**РЕФЕРАТ**

Цель работы: разработка микропроцессорной системы на базе микроконтороллера, пожарной сигнализации на 11 пожарных датчиков. При срабатывании одного из датчиков пожарной сигнализации по радиоканалу поступает кодированный сигнал, чем обеспечивает включение сигнала и указывает номер сработавшего датчика.

Содержание работы: в работе выполнено построение структурной схемы, построение функциональной схемы, сформирован алгоритм работы системы, выбор элементной базы, оптимальной для реализации поставленных задач по диапазону характеристик, разработана программа, разработана принципиальная схема устройства.

**СОДЕРЖАНИЕ**

Введение

1. Описание объекта и функциональная спецификация

2. Описание ресурсов МК

2.1 Расположение выводов

2.2 Исполнение микроконтроллера

2.3 Особенности микроконтроллеров серии PIC16F Х

2.4 Периферийные особенности, специальные особенности, технология

2.5 Эксплуатационные характеристики

3. Разработка алгоритмов устройства

4. Ассемблирование

5. Описание функциональных узлов МПС и алгоритма их взаимодействия

6. Описание выбора элементной базы и работы принципиальной схемы

Заключение

Список литературы

Приложение А

Приложение Б

Приложение В

Приложение Г

Приложение Д

**ВВЕДЕНИЕ**

В последнее время в нашей стране, да и, пожалуй, во всём мире стала наиболее актуальна проблема безопасности объекта. Системы электронных пожарных сигнализаций являются одним из главных препятствий на пути пожара.

Современные интегрированные системы безопасности.

В связи с широким использованием современных электронных компонентов и цифровых методов обработки информации происходит существенная "интеллектуализация" технических средств. Эти средства перестают быть просто вспомогательными и приобретают новые свойства. Современные технические средства охраны могут быть использованы в виде полностью интегрированной системы или системы, состоящей из функционально независимых компонентов.

Тенденции развития электронной техники и электротехники на базе микроминиатюризации требуют применения широкой номенклатуры маломощных и малогабаритных устройств и изделий (преобразователей трансформаторов, усилителей, фильтров, стабилизаторов, выпрямителей и так далее), выполненных на новой конструктивной основе ЭРЭ. Достижения науки и техники на современном этапе развития в области электронной техники позволяют значительно уменьшить массогабаритные характеристики рассматриваемых электронных устройств. В настоящее время конструирование РЭА, РЭУ и ЭРЭ характеризуется резким увеличением применения БИС, что также даёт возможность уменьшить объёмы устройств и одновременно улучшить их качественные характеристики, показатели надёжности и долговечности.

Пожарные системы большой ёмкости обычно применяются для обеспечения безопасности крупных предприятий, банков, гостиниц и характеризуется тем, что обслуживают от нескольких сотен до нескольких тысяч датчиков (извещателей). При этом контролировать нужно не только датчики но и различного вида устройства управляющие работой системы.

Микроконтроллеры серии РІС идеально подходят для организации и развития малого и среднего бизнеса в области производства электронной аппаратуры. При минимальных затратах рабочего времени и денежных средств можно легко наладить производство мини-АТС, телефонных блокираторов, "интеллектуальных" датчиков сигнализации, систем контроля доступа, автомобильной электроники и т п. Причем в таких изделиях от 30 до 90 процентов функциональной нагрузки несет на себе программное обеспечение, которое может быть легко модифицировано и приспособлено к нуждам потребителя.

Немаловажное значение имеет возможность защиты кода программы от несанкционированного копирования или изменения. Наличие этой опции эффективно защищает права разработчика и производителя, и особенно важно в Украине, где эти права систематически нарушаются. Для радиолюбителей микроконтроллеры РІС также интересны доступностью цены и простотой в освоении и применении, а также тем, что открывают для них новые, ранее невиданные области творчества. Радиолюбитель освобождается от непроизводительного труда, направленного на подбор и поиск подходящих компонентов, разработку сложной схемотехники реализованной на микросхемах жесткой логики. Значительно упрощается сама конструкция устройства и повышается надежность.

В данном курсовом проекте реализована микропроцессорная система на базе микроконтроллера для управления пожарной сигнализацией. Проект основывается на микроконтроллере PIC16F84А.

**1. Описание объекта и функциональная спецификация**

Предлагаемое устройство - модулятор и демодулятор на двух микроконтроллерах может работать совместно с передатчиком и приемником с частотной или амплитудной модуляцией. Устройство контролирует ежесекундную додачу импульсов с датчиков пожарной сигнализации. Производится контроль состояния 11 датчиков. При пропадании контрольных импульсов из-за значительного удаления от объекта или выключения передатчика включается звуковой сигнал. После включения тревожного сигнала можно определить, по какому из параметров или датчиков включилась сигнализация.

Функциональная спецификация

Так как устройство содержит две законченные конструкции, разберем каждую конструкции отдельно:

Модулятор

1. Входы

а. 11 пожарных датчиков на размыкание

b. кнопка запуска (включение питания)

2. Выходы

а. к модулятору передатчика

b. включение несущей

3. Функции

а. сбор информации от пожарных датчиков (извещателей)

b. включение несущей передатчика

с. передача сигналов на модулятор передатчика

d. обработка временных характеристик работы устройства и передача в частотном виде

Демодулятор

1. Вход

а. с компаратора приемника

b. кнопка запуска (включение питания)

2. Выходы

а. звуковой излучатель

b. семисегментный индикатор

3. Функции

а. прием управляющих сигналов с компаратора приемника

b. обработка полученной информации

c. подача звукового сигнала «Тревога» на звуковой излучатель

d. вывод информации на семисегментный индикатор, указывающий на объект срабатывания сигнализации и работоспособности устройства

1. **Описание ресурсов МК**

**2.1 Расположение выводов**

Расположение выводов микроконтроллера РIС16F84A изображено на рис. 1.



Рис. 1 -Расположение выводов микроконтроллера РIС16F84A

**2.2 Исполнение микроконтроллера**

Микроконтроллер выпускается в двух видах корпусов.

Расположение выводов и конструктивные размеры различных корпусов приведены на Рис. 2 и Рис.3.

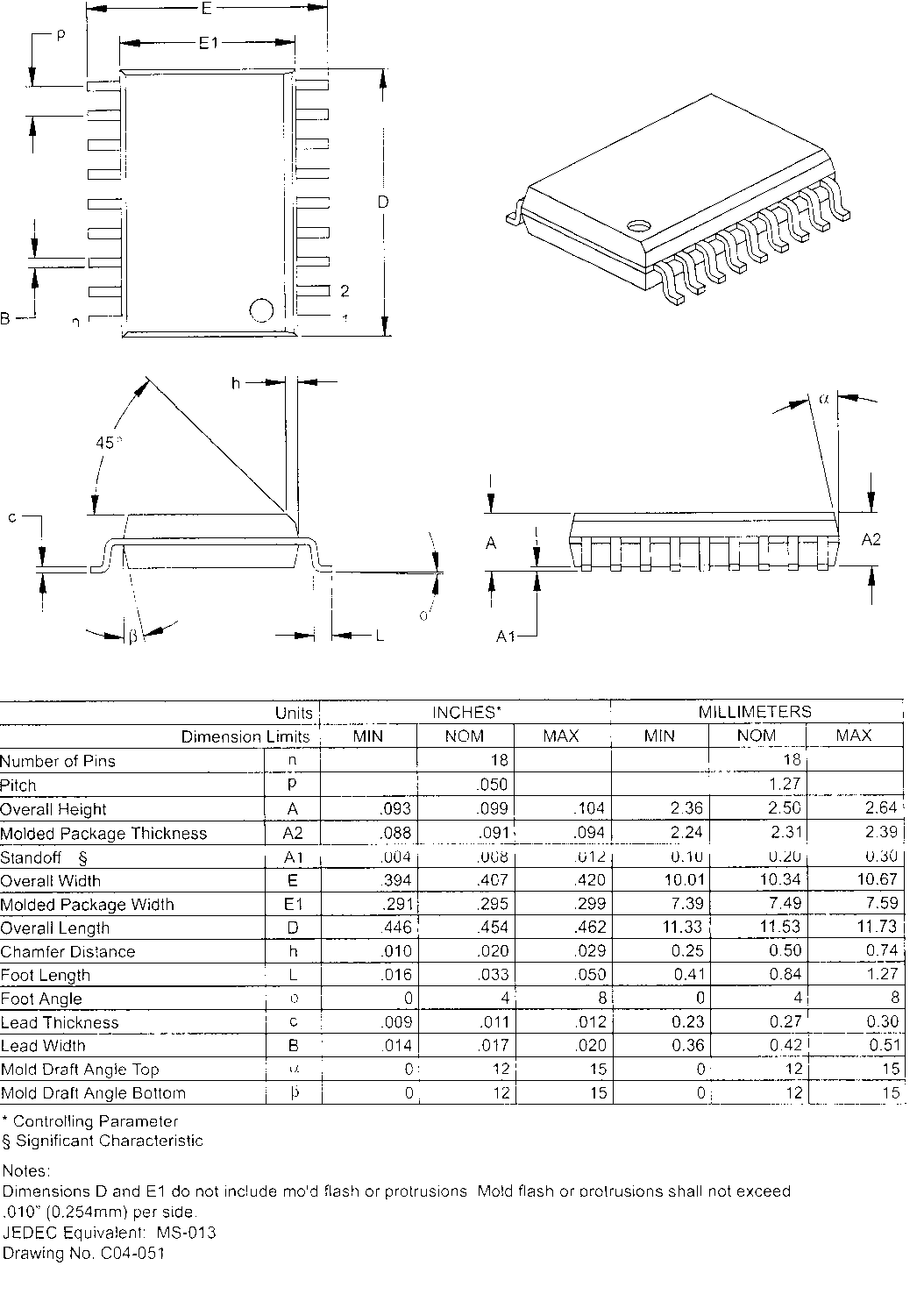


Рис. 2 - Конструктивные размеры микроконтроллера РIС16F84A (исполнение 1)

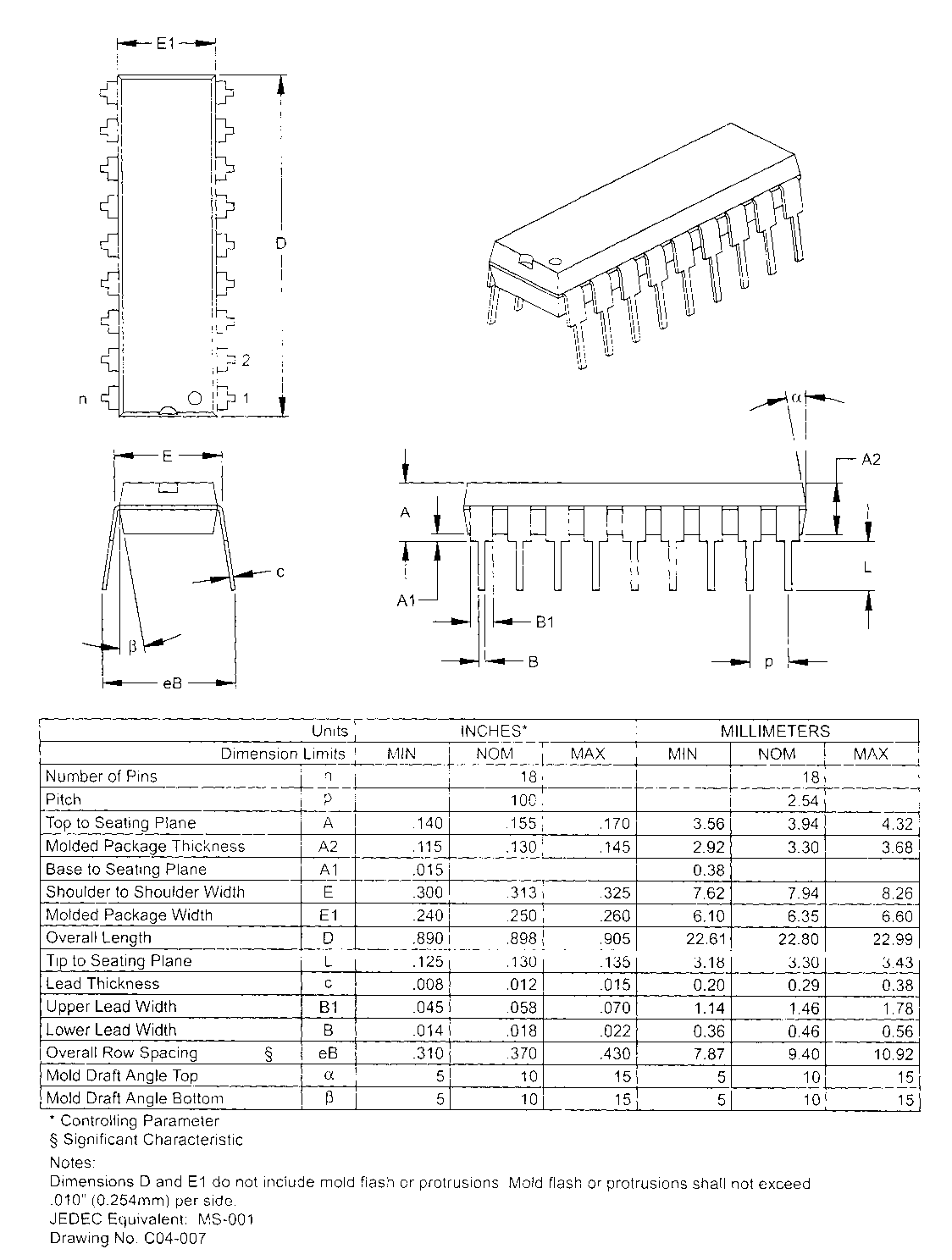


Рис. 3 - Конструктивные размеры микроконтроллера РIС16F84A (исполнение 2)

