Содержание:

ВВЕДЕНИЕ

1. СОСТАВЛЕНИЕ СХЕМЫ ЭЛЕКТРИЧЕСКОЙ СТРУКТУРНОЙ

2. СОСТАВЛЕНИЕ СХЕМЫ ЭЛЕКТРИЧЕСКОЙ ФУНКЦИОНАЛЬНОЙ

3. ОПИСАНИЕ ЭЛЕМЕНТНОЙ БАЗЫ

3.1 Резисторы

3.2 Конденсаторы

3.3 Резонатор

3.4 Светодиодный индикатор АЛ304Г

3.5 Диоды

3.6 Микросхемы

3.6.1 Микроконтроллер PIC16F84

3.6.2 Микросхема КР142ЕН5А

4. ОПИСАНИЕ РАБОТЫ СХЕМЫ ЭЛЕКТРИЧЕСКРОЙ ПРИНЦИПИАЛЬНОЙ ВЫБРАННОГО УСТРОЙСТВА

5. АЛГОРИТМ РАБОТЫ УСТРОЙСТВА И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

6. ЗАКЛЮЧЕНИЕ

7. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЕ А

Введение

Однокристальные микроконтроллеры стремительно занимают ведущее место в электронной аппаратуре. И если каких-нибудь десять лет назад они могли быть использованы при проектировании только профессионалами — слишком много требовалось дорогостоящих средств для их программирования, — то сегодня «однокристалки» используют даже радиолюбители.

Именно микроконтроллер сегодня формирует облик бытовой аудиотехники, видеотехники, средств связи. С передних панелей радиоприемников, магнитол, телевизоров исчезли шкальные индикаторы, ручки настройки, ползунковые регуляторы, переключатели. Их заменили жидкокристаллические табло и кнопки. Десятки разнообразных микросхем, составляющих электронное «нутро» аппаратуры, должны согласованно функционировать, обеспечивая наилучшие технические характеристики и удобство эксплуатации. Решая задачу сопряжения цифровых устройств, разработчики компьютерной техники связали их между собой посредством шин, а способы передачи информации назвали протоколами и стандартизировали их основные принципы. Каждое устройство было снабжено интерфейсом, посредством которого стало возможным подключать его к шине.

Если говорить об аналоговой технике, то специализированные аналоговые микросхемы испокон веков имели уникальные собственные выводы для подключения элементов, с помощью которых устанавливались их рабочие режимы или осуществлялись регулировки в процессе эксплуатации. Объединять эти выводы с целью выполнения универсальной регулировки не представлялось возможным. Современные микросхемы, предназначенные для использования в новой аппаратуре, снабжены цифровым интерфейсом, «спрятали» настроечные выводы внутрь микросборок, а управлять их режимами программно.

Осуществлять управление одной микросхемой очень просто, для этого даже не всегда нужен внешний цифровой интерфейс. А если микросхем несколько, если они должны взаимодействовать друг с другом, обмениваться информацией? Решая эти вопросы, разработчики перспективных микросхем пришли к мысли о необходимости введения общей шины и протокола обмена информацией по ней. Известные по компьютерной технике шины оказались здесь совершенно негодными в силу своей сложности.

Для этих целей фирмой Philips была разработана шина Inter-Integrated Circuit Bus (сокращенно I2С). Принадлежа к классу шин с последовательным способом передачи данных, отличаясь чрезвычайной простотой реализации, шина стала быстро развиваться. Сегодня и другие фирмы, оценившие по достоинству возможности шины, поддерживают ее своей продукцией.

Темой данного курсового проекта, является разработка «Термометра» на основе микроконтроллера.

1 Составление схемы электрической структурной

На основании анализа схемы электрической принципиальной были выделены основные функциональные элементы, и разработана схема электрическая структурная, представленная на рисунке 1.1

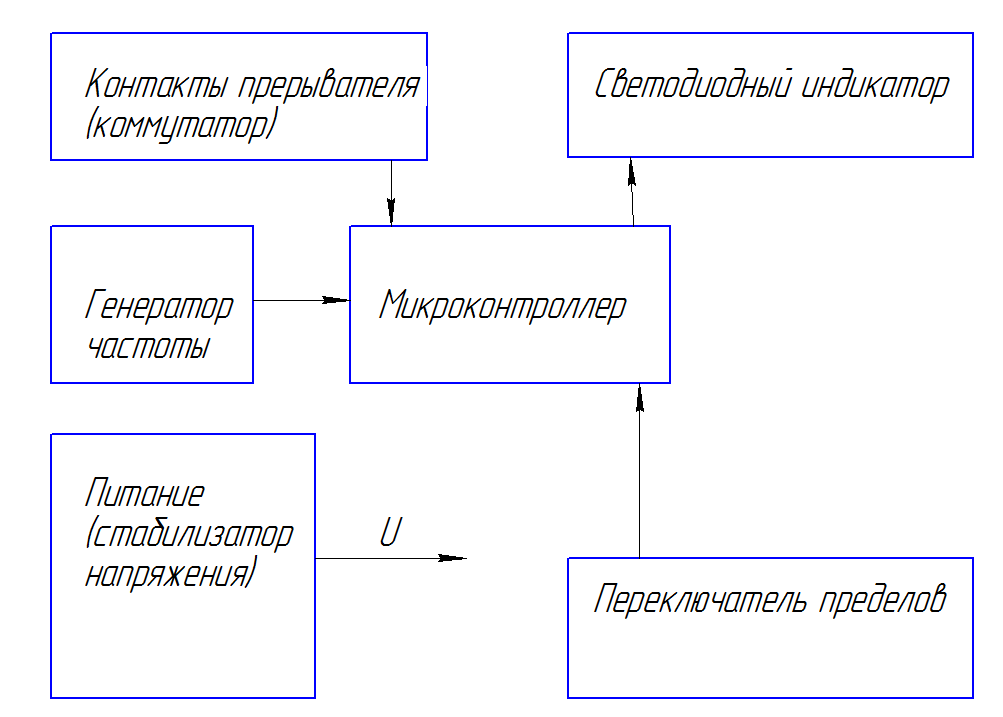


Рисунок 1.1 - Схема электрическая структурная

Схема электрическая структурная содержит: микроконтроллер, контакты прерывателя, переключатель пределов, светодиодный индикатор, генератор тактовой частоты и источник питания.

Этот тахометр предназначен для использования при регулировке холостого хода карбюраторов двигателей внутреннего сгорания. Его можно применять и для контроля частоты вращения вала автомобильных или лодочных двигателей во время движения.

Тахометр имеет три разряда индикации с пределом измерения от 60 мин"1 до 7800 мин-1. Погрешность измерения на пределе 1 секунда равна 30 мин-1, а на пределе 3 секунды — 10 мин-1. Нижний предел ограничен погрешностью измерения, а верхний — количеством прерываний между индикацией. Из-за чего индикация разрядов становится прерывистой.

2 Составление схемы электрической функциональной

Схема электрическая функциональная представлена на рисунке 2.1

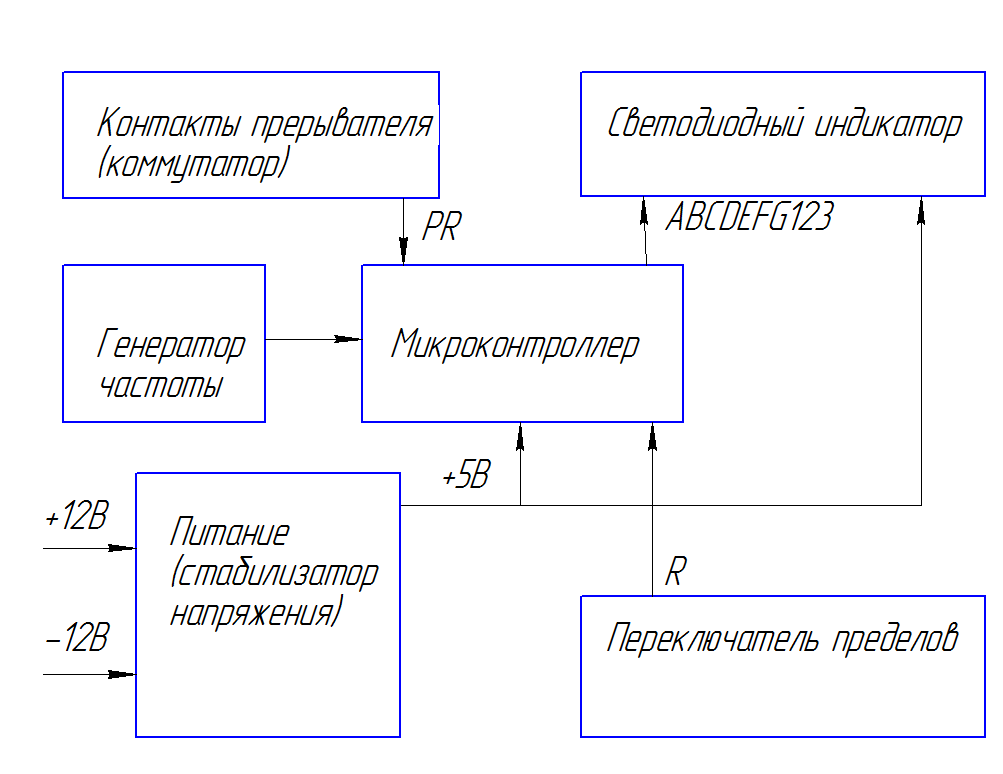


Рисунок 2.1 - Схема электрическая функциональная

3 Описание элементной базы

3.1 Резисторы

(номиналы см. принципиальную схему)

R3-R10 – резисторы С2-23 не проволочные постоянного сопротивления, мощностью 0,125Вт с номинальными сопротивлениями R3-R9 430 Ом, R2 10 кОм с допуском ±10%. R1- не проволочный постоянного сопротивления, мощностью 0,5 Вт с номинальным сопротивлением 34 кОм.

3.2 Конденсаторы

С1 и С2 – конденсаторы К10-17 керамические не полярные постоянной емкости с группой по ТКЕ – Н90, с номинальным напряжением 25В, номинальной емкостью 30 пФ с допуском ±10%. C3 – 0,01 мкФ.

3.3 Резонатор

ZQ – кварцевый резонатор, работающий на частоте параллельного резонанса 32768 Гц.

3.4 Светодиодный индикатор АЛ304Г

АЛ304А, АЛ304Б, АЛ304В, АЛ304Г

Индикаторы знакосинтезирующие, на основе соединения арсенид – фосфид –галлий, эпитаксиально – планарные [41, стр. 475 – 478]. Предназначены для отображения цифровой информации. Индикаторы имеют 7 сегментов и децимальную точку. Выпускаются в пластмассовом корпусе. Высота знака 3 мм.

Масса прибора не более 0,25 г.

