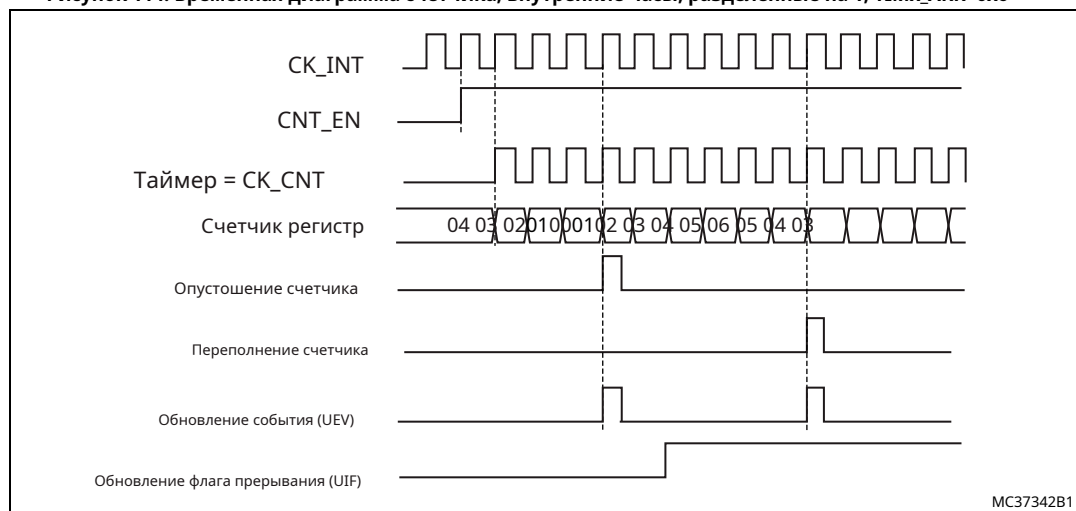
Кроме того, если установлен бит URS (выбор запроса на обновление) в регистре TIMx_CR1, установка бита UG генерирует событие обновления UEV, но без установки флага UIF (таким образом, запрос прерывания или DMA не отправляется). Это делается для того, чтобы избежать генерации прерывания обновления и захвата при очистке счетчика в событии захвата.

Когда происходит событие обновления, обновляются все регистры и устанавливается флаг обновления (бит UIF в регистре TIMx_SR) (в зависимости от бита URS):

- Буфер предварительного делителя перезагружается значением предварительной загрузки (содержимое регистра TIMx_PSC).
- Активный регистр автоматической перезагрузки обновляется значением предварительной загрузки (содержимое регистра TIMx_ARR). Обратите внимание, что если источником обновления является переполнение счетчика, автоперезагрузка обновляется до перезагрузки счетчика, так что следующий период является ожидаемым (счетчик загружается с новым значением).

На следующих рисунках показаны некоторые примеры поведения счетчика для разных тактовых частот.

Рисунок 114. Временная диаграмма счетчика, внутренние часы, разделенные на 1, TIMx_ARR=0x6



1. Здесь используется режим выравнивания по центру 1, для получения более подробной информации см. [Раздел 15.4.1: Регистр управления TIMx 1 \(TIMx_CR1\)](#).

Рисунок 115. Временная диаграмма счетчика, внутренние часы, разделенные на 2

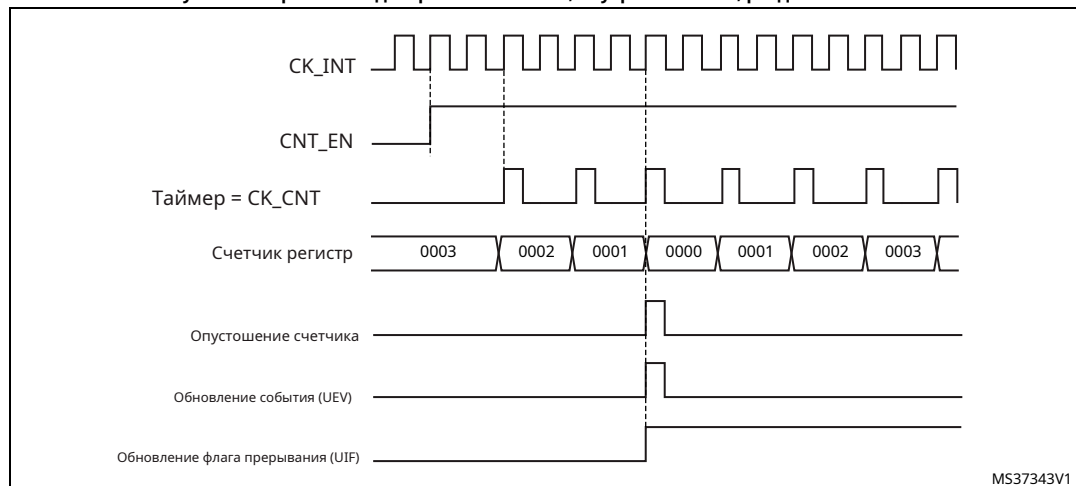
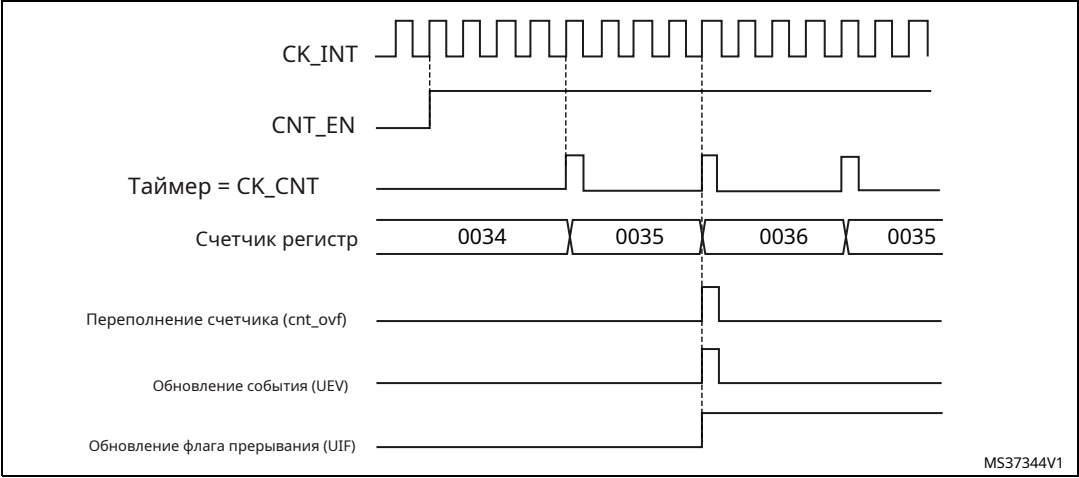


Рисунок 116. Временная диаграмма счетчика, внутренние часы, разделенные на 4, TIMx_ARR=0x36



1. Режим выравнивания по центру 2 или 3 используется с UIF при переполнении.

Рисунок 117. Временная диаграмма счетчика, внутренние часы, разделенные на N

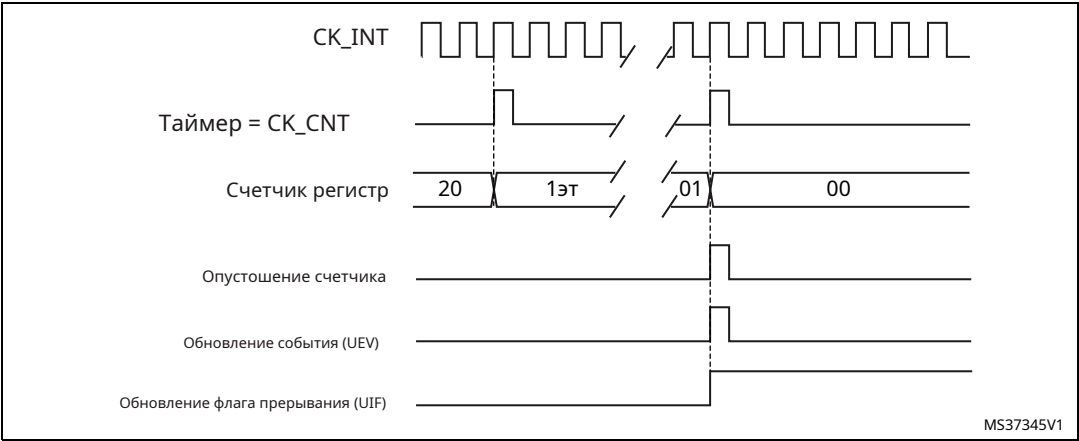


Рисунок 118. Временная диаграмма счетчика, событие обновления с ARPE=1 (опустошение счетчика)

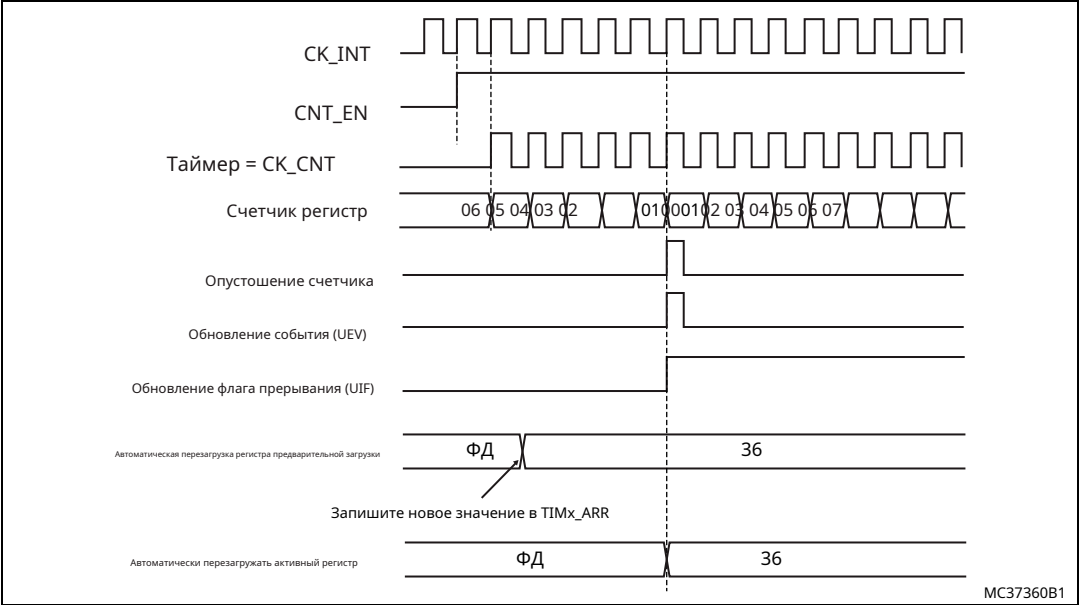
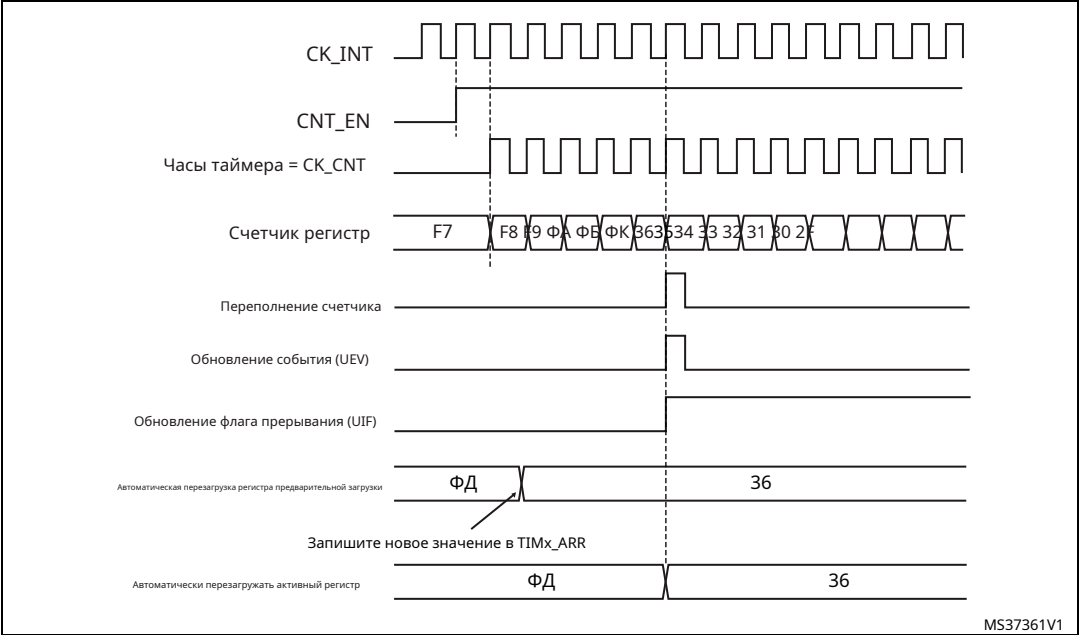


Рисунок 119. Временная диаграмма счетчика, событие обновления с ARPE=1 (переполнение счетчика)



15.3.3 Выбор часов

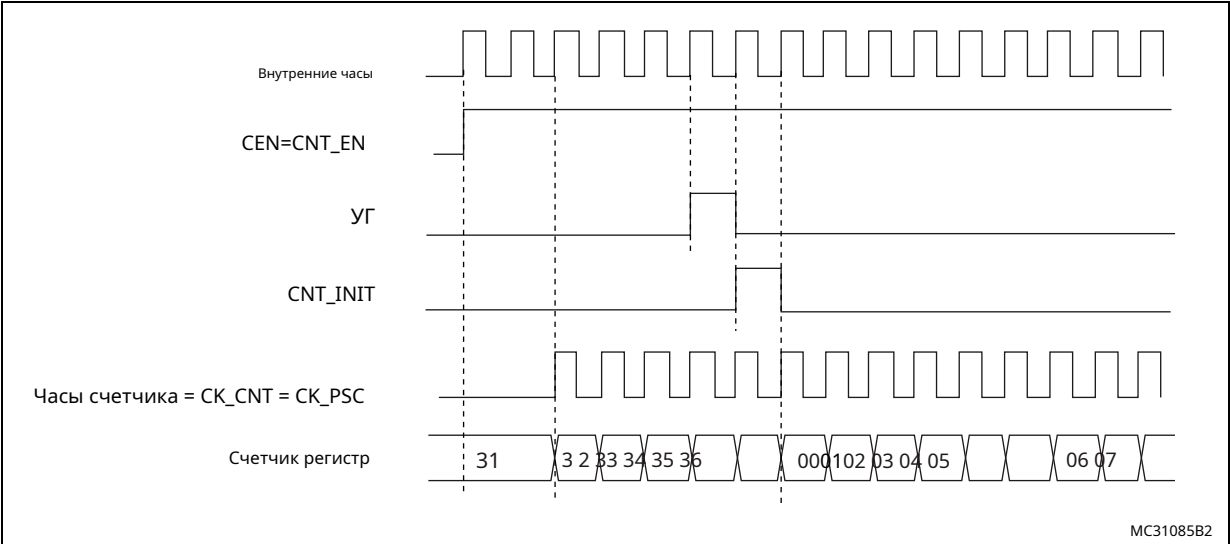
- Часы счетчика могут быть обеспечены следующими источниками часов:
- Внутренние часы (CK_INT)
 - Режим внешней синхронизации 1: внешний входной контакт (TIx) Режим
 - внешней синхронизации 2: вход внешнего триггера (ETR).
 - Внутренние триггерные входы (ITRx): использование одного таймера в качестве предварительного делителя для другого таймера, например, Таймер 1 можно настроить для работы в качестве предварительного делителя для Таймера 2. См.[Использование одного таймера в качестве предделителя для другого таймера](#) Больше подробностей.

Внутренний источник часов (CK_INT)

Если контроллер ведомого режима отключен (SMS=000 в регистре TIMx_SMCR), то биты CEN, DIR (в регистре TIMx_CR1) и биты UG (в регистре TIMx_EGR) являются фактическими управляющими битами и могут быть изменены только программно (кроме UG, который автоматически очищается). Как только бит CEN записывается в 1, предварительный делитель тактируется внутренними часами CK_INT.

Рисунок 120 показано поведение схемы управления и счетчика в нормальном режиме без предделителя.

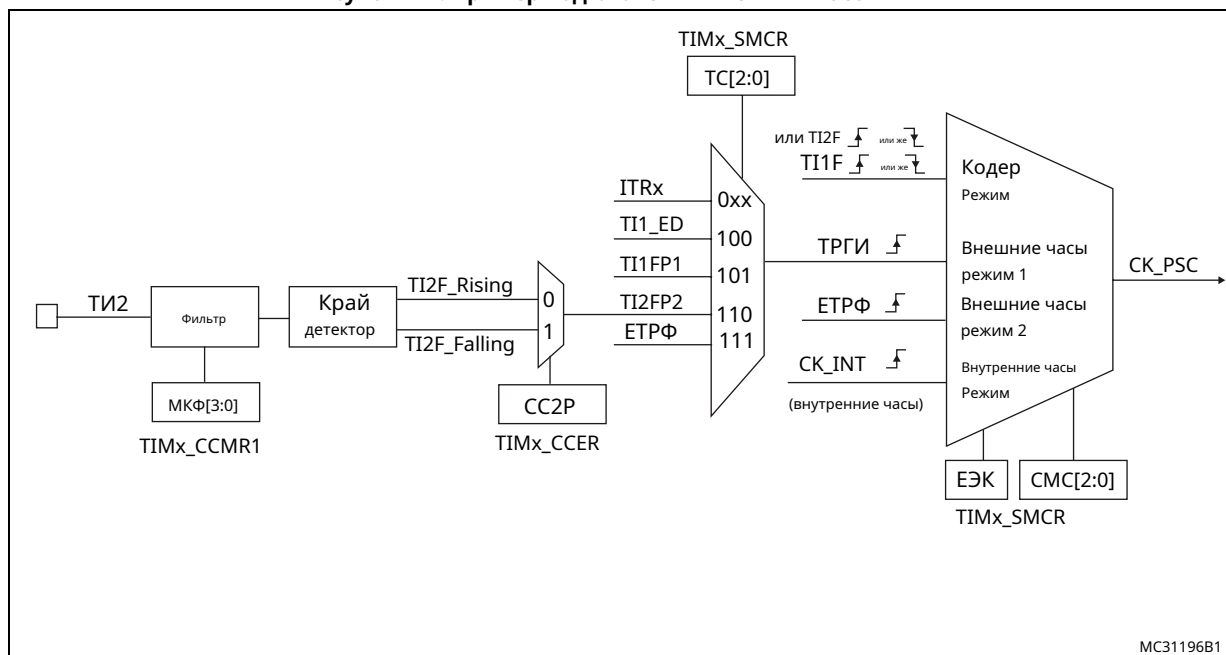
Рисунок 120. Схема управления в штатном режиме, внутренние часы, деленные на 1



Режим внешнего источника синхронизации 1

Этот режим выбирается, когда SMS=111 в регистре TIMx_SMCR. Счетчик может подсчитывать каждый нарастающий или спадающий фронт на выбранном входе.

Рисунок 121. Пример подключения внешних часов TI2



Например, чтобы настроить обратный счетчик для подсчета в ответ на нарастающий фронт на входе TI2, используйте следующую процедуру:

1. Настройте канал 2 для обнаружения нарастающих фронтов на входе TI2, записав CC2S= '01' в регистр TIMx_CCMR1.
2. Настройте длительность входного фильтра, записав биты IC2F[3:0] в регистр TIMx_CCMR1 (если фильтр не нужен, оставьте IC2F=0000).
3. Выберите полярность нарастающего фронта, записав CC2P=0 в регистр TIMx_CCER.
4. Настройте таймер в режиме внешних часов 1, написав SMS=111 в регистре TIMx_SMCR.
5. Выберите TI2 в качестве источника ввода, записав TS=110 в регистр TIMx_SMCR.
6. Включите счетчик, записав CEN=1 в регистр TIMx_CR1.