**2.3 Особенности микроконтроллеров серии PIC16F**

Особенности CPU Высокой Производительности RISC:

• Только 35 единых команд операции над словами для узнавания

• Весь единый цикл инструкций за исключением программы ветви, которые являются два-циклом

• Действие скорости: DC - 20 отмечающего время входа Mгц DC - 200 цикла инструкции ns

• 1024 слова памяти программы

• 68 байтов Оперативной ПАМЯТИ Данных

• 64 байта Данных EEPROM

• 14-разрядные широкие команды

• 8-разрядные широкие байты данных

• 15 Специальных Аппаратных записей Функции

• Восьми-горизонтальный глубокий аппаратный стек

• Направьте, косвенные и относительные способы адресации

• Четыре источника прерывания:

- Внешняя иголка RB0/INT

- Избыток таймера TMR0

- PORTB<7:4> включенное изменение прерывания

- Запись данных EEPROM завершить

**2.4 Периферийные особенности, специальные особенности, технология**

Периферийные Особенности

• 13 входов I/O с индивидуальным контролем направления

• Высокая текущая сточный выход/источник для прямого выхода

- 25 max сточного выхода mA. за направление

- 25 max источника mA. за направление

• TMR0: 8-разрядный таймер/счетчик с 8-куском программируемый prescaler

Специальные Особенности Microcontroller:

• 10,000 стереть/написать Увеличенную ВСПЫШКУ циклов

Типичная память программы

• 10,000,000 типично стереть/написать циклы EEPROM

Типичная память данных

• Сдерживание Данных EEPROM > 40 лет

• Включенная длина окружности Серийный Programming™ (ICSP™) – через два входа

• Энергетический Включенный Reset (POR), Энергетический Верхний Таймер (PWRT)

Таймер Старта Осциллятора (OST)

• Таймер (WDT) watchdog с собственным включенным Осколком RC

Осциллятор для надежного действия

• Защита кода

• Мощность, сохранение метода СНА

• Выбираемые варианты осциллятора

CMOS Увеличенный FLASH/EEPROM

Технология:

• Низкая мощность, технология большой скорости

• Полностью неподвижный проект

• Повсюду ряд операционного напряжения:

- Коммерчески: 2.0V 5.5V

- Индустриально: 2.0V 5.5V

• Низкое энергетическое потребление:

- < 2 mA типично @ 5V, 4 мгц

- 15 типично @ 2V, 32 kHz

- < 0.5 типичных текущих резервирования 2V

**2.5 Эксплуатационные характеристики**

Температура окружающей среды под bias-55°C +125°C

Температура хранения -65°C +150°C

Напряжение на любой входе относительно VSS (кроме того VDD, MCLR, и RA4) -0.3V (VDD + 0.3V)

Напряжение на VDD относительно VSS -0.3 +7.5V

Напряжение на MCLR относительно VSS**(1)**. -0.3 +14V

Напряжение на RA4 относительно VSS -0.3 к +8.5V

Полное энергетическое потребление**(2)**.800 mW

Максимальный ток вне входа. 150 mA

Максимальный ток на входе VDD 100 mA

Входной текущий зажим, IIK (VI < 0 или VI > VDD) 20 mA

Выходной текущий зажим, IOK (VO < 0 или VO > VDD) 20 mA

Максимальный вывод, текущий ниже уровня чем угодно I/O pin. 25 mA

Максимальный вывод текущий sourced любой вход I/O 25 mA

Максимально текущий ниже уровняPORTA 80 mA

Максимальный текущий sourced PORTA 50 mA

Максимально текущий ниже уровня PORTB 150 mA

Максимальный текущий sourced PORTB 100 mA

Примечание 1:Значения напряжения ниже VSS на входе MCLR, индуктируя больше токи, чем 80 mA, может вызвать верхним пределом.

Поэтому, резистор серии 50-100W должен использоваться при обращении “низкого уровня” ко входу MCLR скорее, чем вдергивание этого входа непосредственно к VSS

**3. Разработка алгоритмов устройства**

Алгоритм работы программы модулятора показан на рис. 4. После пуска и инициализации регистров программа переходит к постоянному контролю датчиков на размыкание. Контроль датчиков производится в то время, когда несущая включена. Это происходит через одну секунду. Если любой из 11 датчиков сработал, то номер этого датчика переписывается в регистр модуляции. Далее значение регистра модуляции сравнивается на ноль, и если оно не равно нулю, то регистр декрементируется. Устанавливается логическая единица на выходе RA3, включая тем самым модуляцию передатчика. Отработав паузу, равную 2,7 мс, выход устанавливается в нулевое состояние. Программа переходит на сравнение регистра модуляции на ноль. Таким образом, в то время, когда включена несущая, произойдет модуляция количеством импульсов, равным номеру сработавшего датчика.

Когда регистр модуляции обнулится, программа выключит несущую и установит флаг выключения несущей по срабатыванию датчика. Далее программа ожидает выключения флага несущей по переполнению таймера. Прерывание по переполнению таймера происходит либо из подпрограммы ожидания выключения флага несущей, либо, если датчик не сработал, из подпрограммы опроса датчиков.

При кварцевом резонаторе на частоту 32768 Гц, коэффициенте деления предделителя, равном 32, и коэффициенте деления таймера, равном 256, прерывание по переполнению таймера будет происходить каждую секунду. После сохранения регистров проверяется флаг включения несущей.Если **несущая** была включена, то проверяется флаг выключения несущей по срабатыванию датчиков. Если несущая выключена, то прерывание завершается восстановлением регистров. Если несущая не выключена по срабатыванию датчиков, формируется импульс модуляции, выключается несущая и устанавливается флаг выключения несущей по переполнению таймера. Прерывание завершается.

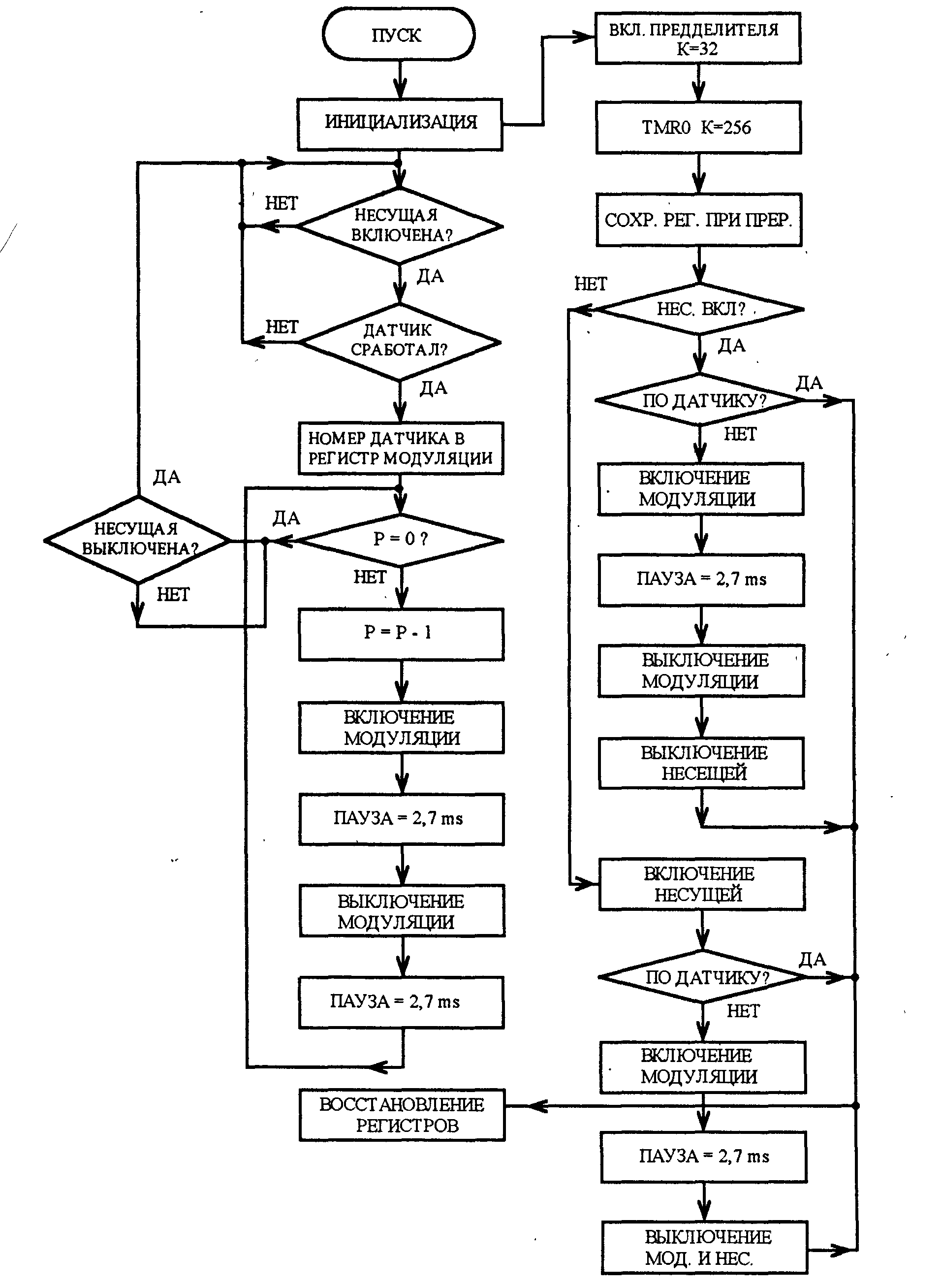


Рис. 4 - Алгоритм работы программы модулятора для пожарной сигнализации

Через секунду все повторится с включения несущей и формирования импульса модуляции, если не произошло выключение несущей по срабатыванию датчиков. Если датчики сработали, то импульс модуляции не формируется. Таким образом, если ни один датчик не разомкнут, каждую секунду будет формироваться импульс модуляции длительностью 2,7 мс. Импульс модуляции будет сформирован в центре импульса включения несущей. Длительность импульса включения несущей равна 8 миллисекундам. Если же какой-либо датчик сработал, то несущая включится на время формирования количества импульсов модуляции, равном номеру датчика. Если охранные датчики не сработали, то идет постоянный опрос датчиков и сброс флага выключения несущей по срабатыванию датчиков.

Алгоритм работы программы демодулятора показан на рис. 5. В демодуляторе используется два прерывания: по переполнению таймера и по изменению сигнала на входе порта «ВО». После пуска и инициализации программа выполняет постоянную индикацию. В дежурном режиме регистр индикации нулевой и все сегменты индикации будут иметь нулевое значение. Для семисегментного индикатора с общим катодом это равносильно выключению индикатора.

В дежурном режиме пожарной сигнализации с компаратора приемника будут поступать импульсы на вход RB0 с интервалом в одну секунду. С такой же частотой будет происходить прерывание по входу RB0. После сохранения значений регистров определяется, почему произошло прерывание-. Если прерывание произошло не по переполнению таймера, то проверяется длительность входного импульса. Если длительность импульса короче, чем половина длительности импульса посылки с передатчика (т.е. 1,34 мс), то такой импульс воспринимается как помеха и не подсчитывается счетчиком. Если длительность импульса больше 1,34 мс, то обнуляется таймер, определитель и счетчик миллисекунд. Если флаг переполнения включен (было переполнение), то инкрементируется регистр счета. Далее процессор проверяет значение регистра счета. Если в регистр записано 20 и более импульсов, то регистр обнуляется и включается звуковой сигнал. Ситуация, когда на счетчик приходит 20 импульсов, принята аварийной. Это возможно при работе пожарной сигнализации в зоне повышенных помех или помех, которые наводятся специально.

