

RM0091 Reference manual

Это справочное руководство предназначено для разработчиков приложений. Оно предоставляет полную информацию о том, как использовать память микроконтроллера STM32F0x1 / STM32F0x2 / STM32F0x8 и периферийные устройства.

Это относится к устройствам STM32F031x4 / x6, STM32F051x8 / xB, STM32F091xB / xC, STM32F042x4 / x6, STM32F072x8 / xB, STM32F038x6, STM32F048x6, STM32F058x8, STM32F078xB и STM32F098xC.

Для целей настоящего руководства микроконтроллеры STM32F0x1 / STM32F0x2 / STM32F0x8 называются "STM32F0xx".

STM32F0xx — это семейство микроконтроллеров с различными размерами памяти, пакетами и периферийными устройствами.

Page 227

13 Аналого-цифровой преобразователь (АЦП)

Далее по тексту АЦП заменено на ADC, потому что Гугль писал когда как. Кроме того, на мой взгляд, это не создает разночтения с примерами.

3.1. Введение

12-разрядный ADC представляет собой последовательный аналого-цифровой преобразователь. Он имеет до 19 мультиплексных каналов, что позволяет ему измерять сигналы от 16 внешних и 3 внутренних источников.

ADC различных каналов может быть выполнено в однократном, непрерывном, сканирующем или дискретном режиме. Результат ADC сохраняется в регистре данных с выровненными по левому краю или выровненным по правому краю.

Функция аналогового сторожевого таймера позволяет приложению обнаруживать, находится ли входное напряжение выше заданных пользователем верхних или нижних порогов.

Реализуется эффективный режим с низким энергопотреблением, позволяющий обеспечить очень низкое потребление на низкой частоте.

13.2. Основные функции ADC

- Высокая производительность
 - 12-битное, 10-битное, 8-битное или 6-битное конфигурируемое разрешение
 - Время преобразования ADC: 1,0 мкс для 12-битного разрешения (1 МГц), 0,93 мкс время преобразования для 10-битного разрешения, более быстрое время преобразования может быть получено путем снижения разрешения.
 - Самокалибровка
 - Программируемое время выборки
 - Выравнивание данных со встроенной когерентностью данных
 - Поддержка DMA
- Малая мощность
 - Приложение может снизить частоту PCLK для работы с низким энергопотреблением при сохранении оптимальной производительности ADC. Например, сохраняется время преобразования 1,0 мкс, независимо от частоты PCLK)
 - Режим ожидания: предотвращает переполнение ADC в приложениях с низкой частотой PCLK
 - Режим автоматического выключения: ADC автоматически отключается, за исключением активной фазы преобразования. Это значительно снижает энергопотребление ADC.
- Аналоговые входные каналы
 - 16 внешних аналоговых входов
 - 1 канал для внутреннего температурного датчика (VSENSE)

- 1 канал для внутреннего опорного напряжения (VREFINT)
 - 1 канал для контроля внешнего штыря питания VBAT.
 - Начало преобразования может быть инициировано:
 - ПО
 - Аппаратными триггерами с настраиваемой полярностью (внутренние события таймера от TIM1, TIM2, TIM3 и TIM15)
 - Режимы конверсии
 - Может конвертировать один канал или может сканировать последовательность каналов.
 - Одиночный режим конвертирует выбранные входы один раз за триггер
 - Непрерывный режим конвертирует выбранные входы непрерывно
 - Прерывистый режим
 - Генерация прерывания в конце выборки, завершении преобразования, завершении преобразования последовательности и в случае событий аналогового сторожевого таймера или переполнения
 - Аналоговый сторожевой таймер
 - Требования к питанию ADC: 2,4 В до 3,6 В
 - Входной диапазон ADC: $V_{SSA} \leq V_{IN} \leq V_{DDA}$
- На рис. 26 показана блок-схема ADC.

13.3 ADC pins and internal signals

Table 41. ADC internal signals

Internal signal name	Signal type	Description
TRGx	Input	ADC conversion triggers
V _{SENSE}	Input	Internal temperature sensor output voltage
VREFINT	Input	Internal voltage reference output voltage
V _{BAT/2}	Input	V _{BAT} pin input voltage divided by 2

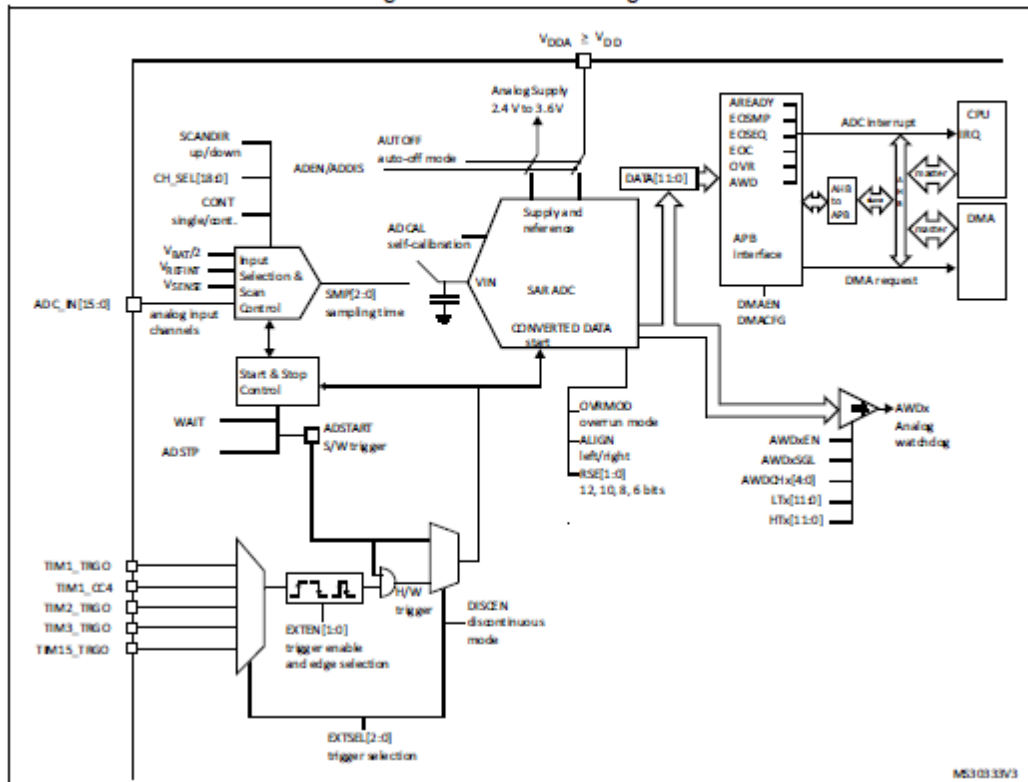
Table 42. ADC pins

Name	Signal type	Remarks
V _{DDA}	Input, analog power supply	Analog power supply and positive reference voltage for the ADC. $V_{DDA} \geq V_{DD}$
V _{SSA}	Input, analog supply ground	Ground for analog power supply. Must be at V _{SS} potential
ADC_IN[15:0]	Analog input signals	16 analog input channels

13.4 ADC functional description

Figure 26 shows the ADC block diagram and Table 42 gives the ADC pin description.

Figure 26. ADC block diagram



13.4.1 Калибровка (ADCAL)

ADC имеет функцию калибровки. Во время процедуры ADC вычисляет коэффициент калибровки, который применяется внутри ADC до следующего выключения питания ADC. Приложение не должно использовать ADC во время калибровки и должно дождаться завершения.

Перед началом аналого-цифрового преобразования необходимо выполнить калибровку. Она устраняет ошибку смещения, которая может изменяться от чипа к чипу вследствие изменения процесса.

Калибровка инициируется программным обеспечением путем установки бита ADCAL = 1. Калибровку можно начать только при отключенном ADC (когда ADEN = 0). Бит ADCAL остается на 1 во время всей калибровочной последовательности. Затем он очищается аппаратными средствами, как только калибровка завершается. После этого калибровочный коэффициент можно считать из регистра ADC_DR (от битов 6 до 0).

Внутренняя аналоговая калибровка сохраняется, если ADC отключен (ADEN = 0). Когда изменяются условия работы ADC (изменения VDDA являются основным фактором изменения смещения ADC и изменения температуры в меньшей степени), рекомендуется повторить цикл калибровки.

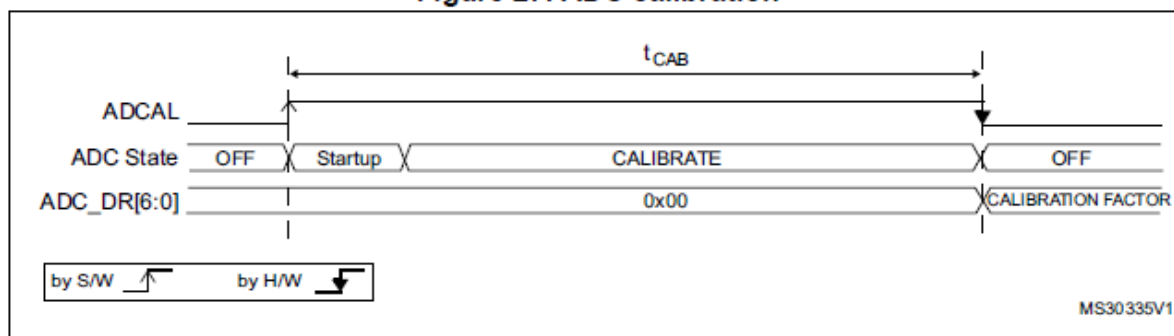
Калибровочный коэффициент теряется каждый раз, когда питание отключается от ADC (например, когда устройство переходит в режим STANDBY).

Процедура программного обеспечения калибровки

1. Убедитесь, что ADEN = 0 и DMAEN = 0
2. Установите ADCAL = 1
3. Подождите, пока ADCAL = 0
4. Калибровочный коэффициент может быть считан из битов 6: 0 ADC_DR.

Пример кода см. в разделе Приложение А.7.1: Пример кода калибровки ADC

Figure 27. ADC calibration



13.4.2 Управление включением и выключением ADC (ADEN, ADDIS, ADRDY)

При включении питания микроконтроллера ADC отключается и переходит в режим пониженного энергопотребления (ADEN = 0).

Как показано на рисунке 28, ADC нуждается в времени стабилизации t_{STAB} до того, как он начнет точное преобразование.

Два бита управления используются для включения или выключения ADC:

- Установите ADEN = 1, чтобы включить ADC. Флаг ADRDY устанавливается, как только ADC готов к работе.

- Установите ADDIS = 1 для отключения ADC и переведите ADC в режим пониженного энергопотребления. Биты ADEN и ADDIS затем автоматически очищаются аппаратными средствами, как только ADC полностью отключен.

После этого преобразование может начаться либо установкой ADSTART = 1 (см. Раздел 13.5: Преобразование по внешнему триггеру и полярности триггера (EXTSEL, EXTEN) на стр. 238) или когда внешнее событие триггера возникает, если триггеры включены.

Выполните эту процедуру, чтобы включить ADC:

1. Очистите бит ADRDY в регистре ADC_ISR, запрограммировав этот бит на 1.
2. Установите ADEN = 1 в регистре ADC_CR.
3. Дождитесь, пока ADRDY = 1 в регистре ADC_ISR и продолжит запись ADEN = 1 (ADRDY устанавливается после времени запуска ADC). Это может быть обработано прерыванием, если прерывание разрешено установкой бита ADRDYIE в регистре ADC_IER.