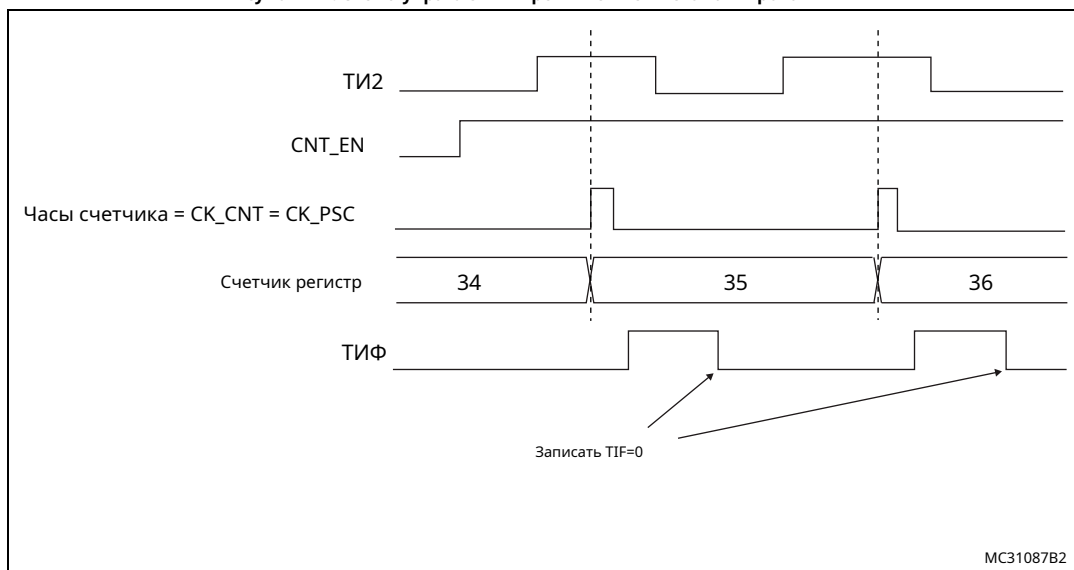
Примечание:

Предделитель захвата не используется для запуска, поэтому его не нужно настраивать.

Когда на TI2 возникает нарастающий фронт, счетчик считает один раз и устанавливается флаг TIF.

Задержка между нарастающим фронтом на TI2 и фактическим тактовым сигналом счетчика обусловлена схемой ресинхронизации на входе TI2.

Рисунок 122. Схема управления в режиме внешнего тактирования 1



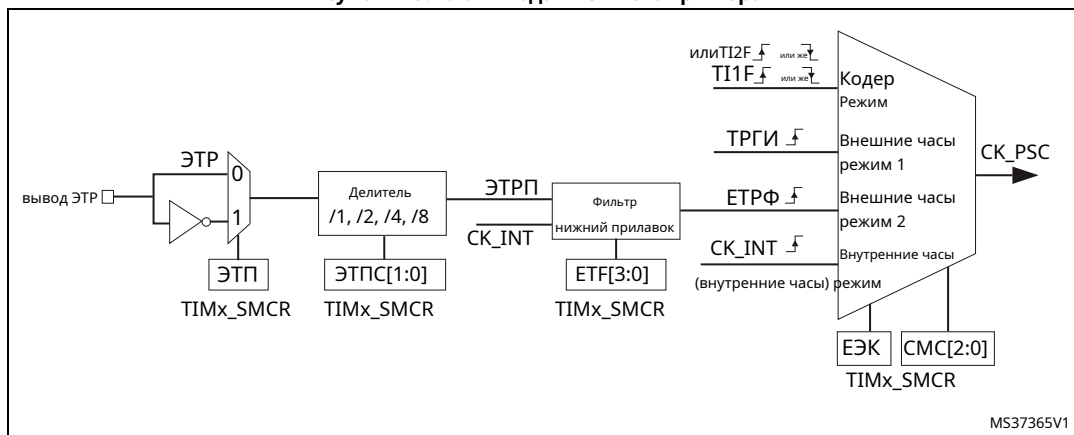
Режим внешнего источника синхронизации 2

Этот режим выбирается записью ECE=1 в регистр TIMx_SMCR.

Счетчик может считать по каждому переднему или заднему фронту на внешнем триггерном входе ETR.

[Рисунок 123](#) дает обзор внешнего триггерного входного блока.

Рисунок 123. Блок ввода внешнего триггера



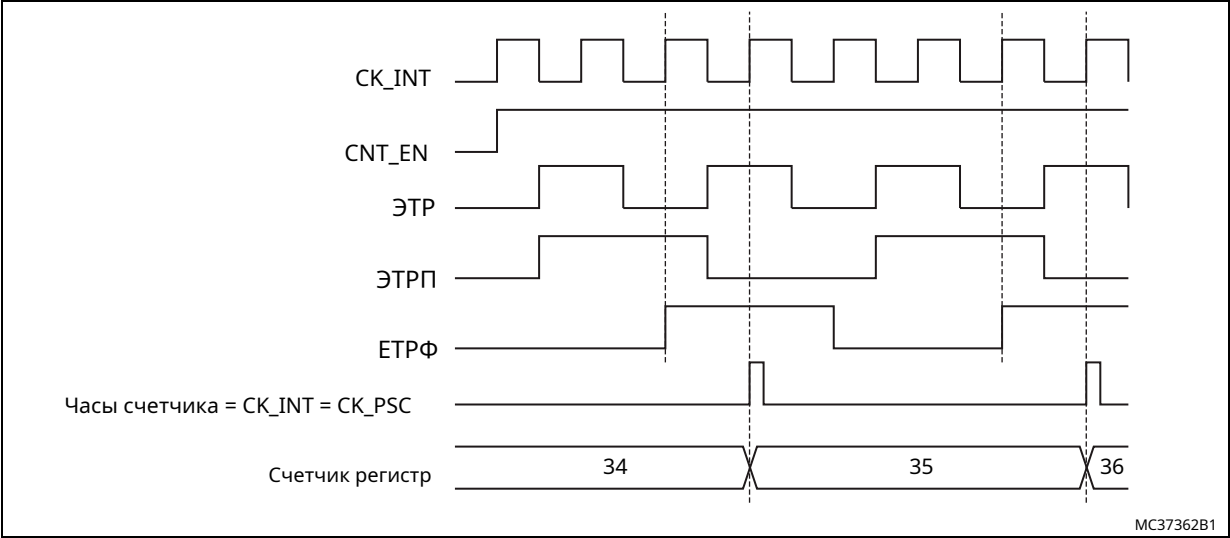
Например, чтобы настроить обратный счетчик для подсчета каждых 2 нарастающих фронтов ETR, используйте следующую процедуру:

1. Поскольку в этом примере фильтр не нужен, запишите ETF[3:0]=0000 в регистр TIMx_SMCR.
2. Установите прескалер, написав ETPS[1:0]=01 в регистре TIMx_SMCR.
3. Выберите обнаружение нарастающего фронта на выводе ETR, написав ETP=0 в регистре TIMx_SMCR.
4. Включите режим внешних часов 2, записав ECE=1 в регистр TIMx_SMCR.
5. Включите счетчик, записав CEN=1 в регистр TIMx_CR1.

Счетчик считает один раз каждые 2 нарастающих фронта ETR.

Задержка между нарастающим фронтом на ETR и фактическим тактовым сигналом счетчика обусловлена схемой ресинхронизации сигнала ETRP.

Рисунок 124. Схема управления в режиме внешнего тактирования 2

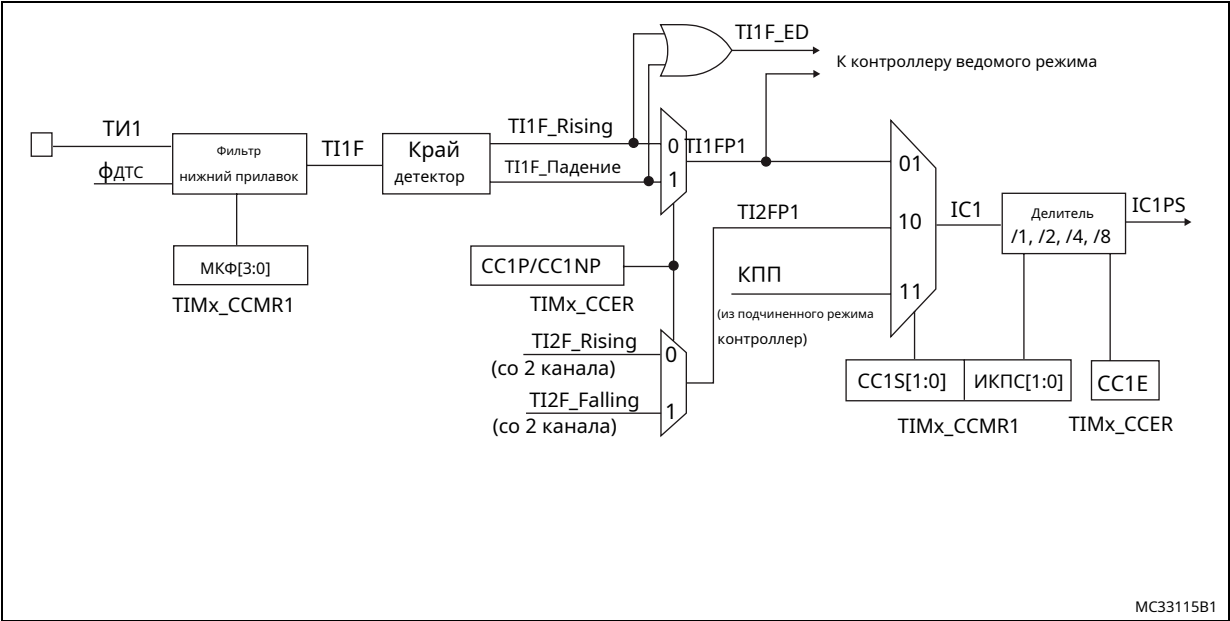


15.3.4 Захват/сравнение каналов

Каждый канал захвата/сравнения (см. [Рисунок 125](#)) построен на основе регистра захвата/сравнения (включая теневой регистр), входного каскада для захвата (с цифровым фильтром, мультиплексированием и предварительным делителем) и выходного каскада (с компаратором и управлением выходом).

Входной каскад производит выборку соответствующего входа TIx для генерации отфильтрованного сигнала TIxF. Затем детектор фронта с выбором полярности генерирует сигнал (TIxFPx), который может использоваться в качестве входного сигнала триггера контроллером ведомого режима или в качестве команды захвата. Он предварительно масштабируется перед регистром захвата (ICxPS).

Рис. 125. Канал захвата/сравнения (пример: входной каскад канала 1)



Выходной каскад генерирует промежуточный сигнал, который затем используется в качестве опорного: OCxRef (активный высокий уровень). Полярность действует в конце цепи.

Рис. 126. Основная цепь захвата/сравнения канала 1

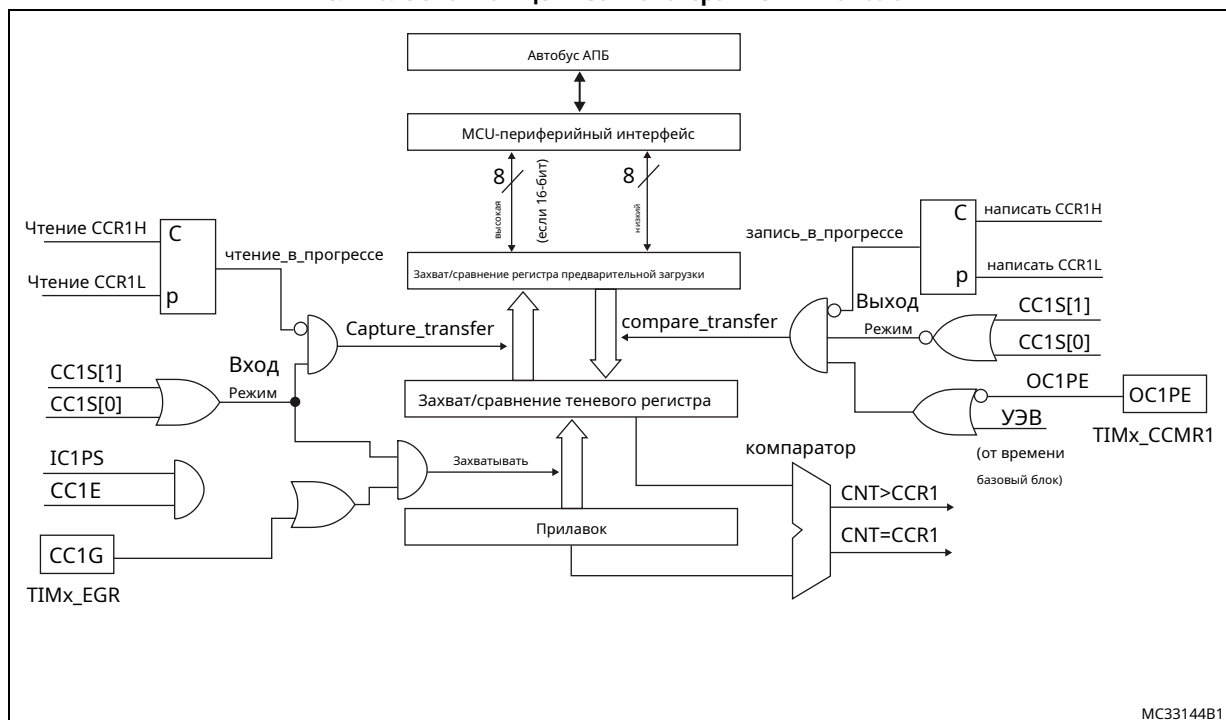
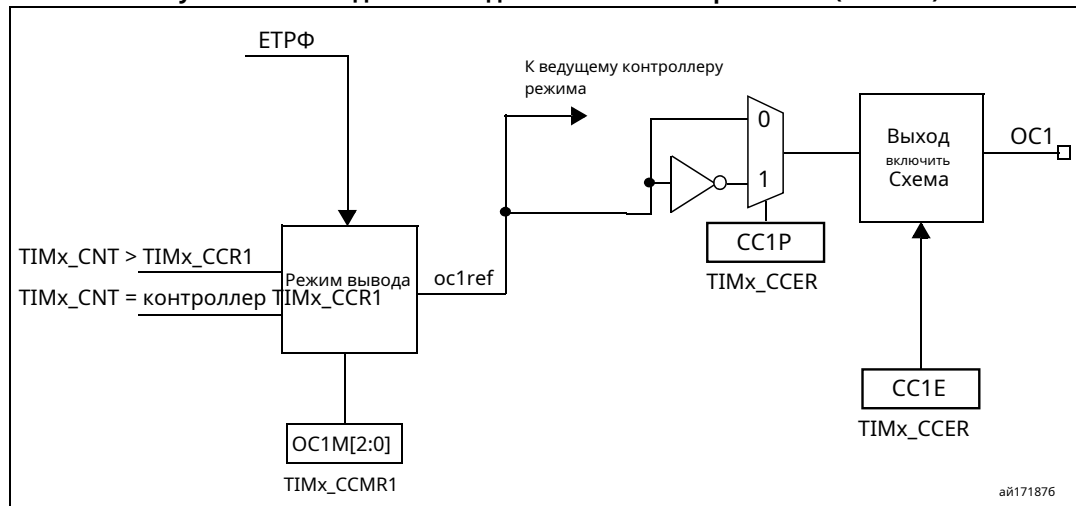


Рисунок 127. Выходной каскад канала захвата/сравнения (канал 1)



Блок захвата/сравнения состоит из одного регистра предварительной загрузки и одного теневого регистра. Запись и чтение всегда обращаются к регистру предварительной загрузки.

В режиме захвата захват фактически выполняется в теновой регистр, который копируется в регистр предварительной загрузки.

В режиме сравнения содержимое регистра предварительной загрузки копируется в теновой регистр, который сравнивается со счетчиком.

15.3.5 Режим захвата ввода

В режиме захвата ввода регистры захвата/сравнения (TIMx_CCRx) используются для фиксации значения счетчика после перехода, обнаруженного соответствующим сигналом ICx. Когда происходит захват, устанавливается соответствующий флаг CCxIF (регистр TIMx_SR) и может быть отправлено прерывание или запрос DMA, если они разрешены. Если захват происходит, когда флаг CCxIF уже имеет высокий уровень, то устанавливается флаг перехвата CCxOF (регистр TIMx_SR). CCxIF можно очистить программно, записав его в 0 или прочитав захваченные данные, хранящиеся в регистре TIMx_CCRx. CCxOF очищается при записи в 0.

В следующем примере показано, как зафиксировать значение счетчика в TIMx_CCR1, когда вход TI1 повышается. Для этого используйте следующую процедуру:

- Выберите активный вход: TIMx_CCR1 должен быть связан с входом TI1, поэтому запишите биты CC1S в 01 в регистре TIMx_CCMR1. Как только CC1S становится отличной от 00, канал настраивается на вход и регистр TIMx_CCR1 становится доступным только для чтения.
- Запрограммируйте необходимую продолжительность входного фильтра по отношению к сигналу, подключенному к таймеру (путем программирования битов ICxP в регистре TIMx_CCMRx, если вход является одним из входов TIx). Представим, что при переключении входной сигнал нестабилен в течение не менее пяти внутренних тактов. Мы должны запрограммировать длительность фильтра больше, чем эти пять тактов. Мы можем подтвердить переход на TI1, когда были обнаружены восемь последовательных выборок с новым уровнем (выборка в фдтс частота). Затем запишите биты IC1F в 0011 в регистре TIMx_CCMR1.
- Выберите фронт активного перехода на канале TI1, записав бит CC1P в 0 в регистре TIMx_CCER (в данном случае нарастающий фронт).
- Запрограммируйте входной прескалер. В нашем примере мы хотим, чтобы захват выполнялся при каждом допустимом переходе, поэтому предварительный делитель отключен (запишите биты IC1PS в 00 в регистре TIMx_CCMR1).
- Включите захват из счетчика в регистр захвата, установив бит CC1E в регистре TIMx_CCER.
- При необходимости включите соответствующий запрос прерывания, установив бит CC1IE в регистре TIMx_DIER, и/или запрос DMA, установив бит CC1DE в регистре TIMx_DIER.

Когда происходит захват ввода:

- Регистр TIMx_CCR1 получает значение счетчика активного перехода.
- Установлен флаг CC1IF (флаг прерывания). CC1OF также устанавливается, если произошло по крайней мере два последовательных захвата, а флаг не был очищен.
- Прерывание генерируется в зависимости от бита CC1IE.
- Запрос DMA генерируется в зависимости от бита CC1DE.

Чтобы справиться с избыточным захватом, рекомендуется читать данные перед флагом избыточного захвата. Это делается для того, чтобы не пропустить перехват, который может произойти после чтения флага и до чтения данных.

Примечание:

Прерывание IC и/или запросы DMA могут генерироваться программным обеспечением путем установки соответствующего бита CCxG в регистре TIMx_EGR.

15.3.6 Режим ввода ШИМ

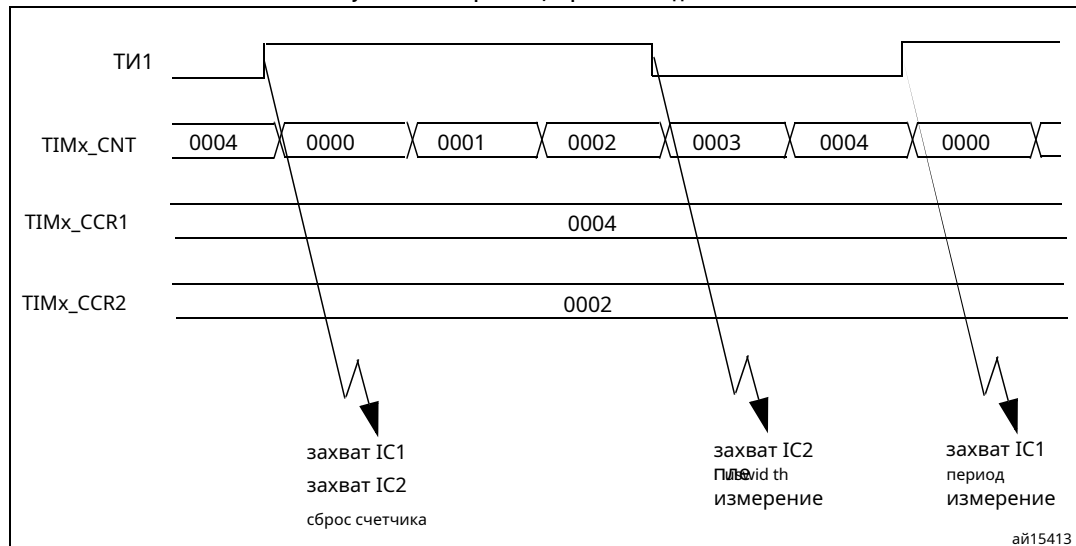
Этот режим является частным случаем режима захвата ввода. Процедура такая же, за исключением:

- Два сигнала ICx отображаются на один и тот же вход TIx. Эти 2 сигнала ICx активны на фронтах с противоположной полярностью.
- Один из двух сигналов TIxFP выбирается в качестве входа триггера, а контроллер ведомого режима конфигурируется в режиме сброса.

