СОДЕРЖАНИЕ

Введение 5
Структура книги 6

1. Обзор 16-битных PIC-микроконтроллеров 8

2. Архитектура микроконтроллеров PIC24F 10

3. Система команд и основы программирования микроконтроллеров PIC24F 18

3.1. Программная модель микроконтроллеров PIC24F 18
3.2. Режимы адресации и система команд 20
3.2.1. Команды перемещения и адресация данных 23
3.2.2. Команды сравнения/выбора и условного перехода 28
3.2.3. Команды работы с битами 32
3.2.4. Команды сдвига и циклического сдвига 35
3.2.5. Команды математических и логических операций 37
3.2.6. Команды условных/безусловных переходов 44
3.3. Разработка и отладка программ на ассемблере в среде MPLAB IDE 46
3.4. Особенности разработки и отладки программ на MPLAB C для PIC24 59

4. Программирование портов ввода/вывода 73

4.1. Аппаратно-программная архитектура портов ввода/вывода 73
4.2. Программирование портов ввода/вывода 77
4.3. Модуль регистрации событий 81

5. Программирование прерываний 89

6. Программирование таймеров 100

6.1. Практическое использование 16-битных таймеров 104
6.2. Работа таймеров в 32-битном режиме 114
6.3. Часы реального времени 118

7. ИНТЕРФЕЙС SPI МИКРОКОНТРОЛЛЕРОВ PIC24F 120

7.1. Аппаратно-программная реализация SPI в микроконтроллерах PIC24F 121
7.2. Практическое программирование обмена данными по SPI 127

8. ИНТЕРФЕЙС I²C МИКРОКОНТРОЛЛЕРОВ PIC24F 140

8.1. Принципы функционирования интерфейса I²C 140
8.2. Модуль интерфейса I²C микроконтроллеров PIC24F 143
8.3. Практическое использование интерфейса I²C 147

9. ПРОГРАММИРОВАНИЕ ИНТЕРФЕЙСА PMP 159

9.1. Режимы работы PMP 161
9.2. Практические примеры программирования интерфейса PMP 164

10. ПОСЛЕДОВАТЕЛЬНЫЙ ИНТЕРФЕЙС МИКРОКОНТРОЛЛЕРОВ PIC24F 182

10.1. Аппаратно-программная архитектура UART 183
10.2. Практическое использование последовательного порта 184

11. ОБРАБОТКА АНАЛОГОВЫХ СИГНАЛОВ В МИКРОКОНТРОЛЛЕРАХ PIC24F 198

11.1. Программная модель интегрированного АЦП 201
11.2. Практическое использование модуля АЦП 205
11.3. Использование внешнего АЦП 217

12. ГЕНЕРАЦИЯ АНАЛОГОВЫХ И ЦИФРОВЫХ СИГНАЛОВ 221

12.1. Модуль генерации цифровых сигналов 221
12.2. Аналоговые компараторы в микроконтроллерах PIC24F 229

Заключение 239

ВВЕДЕНИЕ

Системы управления и контроля на однокристальных микроконтроллерах в настоящее время используются практически во всех сферах человеческой деятельности, причем каждый день появляются все новые и новые области применения таких систем. В последнее время, в связи с бурным развитием электроники расширились возможности и самих микроконтроллеров, позволяющих выполнять многие задачи, ранее недоступные для реализации, такие, например, как обработка и синтез аналоговых аудио- и видеосигналов. Одним из наиболее популярных семейств микроконтроллеров являются устройства, выпускаемые фирмой Microchip и известные под аббревиатурой PIC. В последние годы были разработаны и запущены в производство 16- и 32-битные модели, позволившие существенно расширить области применения систем на базе микроконтроллеров PIC. Для облегчения миграции от 8- к 16-битным устройствам фирма Microchip обеспечила максимальный уровень совместимости аппаратно-программных функций этих микроконтроллеров.

Эта книга посвящена практическим аспектам разработки систем на базе 16-битных микроконтроллеров PIC24F. В книге приводятся основные сведения по аппаратно-программной архитектуре микроконтроллеров PIC24F, а также многочисленные проекты систем обработки данных на базе микроконтроллеров этого семейства. Все примеры, приведенные в книге, разработаны и протестированы на плате «Explorer 16 Development Board» производства Microchip с установленным микроконтроллером PIC24FJ128GA010. Тем не менее, для тестирования приведенных примеров и разработки собственных проектов читатели могут использовать и более дешевую систему начального уровня «MPLAB Starter Kit for PIC24F» той же фирмы или отладочные платы других фирм. Для разработки программного обеспечения проектов, представленных в книге, использовалась интегрированная среда разработки MPLAB IDE версии 8.00 и бесплатная студенческая версия компилятора языка Си, известного под названием MPLAB C для PIC24. Кроме того, при изучении системы команд микроконтроллера PIC24F, а также при анализе процесса отладки некоторых программ на языке Си используется довольно эффективный симулятор/отладчик MPLAB SIM.

Книга рассчитана на широкий круг читателей — от начинающих до опытных разработчиков – и может оказаться полезной для всех, кто желает самостоятельно изучить аппаратно-программную архитектуру 16-битных микроконтроллеров PIC24F и применить эти знания на практике.

СТРУКТУРА КНИГИ

Структура книги рассчитана на последовательное изучение материала, хотя опытные разработчики систем на базе микроконтроллеров PIC могут изучать материал выборочно. Теоретический материал большинства глав подкреплен практическими примерами, что позволяет существенно облегчить изучение.

Книга состоит из 12 глав, краткие сведения о каждой из них приведены далее:

- Глава 1. «Обзор 16-битных PIC-микроконтроллеров». В этой главе рассматриваются общие характеристики 16-битных микроконтроллеров фирмы Microchip и дается краткий обзор инструментальных средств разработки программного обеспечения.
- Глава 2. «Архитектура микроконтроллеров PIC24F». Материал этой главы посвящен обзору аппаратной архитектуры микроконтроллеров PIC24F, организации памяти и функционирования периферийных модулей устройства.
- Глава 3. «Система команд и основы программирования микроконтроллеров PIC24F». В этой главе детально проанализированы принципы реализации системы команд микроконтроллеров PIC24F и приведены многочисленные примеры программного кода на языке ассемблера. Значительная часть главы посвящена вопросам разработки и отладки программного обеспечения в среде MPLAB IDE.
- Глава 4. «Программирование портов ввода/вывода». Эта глава содержит материал по архитектуре и программированию цифровых портов ввода/вывода микроконтроллеров PIC24F. Теоретический материал сопровождается примерами программ на языке Си с детальным анализом программного кода.
- Глава 5. «Программирование прерываний». Материал главы посвящен аппаратно-программной реализации системы прерываний микроконтролле-

ров PIC24F. На многочисленных примерах программного кода проанализированы различные механизмы вызова и обработки пользовательских прерываний.

