

# Технологии nanoWatt и nanoWatt XLP.

## Цель — снижение энергопотребления

Перевод: Илья НИКИФОРОВ

### Введение

Прежде чем вникать в особенности энерго-сберегающих режимов работы, необходимо разобраться: что влияет на потребление тока. Применительно к микроконтроллерам, важно выделить две составляющие — ток потребления в активном (динамическом) режиме и статическом режиме (режиме ожидания).

Потребление в активном режиме обусловлено переключениями цифровой логики и зависит от частоты тактирования, напряжения питания и температуры. Преимущественное влияние имеет частота тактирования.

Статическое потребление тока имеет место при полной остановке тактирования ядра и состоит из токов утечки транзисторов, потребления супервизоров и тактирования схем, обеспечивающих дальнейшую работу контроллера (сторожевой таймер, часы реального времени и т.п.). Преимущественное влияние на статическое потребление имеет напряжение питания и температура.

Напряжение питания существенно влияет на энергопотребление контроллера как в активном режиме, так и в режиме ожидания. Таким образом, существенную выгоду можно получить за счет пониженного питания



Энергопотребление всегда является важной характеристикой любой электрической системы. В современных встроенных системах микроконтроллер должен не только управлять системой и иметь собственное малое потребление, но и управлять потреблением тока всей схемы. В статье рассмотрены возможности PIC-микроконтроллеров, выполненных по технологиям компании Microchip — nanoWatt и nanoWatt XLP, по снижению энергопотребления.

ядра контроллера в сравнении с остальной схемой. Некоторые PIC-контроллеры имеют независимый вход для питания внутренней логики (VDDCORE), что позволяет выиграть не только в активном, но и в статичном потреблении тока.

### Технологии nanoWatt и nanoWatt XLP

С 2003 года компания Microchip Technology Inc. начала выпуск контроллеров с энергосберегающей технологией nanoWatt. Основным требованием к контроллерам технологии nanoWatt было энергопотребление в режиме ожидания (Sleep) порядка нВт. Также были добавлены следующие возможности по энергосбережению:

- Режим Idle.
- Встроенный высокоскоростной тактовый генератор (INTOSC) с PLL и программируемым делителем.
- Сторожевой таймер (WDT) с увеличенным временем ожидания.
- Сверхмалопотребляющий модуль пробуждения (ULPWU).
- Малопотребляющий Timer1 и второй осциллятор (SOSC, 32 кГц).
- Малопотребляющий программно-управляемый модуль сброса (BOR).

Сегодня Microchip расширяет энергосберегающие возможности PIC-контроллеров. Новая технология nanoWatt XLP, являющаяся расширением отлично себя зарекомендовавшей nanoWatt, включает ряд новых возможностей, таких как сверхмалопотребляющий

Таблица 1. Энергосберегающие режимы PIC-контроллеров

Режим работы	Активное тактирование	Активная периферия	Варианты пробуждения	Потребляемый ток	Типичное применение
Deep Sleep <sup>1)</sup>	Timer1/SOSC* INTRC/LPRC	RTCC DSWDT DSBOR INT0	RTCC DSWDT DSBOR INT0 MCLR	< 50 нА	Устройства с батарейным питанием, большую часть времени находящиеся в режиме Sleep <sup>3)</sup>
Sleep	Timer1/SOSC* INTRC/LPRC A/D RC	RTCC WDT АЦП компараторы CVref INTx Timer1 HLVD BOR	Все источники пробуждения (см. в документации на соотв. контроллер)	50–100 нА	Большинство критичных к энергопотреблению устройств
Idle	Timer1/SOSC INTRC/LPRC A/D RC	Вся периферия	Все источники пробуждения (см. в документации на соотв. контроллер)	25% тока потребления в активном режиме (Run)	Каждый раз, когда контроллер находится в ожидании (например, в режиме прерывания)
Doze <sup>2)</sup>	Все источники тактирования	Вся периферия	Программно или по прерыванию	35–75% тока потребления в активном режиме (Run)	Приложения, требующие высокоскоростную периферию и малые ресурсы ЦП
Run	Все источники тактирования	Вся периферия	—	Согласно документации на соотв. контроллер	Обычный режим работы

Примечания. 1) только для XLP контроллеров PIC18 и PIC24; 2) только для PIC24, dsPIC и PIC32; 3) более подробно с разницей между Sleep и Deep Sleep можно ознакомиться в документе "Deciding Between Sleep And Deep Sleep"; \* Timer1/SOSC — второй осциллятор (32 кГц).