Например, пользователь может измерить период (в регистре TIMx_CCR1) и рабочий цикл (в регистре TIMx_CCR2) ШИМ, примененного к TI1, используя следующую процедуру (в зависимости от частоты CK_INT и значения предделителя):

- Выберите активный вход для TIMx_CCR1: запишите биты CC1S в 01 в регистре TIMx_CCMR1 (выбран TI1).
- Выберите активную полярность для TI1FP1 (используется как для захвата в TIMx_CCR1, так и для сброса счетчика): запишите CC1P в '0' (активен по переднему фронту).
- Выберите активный вход для TIMx_CCR2: запишите биты CC2S в 10 в регистре TIMx_CCMR1 (выбран TI1).
- Выберите активную полярность для TI1FP2 (используется для захвата в TIMx_CCR2): запишите бит CC2P в «1» (активен по заднему фронту).
- Выберите допустимый вход триггера: запишите биты TS в 101 в регистре TIMx_SMCR (выбран TI1FP1).
- Настройте контроллер ведомого режима в режиме сброса: запишите биты SMS в 100 в регистре TIMx_SMCR.
- Включите захват: запишите биты CC1E и CC2E в '1' в регистре TIMx_CCER.

Рисунок 128. Синхронизация режима ввода ШИМ



1. Режим ввода ШИМ можно использовать только с сигналами TIMx_CH1/TIMx_CH2 из-за того, что к контроллеру ведомого режима подключены только TI1FP1 и TI2FP2.

15.3.7 Режим принудительного вывода

В режиме вывода (биты CCxS = 00 в регистре TIMx_CCMRx) каждый выходной сигнал сравнения (OCxREF, а затем OCx) может быть переведен в активный или неактивный уровень непосредственно программно, независимо от любого сравнения между выходным регистром сравнения и счетчиком.

Чтобы перевести выходной сигнал сравнения (osxref/OCx) на его активный уровень, пользователю достаточно записать 101 в биты OCxM в соответствующем регистре TIMx_CCMRx. Таким образом, osxref принудительно устанавливается на высокий уровень (OCxREF всегда активен на высоком уровне), а OCx получает значение, противоположное биту полярности CCxP.

например: CCxP=0 (OCx активен на высоком уровне) => OCx принудительно устанавливается на высокий уровень.

Сигнал osxref может быть принудительно установлен на низкий уровень путем записи битов OCxM в 100 в регистре TIMx_CCMRx.

В любом случае, сравнение между теневым регистром TIMx_CCRx и счетчиком все еще выполняется и позволяет установить флаг. Соответственно могут быть отправлены запросы прерывания и DMA. Это описано в следующем разделе.

15.3.8 Режим сравнения выхода

Эта функция используется для управления формой выходного сигнала или индикации истечения периода времени.

Когда найдено совпадение между регистром захвата/сравнения и счетчиком, функция сравнения вывода:

- Назначает соответствующему выходному контакту программируемое значение, определяемое режимом сравнения выходов (биты OCxM в регистре TIMx_CCMRx) и полярностью выхода (бит CCxP в регистре TIMx_CCER). Выходной контакт может сохранять свой уровень (OCxM=000), быть установлен активным (OCxM=001), быть установлен неактивным (OCxM=010) или может переключаться (OCxM=011) при совпадении.
- Устанавливает флаг в регистре статуса прерывания (бит CCxIF в регистре TIMx_SR).
- Генерирует прерывание, если установлена соответствующая маска прерывания (бит CCxIE в регистре TIMx_DIER).
- Отправляет запрос DMA, если установлен соответствующий бит разрешения (бит CCxDE в регистре TIMx_DIER, бит CCDS в регистре TIMx_CR2 для выбора запроса DMA).

Регистры TIMx_CCRx можно запрограммировать с регистрами предварительной загрузки или без них, используя бит OCxPE в регистре TIMx_CCMRx.

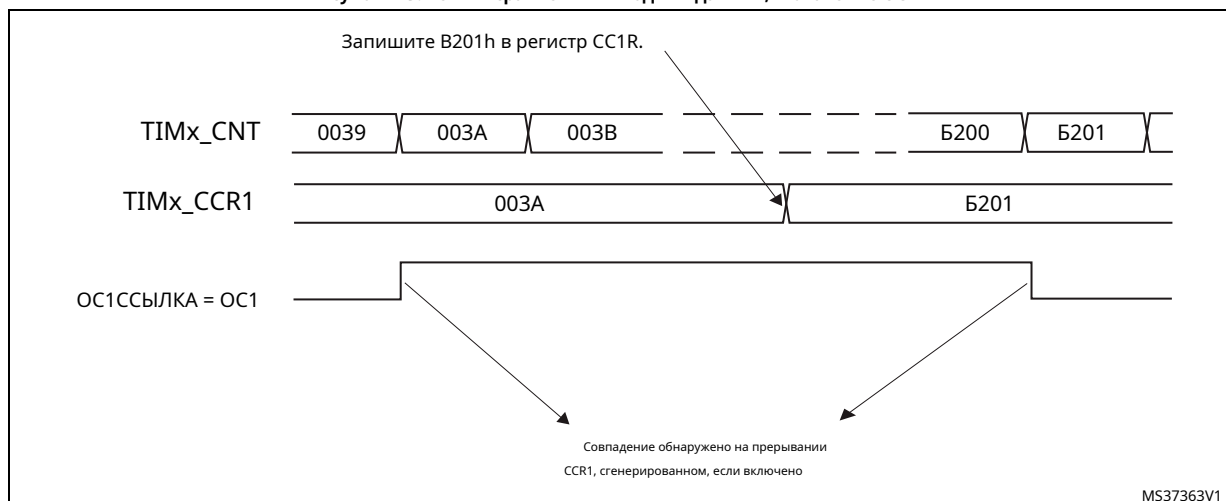
В режиме сравнения вывода событие обновления UEV не влияет на вывод osxref и OCx. Разрешение по времени составляет один отсчет счетчика. Режим сравнения вывода также можно использовать для вывода одиночного импульса (в одноимпульсном режиме).

Процедура:

1. Выберите часы счетчика (внутренние, внешние, предварительный делитель).
2. Запишите нужные данные в регистры TIMx_ARR и TIMx_CCRx.
3. Установите биты CCxIE и/или CCxDE, если необходимо сгенерировать прерывание и/или запрос DMA.
4. Выберите режим вывода. Например, пользователь должен написать OCxM=011, OCxPE=0, CCxP=0 и CCxE=1, чтобы переключить выходной контакт OCx, когда CNT соответствует CCRx, предварительная загрузка CCRx не используется, OCx включен и имеет активный высокий уровень.
5. Включите счетчик, установив бит CEN в регистре TIMx_CR1.

Регистр TIMx_CCRx может быть обновлен в любое время программным обеспечением для управления формой выходного сигнала, при условии, что регистр предварительной загрузки не включен (OSxPE=0, в противном случае теневого регистр TIMx_CCRx обновляется только при следующем событии обновления UEV). Пример приведен в [Рисунок 129](#).

Рисунок 129. Режим сравнения выходных данных, включение OC1



15.3.9 ШИМ-режим

Режим широтно-импульсной модуляции позволяет генерировать сигнал с частотой, определяемой значением регистра TIMx_ARR, и скважностью, определяемой значением регистра TIMx_CCRx.

Режим ШИМ можно выбрать независимо для каждого канала (один ШИМ на каждый выход OCx), записав 110 (режим ШИМ 1) или '111 (режим ШИМ 2) в битах OSxM в регистре TIMx_CCMRx. Пользователь должен включить соответствующий регистр предварительной загрузки, установив бит OSxPE в регистре TIMx_CCMRx, и, в конечном итоге, регистр предварительной загрузки с автоматической перезагрузкой, установив бит ARPE в регистре TIMx_CR1.

Поскольку регистры предварительной загрузки передаются в теньевые регистры только при возникновении события обновления, перед запуском счетчика пользователь должен инициализировать все регистры, установив бит UG в регистре TIMx_EGR.

Полярность OCx программируется с помощью бита CCxP в регистре TIMx_CCER. Его можно запрограммировать как активный высокий или активный низкий. Выход OCx активируется битом CCxE в регистре TIMx_CCER. Дополнительные сведения см. в описании регистра TIMx_CCERx.

В режиме PWM (1 или 2) TIMx_CNT и TIMx_CCRx всегда сравниваются, чтобы определить, является ли TIMx_CCRx-TIMx_CNT или TIMx_CNT-TIMx_CCRx (в зависимости от направления счетчика). Однако, чтобы соответствовать ETRF (OCREF может быть очищен внешним событием через сигнал ETR до следующего периода PWM), сигнал OCREF утверждается только:

- Когда результат сравнения изменяется или
- Когда выходной режим сравнения (биты OSxM в регистре TIMx_CCMRx) переключается с «замороженной» конфигурации (без сравнения, OSxM='000) на один из режимов ШИМ (OSxM='110 или '111).

Это вызывает ШИМ с помощью программного обеспечения во время работы таймера.

Таймер может генерировать ШИМ в режиме с выравниванием по краю или с выравниванием по центру в зависимости от битов CMS в регистре TIMx_CR1.

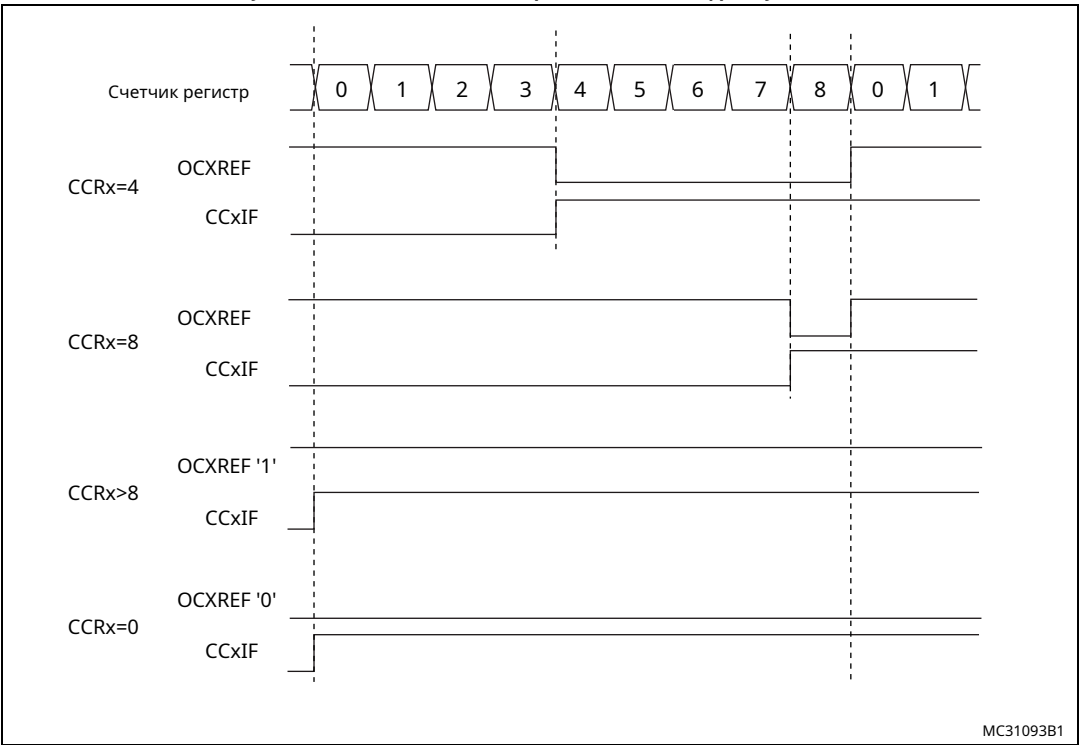
Режим ШИМ с выравниванием по фронту

Конфигурация прямого подсчета

Счет вверх активен, когда бит DIR в регистре TIMx_CR1 низкий. Ссылаться на [Режим прямого счета](#).

В следующем примере мы рассматриваем режим ШИМ 1. Опорный сигнал ШИМ OCxREF имеет высокий уровень, пока TIMx_CNT <TIMx_CCRx, в противном случае он становится низким. Если значение сравнения в TIMx_CCRx больше, чем значение автоматической перезагрузки (в TIMx_ARR), то OCxREF удерживается на уровне '1'. Если значение сравнения равно 0, то OCxREF сохраняется равным '0'. [Рисунок 130](#) показывает некоторые сигналы ШИМ с выравниванием по краю в примере, где TIMx_ARR=8.

Рисунок 130. ШИМ-сигналы с выравниванием по фронту (ARR=8)



Конфигурация обратного отсчета

Понижающий счет активен, когда бит DIR в регистре TIMx_CR1 имеет высокий уровень. Ссылаться на [Режим обратного отсчета](#).

В режиме ШИМ 1 опорный сигнал osxref имеет низкий уровень, пока TIMx_CNT>TIMx_CCRx, в противном случае он становится высоким. Если значение сравнения в TIMx_CCRx больше, чем значение автоматической перезагрузки в TIMx_ARR, то osxref сохраняется равным '1'. 0% ШИМ в этом режиме невозможен.

Режим ШИМ с выравниванием по центру

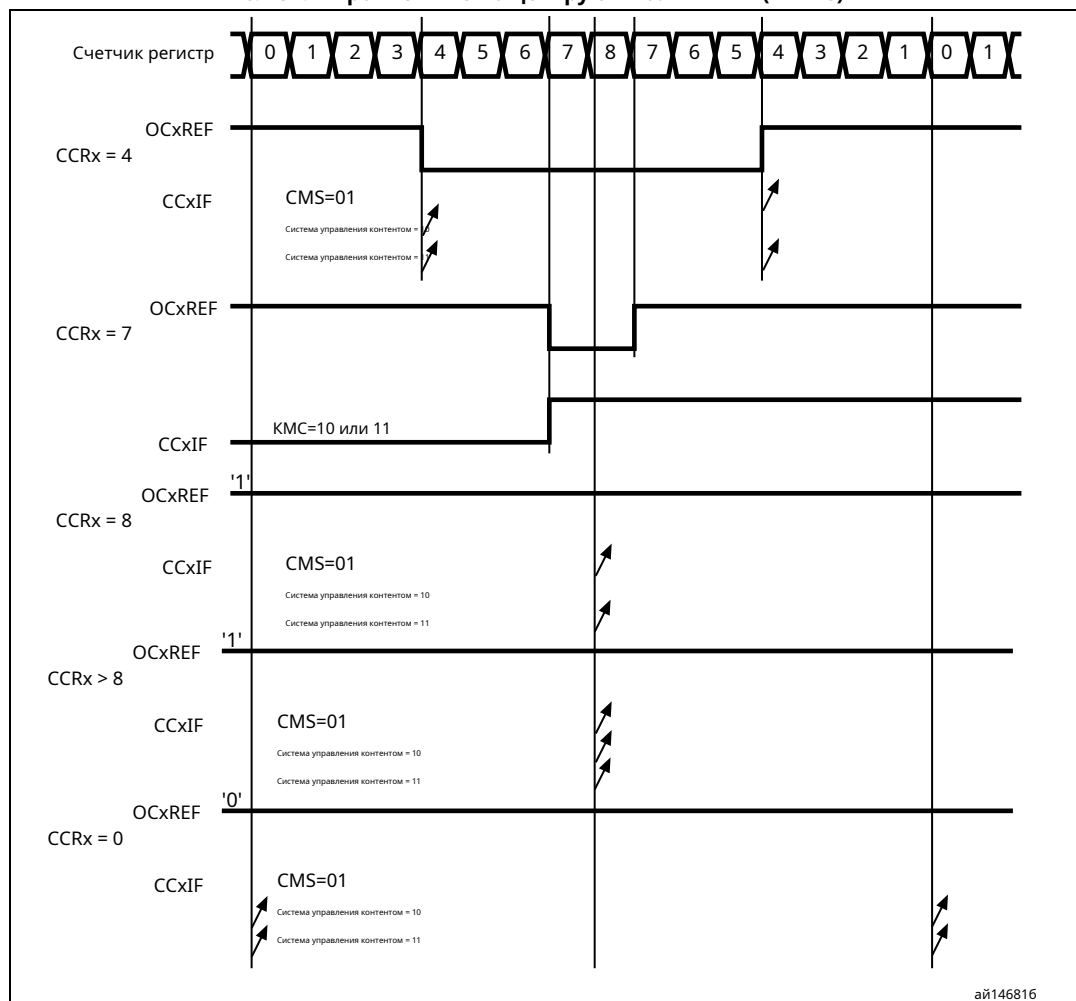
Выровненный по центру режим активен, когда биты CMS в регистре TIMx_CR1 отличны от '00 (все остальные конфигурации имеют одинаковый эффект на сигналы osxref/OCx). Флаг сравнения устанавливается, когда счетчик считает вверх, когда он считает вниз или и то, и другое, когда он считает

вверх и вниз в зависимости от конфигурации битов CMS. Бит направления (DIR) в регистре TIMx_CR1 обновляется аппаратно и не должен изменяться программно. Ссылаться на [Режим выравнивания по центру \(счет вверх/вниз\)](#).

[Рисунок 131](#) показывает некоторые выровненные по центру сигналы ШИМ в примере, где:

- TIMx_ARR=8,
- Режим ШИМ-это режим ШИМ 1,
- Флаг устанавливается, когда счетчик ведет обратный отсчет в соответствии с режимом выравнивания по центру 1, выбранным для CMS=01 в регистре TIMx_CR1.

Рис. 131. Выровненные по центру сигналы ШИМ (ARR=8)



Советы по использованию режима с выравниванием по центру:

- При запуске в режиме с выравниванием по центру используется текущая конфигурация вверх-вниз. Это означает, что счетчик считает вверх или вниз в зависимости от значения, записанного в бите DIR.

в регистре TIMx_CR1. Кроме того, программное обеспечение не должно одновременно изменять биты DIR и CMS.

- Запись в счетчик при работе в режиме выравнивания по центру не рекомендуется, так как это может привести к неожиданным результатам. Особенно:
 - Направление не обновляется, если пользователь записывает в счетчик значение, превышающее значение автоматической перезагрузки ($TIMx_CNT > TIMx_ARR$). Например, если счетчик считал вверх, он продолжает считать вверх.
 - Направление обновляется, если пользователь записывает 0 или записывает значение TIMx_ARR в счетчик, но событие обновления UEV не генерируется.
- Самый безопасный способ использования режима с выравниванием по центру — это сгенерировать программное обновление (установив бит UG в регистре TIMx_EGR) непосредственно перед запуском счетчика, а не записывать счетчик во время его работы.