- Глава 6. «Программирование таймеров». Глава содержит материал по аппаратно-программной архитектуре модулей таймеров микроконтроллеров PIC24F. Рассматриваются различные режимы работы таймеров и их настройка. Теоретический материал сопровождается примерами программ на языке Си с детальным анализом программного кода.
- Глава 7. «Интерфейс SPI микроконтроллеров PIC24F». В главе рассмотрены принципы функционирования и аппаратно-программная реализация модуля интерфейса SPI, а также приведены примеры программирования систем ввода/вывода дискретных данных с использованием данного интерфейса. Все примеры сопровождаются детальным анализом программного кода.
- Глава 8. «Интерфейс I^2C микроконтроллеров PIC24F». Материал главы посвящен рассмотрению принципов функционирования и аппаратно-программной конфигурации модуля интерфейса I^2C в микроконтроллерах PIC24F. Теоретический материал сопровождается примерами программирования обмена данными по шине I^2C на языке Си с детальным анализом программного кода.
- Глава 9. «Программирование интерфейса PMP». В главе рассматривается аппаратно-программная реализация 8-битного параллельного интерфейса обмена данными PMP. Приводятся примеры разработки систем ввода/вывода цифровых данных с использованием этого интерфейса.
- Глава 10. «Последовательный интерфейс микроконтроллеров PIC24F». В этой главе рассматриваются принципы реализации и настройки модуля последовательного интерфейса микроконтроллеров PIC24F. Теоретический материал сопровождается примерами программирования обмена данными с использованием этого модуля, разработанными на языке Си, с детальным анализом программного кода.
- Глава 11. «Обработка аналоговых сигналов в микроконтроллерах PIC24F». Эта глава посвящена методам обработки аналоговых сигналов в микроконтроллерах PIC24F. Здесь рассматривается широкий круг вопросов, связанных с настройкой и использованием модуля аналого-цифрового преобразователя, а также приводятся примеры программирования ввода данных посредством АЦП.
- Глава 12. «Генерация аналоговых и цифровых сигналов». В этой главе рассматривается широкий круг вопросов, связанных с генерацией цифровых и аналоговых сигналов в микроконтроллерах PIC24F. Теоретические аспекты иллюстрируются примерами программного кода на языке Си.

Автор благодарит коллектив издательства «ДМК Пресс» за помощь при подготовке книги к изданию. Особую признательность автор выражает своей жене Юлии за поддержку и помощь при написании книги.

ГЛАВА 1

ОБЗОР 16-БИТНЫХ PIC-МИКРОКОНТРОЛЛЕРОВ

Разработанные фирмой Microchip 16-битные микроконтроллеры являются очередным шагом на пути повышения производительности и эффективности встроенных и мобильных приложений. Эта 16-битная архитектура изначально разрабатывалась как альтернатива 8-битным решениям и призвана заменить в ближайшее время 8-битные микроконтроллеры в большинстве приложений.

Разработанная фирмой Microchip 16-битная платформа реализована в двух семействах 16-битных микроконтроллеров и в двух семействах цифровых сигнальных контроллеров. Все эти семейства объединяет ряд общих характеристик:

- совместимость по назначению выводов различных 16-битных устройств;
- возможность использования для всех устройств одних и тех же инструментальных средств разработки программного обеспечения;
- аппаратно-программная совместимость всех одноименных периферийных модулей микроконтроллеров;
- общая базовая система команд процессора, используемая во всех семействах.

Выбор той или иной модели микроконтроллера или сигнального контроллера зависит от требований к разрабатываемому приложению. Для большинства недорогих устройств средней производительности подходят микроконтроллеры PIC24F, максимальная производительность которых составляет 16 MIPS. Для устройств, требующих высокой производительности, можно использовать микроконтроллеры PIC24H с максимальным быстродействием 40 MIPS. Микроконтроллеры семейств PIC24F и PIC24H работают с одним и тем же набором инструкций процессора, включают одни и те же периферийные модули, имеют одну и ту же цоколевку, и для работы с ними используются одни и те же инструментальные средства для разработки программного обеспечения.

Если требуются дополнительные возможности по обработке сигналов, то вместо микроконтроллеров семейств PIC24F/H можно применить цифровые сигнальные контроллеры семейства dsPIC30F, которые могут помимо всего прочего работать при напряжении питания 5 В, или высокопроизводительные (40 MIPS) контроллеры dsPIC33F, которые имеют большой объем памяти и используют низковольтное (3.3 В) питание. В качестве инструментального

средства разработки программного обеспечения 16-битных микроконтроллеров и цифровых сигнальных контроллеров используется свободно распространяемая интегрированная среда разработки (ИСР) MPLAB IDE фирмы Microchip, которая позволяет разрабатывать и отлаживать 8-, 16- и 32-битные приложения. Программа MPLAB IDE работает под управлением операционных систем Windows 2000/XP/Vista и позволяет выполнить все этапы разработки и отладки программного обеспечения для целевой системы. Среда MPLAB IDE позволяет выполнять тестирование и отладку программ с использованием мощного программного симулятора MPLAB SIM. Кроме того, для разработки программного обеспечения для 16-битных систем в среде MPLAB IDE можно использовать следующие инструментальные средства:

- ассемблер ASM30 — полнофункциональный макроассемблер, в котором можно создавать пользовательские макросы и использовать условное ассемблирование. Многочисленные директивы языка делают макроассемблер очень мощным средством разработки программ;
- компилятор программ, написанных на языке Си, который называется MPLAB C для PIC24. Этот компилятор используется для компиляции и оптимизации программ, написанных для 16-битных микроконтроллеров PIC24F/H и цифровых сигнальных контроллеров dsPIC30/33. Он совместим со стандартом ANSI C и включает полную библиотеку стандартных функций ANSI C, в числе которых функции манипулирования строками, функции работы с динамической памятью, функции преобразования даты/времени и математические функции. В компиляторе MPLAB C для PIC24 имеется мощный оптимизатор, позволяющий почти в 1,5 раза уменьшить размер программного кода по сравнению с компиляторами других фирм-производителей;
- визуальный генератор кода инициализации MPLAB VDI, позволяющий значительно упростить процесс создания инициализационного кода программы. С помощью VDI можно в графическом виде сконфигурировать устройство и по завершении вставить сгенерированный программный код инициализации в программу на языке Си или ассемблере;
- библиотеку периферийных модулей, включающую более чем 270 функций для работы с различными периферийными модулями;
- библиотеку математических функций, совместимую со стандартом IEEE-754, которая включает ряд функций для выполнения операций над обычными вещественными числами и вещественными числами с двойной точностью. Функции этой библиотеки могут использоваться как в программах на языке Си, так и на ассемблере.

Кроме инструментальных средств разработки и отладки программного обеспечения фирмы-производителя на рынке присутствуют и программные средства, выпускаемые многими известными фирмами (Hi-Tech, CCS и т.д.). Из аппаратных средств разработки наиболее известна и популярна отладочная плата «Explorer 16 Development Board» фирмы Microchip, хотя другие фирмы также приступили к выпуску отладочных плат на базе 16-битных микроконтроллеров.

ГЛАВА 2

АРХИТЕКТУРА МИКРОКОНТРОЛЛЕРОВ PIC24F

Микроконтроллеры PIC24F были разработаны как недорогое аппаратно-программное решение для перехода от 8-битных микроконтроллеров семейства PIC18 к 16-битной архитектуре, призванное обеспечить максимальную преемственность как уже разработанных приложений для PIC18, так и вновь создаваемых, более эффективных и недорогих 16-битных решений.

Выбор микроконтроллеров PIC24F оправдан в тех случаях, когда необходимо обеспечить среднюю производительность системы при относительно невысокой стоимости конечного продукта. Для приложений, требующих более высокой производительности (выше, чем 16 MIPS), можно использовать более дорогие микроконтроллеры семейства PIC24H.

Микроконтроллеры PIC24F обладают следующими характеристиками:

- высокая производительность (до 16 MIPS);
- векторная система прерываний с 16 уровнями приоритетов;
- наличие 16 рабочих регистров;
- возможность выполнения 16-битных математических операций;
- возможность выполнения операций умножения с разрядностью 17 × 17 бит за один машинный цикл;
- возможность выполнения сдвига на произвольное количество бит (до 16) за один машинный цикл;
- аппаратно-программная архитектура, оптимизированная для разработки программ на языке Си;
- мощная система команд, которая включает инструкцию повторения `repeat` для циклического выполнения команд, что особенно полезно при использовании команд пересылки данных.

По сравнению с микроконтроллерами PIC18 микроконтроллеры PIC24F обладают многими кардинальными улучшениями. Во-первых, в микроконтроллерах PIC24F расширен объем оперативной и флэш-памяти, что при прочих равных условиях позволяет оптимизировать обработку больших объемов данных и создавать более высокопроизводительные системы управления и обработки данных. Еще одним существенным улучшением стало включение в состав микроконтроллера дополнительных периферийных модулей.

В микроконтроллерах семейства имеется пять 16-битных таймеров, четыре из которых можно каскадировать, получая два 32-битных. Кроме того, на кристалле микроконтроллера находится интегрированный 10-битный аналого-цифровой преобразователь последовательного приближения, который может выполнять преобразование аналоговых сигналов со скоростью до 500 тыс. выборок в секунду. В микроконтроллерах семейства также реализован модуль JTAG, который позволяет выполнять тестирование и программирование микроконтроллера в системе.

Микроконтроллеры PIC24F могут взаимодействовать с различными внешними периферийными устройствами посредством интерфейсов I^2C, SPI и UART. Для этого в состав устройства включены соответствующие модули, которые могут настраиваться и работать независимо друг от друга. Функциональность подсистемы асинхронной последовательной передачи данных улучшена за счет включения в UART аппаратно-программного субмодуля IrDA. Существенно улучшает рабочие характеристики модулей интерфейсов SPI и UART сериализация данных в буфере FIFO, которая позволяет снизить непроизводительные траты процессорного времени на обработку передачи данных. В отличие от многих устройств семейства PIC18, в которых реализован порт параллельной передачи данных, работающий только в режиме «ведомого», в микроконтроллерах PIC24F имеется модуль параллельного обмена данными, который позволяет работать как в режиме «ведомого», так и в режиме «ведущего». Это существенно расширяет возможности PIC24F при обмене данными с периферийными устройствами, имеющими параллельный интерфейс (принтеры, сканеры, устройства внешней памяти и т. д.).

Все устройства семейства PIC24F имеют один и тот же набор базовых периферийных модулей и отличаются объемом флэш-памяти. Обобщенная функциональная схема микроконтроллеров семейства PIC24F показана на **Рис. 2.1**.

Периферийные модули микроконтроллеров PIC24F позволяют создавать системы обработки данных и управления для широкого класса задач, решаемых в промышленности и в лабораторных исследованиях. Мы будем детально рассматривать принципы функционирования большинства из этих периферийных модулей в последующих главах, а сейчас проведем краткий обзор и начнем с модуля аналого-цифрового преобразователя.

В микроконтроллерах семейства Microchip используется 10-битный аналого-цифровой преобразователь последовательного приближения. Ниже приводятся отдельные характеристики этого преобразователя:

- скорость преобразования — до 500 тыс. выборок/с;
- количество каналов входных аналоговых сигналов — 16;
- источник опорного напряжения — внешний или внутренний.