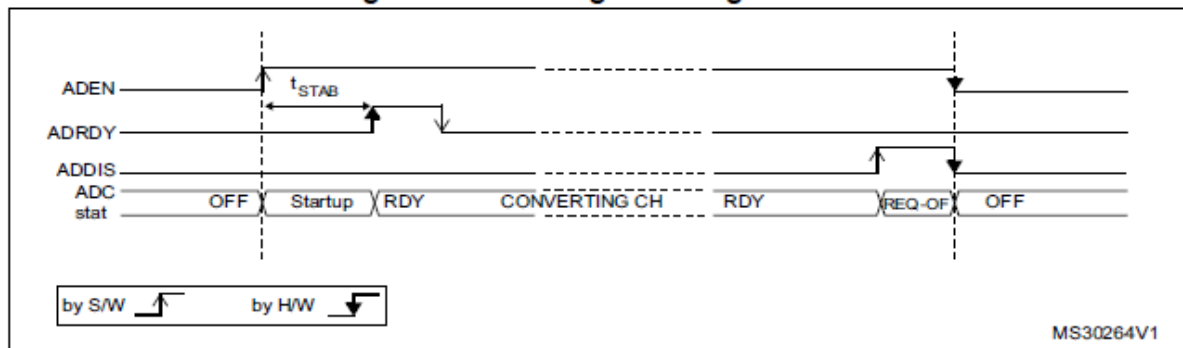
Пример кода см. в разделе Приложение А.7.2: Пример кода последовательности включения ADC.

Выполните эту процедуру, чтобы отключить ADC:

1. Проверьте, что ADSTART = 0 в регистре ADC_CR, чтобы убедиться, что преобразование не выполняется. Если необходимо, остановите любое текущее преобразование, записав 1 в бит ADSTP в регистре ADC_CR и дождитесь, пока этот бит не будет считан 0.
2. Установите ADDIS = 1 в регистре ADC_CR.
3. Если это требуется приложению, дождитесь, пока ADEN = 0 в регистре ADC_CR, показывая, что ADC полностью отключен (ADDIS автоматически сбрасывается после ADEN = 0).
4. Очистите бит ADRDY в регистре ADC_ISR, запрограммировав этот бит на 1 (необязательный). Пример кода см. в разделе Приложение А.7.3: Пример кода последовательности отключения ADC.

Внимание: Бит ADEN нельзя установить при ADCAL = 1 и в течение четырех тактовых импульсов ADC после того, как бит ADCAL будет очищен аппаратно (окончание калибровки).

Figure 28. Enabling/disabling the ADC



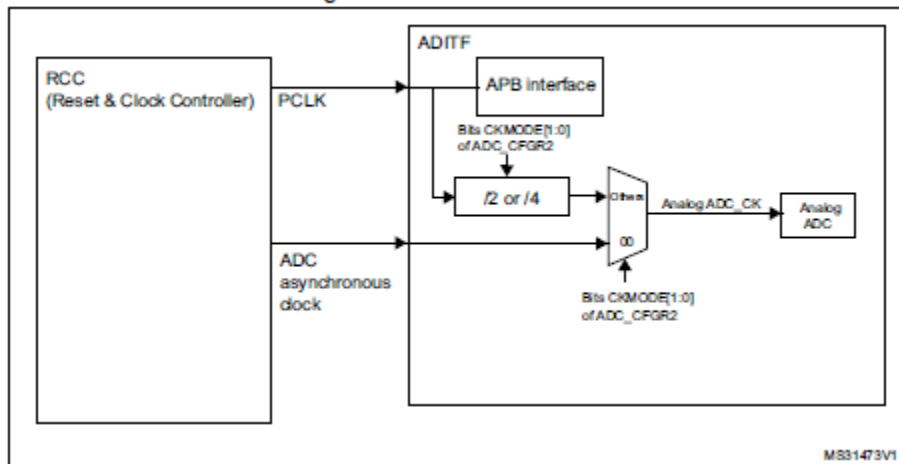
Примечание: в режиме автоматического выключения (AUTOFF = 1) фазы включения / выключения выполняются автоматически, аппаратно и флаг ADRDY не установлен.

13.4.3 ADC clock (CKMODE)

ADC имеет архитектуру с двумя тактовыми доменами, поэтому для ADC можно использовать часы

(Асинхронные часы ADC), не зависящие от тактовых импульсов APB (PCLK).

Figure 29. ADC clock scheme



1. Обратитесь к разделу 6: «Сброс и управление часами» (RCC) на стр. 93, чтобы узнать, как включены асинхронные часы PCLK и ADC

Входные тактовые импульсы аналогового ADC могут быть выбраны между двумя разными источниками тактовых импульсов (см. Рис. 29: схема тактовых импульсов ADC, чтобы увидеть, как включены PCLK и асинхронные часы ADC):

а) Часы ADC могут быть конкретным источником тактовой частоты, называемым «асинхронным синхронизатором ADC», который является независимым и асинхронным с часами APB. Обратитесь к разделу RCC за дополнительной информацией о создании этого источника синхронизации.

Чтобы выбрать эту схему, биты CKMODE [1: 0] регистра ADC_CFGR2 должны быть сброшены.

Пример кода см. в разделе Приложение A.7.4: Пример кода выбора ADC.

б) Часы ADC могут быть получены из тактовой частоты APB интерфейса шины ADC, деленной на программируемый коэффициент (2 или 4) в соответствии с битами CKMODE [1: 0].

Чтобы выбрать эту схему, биты CKMODE [1: 0] регистра ADC_CFGR2 должны отличаться от «00».

Вариант а) имеет то преимущество, что достигается максимальная тактовая частота ADC независимо от выбранной тактовой схемы APB.

Вариант б) имеет преимущество в обход синхронизации синхронизации часов. Это может быть полезно, когда ADC запускается таймером, и если приложение требует, чтобы ADC запускался точно без какой-либо неопределенности (в противном случае неопределенность момента запуска добавляется повторной синхронизацией между двумя областями синхронизации).

Таблица 43. Задержка между запуском и началом преобразования

Источник тактирования	CKMODE[1:0]	Задержка между событием триггера и началом преобразования
Выделенный 14 МГц тактовый сигнал PCLK, деленный на 2	00	Задержка не является детерминированной (дребезг)
	01	Задержка детерминирована (без дребезга) и равна 2.75 тактов ADC
PCLK, деленный на 4	10	Задержка является детерминированной (без дребезга) и равна 2.625 тактам ADC

13.4.4 Конфигурирование ADC

Программное обеспечение должно записывать биты ADCAL и ADEN в регистр ADC_CR, если ADC отключен (ADEN должен быть 0).

Программное обеспечение должно записывать только биты ADSTART и ADDIS в регистре ADC_CR только в том случае, если ADC включен и нет ожидающего запроса на отключение ADC (ADEN = 1 и ADDIS = 0).

Для всех других битов управления в регистрах ADC_IER, ADC_CFGRi, ADC_SMPR, ADC_TR, ADC_CHSELR и ADC_CCR программное обеспечение должно записывать только биты управления конфигурацией, если ADC разрешен (ADEN = 1), и если нет преобразования, то выполняется ADSTART = 0).

Программное обеспечение должно записывать только в бит ADSTP в регистре ADC_CR, если ADC включен (и, возможно, конвертируется), и нет ожидающего запроса на отключение ADC (ADSTART = 1 и ADDIS = 0)

Примечание. Не существует аппаратной защиты, препятствующей программному обеспечению выполнять операции записи, запрещенные вышеуказанными правилами. Если происходит такой запрещенный доступ на запись, ADC может ввести неопределенное состояние. Для восстановления корректной работы в этом случае ADC должен быть отключен (очистить ADEN = 0 и все биты в регистре ADC_CR).

13.4.5 Выбор канала (CHSEL, SCANDIR)

До 19 мультиплексированных каналов:

- 16 аналоговых входов с контактов GPIO (ADC_IN0 ... ADC_IN15)
- 3 внутренних аналоговых входа (датчик температуры, внутреннее опорное напряжение, канал VBAT)
- Возможно преобразование одного канала или автоматическое сканирование последовательности каналов.

Последовательность конвертируемых каналов должна быть запрограммирована в регистре выбора канала ADC_CHSELR: каждый канал аналогового ввода имеет выделенный бит выбора (CHSEL0 ... CHSEL18).

Порядок, в котором каналы будут сканироваться, можно настроить, запрограммировав бит бит SCANDIR в регистре ADC_CFGR1:

- SCANDIR = 0: прямое сканирование. Канал 0 — канал 18
- SCANDIR = 1: обратное сканирование. Канал 18 — канал 0

Датчик температуры, внутренние каналы VREFINT и VBAT

Датчик температуры подключен к каналу ADC_IN16. Внутренний опорный сигнал напряжения VREFINT подключен к каналу ADC_IN17. Канал VBAT подключен к каналу ADC_IN18.

13.4.6 Программируемое время выборки (SMP)

Перед началом преобразования ADC необходимо установить прямое соединение между измеряемым источником напряжения и встроенным конденсатором выборки ADC. Это время выборки должно быть достаточно для того, чтобы источник входного напряжения заряжал образец и удерживал конденсатор до уровня входного напряжения.

Наличие программируемого времени выборки позволяет обрезать скорость преобразования в соответствии с входным сопротивлением источника входного напряжения.

ADC производит выбор входного напряжения для ряда тактовых циклов ADC, которые могут быть изменены с использованием битов SMP [2: 0] в регистре ADC_SMPR.

Это программируемое время выборки является общим для всех каналов. Если это требуется приложением, программное обеспечение может изменять и адаптировать это время выборки между каждым преобразованием.

Общее время преобразования рассчитывается следующим образом:

$T_{CONV} = \text{время выборки} + 12,5 \times \text{такты ADC}$

Пример:

При $ADC_CLK = 14 \text{ МГц}$ и времени выборки 1,5 ADC тактов:

$T_{CONV} = 1,5 + 12,5 = 14 \text{ тактов ADC} = 1 \text{ мкс}$

ADC указывает конец фазы выборки, установив флаг EOSMP.

13.4.7 Режим одиночного преобразования (CONT = 0)

В режиме одиночного преобразования ADC выполняет одну последовательность преобразований, один раз преобразовывая все каналы. Этот режим выбирается, когда CONT = 0 в регистре ADC_CFGR1.

Преобразование начинается с:

- Установка бита ADSTART в регистре ADC_CR
- Событие аппаратного триггера

Внутри последовательности после завершения каждого преобразования:

- Преобразованные данные сохраняются в 16-разрядном регистре ADC_DR
- Установлен флаг EOC (окончание преобразования)
- Прерывание генерируется, если бит EOCIE установлен

После завершения последовательности преобразований:

- Установлен флаг EOSEQ (конец последовательности)
- Прерывание генерируется, если бит EOSEQIE установлен

Затем ADC останавливается до появления нового внешнего события триггера или снова устанавливается бит ADSTART.

Примечание. Чтобы преобразовать один канал, запрограммируйте последовательность длиной 1.

13.4.8 Режим непрерывного преобразования (CONT = 1)

В режиме непрерывного преобразования, когда происходит событие запуска программного или аппаратного обеспечения, ADC выполняет последовательность преобразований, один раз преобразовывая все каналы, а затем автоматически перезапускается и непрерывно выполняет одну и ту же последовательность преобразований. Этот режим выбирается, когда CONT = 1 в регистре ADC_CFGR1. Преобразование начинается с:

- Установка бита ADSTART в регистре ADC_CR
- Событие аппаратного триггера

Внутри последовательности после завершения каждого преобразования:

- Преобразованные данные сохраняются в 16-разрядном регистре ADC_DR
- Установлен флаг EOC (окончание преобразования)
- Прерывание генерируется, если бит EOCIE установлен

После завершения последовательности преобразований:

- Установлен флаг EOSEQ (конец последовательности)
- Прерывание генерируется, если бит EOSEQIE установлен

Затем новая последовательность немедленно перезапускается, и ADC непрерывно повторяет последовательность преобразования.

Примечание. Чтобы преобразовать один канал, запрограммируйте последовательность длиной 1.

