

Рекомендации по работе с АЦП в микроконтроллерах PIC16C7X

Статья основывается на технической документации Fact2
компании Microchip Technology Incorporated, USA.

**© ООО «Микро-Чип»
Москва - 2001**

Распространяется бесплатно.
Полное или частичное воспроизведение материала допускается только с письменного разрешения
ООО «Микро-Чип»
тел. (095) 737-7545
www.microchip.ru

Рекомендации по работе с АЦП в микроконтроллерах PIC16C7X

Статья основывается на технической документации Fact2 компании Microchip Technology Incorporated, USA.

АЦП является одним из важнейших инструментов для преобразования аналогового сигнала в цифровой код. Как только аналоговый сигнал представляется в цифровой форме, он может быть проанализирован, сохранен, изменен и т.д. В микроконтроллерах PICmicro фирмы Microchip интегрирован модуль 8,10 и 12-битного АЦП. Для работы с АЦП требуется минимум знаний, однако, для оптимальной работы модуля АЦП потребуется более высокий уровень понимания проблемы. Эта статья направлена на рассмотрение основных технических аспектов работы модуля АЦП в микроконтроллерах PICmicro.

При рассмотрении параметров работы модуля АЦП используются примеры с граничными рекомендованными значениями исходных данных. В реальном проекте, при оптимизации режима работы АЦП, можно получить лучшие характеристики, чем в примерах данной статьи.

Сначала рассмотрим основные параметры и условия работы модуля АЦП, затем коснемся более сложных параметров:

- скорость преобразования
- точность преобразования

Основные параметры

Рассмотрим модуль АЦП в микроконтроллерах PIC16C7X.

АЦП фирмы Microchip - являются АЦП последовательного приближения, разрядностью 8 бит, с внутренним конденсатором $Chold = 51.2$ пФ. Входные аналоговые каналы подключаются к АЦП через коммутатор каналов, поэтому преобразование сигнала может производиться только с одного из выбранных входов микроконтроллера. Точность преобразования +/- 1 бит, но она может варьироваться в зависимости от требований к АЦП.

Необходимо помнить, что прежде чем начать преобразование внутренний удерживающий конденсатор $Chold$ должен успеть зарядиться/разрядиться до напряжения на аналоговом входе. Выходное сопротивление источника сигнала и сопротивление цепей АЦП вносят постоянную составляющую (RC цепочка), которая должна учитываться при выполнении преобразования с заданной точностью. Как только будет начато преобразование, входной сигнал отключается от конденсатора $Chold$. Напряжение на конденсаторе $Chold$ поддерживается постоянным в течении всего времени преобразования.

Рекомендованная последовательность действий для работы с АЦП:

1. Включить модуль АЦП, установив бит $ADON$ регистра $ADCON0$.

2. При включении АЦП, по умолчанию, все возможные линии АЦП работают как аналоговые входы. При этом аналоговые входы отключены от входных цифровых КМОП буферов. Если напряжение на входе цифрового КМОП буфера будет составлять например 0,5В, то ток потребления данного буфера составит примерно 100 – 150 мкА. Для настройки линии АЦП как цифровой вход, необходимо изменить биты $PCFG$ регистра $ADCON1$. При чтении содержимого регистра порта нули будут установлены в тех разрядах, которые были настроены как аналоговые входы.

3. Выберете входной канал, установив/сбросив биты $CHS2$, $CHS1$, $CHS0$ в регистре $ADCON0$. Для получения детальной информации по регистрам $ADCON0$, $ADCON1$ обратитесь к технической документации по соответствующему микроконтроллеру.

4. С помощью битов $ADCS1$, $ADCS0$ регистра $ADCON0$ выберете источник тактового сигнала для модуля АЦП. Всего доступно 4 варианта

- $Fosc/2$
- $Fosc/8$
- $Fosc/32$
- Frc (внутренний RC генератор)

В таблице представлено рекомендованный тип тактового сигнала для модуля АЦП в зависимости от тактовой частоты микроконтроллера.

| Микроконтроллер | $Fosc/2$ | $Fosc/8$ | $Fosc/32$ | Frc |
|-----------------------|-----------------|--------------|---------------|-------|
| PIC16C71 | ≤ 1 МГц | ≤ 4 МГц | ≤ 16 МГц | любое |
| PIC16C70/71A/72/73/74 | ≤ 1.25 МГц | ≤ 5 МГц | ≤ 20 МГц | любое |

Примечание:

При выборе внутреннего RC генератора типовое значение $Tda = 4$ мкс ($minTda = 2$ мкс, $maxTda = 6$ мкс).

