

Введение

Современное промышленное производство является сложным комплексным процессом, который требует быстрого и многомерного контроля над всеми параметрами. Такой контроль был бы невозможен без применения современной электронной техники и автоматики вследствие того, что существуют многочисленные физические явления недоступные для простого визуального контроля. В настоящее время промышленная автоматика развивается значительными темпами, что связано с постоянно повышающимся уровнем сложности и качества технологических процессов. Электронные промышленные устройства являются сложными системами, в состав которых входят энергетические преобразователи, элементы электропривода, микропроцессорные узлы обработки информации и связи с внешними управляющими объектами, а также датчики различного назначения, устройства согласования с объектом управления.

Очевидно, что задача разработки автоматизированной системы регулирования (АСР) объекта включает в себя комплекс проблем, которые сами по себе представляют отдельную область современной автоматики. Поэтому создание таких разработок требует от студента хороших знаний принципов конструирования, функционирования серийных технических и программно-технических средств; структуры и функциональных возможностей программного обеспечения систем автоматизации.

Курсовая работа предусматривает решение основных задач реального инженерного проектирования электронной техники: выбор технических и программно-технических средств для создания АСР, проектную компоновку технических и программно-технических средств автоматизации, конструирование.

Целью данной курсовой работы является проектирование АСР на микропроцессорных элементах. Для достижения этой цели следует выполнить следующие пункты задания:

- составить структурную схему системы автоматизации;
- сформулировать задачу автоматизации;
- выбрать исполнительные и задающие элементы;
- выбрать и обосновать выбор микропроцессорного элемента управления;
- рассчитать нагрузочные характеристики элементов;
- составить электрическую схему и спецификацию элементов;
- составить алгоритмическую схему управления;
- написать программное обеспечение.

1 Порядок выполнения курсовой работы

1.1 *Структурная схема системы автоматизации* составляется с целью формулировки и конкретизации задачи управления. Обязательным компонентом решения задачи автоматизации в данной курсовой работе является использование микропроцессорного элемента управления. Тогда структурная схема (рисунок 1) должна включать в себя микропроцессорный элемент управления (микроконтроллер) - 1, входные элементы - 2 и исполнительные элементы - 3. В качестве входных элементов могут использоваться датчики, кнопки управления, клавиатура. В качестве исполнительных элементов могут использоваться катушки соленоидов, катушки реле, сирены, звуковые генераторы, маломощные двигатели, светодиоды, лампы. На структурной схеме необходимо указать направление сигналов и обратные связи.

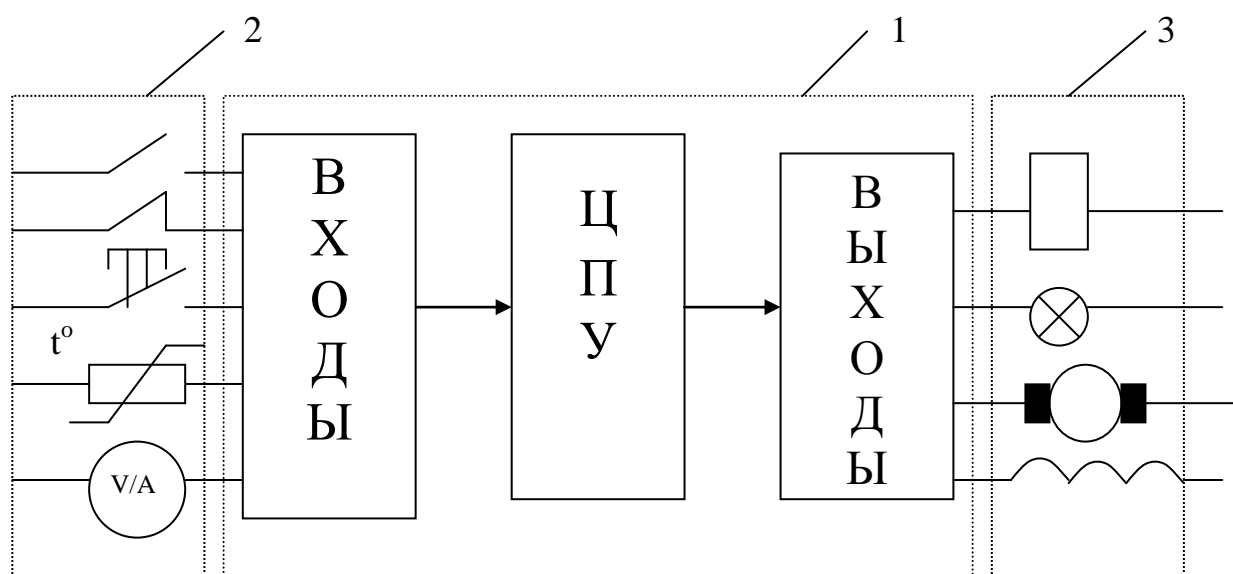


Рисунок 1 - Структурная схема системы автоматизации

1.2 *Формулировка задачи автоматизации* выполняется по следующему шаблону: требуется при следующих входных данных ... установить (сбросить) следующие выходные данные ..., согласно алгоритму ... (здесь следует словесное описание алгоритма работы системы).

1.3 *Выбор задающих и исполнительных элементов системы* производится из каталога [7], в соответствии с формулировкой задачи. При этом следует учитывать электрические характеристики выбираемых устройств (напряжение питания, сопротивление и токи выходных цепей).

