

Программирование микроконтроллеров PIC

Автор курса: к.т.н. Драч В.Е. (доц. ЭИУ1-КФ)

Раздел: Ассемблер для микроконтроллеров

Микроконтроллер и ПЛИС. Сравнение

1. ШИМ - Широтно-импульсная модуляция

Широтно-импульсная модуляция (ШИМ или PWM, Pulse-width modulation) — приближение желаемого сигнала (многоуровневого или непрерывного) действительным бинарным (с двумя уровнями - вкл/выкл), так что в среднем, за отрезок времени, их значения равны. Формально, это можно записать так:

$$\int_{t_1}^{t_2} x(t) dt = \sum A * \Delta T_i$$

где $x(t)$ - желаемый входной сигнал в пределе от 0 до A , а ΔT_i - продолжительность i -го ШИМ импульса, каждый с амплитудой A . ΔT_i подбирается таким образом, что суммарные площади (энергии) обеих величин приблизительно равны за достаточно продолжительный промежуток времени, равны так же и средние значения величин за период:

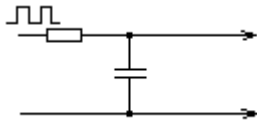
$$\frac{\int_{t_1}^{t_2} x(t) dt}{t_2 - t_1} = \frac{\sum A * \Delta T_i}{t_2 - t_1}$$

Управляемым уровнем как правило является питание силовой установки, например напряжение импульсных преобразователей/регуляторов постоянного напряжения или скорость электродвигателя. Для импульсных источников $x(t) = U_{const}$ стабилизации.

ШИП — широтно-импульсный преобразователь, генерирующий ШИМ-сигнал по заданному значению управляющего напряжения. Основное достоинство ШИП — высокий КПД его усилителей мощности, который достигается за счёт использования их исключительно в ключевом режиме. Это значительно уменьшает выделение мощности на силовом преобразователе (СП).

Техническая реализация ШИМ: при фиксированном периоде меняется длительность импульса пропорционально амплитуде сигнала. Период должен быть меньше периода кодируемого сигнала. Для восстановления сигнала используется интегратор.

Схема для преобразования последовательности импульсов в аналоговый сигнал, показана на рисунке. Т.о., изменяя частоту сигналов, можно добиться изменения амплитуды на выходе. Данный принцип широко используется в системах управления.



В цифровой технике, выходы которой могут принимать только одно из двух значений, приближение желаемого среднего уровня выхода при помощи ШИМ является совершенно естественным. Схема настолько же проста: пилообразный сигнал генерируется N -битным счётчиком. Цифровые устройства (ЦШИП) работают на фиксированной частоте, обычно намного превышающей реакцию управляемых установок (передискретизация). В периоды между фронтами тактовых импульсов, выход ЦШИП остаётся стабильным, на нём действует либо низкий уровень, либо высокий, в зависимости от выхода цифрового компаратора, сравнивающего значение счётчика с уровнем приближаемого цифрового сигнала $V(n)$. Выход за много тактов можно трактовать как череду импульсов с двумя возможными значениями 0 и 1, сменяющимися друг-друга каждый такт T . Частота появления единичных импульсов получается пропорциональной уровню приближаемого сигнала $\sim V(n)$. Единицы, следующие одна за другой, формируют контур одного, более широкого импульса. Длительности полученных импульсов переменной ширины $\sim V(n)$, кратны периоду тактирования T , а частота равна $1/(T*2N)$. Низкая частота означает длительные, относительно T , периоды постоянства сигнала одного уровня, что даёт невысокую равномерность распределения импульсов.

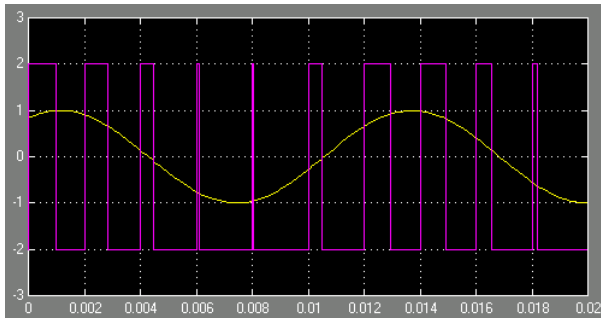


Рис. 1.

Момент возникновения импульса и высокого уровня, и низкого - принято называть фронтом импульса, а момент окончания - спадом импульса. Ясно, что для импульса высокого уровня фронтом служит положительный (или плюсовой) перепад напряжения - от низкого уровня к высокому, а спадом - отрицательный (минусовой) перепад напряжения, когда уровень меняется с высокого на низкий. Понятно также, что фронт импульса высокого уровня является спадом импульса низкого уровня и наоборот.

Рассмотрим различные схемотехнические реализации генератора электрических импульсов.

Порядок изложения – хронологический.

2. Мультивибраторы

Под мультивибратором понимают генератор импульсов с резисторно-емкостной связью, охваченный положительной обратной связью (от латинских слов, multum — много, vibro — колеблю). Широко распространенная схема симметричного мультивибратора на двух транзисторах приведена на рис. 2.

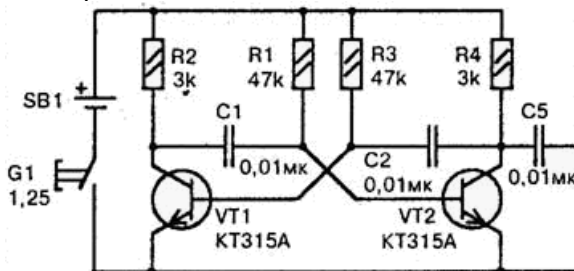


Рис.2

Желаемая частота мультивибратора устанавливается изменением величин резисторов R1 и R3 и конденсаторов C1...C2. Частоту генератора в зависимости от указанных резисторов и конденсаторов можно определить по формуле:

$$f \text{ (Гц)} = 1 / (1,4 \times R \text{ (МОм)} \times C \text{ (мкФ)}).$$

При значениях резисторов и конденсаторов, указанных на схеме, частота первой (основной) гармоники составляет 1000 Гц.

