

**М Г Т У им. Н.Э.Баумана.
Кафедра ЭВМ и С.**

Старший преподаватель кафедры ИУ6 Аристов Б.К.

Методические указания
к лабораторной работе по курсу «Микроэлектроника и микропроцессорные системы»
для студентов специальности «Метрология и взаимозаменяемость»
«разработка и исследование измерительных систем с применением аналогоцифровых и
цифроаналоговых преобразователей».

Лабораторная работа №4

МОСКВА 2012

2012г.

Лабораторная работа №4.

.Разработка и исследование измерительных систем с применением аналогоцифровых и цифроаналоговых преобразователей.

Цель работы – разработка и исследование измерительных систем с применением аналогоцифровых и цифроаналоговых преобразователей, составление и отладка программы для обработки сигналов с преобразователей.. Продолжительность работы– 4 академических часа.

Теоретическая часть.

В состав периферийных устройств МК STM32L152RB входят:

1. один двенадцатибитный аналогоцифровой преобразователь (АЦП, ADC).
2. два двенадцатибитных цифроаналоговых преобразователя (ЦАП, DAC).

АЦП микроконтроллера STM32L152RB работает по принципу последовательного приближения. На рисунке 1 приведена блок-схема АЦП микроконтроллера серии STM32L15xx.