Невозможно включить и прерывистый режим, и непрерывный режим: запрещено устанавливать оба бита DISCEN = 1 и CONT = 1.

13.4.9 Запуск преобразований (ADSTART)

Программное обеспечение запускает преобразования ADC, устанавливая ADSTART = 1.

Когда ADSTART установлен, преобразование:

- Начало немедленно, если EXTEN = 00 (триггер программного обеспечения)
- На следующем активном фронте выбранного аппаратного триггера, если EXTEN ≠ 00

Бит ADSTART также используется для указания, выполняется ли в настоящее время операция ADC. Можно переконфигурировать ADC, пока ADSTART = 0, что говорит о том, что ADC находится в режиме ожидания.

Бит ADSTART очищается аппаратно:

- В одиночном режиме с программным триггером (CONT = 0, EXTEN = 00)
 - На любом конце последовательности преобразования (EOSEQ = 1)
- В прерывистом режиме с программным триггером (CONT = 0, DISCEN = 1, EXTEN = 00)
 - В конце преобразования (EOC = 1)
- Во всех случаях (CONT = x, EXTEN = XX)
 - После выполнения процедуры ADSTP, вызываемой программным обеспечением (см. Раздел 13.4.11: Остановка текущего преобразования (ADSTP) на стр. 238)

Примечание. В непрерывном режиме (CONT = 1) бит ADSTART не очищается оборудованием, если установлен флаг EOSEQ, потому что последовательность автоматически возобновляется.

Когда аппаратный триггер выбран в одиночном режиме (CONT = 0 и EXTEN = 01), ADSTART не очищается оборудованием, если установлен флаг EOSEQ. Это позволяет избежать необходимости в softwar снова устанавливать бит ADSTART и гарантирует, что следующее событие триггера не будет пропущено.

13.4.10 Расчет времени

Истекшее время между началом преобразования и окончанием преобразования представляет собой сумму сконфигурированного времени выборки плюс последовательное время аппроксимации в зависимости от разрешения данных:

$$t_{ADC} = t_{SMPL} + t_{SAR} = [1.5 \text{ } \mu\text{s} + 12.5 \text{ } \mu\text{s} \cdot \text{RES}[\text{2:0}]] \cdot t_{ADC_CLK}$$

$$t_{ADC} = t_{SMPL} + t_{SAR} = 107.1 \text{ ns} \cdot \text{RES}[\text{2:0}] + 892.8 \text{ ns} = 1 \text{ } \mu\text{s} \text{ } \mu\text{min (for } f_{ADC_CLK} = 14 \text{ MHz)}$$

Figure 30. Analog to digital conversion time

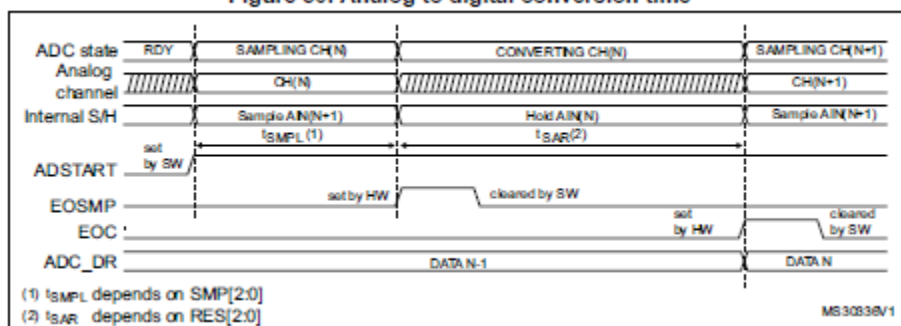
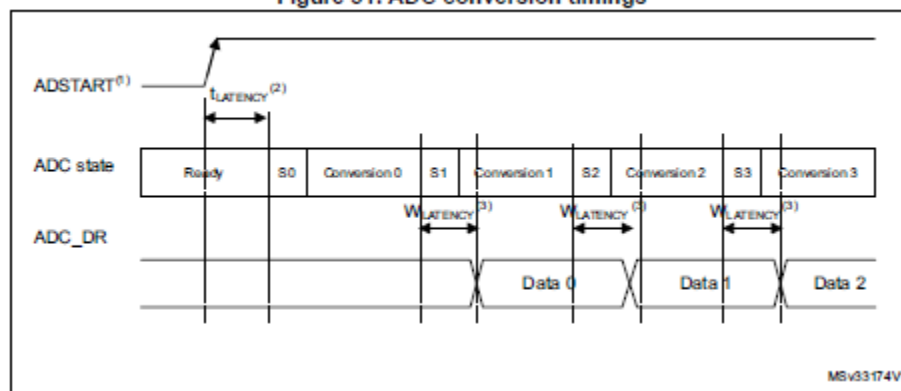


Figure 31. ADC conversion timings



1. EXTEN = 00 или EXTEN ≠ 00
2. Задержка запуска (более подробная информация приведена в техническом описании)
3. Задержка записи в регистр ADC_DR (см. Таблицу для более подробной информации)

13.4.11 Остановка текущего преобразования (ADSTP)

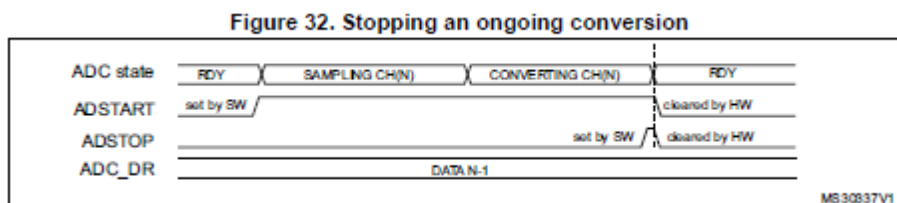
Программное обеспечение может решить остановить любые текущие преобразования, установив $ADSTP = 1$ в регистре ADC_CR .

Это сбросит работу ADC, и ADC будет бездействующим, готовым к новой операции.

Когда бит $ADSTP$ устанавливается программным обеспечением, любое текущее преобразование прерывается и результат отбрасывается (регистр ADC_DR не обновляется текущим преобразованием).

Последовательность сканирования также прерывается и возвращается в исходное состояние (это означает, что перезапуск ADC приведет к повторному запуску новой последовательности).

По завершении этой процедуры биты $ADSTP$ и $ADSTART$ очищаются аппаратно, и программное обеспечение должно дождаться, пока $ADSTART = 0$ не начнет новые преобразования.



13.5 Преобразование на внешнюю триггерную и триггерную полярность (EXTSEL, EXTEN)

Преобразование или последовательность преобразования могут запускаться программным обеспечением или внешним событием (например, захватом таймера). Если управляющие биты $EXTEN [1: 0]$ не равны «0b00», внешние события могут инициировать преобразование с выбранной полярностью. Выбор триггера эффективен, когда программное обеспечение установило бит $ADSTART = 1$.

Все аппаратные триггеры, которые происходят во время преобразования, игнорируются.

Если бит $ADSTART = 0$, любые аппаратные триггеры, которые происходят, игнорируются.

Таблица 44 обеспечивает соответствие между значениями $EXTEN [1: 0]$ и полярностью триггера.

Таблица 44. Настройка полярности триггера

Источник	$EXTEN [1: 0]$
Обнаружение триггеров отключено	00
Обнаружение при подъеме edge	01
Обнаружение на краю падения	10
Обнаружение на восходящем и нисходящем краях	11

Примечание. Полярность внешнего триггера может быть изменена только при отсутствии преобразования ADC ($ADSTART = 0$).

Биты управления $EXTSEL [2: 0]$ используются для выбора, какое из 8 возможных событий может инициировать преобразования.

Таблица 45 дает возможный внешний триггер для регулярного преобразования.

События запуска источника программного обеспечения могут генерироваться путем установки бита $ADSTART$ в регистре ADC_CR .

Table 45. External triggers

Name	Source	EXTSEL[2:0]
TRG0	TIM1_TRGO	000
TRG1	TIM1_CC4	001
TRG2	TIM2_TRGO	010
TRG3	TIM3_TRGO	011
TRG4	TIM15_TRGO	100
TRG5	Reserved	101
TRG6	Reserved	110
TRG7	Reserved	111

Примечание. Выбор триггера может быть изменен только в том случае, если ADC не выполняет преобразование (ADSTART = 0).

13.5.1 Прерывистый режим (DISCEN)

Этот режим активируется установкой бита DISCEN в регистре ADC_CFGR1.

В этом режиме (DISCEN = 1) для запуска каждого преобразования, определенного в последовательности, требуется событие запуска оборудования или программного обеспечения. Наоборот, если DISCEN = 0, одно событие запуска оборудования или программного обеспечения последовательно запускает все преобразования, определенные в последовательности.

Пример:

- DISCEN = 1, каналы для преобразования = 0, 3, 7, 10
 - 1-й триггер: канал 0 преобразуется и генерируется событие EOC
 - 2-й триггер: канал 3 преобразуется и генерируется событие EOC
 - 3-й триггер: канал 7 преобразуется и генерируется событие EOC
 - 4-й триггер: преобразуется канал 10 и генерируются как события EOC, так и EOSEQ.
 - 5-й триггер: канал 0 преобразуется в событие EOC
 - 6-й триггер: канал 3 преобразуется и генерируется событие EOC
 - ...
- DISCEN = 0, каналы для преобразования = 0, 3, 7, 10
 - 1-й триггер: полная последовательность преобразуется: канал 0, затем 3, 7 и 10. Каждое преобразование генерирует событие EOC, а последнее также генерирует событие EOSEQ.
 - Любые последующие триггерные события перезапускают полную последовательность.

Примечание: невозможно включить как прерывистый режим, так и непрерывный режим: запрещено устанавливать оба бита DISCEN = 1 и CONT = 1.

13.5.2 Программируемое разрешение (RES) — режим быстрого преобразования

Можно сократить время преобразования (tSAR) за счет уменьшения разрешения ADC.

Разрешение можно настроить на 12, 10, 8 или 6 бит, запрограммировав биты RES [1: 0] в регистре ADC_CFGR1. Более низкое разрешение позволяет сократить время преобразования

Для приложений, где высокая точность данных не требуется.

Примечание: бит RES [1: 0] может быть изменен только при сбросе бита ADEN.

Результат преобразования всегда равен 12 битам, и любые неиспользуемые биты LSB читаются как нули.

Более низкое разрешение уменьшает время преобразования, необходимое для последовательных этапов аппроксимации, как показано в таблице 46.

Table 46. t_{SAR} timings depending on resolution

RES[1:0] bits	t_{SAR} (ADC clock cycles)	t_{SAR} (ns) at $f_{ADC} = 14$ MHz	t_{SMPL} (min) (ADC clock cycles)	t_{CONV} (ADC clock cycles) (with min. t_{SMPL})	t_{CONV} at $f_{ADC} =$ 14 MHz
12	12.5	893 ns	1.5	14	1000 ns
10	11.5	821 ns	1.5	13	928 ns
8	9.5	678 ns	1.5	11	785 ns
6	7.5	535 ns	1.5	9	643 ns

13.5.3 Окончание преобразования, завершение фазы выборки (EOC, флаги EOSMP)

ADC указывает каждый конец преобразования (EOC).

ADC устанавливает флаг EOC в регистре ADC_ISR, как только новый результат преобразования данных становится доступным в регистре ADC_DR. Прерывание может быть сгенерировано, если бит EOSMP установлен в регистре ADC_IER. Флаг EOC очищается программным обеспечением либо путем записи в него 1, либо путем считывания регистра ADC_DR.

ADC также указывает на завершение фазы выборки, установив флаг EOSMP в регистре ADC_ISR. Флаг EOSMP очищается программным обеспечением путем записи в него 1. Прерывание может быть сгенерировано, если бит EOSMPIE установлен в регистре ADC_IER.

