

# Микроконтроллеры Microchip: шаг второй

**Евгений Рахно**, ведущий специалист ООО «Гамма»  
E-mail: info@microchip.ua

**В предыдущей части (№ 2, 2009) мы рассмотрели список необходимого оборудования для начала работы с микроконтроллерами компании Microchip. В этой статье расскажем о том, что такое «встроенная система», и бегло ознакомимся с интегрированной средой разработки MPLAB IDE.**

Что такое «встроенная система» или «embedded system». Это некое устройство, разработанное с использованием микроконтроллера, к примеру, микроконтроллера PIC® или цифрового сигнального контроллера dsPIC®, и служащее для реализации какой-либо определенной функции. Зачастую, такое устройство состоит из одного или нескольких микроконтроллеров, установленных на печатную плату, а также имеет дополнительные модули, называемые «периферия» или «периферийные устройства». Данное устройство встраивается внутрь механического, электрического или электро-механического изделия, отсюда и название — «встроенные системы». Самый простой и самый понятный пример — кофе-автомат. Вы опускаете купюру в автомат, выбираете требуемый напиток, а он подбирает ингредиенты и следит за приготовлением кофе. Данное устройство слишком сложное для реализации его на микросхемах логики, но слишком простое, для того, чтобы использовать компьютер.

Также хорошим примером встроенной системы может служить детектор дыма. Задача подобной встроенной системы — преобразовать визуальный образ «дым» в звуковой и световой сигнал, оповещающий о пожаре. В такой системе преимущество микроконтроллера над компьютером становится еще более очевидным.

Что такое компьютер — знают все, давайте попробуем разобраться с термином «микроконтроллер».

Микроконтроллер фирмы Microchip состоит из таких структурных частей:

- встроенная Flash память программ для хранения «программы», или «firmware», как сейчас принято говорить;
- Program Counter (PC), который предназначен для адресации памяти программ, а также для обслуживания прерываний и векторов сброса;
- аппаратный стек, служащий для операций call и return, работает совместно с программой, но не является частью памяти программ;
- память данных или «file register».

Память данных состоит из двух видов памяти — SFR (Special Function Registers или регистры специального назначения) и GPR (General Purpose Registers или регистры общего назначения). SFR используется процессором для хранения данных о работе периферийных устройств, а также для своих собственных нужд, в то время как SFR используются для хранения переменных пользователя (именно этот раздел памяти данных мы и называем «память»). Также многие микроконтроллеры содержат дополнительный раздел EEPROM памяти, используемый для длительного энергонезависимого хранения информации.

В дополнение к памяти программ и памяти данных микроконтроллер имеет богатый набор встроенных периферийных модулей. Самым распространенным «периферийным модулем», как ни странно, является модуль портов ввода/вывода. Они могут использоваться для анализа входных данных (нажата

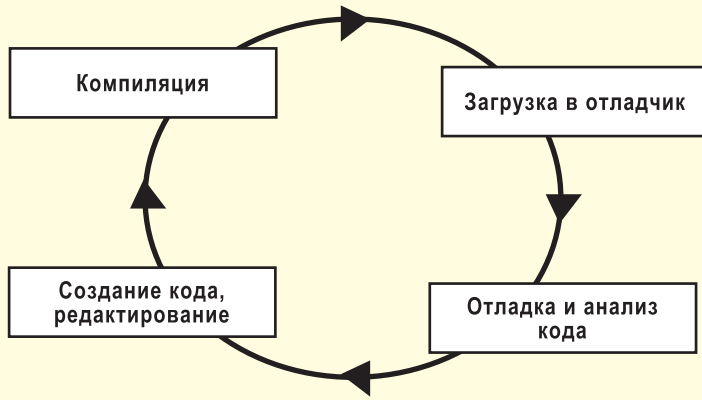
ли кнопка?), а также для установления сигнала низкого/высокого уровня, мигания светодиодами, питания пищалками — словом, для чего угодно, что может передаваться по проводам. В 99 случаях из 100 порт ввода/вывода является двунаправленным портом, то есть он может как принимать данные, так и передавать их.

Для создания полноценной встроенной системы требуются и другие периферийные модули, например, модули аналого-цифрового преобразования для анализа аналоговых сигналов, модули последовательной передачи данных для связи с другими контроллерами или выхода в сеть/интернет, «таймеры» для точного измерения либо генерации временных интервалов и т.д.

Что такое «встроенная система» мы уже разобрались, теперь попробуем понять, как эту систему создать.