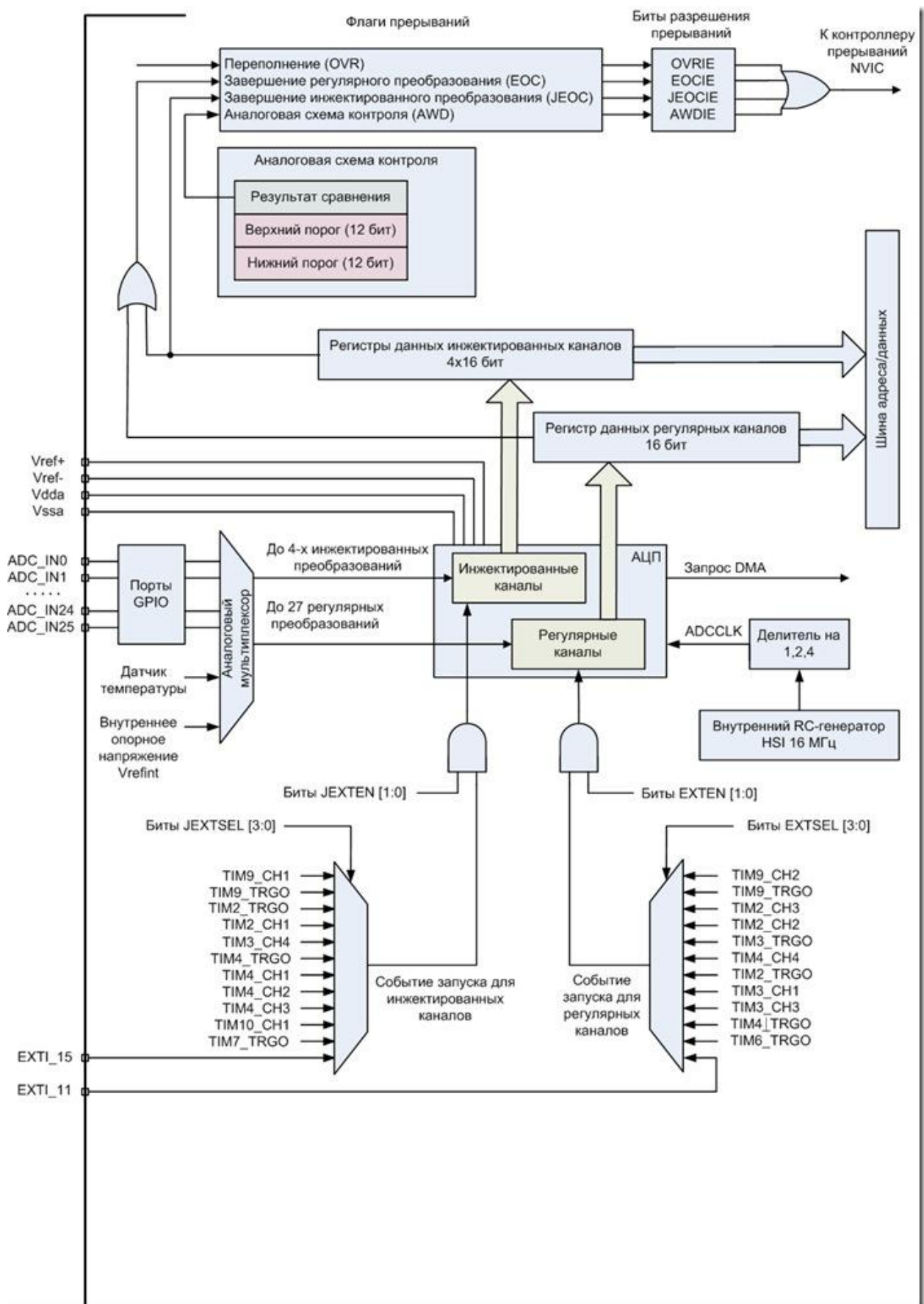


Рис. 1.

Режимы преобразования:

1. режим одиночного преобразования (Single channel),
2. одноканальный продолжительный (Single continuous),
3. многоканальный (scan).

4. многоканальный продолжительный (Scan continuous),
5. прерывистый (Discontinuous).

Каналы АЦП.

Каналы АЦП микроконтроллеров STM32 елятся на две группы:

1. регулярные каналы (regular),
2. инжектированные (injected).

Количество регулярных каналов для одного АЦП равняется 18, среди них 16 внешних и два внутренних - опорное напряжение и температурный датчик. Количество приоритетных каналов равняется четырем. Регулярные каналы подразумевают сохранение результатов преобразования через DMA-контроллер в памяти микроконтроллера, а инжектированные каналы имеют собственные регистры для хранения результата. Существует возможность настраивать работу каналов АЦП в произвольном порядке, несколько раз преобразовывать подряд одни и те же каналы, использовать внешние и программные события для старта преобразования.

В состав АЦП входят:

1. Аналого-цифровой преобразователь (АЦП) разделен на две части: для регулярных и инжектированных каналов. Тактирование АЦП может быть только от внутренней RC-генератор HSI. Частота внутреннего генератора равна 16 МГц. Эту частоту можно поделить на 2 или на 4. Следовательно, частота преобразования может быть установить равной 4, 8 или 16 МГц. Аналоговые величины поступают в преобразователь либо с внешних входов (для этого они должны быть сконфигурированы на работу в аналоговом режиме), либо от двух внутренних источников. Внутренние источники для АЦП – это температурный датчик и внутреннее опорное напряжение. С помощью температурного датчика можно измерить внутреннюю температуру чипа. Подключаются они к следующим каналам: термодатчик – входной канал 16, источник внутреннего опорного напряжения – входной канал 17. Из блок-схемы на Рис. 1. видно, что входных каналов может быть до 26. Два канала подключены к внутренним источникам (термодатчик и Vref – внутреннее опорное напряжение). Количество внешних каналов в зависимости от типа МК:

16 каналов у **STM32L15xCx** (в корпусе на 48 выводов)

20 каналов у **STM32L15xRx** (в корпусе на 64 вывода)

24 канала у **STM32L15xVx** (в корпусе на 100 выводов)

На отладочной плате STM32L-DISCOVERY установлен МК STM32L152RBT6Ю следовательно имеются 20 внешних выводов на корпусе МК, которые можно задействовать

под измерения. Это каналы от ADC_IN0 до ADC_IN21. Исключение составляют два канала ADC_IN16 и ADC_IN17 не имеют внешних выводов на корпусе МК.

22 внешних канала поступают далее на аналоговый мультиплексор для выбора только одного из них.

