

Окончание. Начало в № 2'2003

# Большие возможности маленького контроллера

**В предыдущем номере журнала мы описали схемные методы расширения возможностей 8-выводных Flash-микроконтроллеров PIC12F629/675 производства компании Microchip. Эти недорогие микропотребляющие контроллеры предоставляют разработчику широкое поле деятельности для создания уникальных интеллектуальных портативных устройств. В этой части статьи мы продолжим обзор возможных примеров использования микроконтроллеров, рассмотрим методы аналого-цифрового преобразования с повышенной разрядностью, а также приемы эффективного кодирования и повышения надежности программ.**

**Илья Афанасьев**

afanas@averon.ru

### Клавиатура и джамперы

Кнопки и джамперы могут быть подключены к одним и тем же выводам микроконтроллера для независимого определения их состояния с использованием минимального количества выводов микроконтроллера. Подтягивающие резисторы подключены одновременно к кнопкам и выводам джамперов. Пока кнопка не нажата и перемычка не замкнута, с вывода порта GP0/1/2/3 считывается уровень лог. «0». Для определения состояния переключателей нужно установить в «1» вывод порта GP4 и каждая замкнутая перемычка даст «1» на соответствующем выводе порта GP0/1/2/3. В то же время для определения нажатия кнопки нужно установить выход порта GP4 в «0» и считать состояние GP0/1/2/3.

Программная последовательность:

- 1) когда GP4 = 0, определяем нажатые клавиши.
- 2) когда GP4 = 1 и кнопки не нажаты — считываем состояние переключателей.

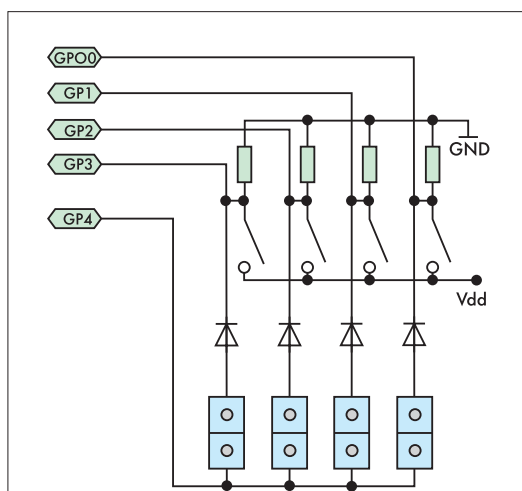


Рис. 1. Определение состояния кнопок и джамперов

### Использование микроконтроллера PIC12F675 в качестве интеллектуального датчика

Благодаря малому количеству выводов и наличию встроенного 10-разрядного АЦП микроконтроллер PIC12F675 можно использовать в качестве интеллектуального датчика физических величин. Так, например, добавив токосъемный резистор Rsen последовательно с нагрузкой, можно создать устройство для измерения и ограничения тока, устройство защитного отключения и сигнализации режимов работы, счетчик наработки ресурса и др. Резистор 10 кОм ограничивает ток через аналоговый вход и служит для его защиты.

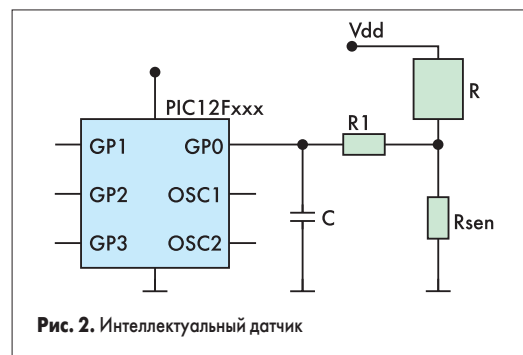


Рис. 2. Интеллектуальный датчик

### Аналого-цифровое преобразование с повышенной разрядностью

Микроконтроллеры со встроенным АЦП используются в системах регулирования, сбора и обработки данных, в устройствах сигнализации. Напряжение с датчиков в таких системах может быть измерено с помощью встроенного в микроконтроллер 10-разрядного АЦП, однако на точность измерения влияют различные внешние факторы, такие, как изменяющаяся температура, точность элементов схемы, нелинейность датчика, изменяющееся напря-

жение батареи и помехи промышленной сети. Для некоторых приложений может понадобиться точность измерения более 10 разрядов. Рассмотрим методы, которыми можно преодолеть описанные сложности с помощью любого микроконтроллера PICmicro.