Цель этого прерывания — синхронизировать обработку с конверсиями.

Как правило, аналоговый мультиплексор может быть доступен в скрытом времени во время фазы преобразования, так что мультиплексор позиционируется, когда начинается следующая выборка.

Примечание. Поскольку между окончанием выборки и окончанием преобразования осталось очень короткое время, он снова начинает использовать опрос или инструкцию WFE, а не прерывание и инструкцию WFI.

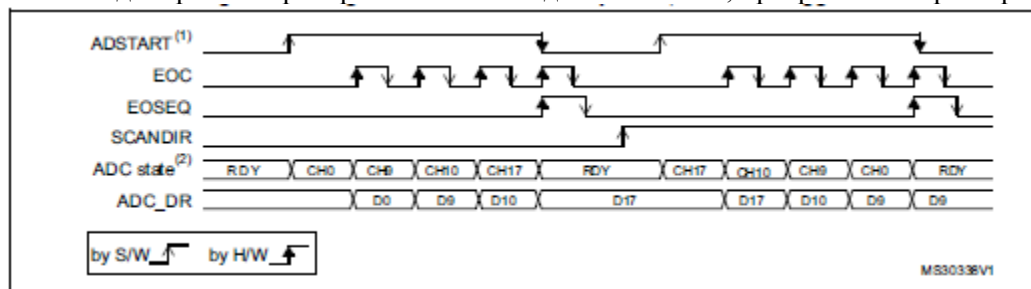
13.5.4 Конец последовательности преобразования (флаг EOSEQ)

ADC уведомляет приложение о каждом конце последовательности (EOSEQ).

ADC устанавливает флаг EOSEQ в регистре ADC_ISR, как только последний результат данных последовательности преобразования становится доступным в регистре ADC_DR. Прерывание может быть сгенерировано, если бит EOSEQIE установлен в регистре ADC_IER. Флаг EOSEQ очищается программным обеспечением путем записи в него 1.

13.5.5 Примеры временных диаграмм (одиночные / непрерывные режимы Аппаратные / программные триггеры)

Рисунок 33. Однократное преобразование последовательности, программный триггер

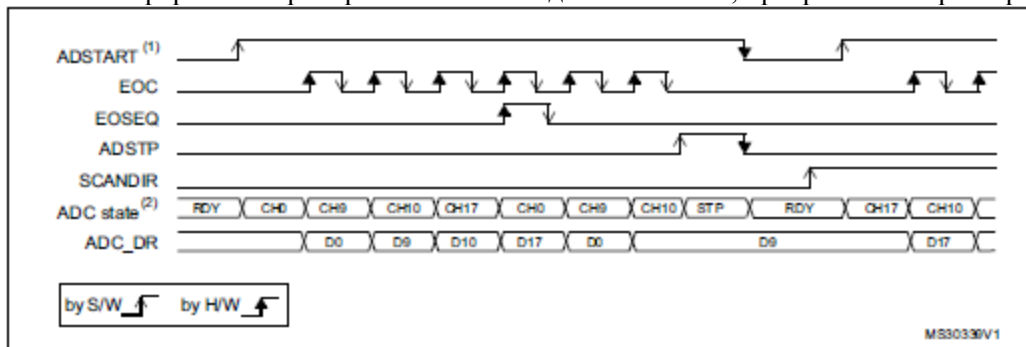


1. EXTEN=00, CONT=0

2. CHSEL=0x20601, WAIT=0, AUTOFF=0

Пример кода см. в разделе Приложение А.7.5. Пример кода последовательности с одним преобразованием — Программный триггер.

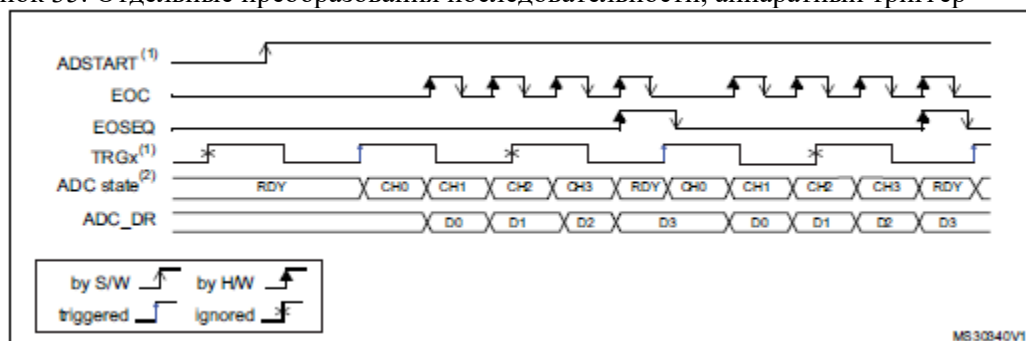
Рисунок 34. Непрерывное преобразование последовательности, программный триггер



1. EXTEN = 00, CONT = 1,
2. CHSEL = 0x20601, WAIT = 0, AUTOFF = 0

Пример кода см. в разделе Приложение А.7.6: Пример кода последовательности непрерывного преобразования — Программный триггер.

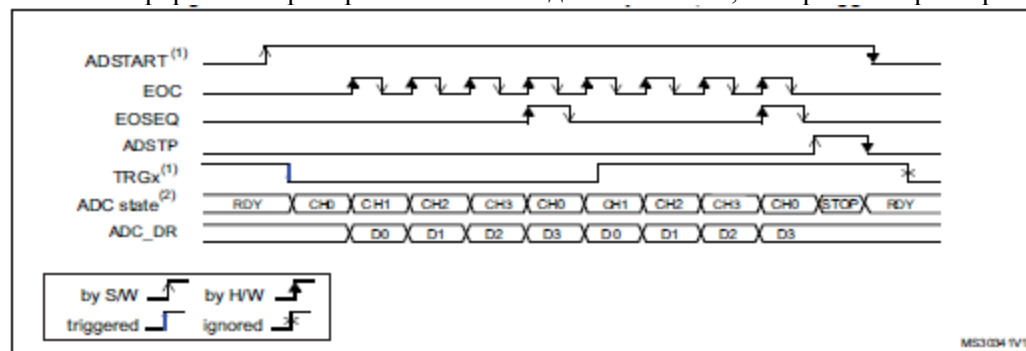
Рисунок 35. Отдельные преобразования последовательности, аппаратный триггер



1. EXTSEL = TRGx (превышение частоты), EXTEN = 01 (нарастающий фронт), CONT = 0
2. CHSEL = 0xF, SCANDIR = 0, WAIT = 0, AUTOFF = 0

Пример кода см. в разделе Приложение А.7.7. Пример кода последовательности с одним преобразованием — Аппаратный триггер.

Рисунок 36. Непрерывное преобразование последовательности, аппаратный триггер



1. EXTSEL = TRGx, EXTEN = 10 (спадающий фронт), CONT = 1
2. CHSEL = 0xF, SCANDIR = 0, WAIT = 0, AUTOFF = 0

Пример кода см. в разделе Приложение А.7.8: Пример кода последовательности непрерывного преобразования — Аппаратный запуск.

13.6 Управление данными

13.6.1 Регистр данных и выравнивание данных (ADC_DR, ALIGN)

В конце каждого преобразования (когда происходит событие EOC) результат преобразованных данных сохраняется в регистре данных ADC_DR, который имеет ширину 16 бит.

Формат ADC_DR зависит от сконфигурированного выравнивания и разрешения данных.

Бит ALIGN в регистре ADC_CFGR1 выбирает выравнивание данных, сохраненных после преобразования. Данные могут выравниваться по правому краю (ALIGN = 0) или выравниваться по левому краю (ALIGN = 1), как показано на рисунке 37.

Рисунок 37. Выравнивание и разрешение данных

ALIGN	RES	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0x0	0x0											DR[11:0]				
	0x1	0x00						DR[9:0]									
	0x2	0x00					DR[7:0]										
	0x3	0x00				DR[5:0]											
1	0x0	DR[11:0]											0x0				
	0x1	DR[9:0]						0x00									
	0x2	DR[7:0]					0x00										
	0x3	0x00				DR[5:0]											

MS30342V1

13.6.2 Переполнение ADC (OVR, OVRMOD)

Флаг переполнения (OVR) указывает на событие переполнения данных, когда преобразованные данные не были прочитаны во времени ЦПУ или DMA, до того как будут доступны данные из нового преобразования.

Флаг OVR устанавливается в регистре ADC_ISR, если флаг EOC все еще находится в '1' в то время, когда новое преобразование завершено. Прерывание может быть сгенерировано, если бит OVRIE установлен в регистре ADC_IER.

Когда возникает переполнение, ADC продолжает работать и может продолжать преобразовывать, если программное обеспечение не решит остановиться и сбросить последовательность, установив бит ADSTP в регистре ADC_CR.

Флаг OVR очищается программным обеспечением путем записи в него 1.

Можно настроить, сохраняются ли данные или перезаписываются, когда происходит событие переполнения, путем программирования бита OVRMOD в регистре ADC_CFGR1:

- OVRMOD = 0

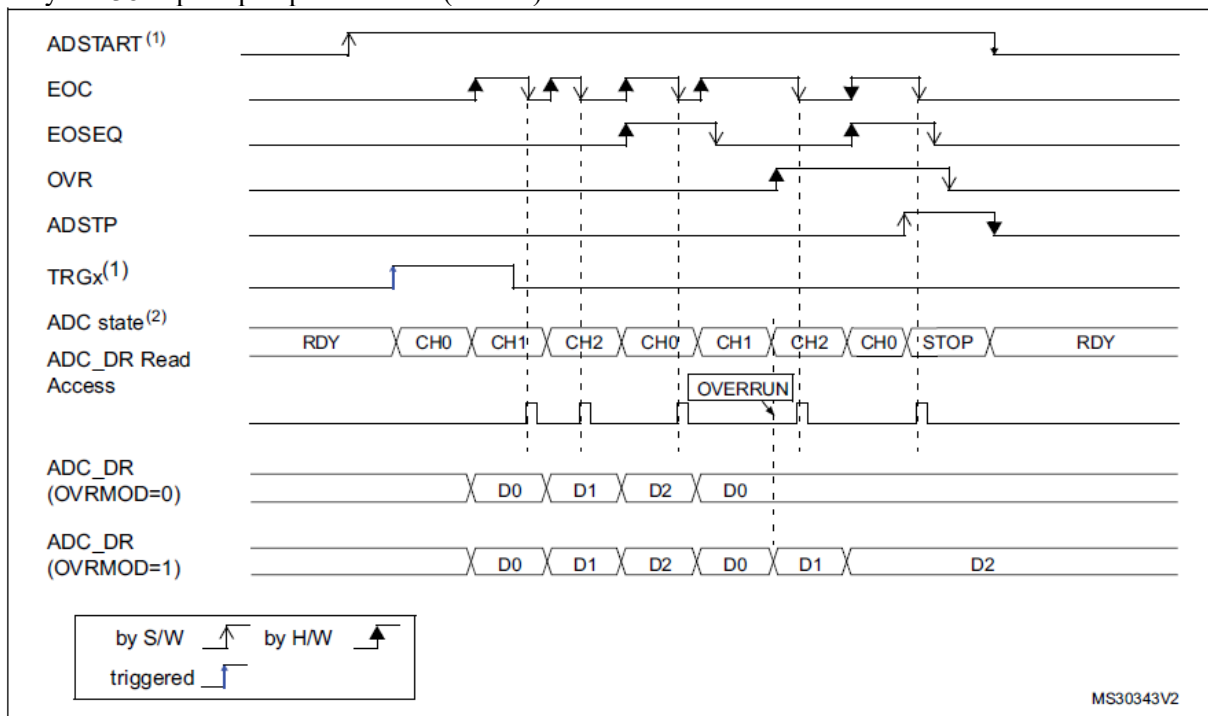
— Событие переполнения сохраняет регистр данных от перезаписывания: старые данные сохраняются, и новое преобразование отбрасывается. Если OVR остается на 1, могут быть выполнены дополнительные преобразования, но результирующие данные будут отброшены.