15.3.10 Одноимпульсный режим

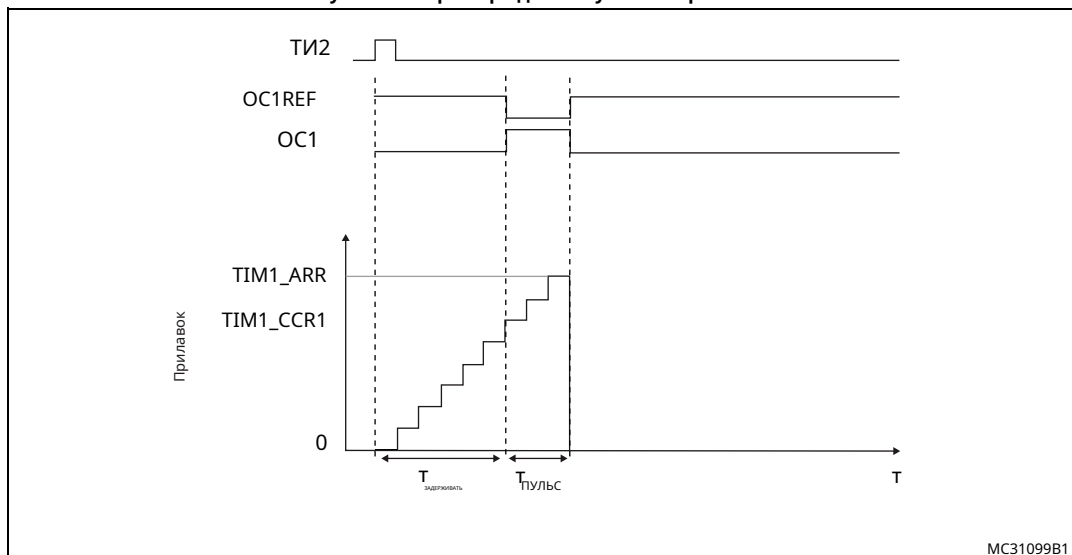
Одноимпульсный режим (ОПМ) является частным случаем предыдущих режимов. Он позволяет запускать счетчик в ответ на стимул и генерировать импульс программируемой длины после программируемой задержки.

Запуском счетчика можно управлять через контроллер ведомого режима. Генерация сигнала может выполняться в режиме сравнения выходных сигналов или в режиме ШИМ. Выберите одноимпульсный режим, установив бит OPM в регистре TIMx_CR1. Это автоматически останавливает счетчик при следующем событии обновления UEV.

Импульс может быть сгенерирован правильно, только если значение сравнения отличается от начального значения счетчика. Перед запуском (когда таймер ожидает срабатывания) конфигурация должна быть:

- При прямом счете: $CNT < CCRx_ARR$ (в частности, $0 < CCRx$), при
- обратном счете: $CNT > CCRx$.

Рисунок 132. Пример одноимпульсного режима



Например, пользователь может захотеть сгенерировать положительный импульс на OC1 длительностью t_{импульс} и после задержки t_{задержка} как только на входе TI2 будет обнаружен положительный фронт.

Давайте используем TI2FP2 в качестве триггера 1:

- Сопоставьте TI2FP2 с TI2, записав CC2S=01 в регистр TIMx_CCMR1. TI2FP2 должен обнаружить нарастающий фронт, записать CC2P=0 в регистр TIMx_CCER.
- Настройте TI2FP2 как триггер для контроллера ведомого режима (TRGI), записав TS=110 в регистр TIMx_SMCR.
- TI2FP2 используется для запуска счетчика путем записи SMS на '110 в регистре TIMx_SMCR (режим запуска).

Форма волны OPM определяется записью регистров сравнения (с учетом тактовой частоты и счетчика прескалера).

- tЗАДЕРЖИВАТЬ определяется значением, записанным в регистре TIMx_CCR1.
- tПУЛЬС определяется разницей между значением автоматической перезагрузки и значением сравнения (TIMx_ARR - TIMx_CCR + 1).
- Допустим, пользователь хочет построить сигнал с переходом от '0 к '1, когда происходит совпадение сравнения, и переходом от '1 к '0, когда счетчик достигает значения автоматической перезагрузки. Для этого включите режим ШИМ 2, записав OC1M=111 в регистр TIMx_CCMR1. Пользователь может дополнительно включить регистры предварительной загрузки, написав OC1PE=1 в регистре TIMx_CCMR1 и ARPE в регистре TIMx_CR1. В этом случае запишите значение сравнения в регистр TIMx_CCR1, значение автоматической перезагрузки в регистр TIMx_ARR, сгенерируйте обновление, установив бит UG, и дождитесь события внешнего триггера на TI2. CC1P записывается в '0 в этом примере.

В нашем примере биты DIR и CMS в регистре TIMx_CR1 должны быть низкими.

Пользователю нужен только один импульс (одиночный режим), поэтому запишите '1 в бит OPM в регистре TIMx_CR1, чтобы остановить счетчик при следующем событии обновления (когда счетчик переходит от значения автоматической перезагрузки обратно к 0). Когда бит OPM в регистре TIMx_CR1 установлен на «0», значит, выбран повторяющийся режим.

Частный случай: быстрое включение ОСх:

В одноимпульсном режиме обнаружение фронта на входе TIx устанавливает бит CEN, который включает счетчик. Затем сравнение между счетчиком и сравниваемым значением приводит к переключению вывода. Но для этих операций требуется несколько тактов, что ограничивает минимальную задержку tЗАДЕРЖИВАТЬ мин мы можем получить.

Для вывода сигнала с минимальной задержкой пользователь может установить бит OCxFE в регистре TIMx_CCMRx. Затем OCxRef (и OCx) форсируется в ответ на стимул без учета сравнения. Его новый уровень такой же, как если бы произошло совпадение сравнения. OCxFE действует, только если канал настроен в режиме PWM1 или PWM2.

15.3.11 Очистка сигнала OCxREF при внешнем событии

Сигнал OCxREF для данного канала может быть переведен в низкий уровень путем подачи высокого уровня на вход ETRF (бит разрешения OCxCE соответствующего регистра TIMx_CCMRx установлен в '1'). Сигнал OCxREF остается низким до тех пор, пока не произойдет следующее событие обновления, UEV.

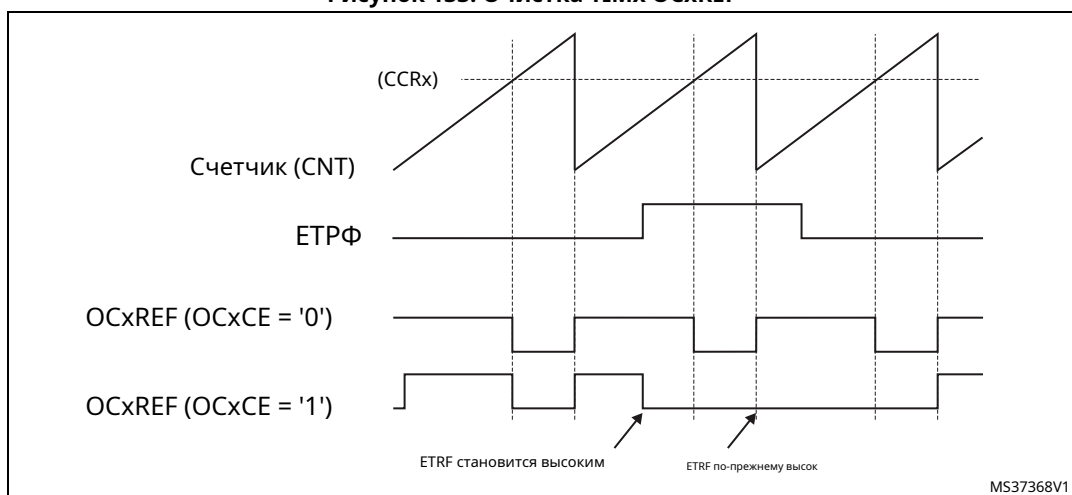
Эта функция может использоваться только в режимах сравнения выхода и ШИМ и не работает в принудительном режиме.

Например, сигнал ЭТР можно подключить к выходу компаратора, который будет использоваться для обработки тока. В этом случае ETR необходимо настроить следующим образом:

1. Предварительный делитель внешнего триггера должен быть отключен: биты ETPS[1:0] в регистре TIMx_SMCR очищаются до 00.
2. Режим внешних часов 2 должен быть отключен: бит ECE в регистре TIM1_SMCR очищается до 0.
3. Полярность внешнего запуска (ETP) и фильтр внешнего запуска (ETF) можно настроить в соответствии с потребностями приложения.

Рисунок 133 показывает поведение сигнала OCxREF, когда вход ETRF становится высоким, для обоих значений бита разрешения OCxCE. В этом примере таймер TIMx запрограммирован в режиме PWM.

Рисунок 133. Очистка TIMx OCxREF



15.3.12 Режим интерфейса энкодера

Чтобы выбрать режим интерфейса кодировщика, напишите SMS='001 в регистре TIMx_SMCR, если счетчик считает только фронты TI2, SMS=010, если он считает только фронты TI1, и SMS=011, если он считает и фронты TI1, и TI2.

Выберите полярность TI1 и TI2, запрограммировав биты CC1P и CC2P в регистре TIMx_CCER. При необходимости также запрограммируйте входной фильтр.

Два входа TI1 и TI2 используются для интерфейса с инкрементным энкодером. Ссылаться на [Таблица 85](#). Счетчик тактируется каждым действительным переходом на TI1FP1 или TI2FP2 (TI1 и TI2 после входного фильтра и выбора полярности, TI1FP1=TI1, если не фильтруется и не инвертируется, TI2FP2=TI2, если не фильтруется и не инвертируется) при условии, что он включен (CEN бит в регистре TIMx_CR1 записан в '1'). Последовательность переходов двух входов оценивается и генерирует счетные импульсы, а также сигнал направления. В зависимости от последовательности, в которой счетчик считает вверх или вниз, бит DIR в регистре TIMx_CR1 соответствующим образом модифицируется аппаратно. Бит DIR вычисляется при каждом переходе на любом входе (TI1 или TI2), независимо от того, считает ли счетчик только TI1, только TI2 или оба TI1 и TI2.

Режим интерфейса энкодера действует просто как внешние часы с выбором направления. Это означает, что счетчик просто непрерывно считает между 0 и значением автоматической перезагрузки в регистре TIMx_ARR (от 0 до ARR или от ARR до 0 в зависимости от направления). Таким образом, пользователь должен настроить TIMx_ARR перед запуском. Точно так же функции захвата, сравнения, предварительного масштабирования и триггерного вывода продолжают работать в обычном режиме.

В этом режиме счетчик изменяется автоматически в соответствии со скоростью и направлением инкрементального энкодера, а его содержимое всегда представляет положение энкодера. Направление счета соответствует направлению вращения подключенного датчика. В таблице приведены возможные комбинации, предполагая, что TI1 и TI2 не переключаются одновременно.

Таблица 85. Направление счета в зависимости от сигналов энкодера

Активный край	Уровень напротив сигнал (TI1FP1 для TI2, TI2FP2 для TI1)	сигнал TI1FP1		сигнал TI2FP2	
		рост	Падение	рост	Падение
Расчитывать на только TI1	Высокая	Вниз	Вверх	Нет счета	Нет счета
	Низкий	Вверх	Вниз	Нет счета	Нет счета
Расчитывать на только TI2	Высокая	Нет счета	Нет счета	Вверх	Вниз
	Низкий	Нет счета	Нет счета	Вниз	Вверх
Расчитывать на TI1 и TI2	Высокая	Вниз	Вверх	Вверх	Вниз
	Низкий	Вверх	Вниз	Вниз	Вверх

Внешний инкрементный энкодер может быть подключен непосредственно к MCU без логики внешнего интерфейса. Однако компараторы обычно используются для преобразования дифференциальных выходных сигналов энкодера в цифровые сигналы. Это значительно повышает помехозащищенность. Третий выход энкодера, который показывает положение механического нуля, может быть подключен к внешнему входу прерывания и вызывать сброс счетчика.

[Рисунок 134](#) дает пример работы счетчика, показывающий генерацию сигнала счета и управление направлением. Также показано, как компенсируется входной джиттер, если выбраны оба фронта. Это может произойти, если датчик расположен рядом с одной из точек переключения. Для этого примера мы предполагаем, что конфигурация следующая:

- CC1S= '01' (регистр TIMx_CCMR1, TI1FP1 отображается на TI1)
- CC2S= '01' (регистр TIMx_CCMR2, TI2FP2 отображается на TI2)
- CC1P= '0', CC1NP = '0', IC1F = '0000' (регистр TIMx_CCER, TI1FP1 неинвертированный, TI1FP1=TI1)
- CC2P= '0', CC2NP = '0', IC2F = '0000' (регистр TIMx_CCER, TI2FP2 неинвертированный, TI2FP2=TI2)
- SMS= '011' (регистр TIMx_SMCR, оба входа активны как по переднему, так и по заднему фронту)
- CEN = 1 (регистр TIMx_CR1, счетчик включен)

Рисунок 134. Пример работы счетчика в режиме интерфейса энкодера

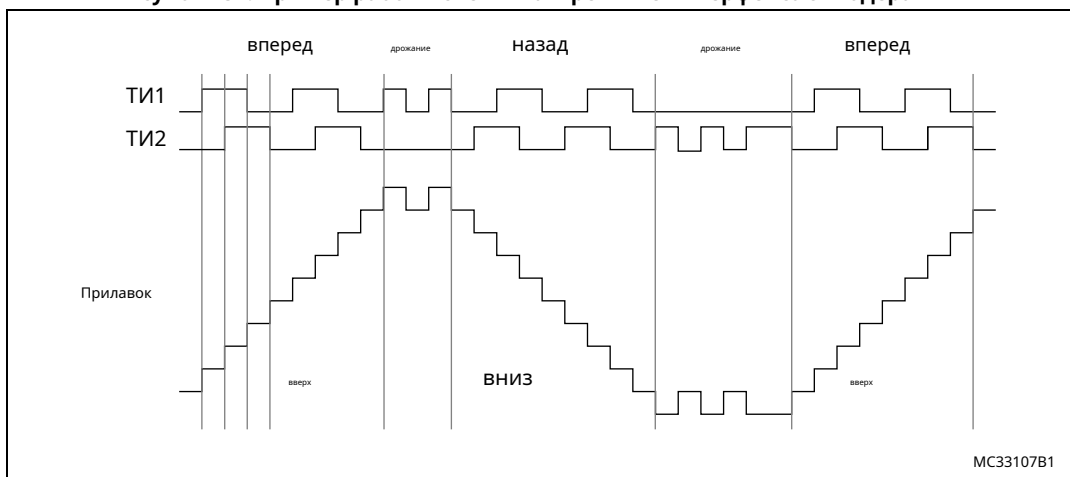
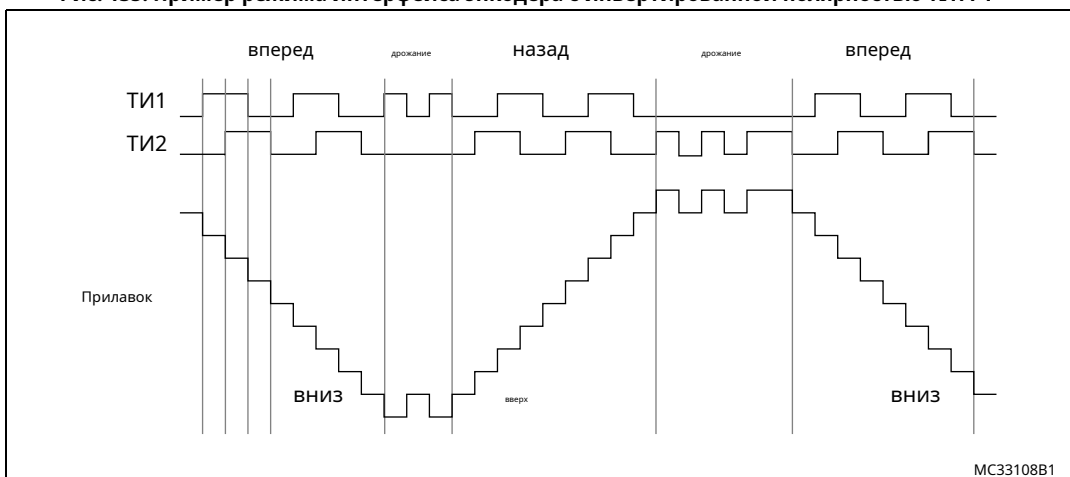


Рисунок 135 дает пример поведения счетчика, когда полярность TI1FP1 инвертирована (та же конфигурация, что и выше, за исключением CC1P=1).

Рис. 135. Пример режима интерфейса энкодера с инвертированной полярностью TI1FP1



Таймер, настроенный в режиме интерфейса энкодера, предоставляет информацию о текущем положении датчика. Пользователь может получить динамическую информацию (скорость, ускорение, замедление), измеряя период между двумя событиями энкодера с помощью второго таймера, настроенного в режиме захвата. Для этой цели можно использовать выход энкодера, который показывает механический ноль. В зависимости от времени между двумя событиями счетчик также может считываться в обычное время. Пользователь может сделать это, зафиксировав значение счетчика в третьем входном регистре захвата, если он доступен (тогда сигнал захвата должен быть периодическим и может генерироваться другим таймером). когда он доступен, его значение также можно прочитать с помощью запроса DMA, сгенерированного часами реального времени.

15.3.13 Вход таймера, функция XOR

Бит TI1S в регистре TIM1_CR2 позволяет подключить входной фильтр канала 1 к выходу вентиля XOR, объединяя три входных контакта TIMx_CH1 — TIMx_CH3.

Выход XOR можно использовать со всеми входными функциями таймера, такими как триггер или захват ввода.

Пример этой функции, используемой для интерфейса датчиков Холла, приведен в [Раздел 14.3.18](#).

15.3.14 Таймеры и синхронизация внешних триггеров

Таймеры TIMx можно синхронизировать с внешним триггером в нескольких режимах: режим сброса, стробируемый режим и режим триггера.

Ведомый режим: режим сброса

Счетчик и его предварительный делитель могут быть повторно инициализированы в ответ на событие на входе триггера. Более того, если бит URS в регистре TIMx_CR1 низкий, генерируется событие обновления UEV. Затем обновляются все предварительно загруженные регистры (TIMx_ARR, TIMx_CCRx).

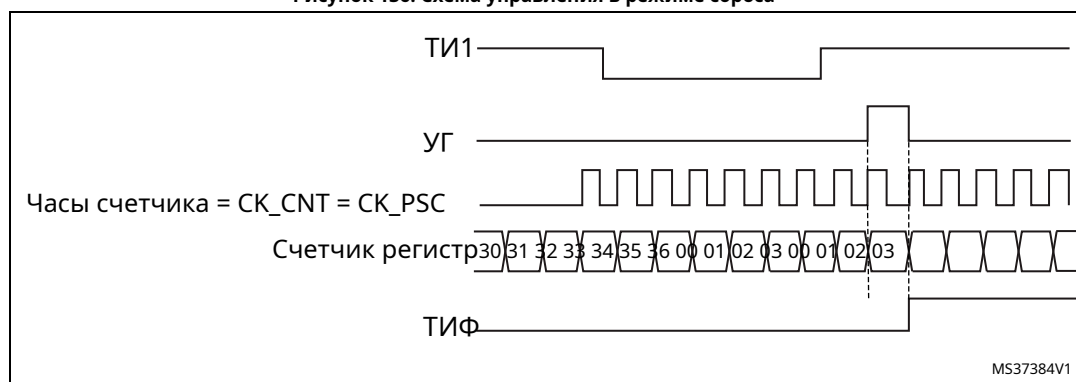
В следующем примере обратный счетчик сбрасывается в ответ на нарастающий фронт на входе TI1:

- Настройте канал 1 для обнаружения нарастающих фронтов на TI1. Настройте продолжительность входного фильтра (в этом примере нам не нужен фильтр, поэтому мы сохраняем IC1F=0000). Предделитель захвата не используется для запуска, поэтому пользователю не нужно его настраивать. Биты CC1S выбирают только источник захвата ввода, CC1S = 01 в регистре TIMx_CCMR1. Запишите CC1P=0 в регистр TIMx_CCER, чтобы проверить полярность (и обнаружить только нарастающие фронты).
- Настройте таймер в режиме сброса, написав SMS=100 в регистре TIMx_SMCR. Выберите TI1 в качестве источника ввода, записав TS=101 в регистр TIMx_SMCR.
- Запустите счетчик, записав CEN=1 в регистр TIMx_CR1.

Счетчик начинает считать по внутреннему тактовому сигналу, затем ведет себя нормально до переднего фронта TI1. Когда TI1 увеличивается, счетчик очищается и перезапускается с 0. Тем временем устанавливается флаг триггера (бит TIF в регистре TIMx_SR) и может быть отправлен запрос прерывания или запрос DMA, если он разрешен (в зависимости от TIE и биты TDE в регистре TIMx_DIER).