Плюсы:

- 1) Не требует навыков
- 2) Не требует спец. оснастки для изготовления

Минусы:

- 1) Большое количество ЭРЭ
- 2) Низкая высокая надежность
- 3) На работу влияют: разброс параметров микросхем, нестабильность напряжения питания, изменения температуры. Может потребоваться использование дополнительных R.

4) Данное схемотехническое решение обеспечивает реализацию только 1 функции (генерация).

Область применения:

- 1) Любительские схемы
- 2) Ширпотреб

3. Мультивибратор на логических элементах

Автоколебательный мультивибратор может быть построен на логических элементах К155ЛА3. Мультивибратор генерирует импульсы с частотой следования около 0,5 Гц.

Схема одного из вариантов автоколебательного мультивибратора показана на рис. 3, а. Она должна напомнить схему симметричного мультивибратора на двух транзисторах.

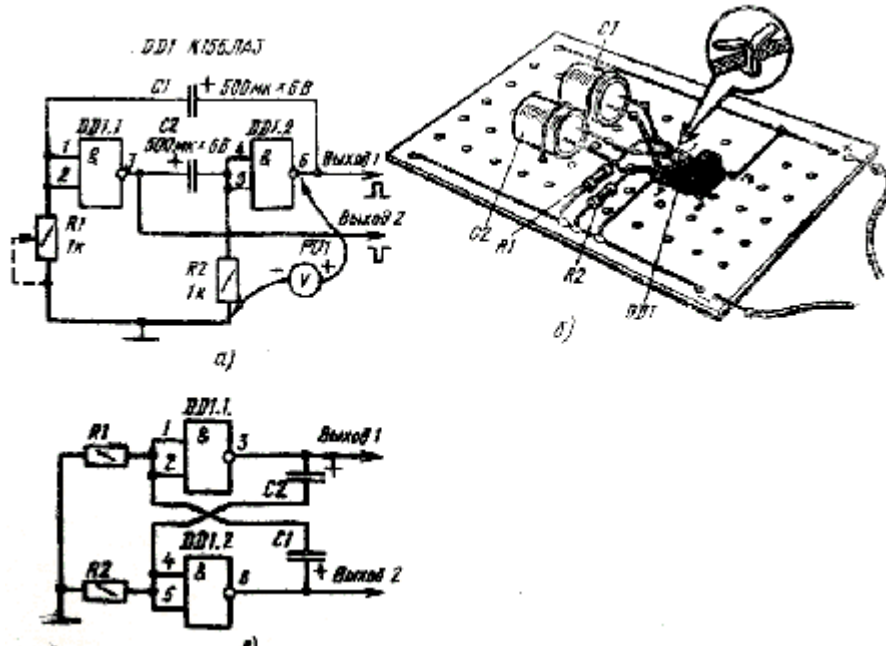


Рис. 3. Автоколебательный мультивибратор

Но здесь функцию активных элементов мультивибратора выполняют логические элементы 2И-НЕ, включенные инверторами. Благодаря двум цепям положительной обратной связи - выхода элемента DD1.2 через конденсатор C1 со входом DD1.1 и выхода элемента DD1.1 через конденсатор C2 со входом DD1.2 - устройство самовозбуждается и генерирует электрические импульсы. Частота следования генерируемых импульсов зависит от номиналов указанных конденсаторов и резисторов R1 и R2.

Те отрезки этой последовательности, когда напряжение принимает высокий уровень, называют импульсами высокого уровня; паузами между ними считают отрезки с низким уровнем напряжения. Однако с тем же основанием можно говорить и об импульсах низкого уровня; в этом случае паузы будут иметь высокий уровень.

В общем случае длительность импульсов может не быть равной длительности пауз между ними. Соотношение этих длительностей оценивают таким параметром, как СКВАЖНОСТЬ, показывающим, во сколько раз период больше длительности импульса.

Увеличение C1 и C2 приведет уменьшению частоты (и наоборот).

Можно изменять емкость лишь одного конденсатора, например C1. Это изменит не только частоту, но и соотношение длительности импульсов и пауз между ними, однако по схемному построению мультивибратор останется симметричным.

Конденсаторы могут быть емкостью по 1...5 мкф. Тогда частота генерируемых импульсов возрастет примерно до 500...1000 Гц (звуковой частоты).

Не исключено, что мультивибратор работает неустойчиво, не всегда возбуждается после замены деталей. Резистор на входе логического элемента, образующего одно из плеч мультивибратора, оказывается включенным в эмиттерную цепь входного транзистора элемента микросхемы. Ток эмиттера создает на этом резисторе падение напряжения, закрывающее транзистор. При сравнительно большом сопротивлении резистора (более 2,2...2,6 кОм) падение напряжения на нем оказывается столь значительным, что транзистор практически не реагирует на входной сигнал. И наоборот, при малом сопротивлении резистора (не более 600...700 Ом) входной транзистор элемента все время открыт к насыщению и, следовательно, оказывается управляемым входными сигналами.

Таким образом, для надежной работы мультивибратора такого варианта сопротивления входных резисторов логических элементов **должны быть** в пределах 800 Ом.. 2,2 кОм. Соответствующим выбором этих резисторов можно добиться устойчивой работы мультивибратора. Кроме того, нужно помнить, что на работу мультивибратора влияют **разброс** параметров микросхем, **нестабильность напряжения источника питания, значительные изменения температуры** окружающей среды.

Более стабилен в работе мультивибратор на трех логических элементах без резисторов в их входной цепи, собранный, например, по схеме на рис. 4, а. Все элементы включены инверторами и соединены между собой последовательно. Временязадающую цепь, определяющую частоту генерации, образуют конденсатор C1 и резистор R1. Частота следования импульсов мультивибратора равна 1 Гц.