Результат преобразования сохраняется в специальных регистрах. Для регулярных каналов регистр данных имеет всего одну ячейку размерностью 16 бит. Для инжектированных каналов можно сохранить результаты сразу 4 преобразований подряд (размерность тоже 16 бит). Перед следующими преобразованиями результат нужно считать, иначе он будет перезаписан новым значением. Для регулярных каналов в этом случае можно сразу же после преобразования пересылать результаты в выделенную область памяти, для этого предусмотрен режим “прямого доступа к памяти” – DMA (Direct Memory Access).

Запуск преобразования может быть как программным, так и внешним. Внизу рисунка показаны еще два мультиплексора. С их помощью задают источник внешнего запуска преобразования для регулярных и инжектированных каналов отдельно. Внешний запуск может происходить от таймеров или с двух внешних входов.

2. Аналоговая схема контроля (Analog Watchdog). По сути это аналоговый компаратор с двумя пороговыми значениями – верхним и нижним, их величины задаются в двух 12-разрядных регистрах. Результат преобразования сравнивается с этими величинами. И если результат ниже нижнего или выше верхнего значений, устройство на это реагирует. Задействовать можно по-разному. Использовать эти пороговые значения как некие критические пределы, уходить за которые нельзя. Либо наоборот оставлять результаты измерения, вышедшие за предельное значение, остальные отбрасывать. Для этого схема контроля формирует прерывание.

Прерываний АЦП формирует всего 4.

OVR – переполнение. Это прерывание вызывается только при преобразованиях в регулярных каналах. Здесь точнее будет сказать “затирание” или же “наслоение” данных. Речь идет, естественно, о регистре данных для результата преобразования в регулярных каналах, который может хранить только одно значение. И если предыдущий результат еще не считан или не передан в память с использованием DMA, а уже прошло следующее преобразование и в регистр “лезут” новые данные, вот тогда произойдет потеря данных. О чем и будет просигнализировано выставлением флага прерывания OVR. То есть в данном случае вовсе не идет речь о превышении напряжения на входе, как можно ошибочно подумать из названия. Кстати, регистр результата преобразования автоматически очищается при чтении из него данных или передаче их в память через DMA.

EOC – завершение преобразования в регулярном канале.

JEOC – завершение преобразования в инжектированном канале

AWD – аналоговая схема контроля обнаружила выход результата измерения за пределы установленных порогов.

Основные возможности встроенного АЦП:

Разрядность АЦП можно изменять – это может быть 6, 8, 10, или 12 разрядов.

Для одного канала можно задать разные режимы: однократно измерить аналоговую величину или же запустить канал в режиме непрерывного измерения.

Режим сканирования. Можно создать группу каналов, задать порядок следования каналов в группе. Тогда измерения будут идти последовательно друг за другом, входной мультиплексор будет подключать внешние каналы к АЦП по очереди, в соответствии с запрограммированным порядком.

Выравнивание результата преобразования влево или вправо.

Время выборки можно запрограммировать индивидуально для каждого канала.

Функция внешнего запуска для регулярных и инжектированных каналов.

Режим “прерывистых” преобразований.

Преобразование всегда происходит на максимальной скорости, она зависит только от частоты тактирования АЦП ($ADCCLK = 4, 8, 16$ МГц). При 4 МГц оно равно 4 мкс, при 16 МГц – 1 мкс.

Функции автоматического включения/выключения АЦП при наличии/отсутствии преобразований. Опять же в целях снижения энергопотребления.

Требования к напряжению питания: 2.4В – 3.6В при работе на максимальной скорости, не менее 1.8 вольт на медленных скоростях.

Размах входного сигнала не должен выходить за пределы опорного напряжения V_{ref} .

Возможность введения временной задержки, автоматически вставляемой между преобразованиями. Длительность задержки программируется.

Генерация запроса для прямого доступа к памяти (режим DMA) во время преобразования в регулярном канале.

Порядок запуска АЦ преобразования.

Регистры АЦП.