- OVRMOD = 1

— Регистр данных перезаписывается последним результатом преобразования, и предыдущие непрочитанные данные теряются. Если OVR остается на 1, могут быть выполнены дополнительные преобразования

И регистр ADC_DR всегда содержит данные из последнего преобразования.

Рисунок 38. Пример переполнения (OVER)



13.6.3 Управление последовательностью преобразованных данных без использования DMA

Если преобразования выполняются достаточно медленно, последовательность преобразования может обрабатываться программным обеспечением.

В этом случае программное обеспечение должно использовать флаг EOC и связанное с ним прерывание для обработки каждого результата данных. Каждый раз, когда преобразование завершено, бит EOC устанавливается в регистр ADC_ISR, и регистр ADC_DR может быть прочитан. Бит OVRMOD в регистре ADC_CFGR1 должен быть настроен на 0, чтобы управлять событиями переполнения как ошибка.

13.6.4 Управление преобразованными данными без использования DMA без переполнения

Может быть полезно разрешить преобразование одного или нескольких каналов ADC без чтения данных после каждого преобразования. В этом случае бит OVRMOD должен быть сконфигурирован в 1, а флаг OVR должен быть проигнорирован программным обеспечением. Когда OVRMOD = 1, событие переполнения не препятствует продолжению преобразования ADC, и регистр ADC_DR всегда содержит последние данные преобразования.

13.6.5 Управление преобразованными данными с использованием DMA

Поскольку все преобразованные значения каналов хранятся в одном регистре данных, эффективно использовать DMA при конвертировании более одного канала. Это позволяет избежать потери результатов преобразования данных, хранящихся в регистре ADC_DR.

Когда режим DMA включен (бит DMAEN установлен в 1 в регистре ADC_CFGR1), запрос DMA генерируется после преобразования каждого канала. Это позволяет передавать преобразованные данные из регистра ADC_DR в место назначения, выбранное программным обеспечением.

Примечание: бит DMAEN в регистре ADC_CFGR1 должен быть установлен после фазы калибровки ADC

Несмотря на это, если происходит переполнение ($OVR = 1$), поскольку DMA не может своевременно обслуживать запрос на перенос DMA, ADC прекращает генерировать запросы DMA и данные, соответствующие новому преобразованию, не передаются DMA. Это означает, что все данные, переданные в ОЗУ, могут считаться действительными.

В зависимости от конфигурации бита OVRMOD данные либо сохраняются, либо перезаписываются (см. Раздел 13.6.2: переполнение ADC (OVR, OVRMOD) на стр. 244).

Запросы на передачу DMA блокируются до тех пор, пока программное обеспечение не очистит бит OVR.

В зависимости от использования приложения предлагаются два разных режима DMA и они конфигурируются с помощью бита DMACFG в регистре ADC_CFGR1:

- Режим однократного DMA (DMACFG = 0). Этот режим следует выбирать, когда DMA запрограммирован на передачу фиксированного количества слов данных.

- Круговой режим DMA (DMACFG = 1)

Этот режим следует выбирать при программировании DMA в циклическом режиме или в режиме двойного буфера.

Режим однократного DMA (DMACFG = 0)

В этом режиме ADC генерирует запрос передачи DMA каждый раз, когда доступно новое слово данных преобразования, и прекращает генерировать запросы DMA, как только DMA достиг последней передачи DMA (когда происходит прерывание DMA_EOT, см. Раздел 10: Контроллер доступа к прямой памяти (см. DMA) на стр. 188), даже если преобразование было начато заново.

Пример кода см. в разделе Приложение А.7.9: Пример кода последовательности DMA в режиме одноразового использования.

Когда передача DMA завершена (все передачи, сконфигурированные в контроллере DMA, выполнены):

- Зарезервировано содержимое регистра данных ADC.
- Любое продолжающееся преобразование прерывается и его частичный результат отбрасывается

- Новый запрос DMA не выдается контроллеру DMA. Это позволяет избежать возникновения ошибки переполнения, если все еще есть запущенные преобразования.

- Последовательность сканирования остановлена и сброшена

- DMA остановлен

Круговой режим DMA (DMACFG = 1)

В этом режиме ADC генерирует запрос передачи DMA каждый раз, когда новое слово данных преобразования доступно в регистре данных, даже если DMA достиг последней передачи DMA.

Это позволяет настроить DMA в циклическом режиме для обработки непрерывного потока аналоговых входных данных.

Пример кода см. в разделе Приложение А.7.10: Пример кода последовательности циклического режима DMA.

13.7 Маломощные функции

13.7.1 Преобразование режима ожидания

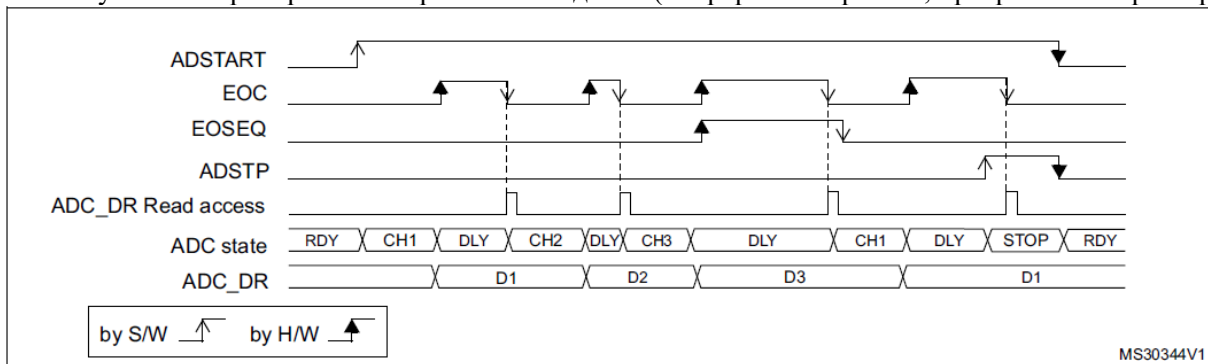
Преобразование режима ожидания может быть использовано для упрощения программного обеспечения, а также для оптимизации производительности приложений, работающих на низкой частоте, где может возникнуть риск переполнения ADC.

Когда бит WAIT установлен в 1 в регистре ADC_CFGR1, новое преобразование может начаться только в том случае, если предыдущие данные были обработаны, как только регистр ADC_DR был прочитан или бит EOC был очищен.

Это способ автоматически адаптировать скорость ADC к скорости системы, которая считывает данные.

Примечание: любые аппаратные триггеры, возникающие во время преобразования или в течение времени ожидания, предшествующего доступу на чтение, игнорируются.

Рисунок 39. Преобразование режима ожидания (непрерывный режим, программный триггер)



1. EXTEN = 00, CONT = 1
2. CHSEL = 0x3, SCANDIR = 0, WAIT = 1, AUTOFF = 0

Пример кода см. в разделе Приложение А.7.11: Пример кода последовательности ожидания.

13.7.2 Режим автоматического выключения (AUTOFF)

ADC имеет автоматическую функцию управления питанием, которая называется режимом автоматического выключения, и включается установкой AUTOFF = 1 в регистре ADC_CFGR1.

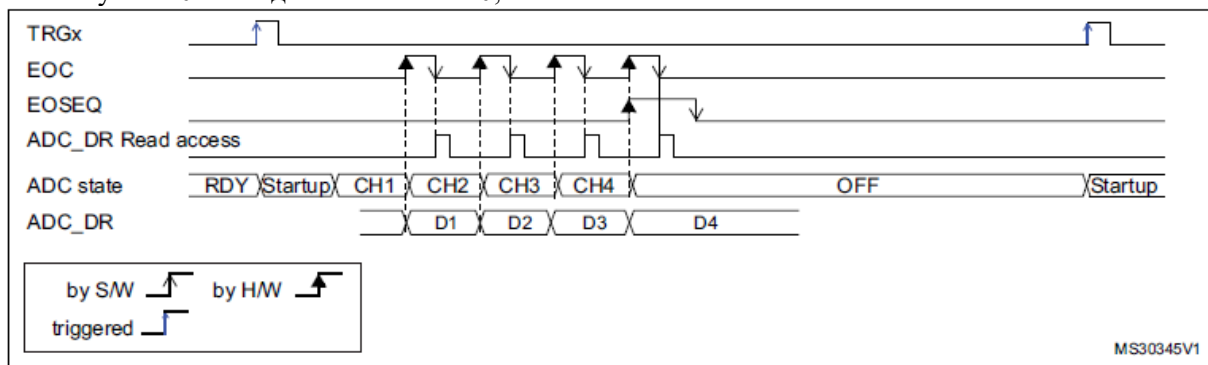
Когда AUTOFF = 1, ADC всегда отключается, когда не преобразовывается и автоматически пробуждается при запуске преобразования (программным или аппаратным триггером). Время запуска автоматически добавляется между событием запуска, которое запускает преобразование, и временем выборки ADC. Затем ADC автоматически отключается, как только последовательность преобразований завершается.

Режим автоматического выключения может привести к резкому сокращению энергопотребления приложений, которым требуется относительно небольшое количество преобразований, или когда запросы преобразования рассчитаны достаточно далеко друг от друга (например, с аппаратным триггером низкой частоты), чтобы оправдать дополнительную мощность и дополнительное время, используемое для переключения. Включается и отключается ADC.

Режим автоматического выключения можно комбинировать с преобразованием режима ожидания (WAIT = 1) для приложений, работающих так низкочастотно. Эта комбинация может обеспечить значительную экономию энергии, если ADC автоматически отключается на фазе ожидания и перезапускается, как только регистр ADC_DR считывается приложением (см. Рис. 41: Поведение с WAIT = 1, AUTOFF = 1).

Примечание. Обратитесь к разделу 6: «Сброс и управление синхронизацией» (RCC) на стр. 93 для описания того, как управлять выделенным внутренним генератором на 14 МГц. Интерфейс ADC может автоматически включать и выключать внутренний генератор на 14 МГц для экономии энергии.

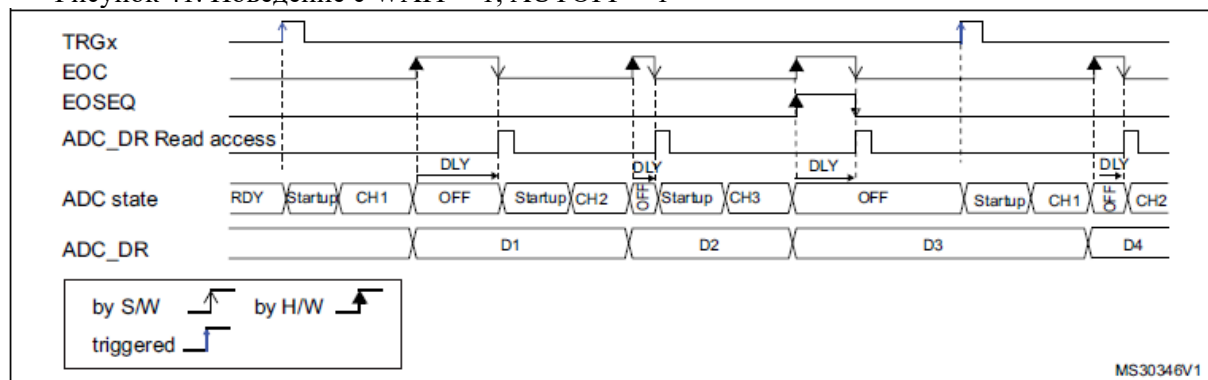
Рисунок 40. Поведение с WAIT = 0, AUTOFF = 1



1. EXTSEL = TRGx, EXTEN = 01 (нарастающий фронт), CONT = x, ADSTART = 1, CHSEL = 0xF, SCANDIR = 0, WAIT = 1, AUTOFF = 1

Пример кода см. в разделе Приложение А.7.12: «Автовыключение» и пример кода последовательности режима ожидания.