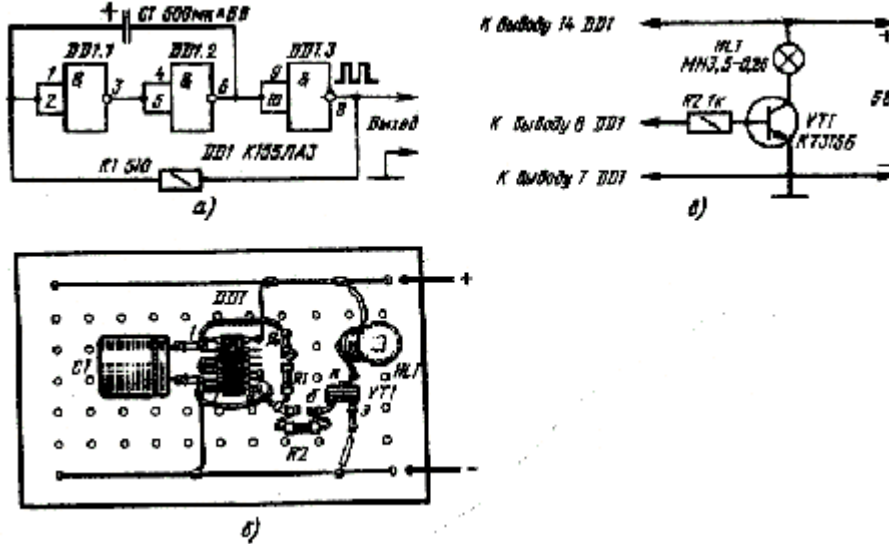


Рис. 4 Мультивибратор на трех логических элементах
Увеличение C1 (уменьшение R) приводит к уменьшению частоты импульсов.

Теперь, разобравшись в работе трехэлементного мультивибратора, исключим из него элемент DD1.3 и переключим правый (по схеме) вывод резистора на выход первого элемента, как показано на рис. 5. Мультивибратор стал двухэлементным. Подключив к его выходу световой индикатор, вы убедитесь, что частота генерируемых импульсов осталась прежней - 1 Гц. Как и в предыдущих вариантах мультивибратора, она будет изменяться при установке деталей других номиналов.

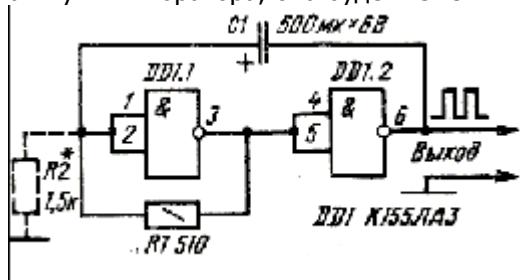


Рис 5. Двухэлементный мультивибратор

R2 создает на входе элемента постоянное напряжение, близкое к пороговому, что облегчает запуск и условия работы мультивибратора в целом.

Такие варианты мультивибратора широко используют в цифровой технике для генерирования импульсов различной частоты и длительности.

Плюсы:

- 1) Простота схемной реализации по сравнению с транзисторным вариантом
- 2) Более высокая надежность

Минусы:

- 1) На работу влияют: разброс параметров микросхем, нестабильность напряжения питания, изменения температуры. Может потребоваться использование дополнительных R.

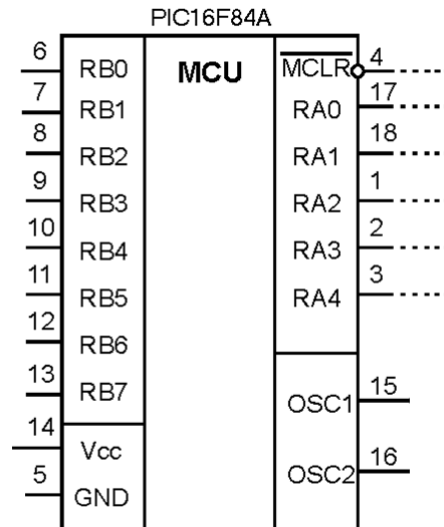
2) Данное схмотехническое решение обеспечивает реализацию только 1 функции (генерация).

Область применения:

- 1) Любительские схемы
- 2) Бытовая техника
- 3) Изделия специального назначения

4. Генератор импульсов на микроконтроллере

Схмотехническая реализация – стандартное включение микроконтроллера. Микроконтроллер будет работать в соответствии с программой, заложенной в него.



Кварцевый генератор стабилизирует тактовую частоту. Питание от 3 до 12 В. Причем колебания напряжения не влияют на режимы функционирования.

Плюсы:

- 1) Простота схемной реализации
- 2) Высокая надежность
- 3) Не требуется изменения схемы для изменения Программное изменение параметров функционирования (частоты)

Самый важный плюс: одно и то же схмотехническое решение для реализации различных устройств (генератор, драйвер шагового двигателя, блок управления холодильником ...). Меняется только периферия.

Минусы:

- 1) Высокая цена (?)
- 2) Наличие техоснастки (программатора)
- 3) Требуется наличие навыков программирования

5. Обзор микроконтроллеров подгруппы PIC16F8X

Основные характеристики

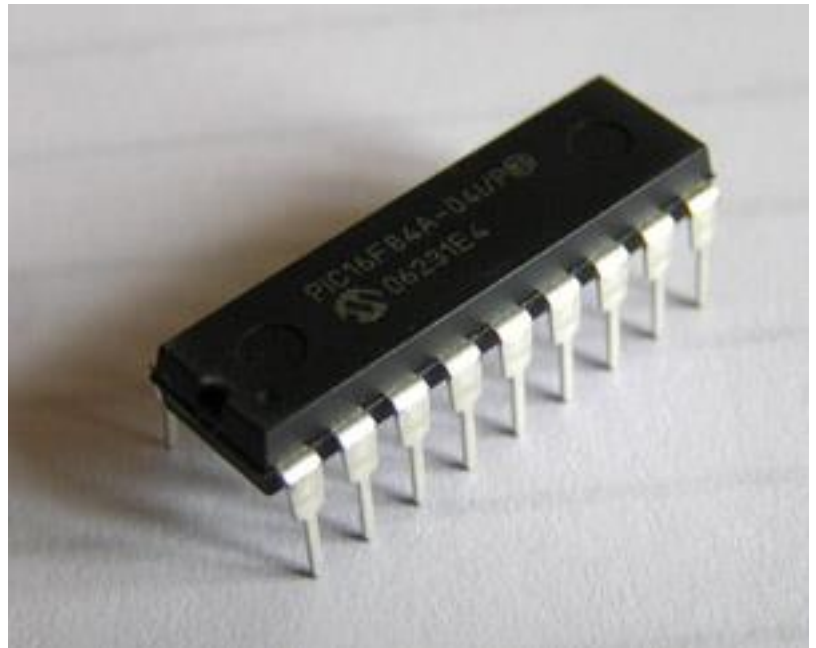
Микроконтроллеры подгруппы PIC16F8X относятся к семейству 8-разрядных КМОП микроконтроллеров группы PIC16CXXX, для которых характерны низкая стоимость, полностью статическая КМОП-технология и высокая производительность.