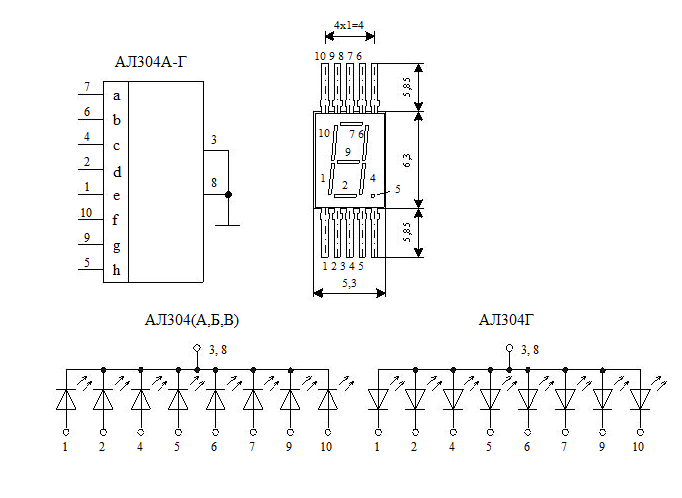


Рисунок 3.1 – Структурная схема и монтажные размеры модулей

АЛ304Г: 1 – катод e; 2 – катод d; 3, 8 – анод общий; 4 – катод c; 5 – катод h; 6 –катод b; 7 – катод a; 9 – катод g; 10 – катод f.

3.5 Диоды

VD1 – стабилитрон КС147А Uст. = 4,23-5,17В Iст. = 58мА, VD2 – диод КД102.

Таблица 3.1- Электрические параметры КД102

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Uоб/Uимп  В/В | Iпр/Iимп  А/А | Uпр/Iпр  В/А | Io(25)Ioм  мкА/мкА | Fmax  кГц |
| 250/250 | 0.1/2 | 1.0/0.05 | 0.1/50 | 4 |

3.6 Микросхемы

3.6.1 Микроконтроллер PIC16F84

Контроллеры PIC16F84, как и все микроконтроллеры с торговой маркой PICmicro™ основаны на развитой RISC-архитектуре. Они имеют расширенные опции ядра, восьмиуровневый стек и различные внутренние и внешние прерывания. 14-битные слова команд и 8-битные данные передаются независимо, по разделенным шинам памяти и данных. Большинство команд исполняется за один машинный цикл кроме команды переходов, которые исполняются за два цикла. Набор команд состоит из 35 инструкций с интуитивно понятной мнемоникой.

На кристалле расположены 64 байта EEPROM памяти констант с гарантированным сроком хранения данных более 40 лет при отключенном питании, до 68 байт памяти данных (рабочие регистры для хранения переменных). Кристаллы выпускаются с максимальными тактовыми частотами 4МГц и 10МГц, имеют 13 портов ввода/вывода, встроенный таймер/счетчик TMR0, сторожевой таймер WDT, экономичный режим засыпания SLEEP.

Тактовую частоту можно задавать при помощи RC-цепочки, недорогого керамического резонатора или кварцевого резонатора. Может быть подключен также внешний генератор тактовой частоты. Встроенного тактового генератора, тактирующего процессор, нет. От встроенного на кристалл RC-генератора могут тактироваться только таймер/счетчик TMR0 или сторожевой таймер, по выбору.

Варианты корпуса 18-pin DIP, SOIC.

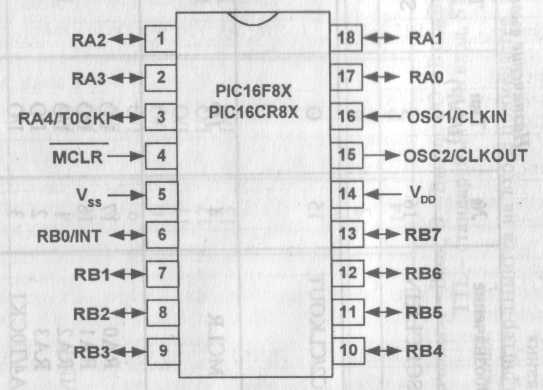


Рисунок 3.2 - Расположение выводов PIC16F84

Таблица 3.2 - Назначение выводов PIC16F84

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Обозначение | № | Тип  (I/O/P) | Тип буфера | Назначение |
| OSC1/CLKIN | 16 | I | ST/CMOS(3) | Вход для подключения кварцевого резонатора, либо RC-цепочки, либо вход  для внешнего тактового генератора |
| OSC2/CLKOUT | 15 | О | - | Выход для подключения кварцевого резонатора в режиме работы с кварцем, в ре-  жиме RC-генератора на выходе присутствуют импульсы с частотой 1/4 от OSC1 |
| MCLR | 4 | I/Р | ST | Сброс по низкому уровню на входе.  При программировании кристалла -  вход напряжения программирования |
| RA0  RA1  RA2  RA3  RA4/T0CKI | 17  18  1  2  3 | I/O  I/O  I/O  I/O  I/O | TTL  TTL  TTL  TTL  ST | Выводы двунаправленного порта А  RA4/T0CKI может быть настроен как  вход импульсов для таймера-счетчика  TMR0. Выход с открытым стоком |
| RB0/INT  RB1  RB2  RB3  RB4  RB5  RB6  RB7 | 6  7  8  9  10 11  12  13 | I/O  I/O  I/O  I/O  I/O  I/O  I/O  I/O | TTL/ST(1)  TTL  TTL  TTL  TTL  TTL  TTL/ST(2)  TTL/ST(2) | Выводы двунаправленного порта В. К  выводам порта могут быть программно  подключены внутренние подтягивающие резисторы с Vdd.  RB0/INT может быть также программно  настроен как вход внешнего прерывания.  RB4…RB7 могут быть также программно настроены как входы прерывания по изменению уровня на любом из этих входов. Направление изменения задается  программно.  При программировании кристалла RB6  используется как тактовый, RB7 как  вход/выход данных. |
| Vss | 5 | Р | — | Общий провод |
| Vdd | 14 | Р | — | Положительное напряжение питания |

Обозначения: I - Вход, О - Выход, I/O - Вход/Выход, Р - вывод питания,

TTL - стандартный TTL вход, ST - вход с триггером Шмитта.

Примечания:

1. Триггер Шмитта на входе применяется, только если вывод сконфигурирован, как вход внешнего прерывания.

2. Триггер Шмитта на входе применяется только в режиме последовательного программирования кристалла.

3. Триггер Шмитта на входе применяется только если включен режим RC-генератора, в остальных случаях как CMOS вход.

Микроконтроллеры PIC16F84 могут программироваться непосредственно на плате устройства, что позволяет отлаживать программу либо записывать константы и калибровочные данные. Для программирования на плате необходимо максимум пять проводов: питание +5В, напряжение программирования, последовательные данные, тактирующие импульсы и общий. Память программ только встроенная.

Таблица 3.3 - Электрические параметры

|  |  |  |
| --- | --- | --- |
| Параметр | Значение | Примечание |
| Рабочая температура Та,°С | 0 ... +40  -40... +85 | Коммерческое исполнение.  Индустриальное исполнение |
| Максимальная тактовая частота Fosc, МГц | 4  10 | PIC16F84-4  PIC16F84-10 |
| Напряжение питания  Vdd,B | 2,0... 6,0  4,0... 6,0 | PIC16LF84  PIC16F84 |
| Потребляемый ток в стандартном режиме Idd, мА | 1,8 ...4,5  5... 10 | Fosc=4 МГц, Vdd=5.5 В Fosc=10MГц, Vdd=5.5B |
| Потребляемый ток в стандартном режиме Idd, мA | 1 ...4  15... 45 | Fosc=2 МГц, Vdd=5.5 В  Fosc=32 кГц, Vdd=2.0 В,  WDT отключен |
| Потребляемый ток в режиме SLEEP Ipd, мкА для PIC16F84 | 7... 28  1 ... 16  1 ... 14 | Vdd=4 В, WDT вкл., инд.  Vdd=4 В, WDT выкл., инд.  Vdd=4 В, WDT выкл., комм. |
| Потребляемый ток в режиме SLEEP Ipd, мкА для PIC16LF84 | 3... 16  0,4 ...9  0,4... 7 | Vdd=2 В, WDT вкл., инд.  Vdd=2 В, WDT выкл., инд.  Vdd=2 В, WDT выкл., инд. |
| Максимальный втекающий ток для любого вывода, мА | 25 | Управление светодиодами без дополнительного буфера (но с резистором!) |
| Максимальный вытекающий ток для любого вывода, мА | 20 | Управление светодиодами без дополнительного буфера (но с резистором!) |
| Количество циклов стирание/запись для флэш-памяти программ, не менее | 1000 |  |
| Количество циклов стирание/запись для памяти данных EEPROM, не менее | 10.000.000 |  |

Организация памяти.

В микроконтроллерах PIC16F84 существует два блока памяти - память программ и память данных. Каждый блок имеет собственную шину, таким образом, доступ к блокам может происходить одновременно.

Память данных, в свою очередь, разделена на специальные регистры и регистры общего применения (ОЗУ пользователя). Специальный регистры применяются для хранения битов состояния, определяющих работу портов вода/вывода, таймеров и других периферийных модулей контроллера.

Кроме специальных регистров и ОЗУ, пространство памяти данных содержит ячейки EEPROM. Эта область памяти не может быть адресована непосредственно, и доступ к ней получают через специальный регистр косвенной адресации EEADR, в который записывают порядковый номер ячейки. 64 байта EEPROM имеют номера с 00h по 3Fh. Обычно EEPROM используется для хранения констант, значения которых не должны пропадать при отключении питания, например кодов управления, индивидуальных номеров и т.п. Важным достоинством EEPROM является то, что данные в ней могут быть изменены даже после занесения программы в однократно программируемый кристалл.

Память программ.

Микроконтроллеры PIC16F84 имеют 13-битный программный счетчик, позволяющий адресовать до 8К х 14 памяти программ. В PIC16F84 доступны первые 1024 (0000h-03FFh) ячеек памяти. Обращение к старшим адресам, лежащим за пределами указанного диапазона физически равносильно обращению к соответствующим адресам внутри диапазона, например, адреса 30h, 430h, 830h, C30h, 1030h, 1430h, 1830h и 1C30h равносильны и адресуют одну и ту же команду.

Старт по сбросу происходит с адреса 0000h, вектор прерывания один и расположен по адресу 0004h. Обычно по адресу 0004h располагают подпрограмма распознавания и обработки прерываний, а по адресу 0000h команду перехода на метку, расположенную за подпрограммой обработки прерывания.

Память данных.

В микроконтроллерах PIC16F84 память данных разбита на две части - специальные регистры и регистры общего применения (ОЗУ пользователя). Кроме этого, память данных разделена на два банка.