Имя регистра	Адрес регистра	Назначение	
SR	0x40012400	Регистр статуса АЦП.	
CR1	0x40012404	Контрольный регистр 1 режима работы АЦП.	
CR2	0x40012408	Контрольный регистр 2 режима работы АЦП.	
SMPR1	0x4001240C	Регистр 1 времени преобразования.	
SMPR2	0x40012410	Регистр 2 времени преобразования.	
SMPR3	0x40012414	Регистр 3 времени преобразования.	
JOFR1	0x40012418	Смещение данных в инжектированной группе..	
JOFR2	0x4001241C	Смещение данных в инжектированной группе..	
JOFR3	0x40012420	Смещение данных в инжектированной группе.	
JOFR4	0x40012424	Смещение данных в инжектированной группе..	
HTR	0x40012428	Максимально допустимое преобразованное значение.	
LTR	0x4001242C	Минимально допустимое преобразованное значение.	
SQR1	0x40012430	Регистр 1 последовательности каналов регулярных преобразований.	
SQR2	0x40012434	Регистр 2 последовательности каналов	

		регулярных преобразований.	
SQR3	0x40012438	Регистр 3 последовательности каналов регулярных преобразований.	
SQR4	0x4001243C	Регистр 4 последовательности каналов регулярных преобразований.	
SQR5	0x40012440	Регистр 5 последовательности каналов регулярных преобразований.	
JSQR	0x40012444	Регистр последовательности инжектированных преобразований.	
JDR1	0x40012448	Регистр 1 последовательности каналов инжектированных преобразований	
JDR2	0x4001244C	Регистр 2 последовательности каналов инжектированных преобразований	
JDR3	0x40012450	Регистр 3 последовательности каналов инжектированных преобразований	
JDR4	0x40012454	Регистр 4 последовательности каналов инжектированных преобразований	
DR	0x40012458	Регистр результата для регулярного канала	

1. **SR** – содержит флаги, указывающие на состояние АЦП.
2. **CR1, CR2** – управляющие регистры, задают режим работы.
3. **SMPR1, SMPR2** – задают время преобразования.
4. **JOFR1...JOFR2** – определяют смещение данных в инжектированной группе.
5. **HTR, LTR** – задают верхнее и нижнее значение срабатывания оконного компаратора.
6. **SQR1...SQR3** – задают последовательность каналов регулярной группы.
7. **JSQR** – задает последовательность каналов инжектированной группы.
8. **JDR1...JDR4** – регистры данных инжектированной группы.
9. **DR** – регистр данных регулярной группы.

Необходимо задать:

1. режим преобразования АЦП,
2. список каналов, участвующих в преобразовании,
3. количество каналов, участвующих в преобразовании,
4. порядок опроса каналов,
5. алгоритм выравнивания результатов преобразования,
6. разрядность ФЦП.

Изначально, необходимо сконфигурировать порты. Определиться по каким каналам будут проводиться измерения, затем соответствующие выводы портов настроить для работы в аналоговом режиме.

Как и для всех других модулей, на АЦП нужно подать тактовый сигнал. То есть, необходимо включить генератор HSI установкой бита HSION в регистре RCC_CR. Затем дождаться готовности генератора к работе – аппаратной установки бита HSIRDY в том же регистре.

После этого можно подключать АЦП к источнику тактирования – устанавливаем бит ADC1EN в регистре RCC_APB2ENR (АЦП тактируется от матрицы шин APB2).

Теперь можно включать АЦП. Это делается установкой бита ADON в регистре ADC_CR2. В регистре статуса ADC_SR контролируем состояние бита ADONS, он аппаратно устанавливается в 1 при готовности АЦП к работе.

Перед непосредственным запуском преобразования необходимо сконфигурировать АЦП. Этот вопрос пока опустим, далее все рассмотрим подробней. Пока же будем считать, что АЦП настроили как нужно, и можно запускать преобразование.

Запускается оно установкой бита SWSTART в регистре ADC_CR2 для регулярных каналов, для инжектированных каналов это будет бит JSWSTART в том же регистре. Тут речь пока идет о программном запуске. Если модуль настроен на ожидание внешнего запуска, этот последний пункт можно исключить.

Режим Power-down. Принудительное выключение АЦП между преобразованиями.