модуль пробуждения (ULPWU), специальные малопотребляющие модуль сторожевого таймера (DSWDT) и детектор снижения напряжения (DSBOR), энергосберегающий режим «глубокого сна» Deep Sleep, а также меньший ток потребления в режимах, унаследованных от nanoWatt. В следующих режимах технология nanoWatt XLP обеспечивает ток потребления контроллера не более:

- 100 нА — режим ожидания (Ipd);
- 800 нА — сторожевой таймер (Iwdt);
- 800 нА — часы реального времени с календарем (Irtcc).

Надо понимать, что в обширной номенклатуре контроллеров Microchip различные энергосберегающие режимы, специальные модули и настройки, а также токи потребления модулей контроллера варьируются от семейства к семейству, поэтому строго выделить отличия между технологиями nanoWatt и nanoWatt XLP невозможно, да и не нужно: данная статья дает общее представление об энергосберегающих возможностях PIC-контроллеров.

Технологии энергосбережения nanoWatt и nanoWatt XLP предоставляют разработчику различные режимы работы с гибкими возможностями по программному управлению аппаратными средствами, что позволяет минимизировать потребление тока при выполнении каждой задачи. В таблице 1 приведен перечень энергосберегающих режимов PIC-контроллеров. По данным таблицы 2 можно сравнить несколько nanoWatt PIC-контроллеров с конкурентами, производства других фирм.

### Режим Deep Sleep

Режим Deep Sleep — самый экономичный режим работы контроллера, когда все модули, которые могут остаться без питания — ядро контроллера, встроенный стабилизатор напряжения, большая часть периферии, ОЗУ — переводятся в энергосберегающий режим. Соответственно, остается только несколько источников пробуждения контроллера:

- Схема сброса по включению питания POR.
- Сброс по MCLR.
- Будильник часов реального времени (RTCC).
- Внешнее прерывание.
- Сторожевой таймер “WDT Deep Sleep”.

В связи с тем, что тактирование ядра приостановлено, выход из режима Deep Sleep возможен только со сбросом контроллера. Программный счетчик и регистры специального назначения (SFR) также сбрасываются, и выполнение программы начинается с вектора сброса. При этом состоянии портов ввода/вывода, Timer1/SOSC и RTCC сохраняются. Следует отметить, что гибкая конфигурация Deep Sleep позволяет сохранить значения в некоторой области RAM, что позволяет программе правильно восстановиться после сброса.

Таблица 2. Сравнительные характеристики по потреблению нескольких nanoWatt PIC-контроллеров

Параметр	Контроллер 1							
	PIC16LF72x	PIC18F46K20	PIC18LF46J11	PIC16LF193x	PIC18LF14K50	PIC24F16KA102	Atmel Atmega168P/328P	TI MSP430F21x1/MSP430F21x2/MSP430F22x2/4
Ток в режиме Deep Sleep, нА	—	—	13	—	—	20	—	—
Ток в режиме Sleep, нА	20	100	54	60	24	25	100 <sup>1)</sup>	100
Ток в режиме WDT, нА	500	500	830	500	450	420	4200 <sup>1)</sup>	300–700
Ток 32 кГц осциллятора RTCC, нА	600	500	820	600	790	520	800	700
Ток утечки портов ввода/вывода, нА	±5	±5	±200 <sup>2)</sup>	±50	±5	±50	±1000 <sup>1,2)</sup>	±50 <sup>2)</sup>
Ток в режиме Run 1 МГц, мкА	110	300	272	150	170	195	300	200–270
Минимальное напряжение питания Vdd, В	1,8	1,8	2	1,8	1,8	1,8	1,8	1,8

**Примечания.**

Указаны типичные значения при напряжении питания Vdd = 1,8–2 В из описаний производителя на контроллер;

1) данные для Vdd = 1,8 В недоступны, указано для 3 В;

2) типичное значение недоступно, указано максимальное согласно документации на контроллер.

После пробуждения программа должна подтвердить сброс, перенастроить периферию и регистры портов ввода/вывода и затем продолжить выполнение кода. Общий алгоритм приведен на блок-схеме (рисунок). Для более подробного описания необходима документация на соответствующий контроллер.