1.4 *Выбор и обоснование выбора микропроцессорного элемента* выполняются на основании вышевыполненных пунктов. Необходимо подсчитать количество входных и выходных сигналов, определить вид каждого сигнала (цифровой, аналоговый или с термометра сопротивления), напряжение питания, нагрузочный ток каждого из подключаемых элементов. Здесь должны быть указаны преимущества выбираемого микропроцессорного элемента в сравнении с другими, его электрические характеристики,

экономические показатели, программные возможности и особенности программирования, а также характеристики надежности.

1.5 *Рассчитать нагрузочные характеристики микроконтроллера* следует с учетом произведенного выбора.

Каждое из подключаемых к контроллеру устройств является нагрузкой на его портах. Выходной ток I_{Π} по каждому порту будет в паспортных данных контроллера. Также указано и внутреннее сопротивление подключаемого устройства R_{Π} . Напряжение питания V_{DD} контроллера и подключаемого устройства должны быть равными или находиться в совместимых пределах. Тогда нагрузочный ток по выходу контроллера рассчитывается по закону Ома:

$$I_{\Pi} = \frac{U_{DD}}{R_{\Pi}}, \text{ А.} \quad (1)$$

При этом рассчитанное значение не должно превышать допустимого значения

$$I_{\Pi} \geq I_{\Pi},$$

где i – номер входного или выходного канала.

1.6 *Составление электрической схемы и спецификации элементов* является основной частью выполнения курсовой работы. Далее приведены основные рекомендации по составлению электрической схемы.

1.6.1 Обязательным элементом является тактирование микроконтроллера. Существуют три схемы тактирования PIC-микроконтроллеров (рисунок 2). Для версий с кварцевым или керамическим резонатором используют схему, изображенную на рисунке 2,а. Значение резисторов C1 и C2 выбирается в зависимости от типа резонатора (кварцевый или керамический) и частоты (таблица 1). Для версии XT резистор R1 не нужен, однако иногда он требуется для микроконтроллеров версии HS. Только точное знание характеристик кварцевого резонатора позволяет определить необходимость в использовании резистора R1, и каким должно быть его значение.

Схема на рисунке 2,б представляет собой реализацию RC-генератора. В этом случае для собственно генерации используется вывод OSC1. Вывод OSC2 является выходом внутренней рабочей частоты микроконтроллера (частоты командных циклов), которая в четыре раза меньше, чем частота генератора.

Стабильность RC-генератора не такая высокая, как у кварцевого. Чтобы на нее не оказывали сильного влияния внешние факторы и внутренние характеристики самой микросхемы, фирма Microchip рекомендует применять резистор с сопротивлением от 5 до 200 кОм и конденсатор емкостью не менее 20 пФ.

Схема на рисунке 2,в демонстрирует способ тактирования PIC* внешним генератором. Понятно, что формируемые внешним генератором уровни должны соответствовать напряжению питания микроконтроллера.