В выключенном состоянии АЦП имеет практически нулевое потребление. Для снижения энергопотребления есть возможность принудительного отключения модуля, когда нет преобразований. Здесь возможны 2 режима, которые задаются установкой бит PDI или PDD в регистре ADC_CR1.

PDI – выключение АЦП во время “простоя” (idle). После каждого преобразования АЦП принудительно выключается до поступления новой команды на выполнение преобразования, программной (SWSTART, JSWSTART) или сигнала внешнего запуска.

PDD – выключение АЦП на время задержки между преобразованиями. Этот режим используется, когда задана временная задержка, автоматически вставляемая после каждого преобразования.

Необходимо помнить, что АЦП каждый раз после включения нужно какое-то время для “пробуждения” (стабилизации) – Tstab. В этом случае преобразование начнется с некоторой задержкой.

Выбор каналов.

Здесь все сделано довольно хитро. Изначально предполагается, что измерения будут проводиться сразу в группе каналов. А сколько каналов в группе: 1 или 27 – это уже второстепенно. Поэтому, если мы измеряем напряжение всего на одном выводе, мы должны понимать, что у нас не один канал для измерения. У нас группа каналов для измерений, но в группе всего один канал, и длина последовательности измерений равна единице. Вот такой подход. С другой стороны, все логично, если предусмотрена возможность проводить измерения друг за другом в разных каналах, и для этого формируется группа каналов, задается общая длина последовательности измерений, порядок следования каналов, то выделять как-то режимы для одиночных измерений нет необходимости.

Итак, существуют 5 регистров для регулярных каналов: ADC_SQR1 — ADC_SQR5. Они разбиты на секции по 5 бит.

ADC_SQR1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved								L[4:0]					Reserved			
								rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserv- ed	SQ27[4:0]					SQ26[4:0]					SQ25[4:0]					
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw	rw

ADC_SQR5

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		SQ6[4:0]					SQ5[4:0]					SQ4[4:1]			
		rw	rw	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
SQ4_0	SQ3[4:0]					SQ2[4:0]					SQ1[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Названия этих секций SQ1 – SQ27 (от слова **sequence – последовательность**). Каждая из этих секций – это номер преобразования в общей последовательности: SQ1 – первое преобразование в группе, SQ2 – второе и т.д. А уже в этих секциях пишем число, соответствующее номеру канала. Первое преобразование делаем в 1 канале – в SQ1 пишем 1. Второе преобразование в 14 канале – в SQ2 пишем 14 и т.д., вплоть до 27 преобразований подряд. А общую длину последовательности задаем в разрядах L регистра ADC_SQR1. Причем, длина не может быть нулевой, если там нули то это означает выполнение одного преобразования в группе, если единица – двух и т.д., вот так:

00000 – 1 преобразование

00001 – 2 преобразования

.....

11010 – 27 преобразований

Для инжестрированных каналов используется всего один регистр ADC_JCQR. Тут все аналогично, только максимальная длина последовательности преобразований не более 4-х, и задается двумя битами JL.

ADC_JSQR

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										JL[1:0]		JSQ4[4:1]			
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ4[0]	JSQ3[4:0]					JSQ2[4:0]					JSQ1[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Примечание: если регистры ADC_SQRx или ADC_JSQR модифицируются во время выполнения преобразований, то текущее преобразование прекращается, и АЦП будет ждать нового запуска. Если инжестрированное преобразование прервало выполнение регулярного, то регулярное преобразование в дальнейшем будет возобновлено.

Термодатчик и внутреннее опорное напряжение могут быть измерены через каналы ADC_IN16 и ADC_IN17 соответственно. Преобразование может быть в этом случае задано как регулярное так и инжестрированное.

Режимы одиночного или непрерывного преобразований.

Выбор режима осуществляется битом CONT в регистре ADC_CR2.

CONT = 0 – одиночное преобразование

CONT = 1 – непрерывное преобразование

При одиночном преобразовании (CONT = 0) номер канала выбирается в разрядах SQ1[4:0] регистра ADC_SQR5 (регулярный канал) или в разрядах JSQ1[4:0] регистра ADC_JSQR (инжектированный канал). Запускается программно (SWSTART для регулярных каналов, JSWSTART для инжектированных), или по внешнему запуску.