[Рисунок 136](#) показывает это поведение, когда регистр автоматической перезагрузки TIMx_ARR=0x36. Задержка между передним фронтом на TI1 и фактическим сбросом счетчика обусловлена схемой ресинхронизации на входе TI1.

Рисунок 136. Схема управления в режиме сброса



Ведомый режим: закрытый режим

Счетчик может быть включен в зависимости от уровня выбранного входа.

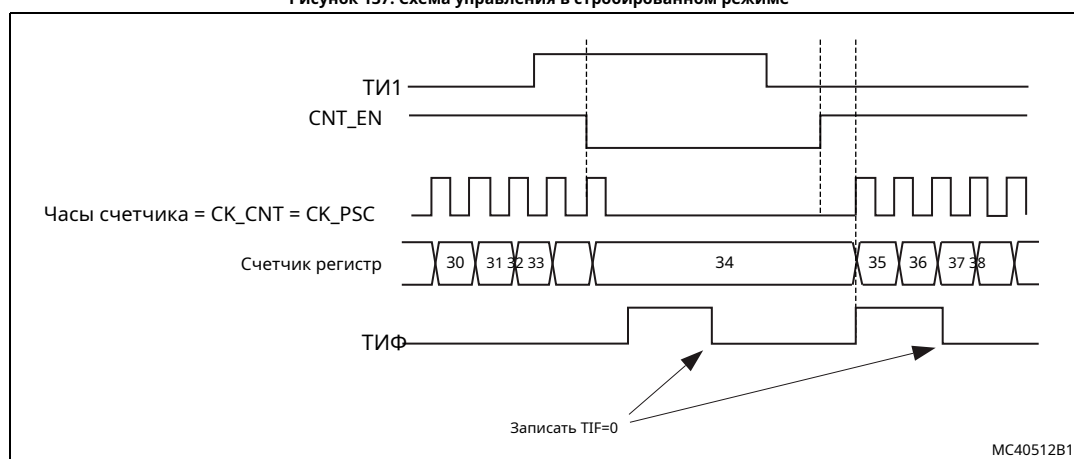
В следующем примере обратный счетчик учитывается только тогда, когда на входе TI1 низкий уровень:

- Настройте канал 1 для обнаружения низких уровней на TI1. Настройте продолжительность входного фильтра (в этом примере нам не нужен фильтр, поэтому мы сохраняем IC1F=0000). Предделитель захвата не используется для запуска, поэтому пользователю не нужно его настраивать. Биты CC1S выбирают только источник захвата ввода, CC1S=01 в регистре TIMx_CCMR1. Запишите CC1P=1 в регистр TIMx_CCER, чтобы проверить полярность (и определить только низкий уровень).
- Настройте таймер в режиме стробирования, написав SMS=101 в регистре TIMx_SMCR. Выберите TI1 в качестве источника ввода, записав TS=101 в регистр TIMx_SMCR.
- Включите счетчик, записав CEN=1 в регистр TIMx_CR1 (в стробируемом режиме счетчик не запускается, если CEN=0, независимо от входного уровня триггера).

Счетчик начинает считать по внутренним часам, пока TI1 имеет низкий уровень, и останавливается, как только TI1 становится высоким. Флаг TIF в регистре TIMx_SR устанавливается как при запуске, так и при остановке счетчика.

Задержка между нарастающим фронтом на TI1 и фактической остановкой счетчика связана со схемой ресинхронизации на входе TI1.

Рисунок 137. Схема управления в стробируемом режиме

**Ведомый режим: режим триггера**

Счетчик может запускаться в ответ на событие на выбранном входе.

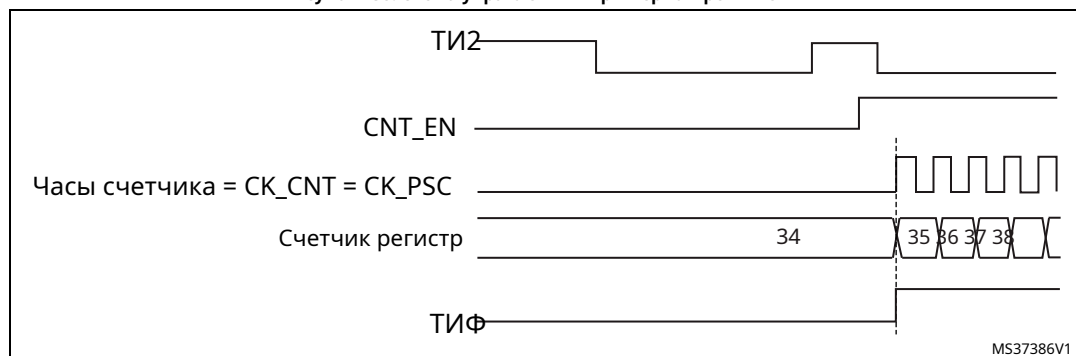
В следующем примере обратный счетчик запускается в ответ на нарастающий фронт на входе TI2:

- Настройте канал 2 для обнаружения нарастающих фронтов на TI2. Настройте продолжительность входного фильтра (в этом примере нам не нужен фильтр, поэтому мы сохраняем IC2F=0000). Предделитель захвата не используется для запуска, поэтому пользователю не нужно его настраивать. Биты CC2S выбирают только источник захвата ввода, CC2S=01 в регистре TIMx_CCMR1. Запишите CC2P=1 в регистр TIMx_CCER, чтобы проверить полярность (и определить только низкий уровень).
- Настройте таймер в режиме триггера, написав SMS=110 в регистре TIMx_SMCR. Выберите TI2 в качестве источника ввода, записав TS=110 в регистр TIMx_SMCR.

Когда на TI2 возникает нарастающий фронт, счетчик начинает считать внутренние часы и устанавливается флаг TIF.

Задержка между передним фронтом на TI2 и фактическим запуском счетчика обусловлена схемой ресинхронизации на входе TI2.

Рисунок 138. Схема управления в триггерном режиме



MS37386V1

Ведомый режим: режим внешних часов 2 + режим триггера

Режим внешних часов 2 можно использовать в дополнение к другому ведомому режиму (кроме режима внешних часов 1 и режима энкодера). В этом случае сигнал ETR используется в качестве внешнего тактового входа, а другой вход может быть выбран в качестве входа триггера при работе в режиме сброса, стробированном режиме или режиме триггера. Не рекомендуется выбирать ETR как TRGI через биты TS регистра TIMx_SMCR.

В следующем примере счетчик увеличивается при каждом переднем фронте сигнала ETR, как только возникает передний фронт TI1:

1. Настройте входную цепь внешнего триггера, запрограммировав регистр TIMx_SMCR следующим образом:

- ETF = 0000: без фильтра
- ETPS = 00: предварительный делитель отключен
- ETR = 0: обнаружение нарастающих фронтов на ETR и ECE=1 для включения режима внешней синхронизации 2.

2. Настройте канал 1 следующим образом, чтобы обнаруживать нарастающие фронты на TI1:

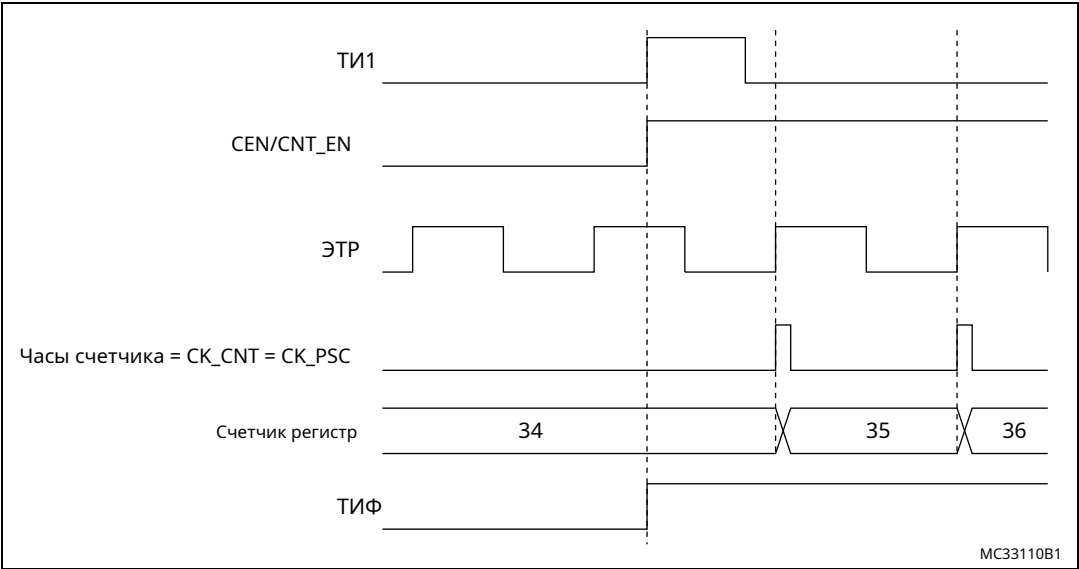
- IC1F = 0000: без фильтра.
- Предварительный делитель захвата не используется для запуска и не требует настройки.
- CC1S = 01 в регистре TIMx_CCMR1 для выбора только источника захвата ввода
- CC1P = 0 в регистре TIMx_CCER для проверки полярности (и обнаружения только нарастающего фронта).

3. Настройте таймер в режиме триггера, написав SMS=110 в регистре TIMx_SMCR. Выберите TI1 в качестве источника ввода, записав TS=101 в регистр TIMx_SMCR.

Нарастающий фронт на TI1 включает счетчик и устанавливает флаг TIF. Затем счетчик считает нарастающие фронты ETR.

Задержка между нарастающим фронтом сигнала ЭТР и фактическим сбросом счетчика связана со схемой ресинхронизации на входе ЭТР.

Рисунок 139. Схема управления в режиме внешнего тактирования 2 + режим триггера



15.3.15 Синхронизация таймера

Таймеры TIMx внутренне связаны друг с другом для синхронизации или объединения таймеров. Когда один таймер настроен в режиме Master, он может сбрасывать, запускать, останавливать или синхронизировать счетчик другого таймера, настроенного в режиме Slave.

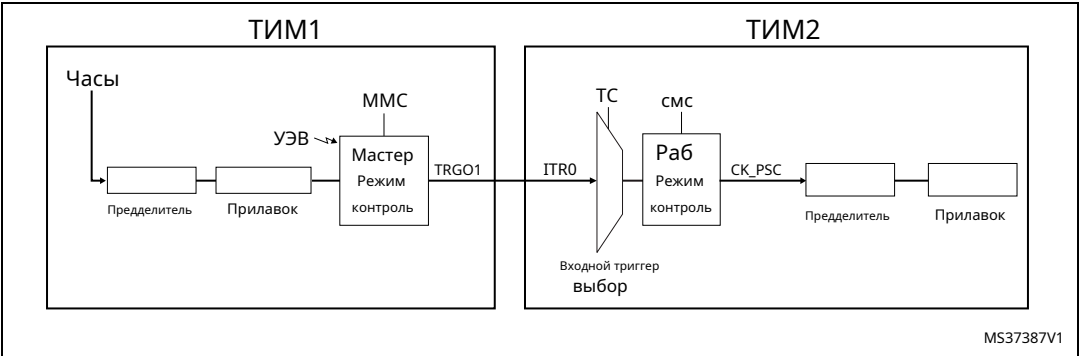
Рисунок 140 представляет обзор выбора триггера и блоков выбора главного режима.

Примечание:

Часы подчиненного таймера должны быть включены до получения событий от главного таймера и не должны изменяться на лету, пока триггеры принимаются от главного таймера.

Использование одного таймера в качестве предделителя для другого таймера

Рисунок 140. Пример таймера Master/Slave



Например, пользователь может настроить Таймер 1 так, чтобы он действовал как делитель для Таймера 2 (см. [Рисунок 140](#)). Сделать это:

- Настройте Таймер 1 в ведущем режиме, чтобы он выдавал периодический триггерный сигнал при каждом событии обновления UEV. Если вы запишете MMS=010 в регистр TIM1_CR2, передний фронт выводится на TRGO1 каждый раз, когда генерируется событие обновления.
- Чтобы подключить выход TRGO1 Таймера 1 к Таймеру 2, Таймер 2 должен быть сконфигурирован в ведомом режиме, используя ITR0 в качестве внутреннего триггера. Вы выбираете это через биты TS в регистре TIM2_SMCR (запись TS=000).
- Затем вы переводите контроллер ведомого режима в режим внешних часов 1 (запишете SMS=111 в регистр TIM2_SMCR). Это приводит к тактированию Таймера 2 по переднему фронту периодического запускающего сигнала Таймера 1 (что соответствует переполнению счетчика Таймера 1).
- Наконец, оба таймера должны быть включены путем установки соответствующих битов CEN (регистр TIMx_CR1).

Примечание:

Если ОСх выбран на Таймере 1 в качестве триггерного выхода (MMS=1xx), его нарастающий фронт используется для тактирования счетчика Таймера 2.

Использование одного таймера для включения другого таймера

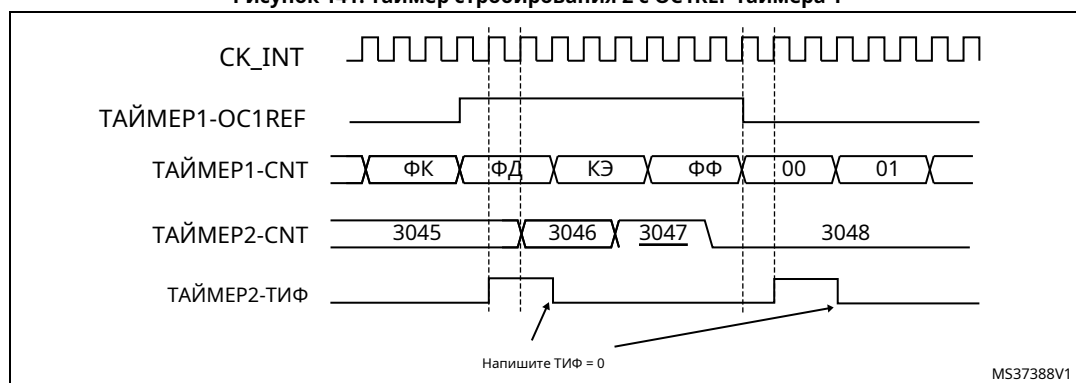
В этом примере мы управляем включением Таймера 2 с выходом сравнения 1 Таймера 1. См. [Рисунок 140](#) для соединений. Таймер 2 считает по разделенным внутренним часам только тогда, когда OC1REF Таймера 1 имеет высокий уровень. Тактовые частоты обоих счетчиков делятся на 3 предделителем по сравнению с CK_INT ($f_{CK_CNT} = f_{CK_INT}/3$).

- Настройте режим ведущего таймера 1 для отправки его опорного сигнала сравнения выхода 1 (OC1REF) в качестве триггерного выхода (MMS=100 в регистре TIM1_CR2).
- Настройте сигнал Таймера 1 OC1REF (регистр TIM1_CCMR1).
- Настройте Таймер 2 на получение входного триггера от Таймера 1 (TS=000 в регистре TIM2_SMCR).
- Настройте Таймер 2 в режиме стробирования (SMS=101 в регистре TIM2_SMCR). Включите Таймер 2, записав '1' в бит CEN (регистр TIM2_CR1).
- Запустите Таймер 1, записав '1' в бит CEN (регистр TIM1_CR1).

Примечание:

Часы счетчика 2 не синхронизированы со счетчиком 1, этот режим влияет только на сигнал включения счетчика Таймера 2.

Рисунок 141. Таймер стробирования 2 с OC1REF таймера 1



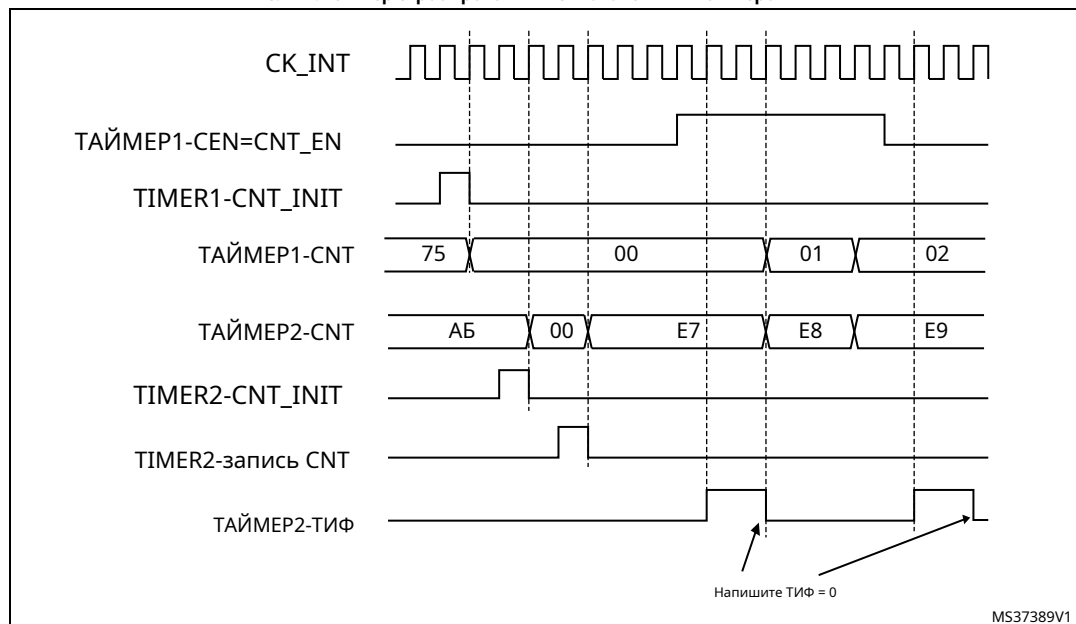
В примере в [Рисунок 141](#), счетчик Таймера 2 и предварительный делитель не инициализируются перед запуском. Таким образом, они начинают считать с их текущего значения. Можно начать с заданного значения, сбросив оба таймера перед запуском Таймера 1. Затем вы можете записать любое значение

вы хотите в счетчиках таймера. Таймеры можно легко сбросить с помощью программного обеспечения, используя бит UG в регистрах TIMx_EGR.

В следующем примере мы синхронизируем Таймер 1 и Таймер 2. Таймер 1 является ведущим и запускается с 0. Таймер 2 является подчиненным и запускается с 0xE7. Коэффициент делителя одинаков для обоих таймеров. Таймер 2 останавливается, когда Таймер 1 отключается путем записи '0' в бит CEN в регистре TIM1_CR1:

- Настройте режим ведущего таймера 1 для отправки его опорного сигнала сравнения выхода 1 (OC1REF) в качестве триггерного выхода (MMS=100 в регистре TIM1_CR2).
- Настройте сигнал Таймера 1 OC1REF (регистр TIM1_CCMR1).
- Настройте Таймер 2 на получение входного триггера от Таймера 1 (TS=000 в регистре TIM2_SMCR).
- Настройте Таймер 2 в режиме стробирования (SMS=101 в регистре TIM2_SMCR). Сбросьте Таймер 1, записав '1' в бит UG (регистр TIM1_EGR).
- Сбросьте Таймер 2, записав '1' в бит UG (регистр TIM2_EGR).
- Инициализируйте Таймер 2 значением 0xE7, записав '0xE7' в счетчик Таймера 2 (TIM2_CNTL). Включите Таймер 2, записав '1' в бит CEN (регистр TIM2_CR1).
- Запустите Таймер 1, записав '1' в бит CEN (регистр TIM1_CR1).
- Остановите Таймер 1, записав '0' в бит CEN (регистр TIM1_CR1).