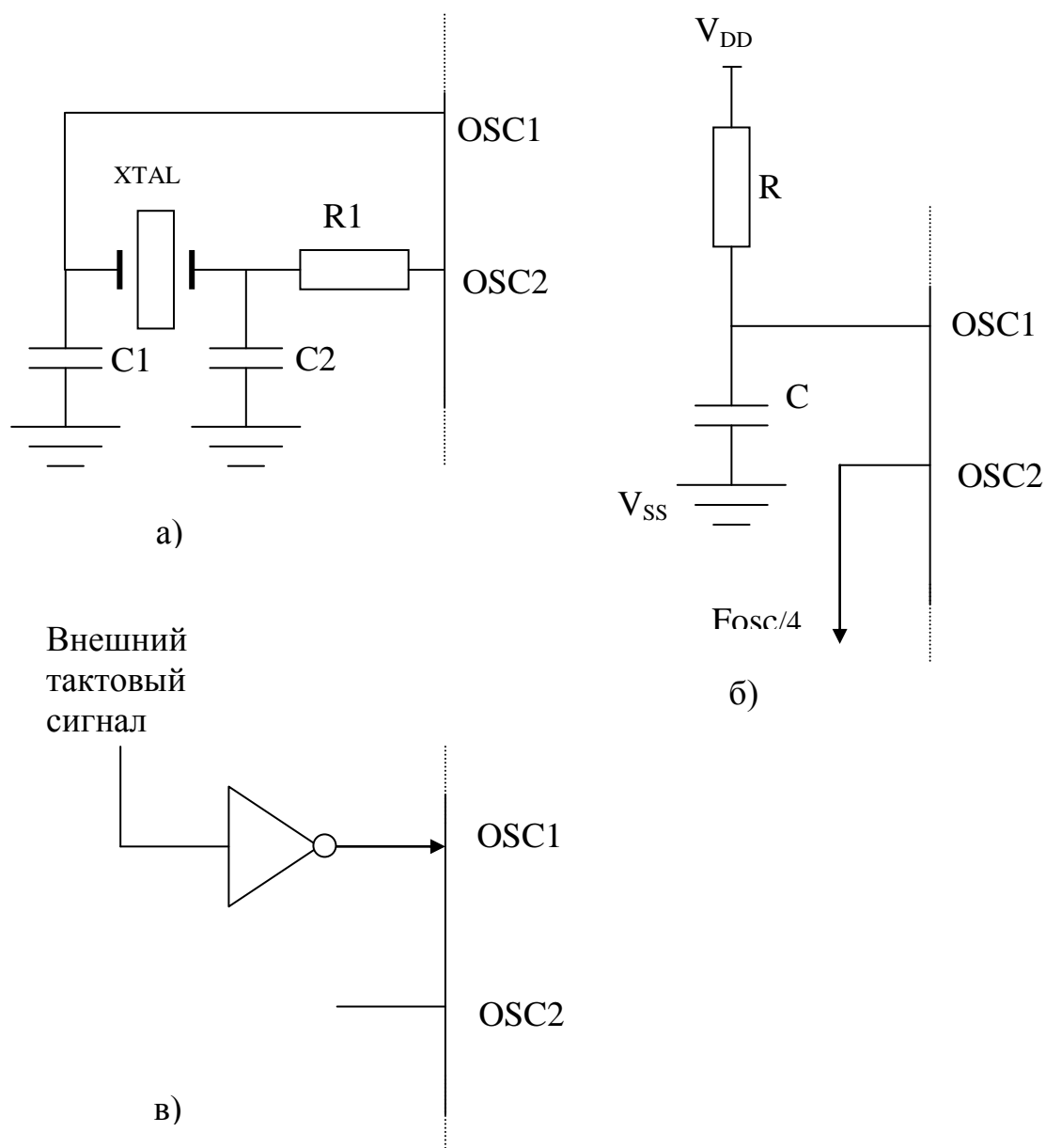


Рисунок 2 - Схемы тактирования PIC-микроконтроллеров

1.6.2 Также следует предусмотреть возможность сброса контроллера. Все микроконтроллеры имеют вывод сброса, называемый чаще всего MCLR. У PIC* – контроллеров внутренняя схема сброса работает, если скорость роста напряжения питания достаточно высока (обычно выше 0,05 В/мс). Упрощенная структурная схема сброса показана на рисунке 3.

Если напряжение питания растет медленно, то требуется ручной сброс (рисунок 4). Следует обратить внимание на резистор R1, значение которого может варьироваться от 100 Ом до 1 кОм. Он служит для защиты входа MCLR микроконтроллера от положительного напряжения на конденсаторе C при выключении питания.

Таблица 1 – Параметры конденсаторов для кварцевого резонатора

Тип резонатора	Частота кГц	C1, пФ	C2, пФ
LP – низкочастотный кварцевый резонатор	32	15	15
	100	15	15
	200	0-15	0-15
ХТ – стандартный кварцевый резонатор с максимальной частотой 4 МГц	100	15-30	200-300
	200	15-30	100-200
	455	15-30	15-100
	1000	15-30	15-30
	2000	15	15
	4000	15	15
HS – кварцевый резонатор с высокой частотой	4000	15	15
	8000	15	15
	20000	15	15