Аналого-цифровой преобразователь может работать в режиме автоматического сканирования входов и поддерживает различные режимы синхронизации. Модуль АЦП допускает автономную работу при переходе процессора в «спящий» режим или режим «холостого хода». Аналого-цифровой преобразователь может производить несколько последовательных выборок, накапливая

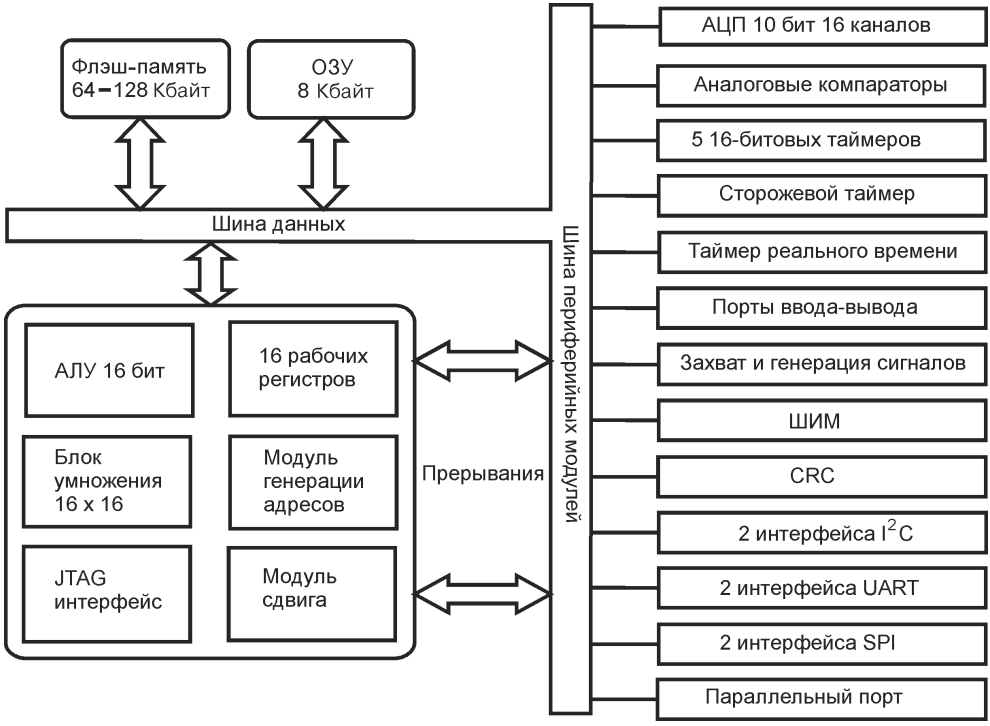

Рис. 2.1. Обобщенная функциональная схема микроконтроллеров PIC24F

результат в 16-уровневом буфере данных, и сохранять результат в одном из четырех форматов.

Следующий периферийный модуль, который мы рассмотрим, — модуль аналоговых компараторов. Это устройство включает в себя два компаратора, которые используются при реализации широкого класса функциональных узлов, например детектора перехода через ноль в схеме синхронизации по переменному току 50 Гц, или при создании более сложных устройств, таких, как 16-битный сигма-дельта аналого-цифровой преобразователь.

Микроконтроллеры PIC24F включают 5 модулей таймеров общего назначения разрядностью 16 бит. Все пять таймеров обладают общими базовыми функциональными возможностями. Регистры периода всех таймеров могут использоваться для генерации прерывания при совпадении содержимого такого регистра с текущим содержимым регистра таймера. Во всех таймерах предусмотрены режим запуска/останова по внешнему сигналу и генерация прерывания по спаду внешнего сигнала. Четыре из пяти таймеров могут объединяться попарно для формирования 32-битных таймеров. С модулями таймеров тесно связан 5-канальный модуль захвата входных сигналов и 5-канальный модуль генерации цифровых сигналов. Модуль захвата входных сигналов используется для измерения интервалов между событиями. Минимальная разре-

шающая способность при таком измерении равна длительности одного машинного цикла. Для синхронизации временны́х меток модуль захвата входных сигналов использует в качестве базовых Таймеры 2 и 3.

Модуль генерации цифровых сигналов используется для генерации одиночных импульсов и импульсных последовательностей и имеет пять каналов. Для отсчета интервалов времени при формировании таких сигналов модуль генерации цифровых сигналов использует в качестве базовых Таймер 2 и Таймер 3. Выходные сигналы этого модуля могут использоваться для управления обычными (термоэлементы) и индуктивными (электродвигатели) нагрузками, а также для синтеза голосовых сигналов.

В микроконтроллерах семейства PIC24F имеется два модуля универсального асинхронного приемопередатчика (UART), которые позволяют реализовать обмен данными в соответствии со стандартами RS-232 и RS-485. Приемопередатчик обеспечивает обмен 8- или 9-битными данными с контролем четности или без такового с одним или двумя стоповыми битами. Устройство поддерживает функцию аппаратного контроля обмена данными, что обеспечивает высокий уровень надежности и производительности, позволяя подключать к микроконтроллеру различные периферийные устройства, например модемы. Скорость передачи данных может изменяться от 15 бод до 1 Мбод. Повышение производительности операций обмена данными и разгрузка процессора обеспечиваются за счет использования 4-уровневых буферов входных/выходных данных и прерываний. Кроме того, модуль UART поддерживает передачу информации по протоколам IrDA и LIN.

Для обмена данными между различными устройствам в настоящее время очень часто используются протоколы I^2C и SPI. Микроконтроллеры PIC24F имеют интегрированные периферийные модули для поддержки этих протоколов. В модуле I^2C применяется аппаратно-программный алгоритм, позволяющий выбрать режим «ведущего» или «ведомого». Модуль поддерживает как 7-, так и 10-битную адресацию устройств на шине I^2C, при этом тактовая частота шины I^2C может быть задана равной 100 или 400 кГц.

Модули SPI микроконтроллеров PIC24F можно сконфигурировать для работы с двумя, тремя или четырьмя сигнальными линиями. В режиме с двумя линиями (синхронизации и данных) интерфейс SPI можно использовать для приема сигналов датчиков. Для работы с аналого-цифровыми преобразователями, сдвиговыми регистрами и микросхемами памяти EEPROM используется, как правило, 3-проводной интерфейс.