Очень важно иметь четкое представление о том, какой именно энергосберегающий режим необходимо использовать на каждом этапе работы контроллера. Deep Sleep — режим, актуальный для приложений, требующих длительного срока работы от батарейного питания. Обычно приложения, для которых необходим режим Deep Sleep, имеют следующие особенности:

- Длительное время пребывания в режиме ожидания (обычно от 1 секунды).
- Не требуется работы периферии в режиме ожидания.
- Точная привязка ко времени при минимальном потреблении тока.
- Широкий диапазон рабочих температур.

### Режим Sleep

Режим Sleep имеют все PIC-контроллеры, выполненные по технологии nanoWatt. В режиме Sleep отключается тактирование ядра и большей части периферии. Потребление тока складывается из питания ОЗУ, SFR и программного счетчика.

Источники пробуждения различны для разных семейств, однако все PIC-контроллеры обладают следующими: сторожевой таймер WDT, 32-кГц таймер/осциллятор (для большинства контроллеров — Timer1) и внешние источники прерываний. Контроллеры PIC18, PIC24 и PIC32 имеют большое число периферийных модулей, обеспечивающих возможность пробуждения контроллера, такие как АЦП, компараторы и модули последовательных коммуникационных интерфейсов.

Sleep — наиболее востребованный, гибко настраиваемый энергосберегающий режим PIC-контроллеров, который характеризуется предельно быстрым входом и выходом

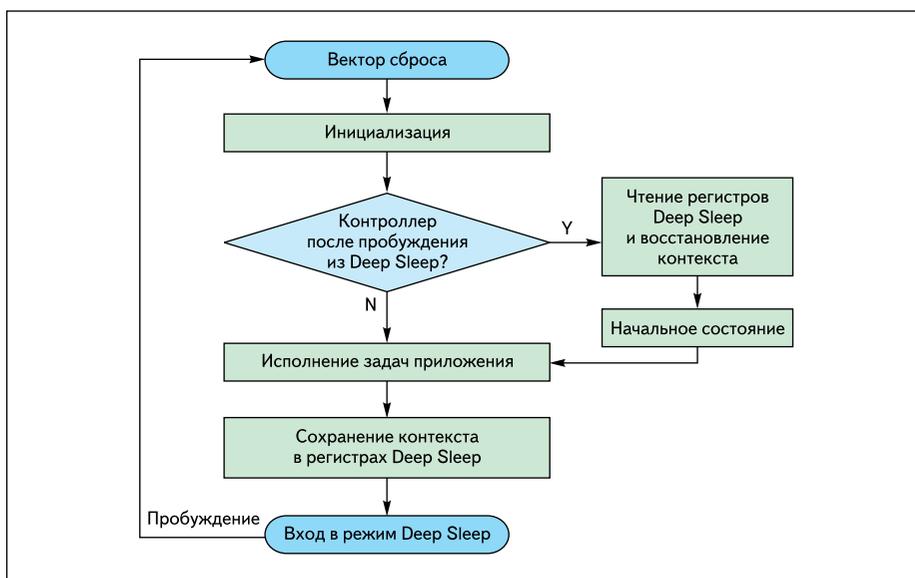


Рисунок. Общий алгоритм работы режима Deep Sleep

из режима ожидания. Как следствие, данный режим предназначен для устройств, для которых необходимы частый переход в статический режим на непродолжительные промежутки времени и быстрое пробуждение. Обычно приложения, предполагающие использование Sleep, имеют следующие особенности:

- Время пребывания в статическом режиме невелико (обычно не более секунды).
- Требуется пробуждение от периферии.
- Требуется работа АЦП или компараторов в энергосберегающем режиме.

### Выбор между Sleep и Deep Sleep

Чтобы выбрать оптимальный режим, необходимо рассчитать характерное время пребывания контроллера в режиме ожидания ( $T_{BE}$ ). Приведем формулы для расчета  $T_{BE}$ :

$$Q_{SLP} = T_{PD} \times I_{PDSL}, \quad (1)$$

$$Q_{DS} = (T_{INIT} \times I_{DD}) + (T_{POR} \times I_{POR}) + (T_{PD} \times I_{PDDS}), \quad (2)$$

$$T_{BE} = T_{PD} = [(T_{INIT} \times I_{DD}) + (T_{POR} \times I_{POR})] / (I_{PDSL} - I_{PDDS}). \quad (3)$$

Сначала рассчитывается потребление энергии в режимах Sleep ( $Q_{SLP}$ ) и Deep Sleep ( $Q_{DS}$ ). Если быть точнее, то рассчитывается протекающий заряд, так как напряжение постоянно и одинаково для обоих режимов. В режиме Sleep энергопотребление определяется произведением тока потребления ( $I_{PDSL}$ ) и времени пребывания в данном режиме ( $T_{PD}$ ) (1).

