**Блок Init\_ADC инициализации контроллера АЦП**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

Блок реализует настройку АЦП микропроцессора. Программная реализация блока, расположена в заголовочном файле adc.h файла main.c проекта Test\_Driver (в среде разработки Keil), который создается при генерации кода проекта и содержит тексты программ алгоритма на языке СИ.

Основными характеристиками АЦП являются разрядность и время

преобразования. В микросхеме K1986ВЕ92QI разрядность АЦП составляет

12 бит. Это позволяет АЦП различать до 2^12 = 4096 различных уровней

напряжения, подаваемого на вход.

Время преобразования зависит от тактовой частоты, подаваемой на

АЦП. Минимальное время преобразования может составлять 1,95 мкс. Это

позволяет производить до 512820 преобразований в секунду. Скорость

преобразования можно регулировать, меняя тактовую частоту, подаваемую

на АЦП. В микроконтроллерах семейства 1986ВЕ9x реализовано сразу 2

независимых АЦП – ADC1 и ADC2, они входят в состав контроллера АЦП.

При работе с каждым из двух АЦП в микроконтроллерах семейства

1986ВЕ9х можно использовать следующие основные режимы:

1. режим одиночного преобразования по одному каналу с опросом бита окончания преобразования;
2. режим одиночного преобразования по одному каналу с

прерыванием по окончанию преобразования;

1. режим многократного преобразования по одному каналу с

использованием прямого доступа к памяти;

1. режим многократного преобразования c автоматическим

переключением нескольких каналов и использованием прямого

доступа к памяти.

Также можно выполнять синхронный запуск сразу двух АЦП,

комбинируя его с перечисленными выше режимами. Это позволяет

одновременно выполнять аналого-цифровые преобразования сразу для двух

разных каналов, что повышает скорость преобразования.

**Входные порты**

**inz\_adc** порт необходим для определения порядка выполнения блока

**Выходные порты**

**ready**  Возвращает 1, это необходимо для определения порядка выполнения последущих блоков ;

**Блок Init\_ADC(1|2) инициализация контроллера ADC(1|2)**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

Блок инициализирует работу АЦП(1|2). Реализация библиотеки находится в файле adc.h в функции Init\_ADC1(), при необходимости пользователь может внести дополнительные правки в код функции.

**Входные порты**

**inz\_adc(1|2)** порт необходим для определения порядка выполнения блока

**Выходные порты**

**ready**  Возвращает 1, это необходимо для определения порядка выполнения последующих блоков ;

**Свойства**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Имя | Описание | По умолчанию |
| **Каналы** | **ADC\_CHANALS** | Каналы АЦП  они формируются по схеме ИЛИ. | ADC\_CH\_ADC7 |
| **Режим переключения каналов** | **ADC\_CH\_SWITCHING\_Disable** | Смысл состоит в том, что можно  заставить микроконтроллер автоматически переключать вход АЦП на другой  канал после выполнения очередного преобразования. Это дает возможность  задействовать режим многократного преобразования c автоматическим  переключением нескольких каналов. | ДА , если установить НЕТ, то задействуется режим с автоматическим переключением каналов |
| **Режим циклического преобразования** | **MODE\_SINGLE\_CONV** | ADC\_SAMPLING\_MODE\_SINGLE\_CONV – одиночное;   ADC\_SAMPLING\_MODE\_CICLIC\_CONV – последовательное.  При одиночном преобразовании после запуска АЦП производится  одно преобразование. Для выполнения следующего преобразования нужно  вновь запустить АЦП.  При последовательном преобразовании после завершения одного  преобразования автоматически запускается новое преобразование. Такой  режим обычно используют совместно с прямым доступом к памяти. | Да, если установить НЕТ, то включается режим последовательного преобразования |

**Примечание!**

**Если автоматическое переключение каналов отключено, то нужно**

**указать конкретный канал, с которым АЦП будет работать.**

**Можно выбирать следующие каналы:**

**ADC\_CH\_ADC0 – канал 0, порт PD0 ;**

**ADC\_CH\_ADC1 – канал 1, порт PD1 ;**

**ADC\_CH\_ADC2 – канал 2, порт PD2 ;**

**ADC\_CH\_ADC3 – канал 3, порт PD3 ;**

**ADC\_CH\_ADC4 – канал 4, порт PD4 ;**

**ADC\_CH\_ADC5 – канал 5, порт PD5 ;**

**ADC\_CH\_ADC6 – канал 6, порт PD6 ;**

**ADC\_CH\_ADC7 – канал 7, порт PD7 ;**

**ADC\_CH\_ADC8 ...ADC\_CH\_ADC15 – каналы 8-15 недоступны**

**в микроконтроллере K986ВЕ92QI;**

**Блок ADC(1|2)\_GetResult0 считываем значения кода АЦП и возвращаем номер опрошенного канала**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да/) |

Блок последовательно возвращает значения кода АЦП, каналов, определенных при настройке АЦП и номер опрошенного канала. При обращении блок возвращает значение кода АЦП одного канала (готового канала), при последующем обращении номер канала может быть другим. Блок удобно использовать в случае если выбран один канал АЦП.