Еще один периферийный модуль — модуль часов реального времени с календарем (RTCC) — предназначен для точного отсчета времени в течение длительных интервалов и оперирует с датами и временем. Он может выполнять функцию будильника, включая внешнее устройство в определенный момент времени в будущем. Модуль RTCC синхронизируется непосредственно от внешнего источника тактового сигнала частотой 32 кГц. Из этого сигнала посредством предделителя формируется внутренний сигнал с периодом 0.5 с, который используется для синхронизации регистров модуля, содержащих эле-

менты даты (год, месяц, день; день недели, часы, минуты и секунды). Данные хранятся в регистрах в удобном BCD-формате. Функция будильника может быть запрограммирована на определенный месяц, день; день недели, час, минуту и секунду. Кроме того, поскольку регистры модуля часов реального времени работают на очень низкой частоте, это позволяет минимизировать энергопотребление устройства. Часы реального времени могут функционировать и в том случае, если процессор находится в «спящем» режиме. Для управления внешними устройствами можно использовать выходной сигнал модуля частотой 1 Гц или сигнал, формируемый при срабатывании будильника.

Модуль параллельного порта позволяет легко реализовывать аппаратно-программный 8-битный интерфейс с внешними устройствами и модулями памяти. Модуль поддерживает мультиплексирование шин адреса/данных, позволяя передавать по 8-битной шине 16-битные данные. В модуле предусмотрена работа с 16 адресными линиями, что дает возможность адресовать до 64 Кбайт памяти, а при использовании дополнительных линий адреса, в качестве которых могут быть задействованы выводы портов общего назначения, — и большее пространство адресов. В модуле предусмотрена функция автоинкремента/автодекремента адреса, что позволяет оптимизировать передачу больших блоков данных.

Последнее периферийное устройство, на котором мы остановимся, — модуль контроля достоверности данных с использованием циклического избыточного кода (CRC). Этот модуль находит применение при контроле ошибок обмена данными с периферийным оборудованием и памятью, особенно при работе с коммуникационным оборудованием по протоколам CAN, USB и Ethernet.

Микроконтроллеры семейства PIC24F имеют 16-разрядную шину данных, что существенно повышает их функциональные возможности по сравнению с устройствами семейства PIC18. Длина инструкции (команды) процессора в микроконтроллерах серии PIC24 равна 24 битам, а частота выполнения команд процессора в 2 раза меньше, чем частота синхронизации устройства. Если использовать принятые в документации фирмы Microchip обозначения, то это соотношение можно выразить формулой:

$$F_{CY} = F_{OSC}/2,$$

где F_{OSC} — тактовая частота синхронизации микроконтроллера, а F_{CY} — тактовая частота процессора.

Счетчик команд (Program Counter, PC) имеет разрядность 23 бита и позволяет адресовать до 4 миллионов инструкций в памяти программ. Высокая производительность микроконтроллера обеспечивается за счет использования механизма конвейеризации инструкций, при котором выборка и декодирование следующей инструкции процессора осуществляются на этапе выполнения предыдущей. При этом все инструкции выполняются за один машинный цикл, за исключением инструкций передачи управления, инструкций, оперирующих двойными словами, и инструкций табличного чтения/записи. Кроме того, определенную оптимизацию загрузки процессора можно осуществить за счет применения инструкции цикла `repeat`.

Микроконтроллеры семейства PIC24F имеют 16 рабочих регистров, оперирующих данными размером в одно слово (16 бит). Каждый из этих регистров может выступать в роли регистра данных, адреса или смещения. Шестнадцатый по счету регистр (W15) используется в качестве указателя стека (SP, Stack Pointer) в операциях вызова процедур и обработчиков прерываний. В микроконтроллерах PIC24F предусмотрена возможность отображения верхних 32 Кбайт памяти данных на адресное пространство памяти программ с выравниванием по границе слова, для чего используется 8-битный регистр PSVPAG (Program Space Visibility Page).

Разработчики семейства PIC24F предусмотрели обратную совместимость инструкций и режимов адресации процессора с микроконтроллерами семейства PIC18 за счет прямого включения подмножества инструкций PIC18 в систему команд PIC24, а также посредством использования макросов.

В микроконтроллерах семейства PIC24F реализована гарвардская архитектура, в которой память программ и память данных разделены, что позволяет осуществлять прямой доступ к памяти программ из памяти данных во время выполнения программного кода. Организация памяти программ одной из наиболее распространенных линеек микроконтроллеров семейства PIC24FJ128GAxxx со 128 Кбайт флэш-памяти показана на **Рис. 2.2**.

Пользовательским программам доступна область памяти в диапазоне адресов 0x000000...0x7FFFFF, за исключением области конфигурации устройства, доступ к которой осуществляется посредством инструкций `tblrd/tblwt`. Память программ организована в виде блоков, адресуемых пословно. Хотя адрес памяти является 24-битным, более удобно рассматривать любой адрес в виде младшего и старшего слова, при этом старший байт старшего слова не используется и равен 0. Младшее слово всегда располагается по четному адресу, а старшее — по нечетному. Следует сказать, что адреса памяти программ всегда выравниваются по границе слова и при выполнении программного кода всегда инкрементируются и декрементируются на 2.

Область памяти программ между адресами 0x000000 и 0x000200 зарезервирована для векторов прерываний. По адресам 0x000000 и 0x000002 размещается команда перехода к фактическому началу программы, при этом по первому адресу располагается код операции инструкции `goto`, а по второму — собственно адрес точки входа в программу. Также в памяти программ размещаются две таблицы векторов прерываний. Одна из них располагается в диапазоне адресов 0x000004...0x0000FF, а вторая (альтернативная) — в диапазоне адресов 0x000100...0x0001FF.

Адресное пространство памяти данных микроконтроллеров PIC24F представляет собой непрерывную область 16-битных адресов с линейной адресацией. Адрес для доступа к данным формируется двумя модулями генерации адресов, один из которых используется в операциях записи, а другой — в операциях чтения данных. Распределение памяти данных микроконтроллеров PIC24F[1] показано на **Рис. 2.3**.