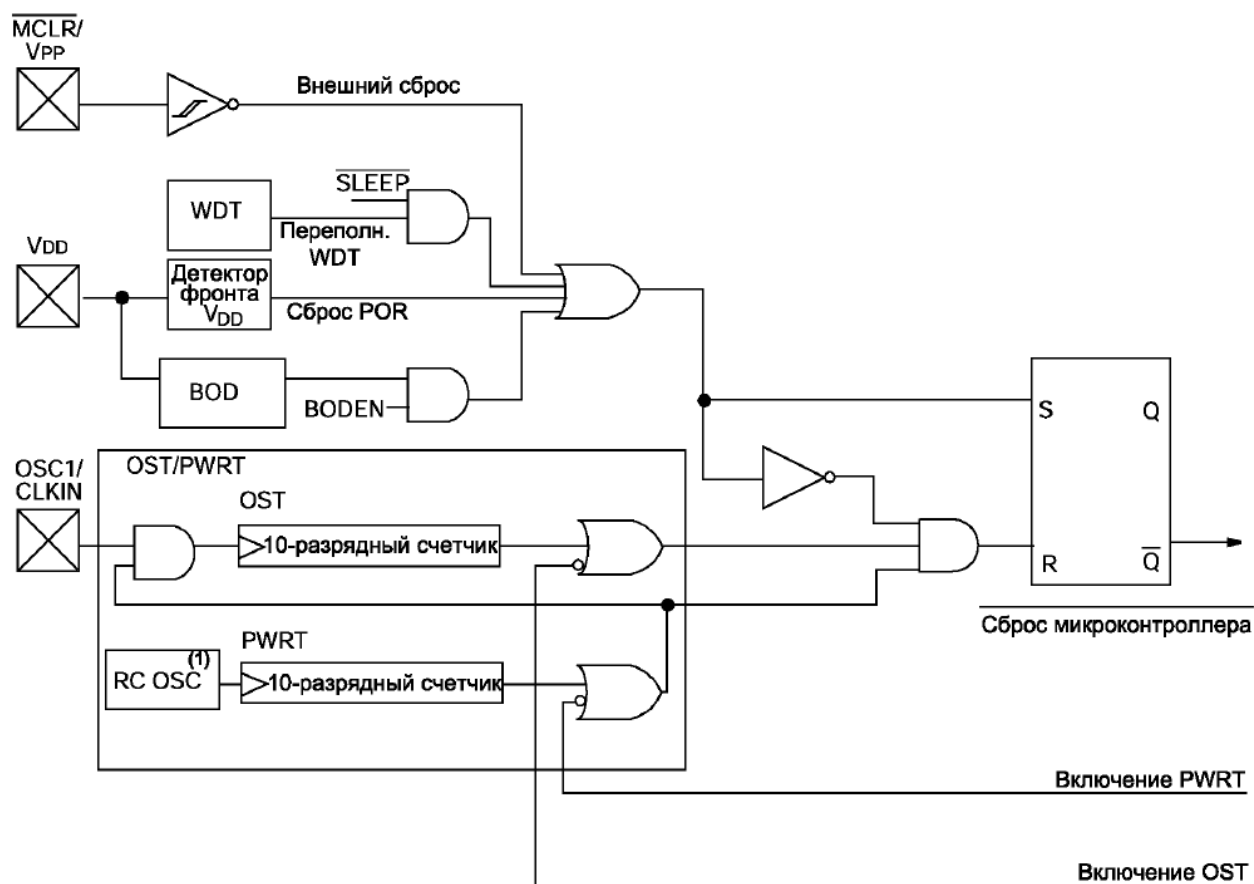


Рисунок 3 – Упрощенная структурная схема сброса

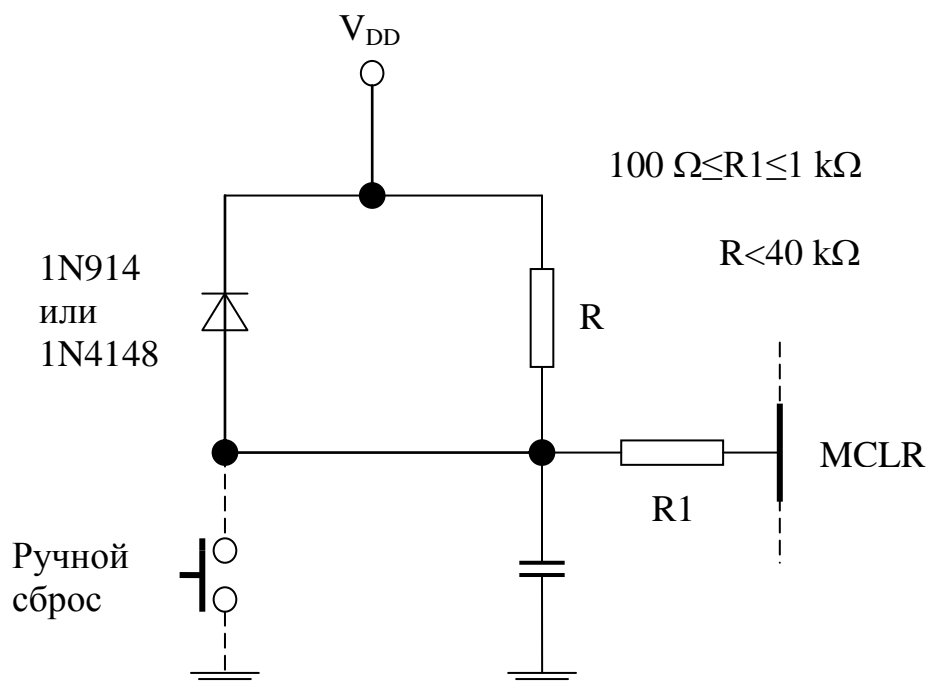


Рисунок 4 – Схема ручного сброса

1.6.3 Подключение к контроллеру различных устройств следует начать с рассмотрения схем подключения светодиодов. Если выходы микроконтроллера рассчитаны на ток большой силы, то светодиод можно подключать к выводам микроконтроллера через ограничивающий резистор (рисунок 5). Допустимая сила тока каждого выхода параллельного порта контроллеров, например, составляет 20 мА, что вполне достаточно для зажигания одного светодиода. Но суммарный ток порта не должен превышать 50 мА. В рассматриваемом случае используется светодиод на 10 мА, чтобы не превышать этого максимума.

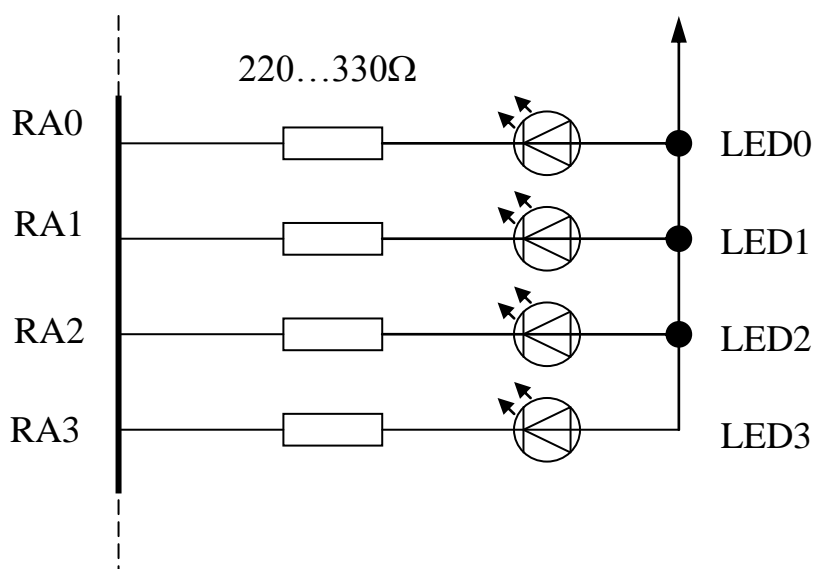


Рисунок 5 – Схема подключения светодиодов

Если требуется, чтобы индикация была хорошо видна, следует использовать светодиоды высокой яркости или применить схему, показанную на рисунке 6, в которой к выходу микроконтроллера подключен усилитель на транзисторе. Ограничительный резистор выбирается в зависимости от силы тока. Учитывая значение сопротивления в цепи базы транзистора и его коэффициент усиления, через светодиод можно получать ток, величиной 100 мА и более, что вполне достаточно.