Рис. 142. Таймер стробирования 2 с включенным таймером 1

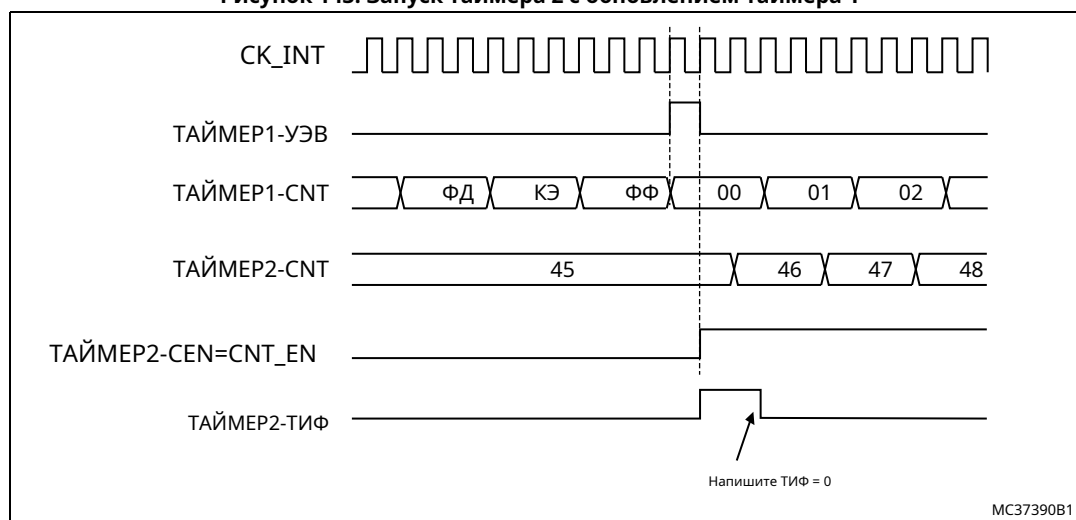


Использование одного таймера для запуска другого таймера

В этом примере мы устанавливаем включение Таймера 2 с событием обновления Таймера 1. См. [Рисунок 140](#) для соединений. Таймер 2 начинает отсчет со своего текущего значения (которое может быть ненулевым) на разделенных внутренних часах, как только событие обновления генерируется Таймером 1. Когда Таймер 2 получает триггерный сигнал, его бит CEN автоматически устанавливается, и счетчик считает до тех пор, пока мы запишем '0 в бит CEN в регистре TIM2_CR1. Тактовые частоты обоих счетчиков делятся на 3 предделителем по сравнению с CK_INT ($f_{CK_CNT} = f_{CK_INT}/3$).

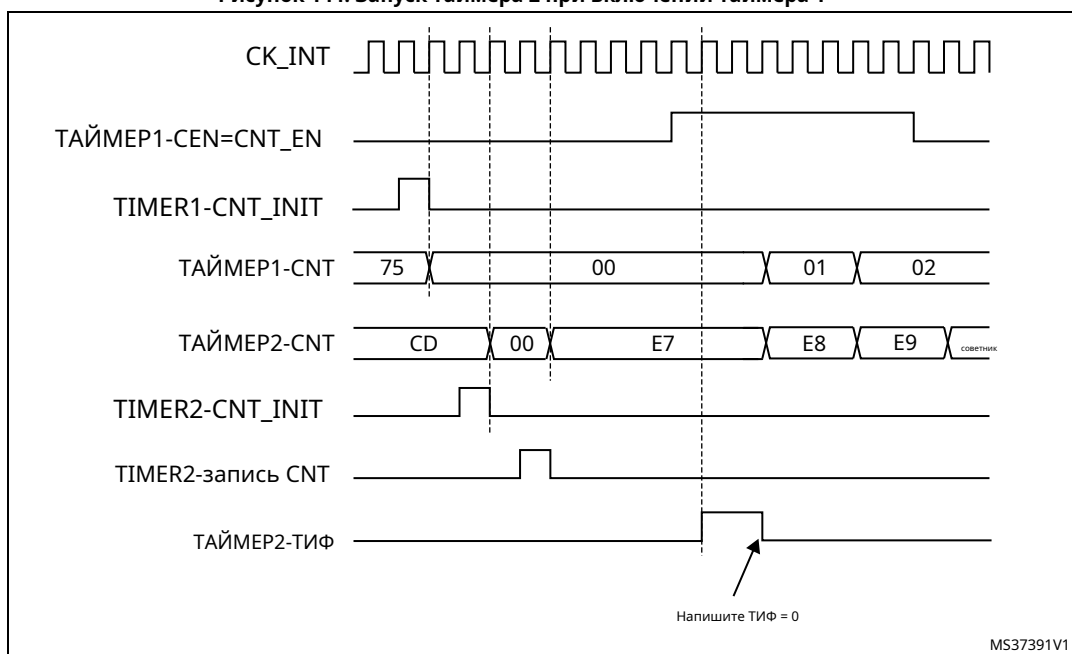
- Настройте главный режим Таймера 1 для отправки его события обновления (UEV) в качестве выхода триггера (MMS=010 в регистре TIM1_CR2).
- Настройте период Таймера 1 (регистры TIM1_ARR).
- Настройте Таймер 2 на получение входного триггера от Таймера 1 (TS=000 в регистре TIM2_SMCR).
- Настройте Таймер 2 в режиме триггера (SMS=110 в регистре TIM2_SMCR).
- Запустите Таймер 1, записав '1 в бит CEN (регистр TIM1_CR1).

Рисунок 143. Запуск таймера 2 с обновлением таймера 1



Как и в предыдущем примере, пользователь может инициализировать оба счетчика перед началом подсчета. [Рисунок 144](#) показывает поведение с той же конфигурацией, что и в [Рисунок 143](#) но в режиме триггера вместо режима стробирования (SMS=110 в регистре TIM2_SMCR).

Рисунок 144. Запуск таймера 2 при включении таймера 1



Синхронный запуск 2 таймеров в ответ на внешний триггер

В этом примере мы устанавливаем включение таймера 1, когда его вход TI1 поднимается, и включение Таймера 2 при включении Таймера 1. См. [Рисунок 140](#) для соединений. Чтобы убедиться, что счетчики выровнены, Таймер 1 должен быть настроен в режиме Master/Slave (ведомый по отношению к TI1, ведущий по отношению к Таймеру 2):

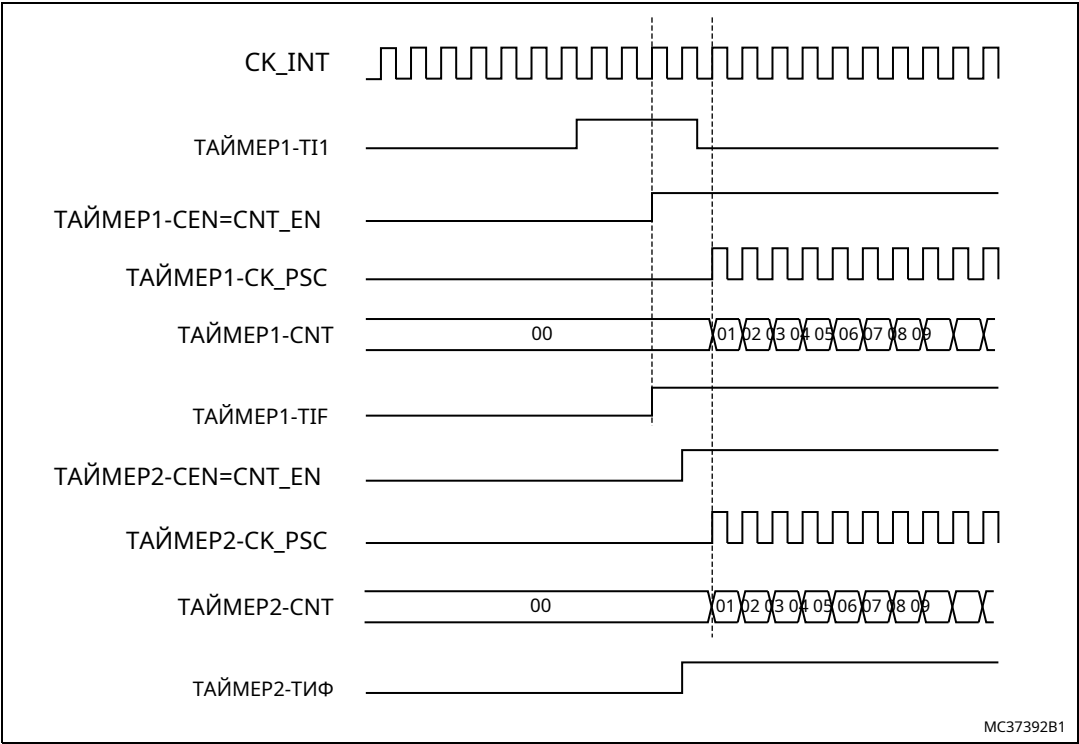
- Настройте режим ведущего Таймера 1 для отправки его разрешения в качестве выхода триггера (MMS=001 в регистре TIM1_CR2).
- Настройте подчиненный режим Таймера 1, чтобы получить входной триггер от TI1 (TS=100 в регистре TIM1_SMCR).
- Настройте Таймер 1 в режиме триггера (SMS=110 в регистре TIM1_SMCR). Настройте Таймер 1 в режиме Master/Slave, написав MSM=1 (регистр TIM1_SMCR).
- Настройте Таймер 2 на получение входного триггера от Таймера 1 (TS=000 в регистре TIM2_SMCR).
- Настройте Таймер 2 в режиме триггера (SMS=110 в регистре TIM2_SMCR).

Когда на TI1 (Таймер 1) возникает нарастающий фронт, оба счетчика начинают синхронный отсчет по внутренним часам, и устанавливаются оба флага TIF.

Примечание:

В этом примере оба таймера инициализируются перед запуском (путем установки соответствующих битов UG). Оба счетчика начинаются с 0, но вы можете легко вставить смещение между ними, записав любой из регистров счетчика (TIMx_CNT). Вы можете видеть, что режим ведущий/ведомый вставляет задержку между CNT_EN и CK_PSC на таймере 1.

Рисунок 145. Запуск таймера 1 и 2 с помощью входа TI1 таймера 1



15.3.16 Режим отладки

Когда микроконтроллер входит в режим отладки (Cortex®-M3 core - остановлен), счетчик TIMx либо продолжает нормально работать, либо останавливается, в зависимости от бита конфигурации DBG_TIMx_STOP в модуле DBGMCU. Для получения более подробной информации см. [Раздел 31.16.2: Поддержка отладки для таймеров, сторожевого таймера, bxCAN и I2C](#).

15,4 регистры TIMx

Ссылаться на [Раздел 2.2](#) список сокращений, используемых в описаниях регистров.

32-битные периферийные регистры должны быть записаны словами (32 бита). Все остальные периферийные регистры должны быть записаны полусловами (16 бит) или словами (32 бита). Доступ для чтения может осуществляться по байтам (8 бит), полусловам (16 бит) или словам (32 бит).

15.4.1 Регистр управления TIMx 1 (TIMx_CR1)

Смещение адреса: 0x00

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный						ХПН[1:0]		АРПЕ	CMS		ДИР	ОПМ	УРС	ЮДИС	CEN
						RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 15:10 Зарезервировано, должно быть сохранено значение сброса.

Биты 9:8 **ХБП**: деление часов

Это битовое поле указывает коэффициент деления между частотой таймера (CK_INT) и частотой дискретизации, используемой цифровыми фильтрами (ETR, TIX).

00: $TDTS = TCK_INT$
01: $TDTS = 2 \times TCK_INT$
10: $TDTS = 4 \times TCK_INT$
11: зарезервировано

Бит 7 **АРПЕ**: Включить автоматическую предварительную загрузку

0: регистр TIMx_ARR не буферизуется
1: регистр TIMx_ARR буферизуется

Биты 6:5 **CMS**: Выбор режима с выравниванием по центру

00: Режим с выравниванием по краю. Счетчик считает вверх или вниз в зависимости от бита направления (DIR).

01: Режим с выравниванием по центру 1. Счетчик попеременно считает вверх и вниз. Флаги прерывания сравнения выходов каналов, сконфигурированных на выходе (CCxS=00 в регистре TIMx_CCMRx), устанавливаются только тогда, когда счетчик ведет обратный отсчет.

10: Режим с выравниванием по центру 2. Счетчик поочередно считает вверх и вниз. Флаги прерывания сравнения выходов каналов, сконфигурированных на выходе (CCxS=00 в регистре TIMx_CCMRx), устанавливаются только тогда, когда счетчик ведет обратный отсчет.

11: Режим с выравниванием по центру 3. Счетчик попеременно считает вверх и вниз. Флаги прерывания сравнения выходов каналов, сконфигурированных на выходе (CCxS=00 в регистре TIMx_CCMRx), устанавливаются как при прямом, так и при обратном счете.

Примечание: Не допускается переключение из режима выравнивания по краю в режим выравнивания по центру, пока включен счетчик (CEN=1).

Бит 4 **ДИР**: Направление

0: Счетчик используется как прямой счетчик 1:
Счетчик используется как обратный счетчик

Примечание: Этот бит считывается только в том случае, если таймер настроен в режиме выравнивания по центру или в режиме энкодера.

Бит 3 **ОПМ**: Одноимпульсный режим

0: счетчик не останавливается при обновлении
1: Счетчик прекращает считать при следующем событии обновления (сброс бита CEN)



Бит 2 УРС: Обновить источник запроса

Этот бит устанавливается и сбрасывается программным обеспечением для выбора источников событий UEV.

0: Любое из следующих событий генерирует прерывание обновления или запрос DMA, если разрешено.

Эти события могут быть:

- Счетчик переполнения/недополнения
- Установка бита UG
- Генерация обновлений через контроллер ведомого режима

1: Только переполнение/опустошение счетчика генерирует прерывание обновления или запрос DMA, если включено.

Бит 1 юдис: обновление отключить

Этот бит устанавливается и сбрасывается программным обеспечением для включения/отключения генерации событий UEV. 0: UEV включен. Событие Update (UEV) генерируется одним из следующих событий:

- Счетчик переполнения/недополнения
- Установка бита UG
- Генерация обновлений через контроллер ведомого режима

Затем буферизованные регистры загружаются с их значениями предварительной загрузки.

1: UEV отключен. Событие Update не генерируется, теневые регистры сохраняют свое значение (ARR, PSC, CCRx). Однако счетчик и делитель повторно инициализируются, если установлен бит UG или если получен аппаратный сброс от контроллера ведомого режима.

Бит 0 CEN: Включение счетчика

0: Счетчик отключен

1: Счетчик включен

Примечание: Внешние часы, режим стробирования и режим энкодера могут работать только в том случае, если бит CEN был предварительно установлен программно. Однако режим триггера может аппаратно устанавливать бит CEN автоматически.

CEN очищается автоматически в одноимпульсном режиме, когда происходит событие обновления.

15.4.2 Регистр управления 2 TIMx (TIMx_CR2)

Смещение адреса: 0x04

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный								TI1C	MMC[2:0]			CCDS	Сдержанный		
								RW	RW	RW	RW	RW			

Биты 15:8 Зарезервировано, должно быть сохранено значение сброса.

Бит 7 **TI1C**: выбор TI1

- 0: вывод TIMx_CH1 подключен к входу TI1.
- 1: Контакты TIMx_CH1, CH2 и CH3 подключены к входу TI1 (комбинация XOR). См. также [Раздел 14.3.18: Взаимодействие с датчиками Холла](#)

Биты 6:4 **MMC[2:0]**: Выбор основного режима

Эти биты позволяют выбрать информацию для отправки в ведущем режиме подчиненным таймерам для синхронизации (TRGO). Комбинация выглядит следующим образом:

000:**Перезагрузить**-бит UG из регистра TIMx_EGR используется как выход триггера (TRGO). Если сброс генерируется триггерным входом (контроллер ведомого режима настроен на режим сброса), то сигнал на TRGO задерживается по сравнению с фактическим сбросом.

001:**включить**-сигнал включения счетчика, CNT_EN, используется в качестве триггерного выхода (TRGO). Полезно запускать несколько таймеров одновременно или управлять окном, в котором включен подчиненный таймер. Сигнал включения счетчика генерируется логическим ИЛИ между управляющим битом CEN и входом триггера, когда он сконфигурирован в стробированном режиме.

Когда сигнал разрешения счетчика управляется триггерным входом, на TRGO есть задержка, за исключением случаев, когда выбран режим ведущий/ведомый (см. описание бита MSM в регистре TIMx_SMCR).

010:**Обновлять**-Событие обновления выбирается как выход триггера (TRGO). Например, главный таймер можно использовать в качестве предделителя для подчиненного таймера.

011:**Сравнить пульс**-Выход триггера посылает положительный импульс, когда должен быть установлен флаг CC1IF (даже если он уже был высоким), как только произошло совпадение захвата или сравнения. (TRGO)

- 100:**Сравнить**-Сигнал OC1REF используется как триггерный выход (TRGO) 101:**Сравнить**-Сигнал OC2REF используется как триггерный выход (TRGO) 110:**Сравнить**-Сигнал OC3REF используется как триггерный выход (TRGO) 111:**Сравнить**-Сигнал OC4REF используется как триггерный выход (TRGO).

Примечание: Часы подчиненного таймера и АЦП должны быть включены до получения событий от главного таймера и не должны изменяться «на лету», пока триггеры принимаются от главного таймера.

Бит 3 **CCDS**: захват/сравнение выбора прямого доступа к памяти

- 0: Запрос CCx DMA отправляется, когда происходит событие CCx 1: Запросы CCx DMA отправляются, когда происходит событие обновления

Биты 2:0 Зарезервировано, должно быть сохранено значение сброса.



15.4.3 Регистр управления подчиненным режимом TIMx (TIMx_SMCR)

Смещение адреса: 0x08

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ЭТП	ЕЭК	ЭТПС[1:0]		ЕТФ[3:0]				MCM	ТС[2:0]			Рез.	СМС[2:0]		
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW	RW

Бит 15 **ЭТП**: Полярность внешнего триггера

Этот бит определяет, используется ли ETR или ETRP для операций запуска. 0: ETR не инвертируется, активен при высоком уровне или нарастающем фронте.

1: ETR инвертирован, активен при низком уровне или заднем фронте

Бит 14 **ЕЭК**: Включение внешних часов

Этот бит включает режим внешних часов 2.

0: режим внешних часов 2 отключен.

1: режим внешних часов 2 включен. Счетчик тактируется любым активным фронтом сигнала ETRF.

1: Установка бита ECE имеет тот же эффект, что и выбор режима внешних часов 1 с TRGI, подключенным к ETRF (SMS=111 и TS=111).

2: Можно одновременно использовать режим внешних часов 2 со следующими подчиненными режимами: режим сброса, режим стробирования и режим триггера. Тем не менее, в этом случае TRGI не должен быть подключен к ETRF (биты TS не должны быть 111).

3: Если режим внешних часов 1 и режим внешних часов 2 включены одновременно, то входом внешних часов является ETRF.

Биты 13:12 **ЭТП**: Предделитель внешнего триггера

Частота внешнего триггерного сигнала ETRP должна быть не более 1/4 частоты CK_INT. Можно включить предварительный делитель для уменьшения частоты ETRP. Это полезно при вводе быстрых внешних часов.

00: Предделитель ВЫКЛ.

01: частота ETRP, разделенная на 2

10: частота ETRP, разделенная на 4

11: частота ETRP, разделенная на 8

Биты 11:8 **ЕТФ[3:0]**: Внешний триггерный фильтр

Затем это битовое поле определяет частоту, используемую для выборки сигнала ETRP, и длину цифрового фильтра, применяемого к ETRP. Цифровой фильтр состоит из счетчика событий, в котором требуется N последовательных событий для проверки перехода на выходе:

0000: Нет фильтра, выборка выполняется в f_{ДТС}

0001: фотБОР ПРОБ=f_{СК_INT}, N=2 ф

0010: отБОР ПРОБ=f_{СК_INT}, N=4 ф

0011: отБОР ПРОБ=f_{СК_INT}, N=8² ф

0100: отБОР ПРОБ=f_{ДТС}/2, N=6 ж

0101: отБОР ПРОБ=f_{ДТС}/2, N=8 ж

0110: отБОР ПРОБ=f_{ДТС}/4, N=6² ф

0111: отБОР ПРОБ=f_{ДТС}/4, N=8

1000: фотБОР ПРОБ=f_{ДТС}/8, N=6

1001: жотБОР ПРОБ=f_{ДТС}/8, N=8

1010: фотБОР ПРОБ=f_{ДТС}/16, N=5

1011: жотБОР ПРОБ=f_{ДТС}/16, N=6

1100: фотБОР ПРОБ=f_{ДТС}/16, N=8

1101: фотБОР ПРОБ=f_{ДТС}/32, N=5

1110: фотБОР ПРОБ=f_{ДТС}/32, N=6

1111: жотБОР ПРОБ=f_{ДТС}/32, N=8

Бит 7 **МСМ:**Ведущий/ведомый режим

0: никаких действий

1: Воздействие события на триггерный вход (TRGI) задерживается, чтобы обеспечить идеальную синхронизацию между текущим таймером и его ведомыми устройствами (через TRGO). Это полезно, если мы хотим синхронизировать несколько таймеров по одному внешнему событию.