Таблица 3.4 - Организация памяти данных

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес |  |  | Адрес |
| 00h | Косвенный адрес | Косвенный адрес | 80h |
| 0lh | TMR0 | OPTION | 81h |
| 02h | PCL | PCL | 82h |
| 03h | STATUS | STATUS | 83h |
| 04h | FSR | FSR | 84h |
| 05h | PORTA | TRISA | 85h |
| 06h | PORTB | TRISB | 86h |
| 07h | Недоступен | Недоступен | 87h |
| 08h | EEDATA | EECON1 | 88h |
| 09h | EEADR | EECON2 | 89h |
| 0Ah | PCLATH | PCLATH | 8Ah |
| 0Bh | INTCON | INTCON | 8Bh |
| 0Ch  4Fh | 68 регистров общего применения (SRAM) | Отображается на пространство банка 0 | 8Ch  CFh |
| 50h  7Fh |  |  | D0h  FFh |
|  | Банк 0 | Банк 1 |  |

Переключение банков происходит при помощи задания 5-го бита в регистре STATUS. Если бит установлен в 0, адресуется нулевой банк, если в 1, соответственно, первый.

Специальные регистры

Специальные регистры представляют собой статическое ОЗУ.

Регистр STATUS (Адрес 03Н, 81Н)

Регистр STATUS хранит арифметические флаги АЛУ, информацию о сбросе и бит выбора банка памяти данных.

bit7 IRP - регистр выбора банка памяти, применяемый при косвенной адресации. Это бит не применяется в PIC16F84 и должен всегда оставаться сброшенным.

0 = bank 0,1 (00h-FFh)

1 = bank 2,3(100h-1FFh)

bit6-5 RP1, RP0 - регистр выбора банка памяти, применяемый при прямой адресации.

00 = bank 0 (00h-7Fh)

01 = bank 1 (80h-FFh)

bit4 TO - флаг срабатывания сторожевого таймера. Устанавливается в 1 при включении питания и командами CLRWDT и SLEEP. Сбрасывается в 0 по завершении выдержки сторожевого таймера.

bit3 PD - режим хранения данных. Устанавливается в 1 при включении питания или выполнении команды CLRWDT. Сбрасывается в 0 командой SLEEP.

bit2 Z - флаг нулевого результата. Устанавливается в 1, если результат арифметической или логической операции равен нулю. Сохраняет свое значение до следующей операции.

bit1 DC - флаг десятичного переноса. Используется для команд ADDWF, ADDLW, SUBWF и SUBLW. Отслеживает перенос из четвертого разряда результата.

1 = Произошел перенос при сложении

0 = Не произошел перенос при сложении

Вычитание в АЛУ выполняется сложением кода первого операнда с дополнительным кодом второго операнда. Значение бита контекстно зависит от того, какая операция выполнялась. Для операции вычитания значения бита инвертированы.

bit0 С - флаг переноса. Используется для команд ADDWF, ADDLW, SUBWF и SUBLW. Отслеживает перенос из старшего разряда в бит переноса при сложении.

1 = произошел перенос при сложении

0 = не произошел перенос при сложении

Вычитание в АЛУ выполняется сложением кода первого операнда с дополнительным кодом второго операнда. Значение бита контекстно зависит от того, какая операция выполнялась. Для операции вычитания значения бита инвертированы.

Используя флаги ТО и PD можно определить, чем был вызван сброс.

|  |  |  |
| --- | --- | --- |
| TO | PD | Событие, вызвавшее состояние "сброс" |
| 1 | 1 | Сброс по включению питания |
| 0 | 1 | Сработал сторожевой таймер (не в режиме SLEEP) |
| 1 | 0 | Сброс по входу MCLR в режиме SLEEP или выход из SLEEP по внешнему прерыванию |
| 0 | 0 | Выход из SLEEP по сигналу сторожевого таймера |
| X | X | Сброс по входу MCLR в обычном режиме |

х - состояние битов не изменилось. Сброс по входу MCLR в обычном режиме не меняет текущие значения битов ТО и PD.

Регистр OPTION\_REG (АДРЕС 81Н)

Специальный регистр OPTION\_REG представляет собой полностью доступный для записи и чтения регистр, в котором находятся биты, управляющие работой предварительного делителя, источниками внешних прерываний, встроенным таймером TMR0 и подтягивающими резисторами для порта В.

bit7 RBPU - включение встроенной нагрузки порта В

1 = нагрузка отключена

0 = нагрузка включена

bit6 INTEDG - выбор фронта прерывающего сигнала

1 = прерывание по нарастанию сигнала на выводе RB0/INT

0 = прерывание по спаду сигнала на выводе RB0/INT

bit5 TOGS - выбор источника тактирования для таймера TMR0

1 = импульсы со входа RA4/T0CKI

0 = внутренняя тактовая частота (CLKOUT)

bit4 T0SE - выбор фронта сигнала для таймера TMR0, если в качестве источника выбран вход RA4/T0CKI (T0CS=l)

1 = инкремент по спаду на выводе RA4/T0CKI

0 = инкремент по нарастанию на выводе RA4/T0CKI

bit3 PSA - бит, управляющий подключением предварительного делителя

1 = предварительный делитель подключен к WDT

0 = предварительный делитель подключен к TMR0

bit2-0 PS2-PS0 - управление коэффициентом деления предварительного делителя в зависимости от подключения

|  |  |  |
| --- | --- | --- |
| Биты | для TMR0 | для WDT |
| 000 | 1:2 | 1:1 |
| 001 | 1:4 | 1:2 |
| 010 | 1:8 | 1:4 |
| 011 | 1:16 | 1:8 |
| 100 | 1:32 | 1:16 |
| 101 | 1:64 | 1:32 |
| 110 | 1:128 | 1:64 |
| 111 | 1:256 | 1:128 |

Регистр INTCON (АДРЕС 0ВН, 8ВН)

Регистр INTCON - это полностью доступный для чтения и записи регистр, в котором хранятся биты, управляющие различными источниками прерываний.

bit7 GIE - бит глобального запрета прерываний

1 = разрешены все немаскируемые прерывания

0 = запрещены все прерывания

bit6 EEIE - разрешение прерывания по окончанию записи в EEPROM

1 = прерывание по окончанию записи разрешено

0 = прерывание по окончанию записи запрещено

bit5 T0IE - разрешение прерывания по переполнению TMR0

1 = прерывание разрешено

0 = прерывание запрещено

bit4 INTE - разрешение прерывания по входу RB0/INT

1 = прерывание разрешено

0 = прерывание запрещено

bit3 RBIE - разрешение прерывания по изменению состояния на входах порта В, линии RB7-RB4

1 = прерывание разрешено

0 = прерывание запрещено

bit2 T0IF - флаг прерывания по переполнению таймера/счетчика TMR0

1 = TMR0 был переполнен (следует сбросить программно!)

0 = TMR0 не был переполнен

Флаг используется для определения источника прерывания,

bit1 INTF - флаг прерывания по входу RB0/INT

1 = произошло прерывание по входу RB0/INT (следует сбросить программно!)

0 = не происходило прерывания по входу RB0/INT

Флаг используется для определения источника прерывания.

bit0 RBIF - флаг прерывания по изменению состояния на входах RB7-RB4

1 = на одном из выводов RB7-RB4 произошло изменение уровня (следует сбросить программно!)

0 = не происходило прерывание по изменению уровня

Флаг используется для определения источника прерывания.

Программный счетчик.

Программный счетчик микроконтроллера (PC) содержит 13 разрядов. Младший байт счетчика является полностью доступным для чтения и записи регистром PCL (адрес 02h, 82h). Старшие пять байтов счетчика непосредственно не доступны для чтения и записи. Обращение к ним происходит через регистр PCLATCH (адрес 0Ah, 8Ah), являющийся буфером-защелкой для старших битов счетчика. Содержимое PCLATCH переносится в старшие биты PC, когда происходит запись нового значения в программный счетчик. Это случается, когда выполняются команды CALL, GOTO или регистр PCL является регистром назначения для результата арифметической операции.

Стек

Микроконтроллеры PIC16F84 имеют 8-уровневый 13-битный аппаратный стек. Стек не является частью памяти данных или программ и указатель стека не доступен для чтения или записи.

Косвенная адресация: регистры INDF и FSR

Регистр INDF не является физическим регистром. При обращении к этому регистру на самом деле адресуется регистр, адрес которого указан в регистре FSR (т.е. FSR является указателем). Такая адресация является косвенной.

Порты ввода-вывода

Контроллеры PIC16F84 имеют два порта ввода-вывода, PORTA и PORTB. Каждый вывод порта может быть запрограммирован на ввод или на вывод установкой соответствующего бита в регистрах TRISA и TRISB. Выводимые значения фиксируются в регистрах-защелках PORTA и PORTB. Направление ввода-вывода может быть изменено в произвольный момент времени.

Двунаправленный порт А, регистры PORTA и TRISA

Регистр PORTA представляет собой 5-битную защелку. Линия RA4 имеет триггер Шмитта на входе в режиме ввода и открытый сток в режиме вывода. Остальные линии порта А по входу работают со стандартными уровнями TTL, выходы подключаются к комплементарным выходным CMOS драйверам. Линия RA4 используется также как вход внешних тактовых импульсов для таймера TMR0.

Направление передачи данных для каждой линии программируется отдельно, установкой или сбросом битов bit0...bit4 регистра TRISA. Установка бита в 1 настраивает соответствующую линию на ввод. Выходной драйвер при этом переходит в высокоимпедансное состояние. Установка бита в 0 настраивает линию порта на вывод и выводит на нее содержимое соответствующего бита защелки PORTA. По умолчанию при включении питания все линии настроены на ввод. При чтении порта А всегда считываются действительные логические уровни на выводах, независимо от того, запрограммированы отдельные разряды как входы или как выходы.