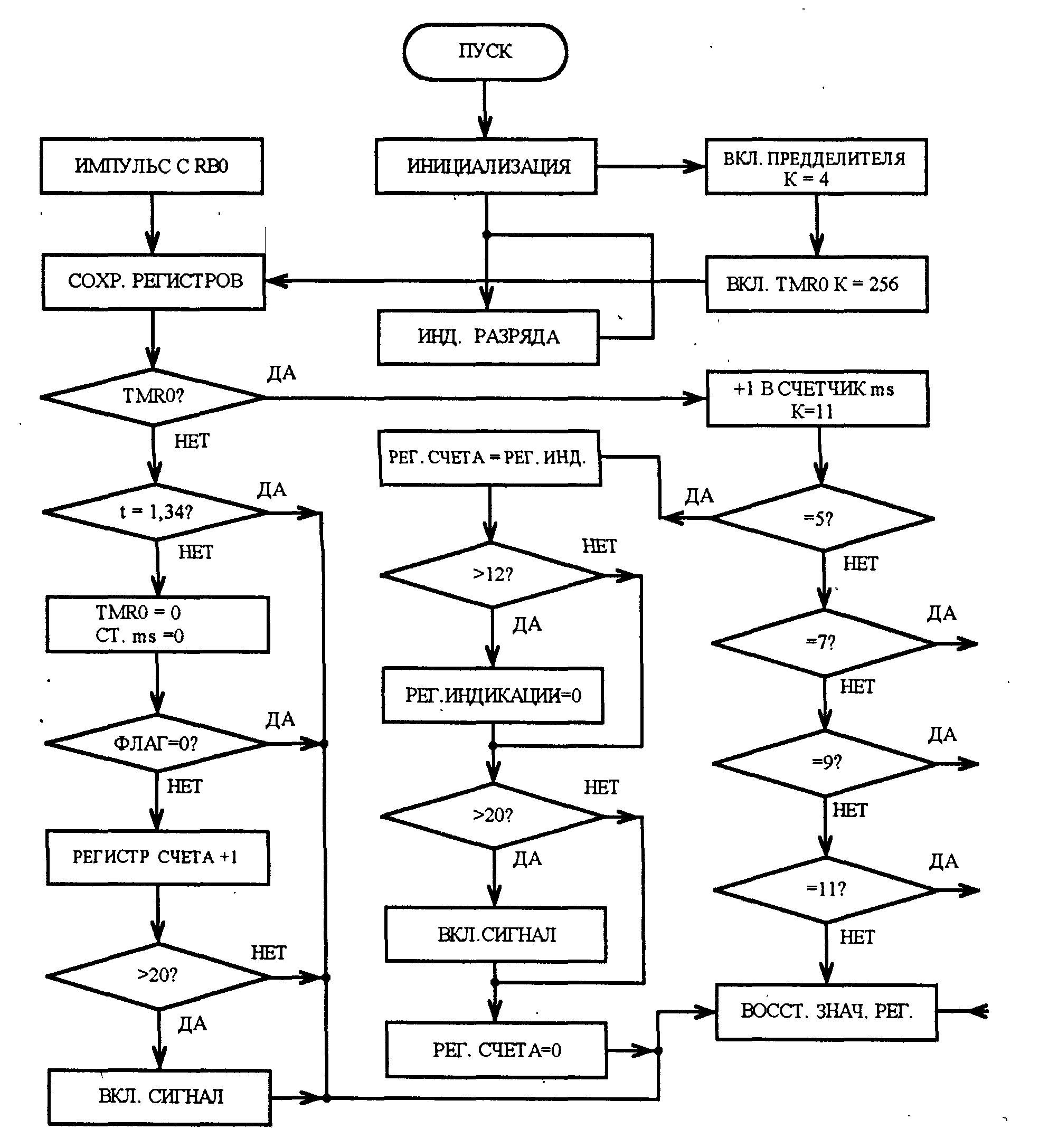


Рис. 5 - Алгоритм работы программы, демодулятора для пожарной сигнализации

Каждые 0,125 секунды может происходить прерывание от переполнения таймера. За одну секунду между импульсами, поступающими на вход RB0, произойдет 8 прерываний по переполнению таймера. При каждом прерывании будет инкрементироваться счетчик миллисекунд. Но поскольку коэффициент деления счетчика миллисекунд равен 11, то переполнения счетчика за одну секунду не будет. А каждая новая секунда будет начинаться с обнуления таймера и счетчика миллисекунд. При значении счетчика, равном 5 (0,6 с), переписывается значение регистра счета в регистр индикации. Это необходимо для того, чтобы при выходе передатчика из аварийного режима обнулялось предыдущее значение регистра счета. Далее проверяется значение регистра индикации, чтобы оно не превышало число 12. Если значение регистра индикации равно или больше 12, то регистр обнуляется. Последующая проверка регистра счета на число 20 является дублирующей аналогичную проверку при инкрементировании регистра счета.

При значении счетчика миллисекунд, равном 7 (0,88 с), выключается звуковой сигнал, если он был включен, и сбрасывается флаг переполнения (рис. 8.). Это также необходимо для нормального выхода из аварийного режима, поскольку эти операции (при значении счетчика 5 и 7) выполняются в интервале времени, равном 1 секунде.

Если после последнего импульса, пришедшего на вход, прошло более 1 секунды, точнее, 1,125, а это соответствует значению счетчика миллисекунд, равному 9, включится звуковой сигнал и установится флаг переполнения. Включение флага переполнения разрешает подсчет входных импульсов. А при каждом входном импульсе сбрасывается счетчик миллисекунд, таймер и предделитель. Поэтому значение счетчика миллисекунд, равное 11, возможно только тогда, когда приемник выходит из зоны действия передатчика или если передатчик прекратил функционирование. В этой ситуации счетчик миллисекунд обнуляется, а звуковой сигнал уже включен.

Таким образом, при поступлении импульсов на вход RB0 каждую секунду, что свидетельствует о нормальной работе охраны, сигнал тревоги включаться не будет, а цифровой индикатор не будет светиться. При размыкании любого из датчиков произойдет переполнение счетчика миллисекунд, включится сигнал, а на индикаторе можно наблюдать номер сработавшего датчика.

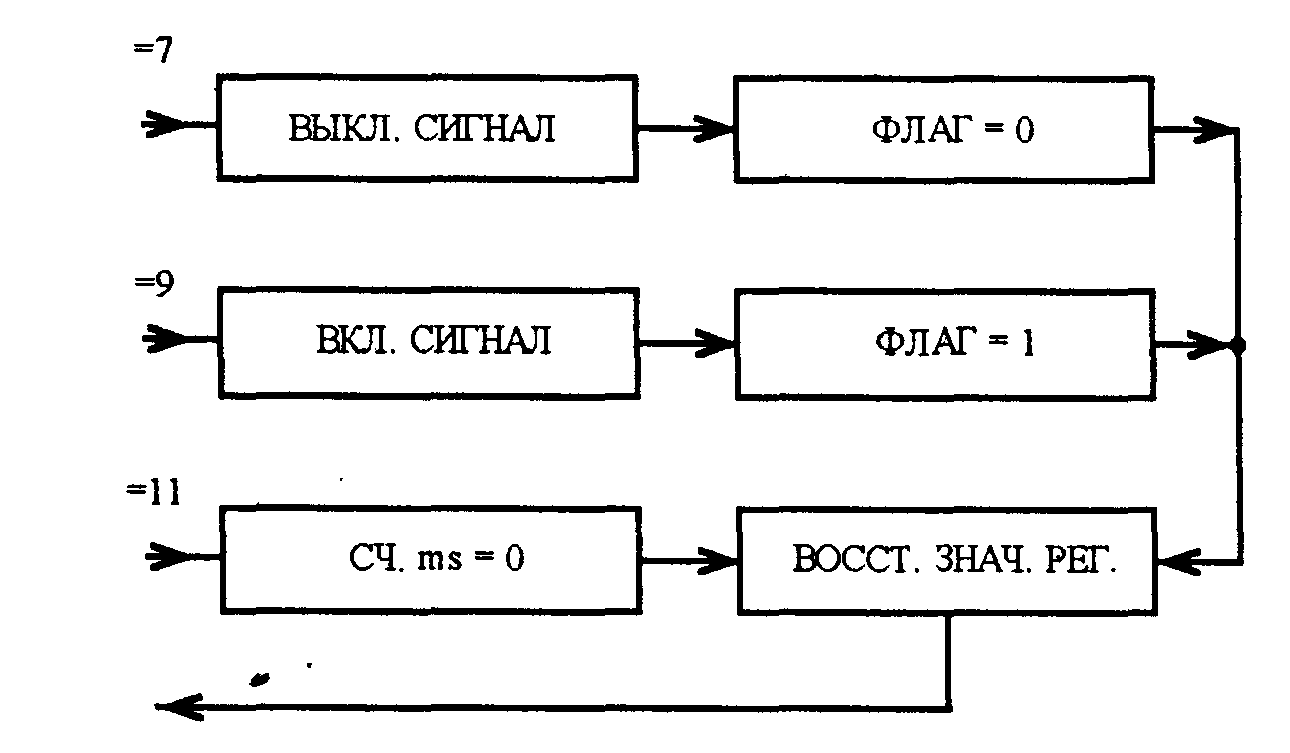


Рис. 6 - Алгоритм работы программы демодулятора для пожарной сигнализации

Для наглядности алгоритма работы программы демодулятора рассмотрим диаграмму, показанную на рис. 7 (для наглядности на диаграмме масштаб не соблюдается). В дежурном режиме каждое включение несущей передатчика сопровождается выдачей импульса модуляции. Импульс модуляции с выхода компаратора приемника будет поступать на демодулятор через одну секунду. В аварийном режиме несущая передатчика будет включаться только на время «t» для формирования импульсов с числом, равным номеру датчика. Учитывая, что период импульсов равен 5,37 мс, максимальное значение «t» будет приблизительно равно 0,06 мс (И импульсов). А максимальное значение длительности паузы между импульсами будет равно 2 - 0,06 = 1,94 с. Счетчик миллисекунд демодулятора до включения аварийного режима имеет коэффициент деления, равный 9, поэтому переполнение счетчика будет происходить через 1,125 секунды, что гораздо меньше максимального значения паузы.

При первом переполнении счетчика миллисекунд включается флаг переполнения. После включения флага переполнения начинает заполняться регистр счета. Перезапись в регистр индикации производится спустя 0,6 с после прихода последнего импульса. Таким образом, индикация обновляется каждые две секунды.

Если число импульсов, поступивших с компаратора, больше 11, то результат не выводится на индикатор как заведомо ложный. Такой вариант возможен при высоком уровне помех. Но даже и в этом случае включение звукового сигнала будет свидетельствовать о несанкционированном вторжении на охраняемый объект. При поступлении более 20 импульсов при любом значении флага переполнения включится тревожный сигнал. Это сделано для того, чтобы невозможно было вывести систему из строя путем подачи непрерывного сигнала помехи.

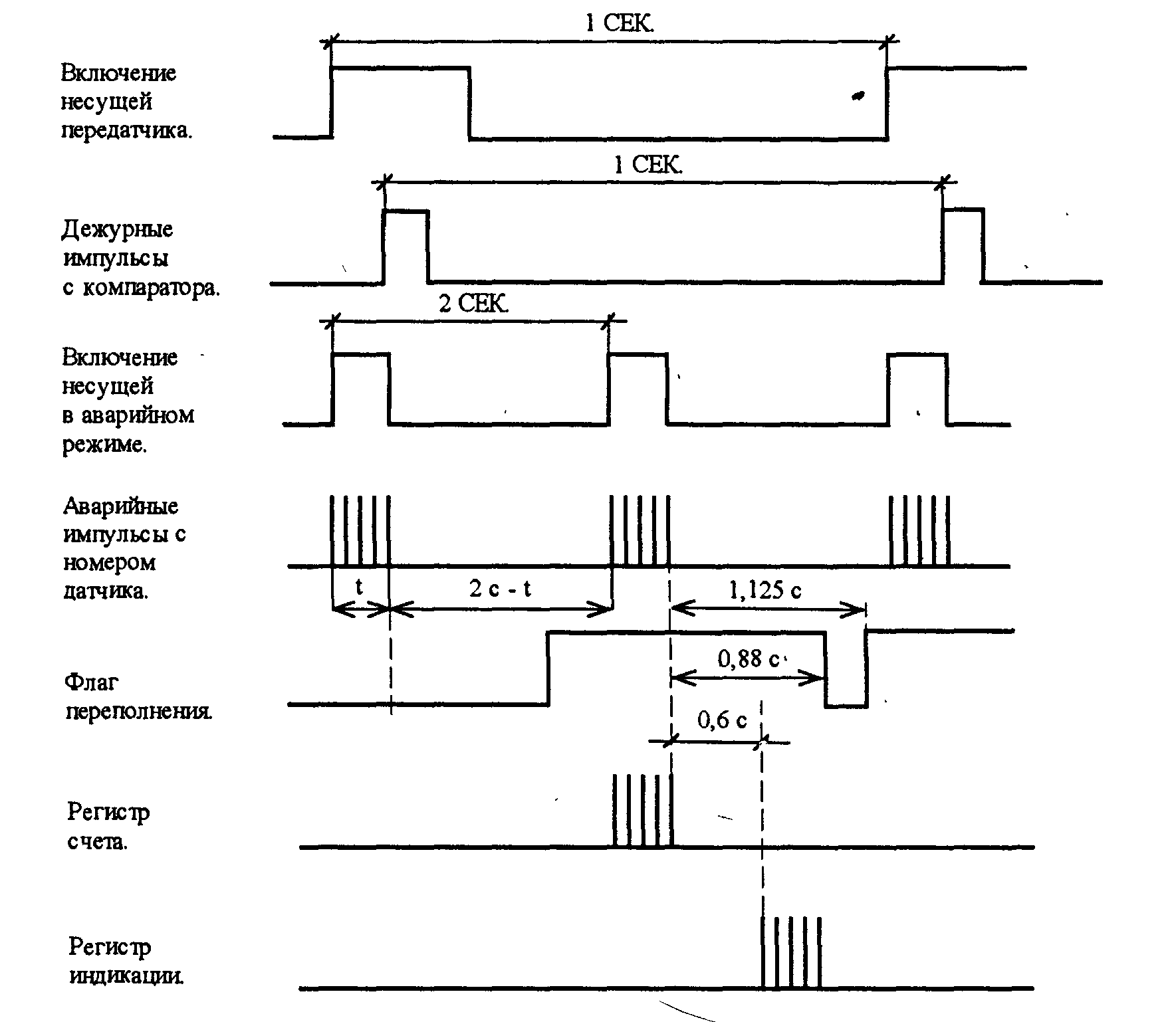


Рис. 7 - Диаграммы работы демодулятора

При экспериментировании с подачей сигнала сильной помехи по общему проводу наблюдался самопроизвольный переход микроконтроллеров в режим повышенного энергопотребления. Это я могу объяснить самопроизвольной переустановкой регистра, который определяет установку входа как выход (TRISA, TRISB).