Рисунок 41. Поведение с WAIT = 1, AUTOFF = 1



1. EXTSEL = TRGx, EXTEN = 01 (нарастающий фронт), CONT = x, ADSTART = 1, CHSEL = 0xF, SCANDIR = 0, WAIT = 1, AUTOFF = 1

Пример кода см. в разделе Приложение А.7.13: Пример кода автоматического отключения и режима ожидания.

13.8 Аналоговый сторожевой таймер (AWDEN, AWDSGL, AWDCH, AWD_HTR / LTR, AWD)

Функция аналогового сторожевого таймера AWD включается установкой бита AWDEN в регистре ADC_CFGR1. Он используется для контроля того, что один выбранный канал или все задействованные каналы (см. Таблицу 48: выбор канала аналогового сторожевого таймера) остаются в пределах сконфигурированного диапазона напряжений (окна), как показано на рисунке 42.

Бит статуса аналогового сторожевого таймера AWD устанавливается, если аналоговое напряжение, преобразованное ADC, ниже нижнего порога или выше более высокого порога. Эти пороги запрограммированы в 12 младших значащих битах 16-разрядных регистров ADC_HTR и ADC_LTR. Прерывание может быть включено путем установки бита AWDIE в регистре ADC_IER.

Флаг AWD очищается программным обеспечением, записывая в него 1.

При преобразовании данных с разрешением менее 12 бит (в соответствии с битами DRES [1:0]) LSB запрограммированных пороговых значений должен быть очищен, так как внутреннее сравнение всегда выполняется на полном 12-битном коде необработанных данных (выравнивание по левому краю).

Пример кода см. в разделе Приложение А.7.14: Пример аналогового сторожевого таймера.

Таблица 47 описывает, как выполняется сравнение для всех возможных разрешений.

Таблица 47. Сравнение аналоговых контрольных сумм

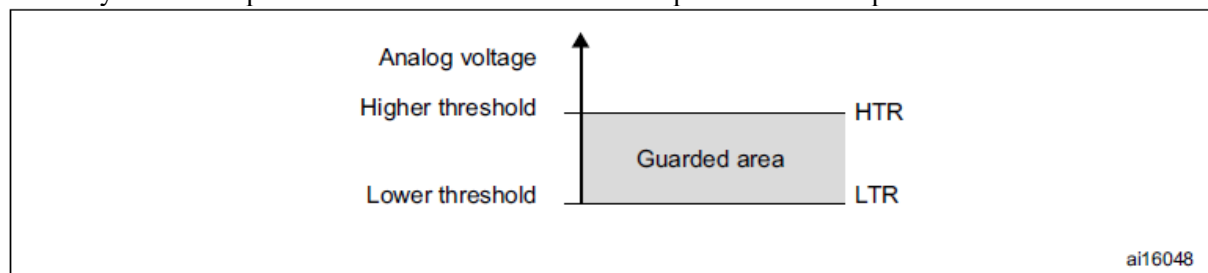
Разрешение Бит RES [1: 0]	Сравнение аналогового сторожевого таймера:		Комментарии
	Исходные преобразованные данные, выравнивание по левому краю (1)	Пороги	
00: 12-bit	DATA [11: 0]	LT [11: 0] и HT [11: 0]	-
01: 10-bit	DATA [11: 2],	00LT [11: 0] и HT [11: 0]	Пользователь должен настроить LT1 [1: 0] и HT1 [1: 0]
10: 8-bit	DATA [11: 4], 0000	LT [11: 0] и HT [11: 0]	Пользователь должен настроить LT1 [3: 0] и HT1 [3: 0] на "0000"
11: 6-bit	DATA [11: 6], 000000	LT [11: 0] и HT [11: 0]	Пользователь должен настроить LT1 [5: 0] и HT1 [5: 0] на "000000"

1. Сравнение сторожевого таймера выполняется на необработанных преобразованных данных перед любым вычислением выравнивания.

В таблице 48 показано, как настроить биты AWDSGL и AWDEN в ADC_CFGR1

Чтобы включить аналоговый сторожевой таймер на одном или нескольких каналах.

Рисунок 42. Охраняемая область аналогового сторожевого таймера



Threshold Порог Guarded area Охраняемая территория

Таблица 48. Выбор канала аналогового сторожевого таймера

Каналы, защищенные битом аналогового watchdog	AWDSGL bit	AWDEN bit
None	x	0
Все каналы	0	1
Одиночный (1) канал	10	1

1. Выбирается битами AWDCN [4: 0]

13.9 Датчик температуры и внутреннее опорное напряжение

Температурный датчик может использоваться для измерения температуры соединения (T_J) устройства. Датчик температуры внутренне подключен к входному каналу ADC_IN16, который используется для преобразования выходного напряжения датчика в цифровое значение. Время дискретизации аналогового вывода датчика температуры должно быть больше минимального значения TS_{temp}, указанного в таблице данных. Когда он не используется, датчик может быть переведен в режим пониженного энергопотребления.

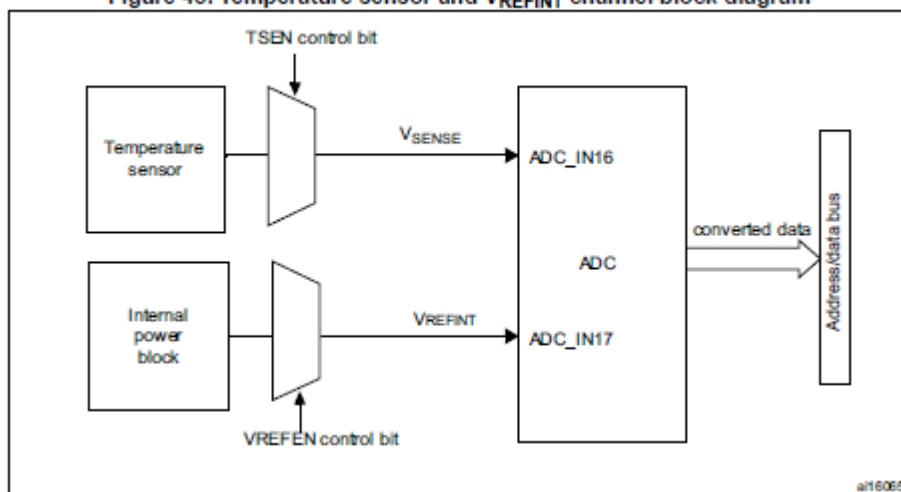
Выходное напряжение датчика температуры изменяется линейно с температурой, однако его характеристики могут значительно отличаться от чипа к чипу из-за технологических изменений. Для повышения точности датчика температуры (особенно для измерения абсолютной температуры), калибровочные значения индивидуально измеряются для каждой детали посредством ST во время производственных испытаний и сохраняются в области системной памяти. Дополнительную информацию см. В отдельном техническом описании устройства.

Внутреннее задание напряжения (VREFINT) обеспечивает стабильный (запрещенный) выход напряжения для ADC и компараторов. VREFINT внутренне подключен к входному каналу ADC_IN17. Точное напряжение VREFINT индивидуально измеряется для каждого ЧИПА посредством ST во время производственных испытаний и сохраняется в области системной памяти.

На рис. 43 показана блок-схема соединений между температурным датчиком, внутренним опорным напряжением и ADC.

Бит TSEN должен быть установлен, чтобы разрешить преобразование ADC_IN16 (датчик температуры), и бит VREFEN должен быть установлен для разрешения преобразования ADC_IN17 (VREFINT).

Figure 43. Temperature sensor and VREFINT channel block diagram



Чтение температуры

1. Выберите входной канал ADC_IN16
2. Выберите подходящее время выборки, указанное в таблице данных устройства (TS_temp).
3. Установите бит TSEN в регистре ADC_CCR, чтобы вывести температурный датчик из режима выключения и дождитесь его времени стабилизации (tSTART)

Пример кода см. в разделе Приложение A.7.15: Пример кода конфигурации температуры.

4. Запустите преобразование ADC, установив бит ADSTART в регистре ADC_CR (или внешним триггером)
5. Считайте полученные данные в регистре ADC_DR
6. Рассчитайте фактическую температуру по следующей формуле:

$$\text{Temperature (in } ^\circ\text{C)} = \frac{110\text{ } ^\circ\text{C} - 30\text{ } ^\circ\text{C}}{\text{TS_CAL2} - \text{TS_CAL1}} \times (\text{TS_DATA} - \text{TS_CAL1}) + 30\text{ } ^\circ\text{C}$$

Где:

- TS_CAL2 — значение калибровки датчика температуры, полученное при 110 °C
- TS_CAL1 — это калибровочное значение датчика температуры, полученное при 30 °C
- TS_DATA — это фактическое значение выходного сигнала датчика температуры, преобразованное ADC

Дополнительную информацию о точках калибровки TS_CAL1 и TS_CAL2 см. В отдельном техническом описании устройства.

Пример кода см. В примере A.7.16: Пример кода вычисления температуры.

Примечание. Датчик имеет время запуска после пробуждения из режима выключения, прежде чем он сможет вывести VSENSE на нужном уровне. ADC также имеет время запуска после включения питания, поэтому, чтобы свести к минимуму задержку, биты ADEN и TSEN должны быть установлены в одно и то же время.

Вычисление фактического напряжения VDDA с использованием внутреннего опорного напряжения

Напряжение питания VDDA, подаваемое на микроконтроллер, может быть изменено или неизвестно точно. Встроенный внутренний опорный сигнал напряжения (VREFINT) и его калибровочные данные, полученные ADC в процессе производства на VDDA = .3 V, могут использоваться для оценки фактического уровня напряжения VDDA.

Следующая формула дает фактическое напряжение VDDA, подающее устройство:

$$\text{VDDA} = .3\text{ V} \times \text{VREFINT_CAL} / \text{VREFINT_DATA}$$

Где:

- VREFINT_CAL — это значение калибровки VREFINT

• VREFINT_DATA — это фактическое выходное значение VREFINT, преобразованное ADC

Преобразование относительного отношения ADC к общему абсолютному значению напряжения

ADC предназначен для передачи цифрового значения, соответствующего отношению между аналоговым источником питания и напряжением, приложенным к преобразованному каналу. Для большинства случаев использования приложения необходимо преобразовать это отношение в напряжение, независимое от VDDA. Для приложений, где VDDA известен, и преобразованные значения ADC выровнены по правому краю, для получения этого абсолютного значения можно использовать следующую формулу:

$$V_{\text{CHANNEL}x} = \frac{V_{\text{DDA}}}{\text{FULL_SCALE}} \times \text{ADC_DATA}_x$$

Для приложений, где значение VDDA неизвестно, вы должны использовать внутреннее опорное напряжение, а VDDA можно заменить выражением, приведенным в разделе Расчет фактического напряжения VDDA с использованием внутреннего опорного напряжения, в результате получим следующую формулу:

$$V_{\text{CHANNEL}x} = \frac{3.3 \text{ V} \times \text{VREFINT_CAL} \times \text{ADC_DATA}_x}{\text{VREFINT_DATA} \times \text{FULL_SCALE}}$$

Где:

- VREFINT_CAL — это значение калибровки VREFINT
- ADC_DATA_x — это значение, измеренное ADC на канале x (выровнено по правому краю)
- VREFINT_DATA — это фактическое выходное значение VREFINT, преобразованное ADC
- full_SCALE — максимальное цифровое значение выхода ADC. Например, с 12-битным разрешением это будет $2^{12} - 1 = 4095$ или с 8-битным разрешением, $2^8 - 1 = 255$.

