

## 22.4.2 Контакты ЦАП и внутренние сигналы

В состав ЦАП входят:

- До двух выходных каналов
- DACx\_OUTy можно отключить от выходного контакта и использовать как обычный GPIO

• Датчик dac\_outx может использовать внутреннее штырьковое соединение со встроенными периферийными устройствами, такими как компаратор, операционный усилитель и АЦП (при наличии).

- Выходной канал ЦАП с буферизацией или без буферизации
- Блок выборки и хранения и регистры работают в режиме остановки, используя источник тактового сигнала LSI/LSE (dac\_hold\_ck) для статического преобразования.

ЦАП включает в себя до двух отдельных выходных каналов. Каждый выходной канал может быть подключен к встроенным периферийным устройствам, таким как компаратор, операционный усилитель и АЦП (при наличии). В этом случае выходной канал ЦАП можно отключить от выходного вывода DACx\_OUTy, а соответствующий GPIO использовать для другой цели.

Выход ЦАП может быть буферизован или нет. Блок выборки и удержания и связанные с ним регистры могут работать в режиме остановки с использованием источника тактового сигнала LSI или LSE (dac\_hold\_ck).

## 22.4.3 Включение канала ЦАП

Каждый канал ЦАП можно включить, установив соответствующий бит ENx в регистре DAC\_CR. Затем канал DAC включается по истечении времени запуска tWAKEUP.

Бит DACxRDY устанавливается в регистре DAC\_SR, когда интерфейс DAC готов принимать данные. Запись новых данных или установка триггера не разрешены, если бит ENx установлен, а сигнал DACxRDY сброшен.

***Примечание.** Бит ENx включает только аналоговый канал ЦАПx. Цифровой интерфейс DAC channelx активен, даже если бит ENx сброшен.*

## 22.4.4 Формат данных ЦАП

В зависимости от выбранного режима конфигурации данные должны быть записаны в указанный регистр, как описано ниже:

- Один канал ЦАП

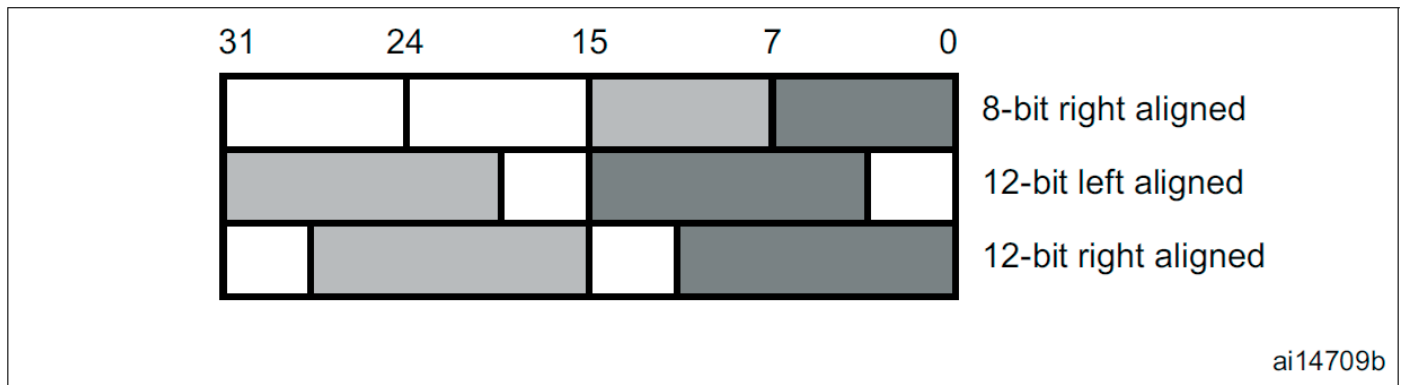
Есть три возможности:

- 8-битное выравнивание по правому краю: программа должна загрузить данные в биты DAC\_DHR8Rx[7:0] (сохраненные в битах DHRx[11:4])
- 12-битное выравнивание по левому краю: программа должна загрузить данные в биты DAC\_DHR12Lx [15:4] (сохраненные в битах DHRx[11:0]).
- 12-битное выравнивание по правому краю: программа должна загрузить данные в биты DAC\_DHR12Rx [11:0] (сохраненные в битах DHRx[11:0]).

В зависимости от загруженного регистра DAC\_DHRyуух данные, записанные пользователем, сдвигаются и сохраняются в соответствующем DHRx (регистр хранения данных x, который является внутренним регистром без ото-

бражения памяти). Затем регистр DHRx загружается в регистр DORx либо автоматически, с помощью программного триггера, либо с помощью триггера внешнего события.

Рисунок 157. Регистры данных в одноканальном режиме ЦАП



• Двойной канал ЦАП (при наличии)

Есть три возможности:

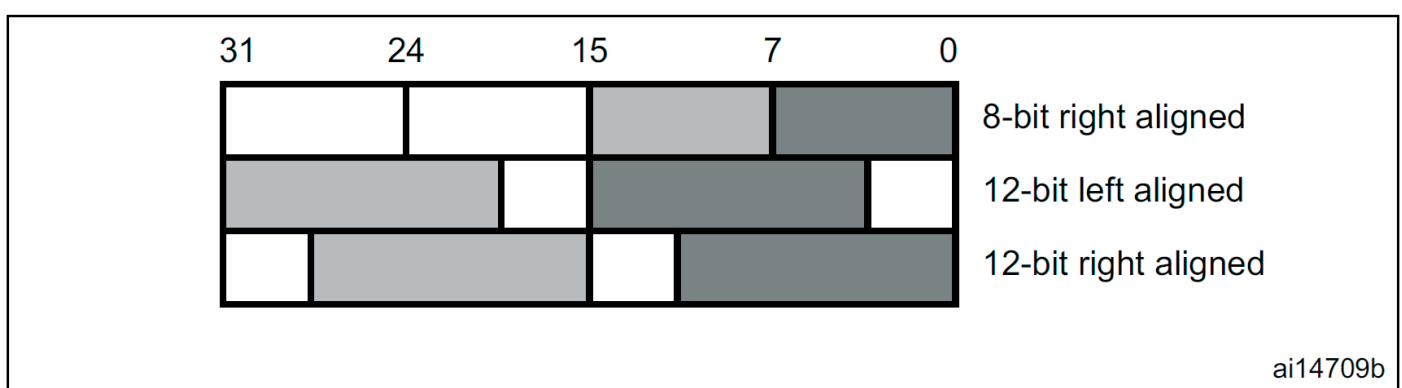
– 8-битное выравнивание по правому краю: данные для канала 1 ЦАП загружаются в биты DAC\_DHR8RD [7:0] (хранятся в битах DHR1 [11:4]), а данные для канала 2 ЦАП загружаются в биты DAC\_DHR8RD [15:8]. ] бит (хранится в битах DHR2[11:4])

– 12-битное выравнивание по левому краю: данные для канала 1 ЦАП загружаются в биты DAC\_DHR12LD [15:4] (хранятся в битах DHR1[11:0]), а данные для канала 2 ЦАП загружаются в биты DAC\_DHR12LD [31:20]. ] бит (хранится в битах DHR2[11:0])

– 12-битное выравнивание по правому краю: данные для канала 1 ЦАП загружаются в биты DAC\_DHR12RD [11:0] (хранятся в битах DHR1[11:0]), а данные для канала 2 ЦАП загружаются в биты DAC\_DHR12RD [27:16]. ] бит (хранится в битах DHR2[11:0])

В зависимости от загруженного регистра DAC\_DHR $uuu$ D данные, записанные пользователем, сдвигаются и сохраняются в DHR1 и DHR2 (регистры хранения данных, которые являются внутренними регистрами, не отображаемыми в память). Затем регистры DHR1 и DHR2 загружаются в регистры DAC\_DOR1 и DOR2, соответственно, либо автоматически, с помощью программного триггера, либо с помощью триггера внешнего события.

Рисунок 158. Регистры данных в двухканальном режиме ЦАП



### Данные со знаком/(unsigned) беззнаковые данные

Входные данные ЦАП беззнаковые: 0x000 соответствует минимальному значению, 0xFFF — максимальному для 12-битного режима.

ЦАП также может обрабатывать входные данные со знаком в формате дополнения до 2. Это делается установкой бита `SINFORMATx` в регистре `DAC_MCR`.

Когда бит `SINFORMATx` установлен, бит `MSB` данных, записываемых в регистры `DHRx`, инвертируется при копировании в регистр `DAC_DORx`, и интерфейс `DAC` может принимать подписанные данные (формат Q1.15, Q1.11 или Q1.7). Регистр `DAC_DHR12Lx` может использоваться для хранения 16-битных данных со знаком в регистрах хранения данных. 12 `MSB` 16-битных данных используются для выходных данных ЦАП, а бит `MSB` инвертируется. Четыре `LSB` просто игнорируются.

Таблица 187. Формат данных (случай 12-битных данных)

<code>SINFORMATx</code> bit	DATA written to <code>DHRx</code> register	DATA transfered to <code>DORx</code> register
0	0x000	0x000
0	0xFFF	0xFFF
1	0x7FF	0xFFF
1	0x000	0x800
1	0xFFF	0x7FF
1	0x800	0x000

### 22.4.5 ЦАП-преобразование

`DAC_DORx` не может быть записан напрямую, и любая передача данных в канал `DAC` `x` должна выполняться путем загрузки регистра `DAC_DHRx` (операция записи в `DAC_DHR8Rx`, `DAC_DHR12Lx`, `DAC_DHR12Rx`, `DAC_DHR8RD`, `DAC_DHR12RD` или `DAC_DHR12LD`).

Данные, хранящиеся в регистре `DAC_DHRx`, автоматически передаются в регистр `DAC_DORx` после одного такта `das_hclk`, если не выбран аппаратный триггер (бит `TENx` в регистре `DAC_CR` сброшен). Однако, когда выбран аппаратный триггер (бит `TENx` в регистре `DAC_CR` установлен) и происходит триггер, передача выполняется через три такта `das_hclk` после триггерного сигнала.

Когда `DAC_DORx` загружен содержимым `DAC_DHRx`, аналоговое выходное напряжение становится доступным через время `tSETTLING`, которое зависит от напряжения источника питания и нагрузки аналогового выхода.

Биты `HFSEL` `DAC_MCR` должны быть установлены, когда тактовая частота `das_hclk` превышает 80 МГц. Это добавляет дополнительную задержку к передаче из регистра `DAC_DHRx` в регистр `DAC_DORx`.

Обратитесь к описанию таблицы `HFSEL` ниже для ограничения скорости обновления `DAC_DORx` в зависимости от битов `HFSEL` и тактовой частоты `das_hclk`.

Если данные обновляются или происходит событие запуска программного/аппаратного обеспечения в течение неразрешенного периода, поведение периферийного устройства становится непредсказуемым.