**4. Ассемблирование**

Для ассемблирования спользуется макpоассемблеp MPASM, он содеpжит все необходимые нам возможности. MPASM входит в пакет программ Microchip MPLAB фирмы Microchip Technology.

Программа на языке Ассемблеp приведена в Приложении А.

**5. Описание функциональных узлов МПС и алгоритма их взаимодействия**

Описание функциональных узлов было описано в пункте разработки алгоритма устройства, поэтому не будем более подробно останавливаться на разборе, т.к. алгоритм взаимодействия будет описан еще при разработке принципиальной схемы и выборе элементарной базы.

В Приложении Б приведена программа файла модулятора, а в Приложении В файла демодулятора.

**6. Описание выбора элементной базы и работы принципиальной схемы**

Принципиальная схема модулятора показана на рис.10 (Приложение Г). На выходе RA3 появляются импульсы модуляции положительной полярности. На выходе RA4 появляется единичный потенциал включения несущей. Все остальные входы микроконтроллера задействованы охранными датчиками. Охранное положение датчиков нормально замкнутое. Программно ко всем входам порта «В» подключены подтягивающие резисторы, поэтому при размыкании датчиков программа обнаружит единичный потенциал на разомкнутом датчике. Входы порта «А» программно не подтягиваются к плюсовому напряжению питания, поэтому установлены резисторы Rl-R3. При подключении вывода RA4 к схеме передатчика необходимо иметь в виду, что этот выход имеет открытый сток, поэтому, в зависимости от схемы передатчика, может потребоваться нагрузочный резистор.

Схема демодулятора показана на рис. 11 (Приложение Д). Вход RB0 подключается к выходу компаратора приемника. Выход компаратора должен быть с уровнями ТТЛ. Остальные выводы порта «В» задействованы под выход сегментов индикатора.

К выходу RA4 подключается зуммер НСМ120бх. Кнопка SA1 включает индикатор. Резистор R1 необходимо подобрать по приемлемой яркости для свечения 3-4 сегментов. Тогда один сегмент будет светиться ярче, а восемь сегментов будут светиться слабее.

Перед подключением пожарной сигнализации к приемнику и передатчику желательно проверить ее на совместную работу. Для этого необходимо выход несущей и модуляции с модулятора подать на входы логического элемента микросхемы К561ЛА7 и, инвертировав сигнал еще одним элементом, подать на вход демодулятора. При проверке индикации число 10 на индикаторе высвечивается нижней чертой (сегмент d), а число 11 - средней чертой (сегмент g).

Все временные характеристики приведены для случая использования кварцевого резонатора на частоту 32768 Гц.

Принципиальные схемы модулятора и демодулятора выполнены в САПР Accel Eda (Рис. 8 и Рис.9).

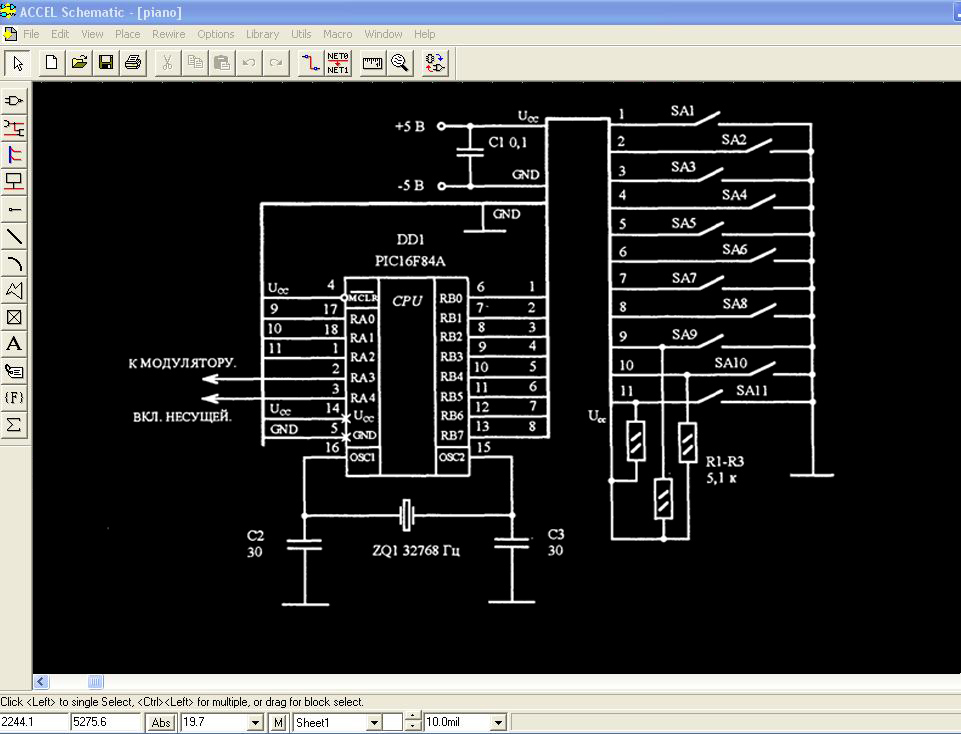


Рис. 8 - Принципиальная схема модулятора пожарной сигнализации в Accel EDA.

Принципиальная схема модулятора пожарной сигнализации приведена в Приложении Г.

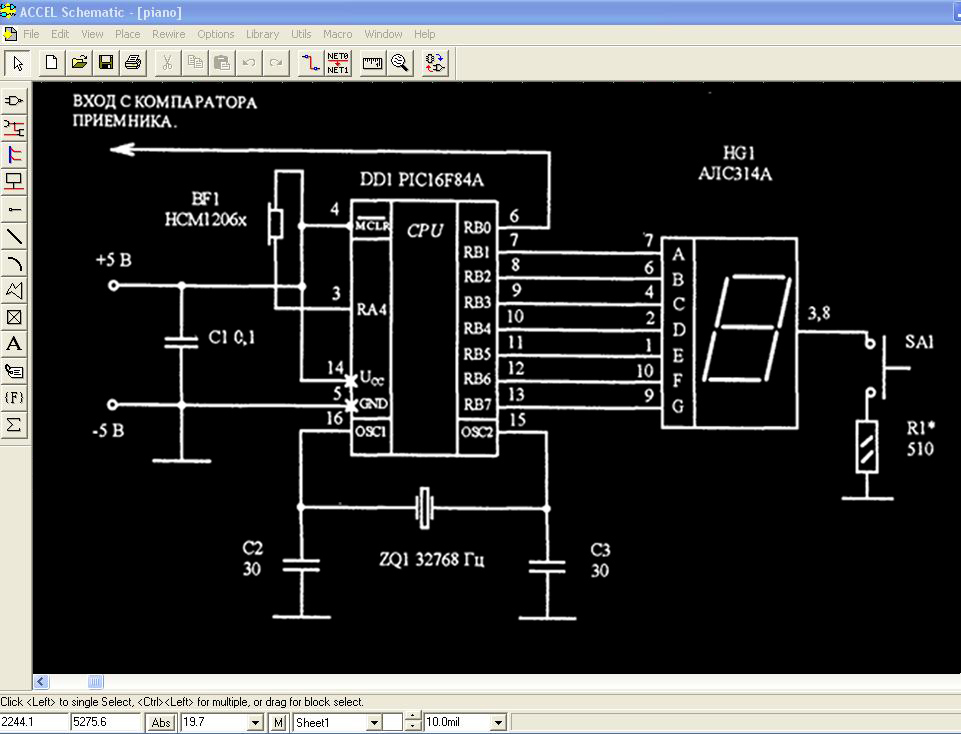


Рис. 9 - Принципиальная схема демодулятора пожарной сигнализации в Accel EDA

Принципиальная схема демодулятора пожарной сигнализации приведена в Приложении Д.

Спецификация компонентов модулятора приведена в Приложении Ж.

Спецификация компонентов демодулятора приведена в Приложении К.

**ЗАКЛЮЧЕНИЕ**

В данном курсовом проекте разработано устройство - пожарная сигнализация на 11 датчиков. Разработана схема электрическая принципиальная этого устройства и программа для микроконтроллера. В результате ассеблирования получена прошивка программы для памяти микроконтроллера. Применение микроконтроллера позволило упростить принципиальную схему и расширить функциональные возможности микроконтроллера, так как для изменения функций устройства достаточно внести изменения в программу микроконтроллера. Преимуществом данного устройства над существующими в том, что для передачи данных о срабатывании датчика пожарной сигнализации используется радиоканал, т.е. осуществляется экономия на электрических проводах и повышается надежность эксплуатации и обслуживания устройства.

**СПИСОК ЛИТЕРАТУРЫ**

1. Белов А.В. Микроконтроллеры АVR в радиолюбительской практике – СП-б, Наука и техника, 2007 – 352с.

2. Проектирование цифровых устройств на однокристальных микроконтроллерах / В.В. Сташин [ и др.]. – М.: Энергоатомиздат, 1990. – 224 с.

3. Евстифеев А.В. Микроконтроллеры Microchip: практическое руководство/А.В.Евстифеев. – М.: Горячая линия – Телеком, 2002. – 296 с.

4. Кравченко А.В. 10 практических устройств на AVR-микро-

контроллерах. Книга 1 – М., Додэка –ХХ1, МК-Пресс, 2008 – 224с.