Конструктивно обычно выполнены в типовом корпусе DIP18. Ниже приведена фотография микроконтроллера PIC16F84.

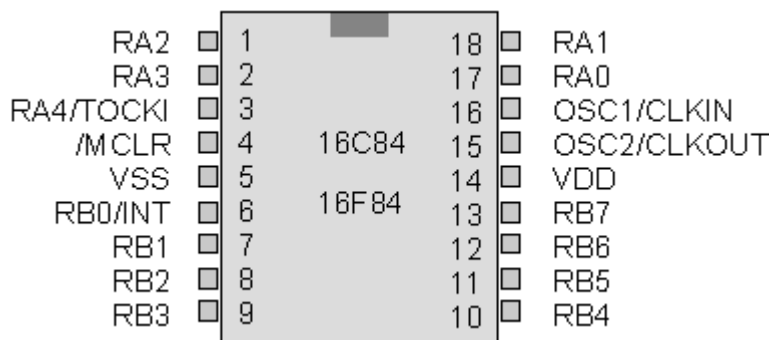
Вид сверху:



(обратите внимание на «ключ» - на фотографиях он слева)



Цоколевка микросхемы представлена на следующем рисунке:



В состав подгруппы входят МК PIC16F83, PIC16CR83, PIC16F84 и PIC16CR84. Основные характеристики МК подгруппы PIC16F8X приведены в табл. 5.1.

Все микроконтроллеры подгруппы PIC16F8X используют гарвардскую архитектуру с RISC-процессором, обладающую следующими основными особенностями:

- используются только 35 простых команд;
- все команды выполняются за один цикл (400 нс при частоте 10 МГц), кроме команд перехода, которые требуют 2 цикла;
- рабочая частота 0 Гц ... 10 МГц;
- отдельные шины данных (8 бит) и команд (14 бит);
- 512 x 14 или 1024 x 14 память программ, выполненная на ПЗУ или электрически перепрограммируемой Flash-памяти;
- 15 восьмиразрядных *регистров* специальных функций (SFR);
- восьмиуровневый аппаратный стек;
- прямая, косвенная и относительная *адресация* данных и команд;
- 36 или 68 восьмиразрядных *регистров* общего назначения (GPR) или ОЗУ;
- четыре источника *прерывания*:
 - внешний вход **RB0/INT**;
 - переполнение *таймера* **TMR0**;
 - изменение сигналов на линиях *порта* **B**;
 - завершение записи данных в память **EEPROM**;
- 64 x 8 электрически перепрограммируемая **EEPROM** память данных с возможностью выполнения 1000000 циклов стирания/записи;

- сохранение данных в **EEPROM** в течение как минимум 40 лет.

Таблица 5.1. Основные характеристики МК подгруппы PIC16F8X.

Параметр	PIC16F83	PIC16CR83	PIC16F84	PIC16CR84
Максимальная частота, МГц	10	10	10	10
Flash-память программ, слов	512	-	1K	-
ПЗУ программ, слов	-	512	-	1K
Память данных, байт	36	36	68	68
Память данных в РПЗУ (EEPROM), байт	64	64	64	64
<i>Таймеры</i>	TMR0	TMR0	TMR0	TMR0
Число источников прерываний	4	4	4	4
Число линий ввода/вывода	13	13	13	13
Диапазон напряжений питания, В	2,0 – 6,0	2,0 – 6,0	2,0 – 6,0	2,0 – 6,0
Число выводов и тип корпуса	18 DIP, SOIC	18 DIP, SOIC	18 DIP, SOIC	18 DIP, SOIC

Микроконтроллеры подгруппы PIC16F8X обладают развитыми возможностями ввода/вывода:

- 13 линий ввода-вывода с индивидуальной установкой направления обмена;
- высокий втекающий/вытекающий ток, достаточный для управления светодиодами;
- максимальный втекающий ток – 25 мА;
- максимальный вытекающий ток – 20 мА;
- 8-битный таймер/счетчик **TMR0** с 8-битным программируемым предварительным делителем.

Специализированные микроконтроллерные функции включают следующие возможности:

- автоматический сброс при включении (Power-on-Reset);
- таймер включения при сбросе (Power-up Timer);
- таймер запуска генератора (Oscillator Start-up Timer);
- сторожевой (Watchdog) таймер **WDT** с собственным встроенным генератором, обеспечивающим повышенную надежность;
- **EEPROM** бит секретности для защиты кода;
- экономичный режим **SLEEP**;
- выбираемые пользователем биты для установки режима возбуждения встроенного генератора;
- последовательное встроенное устройство программирования **Flash/EEPROM** памяти программ и данных с использованием только двух выводов.

КМОП технология обеспечивает МК подгруппы PIC16F8X дополнительные преимущества:

- статический принцип работы;
- широкий диапазон напряжений питания: 2,0 ... 6,0 В;
- низкое энергопотребление:
- менее 2 мА при 5В и 4МГц;
- порядка 15 мкА при 2В и 32КГц;
- менее 1 мкА для **SLEEP**-режима при 2В.

Микроконтроллеры подгруппы PIC16F8X различаются между собой только объемом ОЗУ данных, а также объемом и типом памяти программ. Наличие в составе подгруппы МК с Flash-памятью программ облегчает создание и отладку прототипов промышленных образцов изделий.

Особенности архитектуры

Упрощенная структурная схема МК подгруппы PIC16F8X приведена на рис. 5.1.