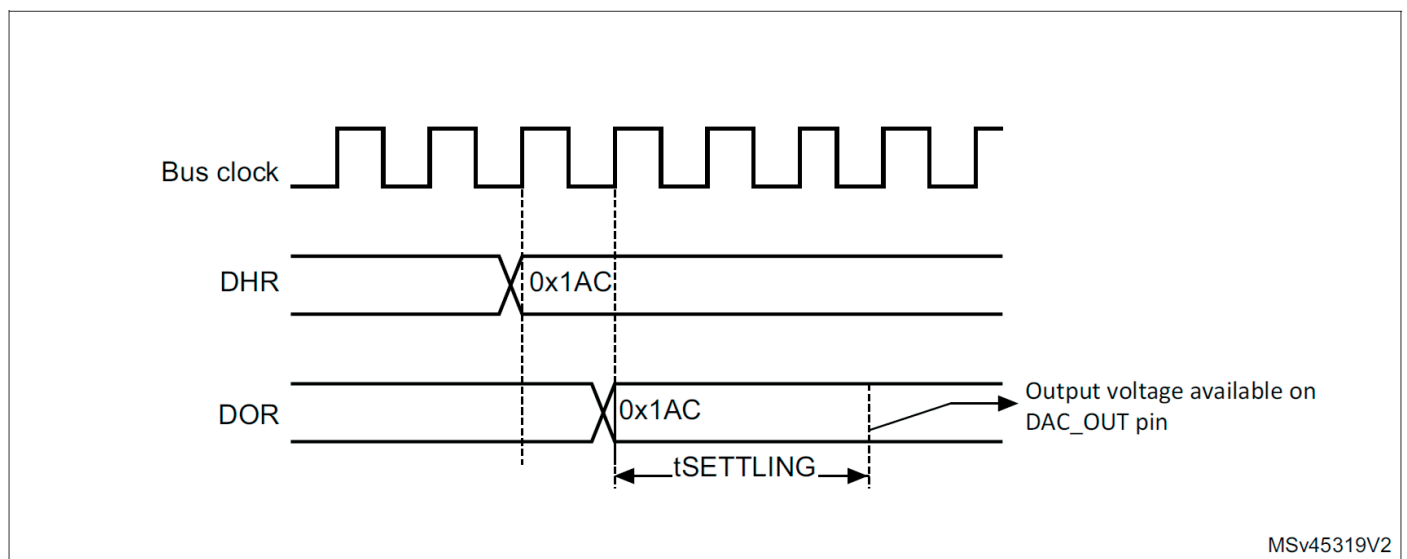
Вышеуказанная синхронизация связана только с ограничением интерфейса DAC. См. также значение параметра  $t_{SETTLING}$  в техническом описании изделия.

Табл. 188. Описание  $HFSEL$

HFSEL[1:0]	Частота АНВ	Функция
00	$\leq 80$ MHz	Частота обновления DAC_DOR до 3 тактов АНВ
01	$>80$ MHz <sup>(1)</sup>	Частота обновления DAC_DOR до 5 тактов АНВ
10	$>160$ MHz	Частота обновления DAC_DOR до 7 тактов АНВ
11	Reserved	-

<sup>1</sup> Обратитесь к техническому описанию устройства для получения значения максимальной частоты АНВ.

Рисунок 159. Временная диаграмма преобразования с отключенным триггером  $TEN = 0$



MSv45319V2

### 22.4.6 Выходное напряжение ЦАП

Цифровые входы преобразуются в выходные напряжения при линейном преобразовании между 0 и  $V_{REF+}$ .

Аналоговые выходные напряжения на каждом выводе канала ЦАП определяются следующим уравнением:

### 22.4.7 Выбор триггера ЦАП

Если бит управления  $TENx$  установлен, преобразование может быть запущено

$$DAC_{output} = V_{REF} \times \frac{DOR}{4096}$$

внешним событием (счетчик таймера, внешняя линия прерывания). Биты управления  $TSELx[3:0]$  определяют, какое из 16 возможных событий запускает преобразование, как показано в битах  $TSELx[3:0]$  регистра  $DAC\_CR$ . Эти события могут быть как программными, так и аппаратными. См. таблицу соединений в Разделе 22.4.2: Контакты ЦАП и внутренние сигналы.

Каждый раз, когда интерфейс DAC обнаруживает нарастающий фронт на выбранном источнике запуска (см. таблицу ниже), последние данные, сохраненные в регистре  $DAC\_DHRx$ , переносятся в регистр  $DAC\_DORx$ . Регистр  $DAC\_DORx$  обновляется через три цикла  $dac\_hclk$  после срабатывания триггера.

Если выбран программный триггер, преобразование начинается после установки бита SWTRIG. SWTRIG сбрасывается аппаратно после загрузки регистра DAC\_DORx с содержимым регистра DAC\_DHRx.

Выбор триггера сброса и выбор триггера приращения генерации пилообразной формы выполняются с помощью управляющих битов STRSTTRIGSELx и STINCTRIGSELx соответственно. Отображение STRSTTRIGSELx похоже на TSELx. Обратитесь к Разделу 22.4.2: Контакты DAC и внутренние сигналы для отображения TSELx, STRSTTRIGSELx и STINCTRIGSELx.

**Примечание.** Бит TSELx[3:0] нельзя изменить, если установлен бит ENx.

Когда выбран программный триггер, передача из регистра DAC\_DHRx в регистр DAC\_DORx занимает только один тактовый цикл `dac_hclk`.

### 22.4.8 Запросы прямого доступа к памяти

Каждый канал ЦАП имеет возможность прямого доступа к памяти. Два канала DMA используются для обслуживания запросов DMA канала DAC.

Когда внешний триггер (но не программный триггер) происходит, когда бит DMAENx установлен, значение регистра DAC\_DHRx передается в регистр DAC\_DORx, когда передача завершена, и генерируется запрос DMA.

В двойном режиме, если установлены оба бита DMAENx, генерируются два запроса DMA. Если требуется только один запрос DMA, необходимо установить только соответствующий бит DMAENx. Таким образом, приложение может управлять обоими каналами DAC в двойном режиме, используя один запрос DMA и уникальный канал DMA.