**Интегрирующий АЦП**

В большинстве методов аналого-цифрового преобразования используется заряд конденсатора для сравнения напряжения на нем с измеряемым сигналом.

Напряжение заряда RC-цепи определяется формулой:

$$V_c(t) = V_{DD} \times (1 - e^{-t/(RC)})$$

тогда

$$t = -RC \ln(1 - V_{REF}/V_{DD})$$

Отношение  $V_{REF}/V_{DD}$  является постоянным, поэтому сопротивление резистора R2 определяется выражением:

$$R2 = (t2/t1) \times R1$$

Опорный (эталонный) резистор используется для увеличения точности измерения. На рис. 3 показаны зарядные кривые для R1 и R2. Время заряда RC-цепи измеряется с помощью таймера и входа порта или компаратора. Зная параметры опорного резистора R1, можно определять параметр R2 и осуществлять калибровку измерений.

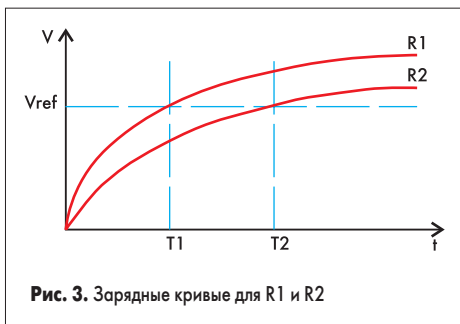


Рис. 3. Зарядные кривые для R1 и R2

Этот метод уменьшает зависимость измерений от температуры и снижает требования к точности применяемых компонентов. Более подробное описание данного метода с примерами программ можно посмотреть в заметках по применению (Application Notes) на сайте [www.microchip.com](http://www.microchip.com): «AN512 Implementing Ohmmeter/Temperature Sensor»; «AN611 Resistance and Capacitance Meter Using a PIC16C622».

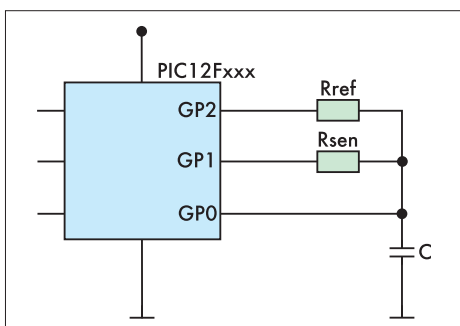


Рис. 4. Интегрирующий АЦП

Рассмотрим принцип действия такого преобразователя. Опорный ( $R_{ref}$ ) и измерительный ( $R_{sen}$ ) резисторы подключены к выходам микроконтроллера. Выход GP0 используется для разряда конденсатора и измерения времени его заряда. Таймер используется для измерения времени заряда конденсатора от момента его разряда до срабатывания встроенного компаратора.

Компаратор и источник опорного напряжения, присутствующие в микроконтроллерах PIC12F629/675, идеально подходят для данной задачи. Так как для измерительного и эталонного резисторов используется один и тот же конденсатор, то нет жестких требований к его температурной стабильности. Для повышения точности лучше использовать пленочный полипропиленовый конденсатор с минимальной остаточной поляризацией диэлектрика.

Программная последовательность:

1. Переключить GP1 и GP2 на входы, а GP0 как выход с уровнем «0» для разряда конденсатора.
2. Настроить GP0 как вход и записать в GP1 лог. «1».
3. Измерить время  $t_{sen}$  (когда состояние входа GP0 изменится в «1»).
4. Повторить пункт 1.
5. Настроить GP0 как вход и записать в GP2 лог. «1».
6. Измерить время  $t_{ref}$  (когда состояние входа GP0 изменится в «1»).
7.  $R_{sen} = (t2/t1) \times R_{ref}$ .

**Метод уравнивания зарядов**

Методы уравнивания зарядов используют конденсатор для отслеживания отношения уровня входного сигнала к эталонному напряжению. Идея этого метода заключается в поддержании постоянного уровня напряжения на измерительном конденсаторе за счет его заряда от источника измеряемого сигнала и разряда через опорный резистор. Поддержание постоянного напряжения на конденсаторе осуществляется с помощью компаратора и таймера. Каждое срабатывание таймера (такт) проверяется отношением напряжения на входе компаратора и уровня опорного напряжения. Если напряжение на конденсаторе больше опорного, то разряжаем конденсатор через резистор  $R_{ref}$ , устанавливая выход GP2 в «0». Если же напряжение на конденсаторе меньше опорного — настраиваем GP2 как вход для заряда конденсатора. Число тактов, когда GP2 установлен в «0» за фиксированный промежуток времени, определяет отношение сопротивлений  $R_{sen}$  и  $R_{ref}$ .