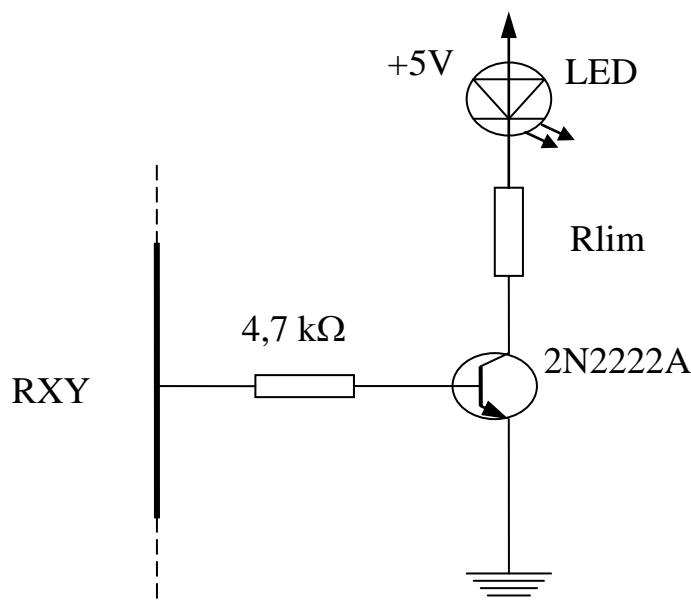


Рисунок 6 – Схема подключения светодиодов

Цифровой светодиодный индикатор представляет собой несколько объединенных в одном корпусе одиночных светодиодов. Поэтому принципы управления цифровым светодиодным индикатором и простым светодиодом аналогичны. В зависимости от количества индикаторов могут применяться различные варианты схем управления.

Если необходима индикация с несколькими цифровыми разрядами, придется использовать динамическое управление, предполагающее быструю коммутацию индицируемых разрядов (временное уплотнение). Динамическое управление экономит порты микроконтроллера. Общий принцип работы такой схемы представлен на рисунке 7.

Сегменты индикаторов всегда управляются восемью линиями параллельного порта (напрямую или через соответствующие усилители), в то время как объединенные катоды (общие катоды) по очереди коммутируются транзисторами через другой порт. Могут быть использованы и индикаторы с общим анодом, при условии замены в схеме n-p-n на транзисторы p-n-p и соединении их эмиттеров не с нулевым потенциалом, а с плюсом питания. Временная диаграмма управления динамической индикацией приведена на блок-схеме (рисунок 8). Порт В в этом случае используется для управления сегментами, а управление цифрами ведется через порт А. Индикация цифр

осуществляется поочередно за счет переключения соответствующих разрядов порта А. Для нормального функционирования индикатора микроконтроллер должен обеспечить достаточную частоту переключения цифр, которая не была бы заметна для глаз (не менее 40 Гц, то есть время цикла должно быть не менее 25 мс).

Другое решение - использование специализированного контроллера индикации (рисунок 9). Контроллер индикации осуществляет не только декодирование двоично-десятичного кода в семисегментный, но и реализует циклограммы необходимых при динамической индикации переключений. Микроконтроллер управления индикацией предполагает матричный доступ к каждому сегменту. Это сокращает общее число необходимых линий и соответственно выводов микросхемы, используемых для управления индикатором. Упрощается и программное обеспечение микроконтроллера: его задача в части поддержки индикации сводится к посылке на микросхему последовательности цифр, которые надо вывести на индикатор. Если приложение требует более четырех индикаторов, можно использовать несколько микросхем.

Схема на рисунке 10 использует жидкокристаллический индикатор фирмы Hitachi, хотя может быть применен любой другой индикатор этого типа, так как сигналы интерфейса у различных марок почти идентичны. Данные на индикатор подаются по восьми линиям порта В (DB0-DB7) PIC – контроллера 16C54, а сигналы трех линий управления формируются через порт А:

- линия Е (Enable) – высокий уровень сигнала – на данной линии разрешает выполнение операции обмена, при этом индикатор может получать команды или данные. Нулевой уровень запрещает доступ к индикатору;
- линия R/W (Read/Write) указывает тип операции при обращении к индикатору (запись или чтение данных). Индикатор имеет внутренний регистр состояния, информация из которого может быть считана;
- линия RS (Register Select) определяет тип передаваемой информации: команды (RS=0) или данные (RS =1).

В режиме передачи данных индикатор принимает и отображает знаки кода ASCII, полученные от микроконтроллера. Позиция курсора при этом изменяется автоматически.

Индикатор может выполнять некоторое количество команд, обеспечивающих стирание отдельных символов, полное стирание всей информации, указание позиции курсора и т.д. Эти команды ускоряют управление индикатором. Некоторые модели индикаторов располагают даже памятью, где хранится конфигурация отображаемых знаков (знакогенератор). С её помощью можно изменять форму знаков.