Биты 6:4 **ТС:**Выбор триггера

Это битовое поле выбирает триггерный вход, который будет использоваться для синхронизации счетчика.

000: Внутренний триггер 0 (ITR0).

001: Внутренний триггер 1 (ITR1). 010:

Внутренний триггер 2 (ITR2). 011:

Внутренний триггер 3 (ITR3). 100: Детектор

фронта TI1 (TI1F_ED) 101: Фильтрованный

вход таймера 1 (TI1FP1) 110: Фильтрованный

вход таймера 2 (TI2FP2) 111: Вход внешнего

триггера (ETRF)

Видеть [Таблица 86: Подключение внутреннего триггера TIMx](#) для более подробной информации о значении ITRx для каждого таймера.

Примечание: Эти биты следует изменять только тогда, когда они не используются (например, когда SMS=000), чтобы избежать неправильного обнаружения фронта при переходе.

Бит 3 Зарезервировано, должно быть сохранено значение сброса.

Биты 2:0 **СМС:**Выбор ведомого режима

Когда выбраны внешние сигналы, активный фронт триггерного сигнала (TRGI) связан с полярностью, выбранной на внешнем входе (см. описание регистра управления входом и регистра управления).

000: Ведомый режим отключен — если CEN = '1, то предварительный делитель тактируется непосредственно внутренними часами.

001: Режим энкодера 1. Счетчик ведет обратный отсчет по фронту TI2FP1 в зависимости от уровня TI1FP2.

010: Режим энкодера 2. Счетчик ведет обратный отсчет по фронту TI1FP2 в зависимости от уровня TI2FP1.

011: Режим энкодера 3. Счетчик ведет обратный/прямой счет на фронтах TI1FP1 и TI2FP2 в зависимости от уровня другого входа.

100: Режим сброса. Нарастающий фронт выбранного триггерного входа (TRGI) повторно инициализирует счетчик и генерирует обновление регистров.

101: Режим стробирования. Часы счетчика включаются, когда на входе триггера (TRGI) высокий уровень. Счетчик останавливается (но не сбрасывается), как только триггер становится низким. И запуск, и остановка счетчика контролируются.

110: Режим триггера. Счетчик запускается по переднему фронту триггера TRGI (но не сбрасывается). Контролируется только начало счетчика.

111: Режим внешней синхронизации 1. Нарастающие фронты выбранного триггера (TRGI) синхронизируют счетчик.

Примечание: Режим стробирования нельзя использовать, если в качестве входа триггера выбран TI1F_ED (TS=100). Действительно, TI1F_ED выдает 1 импульс для каждого перехода на TI1F, тогда как стробируемый режим проверяет уровень триггерного сигнала.

Часы подчиненного таймера должны быть включены до получения событий от главного таймера и не должны изменяться на лету, пока триггеры принимаются от главного таймера.

Табл. 86. Подключение внутреннего триггера TIMx(1)

Раb Тим	ITR0 (TC = 000)	ITR1 (TC = 001)	ITR2 (TC = 010)	ITR3 (TC = 011)
ТИМ2	ТИМ1	ТИМ8	ТИМ3	ТИМ4
ТИМ3	ТИМ1	ТИМ2	ТИМ5	ТИМ4
ТИМ4	ТИМ1	ТИМ2	ТИМ3	ТИМ8
ТИМ5	ТИМ2	ТИМ3	ТИМ4	ТИМ8

1. Если в продукте нет таймера, соответствующий триггер ITRx недоступен.

15.4.4 Регистр TIMx DMA/разрешения прерываний (TIMx_DIER)

Смещение адреса: 0x0C

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Рез.	ТДЭ	Рез	CC4DE	CC3DE	CC2DE	CC1DE	НЖЯ	Рез.	ГАЛСТУК	Рез	CC4IE	CC3IE	CC2IE	CC1IE	ИЮО
	RW		RW	RW	RW	RW	RW		RW		RW	RW	RW	RW	RW

Бит 15 Зарезервировано, должно быть сохранено значение сброса.

Бит 14 **ТДЭ**: активировать запрос DMA
 0: запуск запроса DMA отключен. 1:
 Запрос триггера DMA включен.

Бит 13 Зарезервировано, всегда читается как 0

Бит 12 **CC4DE**: Включение запроса Capture/Compare 4 DMA
 0: Запрос DMA CC4 отключен.
 1: Запрос DMA CC4 включен.

Бит 11 **CC3DE**: Включение запроса Capture/Compare 3 DMA
 0: Запрос DMA CC3 отключен.
 1: Запрос DMA CC3 включен.

Бит 10 **CC2DE**: Включение запроса Capture/Compare 2 DMA
 0: Запрос DMA CC2 отключен.
 1: Запрос DMA CC2 включен.

Бит 9 **CC1DE**: Включить захват/сравнение 1 запроса DMA
 0: Запрос DMA CC1 отключен.
 1: Запрос DMA CC1 включен.

Бит 8 **НЖЯ**: включить запрос обновления DMA
 0: запрос обновления DMA отключен. 1:
 запрос обновления DMA включен.

Бит 7 Зарезервировано, должно быть сохранено значение сброса.

Бит 6 **ГАЛСТУК**: Разрешить прерывание триггера
 0: Прерывание триггера отключено.
 1: прерывание триггера разрешено.

Бит 5 Зарезервировано, должно быть сохранено значение сброса.

- Бит 4 **CC4IE**: Разрешение прерывания захвата/сравнения 4
0: Прерывание CC4 отключено. 1:
прерывание CC4 разрешено.
- Бит 3 **CC3IE**: Разрешение прерывания захвата/сравнения 3
0: Прерывание CC3 отключено. 1:
прерывание CC3 разрешено.
- Бит 2 **CC2IE**: разрешение прерывания захвата/сравнения 2
0: Прерывание CC2 отключено. 1:
прерывание CC2 разрешено.
- Бит 1 **CC1IE**: Разрешение прерывания захвата/сравнения 1
0: Прерывание CC1 отключено. 1:
прерывание CC1 разрешено.
- Бит 0 **ИЮО**: разрешение прерывания обновления
0: Прерывание обновления отключено. 1:
прерывание обновления разрешено.

15.4.5 Регистр состояния TIMx (TIMx_SR)

Смещение адреса: 0x10

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный		CC4OF	CC3OF	CC2OF	CC1OF	Сдержанный		ТИФ	Рез	CC4IF	CC3IF	CC2IF	CC1IF	ПИФ	
		rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

- Бит 15:13 Зарезервировано, должно быть сохранено значение сброса.
- Бит 12 **CC4OF**: Захват/Сравнение 4 флага перехвата
см. описание CC1OF
- Бит 11 **CC3OF**: Захват/Сравнение 3 флажков перехвата
см. описание CC1OF
- Бит 10 **CC2OF**: захват/сравнение 2 флажков перехвата
см. описание CC1OF
- Бит 9 **CC1OF**: Захват/Сравнение 1 флаг перехвата
Этот флаг устанавливается аппаратно только тогда, когда соответствующий канал настроен в режиме захвата ввода. Он очищается программно, записывая его в «0».
0: перехват не обнаружен.
1: значение счетчика было зафиксировано в регистре TIMx_CCR1, хотя флаг CC1IF уже был установлен.
- Биты 8:7 Зарезервировано, должно быть сохранено значение сброса.
- Бит 6 **ТИФ**: флаг прерывания триггера
Этот флаг устанавливается аппаратно при событии триггера (активный фронт обнаружен на входе TRGI, когда контроллер ведомого режима включен во всех режимах, кроме стробируемого режима, оба фронта в случае выбора стробируемого режима). Очищается программно.
0: Триггерное событие не произошло.
1: Ожидание прерывания триггера.
- Бит 5 Зарезервировано, должно быть сохранено значение сброса.



- Бит 4 **CC4IF**: флаг прерывания захвата/сравнения 4
см. описание CC1IF
- Бит 3 **CC3IF**: флаг прерывания захвата/сравнения 3
см. описание CC1IF
- Бит 2 **CC2IF**: Флаг прерывания захвата/сравнения 2
см. описание CC1IF
- Бит 1 **CC1IF**: Захват/сравнение 1 флага прерывания
- Если канал CC1 сконфигурирован как выход:**
Этот флаг устанавливается аппаратно, когда значение счетчика совпадает со значением сравнения, за некоторыми исключениями в режиме с выравниванием по центру (см. биты CMS в описании регистра TIMx_CR1). Очищается программно.
- 0: Нет соответствия.
1: Содержимое счетчика TIMx_CNT соответствует содержимому регистра TIMx_CCR1.
- Если канал CC1 сконфигурирован как вход:**
Этот бит устанавливается аппаратно при захвате. Он очищается программно или чтением регистра TIMx_CCR1.
- 0: Захват ввода не производился.
1: значение счетчика было зафиксировано в регистре TIMx_CCR1 (обнаружен фронт на IC1, который соответствует выбранной полярности).
- Бит 0 **Пиф**: Обновить флаг прерывания
- Этот бит устанавливается аппаратно при событии обновления. Очищается программно. 0: обновление не произошло.
1: Ожидание прерывания обновления. Этот бит устанавливается аппаратно при обновлении регистров:
 - При переполнении или недостатке и если UDIS=0 в регистре TIMx_CR1.
 - Когда CNT повторно инициализируется программным обеспечением с использованием бита UG в регистре TIMx_EGR, если URS=0 и UDIS=0 в регистре TIMx_CR1.
 - Когда CNT повторно инициализируется триггерным событием (см. описание регистра управления синхронизацией), если URS=0 и UDIS=0 в регистре TIMx_CR1.

15.4.6 Регистр генерации событий TIMx (TIMx_EGR)

Смещение адреса: 0x14

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный									ТГ	Рез.	CC4G	CC3G	CC2G	CC1G	УГ
									Ж		Ж	Ж	Ж	Ж	Ж

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

- Бит 6 **ТГ**: Генерация триггера

Этот бит устанавливается программно для генерации события, он автоматически сбрасывается аппаратно.

0: никаких действий

1: Флаг TIF установлен в регистре TIMx_SR. Соответствующее прерывание или передача DMA может произойти, если разрешено.
- Бит 5 Зарезервировано, должно быть сохранено значение сброса.
- Бит 4 **CC4G**: Захват/сравнение 4 поколения

см. описание CC1G
- Бит 3 **CC3G**: Захват/сравнение 3 поколения

см. описание CC1G
- Бит 2 **CC2G**: Захват/сравните 2 поколения

см. описание CC1G
- Бит 1 **CC1G**: захват/сравнение 1 поколения

Этот бит устанавливается программно для генерации события, он автоматически сбрасывается аппаратно.

0: никаких действий

1: Событие захвата/сравнения генерируется на канале 1: **Если канал CC1 сконфигурирован как выход:**

Флаг CC1IF установлен, соответствующее прерывание или запрос DMA отправляется, если включен.

Если канал CC1 сконфигурирован как вход:

Текущее значение счетчика фиксируется в регистре TIMx_CCR1. Флаг CC1IF установлен, соответствующее прерывание или запрос DMA отправляется, если включен. Флаг CC1OF устанавливается, если флаг CC1IF уже был высоким.
- Бит 0 **УГ**: Генерация обновлений

Этот бит может быть установлен программно, он автоматически сбрасывается аппаратно. 0: никаких действий

1: Повторная инициализация счетчика и обновление регистров. Обратите внимание, что счетчик предварительного делителя также очищается (в любом случае коэффициент предварительного делителя не затрагивается). Счетчик очищается, если выбран режим выравнивания по центру или если DIR=0 (прямой счет), в противном случае он принимает значение автоматической перезагрузки (TIMx_ARR), если DIR=1 (нисходящий счет).



15.4.7 Регистр 1 режима захвата/сравнения TIMx (TIMx_CCMR1)

Смещение адреса: 0x18

Значение сброса: 0x0000

Каналы можно использовать на входе (режим захвата) или на выходе (режим сравнения). Направление канала определяется настройкой соответствующих битов CCxS. Все остальные биты этого регистра имеют разные функции в режиме ввода и вывода. Для данного бита OCxx описывает его функцию, когда канал сконфигурирован на выходе, ICxx описывает его функцию, когда канал сконфигурирован на входе. Позаботьтесь о том, чтобы один и тот же бит мог иметь разное значение для входного каскада и для выходного каскада.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]			IC1PSC[1:0]				
RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW		

Выходной режим сравнения

Бит 15 **OC2CE**: Выход сравнения 2 очистить включить

Биты 14:12 **OC2M[2:0]**: Режим сравнения выходов 2

Бит 11 **OC2PE**: Включение предварительной нагрузки сравнения вывода 2

Бит 10 **OC2FE**: Быстрое включение сравнения 2 выходов

Биты 9:8 **CC2S[1:0]**: Захват/Сравнить 2 выбора

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC2 настроен как выход

01: канал CC2 настроен как вход, IC2 отображается на TI2 10:

канал CC2 настроен как вход, IC2 отображается на TI1

11: Канал CC2 настроен как вход, IC2 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC2S доступны для записи только тогда, когда канал выключен (CC2E = 0 в TIMx_CCER).

Бит 7 **OC1CE**: Выход сравнить 1 очистить включить

OC1CE: сравнение выходов 1 Clear Enable

0: OC1Ref не зависит от входа ETRF.

1: OC1Ref сбрасывается, как только на входе ETRF обнаруживается высокий уровень.

Биты 6:4 **OC1M**: Режим сравнения выходов 1

Эти биты определяют поведение выходного опорного сигнала OC1REF, из которого получаются OC1 и OC1N. OC1REF имеет высокий активный уровень, тогда как активный уровень OC1 и OC1N зависит от битов CC1P и CC1NP.

000: Frozen — сравнение между выходным регистром сравнения TIMx_CCR1 и счетчиком TIMx_CNT не влияет на выходы (этот режим используется для генерации базы синхронизации).

001: Установите канал 1 на активный уровень при совпадении. Сигнал OC1REF принудительно устанавливается на высокий уровень, когда счетчик TIMx_CNT совпадает с регистром захвата/сравнения 1 (TIMx_CCR1).

010: Установить канал 1 на неактивный уровень при совпадении. Сигнал OC1REF принудительно устанавливается на низкий уровень, когда счетчик TIMx_CNT совпадает с регистром захвата/сравнения 1 (TIMx_CCR1).

011: Toggle — OC1REF переключается, когда TIMx_CNT=TIMx_CCR1. 100: Принудительно неактивный уровень — OC1REF принудительно устанавливается на низкий уровень.

101: Принудительный активный уровень — OC1REF принудительно устанавливается на высокий уровень.

110: ШИМ-режим 1 — при восходящем счете канал 1 активен до тех пор, пока TIMx_CNT < TIMx_CCR1, в противном случае неактивен. При обратном счете канал 1 неактивен (OC1REF=0), пока активен TIMx_CNT > TIMx_CCR1 (OC1REF=1).

111: ШИМ-режим 2. При восходящем счете канал 1 неактивен, пока активен TIMx_CNT < TIMx_CCR1. При обратном счете канал 1 активен до тех пор, пока TIMx_CNT > TIMx_CCR1, в противном случае неактивен.

Примечание: В режиме ШИМ 1 или 2 уровень OCREF изменяется только при изменении результата сравнения или при переключении режима сравнения выходов из «замороженного» режима в режим «ШИМ».

Бит 3 **OC1PE**: Выходное сравнение 1 включает предварительную нагрузку

0: регистр предварительной загрузки на TIMx_CCR1 отключен. TIMx_CCR1 можно записать в любое время, новое значение учитывается сразу.

1: регистр предварительной загрузки на TIMx_CCR1 включен. Операции чтения/записи обращаются к регистру предварительной загрузки.

Значение предварительной загрузки TIMx_CCR1 загружается в активный регистр при каждом событии обновления.

Примечание: 1: Эти биты не могут быть изменены, пока запрограммирован уровень LOCK 3 (биты LOCK в регистре TIMx_BDTR) и CC1S=00 (канал сконфигурирован на выходе).

2: Режим PWM можно использовать без проверки регистра предварительной загрузки только в одноимпульсном режиме (бит OPM установлен в регистре TIMx_CR1). В противном случае поведение не гарантируется.

Бит 2 **OC1FE**: Быстрое включение сравнения вывода 1

Этот бит используется для ускорения воздействия события на триггер на входе на выходе CC. 0: CC1 ведет себя нормально в зависимости от значений счетчика и CCR1, даже когда триггер включен. Минимальная задержка активации выхода CC1 при появлении фронта на входе триггера составляет 5 тактовых циклов.

1: Активный фронт на входе триггера действует как совпадение сравнения на выходе CC1. Затем OC устанавливается на уровень сравнения независимо от результата сравнения. Задержка для выборки триггерного входа и активации выхода CC1 уменьшена до 3 тактовых циклов. OCFE действует, только если канал сконфигурирован в режиме PWM1 или PWM2.

Биты 1:0 **CC1S**: Захватить/Сравнить 1 выбор

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: Канал CC1 настроен как выход.

01: канал CC1 настроен как вход, IC1 отображается на TI1. 10:

канал CC1 настроен как вход, IC1 отображается на TI2.

11: Канал CC1 настроен как вход, IC1 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC1S доступны для записи только тогда, когда канал выключен (CC1E = 0 в TIMx_CCER).

Входной режим захвата

Биты 15:12 **IC2F**: Входной фильтр захвата 2

Биты 11:10 **IC2PSC[1:0]**: Предварительный делитель входного захвата 2

Биты 9:8 **CC2S**: Захват/сравнение 2 выбранных

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: Канал CC2 настроен как выход.

01: канал CC2 настроен как вход, IC2 отображается на TI2. 10:

канал CC2 настроен как вход, IC2 отображается на TI1.

11: Канал CC2 настроен как вход, IC2 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC2S доступны для записи только тогда, когда канал выключен (CC2E = 0 в TIMx_CCER).

Биты 7:4 **IC1F**: Входной фильтр захвата 1

Это битовое поле определяет частоту, используемую для выборки входного сигнала TI1, и длину цифрового фильтра, применяемого к TI1. Цифровой фильтр состоит из счетчика событий, в котором требуется N последовательных событий для проверки перехода на выходе:

0000: Нет фильтра, выборка выполняется в f_{ДТС}

0001: ФОТБОР ПРОБ=f_{СК_И}NT, N=2

0010: ЖОТБОР ПРОБ=f_{СК_И}NT, N=4

0011: ЖОТБОР ПРОБ=f_{СК_И}NT, N=8

0100: ФОТБОР ПРОБ=f_{ДТС}/2, N=6

0101: ЖОТБОР ПРОБ=f_{ДТС}/2, N=8

0110: ФОТБОР ПРОБ=f_{ДТС}/4, N=6

0111: ЖОТБОР ПРОБ=f_{ДТС}/4, N=8

1000: ФОТБОР ПРОБ=f_{ДТС}/8, N=6

1001: ЖОТБОР ПРОБ=f_{ДТС}/8, N=8

1010: ЖОТБОР ПРОБ=f_{ДТС}/16, N=5

1011: ЖОТБОР ПРОБ=f_{ДТС}/16, N=6

1100: ЖОТБОР ПРОБ=f_{ДТС}/16, N=8

1101: ЖОТБОР ПРОБ=f_{ДТС}/32, N=5

1110: ЖОТБОР ПРОБ=f_{ДТС}/32, N=6

1111: ЖОТБОР ПРОБ=f_{ДТС}/32, N=8

Биты 3:2 **IC1PSC**: Входной захват 1 прескалера

Это битовое поле определяет отношение предварительного делителя, действующего на вход

CC1 (IC1). Предделитель сбрасывается, как только CC1E=0 (регистр TIMx_CCER).