Для вычисления значения T_{da} необходимо воспользоваться формулой

$$T_{da} = X/F_{osc}, \text{ где } X = 2; 8 \text{ или } 32.$$

5. Выждать время необходимое для заряда/разряда конденсатора $Chold$ (51.2 пФ), состоящее из:

- T_{amp} – время установки выходного напряжения на усилителе
- T_{charge} – время зарядки конденсатора $Chold$
- T_{temp} – температурный коэффициент

$$T_{trac} = T_{amp} + T_{charge} + T_{temp}$$

$$T_{amp} = 5 \text{ мкс.}$$

$$T_{temp} = (Temp - 25 \text{ гр.С})(0.05 \text{ мкс/гр.С}), \text{ при температуре } Temp > 25 \text{ гр.С}$$

$$T_{temp} = 0, \text{ при температуре } Temp \leq 25 \text{ гр.С}$$

Получим уравнение для T_{charge}

Основное уравнение заряда конденсатора:

$$V_{cap} = V_{final} - (V_{final} - V_{initial}) e^{-(T/RC)}$$

В нашем случае это уравнение принимает вид:

$$V_{holdcap} = V_{ref} - (V_{ref} - 0) e^{-(T_{charge}/R_{chold})}$$

Сопротивление R состоит из:

R_s – сопротивление источника сигнала

R_{ic} – внутренне сопротивление канала АЦП

R_{ss} – сопротивление защелки конденсатора $Chold$

$$R = R_s + R_{ic} + R_{ss}$$

Для микроконтроллеров PIC16C7X:

$$Chold = 51.2 \text{ пФ}; V_{holdcap} = V_{ref} - V_{ref}/error$$

Поэтому

$$(V_{ref} - V_{ref}/error) = V_{ref} (1 - e^{-(T_{charge}/(R_{ic} + R_{ss} + R_s)Chold)})$$

Выполняя преобразования, получаем:

$$T_{charge} = - (R_{ic} + R_{ss} + R_s)Chold \ln(1/error)$$

Выдержка времени, меньшая, чем полученное значение T_{charge} , может значительно снизить точность преобразования.

Пример 1.

$$V_{dd} = V_{ref} = 5 \text{ В}; R_{ic} = 1 \text{ кОм}; R_{ss} = 7 \text{ кОм}; R_s = 10 \text{ кОм}$$

$$Error = 8 \text{ бит} + - 1/2 Lsb = 2^9 = 512$$

$$T_{charge} = - (1 \text{ кОм} + 7 \text{ кОм} + 10 \text{ кОм})(51.2 \text{ пФ}) \ln(1/512) = 5.75 \text{ мкс}$$

Пример 2.

При тех же условиях, что и в примере 1, добавим разделительный конденсатор емкостью 0.1 мкФ на входе АЦП. При этом сопротивление источника сигнала становится равным $R_s = 50 \text{ Ом}$.

$$T_{charge} = (1 \text{ кОм} + 7 \text{ кОм} + 50 \text{ Ом})(51.2 \text{ пФ}) \ln(1/512) = 2.57 \text{ мкс}$$

6. Запустить процесс преобразования установкой бита GO/DONE регистра ADCON0.

7. Ожидать, окончания преобразования. Пока бит GO/DONE не будет сброшен, или установлен флаг прерывания от АЦП ADIF. Результат преобразования сохранен в регистре ADRES.

8. Конденсатор $Chold$ будет автоматически подключен к выбранному аналоговому каналу через $2T_{da}$ после завершения преобразования.

9. Полное время преобразования, с помощью которого можно определить максимальную частоту выборки состоит из:

- времени заряда конденсатора $Chold$
- времени преобразования
- времени подключения конденсатора $Chold$ аналоговому входу после окончания преобразования.