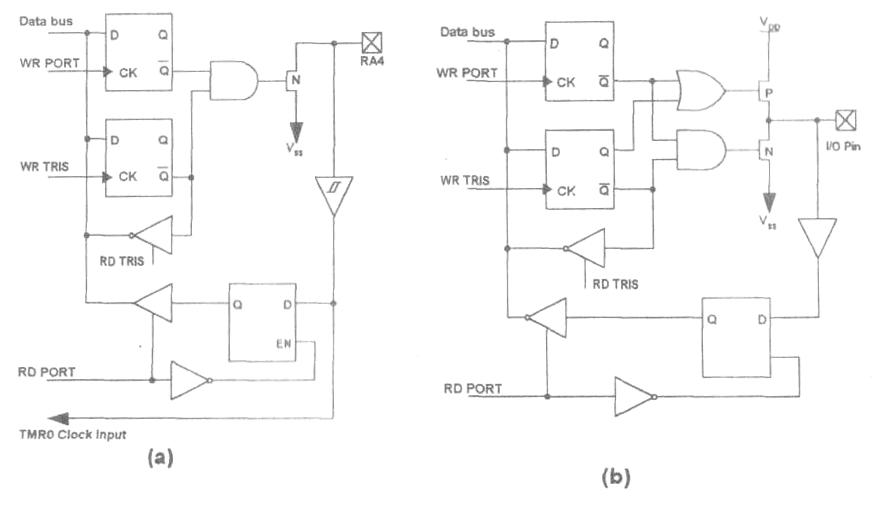


Рисунок 3.3 - Блок-схема порта А, линии RA4 (а) и RA0-RA3 (b)

Двунаправленный порт В, регистры PORTB и TRISB

Порт В представляет собой 8-битный двунаправленный порт. Выходные значения записываются в регистр-защелку PORTB. Направление ввода-вывода определяется установкой или сбросом битов регистра TRISB. Установка бита в 1 настраивает соответствующую линию на ввод, переводя выходной драйвер в высокоимпедансное состояние, а-0 - на вывод. При включении питания все линии по умолчанию настроены на ввод. Как и для порта А, чтение порта В всегда возвращает действительные значения на выводах, независимо от направления передачи данных для каждого вывода.

Все выводы порта В имеют встроенную отключаемую нагрузку в виде резисторов, подключенных к шине питания (подтягивающие резисторы). Нагрузка включается и отключается одновременно для всех выводов при помощи бита 7 RBPU регистра OPTION\_REG. При включении питания RBPU= 1 и нагрузка отключена. Программное обнуление бита RBPU подключает нагрузку, но для линий, настроенных на вывод нагрузка автоматически отключается.

Линии RB4...RB7 могут использоваться как входы прерывания по изменению уровня. В этом качестве используются только линии, настроенные на ввод. В каждом командном цикле происходит сравнение текущих значений на выводах с предыдущими, зафиксированными в специальной защелке. Если хотя бы на одном из этих выводов произошло изменение уровня, формируется прерывание. Длительность импульса, который распознается как изменение уровня, должна быть не менее 4-х периодов тактовой частоты. Программно распознать, по какой из линий RB4...RB7 произошло прерывание, невозможно.

Это прерывание выводит контроллер из состояния SLEEP.

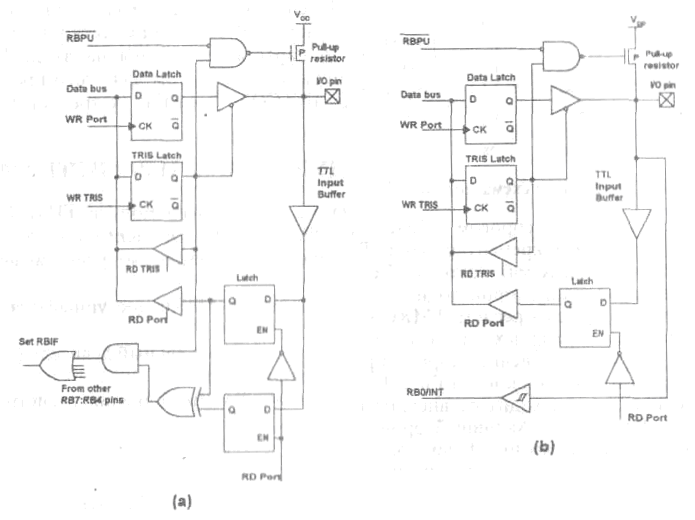


Рисунок 3.4 - Блок-схема порта В, линии RB7-RB4 (а) и RB3-RB0 (b)

Модуль TIMER0 и регистр TMR0

TIMER0 является программируемым модулем таймера/счетчика. Он имеет в своем составе:

* 8-битный таймер/счетчик TMR0, доступный для чтения и записи как регистр,
* программируемый предварительный делитель (предделитель) мультиплексор входного сигнала
* генератор прерывания по переполнению регистра TMR0 с FFh в 00h.

Предделитель

Предделитель является 8-битным счетчиком, который также может быть использован, как выходной делитель (постделитель) сторожевого таймера. Если предделитель подключен к модулю TIMER0, то он не может использоваться со сторожевым таймером, и наоборот. Когда предделитель подключен к таймеру/счетчику, все команды, использующие запись в регистр TMR0, обнуляют предделитель. Если предделитель подключен к сторожевому таймеру, они обнуляются совместно, командой CLRWDT. Предделитель недоступен для прямой записи или чтения.

Подключение предделителя может быть изменено "на лету", то есть во время выполнения программы.

Работа с EEPROM

Память данных EEPROM доступна для чтения и записи во всем рабочем диапазоне питающих напряжений и предназначена для хранения 8-битных значений. Перед записью нового значения предыдущее стирается. Микроконтроллеры PIC16F84 имеют 64 байта EEPROM с адресами от 00h до 3Fh, но эти ячейки недоступны путем прямой адресации в адресном пространстве микроконтроллера. Для доступа к ним используется косвенная регистровая адресация через специальные регистры. Всего при работе с EEPROM используется четыре специальных регистра:

* EECON1
* EECON2
* EEDATA
* EEADR

Регистр обмена EEDATA содержит 8-битные данные чтения/записи. EEADR хранит адрес ячейки, к которой происходит обращение. Несмотря на то, что в PIC16F84 физически существует только 64 байта EEPROM, декодируются все биты адреса. Поэтому необходимо следить за значением в EEADR, чтобы не выйти за пределы адресного пространства. При записи в EEPROM требуется строго выдерживать временной интервал, который контролируется встроенным таймером. Время записи может варьироваться от кристалла к кристаллу, а также в зависимости от питающего напряжения и температуры.

Когда у микроконтроллера установлен бит защиты кода, процессор может читать и записывать EEPROM, но для программатора эта память становится недоступна.

Регистры EECON1 и EECON2

Регистр EECON1 является контрольным регистром, у которого физически доступны младшие пять бит. Старшие три недействительны и всегда читаются как "0".

bit 7-5 Физически недоступны, всегда читаются как "0"

bit 4 EEIF - флаг прерывания по окончанию записи

1 = запись завершена (должен быть сброшен программно)

0 = запись не завершена или не начиналась

bit 3 WRERR - флаг ошибки записи в EEPROM

1 = запись преждевременно прервана

0 = запись прошла успешно

bit 2 WREN - разрешение записи в EEPROM

1 = разрешен цикл записи

0 = запрещена запись

bit1 WR - бит управления записью

1 = начать цикл записи. Программная установка бита является командой начать цикл записи. Сбрасывается этот бит только аппаратно, когда цикл записи окончен.

0 = цикл записи данных завершен

bit1 RD - бит управления чтением

1 = начать чтение данных из EEPROM. Чтение занимает один командный цикл. Программная установка бита является командой начать чтение. Бит сбрасывается только аппаратно.

0 = чтение не начато

Невозможность программно сбросить бит WR предохраняет от случайного преждевременного прерывания цикла записи, поскольку этот цикл занимает несколько машинных тактов.

Бит WREN при включении питания сброшен, что предохраняет от случайной записи. Бит WRERR устанавливается, когда операция записи прервана сбросом по входу MCLR или сбросом по переполнению сторожевого таймера. В этом случае, при повторном старте, пользователь может проверить бит WRERR и, при необходимости, повторить запись. Данные и адрес в регистрах EEDATA и EEADR при сбросе не теряются.

Регистр EECON2 не является физическим регистром и используется исключительно в качестве служебного регистра при записи. Чтение этого регистра всегда возвращает значение "0".

Чтение данных из EEPROM

Для чтения данных необходимо записать адрес в регистр EEADR и установить бит RD регистра EECON1. В следующем цикле данные уже доступны для чтения из регистра EEDATA. Прочитанные данные хранятся в этом регистре, пока не будут прочитаны новые данные или пока в него не будут занесены данные для записи.

Запись данных в EEPROM

Для записи данных в EEPROM необходимо сначала записать адрес в регистр EEADR и данные для записи в регистр EEDATA, а затем выполнить обязательную последовательность команд, рекомендованных изготовителем:

MOVLW 55h

MOVWF EECON2 ;записываем 55h

MOVLW AAh

MOVWF EECON2 ;записываем AAh

BSF EECON1, WR ;старт записи данных

Процесс записи не будет инициирован, если не будет выполнена поочередная запись 55h и AAh в регистр EECОN2. перед тем, как установить бит WR. Изготовитель настоятельно рекомендует запрещать все прерывания на момент исполнения этого фрагмента программы. Если прерывания в работе устройства вообще не используются, то нет надобности в запрете и последующем разрешении прерываний, так как по включению питания все прерывания запрещены.

Бит WREN не сбрасывается аппаратно, поэтому его необходимо сбросить программно после окончания записи всех данных. Этот бит предохраняет от записи случайных данных в EEPROM, например, при сбое программы. Необходимо тщательно следить, чтобы этот бит был сброшен всегда, когда не производится запись данных. Сброс бита WREN во время начатого цикла записи не повлияет на его успешное завершение.

Слово конфигурации CPU

Слово конфигурации расположено по адресу 2007h. Этот адрес находится за пределами пользовательской памяти программ и входит в состав специального адресного пространства (2000h - 3FFFh), которое доступно только для программатора во время программирования. Слово конфигурации содержит 14 бит.

bit 13-4 СР - бит защиты программного кода

1 = защита отключена

0 = защита установлена

bit 3 PWRTE - бит разрешения задержки при включении питания

1 = задержка отключена

0 = задержка включена

bit 2 WDTE - бит включения сторожевого таймера

1 = сторожевой таймер включен

0 = сторожевой таймер выключен

bit 1-0 FOSC1-FOSC0 - бит выбора режима тактового генератора

11= RC-генератор

10 = HS резонатор

01 = XT резонатор

00 = LP резонатор

Обозначения резонаторов в данном случае следующие: XT - стандартный кварцевый или керамический резонатор 4МГц, LP - низкочастотный (обычно часовой, 32768Гц) резонатор для экономичных приложений, HS - высокочастотный кварцевый резонатор 10МГц, RC - генератор на основе внешней RC-цепочки.

Внешние источники тактовой частоты.

Микроконтроллеры PIC16F84 не имеют встроенного тактового генератора, работающего без внешних элементов. Для тактирования необходим либо кварцевый резонатор, либо независимый тактовый генератор.

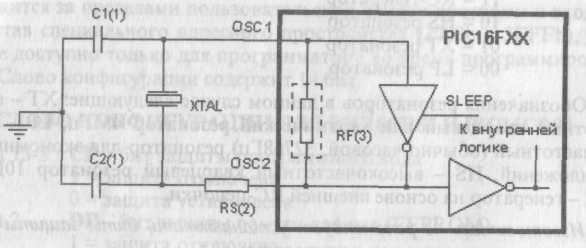


Рисунок 3.5 - Схема подключения кварцевого резонатора

Для нормальной работы собственного генератора требуется кварц, работающий на частоте параллельного резонанса.