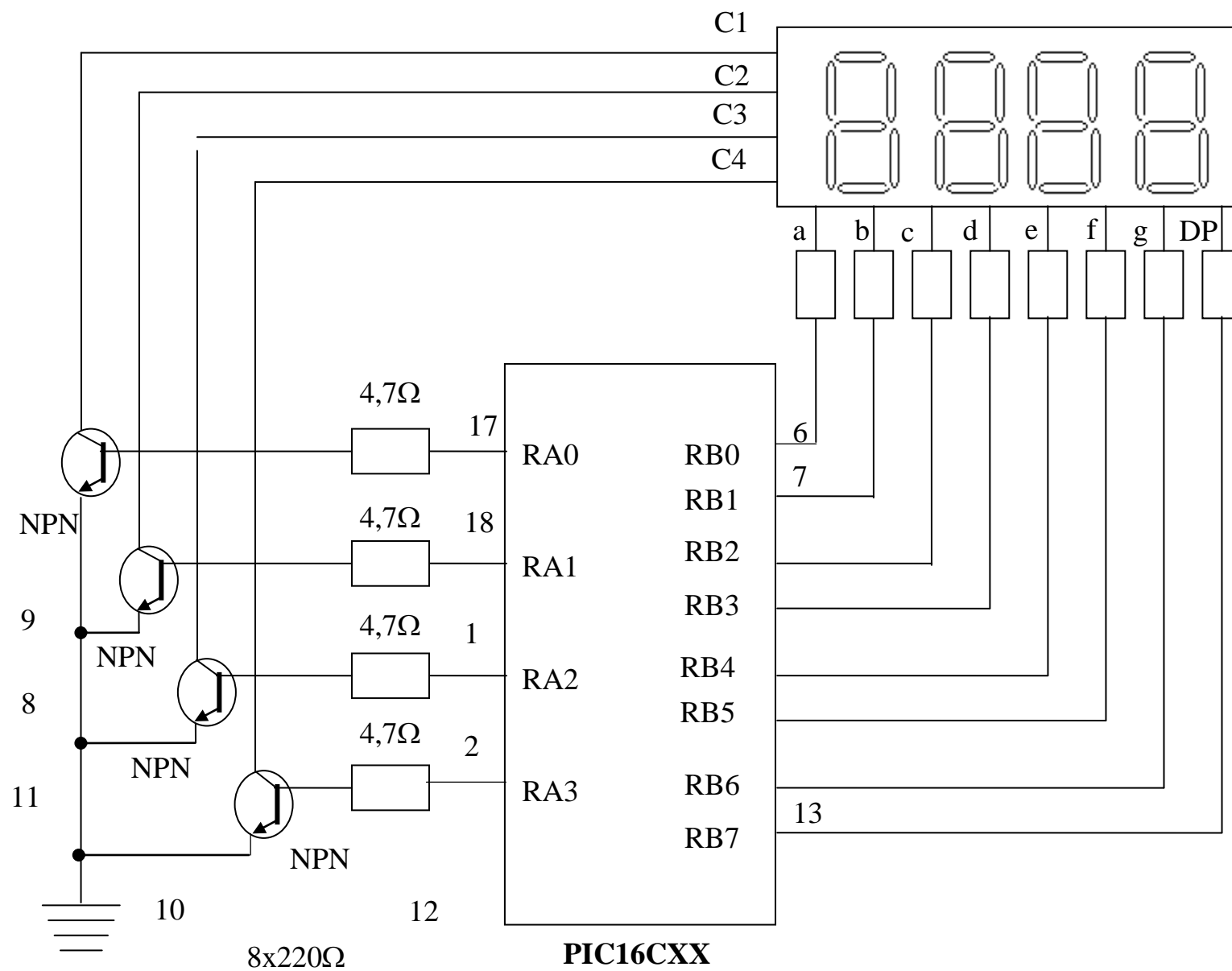


Рисунок 7 – Общий принцип работы схемы с коммутацией индицируемых разрядов



Рисунок 8 – Блок-схема временной диаграммы управления динамической индикацией

Диалог с таким индикатором предельно прост. Запись информации в индикатор происходит, например, в следующей последовательности:

- установить сигнал линии R/W в значение ноль;
- указать состояние линии RS, которое должно определять тип передаваемой информации (данные или команда);
- установить код данных или команды по шине DB0-DB7;
- установить уровень логической единицы на линию E, разрешая индикатору принять информацию;
- обнулить сигнал линии E, заканчивая обмен.

Пример программы индикатора и схема подключения ещё одного жидкокристаллического индикатора приведен в приложении А, а также в качестве тестовой программы ind1.asm в папке MPLAB лабораторного комплекса УМК-7.