По завершению преобразования результат сохраняется в регистре ADC_DR (регулярный канал) или ADC_JDR1 (инжектированный канал). Выставляются флаги окончания преобразования: EOC (регулярный канал) или JEOC (инжектированный канал) в регистре статуса ADC_SR. Генерируются прерывания, если они заранее были разрешены установкой бит EOCIE или JEOCIE в регистре ADC_CR1 (опять же для регулярных или инжектированных каналов соответственно). Затем АЦП останавливается до новой команды запуска.

В режиме непрерывного преобразования (CONT = 1 в регистре ADC_CR2), для регулярных каналов каналов каждое новое преобразование начинается автоматически после предыдущего. Запускаем теми же способами, что и одиночное. По окончании каждого преобразования все происходит так же, результат сохраняется в ADC_DR, выставляется флаг EOC и генерируется прерывание, если оно разрешено. Остановить преобразования теперь можно только принудительно.

В инжектированных каналах режим непрерывного преобразования не предусмотрен. За исключением случая, когда инжектированные каналы сконфигурированы на автоматический запуск после регулярных каналов в непрерывном режиме. Об этом режиме будет далее.

Режим сканирования.

В этом режиме опрашивается группа каналов. Режим выбирается установкой бита SCAN в регистре ADC_CR1. АЦП опрашивает все каналы, выбранные в регистрах ADC_SQRx (регулярные каналы) или в регистре ADC_JSQR (инжектированные каналы). Для каждого канала группы выполняется одиночное преобразование. После окончания каждого преобразования следующий канал в группе опрашивается автоматически. Если к тому же установлен режим непрерывного преобразования (CONT = 1), то после последнего преобразования в группе, преобразования не прекращаются, а заново начинаются от первого выбранного канала в группе. Если установлен бит DMA, то данные из регистра ADC_DR пересылаются в память после каждого преобразования. Результат преобразования инжектированных каналов всегда сохраняется в регистрах ADC_JDRx. В регулярных каналах флаг окончания преобразования EOC может устанавливаться либо в конце каждого преобразования, либо только по окончании всей последовательности (а соответственно и генерация прерывания). Для этого в регистре ADC_CR2 предусмотрен управляющий бит EOCS (устанавливается программно)

EOCS = 0 – бит окончания преобразования EOC устанавливается после завершения всей последовательности регулярных преобразований.

EOCS = 1 – бит окончания преобразования EOC устанавливается после завершения каждого регулярного преобразования.

Здесь задаются времена выборки для каналов 0-9. В регистре ADC_SMPR2 – для каналов 10-19, в регистре ADC_SMPR1 – для каналов 20-25. Значения выбираются следующие:

000: 4 такта

001: 9 тактов

010: 16 тактов

011: 24 такта

100: 48 тактов

101: 96 тактов

110: 192 такта

111: 384 такта

Разрядность АЦП.

Возможны следующие варианты 6, 8, 10, 12 разрядов. Разрядность устанавливается битами RES[1:0] регистра ADC_CR1. При уменьшении разрядности также уменьшается и время преобразования. Варианты такие (при времени выборки 4 такта):

00 – разрядность 12 бит (время преобразования $12 + 4 = 16$ тактов)

01 – разрядность 10 бит (время преобразования $11 + 4 = 15$ тактов)

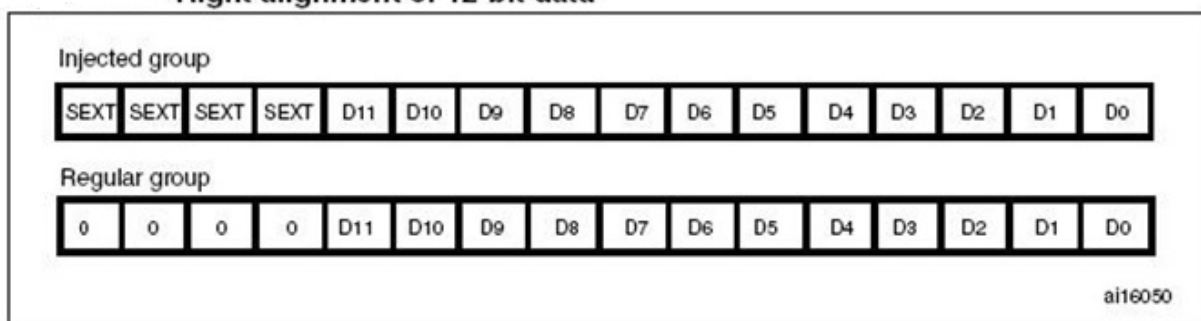
10 – разрядность 8 бит (время преобразования $9 + 4 = 13$ тактов)