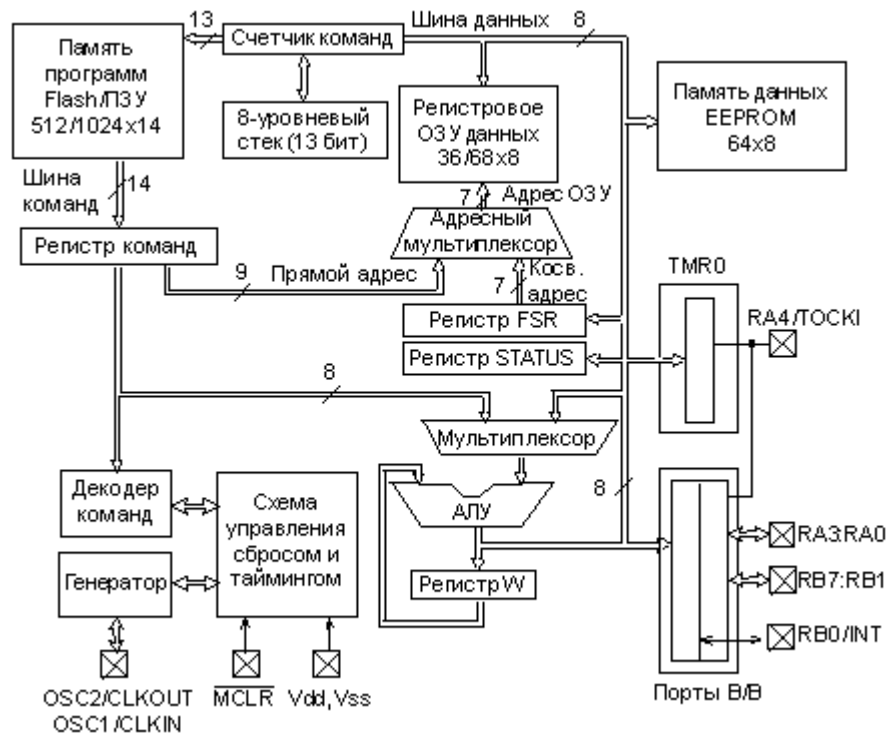


Рис. 5.1. Структурная схема МК подгруппы PIC16F8X.

Архитектура основана на концепции отдельных шин и областей памяти для данных и для команд (гарвардская архитектура). Шина данных и память данных (ОЗУ) – имеют ширину 8 бит, а программная шина и программная память (ПЗУ) имеют ширину 14 бит. Такая концепция обеспечивает простую, но мощную систему команд, разработанную так, что битовые, байтовые и регистровые операции работают с высокой скоростью и с перекрытием по времени выборок команд и циклов выполнения. 14-битовая ширина программной памяти обеспечивает выборку 14-битовой команды в один цикл. Двухступенчатый конвейер обеспечивает одновременную выборку и исполнение команды. Все команды выполняются за один цикл, исключая команды переходов.

Микроконтроллеры PIC16F83 и PIC16CR83 адресуют 512x14 памяти программ, а PIC16F84 и PIC16CR84 – 1Kx14 памяти программ. Вся память программ является внутренней.

Микроконтроллер может прямо или косвенно обращаться к *регистрам* или памяти данных. Все *регистры* специальных функций, включая счетчик команд, отображаются на память данных. Ортогональная (симметричная) система команд позволяет выполнять любую команду над любым *регистром* с использованием произвольного метода *адресации*. Ортогональная архитектура и отсутствие специальных исключений делает программирование МК группы PIC16F8X простым и эффективным.

Назначение выводов МК подгруппы PIC16F8X приведено в табл. 5.2.

Таблица 5.2. Назначение выводов МК подгруппы PIC16F8X.

Обозначение	Тип	Буфер	Описание
OSC1/CLKIN	I	ТШ/КМОП ¹⁾	Вход кристалла генератора, RC-цепочки или вход внешнего тактового сигнала
OSC2/CLKOUT	O	-	Выход кристалла генератора. В RC-режиме – выход 1/4 частоты OSC1
/MCLR	I/P	ТШ	Сигнал сброса/вход программирующего напряжения. Сброс низким уровнем.
RA0	I/O	ТТЛ	PORTA – двунаправленный порт ввода/вывода
RA1	I/O	ТТЛ	

RA2	I/O	ТТЛ	RA4/T0CKI может быть выбран как тактовый вход таймера/счетчика TMR0. Выход с открытым стоком.
RA3	I/O	ТТЛ	
RA4	I/O	ТШ	
/T0CKI			
RB0/INT	I/O	ТТЛ/ТШ ²⁾	PORTB – двунаправленный порт ввода/вывода. Может быть запрограммирован в режиме внутренних активных нагрузок на линию питания по всем выводам. Вывод RB0/INT может быть выбран как внешний вход прерывания. Выводы RB4...RB7 могут быть программно настроены как входы прерывания по изменению состояния на любом из входов. При программировании МК RB6 используется как тактовый, а RB7 как вход/выход данных.
RB1	I/O	ТТЛ	
RB2	I/O	ТТЛ	
RB3	I/O	ТТЛ	
RB4	I/O	ТТЛ	
RB5	I/O	ТТЛ	
RB6	I/O	ТТЛ/ТШ ³⁾	
RB7	I/O	ТТЛ/ТШ ⁴⁾	

Таблица 5.2. Назначение выводов МК подгруппы PIC16F8X (продолжение).

Обозначение	Тип	Буфер	Описание
Vdd	P	-	Положительное напряжение питания
Vss	P	-	Общий провод (земля)

В таблице использованы следующие обозначения: I — вход; O — выход; I/O — вход/выход; P — питание; – — не используется; ТТЛ — ТТЛ вход; ТШ — вход триггера Шмитта.

Микроконтроллер содержит 8-разрядное АЛУ и рабочий регистр W. АЛУ является арифметическим модулем общего назначения и выполняет арифметические и логические функции над содержимым рабочего регистра и любого из регистров контроллера. АЛУ может выполнять операции сложения, вычитания, сдвига и логические операции. Если не указано иное, то арифметические операции выполняются в дополнительном двоичном коде.

В зависимости от результата операции, АЛУ может изменять значения бит регистра STATUS: C (Carry), DC (Digit carry) и Z (Zero).