Примечание. Если измерения ADC выполняются с использованием выходного формата, отличного от 12-разрядного выровненного по правому краю, все параметры сначала должны быть преобразованы в совместимый формат перед выполнением расчета.

13.10 Контроль напряжения батареи

Бит VBATEN в регистре ADC_CCR позволяет приложению измерять напряжение резервной батареи на выводе VBAT. Так как напряжение VBAT может быть выше, чем VDDA, для обеспечения правильной работы ADC вывод VBAT внутренне подключен к делителю моста на 2. Этот мост автоматически активируется, когда VBATEN установлен, для подключения VBAT / 2 к ADC_IN18 Входной канал. Как следствие, преобразованное цифровое значение составляет половину напряжения VBAT. Чтобы предотвратить нежелательное потребление батареи, рекомендуется включить мостовой делитель только тогда, когда это необходимо для преобразования ADC

13.11. Прерывания ADC

Прерывание может быть сгенерировано любым из следующих событий:

- Питание ADC, когда ADC готов (флаг ADRDY)
- Конец преобразования (флаг EOC)
- Конец последовательности преобразований (флаг EOSEQ)
- Когда происходит обнаружение аналогового сторожевого таймера (флаг AWD)
- Когда заканчивается фаза выборки (флаг EOSMP)
- при переполнении данных (флаг OVR)

Для обеспечения гибкости доступны отдельные биты разрешения прерываний.

Таблица 49. Прерывания ADC

Table 49. ADC interrupts

Interrupt event	Event flag	Enable control bit
ADC ready	ADRDY	ADRDYIE
End of conversion	EOC	EOCIE
End of sequence of conversions	EOSEQ	EOSEQIE
Analog watchdog status bit is set	AWD	AWDIE
End of sampling phase	EOSMP	EOSMPIE
Overrun	OVR	OVRIE

ADC готов к ADRDYADRDYIE

Конец преобразования EOCEOCIE

Конец последовательности преобразований EOSEQEOSEQIE

Установлен бит состояния аналогового сторожевого таймера AWD AWDIE

Конец этапа выборки EOSMPEOSMPIE

Перезапуск OVR OVRIE

13.12 Регистры ADC

См. Раздел 1.1 на стр. 42 для получения списка сокращений, используемых в описаниях регистров.

13.12.1. Регистр прерываний и состояния ADC (ADC_ISR)

Смещение адреса: 0x00

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD	Res.	Res.	OVR	EOSEQ	EOC	EOSMP	ADRDY
								rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

Биты 31: 8 Зарезервированы, должны храниться в значении сброса.

Бит 7 AWD: флаг аналогового сторожевого таймера

Этот бит устанавливается аппаратно, когда преобразованное напряжение пересекает значения, запрограммированные в регистрах ADC_LTR и ADC_HTR. Он очищается программным обеспечением, записывающим 1.

0: Аналогового сторожевого события не произошло (или событие флага уже было подтверждено и очищено программным обеспечением)

1: произошло событие аналогового сторожевого таймера

Бит 6: 5 Зарезервирован, должен быть сохранен в значении сброса.

Бит 4 OVR: переполнение ADC

Этот бит устанавливается аппаратными средствами, когда происходит переполнение, что означает, что новое преобразование завершено, а флаг EOC уже установлен. Он очищается программным обеспечением, записывающим 1.

0: переполнение не произошло (или событие флага уже было подтверждено и очищено программным обеспечением)

1: произошло переполнение

Бит 3 EOSEQ: Флаг завершения последовательности

Этот бит устанавливается аппаратными средствами в конце преобразования последовательности каналов, выбранных битами CHSEL. Он очищается программным обеспечением, записывающим 1.

0: последовательность преобразования не завершена (или событие флага уже было подтверждено и очищено программным обеспечением)

1: последовательность преобразования завершена

Бит 2 EOS: флаг окончания конвертации

Этот бит устанавливается аппаратными средствами в конце каждого преобразования канала, когда новый результат данных доступен в регистре ADC_DR. Он очищается программным обеспечением, записывающим в него 1 или считывая регистр ADC_DR.

0: преобразование канала не завершено (или событие флага уже подтверждено и очищено программным обеспечением)

1: Преобразование канала завершено

Бит 1 EOSMP: Флаг завершения отсчета

Этот бит устанавливается аппаратно во время преобразования, в конце фазы выборки. Он очищается программным обеспечением путем программирования его на «1».

0: не достигнуто окончание фазы выборки (или событие флага уже было подтверждено и очищено программным обеспечением)

1: Окончание фазы выборки достигнуто

Бит 0 ADRDY: готов ADC

Этот бит устанавливается аппаратно после включения ADC (бит ADEN = 1), и когда ADC достигает состояния, когда он готов принять запросы на преобразование. Он очищается программным обеспечением, записывающим 1.

0: ADC еще не готов начать преобразование (или событие флага уже было подтверждено и очищено программным обеспечением)

1: ADC готов начать преобразование

Примечание: в режиме автоматического выключения (AUTOFF = 1) фазы включения / выключения выполняются автоматически, аппаратно и флаг ADRDY не установлен.

13.12.2. Регистр разрешения прерываний ADC (ADC_IER)

Смещение адреса: 0x04

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD IE	Res.	Res.	OVRIE	EOSEQ IE	EOCIE	EOSMP IE	ADRDY IE
								rw			rw	rw	rw	rw	rw

Биты

31: 8 Зарезервированы, должны храниться в значении сброса.

Бит 7 AWDIE: Разрешение прерывания аналогового сторожевого таймера

Этот бит устанавливается и очищается программным обеспечением для включения / выключения аналогового прерывания сторожевого таймера.

0: прерывание аналогового сторожевого таймера отключено

1: Разрешение аналогового сторожевого таймера включено

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 6: 5 Зарезервирован, должен быть сохранен при значении сброса.

Бит 4 OVRIE: Разрешение прерывания переполнения

Этот бит устанавливается и очищается программным обеспечением для включения / выключения прерывания переполнения.

0: Прерывание переполнения отключено

1: Прерывание переполнения включено. Прерывание генерируется, когда установлен бит OVR.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 3 EOSEQIE: Разрешение прерывания завершения последовательности преобразования
Этот бит устанавливается и очищается программным обеспечением для включения / выключения конца последовательности прерываний преобразования.

0: прерывание EOSEQ отключено

1: Включено прерывание EOSEQ. Прерывание генерируется, когда бит EOSEQ установлен.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 2 EOCIE: Разрешение прерывания завершения преобразования

Этот бит устанавливается и очищается программным обеспечением для включения / выключения конца прерывания преобразования.

0: прерывание EOC отключено

1: Включено прерывание EOC. Прерывание генерируется, когда бит EOC установлен.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 1 EOSMPIE: Конец разрешения прерывания флага выборки

Этот бит устанавливается и очищается программным обеспечением для включения / выключения конца прерывания фазы выборки.

0: Прерывание EOSMP отключено.

1: Включено прерывание EOSMP. Прерывание генерируется, когда бит EOSMP установлен.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Биты 0 ADRDYIE: Разрешение готовности прерывания ADC

Этот бит устанавливается и очищается программным обеспечением для включения / выключения прерывания ADC.

0: прерывание ADRDY отключено.

1: Прерывание ADRDY включено. Прерывание генерируется, когда бит ADRDY установлен.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

13.12.3 Регистр управления ADC (ADC_CR)

Смещение адреса: 0x08

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AD CAL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rs															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADSTP	Res.	ADSTAR	ADDIS	ADEN
											rs		rs	rs	rs

Бит 31 ADCAL: калибровка ADC

Этот бит устанавливается программным обеспечением, чтобы начать калибровку ADC.

После завершения калибровки он очищается аппаратными средствами.

0: калибровка завершена

1: запись 1 для калибровки ADC. Чтение на 1 означает, что выполняется калибровка.

Примечание: программному обеспечению разрешено устанавливать ADCAL только при отключенном ADC (ADCAL = 0, ADSTART = 0, ADSTP = 0, ADDIS = 0 и ADEN = 0).

Биты 30: 5 Зарезервированы, должны храниться в значении сброса.

Бит 4 ADSTP: команда прекращения преобразования ADC

Этот бит устанавливается программным обеспечением, чтобы остановить и отменить текущее преобразование (команда ADSTP).

Он очищается аппаратными средствами, когда преобразование эффективно сбрасывается, и ADC готов принять новую команду начального преобразования.

0: команда преобразования останова ADC не выполняется

1: Запишите 1, чтобы остановить ADC. Чтение 1 означает, что выполняется команда ADSTP.

Примечание. Установка ADSTP в значение «1» действует только в том случае, если ADSTART = 1 и ADDIS = 0 (ADC включен и может преобразовываться, и нет запроса на отключение ADC)

Бит 3 Зарезервирован, должен быть сохранен при значении сброса.

Бит 2 Команда ADSTART: команда запуска преобразования ADC

Этот бит устанавливается программным обеспечением для запуска преобразования ADC. В зависимости от битов конфигурации EXTEN [1: 0], конвертация либо начинается немедленно (конфигурация программного триггера), либо когда происходит аппаратное триггерное событие (конфигурация аппаратного запуска).

Он очищается аппаратными средствами:

— В режиме одиночного преобразования (CONT = 0, DISCEN = 0), когда выбран программный триггер (EXTEN = 00):

При утверждении флага окончания последовательности конверсии (EOSEQ).

— В режиме прерывистого преобразования (CONT = 0, DISCEN = 1), когда выбран программный триггер (EXTEN = 00): при утверждении флага завершения конверсии (EOC).

— Во всех остальных случаях: после выполнения команды ADSTP, одновременно с битом ADSTP

Очищается аппаратными средствами.

0: Преобразование ADC не выполняется.

1: Запишите 1, чтобы запустить ADC. Чтение 1 означает, что ADC работает и может преобразовываться.

Примечание: программному обеспечению разрешено устанавливать ADSTART только в том случае, если ADEN = 1 и ADDIS = 0 (ADC включен и нет ожидающего запроса для отключения ADC)

Бит 1 Команда ADDIS: ADC disable

Этот бит устанавливается программным обеспечением для отключения команды ADC (ADDIS) и перевода ее в состояние выключения питания

(Состояние ВЫКЛЮЧЕНО).

Он очищается аппаратными средствами, как только ADC фактически отключается (ADEN также очищается аппаратными средствами в это время).

0: команда ADDIS не выполняется

1: Запишите 1, чтобы отключить ADC. Чтение 1 означает, что выполняется команда ADDIS.

Примечание. Установка ADDIS на «1» эффективна только тогда, когда ADEN = 1 и ADSTART = 0 (что гарантирует, что преобразование не выполняется)

Бит 0 ADEN: команда разрешения ADC

Этот бит устанавливается программным обеспечением для включения ADC. ADC будет эффективно готов к работе после Флаг ADRDY установлен.

Он очищается аппаратными средствами, когда ADC отключен после выполнения команды ADDIS.

0: ADC отключен (состояние ВЫКЛ)

1: запись 1 для включения ADC.

Примечание: программному обеспечению разрешено устанавливать ADEN только тогда, когда все биты регистров ADC_CR равны 0 (ADCAL = 0, ADSTP = 0, ADSTART = 0, ADDIS = 0 и ADEN = 0)

13.12.4 Регистр конфигурации ADC 1 (ADC_CFGR1)

Смещение адреса: 0x0C

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWDCH[4:0]					Res.	Res.	AWDEN	AWDSGL	Res.	Res.	Res.	Res.	Res.	DISCEN
	rw	rw	rw	rw	rw			rw	rw						rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUTOFF	WAIT	CONT	OVRMOD	EXTEN[1:0]		Res.	EXTSEL[2:0]		ALIGN	RES[1:0]	SCAND IR	DMAC FG	DMAEN		
rw	rw	rw	rw	rw			rw		rw	rw	rw	rw	rw		rw

Бит 31 Зарезервирован, должен храниться в значении сброса.