00: без предварительного делителя, захват выполняется каждый раз при обнаружении фронта на входе захвата

01: захват выполняется каждые 2 события

10: захват выполняется один раз каждые 4 события

11: захват выполняется один раз каждые 8 событий

Биты 1:0 **CC1S**: Захватить/Сравнить 1 выбор

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC1 настроен как выход

01: канал CC1 настроен как вход, IC1 отображается на TI1 10:

канал CC1 настроен как вход, IC1 отображается на TI2

11: Канал CC1 настроен как вход, IC1 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC1S доступны для записи только тогда, когда канал выключен (CC1E = 0 в TIMx_CCER).

15.4.8 Регистр 2 режима захвата/сравнения TIMx (TIMx_CCMR2)

Смещение адреса: 0x1C

Значение сброса: 0x0000

Обратитесь к приведенному выше описанию регистра CCMR1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]			IC3PSC[1:0]				
RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW		

Выходной режим сравненияБит 15 **OC4CE**: Сравнение выходов 4 очистить разрешениеБиты 14:12 **OC4M**: Режим сравнения выходов 4Бит 11 **OC4PE**: Выход сравнения 4 разрешения предварительной нагрузкиБит 10 **OC4FE**: Сравнение выходов 4 быстрого включенияБиты 9:8 **CC4S**: Захват/Сравнение 4 вариантов

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC4 настроен как выход, IC4 отображается на TI4 10:

01: канал CC4 настроен как вход, IC4 отображается на TI3

канал CC4 настроен как вход, IC4 отображается на TI3

11: Канал CC4 настроен как вход, IC4 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

*Примечание. Биты CC4S доступны для записи только тогда, когда канал выключен (CC4E = 0 в TIMx_CCER).*Бит 7 **OC3CE**: Выход сравнить 3 ясно включитьБиты 6:4 **OC3M**: Режим сравнения выходов 3Бит 3 **OC3PE**: Выход сравнения 3 разрешения предварительной нагрузкиБит 2 **OC3FE**: Сравнение выходов 3, быстрое включениеБиты 1:0 **CC3S**: Захват/Сравнить 3 варианта

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC3 настроен как выход

01: канал CC3 настроен как вход, IC3 отображается на TI3 10:

канал CC3 настроен как вход, IC3 отображается на TI4

11: Канал CC3 настроен как вход, IC3 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC3S доступны для записи только тогда, когда канал выключен (CC3E = 0 в TIMx_CCER).

Входной режим захвата

Биты 15:12 **IC4F**: Входной захват 4 фильтр

Биты 11:10 **IC4PSC**: Предварительный делитель входного захвата 4

Биты 9:8 **CC4S**: Захват/Сравнение 4 вариантов

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC4 настроен как выход

01: канал CC4 настроен как вход, IC4 отображается на TI4 10:

канал CC4 настроен как вход, IC4 отображается на TI3

11: Канал CC4 настроен как вход, IC4 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC4S доступны для записи только тогда, когда канал выключен (CC4E = 0 в TIMx_CCER).

Биты 7:4 **IC3F**: Входной фильтр захвата 3

Биты 3:2 **IC3PSC**: Предварительный делитель входного захвата 3

Биты 1:0 **CC3S**: Захват/Сравнить 3 варианта

Это битовое поле определяет направление канала (вход/выход), а также используемый вход.

00: канал CC3 настроен как выход

01: канал CC3 настроен как вход, IC3 отображается на TI3 10:

канал CC3 настроен как вход, IC3 отображается на TI4

11: Канал CC3 настроен как вход, IC3 отображается на TRC. Этот режим работает, только если через бит TS (регистр TIMx_SMCR) выбран внутренний триггерный вход.

Примечание. Биты CC3S доступны для записи только тогда, когда канал выключен (CC3E = 0 в TIMx_CCER).

15.4.9 Регистр разрешения захвата/сравнения TIMx (TIMx_CCER)

Смещение адреса: 0x20

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный	CC4P	CC4E	Сдержанный	CC3P	CC3E	Сдержанный	CC2P	CC2E	Сдержанный	CC1P	CC1E				
	RW	RW		RW	RW		RW	RW		RW					

Биты 15:14 Зарезервировано, должно быть сохранено значение сброса.

Бит 13 **CC4P**: Захват/Сравнение полярности 4 выходов
см. описание CC1P

Бит 12 **CC4E**: Захват/Сравнение 4 выходных данных включен
см. описание CC1E

Биты 11:10 Зарезервировано, должно быть сохранено значение сброса.

Бит 9 **CC3P**: захват/сравнение полярности 3 выходов
см. описание CC1P

Бит 8 **CC3E**: Захват/Сравнение 3 выходных данных включен
см. описание CC1E

Биты 7:6 Зарезервировано, должно быть сохранено значение сброса.

Бит 5 **CC2P**: Захват/Сравнение полярности 2 выходов
см. описание CC1P

Бит 4 **CC2E**: Захват/Сравнение 2 выходов включен
см. описание CC1E

Биты 3:2 Зарезервировано, должно быть сохранено значение сброса.

Бит 1 **CC1P**: захват/сравнение 1 выходной полярности

Канал CC1 настроен как выход: 0: OC1 активен на высоком уровне. 1: OC1 активен на низком уровне.

Канал CC1 настроен как вход:

Этот бит выбирает, используется ли IC1 или IC1 для операций запуска или захвата.

0: неинвертированный: захват выполняется по переднему фронту IC1. При использовании в качестве внешнего триггера IC1 не инвертируется.

1: инвертированный: захват выполняется по заднему фронту IC1. При использовании в качестве внешнего триггера IC1 инвертируется.

Бит 0 **CC1E**: Включение захвата/сравнения 1 выхода

Канал CC1 настроен как выход: 0:

Выкл. — OC1 не активен.

1: On — сигнал OC1 выводится на соответствующий выходной контакт.

Канал CC1 настроен как вход:

Этот бит определяет, действительно ли можно выполнить захват значения счетчика во входной регистр захвата/сравнения 1 (TIMx_CCR1) или нет.

0: Захват отключен.

1: Захват включен.

Таблица 87. Бит управления выходом для стандартных каналов OCx

Бит CCxE	Состояние выхода OCx
0	Выход отключен (OCx=0, OCx_EN=0)
1	OCx=OCxREF + Полярность, OCx_EN=1

Примечание:

Состояние внешних контактов ввода-вывода, подключенных к стандартным каналам OCx, зависит от состояния канала OCx и регистров GPIO и AFIO.

15.4.10 Счетчик TIMx (TIMx_CNT)

Смещение адреса: 0x24

Значение сброса: 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HKT[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 15:0HKT[15:0]: Значение счетчика

15.4.11 Предделитель TIMx (TIMx_PSC)

Смещение адреса: 0x28

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 15:0PSC[15:0]: Значение предварительного делителя

Тактовая частота счетчика CK_CNT равна $f_{CK_PSC} / (PSC[15:0] + 1)$.

PSC содержит значение, которое будет загружено в активный регистр предварительного делителя при каждом событии обновления (в том числе, когда счетчик очищается через бит UG регистра TIMx_EGR или через контроллер триггера, когда он сконфигурирован в «режиме сброса»).

15.4.12 Регистр автоматической перезагрузки TIMx (TIMx_ARR)

Смещение адреса: 0x2C

Значение сброса: 0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
APP[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 15:0

APP[15:0]: Значение предварительного делителя

ARR — это значение, которое должно быть загружено в фактический регистр автоматической перезагрузки.

Обратитесь к [Раздел 15.3.1: Единица измерения времени](#) для получения более подробной информации об обновлении и поведении ARR.

Счетчик заблокирован, пока значение автоматической перезагрузки равно нулю.

15.4.13 Регистр захвата/сравнения TIMx 1 (TIMx_CCR1)

Смещение адреса: 0x34

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая

Биты 15:0CCR1[15:0]: захват/сравнение 1 значения

Если канал CC1 сконфигурирован как выход:

CCR1 — это значение, которое должно быть загружено в фактический регистр захвата/сравнения 1 (значение предварительной загрузки). Он загружается постоянно, если функция предварительной загрузки не выбрана в регистре TIMx_CCMR1 (бит OC1PE). В противном случае значение предварительной загрузки копируется в активный регистр захвата/сравнения 1 при возникновении события обновления.

Активный регистр захвата/сравнения содержит значение, которое нужно сравнить со счетчиком TIMx_CNT и передать на выходе OC1.

Если канал CC1 сконфигурирован как вход:

CCR1 — это значение счетчика, переданное последним событием захвата ввода 1 (IC1). Регистр TIMx_CCR1 доступен только для чтения и не может быть запрограммирован.

15.4.14 Регистр захвата/сравнения TIMx 2 (TIMx_CCR2)

Смещение адреса: 0x38

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая

Биты 15:0CCR2[15:0]: Захват/Сравнение 2 значений

Если канал CC2 сконфигурирован как выход:

CCR2 — это значение, которое должно быть загружено в фактический регистр захвата/сравнения 2 (значение предварительной загрузки).

Он загружается постоянно, если функция предварительной загрузки не выбрана в регистре TIMx_CCMR2 (бит OC2PE). В противном случае значение предварительной загрузки копируется в активный регистр захвата/сравнения 2 при возникновении события обновления.

Активный регистр захвата/сравнения содержит значение, которое нужно сравнить со счетчиком TIMx_CNT и передать на выходе OC2.

Если канал CC2 сконфигурирован как вход:

CCR2 — это значение счетчика, переданное последним событием захвата входа 2 (IC2). Регистр TIMx_CCR2 доступен только для чтения и не может быть запрограммирован.

15.4.15 Регистр захвата/сравнения TIMx 3 (TIMx_CCR3)

Смещение адреса: 0x3C

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая

Биты 15:0CCR3[15:0]: Захват/Сравнение значения

Если канал CC3 сконфигурирован как выход:

CCR3 — это значение, которое должно быть загружено в фактический регистр захвата/сравнения 3 (значение предварительной загрузки).

Он загружается постоянно, если функция предварительной загрузки не выбрана в регистре TIMx_CCMR3 (бит OC3PE). В противном случае значение предварительной загрузки копируется в активный регистр захвата/сравнения 3 при возникновении события обновления.

Активный регистр захвата/сравнения содержит значение, которое нужно сравнить со счетчиком TIMx_CNT и передать на выходе OC3.

Если канал CC3 сконфигурирован как вход:

CCR3 — это значение счетчика, переданное последним событием захвата входа 3 (IC3). Регистр TIMx_CCR3 доступен только для чтения и не может быть запрограммирован.

15.4.16 Регистр захвата/сравнения TIMx 4 (TIMx_CCR4)

Смещение адреса: 0x40

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая	прямая/прямая

- Биты 15:0 **CCR4[15:0]**: Захват/Сравнение значения
1. если канал CC4 сконфигурирован как выход (биты CC4S):
- CCR4 — это значение, которое должно быть загружено в фактический регистр захвата/сравнения 4 (значение предварительной загрузки). Он загружается постоянно, если функция предварительной загрузки не выбрана в регистре TIMx_CCMR4 (бит OC4PE). В противном случае значение предварительной загрузки копируется в активный регистр захвата/сравнения 4, когда происходит событие обновления.
- Активный регистр захвата/сравнения содержит значение, которое нужно сравнить со счетчиком TIMx_CNT и передать на выходе OC4.
2. если канал CC4 сконфигурирован как вход (биты CC4S в регистре TIMx_CCMR4): CCR4 — это значение счетчика, переданное последним событием захвата 4 входа (IC4). Регистр TIMx_CCR4 доступен только для чтения и не может быть запрограммирован.

15.4.17 Регистр управления DMA TIMx (TIMx_DCR)

Смещение адреса: 0x48

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Сдержанный		ДБЛ[4:0]					Сдержанный					БД [4:0]			
		RW	RW	RW	RW	RW						RW	RW	RW	RW

- Биты 15:13 Зарезервировано, должно быть сохранено значение сброса.
- Биты 12:8 **дБЛ[4:0]**: длина пакета прямого доступа к памяти
- Этот 5-битный вектор определяет количество передач DMA (таймер распознает пакетную передачу, когда доступ для чтения или записи выполняется по адресу TIMx_DMAR).
- 00000: 1 передача,
00001: 2 передачи,
00010: 3 передачи,
...
10001: 18 передач.
- Биты 7:5 Зарезервировано, должно быть сохранено значение сброса.
- Биты 4:0 **БД [4:0]**: базовый адрес DMA
- Этот 5-битный вектор определяет базовый адрес для передачи DMA (когда доступ для чтения/записи осуществляется через адрес TIMx_DMAR). DBA определяется как смещение, начинающееся с адреса регистра TIMx_CR1.
- Пример:
00000: TIMx_CR1,
00001: TIMx_CR2,
00010: TIMx_SMCR,
...
- Пример:**Рассмотрим следующую передачу: DBL = 7 передач и DBA = TIMx_CR1. В этом случае передача осуществляется в/из регистров, начиная с адреса TIMx_CR1.

15.4.18 Адрес TIMx DMA для полной передачи (TIMx_DMAR)

Смещение адреса: 0x4C

Значение сброса: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ДМАБ[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 15:0 **ДМАБ[15:0]**: регистр DMA для пакетного доступа

Операция чтения или записи в регистр DMAR обращается к регистру, расположенному по адресу
(адрес TIMx_CR1) + (индекс DBA + DMA) x 4

где адрес TIMx_CR1 — это адрес управляющего регистра 1, DBA — это базовый адрес DMA, настроенный в регистре TIMx_DCR, индекс DMA автоматически контролируется передачей DMA и находится в диапазоне от 0 до DBL (DBL, настроенный в TIMx_DCR).

Пример того, как использовать функцию пакета DMA

В этом примере функция пакетного DMA таймера используется для обновления содержимого регистров CCRx (x = 2, 3, 4) с передачей полуслов DMA в регистры CCRx.

Это делается в следующие шаги:

1. Настройте соответствующий канал DMA следующим образом:
 - Периферийный адрес канала DMA – это адрес регистра DMAR.
 - Адрес памяти канала DMA – это адрес буфера в RAM, содержащего данные, которые должны быть переданы DMA в регистры CCRx.
 - Количество данных для передачи = 3 (см. примечание ниже).
 - Круговой режим отключен.
2. Настройте регистр DCR, настроив битовые поля DBA и DBL следующим образом:
DBL = 3 передачи, DBA = 0xE.
3. Включите запрос TIMx на обновление DMA (установите бит UDE в регистре DIER).
4. Включите TIMx
5. Включите канал DMA

Примечание:

Этот пример предназначен для случая, когда каждый регистр CCRx обновляется один раз. Например, если каждый регистр CCRx должен обновляться дважды, число передаваемых данных должно быть равно 6. Возьмем в качестве примера буфер в ОЗУ, содержащий данные1, данные2, данные3, данные4, данные5 и данные6. Данные передаются в регистры CCRx следующим образом: при первом запросе обновления DMA данные1 передаются в CCR2, данные2 передаются в CCR3, данные3 передаются в CCR4 и при втором запросе обновления DMA данные4 передаются в CCR2, данные5 передается в CCR3, а данные6 передаются в CCR4.

15.4.19 Карта регистров TIMx

Регистры TIMx отображаются, как описано в таблице ниже:

Табл. 88. Карта регистров TIMx и значения сброса

Компенсировать	регистр	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																			
		Сдержанный																						ХБП [1:0]	APIE	CMS [1:0]	ДНР	ОПМ	УРС	ЮДИС	CEN																					
0x00	TIMx_CR1	Сдержанный																						0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Сбросить значение																																																			
0x04	TIMx_CR2	Сдержанный																						0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Сбросить значение																																																			
0x08	TIMx_SMCR	Сдержанный																ЭТП	ЭЗК	ЭТП [1:0]	ETF[3:0]		МСМ	ТС[2:0]		Сдержанный	СМC[2:0]																									
	Сбросить значение																	0	0	0	0	0		0	0		0	0	0	0	0	0	0	0	0	0	0	0	0													
0x0C	TIMx_DIER	Сдержанный																ТДЭ	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	НЖЯ	Сдержанный	ГЛСТУХ	Сдержанный	CC4IE	CC3IE	CC2IE	CC1IE	ИЮО																				
	Сбросить значение																	0	0	0	0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x10	TIMx_SR	Сдержанный																CC40F	CC30F	CC20F	CC10F	Сдержанный	ТИФ	Сдержанный	CC4IF	CC3IF	CC2IF	CC1IF	ПИФ																							
	Сбросить значение																																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TIMx_EGR	Сдержанный																						0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Сбросить значение																																																			
0x18	TIMx_CCMR1 Выход сравнивать Режим	Сдержанный																OC2CE	OC2M [2:0]		OC2PE	OC2FE	CC2S [1:0]	OC1CE	OC1M [2:0]		OC1PE	OC1FE	CC1S [1:0]																							
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
	TIMx_CCMR1 Входной захват Режим	Сдержанный																IC2F[3:0]		IC2 PSC [1:0]	CC2S [1:0]	IC1F[3:0]		IC1 PSC [1:0]	CC1S [1:0]																											
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	TIMx_CCMR2 Выход сравнивать Режим	Сдержанный																OC4CE	OC4M [2:0]		OC4PE	OC4FE	CC4S [1:0]	OC3CE	OC3M [2:0]		OC3PE	OC3FE	CC3S [1:0]																							
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
	TIMx_CCMR2 Входной захват Режим	Сдержанный																IC4F[3:0]		IC4 PSC [1:0]	CC4S [1:0]	IC3F[3:0]		IC3 PSC [1:0]	CC3S [1:0]																											
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x20	TIMx_CCER	Сдержанный																Сдержанный	CC4P	CC4E	Сдержанный	CC3P	CC3E	Сдержанный	CC2P	CC2E	Сдержанный	CC1P	CC1E																							
	Сбросить значение																		0	0		0	0		0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0											

Табл. 88. Карта регистров TIMx и значения сброса (продолжение)

Компенсировать	регистр	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0x24	TIMx_CNT	Сдержанный																HKT[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Сдержанный																ПСК[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Сдержанный																APP[15:0]																
	Сбросить значение																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	Сдержанный																																	
0x34	TIMx_CCR1	Сдержанный																CCR1[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIMx_CCR2	Сдержанный																CCR2[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	TIMx_CCR3	Сдержанный																CCR3[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	TIMx_CCR4	Сдержанный																CCR4[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	Сдержанный																																	
0x48	TIMx_DCR	Сдержанный																		ДБЛ[4:0]				Сдержанный		БД [4:0]								
	Сбросить значение																			0	0	0	0			0	0	0	0	0	0	0	0	0
0x4C	TIMx_DMAR	Сдержанный																ДМАБ[15:0]																
	Сбросить значение																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Ссылаться на [Раздел 3.3: Карта памяти](#) для граничных адресов регистра.



16 Таймеры общего назначения (от TIM9 до TIM14)

Устройства с низкой плотностью это микроконтроллеры STM32F101xx, STM32F102xx и STM32F103xx с плотностью флэш-памяти от 16 до 32 Кбайт.

Устройства средней плотности это микроконтроллеры STM32F101xx, STM32F102xx и STM32F103xx с плотностью флэш-памяти от 64 до 128 Кбайт.

Устройства высокой плотности это микроконтроллеры STM32F101xx и STM32F103xx с плотностью флэш-памяти от 256 до 512 Кбайт.

Устройства XL-плотности это микроконтроллеры STM32F101xx и STM32F103xx, в которых плотность флэш-памяти колеблется от 768 Кбайт до 1 Мбайт.

Устройства линии связи это микроконтроллеры STM32F105xx и STM32F107xx.

Этот раздел относится к *Только устройства XL-плотности*.

16.1 Введение от TIM9 до TIM14

Таймеры общего назначения от TIM9 до TIM14 состоят из 16-разрядного счетчика с автоматической перезагрузкой, управляемого программируемым делителем.

Их можно использовать для различных целей, включая измерение длительности импульсов входных сигналов (захват входных сигналов) или генерацию форм выходных сигналов (сравнение выходных сигналов, ШИМ).

Длины импульсов и периоды сигналов можно модулировать от нескольких микросекунд до нескольких миллисекунд с помощью предварительного делителя таймера и предварительного делителя контроллера тактовых импульсов RCC.

Таймеры от TIM9 до TIM14 полностью независимы и не используют общие ресурсы. Их можно синхронизировать вместе, как описано в [Раздел 16.3.12](#).