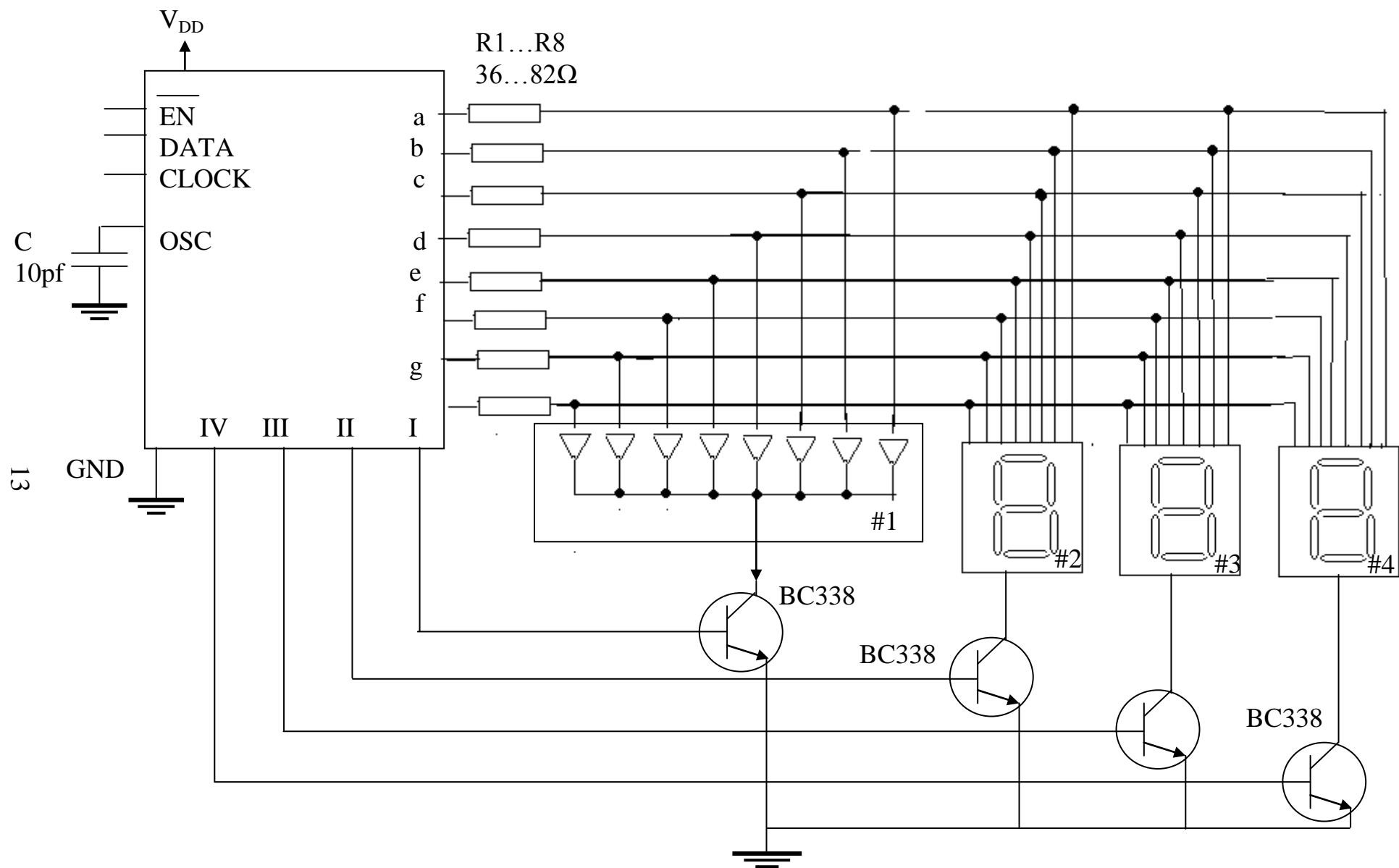


Рисунок 9 – Схема с использованием специализированного контроллера индикации

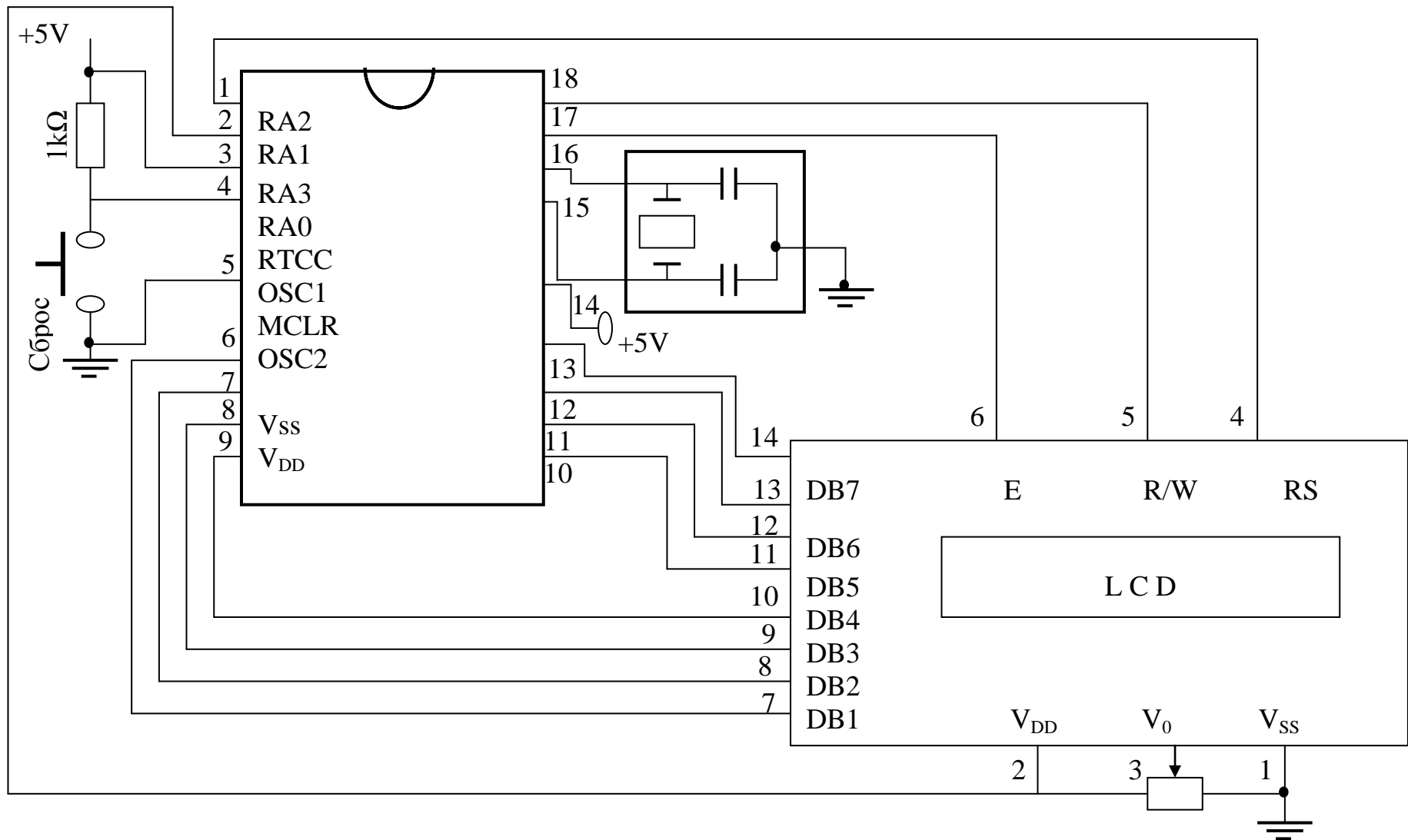


Рисунок 10 – Схема с использованием жидкокристаллического индикатора

Когда микроконтроллер должен получать информацию от устройств, находящихся под высоким напряжением или связанных с электрической сетью, самое лучшее решение состоит в том, чтобы обеспечить гальваническую развязку входа, например, посредством оптрона. Данный принцип иллюстрируется схемой, представленной на рисунке 11. Когда на внешнюю часть схемы подается напряжение, через светодиод оптрона проходит ток и фототиристор оптрона открывается, переводя вход микроконтроллера на низкий логический уровень. Для нормальной работы схемы необходимо, чтобы протекающий через светодиод ток не превышал предельно допустимый, но был достаточным для перевода фототранзистора