В режиме Deep Sleep необходимо учесть три составляющих (2):

- Потребление в режиме ожидания (ток потребления ( $I_{PDDS}$ ) на время пребывания в режиме Deep Sleep ( $T_{PD}$ )).
- Потребление в момент пробуждения по сбросу POR (с момента появления события, вызвавшего пробуждение, до начала выполнения первой инструкции). Обычно  $T_{POR}$  — значение из документации на контроллер,  $I_{POR}$  — замеряется эксперименталь-

но и зависит от условий работы контроллера. Также здесь учитывается время и ток, необходимые на зарядку конденсатора  $V_{CAP}$  (если он разряжается в режиме Deep Sleep) для контроллеров со встроенным стабилизатором напряжения.

- Потребление контроллера ( $I_{DD}$ ) на время исполнения кода инициализации ( $T_{INIT}$ ) до входа в основной цикл программы. Потребление контроллера в активном режиме указано в документации на контроллер, а время, необходимое для инициализации, удобно замерять при помощи секундомера Stopwatch в единой среде разработки MPLAB IDE.

Характерное время пребывания в режиме ожидания ( $T_{BE}$ ) соответствует равенству  $Q_{SLP}$  и  $Q_{DS}$ . Режим Deep Sleep предпочтительнее, когда ожидаемое время пребывания в энергосберегающем режиме превышает  $T_{BE}$  Sleep — в обратном случае.

### Режимы Idle и Doze

Режимы Idle и Doze занимают промежуточное положение между энергосберегающими режимами и основным (Run).

В режиме Idle тактирование ядра приостановлено, но большая часть, а то и вся периферия сохраняет свою функциональность и может продолжать работать (подробнее об этом сказано в документации на соответствующий контроллер). В контроллерах PIC24, PIC32 и dsPIC есть возможность индивидуальной настройки периферийных модулей, задействованных в режиме Idle.

В режиме Doze (он имеется только в PIC24, PIC32 и dsPIC) периферия может работать на полной скорости, а тактирование ядра возможно на пониженной частоте, полученной делением системной на программно заданный коэффициент.

### Выбор между Idle и Doze

Режимы Idle и Doze позволяют сократить потребление тока в тех случаях, когда вход в режим ожидания невозможен. Например:

- Необходимы длинные передачи посредством DMA.
- Требуется прием или передача данных по последовательным протоколам.
- Требуется работа высокоскоростного АЦП.
- Ожидание по синхронному таймеру.
- Сбор данных от внешних датчиков.
- Требуется работа модулей захвата/сравнения/ШИМ.

Каждый раз, когда контроллер находится в ожидании некоторого события (прерывания от периферии), обычно имеет смысл перейти в режим Idle или Doze. Правильный выбор оптимального энергосберегающего режима позволяет существенно снизить потребляемый конечным устройством ток.

### Изменение частоты тактирования

Возможность изменения частоты тактирования — очень важный инструмент, грамотное использование которого позволяет существенно снизить энергопотребление в активном режиме. В то время как режимы Idle и Doze позволяют снизить скорость работы ядра, периферия тактируется максимальной частотой, потребляя существенный ток. Поэтому целесообразно адаптивно выбирать частоту тактирования периферии в каждом конкретном случае.

Микроконтроллеры PIC позволяют гибко управлять тактированием ядра и периферийных модулей. Зачастую правильное управление тактированием позволяет получить потребление тока в активном режиме ниже, чем в режимах Idle и Doze!

### Заключение

Новые технологии nanoWatt и nanoWatt XLP предоставляют разработчикам возможность гибко управлять энергопотреблением, что позволяет существенно снизить потребление тока. Благодаря технологии nanoWatt XLP, компании Microchip удалось создать PIC-контроллеры с самым низким потреблением тока в режиме Sleep. ■