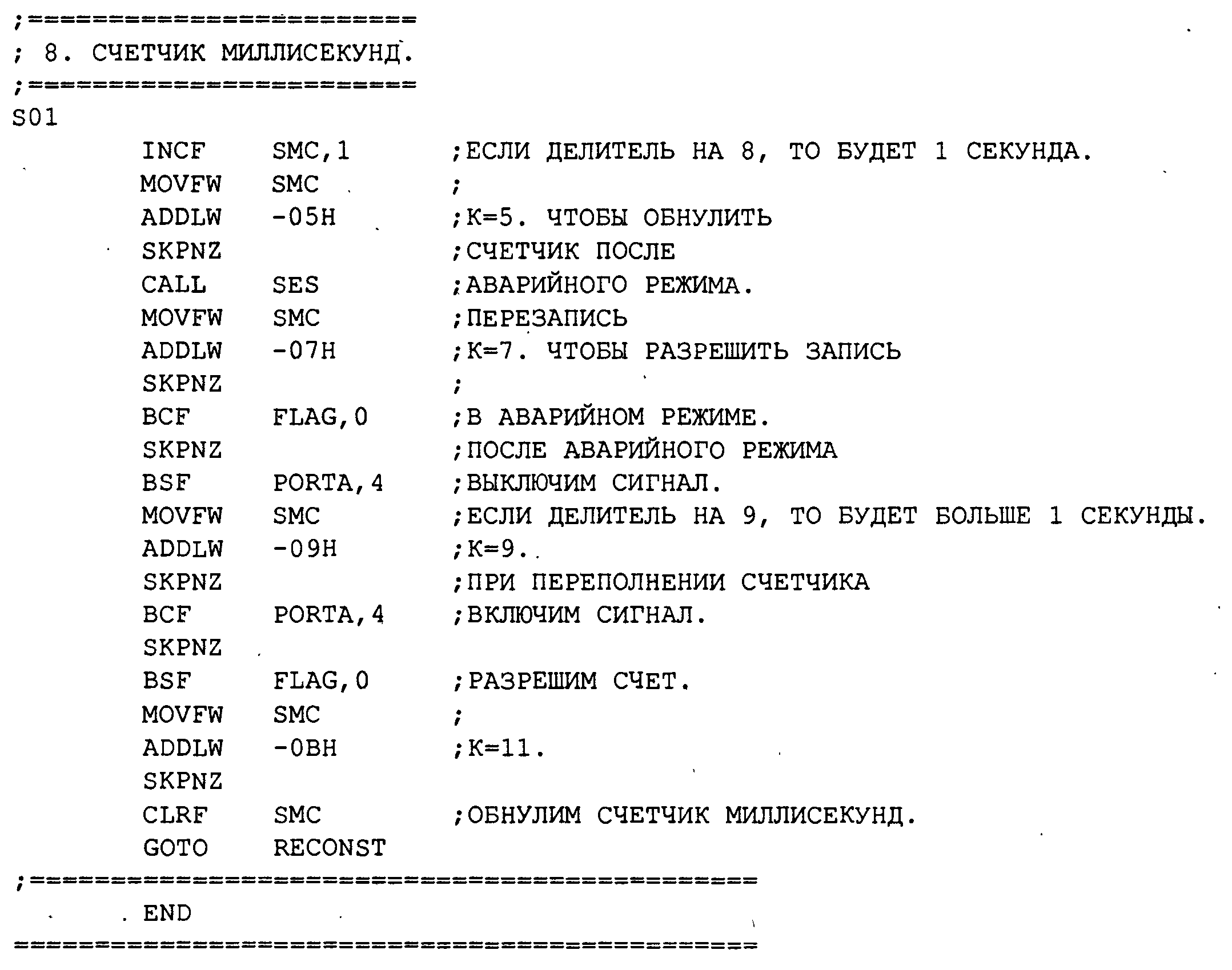
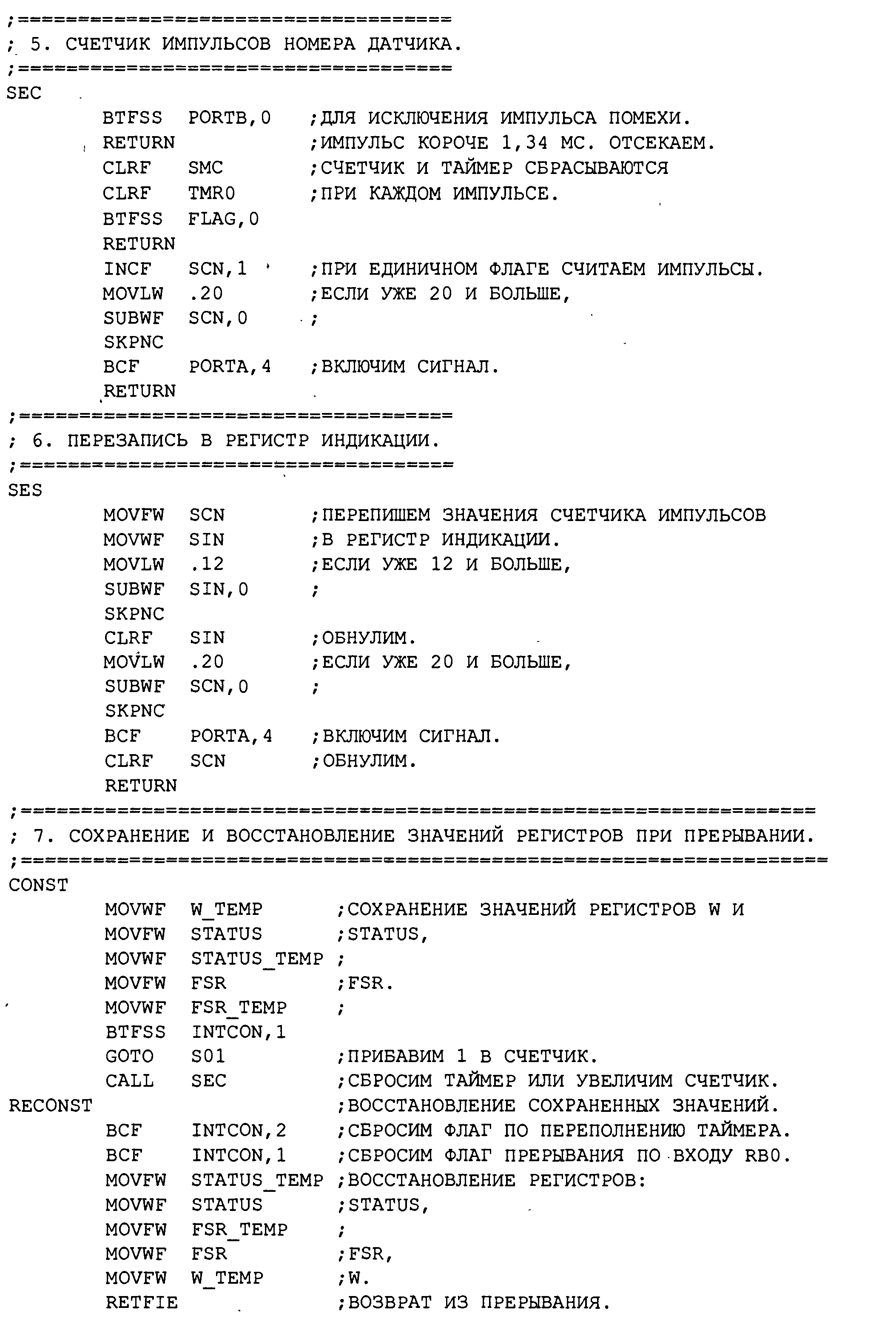
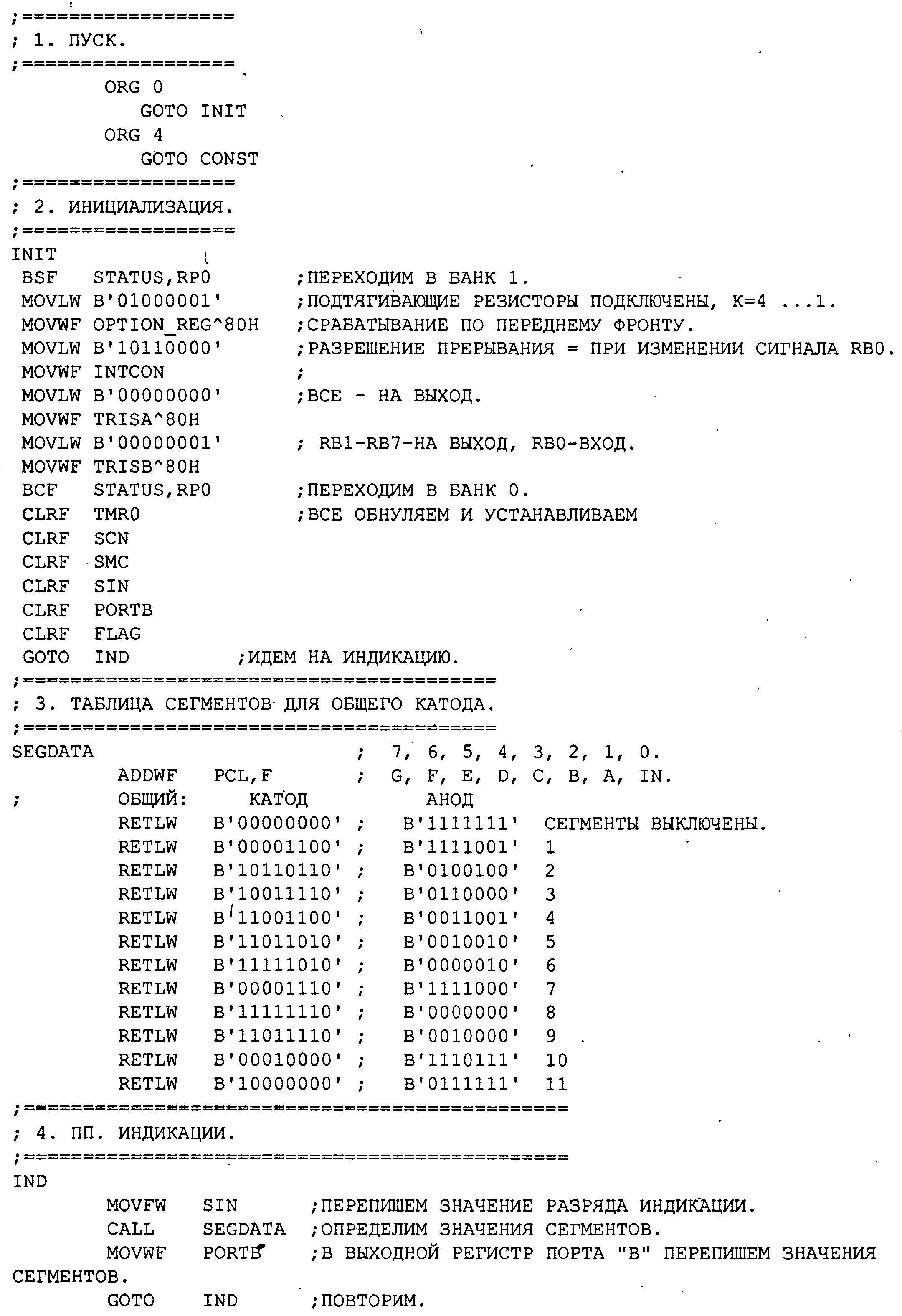
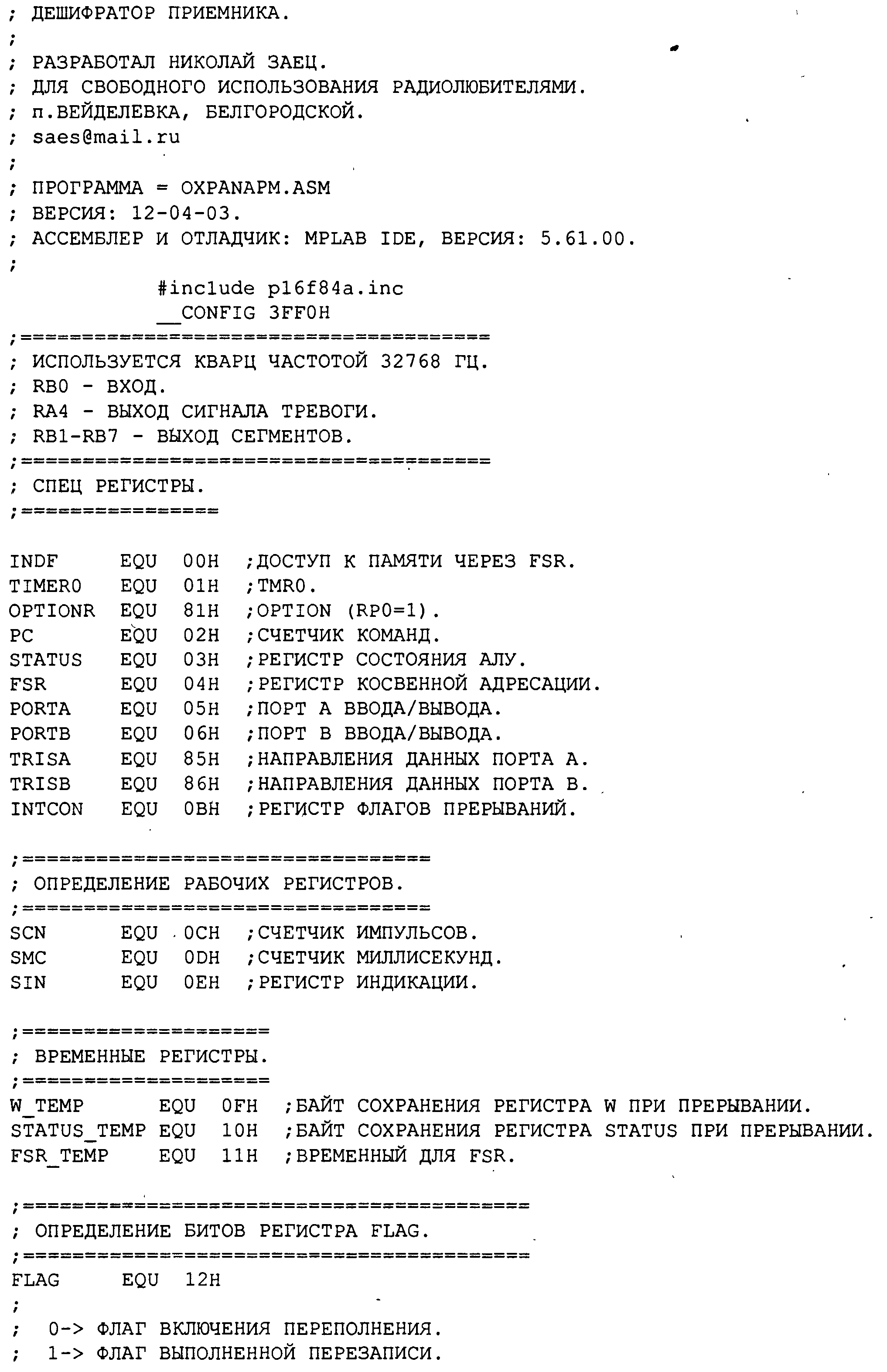
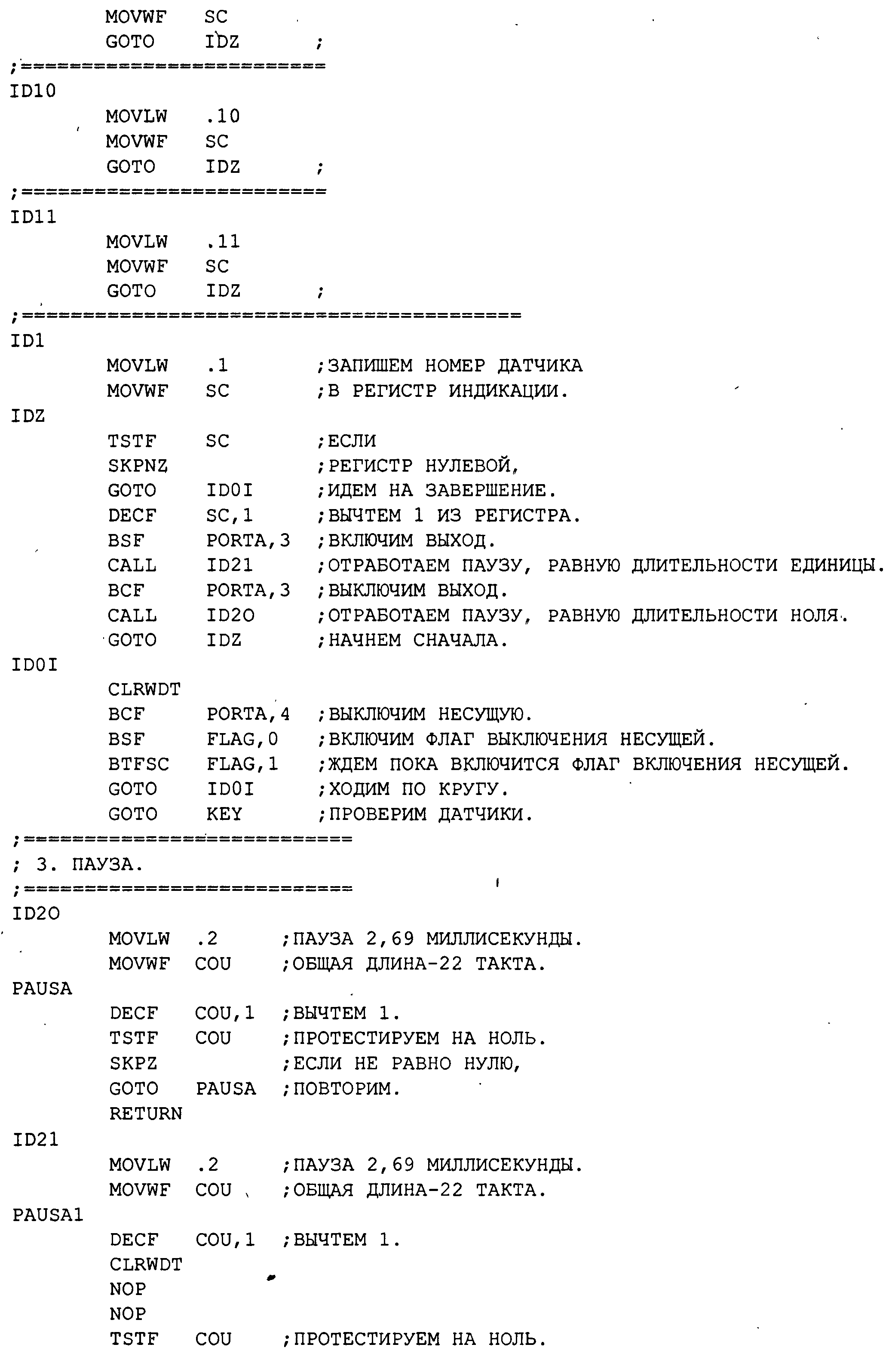
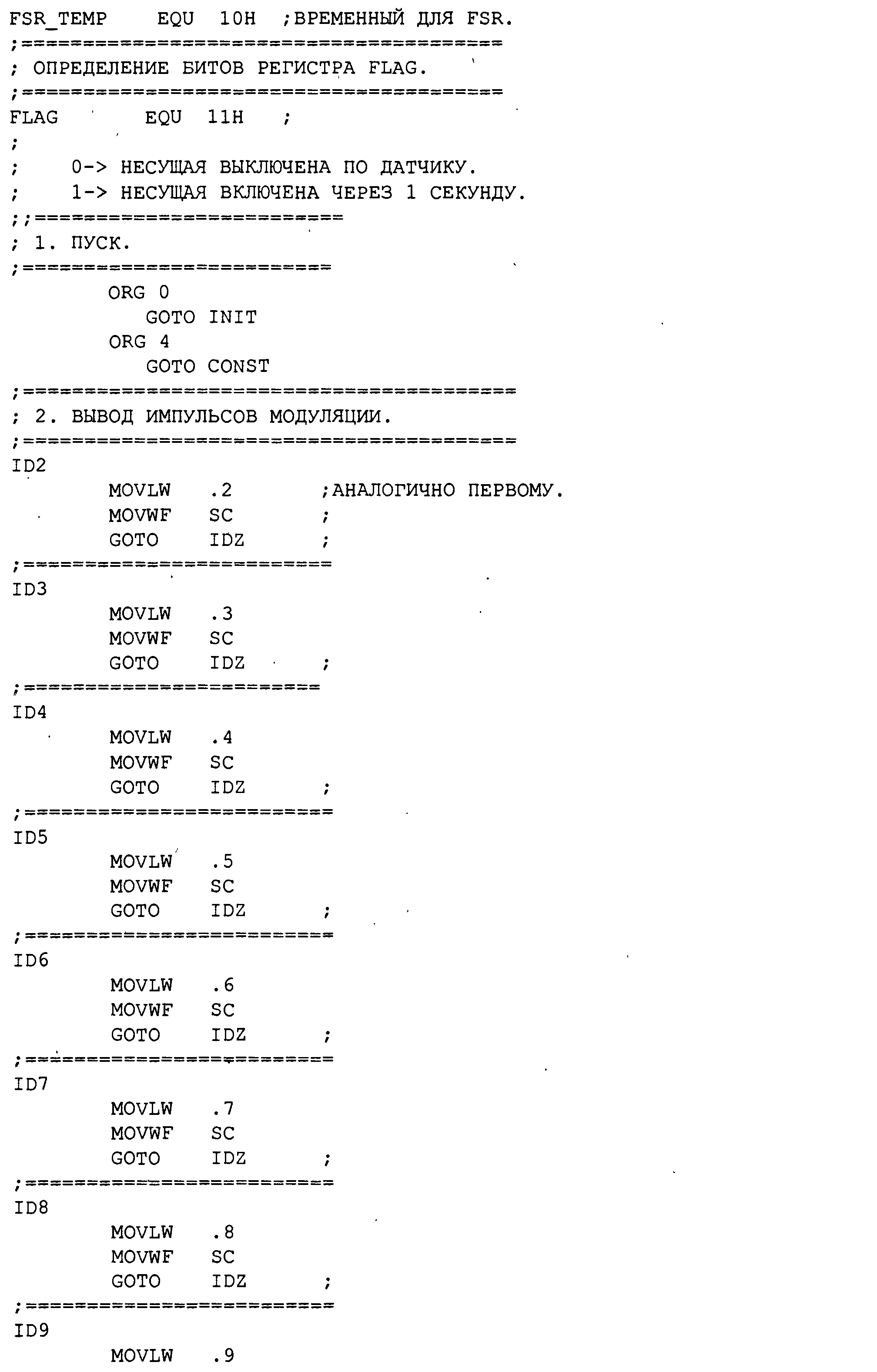
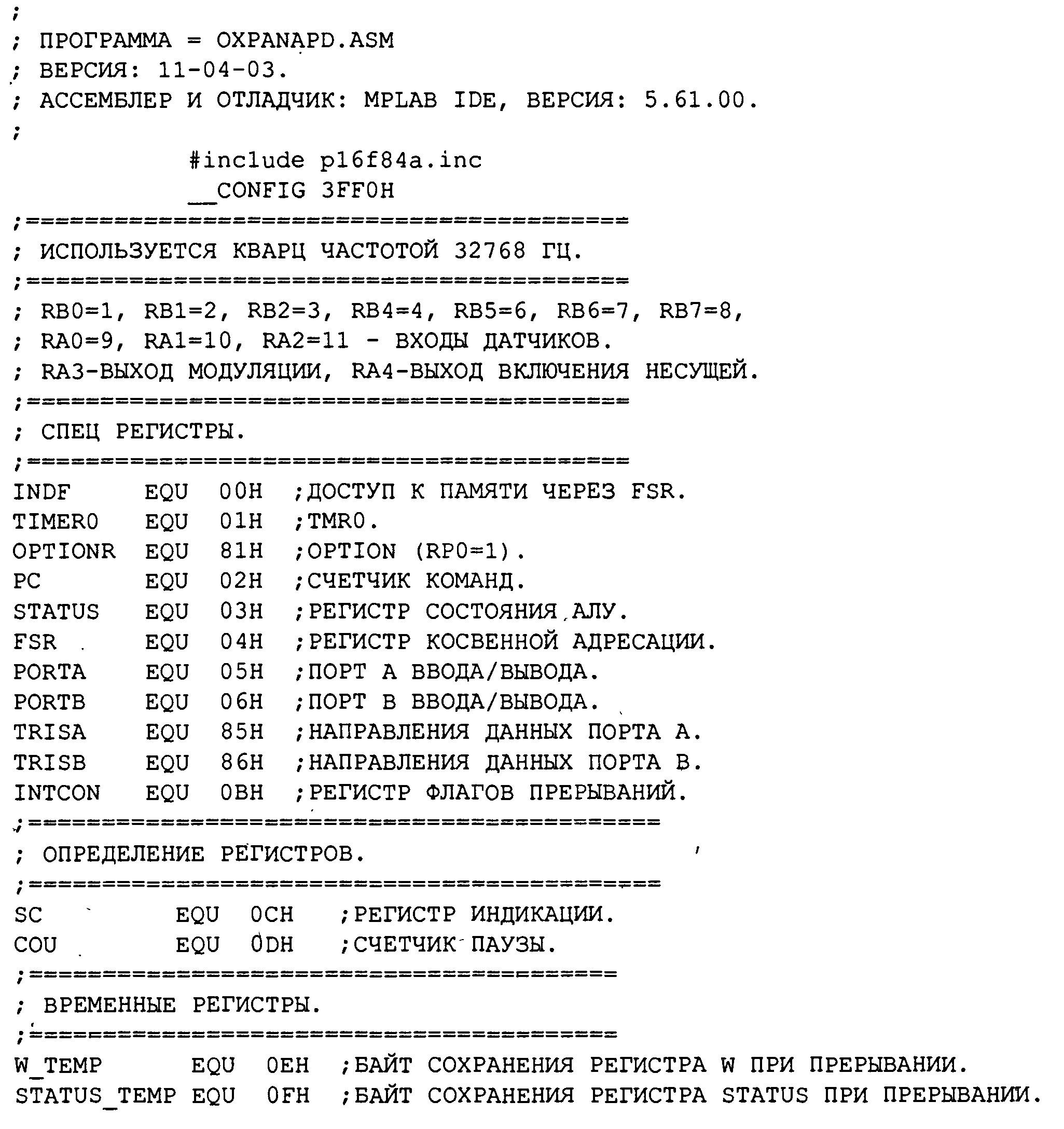
5.Трамперт В. Измерение, управление и регулирование с помощью АVR-микроконтроллеров: Пер. с нем – К., МК-Пресс, 2006 – 208с.