Поскольку передача данных из DAC\_DHRx в DAC\_DORx произошла до запроса DMA, самые первые данные должны быть записаны в DAC\_DHRx до того, как произойдет первое событие триггера.

#### Опустошение прямого доступа к памяти

- В режиме периферия-память FIFO может быть насыщен (overflow, переполнение), если шина памяти не была предоставлена для некоторых запросов периферии.
- В режиме память-периферия, может произойти недогрузка (underrun, опустошения) если шина памяти не была предоставлена перед возникновением запроса периферии. (Из другого документа))

Запрос DAC DMA не ставится в очередь, поэтому, если второй внешний триггер поступает до получения подтверждения для первого внешнего триггера (первый запрос), то новый запрос не выдается и устанавливается флаг недогрузки канала DMA DMAUDRx в регистре DAC\_SR, сообщение о состоянии ошибки. Канал DACx продолжает преобразовывать старые данные. Программное обеспечение должно очистить флаг DMAUDRx, записав 1, очистить бит DMAEN используемого потока DMA и повторно инициализировать каналы DMA и DAC x, чтобы правильно перезапустить передачу.

Программное обеспечение должно изменить частоту преобразования триггера ЦАП или уменьшить рабочую нагрузку прямого доступа к памяти, чтобы избежать нового опустошения прямого доступа к памяти. Наконец, преобразование DAC можно возобновить, включив как передачу данных DMA, так и триггер преобразования.

Для каждого канала DAC x также генерируется прерывание, если разрешен соответствующий бит DMAUDRIEx в регистре DAC\_CR.



## Двойной режим данных DMA

Когда контроллер DMA используется в нормальном режиме, запрос DMA передает только 12-битные (или 8-битные) данные. Поскольку ширина АНВ составляет 32 бита, два 12-битных данных могут передаваться одновременно. Чтобы использовать этот режим, установите бит DMADOUBLEx регистра DAC\_MCR.

Запрос DAC DMA генерируется через каждые два внешних триггера (кроме программных триггеров), когда установлен бит DMAENx:

1. При обнаружении первого триггера значения регистров DAC\_DHRx и DAC\_DHRVx передаются в регистры DAC\_DORx и DAC\_DORVx. Фактические данные ЦАП загружаются в регистр DAC\_DORx. Затем генерируется запрос DMA. DMA записывает новые данные в регистры данных DAC\_DHRx и DAC\_DHRVx.

2. При обнаружении следующего триггера фактические данные ЦАП загружаются в регистр DAC\_DHRVx. Этот второй триггер не генерирует никаких запросов DMA. Бит DORSTATx указывает, какие данные DOR фактически загружены на аналоговый вход ЦАП.

Функция недогрузки DMA также поддерживается в режиме данных DMA Double.

В режиме DMA Double запросы DMA могут обрабатывать только один канал DAC. Для использования двухканальных выходов в режиме DMA Double каждый канал DMA необходимо настроить отдельно.

Для перехода от режима двойных данных к режиму одиночных данных или наоборот должны быть выполнены следующие условия:

- ЦАП должен быть отключен.
- Бит DMAEN должен быть очищен ( $ENx = 0$  и  $DMAEN = 0$ ).

### 22.4.9 Генерация шума

Для создания псевдошумов переменной амплитуды доступен LFSR (регистр сдвига с линейной обратной связью). Генерация шума ЦАП выбирается установкой WAVEx[1:0] на 01. Предварительно загруженное значение в LFSR равно 0xAAA. Этот регистр обновляется через три такта `dac_hclk` после каждого события триггера, следуя определенному алгоритму расчета.

Значение LFSR, которое может быть частично или полностью замаскировано с помощью битов MAMPx[3:0] в регистре DAC\_CR, добавляется к содержимому DAC\_DHRx без переполнения, а затем это значение передается в регистр DAC\_DORx.

Если LFSR равен 0x0000, в него вводится «1» (антиблокировочный механизм). Генерацию волны LFSR можно сбросить, сбросив биты WAVEx[1:0].

***Примечание.** Триггер DAC должен быть включен для генерации шума путем установки бита TENx в регистре DAC\_CR.*

### 22.4.10 Генерация треугольных волн

Можно добавить треугольный сигнал малой амплитуды к постоянному или медленно меняющемуся сигналу. Генерация треугольной волны ЦАП выбирается установкой WAVEx[1:0] на 10". Амплитуда настраивается битами MAMPx[3:0] в ре-