Так как измерения производятся на фиксированном интервале времени и выходной код пропорционален среднему значению измеряемой величины за этот интервал, то данный метод может применяться для подавления помехи, например, наводки с частотой питающей сети и ее гармоник. Для этого интервал измерения должен быть кратным периоду напряжения сети. В описанном методе также нет особых требований к свойствам измерительного конденсатора.

Наличие компаратора напряжения и программируемого источника опорного напряже-

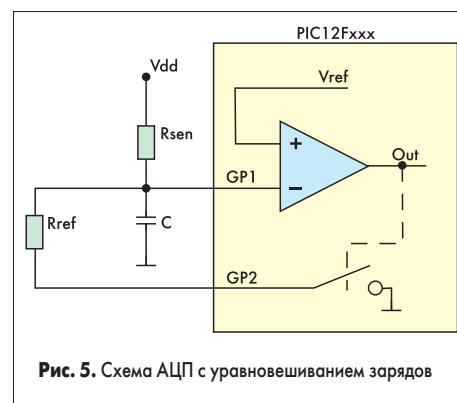


Рис. 5. Схема АЦП с уравниванием зарядов

ния в микроконтроллерах PIC12F629/675 позволяет использовать эти контроллеры для данного метода измерения.

**Расширение динамического диапазона АЦП**

Часто для измерения температуры применяются датчики с отрицательным температурным коэффициентом, которые имеют нелинейную температурную зависимость. При уменьшении температуры результирующее сопротивление датчика становится все меньше и меньше. Обычно такие датчики имеют ограниченный полезный диапазон, так как со снижением температуры измеряемый сигнал становится соизмеримым с разрешением АЦП. Динамический диапазон измерения может быть расширен, если в зависимости от уровня входного сигнала изменять сопротивление делителя напряжения, в который включен датчик. Для выбора верхнего температурного диапазона необходимо установить выход GP1 в «1» а GP2 как вход. Для измерений в области малых температур — наоборот: GP2 выход с лог. «1» а GP1 как вход. Более низкий диапазон увеличит напряжение датчика при снижении температуры и расширит диапазон измерения.

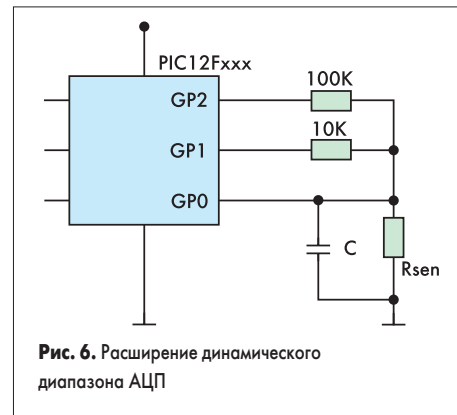


Рис. 6. Расширение динамического диапазона АЦП

**Советы по программированию**

Размер памяти микроконтроллера определяет его стоимость: чем больше объем памяти, тем больше площадь кристалла и тем больше стоимость микросхемы. Поэтому для уменьшения стоимости устройства разработчику требуется эффективно использовать память программ.

Оптимизация кода помогает избежать применения дорогих контроллеров с большим объемом памяти. Рассмотрим несколько примеров, позволяющих уменьшить размер программы.

**Временные задержки**

Микроконтроллеры обычно используются для связи с «внешним миром» посредством интерфейсных шин, светодиодов, индикаторов, кнопок и т. д. Так как контроллеры работают на фиксированной частоте, в программе часто встречаются подпрограммы задержки на определенное число тактов.

Для формирования больших временных задержек хорошо подходят команды DECFSZ и INCF SZ. Задавая начальное значение переменной, которая будет изменяться этими командами до значения нуля, можно получить задержку на нужное число тактов. Для формирования задержек на несколько циклов применение этих команд становится нецелесообразным. Для уменьшения размера кода можно использовать следующие приемы.