Схема тактирования и цикл выполнения команды

Входная тактовая частота, поступающая с вывода OSC1/CLKIN, делится внутри на четыре, и из нее формируются четыре циклические не перекрывающиеся тактовые последовательности Q1, Q2, Q3 и Q4. Счетчик команд увеличивается в такте Q1, команда считывается из памяти программы и защелкивается в регистре команд в такте Q4. Команда декодируется и выполняется в течение последующего цикла в тактах Q1...Q4.

Организация памяти программ и стека

Счетчик команд в МК PIC16F8X имеет ширину 13 бит и способен адресовать 8Кх14бит объема программной памяти. Однако физически на кристаллах PIC16F83 и PIC16CR83 имеется только 512х14 памяти (адреса 0000h-01FFh), а в МК PIC16F84 и PIC16CR84 – 1Кх14 памяти (адреса 0000h-03FFh). Обращение к адресам выше 1FFh (3FFh) фактически есть адресация в те же первые 512 адресов (первые 1К адресов).

Организация памяти программ и стека приведена на рис. 5.4.

В памяти программ есть выделенные адреса. Вектор сброса находится по адресу 0000h, вектор прерывания – по адресу 0004h. Обычно по адресу 0004h располагается подпрограмма идентификации и обработки прерываний, а по адресу 0000h – команда перехода на метку, расположенную за подпрограммой обработки прерываний.

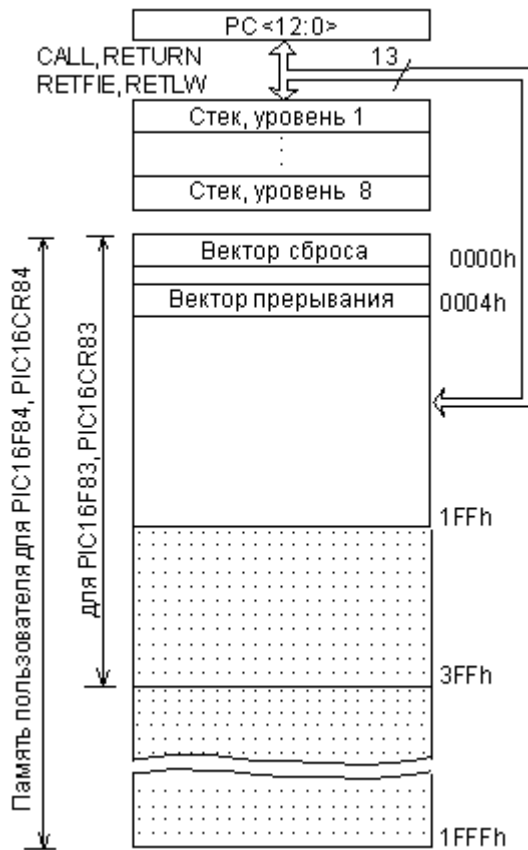


Рис. 5.4. Организация памяти программ и стека.

Организация памяти данных

Память данных МК разбита на две области. Первые 12 адресов – это область *регистров* специальных функций (SFR), а вторая – область *регистров* общего назначения (GPR). Область SFR управляет работой прибора.

Обе области разбиты в свою очередь на банки 0 и 1. Банк 0 выбирается обнулением бита **RPO** регистра статуса (**STATUS**). Установка бита **RPO** в единицу выбирает банк 1. Каждый банк имеет протяженность 128 байт. Однако для PIC16F83 и PIC16CR83 память данных существует только до адреса **02Fh**, а для PIC16F84 и PIC16CR84 – до адреса **04Fh**.

На рис. 5.5 изображена организация памяти данных.

Некоторые *регистры* специального назначения продублированы в обоих банках, а некоторые расположены в банке 1 отдельно.

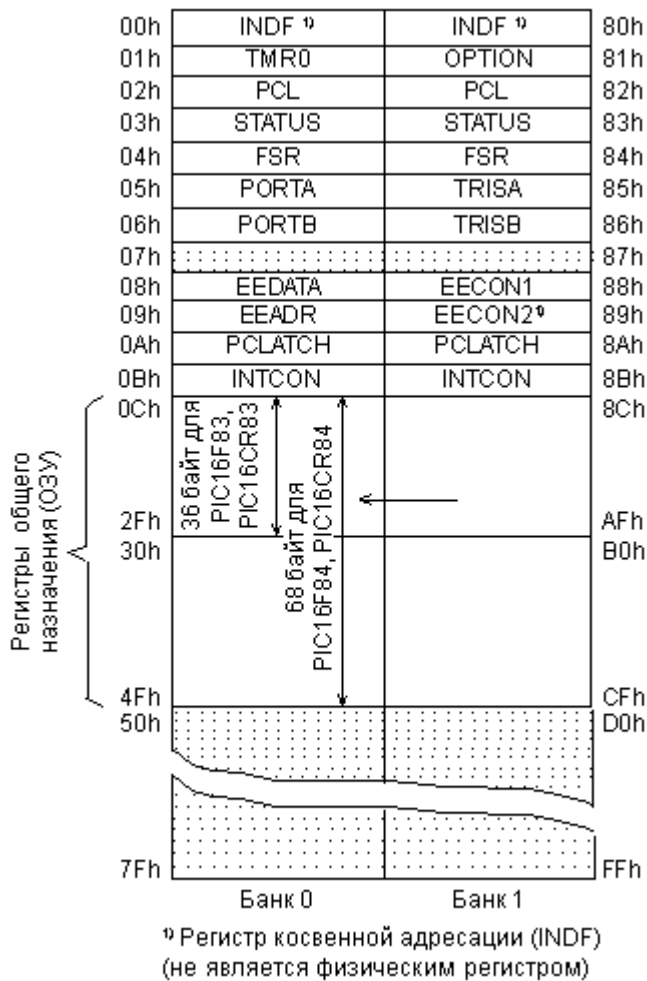


Рис. 5.5. Организация памяти данных.

Регистры с адресами **0Ch-4Fh** могут использоваться как *регистры* общего назначения, которые представляют собой статическое ОЗУ. Адреса *регистров* общего назначения банка 1 отображаются на банк 0. Следовательно, когда установлен банк 1, то обращение к адресам **8Ch-CFh** фактически адресует банк 0.