11 – разрядность 6 бит (время преобразования $7 + 4 = 11$ тактов)

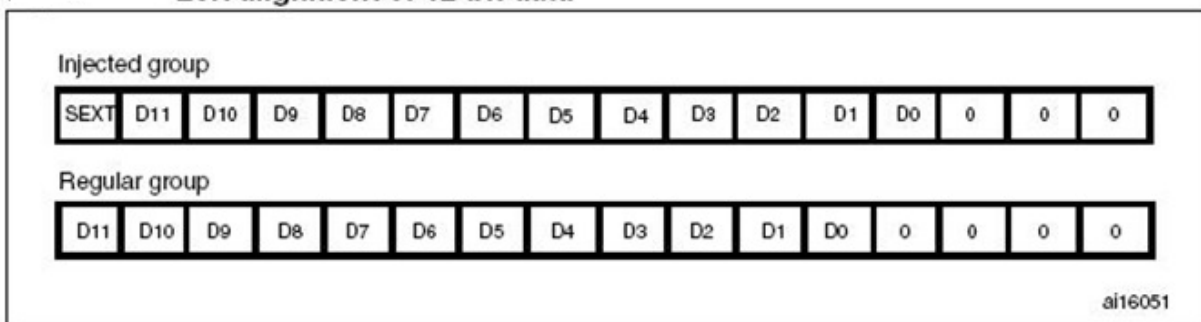
Выравнивание данных.

Задается битом ALIGN регистра ADC_CR2. Выравнивает результат преобразования в регистре данных по левому или правому краю. 0 – выравнивание вправо, 1 – влево.

Right alignment of 12-bit data



Left alignment of 12-bit data



Вот пример выравнивания 12-разрядного результата преобразования. Обратите внимание на биты SEXT. Это биты знака числа. Для инжектированных каналов можно задать еще и дополнительное смещение, вычитаемое из результата преобразования. Задается смещение в четырех регистрах ADC_JOFRx (поскольку у инжектированных каналов под результаты преобразования отведены 4 регистра, то и смещений можно задать 4). Поэтому числа в

регистрах данных для инжектированных каналов могут получиться отрицательными. Для регулярных каналов такое смещение не предусмотрено.

Внешний запуск.

Преобразование может быть запущено от внешнего события. Например, по захвату таймера или сигналом с внешнего вывода (EXTI). В этом случае, если биты EXTEN[1:0] (регулярные каналы) или JEXTEN[1:0] (инжектированные каналы) регистра ADC_CR2 отличны от 0b00, внешнее событие способно вызвать запуск преобразования. Значения бит EXTEN / JEXTEN задают такие события для запуска:

00 – внешний запуск не предусмотрен

01 – запуск по переднему срезу сигнала

10 – запуск по заднему срезу сигнала

11 – запуск в обоих случаях: и по переднему и по заднему срезам сигнала

Источники запуска выбираются в разрядах EXTSEL и JEXTSEL для регулярных и инжектированных каналов соответственно.

Режим DMA.

Поскольку для хранения результата преобразования в регулярных каналах отведен всего один регистр, есть вероятность потери данных при медленной обработке. В этом случае может помочь непосредственная пересылка данных сразу в заданную область памяти с использованием режима прямого доступа к памяти — DMA. Пока эта возможность не будет подробно описана, поскольку тут задействован модуль DMA, работа которого еще не рассматривалась. Сейчас же ограничимся знанием того, что такая возможность у АЦП есть.