---

[1] Моделей с 8 Кбайт ОЗУ. — *Примеч. науч. ред.*

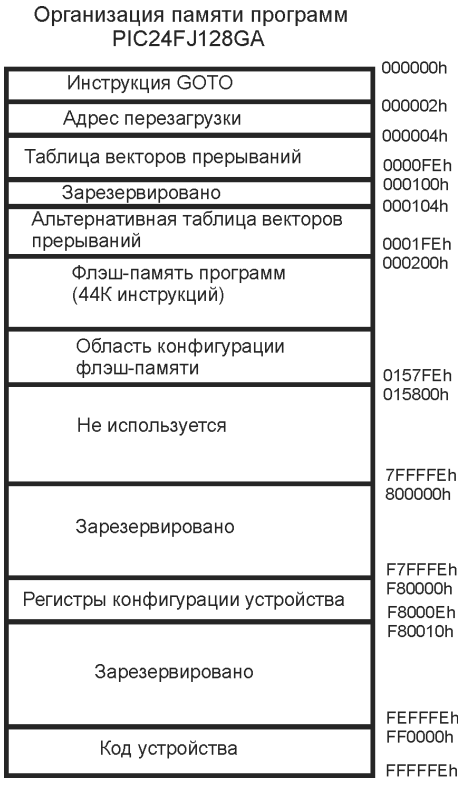

Рис. 2.2. Схема организации памяти программ микроконтроллеров PIC24FJ128GA

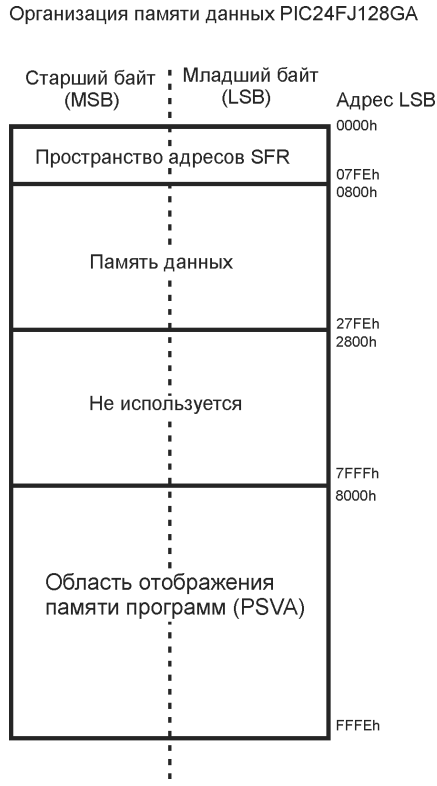

Рис. 2.3. Схема распределения памяти данных

Все действующие адреса памяти данных (Effective Addresses, EA) имеют размер 16 бит, что позволяет адресовать 64 Кбайт памяти. Нижняя половина адресов памяти данных, для которых старший бит действующего адреса равен 0, используется для адресации данных, а старшая половина адресов, для которых старший бит адреса равен 1, — для отображения памяти программ на память данных (Program Space Visibility Area, PSVA). Микроконтроллеры PIC24FJ128GA010 имеют объем памяти данных 8 Кбайт; при обращении к памяти данных по адресу, лежащему за пределами этой области, возвращается нулевой байт или слово.

Данные в памяти выравниваются по границе слова, при этом младшие байты имеют четные, а старшие — нечетные адреса. Для обратной совместимости с 8-битными микроконтроллерами в систему команд PIC24F включены команды, поддерживающие операции как со словами, так и с отдельными байтами. При этом процессор распознает, когда выполняется операция над байтом, а когда – над словом, и генерирует соответствующий адрес. Например, если в команде используется косвенная адресация с пост-инкрементом, такая, как

[W1++], то для байтовой команды генерируется адрес, равный W1+1, а для команды, работающей со словом, — W1+2. При любых операциях со словами данные должны иметь четный адрес, поэтому компилятор будет выдавать ошибку при сборке программ, если в программном коде происходит обращение к слову по нечетному адресу. Если такая ситуация возникает при выполнении программы, то генерируется ошибка адресации и вызывается обработчик этой исключительной ситуации. По этой причине следует соблюдать осторожность, если в программе одновременно присутствуют инструкции, работающие с байтами и со словами.

Для повышения эффективности работы программ в систему команд включен ряд инструкций, позволяющих преобразовывать байты данных в слова. Это инструкция расширения знака `se` (Sign-Extend), приводящая 8-битное число со знаком к 16-битному формату, и инструкция расширения нуля `ze` (Zero-Extend), заполняющая старший байт беззнакового целого нулями. Большинство инструкций процессора могут работать как с байтами, так и со словами, но некоторые инструкции оперируют только со словами.

Первые два килобайта адресного пространства памяти данных (от 0x0000 до 0x7FF) выделяются для регистров специальных функций (Special Function Register, SFR), которые используются ядром и периферийными модулями микроконтроллера PIC24F для выполнения различных операций. Эти регистры распределены между модулями, с которыми они связаны, и, как правило, группируются по принадлежности к тому или иному периферийному модулю. Большинство адресов, выделяемых под регистры специальных функций, на данный момент не задействованы и читаются как 0.

Мы продолжим изучение особенностей аппаратно-программной архитектуры микроконтроллеров PIC24F в следующей главе, в которой подробно рассмотрим особенности программирования этих устройств.

ГЛАВА 3

СИСТЕМА КОМАНД И ОСНОВЫ ПРОГРАММИРОВАНИЯ МИКРОКОНТРОЛЛЕРОВ PIC24F

Микроконтроллеры PIC24 представляют собой качественно новый тип устройств, разработанных фирмой Microchip. Программная архитектура этих микроконтроллеров существенно отличается от предыдущих семейств 8-битных устройств, хотя разработчики сделали все возможное, чтобы обеспечить безболезненный переход от 8-битной архитектуры к 16-битной.

В этой главе рассматривается базовая программная модель микроконтроллеров PIC24F без учета особенностей реализации периферийных модулей, которые детально анализируются в последующих главах.

3.1. Программная модель микроконтроллеров PIC24F

Для разработчика программного обеспечения программную модель микроконтроллеров PIC24F можно представить так, как показано на **Рис. 3.1**.

В качестве рабочих используются 16 регистров процессора (W), которые могут функционировать как регистры данных или адресов. Функция, выполняемая W-регистром, определяется режимом адресации, который используется в данной инструкции.