Очевидно, что наиболее рациональной последовательностью действий является такая:

1. Разработка технического задания (ТЗ). Исходя из функций, описанных в ТЗ, необходимо определиться с выбором микроконтроллера и набором периферии, которые требуются для выполнения поставленной задачи. Разработка схемотехники и программного обеспечения.
2. Написание кода программы, компиляция и сбор кода в единое целое, преобразование кода из  $a = b + 7 \cdot c$  в «нолики и единички», понятные микроконтроллеру.
3. Тестирование программы. Зачастую, даже самый опытный программист пишет абсолютно не работающую программу. Несколько десятков «глюков», синтаксические ошибки и логические промахи делают ее недееспособной. Процессы тестирования и отладки дают возможность уви-



**Рисунок 1** Написание программы

дет, как будут вести себя «нолики и единички» в реальном контроллере, а также позволяют программисту получить ответы на многочисленные вопросы «а что, если...?»

4. Прошивка, т.е. «программирование» микроконтроллера, и проверка работоспособности программы в реальном устройстве.

Схематически работу над программным обеспечением можно изобразить так, как показано на рис. 1.

Первая стадия проекта, безусловно, возлагается на разработчика, а вот для пунктов 2–4 предназначена интегрированная среда разработки MPLAB IDE ([www.microchip.com/ide](http://www.microchip.com/ide)).

Все эти этапы Вы можете осуществить под управлением одной единственной программы — MPLAD IDE.

Неотъемлемой частью написания программы является «ассемблер» либо «компилятор» (рис. 2).

При помощи компилятора Вы можете преобразовать понятный Вам язык программирования в исполняемый микроконтроллером код.

Менеджер проектов MPLAB IDE позволяет использовать не только встроенный в MPLAB IDE бесплатный язык программирования ассемблер, но

и подключать как собственные платные, языки программирования C, так и языки программирования сторонних фирм.

На сегодняшний день осуществляется поддержка таких языков как:

- Microchip MPASM;
- Microchip C18 (PIC18);
- Microchip ASM30 (PIC24 и dsPIC);
- Microchip C30 (PIC24 и dsPIC);
- Microchip ASM32 (PIC32MX);
- Microchip C32 (PIC32MX);
- В Knudsen Data (PIC12/14/16/18);
- Byte Craft (Assembler/C Compiler для PIC12/14/16/17);
- CCS (PIC12/14/16/18 и dsPIC)
- HI-TECH (PIC10/12/16/18 и dsPIC/PIC24);
- IAR (PIC16/17/18 и dsPIC);
- microEngineering Labs (PicBasic).

К сожалению, большинство компиляторов языка высокого уровня являются платными, однако оценить преимущества того или иного продукта позволяющие бесплатные демо-версии.

MPLAB IDE включает в себя следующие модули:

- **редактор кода;**
  - **модуль работы с отладчиком.**
- Существует три типа отладчиков:
- симулятор (только «виртуальная» отладка программы, пос-

тавляется бесплатно в составе MPLAB IDE) — MPLAB® SIM Simulator;

- внутрисхемный отладчик (аппаратно-программная модель): MPLAB® ICD 3; MPLAB ICD 2; PICkit™ 3 Debug Express; PICkit 2 Debug Express (именно работу с этим отладчиком мы рассмотрим в дальнейшем);
- внутрисхемный эмулятор (расширенная аппаратно-программная модель с поддержкой обширного набора сервисных функций): MPLAB® REAL ICE™; MPLAB ICE 2000;

#### • **Модуль работы с программатором:**

- MPLAB® PM3;
- PICSTART® Plus;
- MPLAB REAL ICE;
- MPLAB ICD 3;
- MPLAB ICD 2;
- PICkit™ 3 Debug Express;
- PICkit 2 Debug Express.

На этом мы закончим очень краткое описание MPLAB IDE.

На нашем сайте Вы найдете подробное описание следующих приемов работы с MPLAB IDE:

- выбор устройства;
- создание проекта;
- выбор компилятора;
- сохранение проекта;
- добавление файла в проект;
- компиляция проекта;
- создание hex файла;
- тестирование кода в симуляторе;
- редактирование кода.

#### **Более детальную информацию можно получить в офисе ООО «Гамма»:**

**49005, г. Днепропетровск, ул. Фурманова, 15, оф. 101, тел. (0562) 36-07-92, [http:// www.microchip.ua](http://www.microchip.ua)**

```

int main (void)
{
    counter = 1;
    TRISB =0;
    while (input1 = 0)
    {
        PORTB = count;
        counter++;
    }
}
  
```

компилятор

```

01101111
10001000
11001101
10100001
00110011
01011101
00110001
11100101
  
```

**Рисунок 2** Компиляция