6. Мортон Дж. Микроконтроллеры АVR. Вводный курс /Пер. с англ. – М., Додэка –ХХ1, 2006 – 272с.

7. Техническая документация на микроконтроллеры PIC16F84А компании Microchip Technology Incorporated. ООО «Микро-Чип», Москва, 2002.-184 с.

**ПРИЛОЖЕНИЕ А**

**Листинг программы и объектный файл**



**Приложение Б**

; ПОЖАРНАЯ СИГНАЛИЗАЦИЯ НА 11 ДАТЧИКОВ,

; МОДУЛЯТОР ПЕРЕДАТЧИКА.

; РАЗРАБОТАЛ ЧЕБОТАРЕВ А,А,.

; ДЛЯ СВОБОДНОГО ИСПОЛЬЗОВАНИЯ.

; СЕВЕРОДОНЕЦК

;

; ПРОГРАММА = OXPANAPD.ASM

; ВЕРСИЯ: 11-04-03.

; АССЕМБЛЕР И ОТЛАДЧИК: MPLAB IDE, ВЕРСИЯ: 5.61.00.

;

#include p16f84a.inc

\_\_CONFIG 3FF0H

;=========================================

; ИСПОЛЬЗУЕТСЯ КВАРЦ ЧАСТОТОЙ 32768 ГЦ.

;=========================================

; RB0=1, RB1=2, RB2=3, RB4=4, RB5=6, RB6=7, RB7=8,

; RA0=9, RA1=10, RA2=11 - ВХОДЫ ДАТЧИКОВ.

; RA3-ВЫХОД МОДУЛЯЦИИ, RA4-ВЫХОД ВКЛЮЧЕНИЯ НЕСУЩЕЙ.

;=========================================

; СПЕЦ РЕГИСТРЫ.

;=========================================

INDF EQU 00H ;ДОСТУП К ПАМЯТИ ЧЕРЕЗ FSR.

TIMER0 EQU 01H ;TMR0.

OPTIONR EQU 81H ;OPTION (RP0=1).

PC EQU 02H ;СЧЕТЧИК КОМАНД.

STATUS EQU 03H ;РЕГИСТР СОСТОЯНИЯ АЛУ.

FSR EQU 04H ;РЕГИСТР КОСВЕННОЙ АДРЕСАЦИИ.

PORTA EQU 05H ;ПОРТ А ВВОДА/ВЫВОДА.

PORTB EQU 06H ;ПОРТ В ВВОДА/ВЫВОДА.

TRISA EQU 85H ;НАПРАВЛЕНИЯ ДАННЫХ ПОРТА А.

TRISB EQU 86H ;НАПРАВЛЕНИЯ ДАННЫХ ПОРТА В.

INTCON EQU 0BH ;РЕГИСТР ФЛАГОВ ПРЕРЫВАНИЙ.

;===========================================

; ОПРЕДЕЛЕНИЕ РЕГИСТРОВ.

;===========================================

SC EQU 0CH ;РЕГИСТР ИНДИКАЦИИ.

COU EQU 0DH ;СЧЕТЧИК ПАУЗЫ.

;========================================

; ВРЕМЕННЫЕ РЕГИСТРЫ.

;========================================

W\_TEMP EQU 0EH ;БАЙТ СОХРАНЕНИЯ РЕГИСТРА W ПРИ ПРЕРЫВАНИИ.

STATUS\_TEMP EQU 0FH ;БАЙТ СОХРАНЕНИЯ РЕГИСТРА STATUS ПРИ ПРЕРЫВАНИИ.