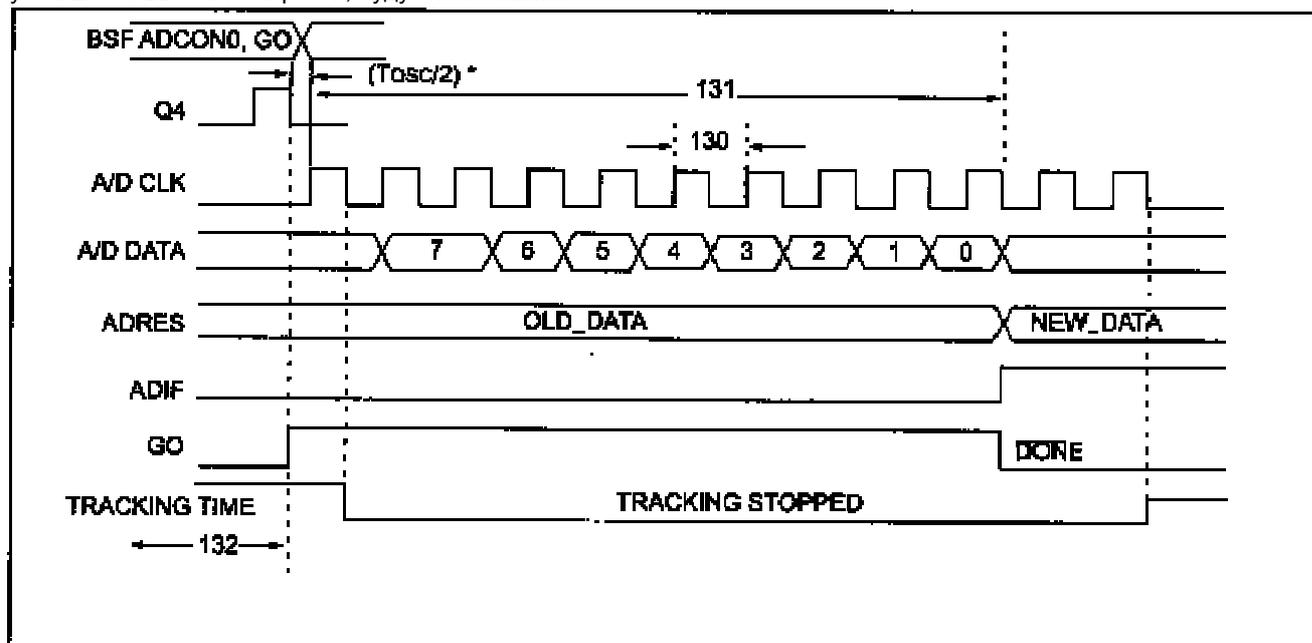
Скорость преобразования

Если необходимо использовать модуль АЦП с периодом выборок меньшим чем 60 мкс, то возможны следующие варианты сокращения полного времени преобразования.

1. Выбором источника тактового сигнала АЦП, необходимо получить значение T_{da} близкое (но не менее) к 2 мкс для PIC16C71 и 1.6 мкс для PIC16C70, PIC16C71A, PIC16C72, PIC16C73 и PIC16C74.

2. Необходимо определиться, какая точность преобразования необходима 4, 6 или 8 бит. Это важно, т.к. можно ускорять процесс преобразования по достижению нужной точности. Нельзя прерывать процесс преобразования, т.к. при этом промежуточный результат не сохраняется в регистре ADRES.

Запись в регистр ADRES производится в не зависимости от того, с какой точностью был получен каждый из 8 битов результата. На рисунке показана работа модуля АЦП по тактам. Модуль АЦП работает синхронно относительно внутреннего цикла микроконтроллера (если не выбран внутренний RC генератор) и можно точно определить, когда нужно перейти на более высокую скорость, например с $F_{osc}/8$ на $F_{osc}/2$. Биты результат, полученные на высокой скорости, будут не точными.



Определить время, после которого допускается перейти на более высокую скорость, можно по формуле

$$T_{con} = 1.5 T_{da} + N T_{da} + (8 - N)(2 T_{osc})$$

(где T – требуемое разрешение АЦП)

Пример 3. 8-битный результат, режим быстрого преобразования

Микроконтроллер PIC16C74

Тактовый сигнал 20МГц

Минимально допустимое время $T_{da} = 1.6$ мкс

$$T_{da} = 32/F_{osc}; T_{da} = 1.6 \text{ мкс}$$

$$T_{trac} = (5 \text{ мкс} + 0 \text{ мкс} + 2.57 \text{ мкс}) = 7.57 \text{ мкс (взято из примера 2)}$$

$$T_{con} = 1.5 * 1.6 \text{ мкс} + 8 * 1.6 \text{ мкс} + (8 - 8)(2 * 50 \text{ нс}) = 15.2 \text{ мкс}$$

Полное время преобразования

$$T = 15.2 \text{ мкс} + 7.57 \text{ мкс} + 2 * 1.6 \text{ мкс} = 25.97 \text{ мкс}$$