**Входные порты**

Is\_init порт сортировки блока определяет порядок выполнения блока

**Выходные порты**

**Du** код опрошенного канала АЦП

**Status** номер канала

**Свойства**

**Нет**

**Пример adc1.prt**

**Блок ADC(1|2)\_GetResult производит последовательный опрос всех заданных каналов АЦП**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

Блок при обращении последовательно опрашивает все заданные каналы АЦП.

**Входные порты**

**Входные порты**

**Is\_init** порт сортировки блока определяет порядок выполнения блока

**Выходные порты**

**Adc\_mean** массив uint32\_t целочисленных 16 битовых значений (содержит 12 битовый код АЦП и номер канала), размерность массива равна количеству опрашиваемых каналов

**Свойства**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Имя | Описание | По умолчанию |
| **Количество каналов** | **CH\_NUM** | Какое количество каналов АЦП мы хотим опросить при обращении к блоку | 3 |

**Пример adc2.prt**

**Блок** **Switch\_chanal0 распределение 12 битовых кодов АЦП по каналам**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

На вход блока поступает массив uint32\_t c 16 битовых чисел, значений АЦП, блок определяет 12 битовый код АЦП и номер канала АЦП

**Входные порты**

**adc** входной массив 16 битовых значений полученных после опроса определенных каналов АЦП

**Выходные порты**

**Adc\_ch(1..7)** номера портов соответствуют номерам каналов АЦП и на выходе 12 код АЦП, хранящийся в переменной размер uint32\_t

**Adc\_vref** канал датчика опорного напряжения VREF, только для ADC1

**Adc\_temp** канал температурного датчика, только для ADC1

**Свойства**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Имя | Описание | По умолчанию |
| **Количество каналов** | **SIZE** | Какое количество каналов АЦП мы хотим опросить при обращении к блоку | 3 |

**Пример adc2.prt**

**Блок** **Switch\_chanal** **определения каналов АЦП**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

На вход блока поступает массив 32 битовых чисел, считанных контроллером DMA, значений АЦП, блок определяет 12 битовый код АЦП и номер канала АЦП и заполняет массив соответствующего канала АЦП.

**Входные порты**

**adc** входной массив 32 битовых значений полученных после опроса определенных каналов АЦП контроллером DMA

**Выходные порты**

**Adc\_ch(1..7)** номера портов соответствуют номерам каналов АЦП и на выходе массив uint32\_t в нем расположены 12 битные коды АЦП

**Adc\_vref** порт канала датчика опорного напряжения VREF, только для ADC1, содержит массив значений датчика опорного напряжения

**Adc\_temp** порт канал температурного датчика, только для ADC1, содержит массив значений датчика температуры

Суммарное количество элементов в выходных массивах портов равно **SIZE**.

**Свойства**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Имя | Описание | По умолчанию |
| **Количество каналов** | **SIZE** | Размер массива | 32 |

**Пример dma\_adc.prt, dma\_adc\_dac.prt, adc\_dac.prt**

**Блок** **Get\_MeanStatus** **разбор массива АЦП**

|  |  |  |
| --- | --- | --- |
| Картинка блока в палитре | Картинка блока на схеме | Векторизован ли блок, генерится ли блок в СИ код (да) |

На вход блока поступает массив 32 битовых чисел, значений АЦП, блок преобразует массив в два массива:

- 12 битовый код АЦП;

- номер канала АЦП;

Этот блок удобно использовать после выхода блока DMA.

**Входные порты**

**adc** входной массив 32 битовых значений полученных после опроса определенных каналов АЦП

**Выходные порты**

**Adc\_mean** массив uint32\_t c 12 битовыми кодами АЦП

**Adc\_status** массив uint32\_t c номеров каналов АЦП

**Свойства**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Имя | Описание | По умолчанию |
| **Количество каналов** | **SIZE** | Размер массива 16 битовых чисел, значений АЦП | 32 |

**Пример dma\_adc.prt, dma\_adc\_dac.prt, adc\_dac.prt**