В *регистре* статуса помимо бита **RP0** есть еще бит **RB1**, что позволяет обращаться к четырем страницам (банкам) будущих модификаций этого кристалла.

К ячейкам ОЗУ можно адресоваться прямо, используя абсолютный адрес каждого *регистра*, или косвенно, через *регистр* указатель **FSR**. Косвенная *адресация* использует текущее значение разрядов **RP1:RP0** для доступа к банкам. Это относится и к **EEPROM** памяти данных. В обоих случаях можно адресовать до 512 *регистров*.

Регистры специального назначения

Регистр статуса (**STATUS**) содержит признаки операции (арифметические флаги) АЛУ, состояние контроллера при сбросе и биты выбора страниц для памяти данных. Назначение бит *регистра* приведено в табл. 5.3.

Регистр статуса доступен для любой команды так же, как любой другой *регистр*. Однако если *регистр* **STATUS** является *регистром* назначения для команды, влияющей на биты **Z**, **DC** или **C**, то запись в эти три бита запрещается. Кроме того, биты **/TO** и **/PD** устанавливаются аппаратно и не могут быть записаны в статус программно. Это следует иметь в виду при выполнении команды с использованием *регистра* статуса. Например, команда **CLRF STATUS** обнулит все биты, кроме битов **/TO** и **/PD**, а затем установит бит **Z=1**. После выполнения этой команды *регистр* статуса может и не иметь нулевого значения (из-за битов **/TO** и **/PD**) **STATUS=000uu1uu**, где **u** – неизменяемое

состояние. Поэтому рекомендуется для изменения *регистра* статуса использовать только команды битовой установки **BCF**, **BSF**, **MOVWF**, которые не изменяют остальные биты статуса. Воздействие всех команд на биты статуса рассматривается в разделе "Описание системы команд".

Регистр конфигурации (**OPTION**) является доступным по чтению и записи *регистром*, который содержит управляющие биты для конфигурации предварительного делителя (делителя), внешних *прерываний*, *таймера*, а также резисторов "pull-up" на выводах **PORTB**. Назначение бит *регистра* приведено в табл. 5.4.

В том случае, когда предделитель обслуживает сторожевой *таймер* **WDT**, *таймеру* **TMR0** назначается коэффициент предварительного деления 1:1.

Регистр условий *прерывания* (**INTCON**) является доступным по чтению и записи *регистром*, который содержит биты доступа для всех источников *прерываний*. Назначение бит *регистра* приведено в табл. 5.5.

Бит разрешения всех *прерываний* **GIE** сбрасывается автоматически при следующих обстоятельствах:

- по включению питания;
- по внешнему сигналу **/MCLR** при нормальной работе;
- по внешнему сигналу **/MCLR** в режиме **SLEEP**;
- по окончанию задержки *таймера* **WDT** при нормальной работе;
- по окончанию задержки *таймера* **WDT** в режиме **SLEEP**.

Прерывание **INT** может вывести процессор из режима **SLEEP**, если перед входом в этот режим бит **INTE** был установлен в единицу. Состояние бита **GIE** также определяет: будет ли процессор переходить на подпрограмму *прерывания* после выхода из режима **SLEEP**.

Сброс битов – запросов *прерываний* – должен осуществляться соответствующей программой обработки.

Счетчик команд

Счетчик команд **PCL** и **PCLATH** имеет разрядность 13 бит. Младший байт счетчика (**PCL**) доступен для чтения и записи и находится в *регистре* **02h**. Старший байт счетчика команд не может быть напрямую записан или считан и берется из *регистра* **PCLATH** (PC latch high), адрес которого **0Ah**. Содержимое **PCLATH** передается в старший байт счетчика команд, когда он загружается новым значением.

В зависимости от того, загружается ли в счетчик команд новое значение во время выполнения команд **CALL**, **GOTO**, или в младший байт счетчика команд (**PCL**) производится запись, – старшие биты счетчика команд загружаются из **PCLATH** разными способами, как показано на рис. 5.6.

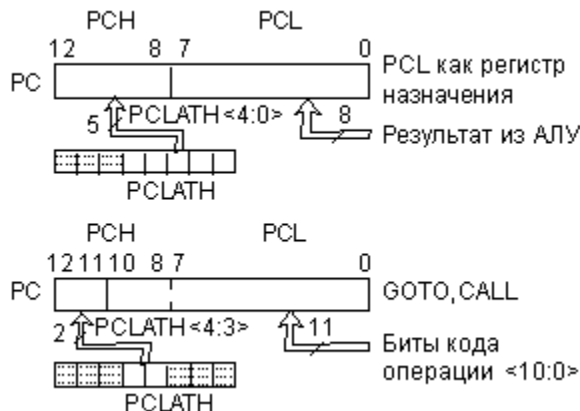


Рис. 5.6. Загрузка старших бит счетчика команд.

Команды **CALL** и **GOTO** оперируют 11-разрядным адресным диапазоном, достаточным для смещения в пределах страницы программной памяти объемом 2К слов. Для МК подгруппы PIC16F8X этого хватает. С целью обеспечения возможности расширения памяти команд для будущих моделей МК предусмотрена загрузка двух старших бит счетчика команд из регистра **PCLATH<4:3>**. При использовании команд **CALL** и **GOTO** пользователь должен убедиться в том, что эти страничные биты запрограммированы для выхода на нужную страницу. При выполнении команды **CALL** или выполнении прерывания весь 13-битный счетчик команд помещается в стек, поэтому для возвращения из подпрограммы не нужны манипуляции с разрядами **PCLATH<4:3>**.

Микроконтроллеры подгруппы PIC16F8X игнорируют значения бит **PCLATH<4:3>**, которые используются для обращения к страницам 1, 2 и 3 программной памяти. Однако применять биты **PCLATH<4:3>** в качестве ячеек памяти общего назначения не рекомендуется, так как это может повлиять на совместимость с будущими поколениями изделий.

Возможность выполнять арифметические операции непосредственно над счетчиком команд позволяет очень быстро и эффективно осуществлять табличные преобразования в PIC-контроллерах.