Максимальная частота выборок

$$F = 38.5 \text{ КГц}$$

Пример 4. 4-битный результат, режим быстрого преобразования

Микроконтроллер PIC16C71

Тактовый сигнал 16МГц

Минимально допустимое время $T_{da} = 2$ мкс

$$T_{da} = 32/F_{osc}; T_{da} = 2 \text{ мкс}$$

$$T_{trac} = (5 \text{ мкс} + 0 \text{ мкс} + 2.57 \text{ мкс}) = 7.57 \text{ мкс (взято из примера 2)}$$

$$T_{con} = 1.5 * 2 \text{ мкс} + 8 * 2 \text{ мкс} + (8 - 8)(2 * 62.5 \text{ нс}) = 11.5 \text{ мкс}$$

Полное время преобразования

$$T = 11.5 \text{ мкс} + 7.57 \text{ мкс} + 2 * 2 \text{ мкс} = 23.07 \text{ мкс}$$

Максимальная частота выборок

$$F = 43.3 \text{ КГц}$$

Значение регистра ADRES изменяется только после завершения преобразования, когда сбрасывается бит GO/DONE и устанавливается флаг прерывания от АЦП ADIF.

Для максимального увеличения частоты выборок, допускается производить переключение входного канала сразу после старта преобразования (установки бита GO/DONE). Такое действие не влияет на текущее преобразование, т.к. конденсатор $Chold$ отключен от аналогового входа.

Точность преобразования

Факторы, влияющие на точность преобразования АЦП. Максимально возможная точность измерений +/- 1Lsb.

1. Пульсации на входе питания Vdd сильно снижают точность преобразования. Устанавливаете блокировочные конденсаторы на шине питания микроконтроллера. Помните, что блокировочные конденсаторы не значительно снижают низкочастотные пульсации.

2. При старте преобразования потребление с источника опорного напряжения возрастает до 1 мА за 8 – 20 нс. Если вы используете внешний источник опорного напряжения, убедитесь, что он обеспечивает необходимую скорость нарастания тока потребления.

3. Использовать Vdd в качестве опорного напряжения лучше, чем с вывода RA3 при значительных пульсациях на шине питания.

Пример 5.

Шум 20 мВ при Vdd = 5.0 В = Vref

Шум 20 мВ при RA3 = 3.0 В = Vref

4. Шум на аналоговом входе, тоже вносит погрешность измерений. При этом рекомендуется использовать конденсаторные фильтры.

Примечание: В микроконтроллерах PIC16C71 и PIC16C70 вывод RA0(AN0) расположен в непосредственной близости к тактовому входу OSC1, что вносит некоторую погрешность при преобразовании с этого входа. Не рекомендуется использовать RA0 как аналоговый вход.

5. Увеличивая время Ttrac > 5 мкс, тем самым увеличиваете точность преобразования. Рекомендованное значение Ttrac = 20 мкс.

6. При времени Tda=4 мкс, внутренний компаратор сравнения АЦП работает с наилучшим качеством. Замете, что при выборе внутреннего RC генератора для тактирования АЦП типовое значение Tda = 4 мкс.

7. Оптимальная температура работы модуля АЦП 25 гр.С. Высокая температура увеличивает токи утечки, низка смещает переходные уровни внутренней логики. Если проектируемо устройство может быть подвержено воздействию высокой температуры, выбирайте значение Tda минимально возможные.

8. Не изменяйте логические уровни на портах вывода во время преобразования, т.к. это может вызвать резкое изменение тока потребления, что приведет к дополнительному шуму на выводе Vdd и снижению точности измерения.

9. Модуль АЦП может работать в SLEPP режиме микроконтроллера. При этом в качестве тактовых импульсов для АЦП должен быть выбран внутренний RC генератор.

Примечание: При использовании режима SLEEP не рекомендуется использовать RC тактовый генератор микроконтроллера. Т.к. при выходе из режима SLEEP требуется большее время на «пробуждение» по сравнению с режимами LP, XT и HS.

Если разрешить прерывание от АЦП (установкой бита ADIE (для PIC16C71)), но запретить прерывания глобально (сброс бита GIE), то после запуска преобразования и перехода в режим SLEEP микроконтроллер «пробудиться» по окончании преобразования, но не произойдет переход по вектору прерывания, а будет выполнена следующая команда за инструкцией SLEEP.

Пример (для PIC16C71 и PIC16C70):

```
CLRF INTCON
BSF INTCON, ADIE
BSF ADCON, GO
SLEEP
MOVF ADRES,0
```

Статья основывается на технической документации Fact2
компании Microchip Technology Incorporated, USA.