Таблица 3.5 - Значения емкостей С1 и С2 для кварцевых резонаторов

|  |  |  |
| --- | --- | --- |
| Режим | Частота | С1,С2 |
| LP | 32 кГц  200 кГц | 68-100пФ  15-ЗЗпФ |
| XT | 100 кГц  2.0 МГц  4.0 МГц | 100-150 пФ  15 - 33 пФ  15-ЗЗпФ |
| HS | 8.0 МГц  10. 0 МГц | 15-ЗЗпФ  15-ЗЗпФ |

При напряжении питания больше 4,5 В изготовитель рекомендует применять конденсаторы со значениями С1 = С2 = 30 пФ.

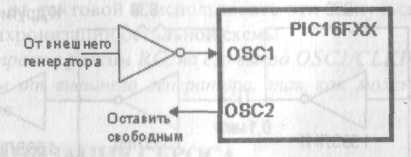


Рисунок 3.6 - Схема подключения внешнего генератора

При использовании независимого внешнего генератора, его схема может быть любой. Выход генератора подключается к выводу OSC1, вывод OSC2 обязательно должен остаться свободным, в противном случае контроллер может выйти из строя. Контроллер в этом случае настраивается в режим XT, LP или HS.

Если приложение некритично к величине и стабильности тактовой частоты, можно применить недорогой RC-генератор. Резистор и конденсатор являются внешними элементами.

Рекомендуемый номинал резистора лежит в пределах от 5 кОм до 100 кОм. При использовании резистора менее 4 кОм генерация может быть нестабильной или вообще не возникнет. При слишком большом номинале, порядка 1 МОм и выше, на работу генератора начинают влиять внешние наводки и шумы схемы, а также монтажная емкость и влажность платы. Несмотря на то, что генератор может работать вообще без внешнего конденсатора, рекомендуется применять конденсатор с емкостью порядка 20 пФ для увеличения стабильности и помехоустойчивости генератора.

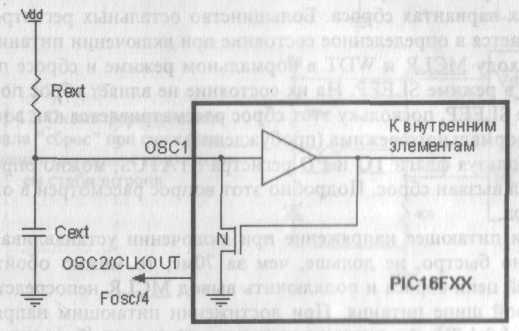


Рисунок 3.7 - Схема внешнею RC-генератора

Частота тактового генератора зависит от мигающего напряжения, номиналов резистора и конденсатора и варьируется от кристалла к кристаллу.

В режиме RC с вывода OSC2/CLKOUT можно снимать импульсы с частотой одна четвертая от тактовой и использовать эти импульсы для тактирования или синхронизации остальной схемы.

Когда контроллер настроен в режим RC, на его вывод OSC1/CLKIN нельзя подавать импульсы от внешнего генератора, так как можно вывести кристалл из строя.

Организация сброса.

Для PIC16F84 доступны следующие пять вариантов сброса:

* Сброс по включению питания
* Сброс по входу MCLR во время нормальной работы
* Сброс по входу MCLR в режиме SLEEP
* Сброс по переполнению сторожевого таймера (WDT) во время нормальной работы
* Сброс по переполнению сторожевого таймера (WDT) в режиме SLEEP

Если питающее напряжение при включении устанавливается достаточно быстро, не дольше, чем за 70мс, то можно обойтись без внешней цепи сброса и подключить вывод MCLR непосредственно к плюсовой шине питания. При достижении питающим напряжением уровня 1.2-1.7V сформируется внутренний сигнал сброса и начнется отсчет времени задержки сброса специальным внутренним таймером PWRT (Power-up timer). За это время питающее напряжение должно подняться до нормального рабочего уровня. Таймер PWRT работает от независимого встроенного RC-генератора, время задержки равняется примерно 72мс и может несколько изменяться от кристалла к кристаллу, а также в зависимости от температуры. После окончания задержки таймера PWRT включается таймер запуска основного тактового генератора, но он тактируется непосредственно от этого генератора и отсчитывает 1024 импульса.

Таймер PWRT может быть включен или выключен изменением бита PWRTE в слове конфигурации.

Если питающее напряжение нарастает медленно, может понадобиться внешняя цепь сброса.

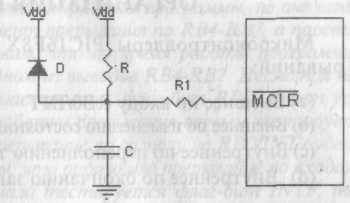


Рисунок 3.8 - Внешняя цепь сброса

Организация прерываний

Микроконтроллеры PIC16F84 имеют четыре источника прерываний:

* Внешнее по входу RB0/INT
* Внешнее по изменению состояния одной из линий RB4-RB7 порта В
* Внутреннее по переполнению таймера/счетчика TMR0
* Внутреннее по окончанию записи в EEPROM.

Для каждого из прерываний 1, 2 и 3 существует свой флаг-бит в регистре INTCON, сигнализирующий о поступлении конкретного вида запроса. Для прерывания 4 флаг-бит хранится в регистре EECON1. Анализируя эти биты подпрограмма обработки прерываний, начинающаяся с адреса 0004h, определяет источник прерывания. Кроме этого, в регистре INTCON содержится бит глобального запрета прерываний GIE и биты индивидуального запрета каждого прерывания. Когда бит GIE сброшен, все прерывания запрещены. При включении питания бит GIE по умолчанию сброшен.

Когда поступает прерывание, бит GIE сбрасывается, чтобы не допустить возникновения нового прерывания (поскольку у микроконтроллера PIC16F84 существует только один вектор прерывания), адрес возврата загружается в стек, а в программный счетчик загружается адрес-вектор 0004h. Инструкция возврата из подпрограммы обработки прерывания RETFIE устанавливает бит GIE в единицу, тем самым разрешая прерывания.

Обработка запроса внешнего прерывания может занимать три или четыре командных цикла процессора, в зависимости от того, в какой момент времени обнаружен запрос.

Все флаг-биты прерываний должны быть сброшены программно до того, как прерывания вновь будут разрешены установкой бита GIE. В противном случае может произойти повторный вход в прерывание и зацикливание программы. Также следует помнить, что флаг-биты прерываний устанавливаются в соответствии с прерывающими событиями независимо от того, замаскированы эти прерывания, или нет. Игнорирование этих фактов приводит к достаточно распространенным ошибкам программирования, которые могут проявить себя не сразу.

Энергосберегающий режим SLEEP.

Микроконтроллер переводится в режим SLEEP при выполнении специальной команды SLEEP. Если сторожевой таймер включен, то он обнуляется и начинает отсчет задержки заново. В регистре STATUS сбрасывается бит PD и устанавливается бит ТО. Тактовый генератор отключается. Выводы портов сохраняют состояние, которое было непосредственно перед исполнением команды SLEEP.

Пробуждение из режима SLEEP.

Процессор может быть выведен из режима SLEEP тремя различными способами:

* внешним сбросом по входу MCLR;
* при переполнении сторожевого таймера (если он включен);
* прерыванием по входу RB0/INT, по изменению состояния входов RB4-RB7 или по окончанию записи в EEPROM.

Событие 1 приводит к сбросу процессора и исполнению программы с начального адреса. Два остальных события приводят к продолжению исполнения программы. Во время исполнения команды SLEEP процессор загружает в буфер команд следующую команду (РС+1). Чтобы контроллер вышел из режима SLEEP по прерыванию, оно должно быть разрешено соответствующими битами.

Порядок пробуждения по прерыванию зависит от состояния бита GIE. Если этот бит сброшен, то после пробуждения выполняется команда, следующая за командой SLEEP (и уже загруженная в буфер) и далее по порядку. Если бит GIE установлен в 1, то сначала исполняется команда, загруженная в буфер, а затем процессор переходит на адрес-вектор прерывания 0004h. Если исполнение команды, следующей за SLEEP, при выходе по прерыванию нежелательно, то сразу после команды SLEEP должна следовать команда NOP.

Система команд

Каждая команда контроллера PIC16F84 представляет собой 14-битное слово, состоящее из кода команды (OPCODE) и одного или нескольких операндов.

f - Адрес специального регистра или регистра пользователя. Диапазон значений от 0x00 до 0x7F. Фирменный ассемблер допускает применять вместо цифровых значений непосредственно имена регистров, определенные ранее.

W - Рабочий регистр (аккумулятор)

b - Битовый адрес, используемый с 8-битным регистром, и указывающий внутри регистра на бит, с которым выполняется битовая операция. В ассемблерном тексте обозначает константу, представленную в двоичном счислении.

k - Литерал, константа или метка.

d - Указатель приемника результата операции. Если d=0, результат сохраняется в W, если d= 1, результат сохраняется в регистре, объявленном в команде. По умолчанию d=l. Никакой другой регистр, кроме используемого в операции, не может быть назначен приемником. Фирменный ассемблер для большей наглядности допускает применять вместо значений 0 и 1 соответственно символы w и f.