FSR\_TEMP EQU 10H ;ВРЕМЕННЫЙ ДЛЯ FSR.

;=======================================

; ОПРЕДЕЛЕНИЕ БИТОВ РЕГИСТРА FLAG.

;=======================================

FLAG EQU 11H ;

;

; 0-> НЕСУЩАЯ ВЫКЛЮЧЕНА ПО ДАТЧИКУ.

; 1-> НЕСУЩАЯ ВКЛЮЧЕНА ЧЕРЕЗ 1 СЕКУНДУ.

;

;=========================

; 1. ПУСК.

;=========================

ORG 0

GOTO INIT

ORG 4

GOTO CONST

;========================================

; 4. ВЫВОД ИМПУЛЬСОВ МОДУЛЯЦИИ.

;========================================

ID2

MOVLW.2 ;АНАЛОГИЧНО ПЕРВОМУ.

MOVWF SC ;

GOTO IDZ ;

;=========================

ID3

MOVLW.3

MOVWF SC

GOTO IDZ ;

;========================

ID4

MOVLW.4

MOVWF SC

GOTO IDZ ;

;=========================

ID5

MOVLW.5

MOVWF SC

GOTO IDZ ;

;=========================

ID6

MOVLW.6

MOVWF SC

GOTO IDZ ;

;=========================

ID7

MOVLW.7

MOVWF SC

GOTO IDZ ;

;=========================

ID8

MOVLW.8

MOVWF SC

GOTO IDZ ;

;=========================

ID9

MOVLW.9

MOVWF SC

GOTO IDZ ;

;=========================

ID10

MOVLW.10

MOVWF SC

GOTO IDZ ;

;=========================

ID11

MOVLW.11

MOVWF SC

GOTO IDZ ;

;=========================================

ID1

MOVLW.1 ;ЗАПИШЕМ НОМЕР ДАТЧИКА

MOVWF SC ;В РЕГИСТР ИНДИКАЦИИ.

IDZ

TSTF SC ;ЕСЛИ

SKPNZ ;РЕГИСТР НУЛЕВОЙ,

GOTO ID0I ;ИДЕМ НА ЗАВЕРШЕНИЕ.

DECF SC,1 ;ВЫЧТЕМ 1 ИЗ РЕГИСТРА.

BSF PORTA,3 ;ВКЛЮЧИМ ВЫХОД.

CALL ID21 ;ОТРАБОТАЕМ ПАУЗУ, РАВНУЮ ДЛИТЕЛЬНОСТИ ЕДИНИЦЫ.

BCF PORTA,3 ;ВЫКЛЮЧИМ ВЫХОД.

CALL ID2O ;ОТРАБОТАЕМ ПАУЗУ, РАВНУЮ ДЛИТЕЛЬНОСТИ НОЛЯ.

GOTO IDZ ;НАЧНЕМ СНАЧАЛА.

ID0I

CLRWDT

BCF PORTA,4 ;ВЫКЛЮЧИМ НЕСУЩУЮ.

BSF FLAG,0 ;ВКЛЮЧИМ ФЛАГ ВЫКЛЮЧЕНИЯ НЕСУЩЕЙ.

BTFSC FLAG,1 ;ЖДЕМ ПОКА ВКЛЮЧИТСЯ ФЛАГ ВКЛЮЧЕНИЯ НЕСУЩЕЙ.

GOTO ID0I ;ХОДИМ ПО КРУГУ.

GOTO KEY ;ПРОВЕРИМ ДАТЧИКИ.

;===========================

; 5. ПАУЗА.

;===========================

ID2O

MOVLW.2 ;ПАУЗА 2,69 МИЛЛИСЕКУНДЫ.

MOVWF COU ;ОБЩАЯ ДЛИНА-22 ТАКТА.

PAUSA

DECF COU,1 ;ВЫЧТЕМ 1.

TSTF COU ;ПРОТЕСТИРУЕМ НА НОЛЬ.

SKPZ ;ЕСЛИ НЕ РАВНО НУЛЮ,

GOTO PAUSA ;ПОВТОРИМ.

RETURN

ID21

MOVLW.2 ;ПАУЗА 2,69 МИЛЛИСЕКУНДЫ.

MOVWF COU ;ОБЩАЯ ДЛИНА-22 ТАКТА.

PAUSA1

DECF COU,1 ;ВЫЧТЕМ 1.

CLRWDT

NOP

NOP

TSTF COU ;ПРОТЕСТИРУЕМ НА НОЛЬ.

SKPZ ;ЕСЛИ НЕ РАВНО НУЛЮ,

GOTO PAUSA1 ;ПОВТОРИМ.

RETURN

;=============================

; 3. ОПРОС ДАТЧИКОВ.

;=============================

KEY

CLRWDT

BTFSS FLAG,1 ;ЕСЛИ НЕСУЩАЯ ВЫКЛЮЧЕНА,

GOTO KEY ;ОПРОС ДАТЧИКОВ НЕ ПРОИЗВОДИТСЯ.

BTFSC PORTB,0 ;ЕСЛИ ДАТЧИК РАЗОРВАН,

GOTO ID1 ;ИДЕМ ФОРМИРОВАТЬ ИМПУЛЬС МОДУЛЯЦИИ.

BTFSC PORTB,1

GOTO ID2

BTFSC PORTB,2

GOTO ID3

BTFSC PORTB,3

GOTO ID4

BTFSC PORTB,4

GOTO ID5

BTFSC PORTB,5

GOTO ID6

BTFSC PORTB,6

GOTO ID7

BTFSC PORTB,7

GOTO ID8

BTFSC PORTA,0

GOTO ID9

BTFSC PORTA,1

GOTO ID10

BTFSC PORTA,2

GOTO ID11

BCF FLAG,0

GOTO KEY ;ЕЩЕ РАЗ ПРОВЕРИМ ДАТЧИКИ.

;== ================================================

; 6. СОХРАНЕНИЕ И ВОССТАНОВЛЕНИЕ ЗНАЧЕНИЙ РЕГИСТРОВ ПРИ ПРЕРЫВАНИИ.

;======================================================

CONST

MOVWF W\_TEMP ;СОХРАНЕНИЕ ЗНАЧЕНИЙ РЕГИСТРОВ W И

MOVFW STATUS ;STATUS,

MOVWF STATUS\_TEMP ;

MOVFW FSR ;FSR.

MOVWF FSR\_TEMP ;

CALL S1 ;ПРИБАВИМ 1 В СЧЕТЧИК.

RECONST ;ВОССТАНОВЛЕНИЕ СОХРАНЕННЫХ ЗНАЧЕНИЙ.

BCF INTCON,2 ;СБРОС ФЛАГА (T0IF) ПЕРЕПОЛНЕНИЯ ТАЙМЕРА.

MOVFW STATUS\_TEMP ;ВОССТАНОВЛЕНИЕ РЕГИСТРОВ:

MOVWF STATUS ;STATUS,

MOVFW FSR\_TEMP ;

MOVWF FSR ;FSR,

MOVFW W\_TEMP ;W.

RETFIE ;ВОЗВРАТ ИЗ ПРЕРЫВАНИЯ.

;============================

; 7. ВКЛЮЧЕНИЕ НЕСУЩЕЙ.

;============================

S1

BTFSC FLAG,1 ;ЕСЛИ НЕСУЩАЯ БЫЛА ВКЛЮЧЕНА,

GOTO S10 ;ТО ИДЕМ ВЫКЛЮЧАТЬ.

BSF PORTA,4 ;ВКЛЮЧИМ НЕСУЩУЮ,

BSF FLAG,1 ;ПОСТАВИМ ФЛАГ ВКЛЮЧЕНИЯ.

BTFSC FLAG,0 ;ЕСЛИ НЕСУЩАЯ ВЫКЛЮЧАЛАСЬ ПО ДАТЧИКУ,

RETURN ;ИМПУЛЬС НЕ ФОРМИРУЕТСЯ.

CALL ID2O ;ПАУЗА, РАВНАЯ НУЛЮ.

BSF PORTA,3 ;ВКЛЮЧИМ ВЫХОД.

CALL ID21 ;ОТРАБОТАЕМ ПАУЗУ, РАВНУЮ ДЛИТЕЛЬНОСТИ ЕДИНИЦЫ.

BCF PORTA,3 ;ВЫКЛЮЧИМ ВЫХОД.

CALL ID2O ;ПАУЗА, РАВНАЯ НУЛЮ.

BCF PORTA,4 ;ВЫКЛЮЧИМ НЕСУЩУЮ.

RETURN

S10

BCF FLAG,1 ;СБРОСИМ ФЛАГ ВКЛЮЧЕНИЯ НЕСУЩЕЙ.

BTFSC FLAG,0 ;ЕСЛИ НЕСУЩАЯ ВЫКЛЮЧАЛАСЬ ПО ДАТЧИКУ,

RETURN ;ИМПУЛЬС НЕ ФОРМИРУЕТСЯ.

BSF PORTA,4 ;ВКЛЮЧИМ НЕСУЩУЮ,

CALL ID2O ;ПАУЗА, РАВНАЯ НУЛЮ.

BSF PORTA,3 ;ВКЛЮЧИМ ВЫХОД.

CALL ID21 ;ОТРАБОТАЕМ ПАУЗУ, РАВНУЮ ДЛИТЕЛЬНОСТИ ЕДИНИЦЫ.

BCF PORTA,3 ;ВЫКЛЮЧИМ ВЫХОД.

CALL ID2O ;ПАУЗА, РАВНАЯ НУЛЮ.

BCF PORTA,4 ;ВЫКЛЮЧИМ НЕСУЩУЮ.

RETURN

;======================

; 2. ИНИЦИАЛИЗАЦИЯ.

;======================

INIT

BSF STATUS,RP0 ;ПЕРЕХОДИМ В БАНК 1.

MOVLW B'00000100';ПРЕДДЕЛИТЕЛЬ ПЕРЕД ТАЙМЕРОМ, К=32...100,

MOVWF OPTION\_REG^80H;ПОДТЯГИВАЮЩИЕ РЕЗИСТОРЫ ВКЛЮЧЕНЫ.

MOVLW B'10100000' ;РАЗРЕШЕНИЕ ПРЕРЫВАНИЯ = ПРИ ПЕРЕПОЛНЕНИИ ТАЙМЕРА.

MOVWF INTCON ;

MOVLW B'00000111' ;RA3,RA4-НА ВЫХОД, RA0-RA2-ВХОД.

MOVWF TRISA^80H

MOVLW B'11111111' ;RB0-RB7-НА ВХОД.

MOVWF TRISB^80H

BCF STATUS,RP0 ;ПЕРЕХОДИМ В БАНК 0.

CLRF TMR0 ;ВСЕ ОБНУЛЯЕМ И УСТАНАВЛИВАЕМ

CLRF FLAG

CLRF PORTA

CLRF COU

GOTO KEY ;ИДЕМ НА ОПРОС ДАТЧИКОВ.

;==============================================

END

;

**Приложение В**

ПОЖАРНАЯ СИГНАЛИЗАЦИЯ НА 11 ДАТЧИКОВ,

; ДЕШИФРАТОР ПРИЕМНИКА.

;

; РАЗРАБОТАЛ ЧЕБОТАРЕВ А,А,

; ДЛЯ СВОБОДНОГО ИСПОЛЬЗОВАНИЯ.

; СЕВЕРОДОНЕЦК.

;

; ПРОГРАММА = OXPANAPM.ASM

; ВЕРСИЯ: 12-04-03.

; АССЕМБЛЕР И ОТЛАДЧИК: MPLAB IDE, ВЕРСИЯ: 5.61.00.

;

#include p16f84a.inc

\_\_CONFIG 3FF0H

;======================================

; ИСПОЛЬЗУЕТСЯ КВАРЦ ЧАСТОТОЙ 32768 ГЦ.

; RB0 - ВХОД.

; RA4 - ВЫХОД СИГНАЛА ТРЕВОГИ.

; RB1-RB7 - ВЫХОД СЕГМЕНТОВ.

;======================================

; СПЕЦ РЕГИСТРЫ.

;================

INDF EQU 00H ;ДОСТУП К ПАМЯТИ ЧЕРЕЗ FSR.

TIMER0 EQU 01H ;TMR0.

OPTIONR EQU 81H ;OPTION (RP0=1).

PC EQU 02H ;СЧЕТЧИК КОМАНД.

STATUS EQU 03H ;РЕГИСТР СОСТОЯНИЯ АЛУ.

FSR EQU 04H ;РЕГИСТР КОСВЕННОЙ АДРЕСАЦИИ.

PORTA EQU 05H ;ПОРТ А ВВОДА/ВЫВОДА.

PORTB EQU 06H ;ПОРТ В ВВОДА/ВЫВОДА.

TRISA EQU 85H ;НАПРАВЛЕНИЯ ДАННЫХ ПОРТА А.

TRISB EQU 86H ;НАПРАВЛЕНИЯ ДАННЫХ ПОРТА В.

INTCON EQU 0BH ;РЕГИСТР ФЛАГОВ ПРЕРЫВАНИЙ.

;=================================

; ОПРЕДЕЛЕНИЕ РАБОЧИХ РЕГИСТРОВ.

;=================================

SCN EQU 0CH ;СЧЕТЧИК ИМПУЛЬСОВ.

SMC EQU 0DH ;СЧЕТЧИК МИЛЛИСЕКУНД.