В регистре W15 содержится указатель стека (Stack Pointer, SP), который автоматически модифицируется при обработке исключений, вызовов подпрограмм и прерываний. Тем не менее, регистр W15 можно использовать в инструкциях в качестве обычного рабочего регистра. Во многих случаях такое использование упрощает манипуляции с указателем стека. Бит 0 этого регистра всегда сброшен в 0 для выравнивания содержимого стека.

При инициализации микроконтроллера в регистр W15 записывается значение 0x0800, и это обеспечивает установку указателя стека на начало области памяти, которая доступна пользователю. Программа пользователя может модифицировать SP так, чтобы он указывал на любую требуемую область памяти. Указатель стека всегда указывает на первое доступное слово и смещается при заполнении от младших адресов к старшим. Указатель стека всегда декремен-

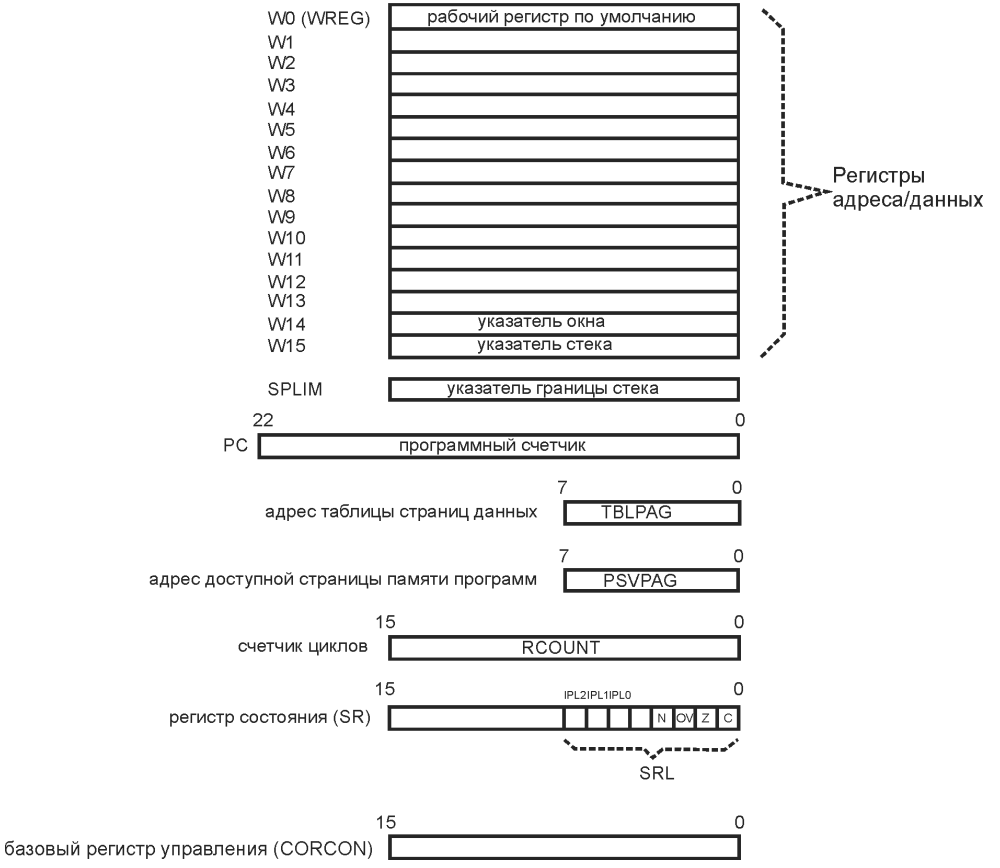

Рис. 3.1. Программная модель микроконтроллера PIC24F

тируется перед извлечением значения из стека и инкрементируется после помещения значения в стек. Функционирование стека показано на **Рис. 3.2**.

Когда содержимое счетчика команд помещается в стек, то биты 15...0 располагаются по первому доступному адресу в стеке, а биты 22...16 помещаются по следующему адресу, при этом старшие биты второго слова дополняются нулями.

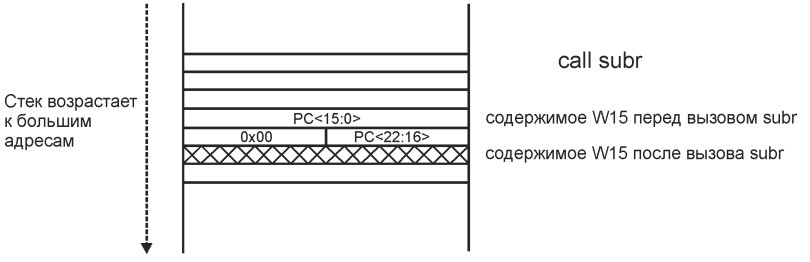

Рис. 3.2. Функционирование стека

Процессор микроконтроллеров PIC24F имеет 16-битный регистр состояния (Status Register, SR). Младшая часть регистра состояния (SRL) содержит флаги арифметических операций (нуля, переноса, переполнения и знака), а также 3 бита приоритета (IPL2...IPL0), в которых устанавливается текущий приоритет прерываний процессора (см. **Рис. 3.1**).

Регистр управления процессором (Core Control Register, CORCON) имеет всего 2 бита, которые можно использовать при разработке программ. Бит 3 (Interrupt Priority Status Level, IPL3) задает уровень приоритета прерываний процессора. Если этот бит установлен, то уровень приоритета прерываний процессора выше 7 — в этом случае пользовательские прерывания однозначно запрещены. Если этот бит сброшен, то уровень приоритета меньше или равен 7 — в этом случае прерывания пользователя могут быть разрешены. Бит 2 (Program Space Visibility, PSV) указывает на возможность доступа к памяти программ из памяти данных: если бит установлен, то заданная область памяти программ доступна из памяти данных, а если сброшен — недоступна.

К программно-доступным регистрам ядра процессора микроконтроллеров PIC24F относятся еще два регистра, которые выполняют специальные функции и могут быть полезны при работе с блоками памяти и таблицами. Регистр TBLPAG используется для хранения старших 8 битов адреса памяти программ при операциях чтения/записи табличных данных. С помощью табличных инструкций процессора можно перемещать данные из памяти программ в память данных. Регистр PSVPAG (Program Memory Visibility Page Address Pointer) позволяет пользователю отобразить страницу памяти программ размером 2 Кбайт на верхние 32 Кбайт адресного пространства памяти данных. С помощью этого регистра можно получить доступ к секции констант программы.