Рисунок 160. Алгоритм расчета регистра DAC LFSR

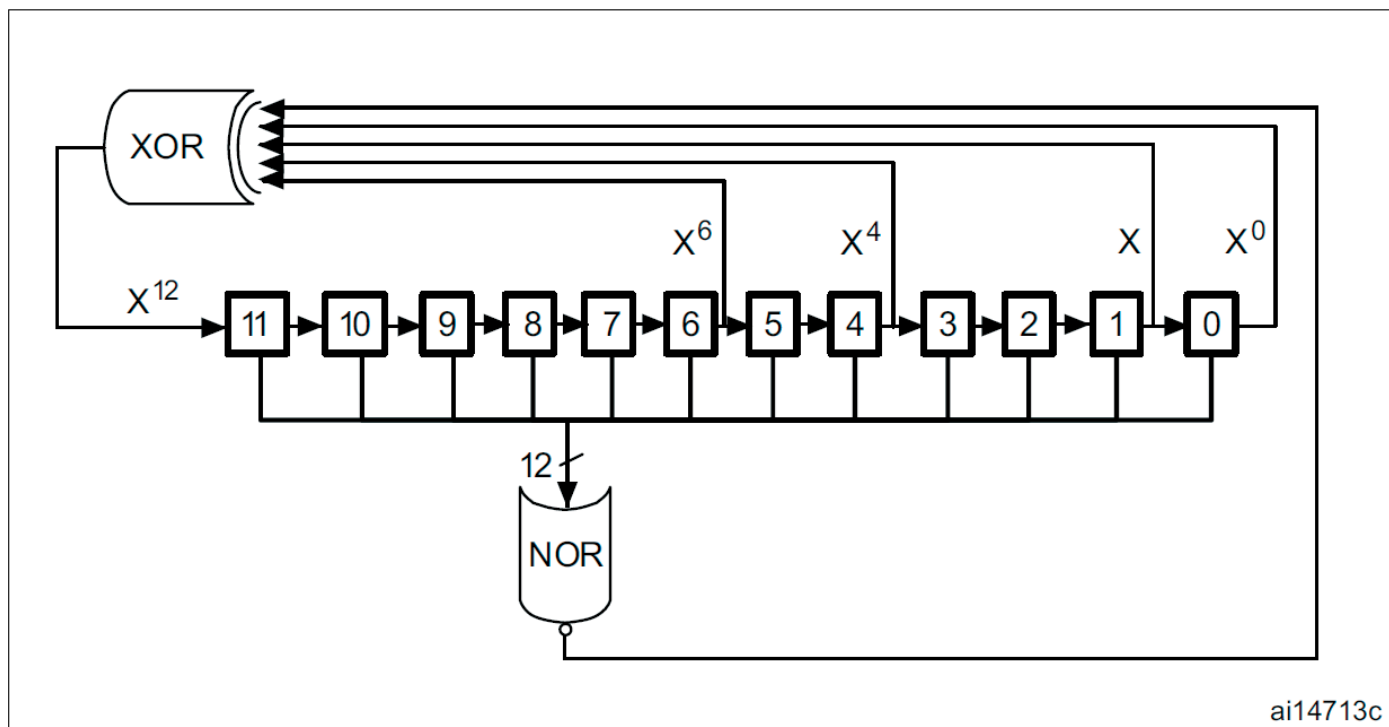
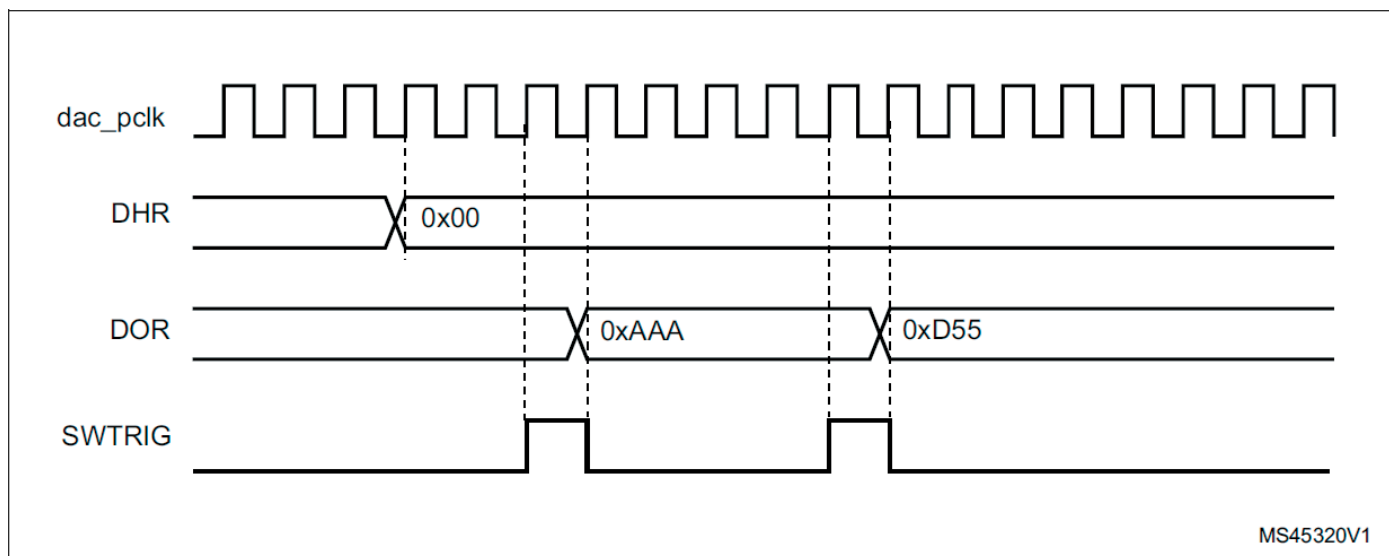


Рис. 161. Преобразование ЦАП (включен триггер SW) с генерацией волны LFSR



гистре `DAC_CR`. Внутренний счетчик треугольников увеличивается на три такта `dac_hclk` после каждого события триггера. Затем значение этого счетчика добавляется к регистру `DAC_DHRx` без переполнения, и сумма передается в регистр `DAC_DORx`. Счетчик треугольников увеличивается до тех пор, пока он меньше максимальной амплитуды, определяемой битами `MAMPx[3:0]`. Как только сконфигурированная амплитуда достигнута, счетчик уменьшается до 0, затем снова увеличивается и так далее.

Генерацию треугольной волны можно сбросить, сбросив биты `WAVEx[1:0]`.

**Примечание.** Для генерации треугольной волны необходимо включить триггер DAC, установив бит `TENx` в регистре `DAC_CR`.

Биты `MAMPx[3:0]` должны быть настроены перед включением ЦАП, иначе их нельзя будет изменить.