Таблица 3.6 - Система команд микроконтроллера PIC16F84

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Мнемоника  операнды | | Расшифровка мнемоники | Количество  циклов | | Изменяемые  регистры | | Примечание | |
| БАЙТОВЫЕ ОПЕРАЦИИ С РЕГИСТРАМИ | | | | | | | | |
| ADDWF f, d | | Add W and F | 1 | | C, DC, Z | | 1,2 | |
| ANDWF f, d | | AND W with F | 1 | | Z | | 1,2 | |
| CLRF f | | Clear F | 1 | | Z | | 2 | |
| CLRW | | Clear W | 1 | | Z | |  | |
| СОМF f, d | | Complement F | 1 | | Z | | 1,2 | |
| DECF f, d | | Decrement F | 1 | | Z | | 1,2 | |
| DECFSZ f, d | | Decrement F, Skip if Zero | 1(2) | |  | | 1,2,3 | |
| INCF f, d | | Increment F | 1 | | Z | | 1,2 | |
| INCFSZ f, d | | Increment F, Skip if Zero | 1(2) | |  | | 1,2,3 | |
| IORWF f, d | | Inclusive OR W with F | 1 | | Z | | 1,2 | |
| MOVF f, d | | Move F | 1 | | Z | | 1,2 | |
| MOVWF f | | Move W to F | 1 | |  | |  | |
| NOP | | No Operation | 1 | |  | |  | |
| RLF f,d | | Rotate Left F through carry | 1 | | C | | 1,2 | |
| RRF f,d | | Rotate Right through carry | 1 | | C | | 1,2 | |
| SUBWF f, d | | Subtract W from F | 1 | | C, DC, Z | | 1,2 | |
| SWAPF f, d | | Swap nibbles in F | 1 | |  | | 1,2 | |
| XORWF f, d | | Exclusive OR W with F | 1 | | Z | | 1,2 | |
| Мнемоника  операнды | | Расшифровка мнемоники | Количество  циклов | | Изменяемые  регистры | | Примечание | |
| БИТОВЫЕ ОПЕРАЦИИ С РЕГИСТРАМИ | | | | | | | | |
| BCF f, b | | Bit Clear F | 1 | |  | | 1,2 | |
| BSF f, b | | Bit Set F | 1 | |  | | 1,2 | |
| BTFSC f, b | | Bit Test F, Skip if Clear | 1(2) | |  | | 3 | |
| BTFSS f, b | | Bit Test F, Skip if Set | 1(2) | |  | | 3 | |
| ОПЕРАЦИИ С ЛИТЕРАЛАМИ И ПЕРЕХОДЫ | | | | | | | | |
| ADDLW k | Add Literal and W | | 1 | C, DC, Z | |  | |
| ANDLW k | AND Literal with W | | 1 | Z | |  | |
| CALL k | Call subroutine | | 2 |  | |  | |
| CLRWDT | Clear Watchdog Timer | | 1 | TO, PD | |  | |
| GOTO k | Go To address | | 2 |  | |  | |
| IORLW k | Inclusive OR Literal with W | | 1 | Z | |  | |
| MOVLW k | Move Literal to W | | 1 |  | |  | |
| RETFIE | Return From Interrupt | | 2 |  | |  | |
| RETLW k | Return with Literal in W | | 2 |  | |  | |
| RETURN | Return from subroutine | | 2 |  | |  | |
| SLEEP | Sleep into standby mode | | 1 | TO, PD | |  | |
| SUBLW k | Subtract W from Literal | | 1 | C, DC, Z | |  | |
| XORLW k | Exclusive OR Literal with W | | 1 | Z | |  | |

Примечания:

1) Когда происходит чтение-модификация-запись портов ввода-вывода, то всегда считывается реальные значения напряжений на выводах, независимо от того, как настроены линии порта и что записано в триггер-защелку. Операция производится над считанным реальным значением и результат записывается в защелку.

2) Когда эта инструкция выполняется над регистром TMR0 и d= 1, предделитель сбрасывается, если подключен к модулю Timer0.

3) Если программный счетчик модифицируется или проверка на условие возвращает истинный результат, инструкция требует два машинных цикла. Второй цикл исполняется процессором как NOP.

3.6.2 Микросхема КР142ЕН5А

Микросхема КР142ЕН5А трехвыводный стабилизатор с фиксированным выходным напряжением 5 вольт могут найти применение в широком спектре радиоэлектронных устройств в качестве источниках питания логических систем, измерительной технике, устройств высококачественного воспроизведения и других радиоэлектронных устройств.

Внешние компоненты могут быть использованны для ускарения переходных процессов. Входной конденсатор необходим только в том случае, если регулятор находиться на растоянии более 5 см от фильтрующего конденсатора источника питания.Основные характеристики:

-Допустимый выходной ток 1А

-Не требуются внешние компоненты

-Внутренняя термозащита

-Защита выходного транзистора

-Внутреннее ограничение тока КЗ

Таблица 3.7 - Электрические характеристики КР142ЕН5А

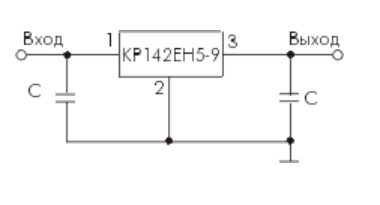
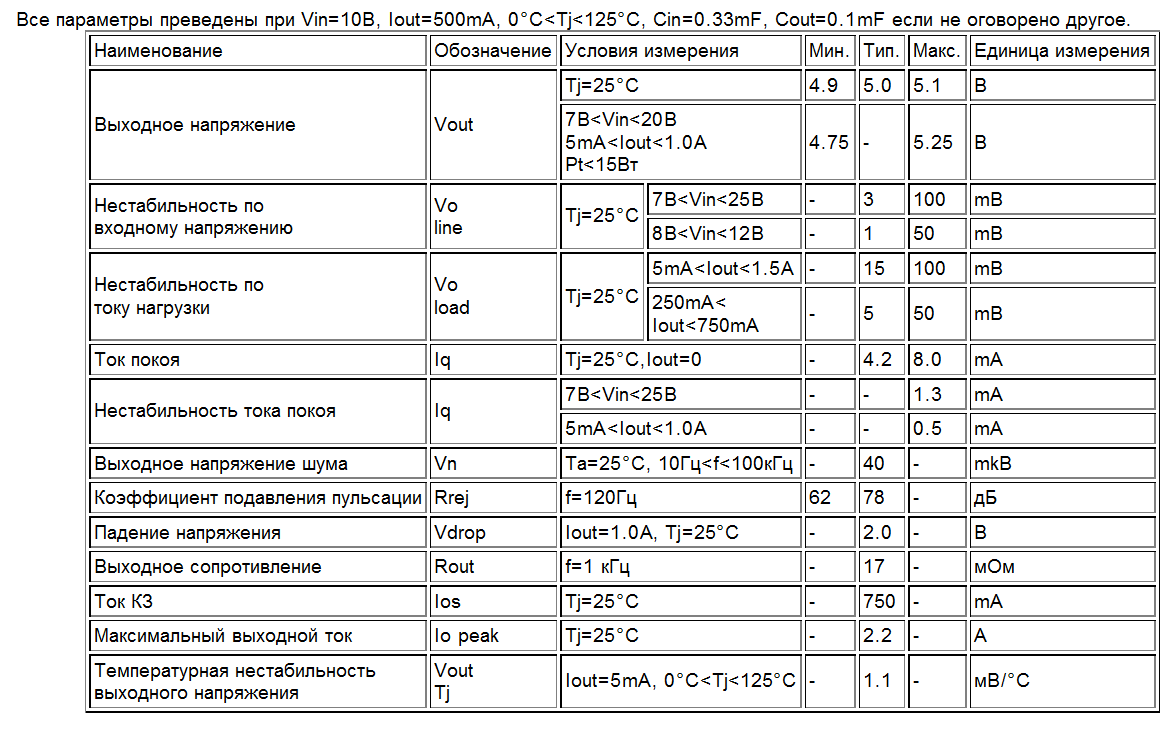


Рис. 3.9 - Схема включения КР142ЕН5А

4 Описание схемы электрической принципиальной

Схема тахометра показана на рис. . Входной сигнал с контактов прерывателя стабилизируется стабилитроном VD1 на уровень ТТЛ и подается на вход RB0. При бесконтактной системе зажигания сигнал снимается с выхода коммутатора, выдающего перепад напряжения 3 В. Этого напряжения достаточно для срабатывания микроконтроллера.

Входы RA0-RA2 коммутируют аноды светодиодов, реализуя динамическую индикацию. Вход RA3 нагружен переключателем предела измерения

«Is—3s». Внутренние подтягивающие резисторы программно отключены, но ими снабжены только входы «В» микроконтроллера, поэтому возникла необходимость в установке резистора R2. Входы RBI—RB7 использованы для вывода значений сегментов. Поскольку микроконтроллер работает при верхнем питающем напряжении 6 В, то микросхему стабилизатора напряжения КР142ЕН5 можно взять с любой буквой, обеспечивающей это напряжение.

Потребляемый тахометром ток около 25 мА, поэтому микросхема стабилизатора напряжения не нуждается в радиаторе. Диод VD2 защищает прибор от переполюсовки. Если тахометр будет использоваться при регулировке карбюраторов, то светодиоды АЛ304Г нежелательно заменять индикаторами с большими размерами цифр.