3.2. Режимы адресации и система команд

Систему инструкций (команд) микроконтроллеров PIC24F можно разделить на несколько групп:

- команды перемещения;
- команды математических операций;
- команды логических операций;
- команды сдвига и циклического сдвига;
- команды работы с битами;
- команды сравнения/выбора;
- команды условных переходов;
- команды работы со стеком;
- команды управления.

Мы не будем подробно описывать функционирование каждой команды, а остановимся лишь на общих принципах реализации команд и используемых ими методах адресации. В этом разделе мы рассмотрим многочисленные примеры использования команд микроконтроллеров семейства PIC24F с применением различных методов адресации. Инструкции процессора в PIC24F могут быть ре-

гистровыми и адресными. Основное различие между этими типами инструкций состоит в том, что регистровые инструкции модифицируют содержимое регистров[1], в то время как адресные инструкции используют содержимое рабочих регистров для доступа к данным, находящимся в памяти. В свою очередь адресные инструкции могут быть как с прямой регистровой адресацией (register direct addressing), так и с косвенной регистровой адресацией (register indirect addressing).

Прямая регистровая адресация используется для доступа к содержимому 16 рабочих регистров W0...W15, причем обращение возможно как к целому слову (16 бит), так и к байту.

Косвенная регистровая адресация применяется для доступа к любой ячейке памяти данных посредством формирования исполнительного адреса данных по содержимому одного или нескольких рабочих регистров. Таким образом, содержимое рабочего регистра или регистров является указателем на область памяти данных. Дополнительные возможности такого метода адресации обеспечиваются механизмами пред- и пост-инкремента содержимого регистра, что позволяет последовательно обрабатывать непрерывный диапазон адресов в памяти данных.

Методы формирования эффективного адреса (EA) с помощью косвенной регистровой адресации показаны в **Табл. 3.1**.

Таблица 3.1. Формирование эффективного адреса методом косвенной регистровой адресации

Режим адресации	Байтовая адресация	Адресация слова	Описание
Косвенная без модификации	EA = [Wn]	EA = [Wn]	Эффективный адрес формируется по содержимому регистра Wn
Косвенная с пост-инкрементом	EA = [Wn]+=1	EA = [Wn]+=2	Эффективный адрес формируется по содержимому регистра Wn, затем содержимое Wn инкрементируется
Косвенная с пост-декрементом	EA = [Wn]–=1	EA = [Wn]–=2	Эффективный адрес формируется по содержимому регистра Wn, затем содержимое Wn декрементируется
Косвенная с пред-инкрементом	EA = [Wn+=1]	EA = [Wn+=2]	Перед формированием эффективного адреса содержимое Wn инкрементируется
Косвенная с пред-декрементом	EA = [Wn–=1]	EA = [Wn–=2]	Перед формированием эффективного адреса содержимое Wn декрементируется
Косвенная со смещением, заданным в регистре	EA = [Wn + Wb]	EA = [Wn + Wb]	Эффективный адрес формируется как сумма Wn и Wb
Косвенная со смещением, заданным константой	EA = [Wn + + Slit5]	EA = [Wn + + 2*Slit5]	Эффективный адрес формируется как сумма Wn и константы Slit5 (значение в диапазоне от –16 до +15)

[1] Точнее, области памяти в диапазоне адресов 0x0000...0x1FFF (команда mov — в диапазоне адресов 0x0000...0xFFFE). – *Примеч. науч. ред.*

Регистр W0 является особым регистром, поскольку это единственный из рабочих регистров, который можно использовать в регистровых инструкциях. Регистровые инструкции оперируют с адресами памяти и содержимым регистра W0. Рабочие регистры W1...W15 в таких инструкциях в качестве приемника использовать нельзя. Регистровые инструкции обратно совместимы с инструкциями, которые входят в систему команд микроконтроллеров предыдущих семейств и работают только с одним W-регистром. Синтаксис ассемблера включает обозначение «WREG», призванное подчеркнуть, что регистр W0 используется в регистровой инструкции. Ниже приводятся примеры регистровых инструкций:

```
mov    WREG,0x0100
add    0x0100,WREG
```

Первая инструкция помещает содержимое регистра W0 по адресу 0x0100, а вторая складывает значение по адресу 0x0100 с содержимым регистра WREG и помещает результат в WREG.

Все рабочие регистры отображены на адресное пространство памяти данных, поэтому их можно использовать в таких, например, инструкциях:

```
mov    0x0004,W10
```

Здесь 0x0004 — адрес регистра W2. Эта инструкция эквивалентна следующей:

```
mov    W2,W10
```

Для более глубокого изучения различных режимов адресации и функционирования команд микроконтроллеров PIC24F мы рассмотрим многочисленные примеры программного кода, написанного на языке ассемблера MPLAB ASM30. В качестве целевого устройства будем использовать микроконтроллер PIC24FJ128GA010, хотя программы будут работать и с другими моделями семейства PIC24F.

Изучение функционирования микроконтроллера на уровне команд ассемблера обязательно, если вы намерены стать высококвалифицированным разработчиком программного обеспечения. Даже если в дальнейшем вам придется разрабатывать программы на одном из языков высокого уровня (Си, Бейсик или Паскаль), знание языка ассемблера и программной архитектуры окажет неоценимую помощь при отладке программ. Зачастую, как показывает практика, без знания ассемблера бывает крайне сложно отладить программу, особенно сложную, до требуемого уровня функциональности. Очень часто знание ассемблера помогает решить те или иные проблемы, которые на языке высокого уровня решить довольно трудно (особенно это касается работы с внешними устройствами и интерфейсами). Разумное сочетание языка высокого уровня и ассемблера позволяет решить любую задачу в пределах аппаратных возможностей самого устройства.

Компиляцию тестовых программ на ассемблере мы будем проводить в среде MPLAB IDE, а отладку проводить с помощью программного симулятора MPLAB SIM. Все тестовые программы на ассемблере разработаны с использованием шаблона, который показан далее:

Конец ознакомительного фрагмента.
Приобрести книгу можно
в интернет-магазине «Электронный универс»
(e-Univers.ru)