Рисунок 162. Генерация треугольной волны DAC

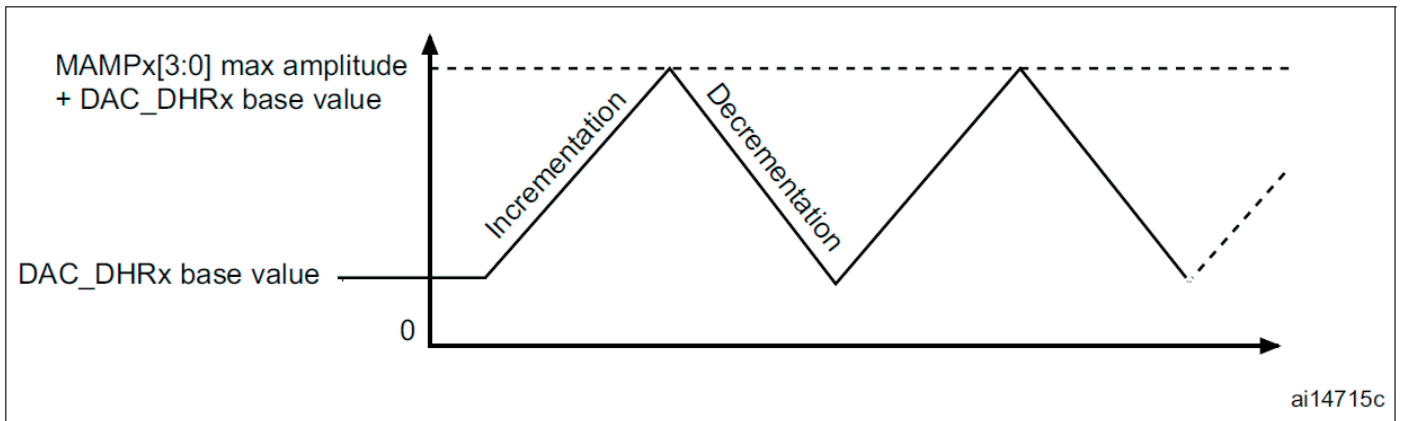
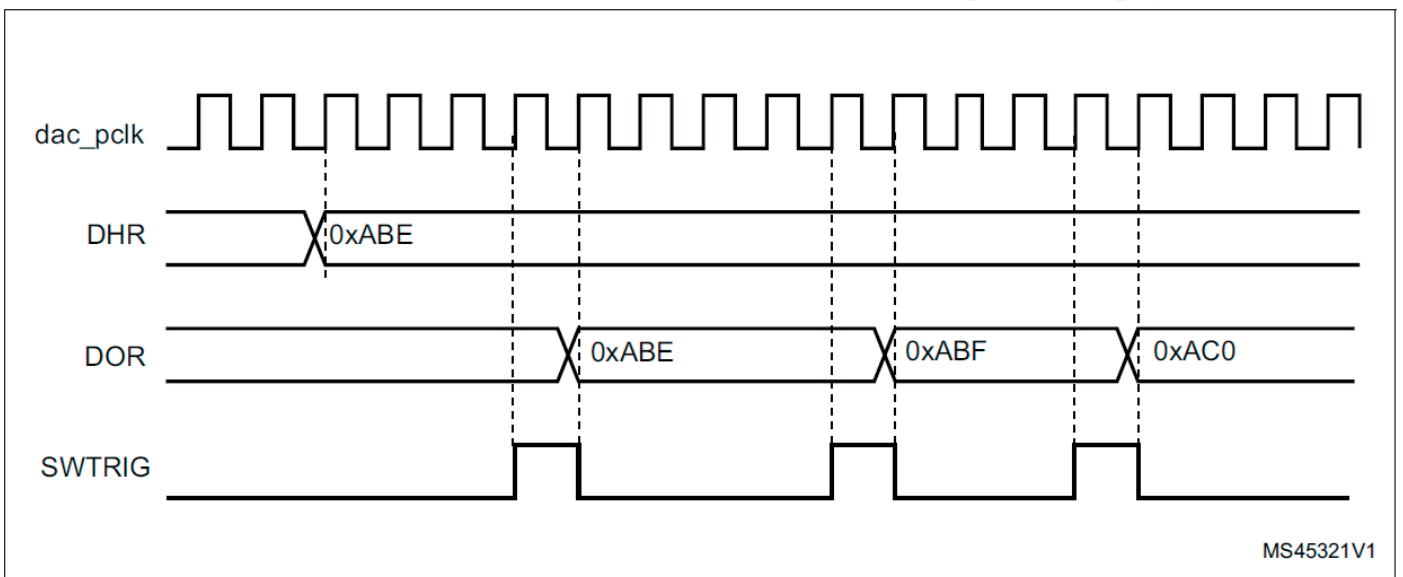


Рис. 163. ЦАП-преобразование (включен программный триггер) с генерацией треугольной волны



### 22.4.11 Генерация пилообразных сигналов ЦАП

ЦАП может генерировать сигнал пилообразной формы. Требуются специальные настройки регистра для начального значения, значения приращения и управления направлением:

- Генерация пилообразных сигналов ЦАП выбирается установкой `WAVEx[1:0]` в 11 в регистре `DAC_CR`.
- Начальное значение пилообразного счетчика (значение сброса) настраивается битами `STRSTDATAx[11:0]` в регистре `DAC_STRx`.
- Величина приращения определяется битами `STINCDATAx[15:0]` в регистре `DAC_STRx`.
- Пилообразное направление определяется битом `STDIRx` в регистре `DAC_STRx`.

Счетчик пилообразной формы начинается с `STRSTDATAx[11:0]` (биты с 12 по 15 установлены на 0000), каждый триггер увеличения затем увеличивает (или уменьшает) значение `STINCDATAx[15:0]`.