5 Алгоритм работы устройства и программное обеспечение

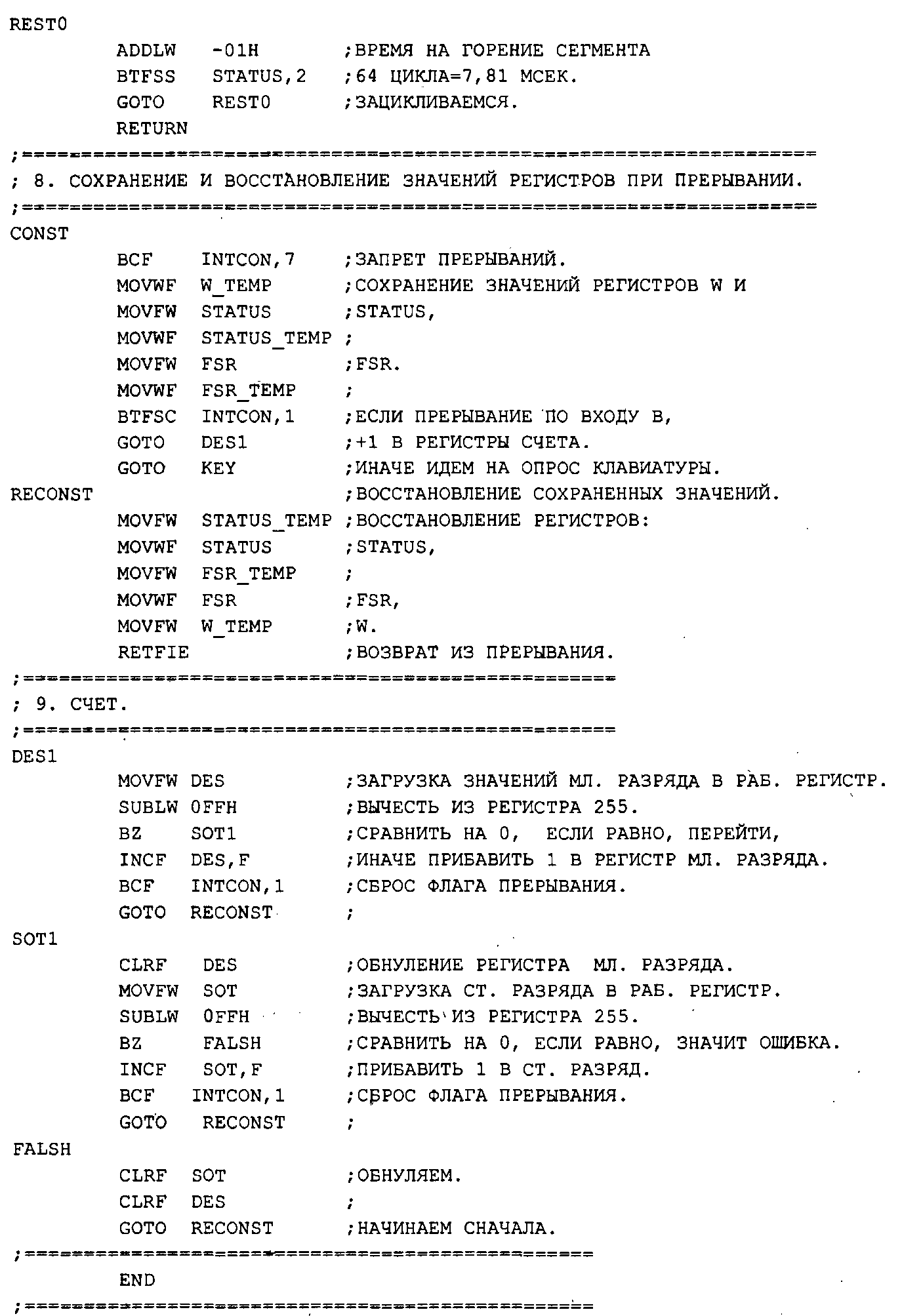
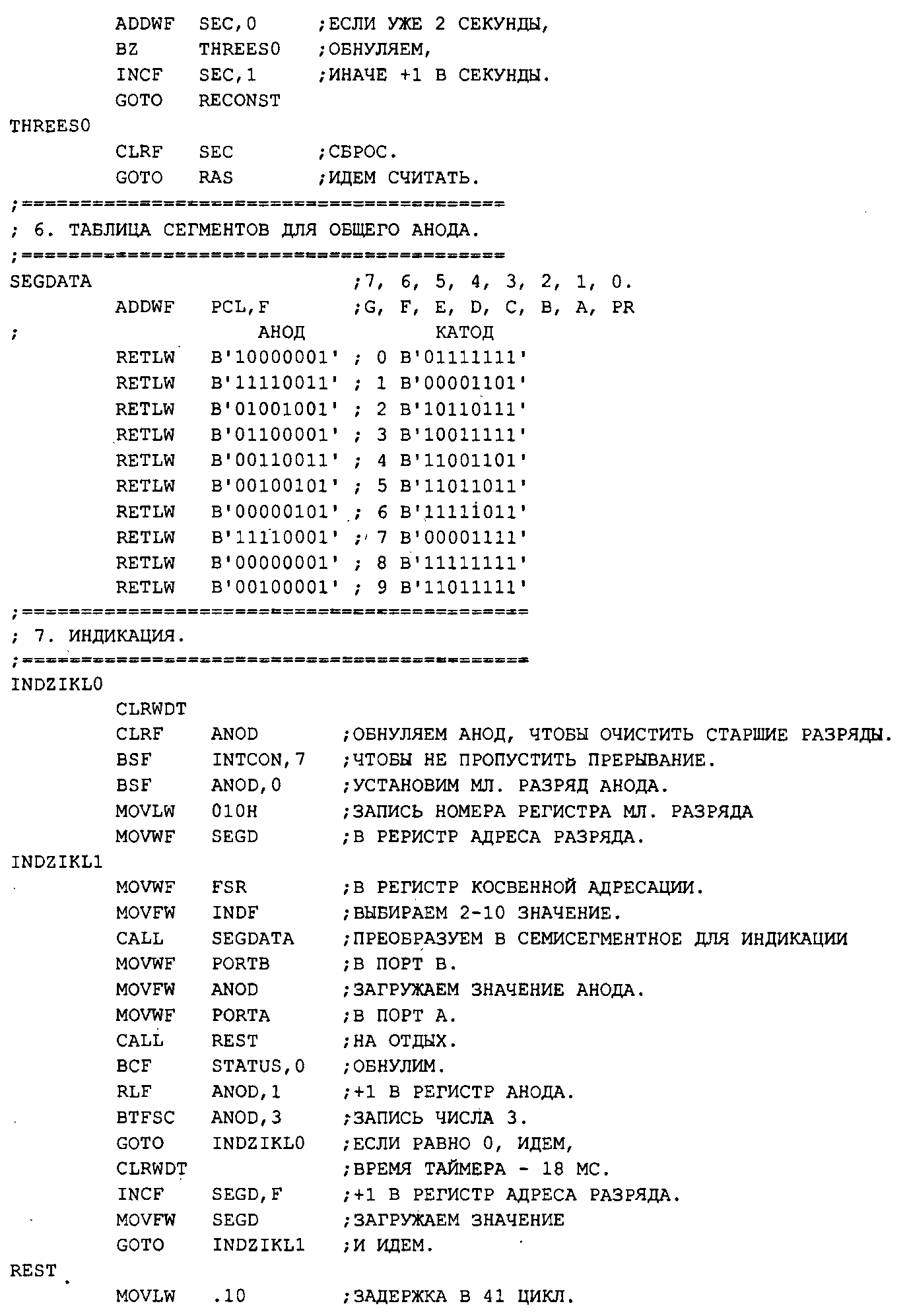
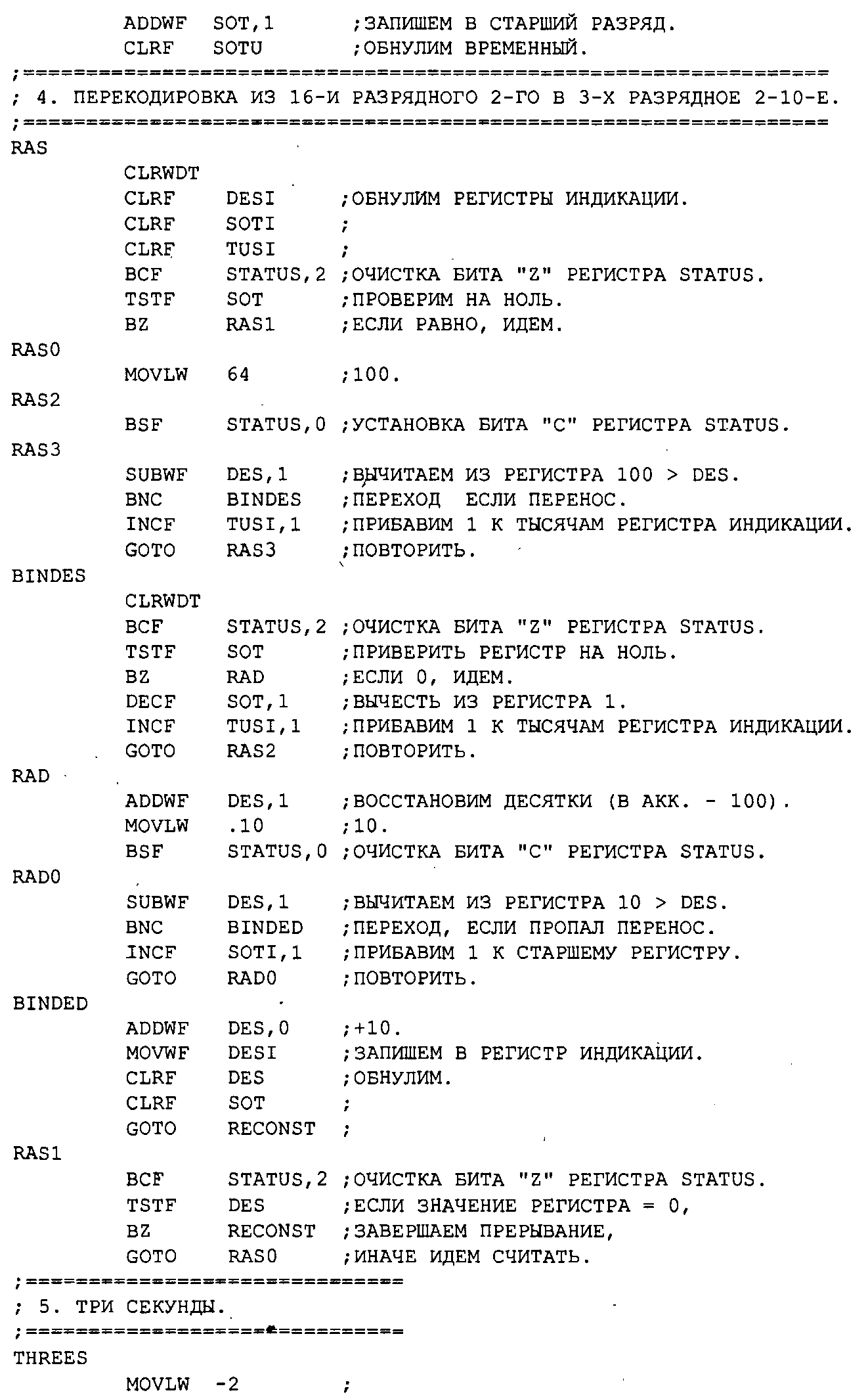
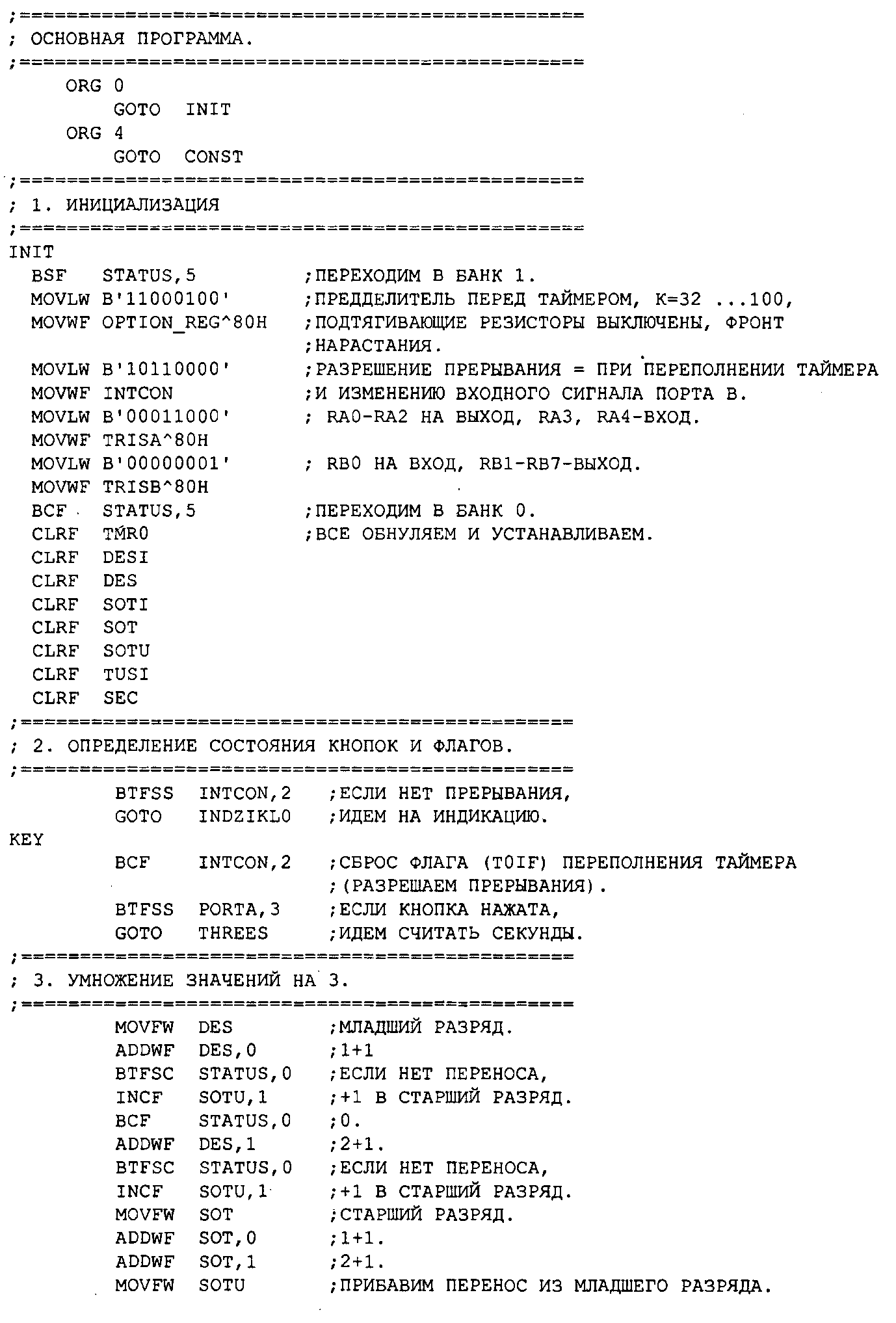
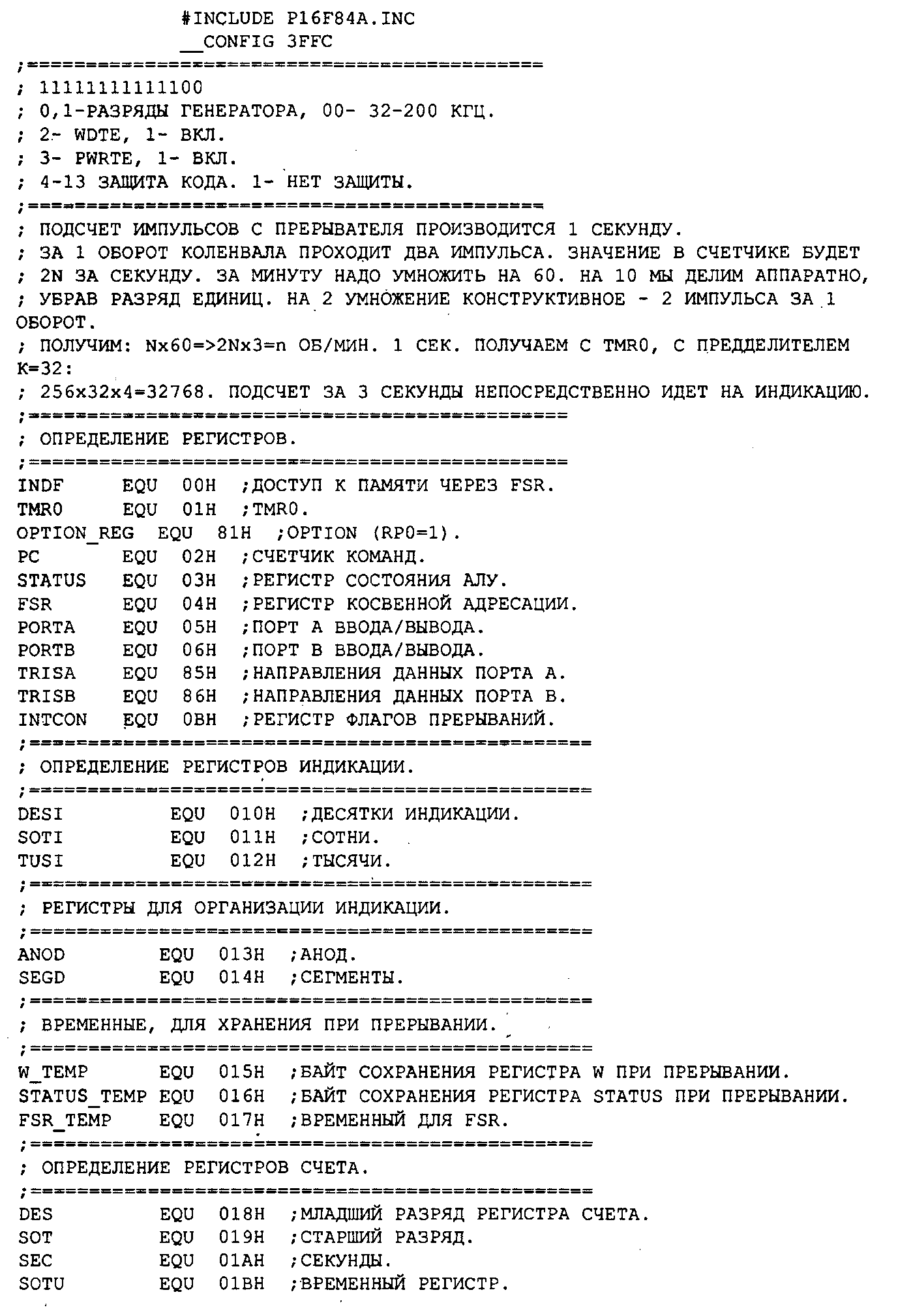
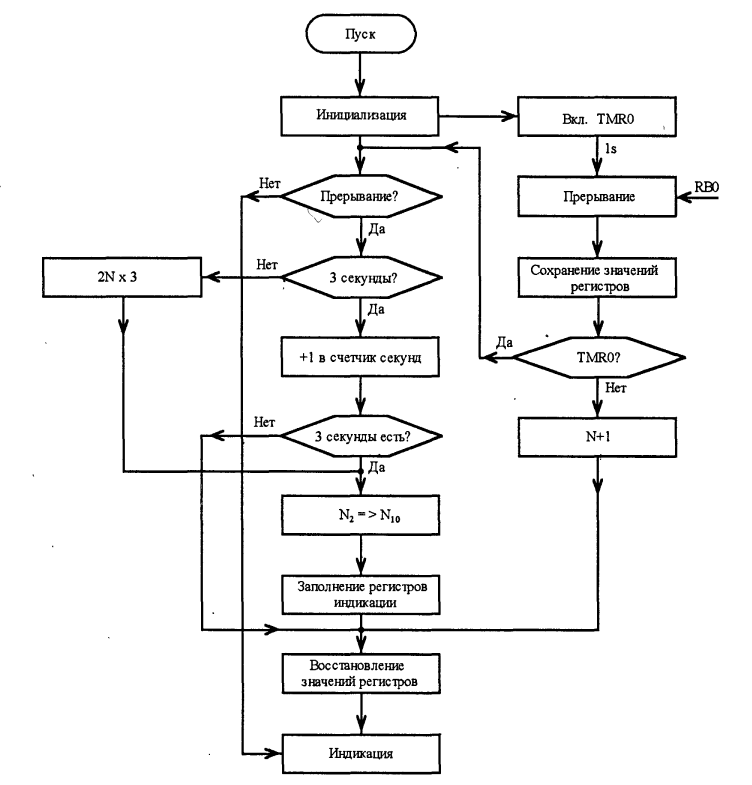
Тахометр реализован на одной микросхеме микроконтроллера PIC16F84A.