SIN EQU 0EH ;РЕГИСТР ИНДИКАЦИИ.

;====================

; ВРЕМЕННЫЕ РЕГИСТРЫ.

;====================

W\_TEMP EQU 0FH ;БАЙТ СОХРАНЕНИЯ РЕГИСТРА W ПРИ ПРЕРЫВАНИИ.

STATUS\_TEMP EQU 10H ;БАЙТ СОХРАНЕНИЯ РЕГИСТРА STATUS ПРИ ПРЕРЫВАНИИ.

FSR\_TEMP EQU 11H ;ВРЕМЕННЫЙ ДЛЯ FSR.

;=========================================

; ОПРЕДЕЛЕНИЕ БИТОВ РЕГИСТРА FLAG.

;=========================================

FLAG EQU 12H

;

; 0-> ФЛАГ ВКЛЮЧЕНИЯ ПЕРЕПОЛНЕНИЯ.

; 1-> ФЛАГ ВЫПОЛНЕННОЙ ПЕРЕЗАПИСИ.

;==================

; 1. ПУСК.

;==================

ORG 0

GOTO INIT

ORG 4

GOTO CONST

;==================

; 2. ИНИЦИАЛИЗАЦИЯ.

==================

INIT

BSF STATUS,RP0 ;ПЕРЕХОДИМ В БАНК 1.

MOVLW B'01000001' ;ПОДТЯГИВАЮЩИЕ РЕЗИСТОРЫ ПОДКЛЮЧЕНЫ, К=4...1.

MOVWF OPTION\_REG^80H ;СРАБАТЫВАНИЕ ПО ПЕРЕДНЕМУ ФРОНТУ.

MOVLW B'10110000' ;РАЗРЕШЕНИЕ ПРЕРЫВАНИЯ = ПРИ ИЗМЕНЕНИИ СИГНАЛА RВ0.

MOVWF INTCON ;

MOVLW B'00000000' ;ВСЕ - НА ВЫХОД.

MOVWF TRISA^80H

MOVLW B'00000001' ; RB1-RB7-НА ВЫХОД, RB0-ВХОД.

MOVWF TRISB^80H

BCF STATUS,RP0 ;ПЕРЕХОДИМ В БАНК 0.

CLRF TMR0 ;ВСЕ ОБНУЛЯЕМ И УСТАНАВЛИВАЕМ

CLRF SCN

CLRF SMC

CLRF SIN

CLRF PORTB

CLRF FLAG

GOTO IND ;ИДЕМ НА ИНДИКАЦИЮ.

;========================================

; 3. ТАБЛИЦА СЕГМЕНТОВ ДЛЯ ОБЩЕГО КАТОДА.

;========================================

SEGDATA ; 7, 6, 5, 4, 3, 2, 1, 0.

ADDWF PCL,F ; G, F, E, D, C, B, A, IN.

; ОБЩИЙ: КАТОД АНОД

RETLW B'00000000' ; B'1111111' СЕГМЕНТЫ ВЫКЛЮЧЕНЫ.

RETLW B'00001100' ; B'1111001' 1

RETLW B'10110110' ; B'0100100' 2

RETLW B'10011110' ; B'0110000' 3

RETLW B'11001100' ; B'0011001' 4

RETLW B'11011010' ; B'0010010' 5

RETLW B'11111010' ; B'0000010' 6

RETLW B'00001110' ; B'1111000' 7

RETLW B'11111110' ; B'0000000' 8

RETLW B'11011110' ; B'0010000' 9

RETLW B'00010000' ; B'1110111' 10

RETLW B'10000000' ; B'0111111' 11

;==============================================

; 4. ПП. ИНДИКАЦИИ.

;==============================================

IND

MOVFW SIN ;ПЕРЕПИШЕМ ЗНАЧЕНИЕ РАЗРЯДА ИНДИКАЦИИ.

CALL SEGDATA ;ОПРЕДЕЛИМ ЗНАЧЕНИЯ СЕГМЕНТОВ.

MOVWF PORTB ;В ВЫХОДНОЙ РЕГИСТР ПОРТА "B" ПЕРЕПИШЕМ ЗНАЧЕНИЯ СЕГМЕНТОВ.

GOTO IND ;ПОВТОРИМ.

;====================================

; 5. СЧЕТЧИК ИМПУЛЬСОВ НОМЕРА ДАТЧИКА.

;====================================

SEC

BTFSS PORTB,0 ;ДЛЯ ИСКЛЮЧЕНИЯ ИМПУЛЬСА ПОМЕХИ.

RETURN ;ИМПУЛЬС КОРОЧЕ 1,34 МС. ОТСЕКАЕМ.

CLRF SMC ;СЧЕТЧИК И ТАЙМЕР СБРАСЫВАЮТСЯ

CLRF TMR0 ;ПРИ КАЖДОМ ИМПУЛЬСЕ.

BTFSS FLAG,0

RETURN

INCF SCN,1 ;ПРИ ЕДИНИЧНОМ ФЛАГЕ СЧИТАЕМ ИМПУЛЬСЫ.

MOVLW.20 ;ЕСЛИ УЖЕ 20 И БОЛЬШЕ,

SUBWF SCN,0 ;

SKPNC

BCF PORTA,4 ;ВКЛЮЧИМ СИГНАЛ.

RETURN

;====================================

; 6. ПЕРЕЗАПИСЬ В РЕГИСТР ИНДИКАЦИИ.

;====================================

SES

MOVFW SCN ;ПЕРЕПИШЕМ ЗНАЧЕНИЯ СЧЕТЧИКА ИМПУЛЬСОВ

MOVWF SIN ;В РЕГИСТР ИНДИКАЦИИ.

MOVLW.12 ;ЕСЛИ УЖЕ 12 И БОЛЬШЕ,

SUBWF SIN,0 ;

SKPNC

CLRF SIN ;ОБНУЛИМ.

MOVLW.20 ;ЕСЛИ УЖЕ 20 И БОЛЬШЕ,

UBWF SCN,0 ;

SKPNC

BCF PORTA,4 ;ВКЛЮЧИМ СИГНАЛ.

CLRF SCN ;ОБНУЛИМ.

RETURN

;======================================================

; 7. СОХРАНЕНИЕ И ВОССТАНОВЛЕНИЕ ЗНАЧЕНИЙ РЕГИСТРОВ ПРИ ПРЕРЫВАНИИ.

;======================================================

CONST

MOVWF W\_TEMP ;СОХРАНЕНИЕ ЗНАЧЕНИЙ РЕГИСТРОВ W И

MOVFW STATUS ;STATUS,

MOVWF STATUS\_TEMP ;

MOVFW FSR ;FSR.

MOVWF FSR\_TEMP ;

BTFSS INTCON,1

GOTO S01 ;ПРИБАВИМ 1 В СЧЕТЧИК.

CALL SEC ;СБРОСИМ ТАЙМЕР ИЛИ УВЕЛИЧИМ СЧЕТЧИК.

RECONST ;ВОССТАНОВЛЕНИЕ СОХРАНЕННЫХ ЗНАЧЕНИЙ.

BCF INTCON,2 ;СБРОСИМ ФЛАГ ПО ПЕРЕПОЛНЕНИЮ ТАЙМЕРА.

BCF INTCON,1 ;СБРОСИМ ФЛАГ ПРЕРЫВАНИЯ ПО ВХОДУ RВ0.

MOVFW STATUS\_TEMP ;ВОССТАНОВЛЕНИЕ РЕГИСТРОВ:

MOVWF STATUS ;STATUS,

MOVFW FSR\_TEMP ;

MOVWF FSR ;FSR,

MOVFW W\_TEMP ;W.

; CLRF TMR0

RETFIE ;ВОЗВРАТ ИЗ ПРЕРЫВАНИЯ.

;========================

; 8. СЧЕТЧИК МИЛЛИСЕКУНД.

;========================

S01

INCF SMC,1 ;ЕСЛИ ДЕЛИТЕЛЬ НА 8, ТО БУДЕТ 1 СЕКУНДА.

MOVFW SMC ;

ADDLW -05H ;К=5. ЧТОБЫ ОБНУЛИТЬ

SKPNZ ;СЧЕТЧИК ПОСЛЕ

CALL SES ;АВАРИЙНОГО РЕЖИМА.

MOVFW SMC ;ПЕРЕЗАПИСЬ

ADDLW -07H ;К=7. ЧТОБЫ РАЗРЕШИТЬ ЗАПИСЬ

SKPNZ ;

BCF FLAG,0 ;В АВАРИЙНОМ РЕЖИМЕ.

SKPNZ ;ПОСЛЕ АВАРИЙНОГО РЕЖИМА

BSF PORTA,4 ;ВЫКЛЮЧИМ СИГНАЛ.

MOVFW SMC ;ЕСЛИ ДЕЛИТЕЛЬ НА 9, ТО БУДЕТ БОЛЬШЕ 1 СЕКУНДЫ.

ADDLW -09H ;К=9.

SKPNZ ;ПРИ ПЕРЕПОЛНЕНИИ СЧЕТЧИКА

BCF PORTA,4 ;ВКЛЮЧИМ СИГНАЛ.

SKPNZ

BSF FLAG,0 ;РАЗРЕШИМ СЧЕТ.

SKPNZ

CLRF SMC ;

GOTO RECONST

MOVFW SMC ;

ADDLW -0AH ;К=11.

SKPNZ

CLRF SMC ;ОБНУЛИМ СЧЕТЧИК МИЛЛИСЕКУНД.

GOTO RECONST

;=============================================

END

==============================================

**Приложение Г**



**Приложение Д**

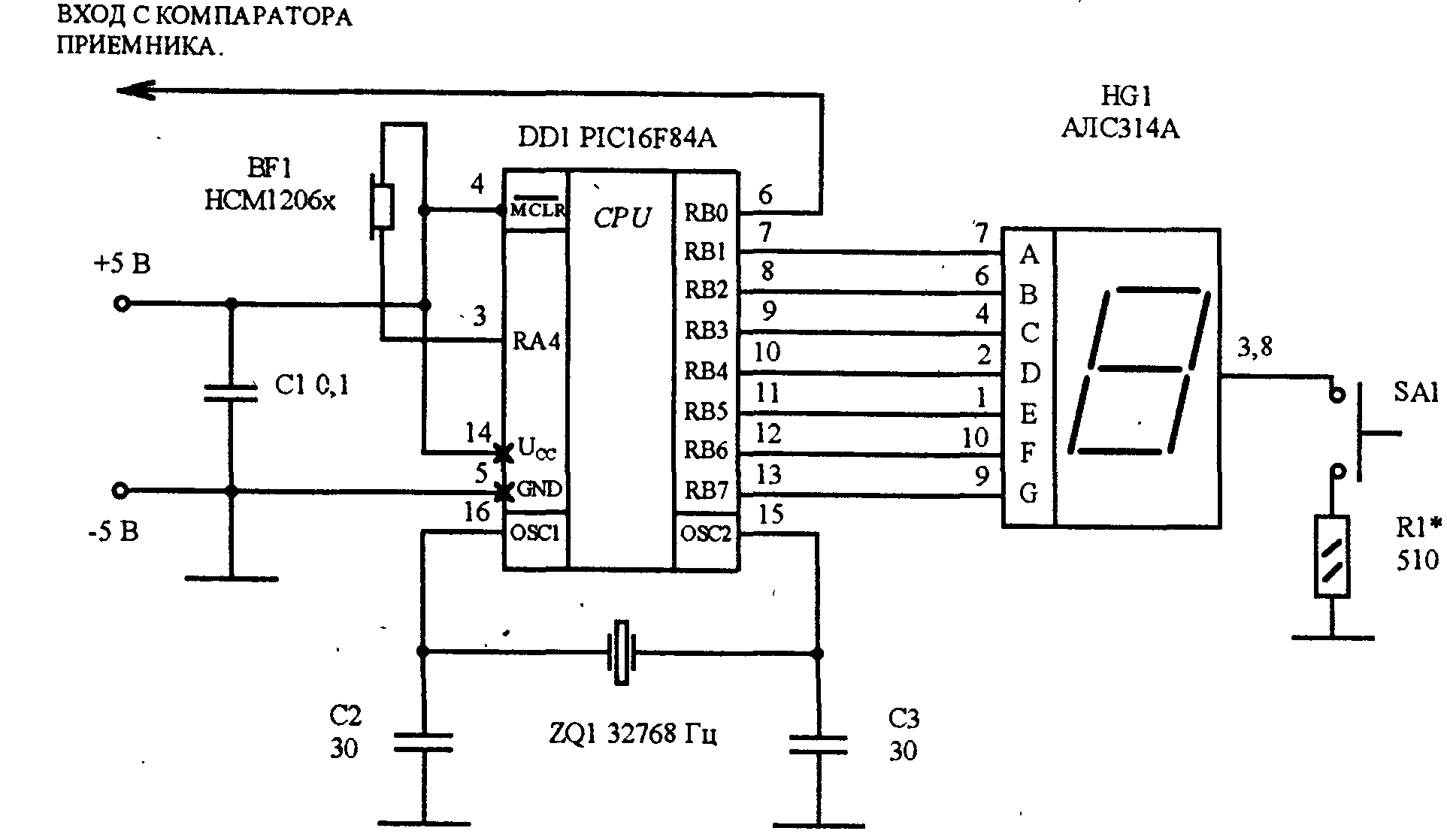


Схема принципиальная электрическая