Выход ЦАП используется из 12 старших разрядов этого значения счетчика. Когда счетчик достигает 0x0000 или 0xFFFF, значение становится насыщенным. Триггерный сигнал пилообразного сброса инициализирует значение счетчика значением `STRSTDATAx[11:0]` (биты с 12 по 15 установлены на 0000).



Триггер приращения и триггер сброса должны быть выбраны с помощью битов  $STINCTRIGSELx[3:0]$  и  $STRSTTRIGSELx[3:0]$ .

Рис. 164. Генерация пилообразных сигналов ЦАП ( $STDIRx = 0$ )

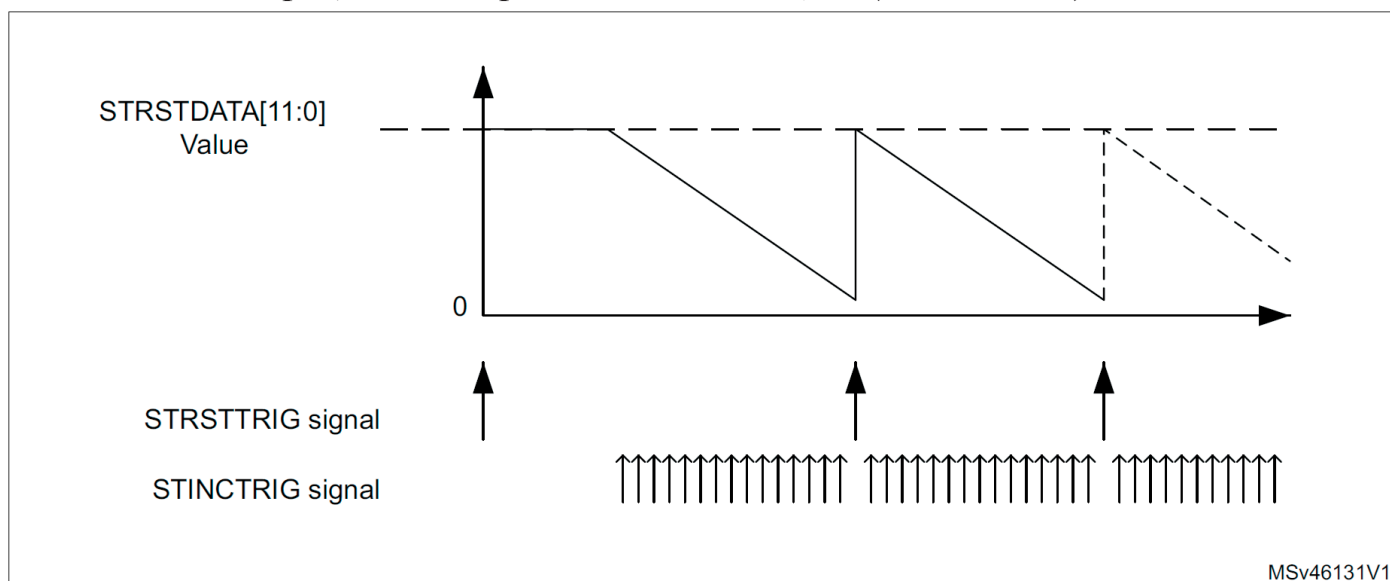
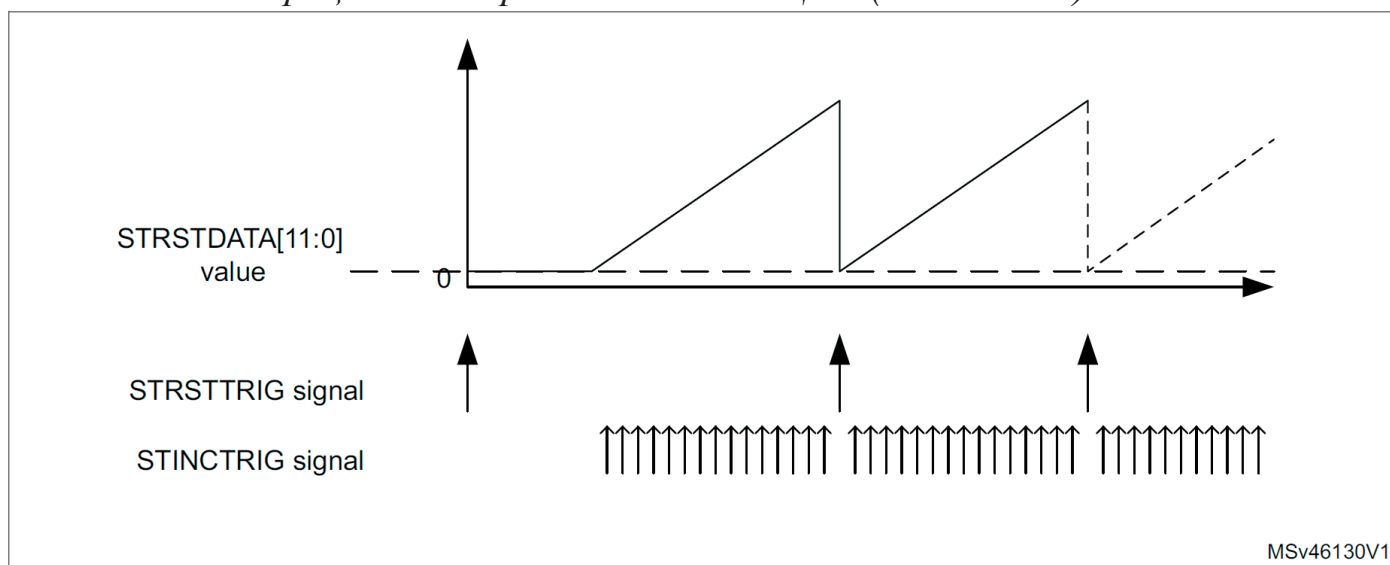
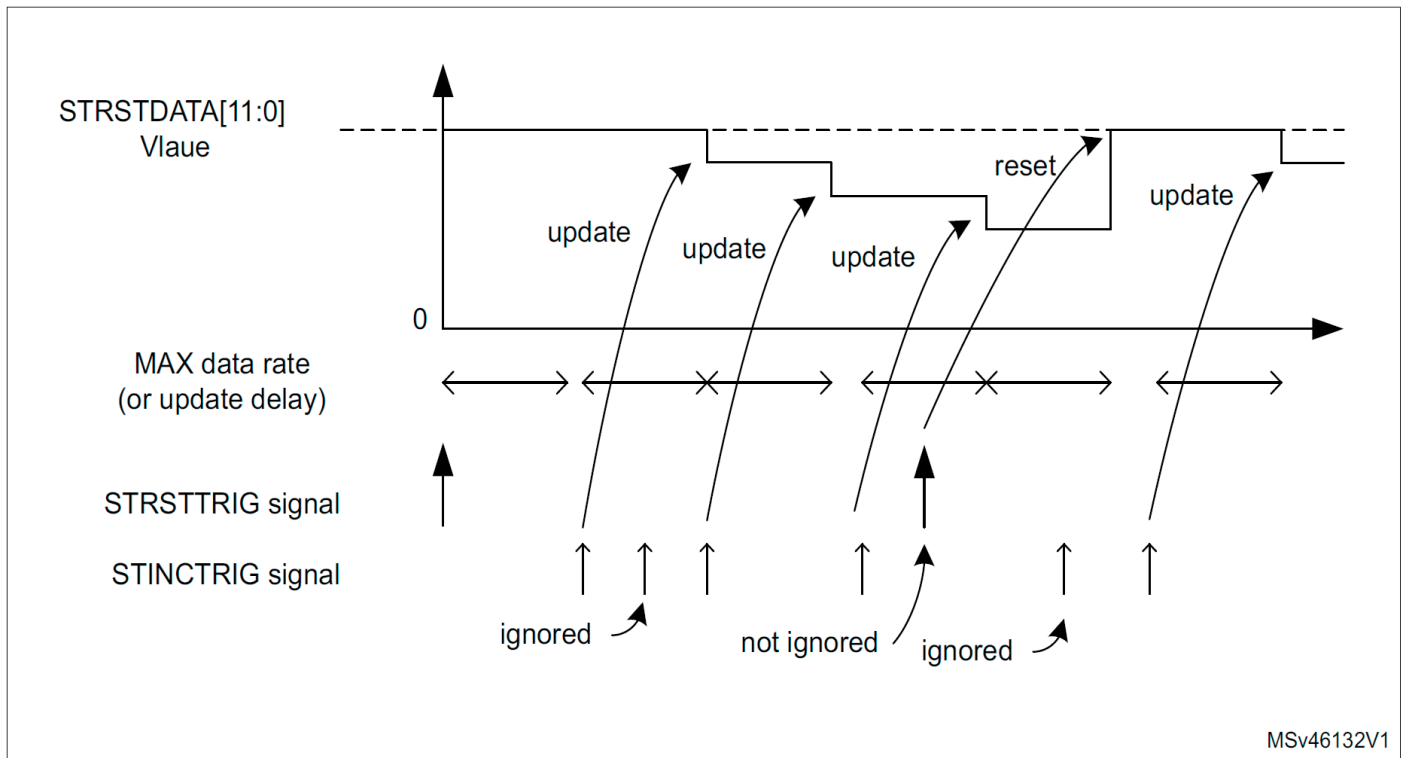