Алгоритм программы тахометра представлен на рис. 16. После включения питания происходит начальная инициализация всех регистров с последующей индикацией. После инициализации вступает в работу таймер TMR0. Таймер имеет коэффициент деления, равный 256, что вместе с предцелителем, имеющим коэффициент деления, равный 32, и циклом процессора, равным 4, дает прерывания каждую секунду (4 х 32 х 256 = 3,2768).

При замыкании контактов прерывателя с входа RB0 также происходит прерывание. При прерывании сохраняются значения регистров, задействованных на момент прерывания, и определяется происхождение прерывания. Если прерывание с входа RB0, то двоичный 16-разрядный счетчик увеличивается на единицу. Таким образом, подсчитывается количество прерываний с входа RB0 между прерываниями от переполнения таймера, то есть за 1 секунду. Каждое прерывание заканчивается восстановлением ранее сохраненных значений регистров, и процессор переключается на работу с индикацией.

Если прерывание произошло по переполнению таймера, то определяется состояние переключателя предела измерения и, если переключатель на пределе одной секунды, двоичное значение 16-разрядного счетчика умножается на 3 (2N х 3). 16-разрядный счетчик обнуляется, готовясь к новому циклу измерения. Полученное двоичное значение перекодируется в трехразрядное двоично-десятичное число и переписывается в регистры индикации. После восстановления значений регистров индикация происходит с новыми данными. То есть индикация обновляется каждую секунду.

Если установлен предел измерения, равный 3 секундам, то при переполнении таймера значение счетчика секунд увеличивается на единицу. Если значение счетчика секунд еще не равно трем, прерывание завершается без обнуления 16-разрядного счетчика. В противном случае в 16-разрядном счетчике накапливается количество прерываний с входа RB0 за три секунды. Это значение перекодируется в двоично-десятичное число и переписывается в регистры индикации. Двоичный счетчик обнуляется и цикл повторяется. В данном случае индикация обновляется каждые три секунды.



Заключение

В ходе выполнения курсовой работы был разработан «Цифровой автомобильный тахометр». В пояснительной записки представлены схемы: электрическая структурная, электрическая функциональная, электрическая принципиальная с описанием. Также приведено описание всей элементной базы устройства, приведено описание работы пользователя с устройством.

Применение микроконтроллера PIC16F84 позволило создать компактное, эргономичное устройство с минимальными массогабаритными и экономическими показателями.

Прошивка микроконтроллера приведена в приложении к пояснительной записки.

В графической части представлены чертежи схем: электрической структурной и электрической принципиальной в соответствии со стандартами ЕСКД.

Список использованных источников

1. www.microchip.ru;
2. Радиолюбительские конструкции на PIC-микроконтроллерах кн.1(Заец Н.) 2003;
3. www.integral.by.

Ссылки для скачивания чертежеи:

http://depositfiles.com/files/otnsonrcc

http://depositfiles.com/files/ry5uz2l9v

http://depositfiles.com/files/gohzre6rd

http://depositfiles.com/files/fusrx4ov8

http://depositfiles.com/files/e7f9n2rx2

http://depositfiles.com/files/gacx7bz15

http://depositfiles.com/files/q9fphx1br

http://depositfiles.com/files/2t07phfob

http://depositfiles.com/files/rfrlwdeit