Аналоговая схема контроля – AWD (Analog WatchDog).

Как уже упоминал вначале, этот модуль предназначен для контроля за нахождением измеряемого значения в заданном диапазоне. Включается установкой бит AWDEN (регулярные каналы), JAWDEN (инжектированные каналы) в регистре ADC_CR1. Там же разрешаются прерывание при выходе за пределы диапазона, установкой бита AWDIE. Бит AWDSGL отвечает за контроль в сканирующем режиме.

AWDSGL = 0 – контроль разрешен для всех каналов

AWDSGL = 1 – контроль разрешен для одного канала

Контролируемый канал для последнего варианта выбирается в битах AWDCH[4:0].

Предельные пороги задаются в двух регистрах ADC_HTR (верхний предел, 12-разрядное число) и ADC_LTR (нижний предел, 12-разрядное число). При выходе за эти пределы аппаратно устанавливается флаг AWD в регистре статуса ADC_SR.

Пример программы.

Ниже приведен фрагмент программы, реализующий одиночное преобразование.

```
#include "stm3211xx.h"

void adc_settings(void); //Объявляем функцию конфигурирования АЦП
void gpio_init(void); //Объявляем функцию инициализации портов
```

```

int main()
{
    gpio_init(); //Вызов функции инициализации портов
    adc_settings(); //Вызов функции конфигурирования АЦП
    ADC1->CR2 |= ADC_CR2_SWSTART; //Запуск преобразования
    while(1);
}

void gpio_init(void)
{
    RCC->AHBENR |= RCC_AHBENR_GPIOAEN; //Разрешаем тактирование порта А
    GPIOA->MODER |= GPIO_MODER_MODER5; //Устанавливаем 11 - аналоговый вход/выход для
    PA5
    GPIOA->PUPDR &= ~GPIO_PUPDR_PUPDR5; //Очищаем биты 00 - аналоговый вход/выход без
    подтяжки для PA5
}

void adc_settings(void)
{
    RCC->CR |= RCC_CR_HSION; //Включаем внутренний генератор HSI - 16МГц
    while(!(RCC->CR&RCC_CR_HSIRDY)); //Ждем его стабилизации
    RCC->APB2ENR |= RCC_APB2ENR_ADC1EN; //Разрешаем тактирование АЦП
    ADC1->CR2 |= ADC_CR2_ADON; //Включаем АЦП
    while(!(ADC1->SR&ADC_SR_ADONS)); //Ждем готовности АЦП
    ADC1->SQR5 |= (ADC_SQR5_SQ1_2 | ADC_SQR5_SQ1_0); //Устанавливаем биты 0 и 2. Таким
    образом, выбираем канал ADC_IN5 для выполнения 1-го преобразования.
    ADC1->CR1 &= ~ADC_CR1_RES; //Разрядность АЦП - 12 бит
    ADC1->CR2 &= ~ADC_CR2_ALIGN; //Выравнивание результата вправо
}

```

Результат АЦ преобразования можно посмотреть в регистре ADC_DR в режиме отладки после выполнения команды запуска преобразования ADC1->CR2 |= ADC_CR2_SWSTART. При этом, опорное напряжение равно 3 вольтам. При 3 вольтах на входе в регистре ADC_DR будет максимальное значение — 0x0FFF

Задание :

1. Составить программу для измерения напряжения с помощью АЦП.
2. Отладить программу используя отладочную плату STM32L-DISCOVERY.
3. Определить точность полученных результатов.
4. Составить отчет.
- 5 Защитить работу.

Литература.

1. **П.П. Редькин. 32/16-битные микроконтроллеры ARM7 семейства AT91SAM7 фирмы ATMEL. Руководство пользователя. Москва, издательский дом «Додека XXI». 2008.**
2. **Б. Пахомов. C/C++ и MS Visual C++ 2008 для начинающих. Санкт-Петербург, «БХВ-Петербург» 2009.**
3. **Joseph Yiu. The Defi nitive Guide to the ARM Cortex-M3.**