Рис. 165. Генерация пилообразных сигналов ЦАП ( $STDIRx = 1$ )



Сигнал  $STRSTTRIG$  имеет более высокий приоритет, чем  $STINCTRIG$ . Сигнал запуска не может быть быстрее скорости обновления  $DAC\_DORx$ , определенной в Таблице 188: описание  $HFSEL$ . Если  $STINCTRIG$  устанавливается быстрее допустимой скорости обновления данных, триггер  $STINCTRIG$  игнорируется. Если сигнал  $STRSTTRIG$  применяется после  $STINCTRIG$  и перед ограничениями скорости обновления  $DAC\_DORx$ , сигнал  $STRSTTRIG$  приостанавливается. Затем, сразу после приращения данных, применяется триггер сброса.

Рис. 166. Пилообразный ЦАП STINCTRIG и приоритет STRSTTRIG (STDIR = 0)



### 22.4.12 Режимы канала ЦАП

Каждый канал ЦАП можно настроить в обычном режиме или в режиме выборки и удержания. Выходной буфер может быть включен для получения высокой производительности привода. Перед включением выходного буфера необходимо откалибровать смещение напряжения. Эта калибровка выполняется на заводе (загружается после сброса) и может быть скорректирована программным обеспечением во время работы приложения.

#### Нормальный режим

В нормальном режиме есть четыре комбинации: изменение состояния буфера и изменение соединения контактов DACx\_OUTy. Чтобы включить выходной буфер, биты MODEx[2:0] в регистре DAC\_MCR должны быть:

- 000: ЦАП подключен к внешнему контакту
- 001: ЦАП подключен к внешнему контакту и к периферийным устройствам на кристалле. Чтобы отключить выходной буфер, биты MODEx[2:0] в регистре DAC\_MCR должны быть:
- 010: ЦАП подключен к внешнему контакту
- 011: ЦАП подключен к встроенной периферии