- Для задержки на два цикла обычно ставят две команды NOP, занимающие два слова программной памяти. Ту же самую задержку можно получить, используя всего одну команду «goto \$+1» (в ассемблере MPASM символ «\$» обозначает адрес текущей команды). При выполнении данной команды, как и при выполнении команды NOP, программный счетчик перейдет на выполнение следующей команды, но команда GOTO выполняется за два машинных цикла.
- Для задержки на восемь циклов служит следующий пример программы, состоящей всего из трех команд:

```

CALL    DELAY8t
...
DELAY8t CALL    Rtrn
Rtrn    RETURN
    
```

Обмен данными между регистром и аккумулятором можно осуществлять без исполь-

**Таблица**

W	REG	Instruction
10101100	01011100	XORWF REG,F
10101100	11110000	XORWF REG,W
01011100	11110000	XORWF REG,F
01011100	10101100	Результат

зования дополнительного регистра ОЗУ, используя всего три команды. Этот прием не требует дополнительного регистра и может быть оформлен в виде макроса (см. таблицу).

**Помехоустойчивость программ**

Малые размеры, низкая стоимость и экономичность контроллеров PIC12F629/675 позволяют их использовать в необслуживаемых датчиках, устройствах контроля и сигнализации. Поэтому важной задачей при разработке подобных устройств является обеспечение помехоустойчивости микроконтроллера и его безотказной работы длительное время. На устойчивость микропроцессорного устройства к воздействию электромагнитных помех, помимо условий эксплуатации, конструктивного исполнения устройства и разводки печатной платы, влияет и реализация программного обеспечения. Правильно спроектированное устройство должно восстанавливать свою работоспособность без вмешательства обслуживающего персонала. Остановимся подробнее на программных методах увеличения устойчивости микроконтроллерного устройства.

Вследствие воздействия помехи существует вероятность изменения состояния какого-либо служебного регистра. Так, изменение программного счетчика может привести к выходу программы на адреса, не занятые программой. Поэтому всю свободную память программ рекомендуется заполнить командой перехода

на обработчик такого события. В ассемблере MPASM для этого случая существует специальная директива FILL:

```

FILL (GOTO Reset), (MAXROM — $) ;
    
```

где Reset — адрес обработчика сбоя адресации; MAXROM — максимальный адрес памяти программ для данного контроллера.

Особое внимание следует уделить обработке прерываний. При установке флага запроса на прерывание программный счетчик аппаратно устанавливается на вектор прерываний при условии разрешения данного прерывания. Если данный флаг установился из-за воздействия помехи, а прерывание не предусмотрено и не анализируется программой, то произойдет «зависание» работы микроконтроллера. Программа будет постоянно крутиться в цикле обработки прерываний (в связи с возможностью возникновения подобной ситуации не следует помещать команду сброса сторожевого таймера WDT в прерывания). Для предотвращения подобной ситуации необходимо при возникновении несанкционированного прерывания произвести переинициализацию служебных регистров или, если это допускается в данном устройстве, сбросить контроллер, используя сторожевой таймер WDT.

```

org    h'04'
btfs   INTCON,T01F
goto   INT_1 ; обработка прерывания TMR0
...
btfs   PIR1,TMR1IF
goto   INT_N ; обработка прерывания TMR1
goto   INIT_CPU ; несанкционированное прерывание,
                инициализация
    
```

**3. Начальная инициализация**

Следует помнить, что при включении микроконтроллера или при его рестарте от сторожевого таймера состояние служебных регистров может быть неопределено или принимать значение, которое было до сброса. Игнорирование этого факта может привести к тому, что программа будет работать по-разному на разных контроллерах, а также к тому, что программа, отлаженная в симуляторе, не будет работать в «железе» и т. п. Например, из-за некорректного содержания регистра PCLATH, при использовании команд GOTO и CALL произойдет некорректный переход. Поэтому при старте программы рекомендуется провести начальную инициализацию всех служебных регистров.

Микроконтроллеры PIC12F629/675 программно и pin-to-pin-совместимы с PIC12C6xx, что облегчает перенос программ на новое семейство, а их меньшая стоимость, гибкость Flash-архитектуры, наличие памяти данных EEPROM и большая функциональность делают этот переход экономически обоснованным. В помощь разработчику фирма Microchip выпускает внутрисхемный отладчик ICD2, который позволяет производить отладку программы «внутри» устройства как в пошаговом режиме, так и в режиме реального времени, обеспечивая доступ ко всем регистрам микроконтроллера. Подробную техническую документацию и примеры программ можно получить на сайте фирмы Microchip ([www.microchip.com](http://www.microchip.com)).