Биты 30:26 AWDCH [4: 0]: выбор канала аналогового сторожевого таймера

Эти биты устанавливаются и очищаются с помощью программного обеспечения. Они выбирают входной канал, который должен охраняться аналоговым сторожевым таймером.

00000: аналоговый вход ADC Канал 0, контролируемый AWD

00001: аналоговый вход ADC Канал 1, контролируемый AWD

.....

10010: аналоговый вход ADC Канал 18, контролируемый AWD

Другие значения: Зарезервировано, не должно использоваться

Примечание: Канал, выбранный битами AWDCH [4: 0], также должен быть установлен в регистр CHSELR

Примечание: программному обеспечению разрешено записывать эти биты только тогда, когда ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Биты 25:24 Зарезервированы, должны храниться по значению сброса.

Бит 23 AWDEN: Разрешение аналогового сторожевого таймера

Этот бит устанавливается и очищается программным обеспечением.

0: Аналоговый сторожевой таймер отключен

1: Включен аналоговый сторожевой таймер

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 22 AWDSGL: включение сторожевого таймера по отдельному каналу или по всем каналам

Этот бит устанавливается и очищается программным обеспечением для включения аналогового сторожевого таймера на канале, идентифицированном битами AWDCH [4: 0] или на всех каналах

0: Аналоговый сторожевой таймер включен во всех каналах

1: Аналоговый сторожевой таймер включен на одном канале

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Биты 21:17 Зарезервированы, должны храниться по значению сброса.

Бит 16 DISCEN: прерывистый режим

Этот бит устанавливается и очищается программным обеспечением для включения / выключения прерывистого режима.

0: прерывистый режим отключен

1: включен прерывистый режим

Примечание: невозможно включить как прерывистый режим, так и непрерывный режим: запрещено устанавливать оба бита DISCEN = 1 и CONT = 1.

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 15 AUTOFF: режим автоматического выключения

Этот бит устанавливается и очищается программным обеспечением для включения / выключения режима автоматического выключения.

0: Режим автоматического отключения отключен

1: включен режим автоматического выключения

Примечание: программному обеспечению разрешено записывать этот бит только при $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Бит 14 WAIT: Ожидание режима преобразования

Этот бит устанавливается и очищается программным обеспечением для включения / выключения режима ожидания ожидания.

0: Ожидание режима конвертации

1: Ожидание режима конвертации

Примечание: программному обеспечению разрешено записывать этот бит только при $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Бит 13 CONT: режим одиночного / непрерывного преобразования

Этот бит устанавливается и очищается программным обеспечением. Если он установлен, преобразование происходит непрерывно, пока он не будет очищен.

0: Режим одиночного преобразования

1: Режим непрерывного преобразования

Примечание: невозможно включить как прерывистый режим, так и непрерывный режим: запрещено устанавливать оба бита $DISCEN = 1$ и $CONT = 1$.

Примечание: программному обеспечению разрешено записывать этот бит только при $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Бит 12 OVRMOD: режим управления переполнением

Этот бит устанавливается и очищается программным обеспечением и настраивается способ управления перерасходом данных.

0: Регистр ADC_DR сохраняется со старыми данными при обнаружении переполнения.

1: Регистр ADC_DR перезаписывается последним результатом преобразования при обнаружении переполнения.

Примечание: программному обеспечению разрешено записывать этот бит только при $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Биты 11:10 EXTEN [1: 0]: включение внешнего триггера и выбор полярности

Эти биты устанавливаются и очищаются программным обеспечением для выбора полярности внешнего триггера и включения триггера.

00: Обнаружение аппаратного триггера отключено (конверсии могут запускаться программным обеспечением)

01: Обнаружение аппаратного триггера по переднему краю

10: Обнаружение аппаратного триггера на заднем крае

11: Обнаружение аппаратного триггера как на восходящем, так и на спадающем фронтах

Примечание: программному обеспечению разрешено записывать эти биты только тогда, когда $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Бит 9 Зарезервирован, должен храниться в значении сброса.

Биты 8: 6 EXTSEL [2: 0]: выбор внешнего триггера

Эти биты выбирают внешнее событие, используемое для инициирования начала преобразования (подробности см. В таблице 45: внешние триггеры):

000: TRG0

001: TRG1

010: TRG2

011: TRG3

100: TRG4

101: TRG5

110: TRG6

111: TRG7

Примечание: программному обеспечению разрешено записывать эти биты только при $ADSTART = 0$ (что гарантирует, что преобразование не выполняется).

Бит 5 ALIGN: выравнивание данных

Этот бит устанавливается и очищается программным обеспечением для выбора выравнивания по правому или левому краю. См. Рис. 37: Выравнивание и разрешение данных на стр. 244

0: выравнивание по правому краю

1: выравнивание по левому краю

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 4: 3 RES [1: 0]: разрешение данных

Эти биты записываются программным обеспечением для выбора разрешения преобразования.

00: 12 бит

01: 10 бит

10: 8 бит

11: 6 бит

Примечание: программному обеспечению разрешено записывать эти биты только тогда, когда ADEN = 0.

Бит 2 SCANDIR: Направление последовательности сканирования

Этот бит устанавливается и очищается программным обеспечением, чтобы выбрать направление, в котором каналы будут сканироваться в последовательности.

0: Сканирование вверх (от CHSEL0 до CHSEL18)

1: обратное сканирование (от CHSEL18 до CHSEL0)

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 1 DMACFG: конфигурация прямого доступа к памяти

Этот бит устанавливается и очищается программным обеспечением для выбора между двумя режимами DMA и действует только тогда, когда DMAEN = 1.

0: выбран однократный режим DMA

1: выбран кольцевой режим DMA

Для получения дополнительной информации см. Раздел 13.6.5: Управление преобразованными данными с использованием DMA на стр. 245

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 0 DMAEN: разрешение прямого доступа к памяти

Этот бит устанавливается и очищается программным обеспечением, чтобы обеспечить генерацию запросов DMA. Это позволяет использовать контроллер DMA для автоматического управления преобразованными данными. Для получения дополнительной информации см. Раздел 13.6.5: Управление преобразованными данными с использованием DMA на стр. 245.

0: DMA отключено

1: DMA включен

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

13.12.5 Регистр конфигурации ADC 2 (ADC_CFGR2)

Смещение адреса: 0x10

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
СКМОДЕ[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

Биты 31:30 СКМОДЕ [1: 0]: тактовый режим ADC

Эти биты устанавливаются и сбрасываются программным обеспечением, чтобы определить, как работает аналоговый ADC:

00: ADCCLK (режим асинхронных часов), генерируемый на уровне продукта (см. Раздел RCC)

01: PCLK / 2 (синхронный тактовый режим)

10: PCLK / 4 (синхронный тактовый режим)

11: зарезервировано

Во всех режимах синхронных часов дрожание в задержке от триггера таймера до начала преобразования отсутствует.

Примечание: программному обеспечению разрешено записывать эти биты только при отключенном ADC (ADCAL = 0, ADSTART = 0, ADSTP = 0, ADDIS = 0 и ADEN = 0).

Биты 29: 0 Зарезервированы, должны храниться в значении сброса.

13.12.6 Регистр времени выборки ADC (ADC_SMPR)

Смещение адреса: 0x14

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMP[2:0]	
														rw	

Биты 31: 3 Зарезервированы, должны храниться по значению сброса.

Биты 2: 0 SMP [2: 0]: выбор времени выборки

Эти биты записываются программным обеспечением, чтобы выбрать время выборки, которое применяется ко всем каналам.

000: 1,5 тактов ADC

001: 7.5 ADC тактов

010: 13,5 тактов ADC

011: 28,5 тактовые циклы ADC

100: 41,5 тактов ADC

101: 55,5 тактов ADC

110: 71,5 тактов ADC

111: 239.5 Циклы синхронизации ADC

Примечание: программному обеспечению разрешено записывать эти биты только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

13.12.7 Контрольный пороговый регистр ADC (ADC_TR)

Смещение адреса: 0x20

Сбросить значение: 0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:28 Зарезервированы, должны храниться по значению сброса.

Бит 27:16 HT [11: 0]: более высокий порог аналогового сторожевого таймера

Эти биты записываются программным обеспечением, чтобы определить более высокий порог для аналогового сторожевого таймера. См. Раздел 13.8: Аналоговый сторожевой таймер (AWDEN, AWDSGL, AWDCH, AWD_HTR / LTR, AWD) на стр. 249

Примечание: программному обеспечению разрешено записывать эти биты только тогда, когда ADSTART = 0 (что гарантирует непрерывность noconversion).

Биты 15:12 Зарезервированы, должны храниться по значению сброса.

Бит 11: 0 LT [11: 0]: Нижний порог аналогового сторожевого таймера

Эти биты записываются программным обеспечением, чтобы определить нижний порог для аналогового сторожевого таймера. См. Раздел 13.8: Аналоговый сторожевой таймер (AWDEN, AWDSGL, AWDCH, AWD_HTR / LTR, AWD) на стр. 249

Примечание: программному обеспечению разрешено записывать эти биты только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

13.12.8 Регистр выбора канала ADC (ADC_CHSELR)

Смещение адреса: 0x28

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHSEL 18	CHSEL 17	CHSEL 16
													RW	RW	RW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHSEL 15	CHSEL 14	CHSEL 13	CHSEL 12	CHSEL 11	CHSEL 10	CHSEL 9	CHSEL 8	CHSEL 7	CHSEL 6	CHSEL 5	CHSEL 4	CHSEL 3	CHSEL 2	CHSEL 1	CHSEL 0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Биты 31:19 Зарезервированы, должны храниться по значению сброса.

Биты 18: 0 CHSELx: выбор канала-х

Эти биты записываются программным обеспечением и определяют, какие каналы являются частью последовательности каналов, подлежащих преобразованию.

0: Входной канал-х не выбран для преобразования

1: Входной канал-х выбран для преобразования

Примечание: программному обеспечению разрешено записывать эти биты только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

13.12.9 Регистр данных ADC (ADC_DR)

Смещение адреса: 0x40

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Биты 31:16 Зарезервированы, должны храниться по значению сброса.

Биты 15: 0 DATA [15: 0]: Преобразованные данные Эти биты доступны только для чтения. Они содержат результат преобразования из последнего преобразованного канала. Данные выравниваются влево или вправо, как показано на рисунке 37: Выравнивание и разрешение данных на стр. 244.

Как только калибровка завершена, DATA [6: 0] содержит калибровочный коэффициент.

13.12.10 Общий регистр конфигурации ADC (ADC_CCR)

Смещение адреса: 0x308

Сбросить значение: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBAT EN	TS EN	VREF EN	Res.	Res.	Res.	Res.	Res.	Res.
							rw	rw	rw						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

Биты 31:25 Зарезервированы, должны храниться по значению сброса.

Бит 24 VBATEN: включение VBAT

Этот бит устанавливается и очищается программным обеспечением для включения / выключения канала VBAT.

0: канал VBAT отключен.

1: включен VBAT-канал

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 23 TSEN: Разрешение датчика температуры

Этот бит устанавливается и очищается программным обеспечением для включения / выключения датчика температуры.

0: датчик температуры отключен

1: Датчик температуры включен

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Бит 22 VREFEN: Разрешение VREFINT

Этот бит устанавливается и очищается программным обеспечением для включения / выключения VREFINT.

0: VREFINT отключен

1: включен режим VREFINT

Примечание: программному обеспечению разрешено записывать этот бит только при ADSTART = 0 (что гарантирует, что преобразование не выполняется).

Биты 21: 0 Зарезервированы, должны храниться в значении сброса

13.12.11 Карта регистров ADC

В следующей таблице приведены регистры ADC.