Микроконтроллеры подгруппы PIC16F8X имеют восьмиуровневый аппаратный стек шириной 13 бит (см. рис. 5.4). Область стека не принадлежит ни к программной области, ни к области данных, а указатель стека пользователю недоступен. Текущее значение счетчика команд посылается в стек, когда выполняется команда **CALL** или производится обработка прерывания. При выполнении процедуры возврата из подпрограммы (команды **RETLW**, **RETFIE** или **RETURN**) содержимое счетчика команд восстанавливается из стека. Регистр **PCLATH** при операциях со стеком не изменяется.

Стек работает как циклический буфер. Следовательно, после того как стек был загружен 8 раз, девятая загрузка переписет значение первой. Десятая загрузка переписет вторую и т.д. Если стек был выгружен 9 раз, счетчик команд становится таким же, как после первой выгрузки.

Признаков положения стека в контроллере не предусмотрено, поэтому пользователь должен самостоятельно следить за уровнем вложения подпрограмм.

Прямая и косвенная адресации

Когда производится прямая 9-битная адресация, младшие 7 бит берутся как прямой адрес из кода операции, а два бита указателя страниц (**RP1**, **RP0**) из регистра статуса, как показано на рис. 5.7.

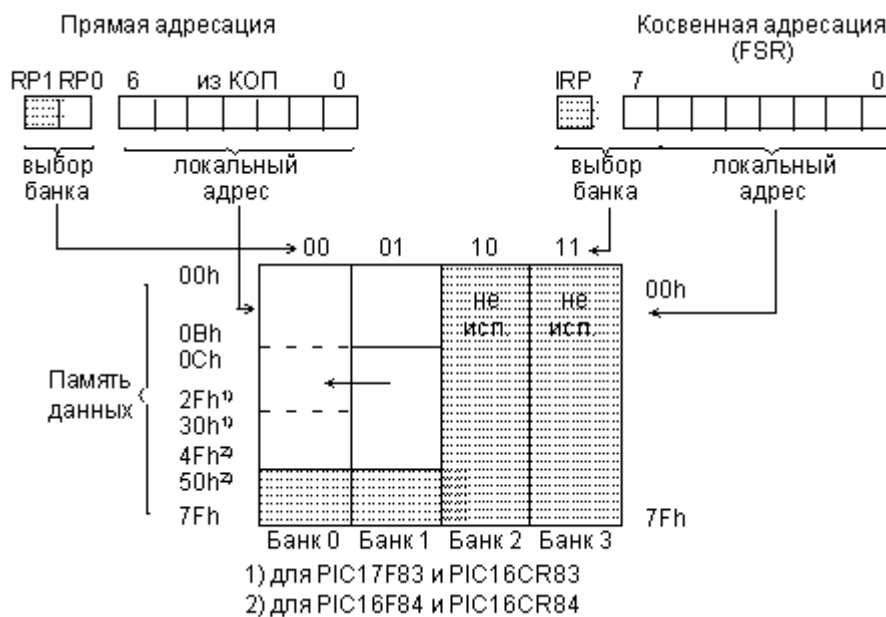


Рис. 5.7. Методы адресации данных.

Признаком косвенной *адресации* является обращение к *регистру* **INDF**. Любая команда, которая спользует **INDF** (адрес **00h**) в качестве *регистра* фактически обращается к указателю, который хранится в **FSR** (адрес **04h**). Чтение косвенным образом самого *регистра* **INDF** даст результат **00h**. Запись в *регистр* **INDF** косвенным образом будет выглядеть как **NOP**, но биты статуса могут быть изменены. Необходимый 9-битный адрес формируется объединением содержимого 8-битного **FSR** *регистра* и бита **IRP** из *регистра* статуса (см. рис. 5.7).

Обратите внимание, что некоторые *регистры* специальных функций располагаются в банке 1. Чтобы адресоваться к ним, нужно дополнительно установить в единицу бит **RPO** в *регистре* статуса.

Порты ввода/вывода

Контроллеры подгруппы PIC16F8X имеют два *порта*: **PORTA** (5 бит) и **PORTB** (8 бит) с побитовой индивидуальной настройкой на ввод или на вывод.

Порт А (**PORTA**) представляет собой 5-битовый фиксатор, соответствующий выводам контроллера **RA<4:0>**. Линия **RA4** имеет вход триггера Шмитта и выход с открытым стоком. Все остальные линии *порта* имеют ТТЛ входные уровни и КМОП выходные буферы. Адрес *регистра порта А* – **05h**.

Модуль таймера и регистр таймера

Структура модуля *таймера/счетчика* **TIMERO** и его взаимосвязь с *регистрами* **TMR0** и **OPTION** показаны на рис. 5.12. **TIMERO** является программируемым модулем и содержит следующие компоненты:

- 8-разрядный *таймер/счетчик* **TMR0** с возможностью чтения и записи как *регистр*;
- 8-разрядный программно управляемый предварительный делитель (предделитель);
- мультиплексор входного сигнала для выбора внутреннего или внешнего тактового сигнала;
- схему выбора фронта внешнего тактового сигнала;
- формирователь запроса *прерывания* по переполнению *регистра* **TMR0** с **FfH** до **00h**.

Память данных в РПЗУ (EEPROM)

Микроконтроллеры подгруппы PIC16F8X имеют энергонезависимую память данных 64x8 **EEPROM** бит, которая допускает запись и чтение во время нормальной работы (во всем диапазоне питающих напряжений). Эта память не принадлежит области регистровой памяти ОЗУ. Доступ к ней осуществляется посредством косвенной *адресации* через *регистры* специальных функций: **EEDATA <08h>**, который содержит 8-битовые данные для чтения/записи и **EEADR <09h>**, включающий адрес ячейки, к которой идет обращение. Для управления процессом чтения/записи используются два *регистра*: **EECON1 <88h>** и **EECON2 <89h>**.

Организация прерываний

МК подгруппы PIC16F8X имеют четыре источника *прерываний*:

- внешнее *прерывание* с вывода **RB0/INT**;
- *прерывание* от переполнения счетчика/*таймера* **TMR0**;
- *прерывание* от изменения сигналов на линиях *порта* **RB<7:4>**;
- *прерывание* по окончании записи данных в **EEPROM**.