в режим насыщения, который гарантирует получение на входе микроконтроллера низкого логического уровня. Это требование обеспечивается выбором соответствующего оптрона или использованием усилителя.

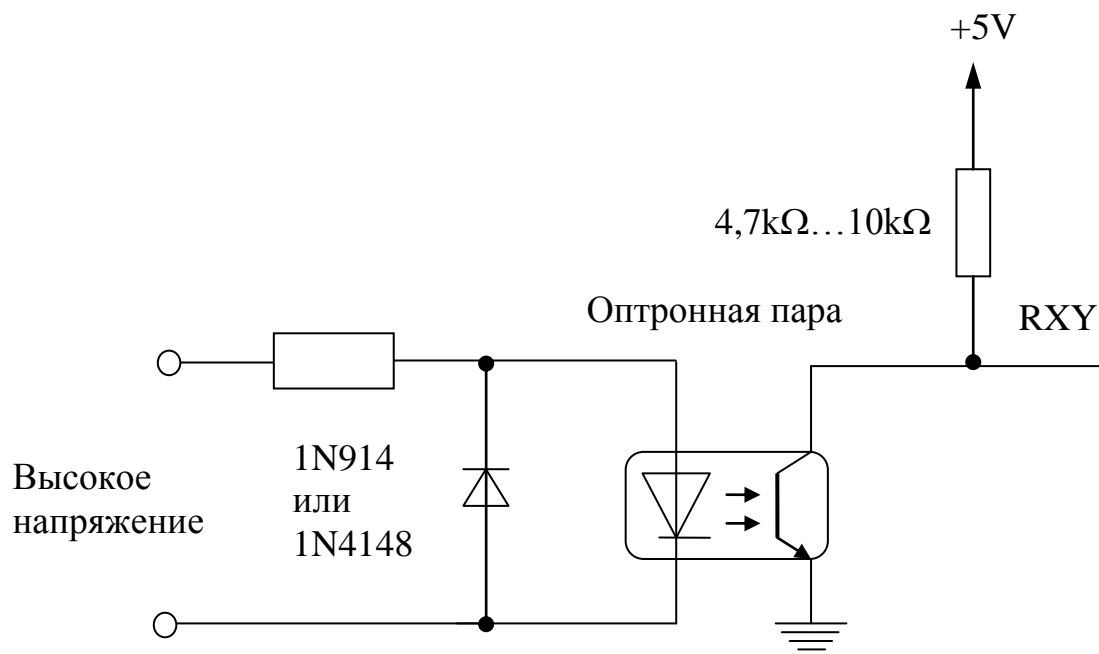


Рисунок 11 – Схема гальванической развязки входа с помощью оптрона

1.6.4 Составление схемы подключения клавиатуры [7]. Если клавиатура состоит из нескольких клавиш, они могут быть подключены к микроконтроллеру как отдельные кнопки, то есть каждый через свой порт (лабораторная №7 УМК-7). Если клавиатура большая, необходимо искать другое решение, поскольку портов у микроконтроллера не слишком много. Здесь возможно следующее решение – использование микроконтроллера в матричных клавиатурах. На рисунке 12 представлен вариант подключения к микроконтроллеру матричной клавиатуры с шестнадцатью клавишами, причем их число может быть легко увеличено.

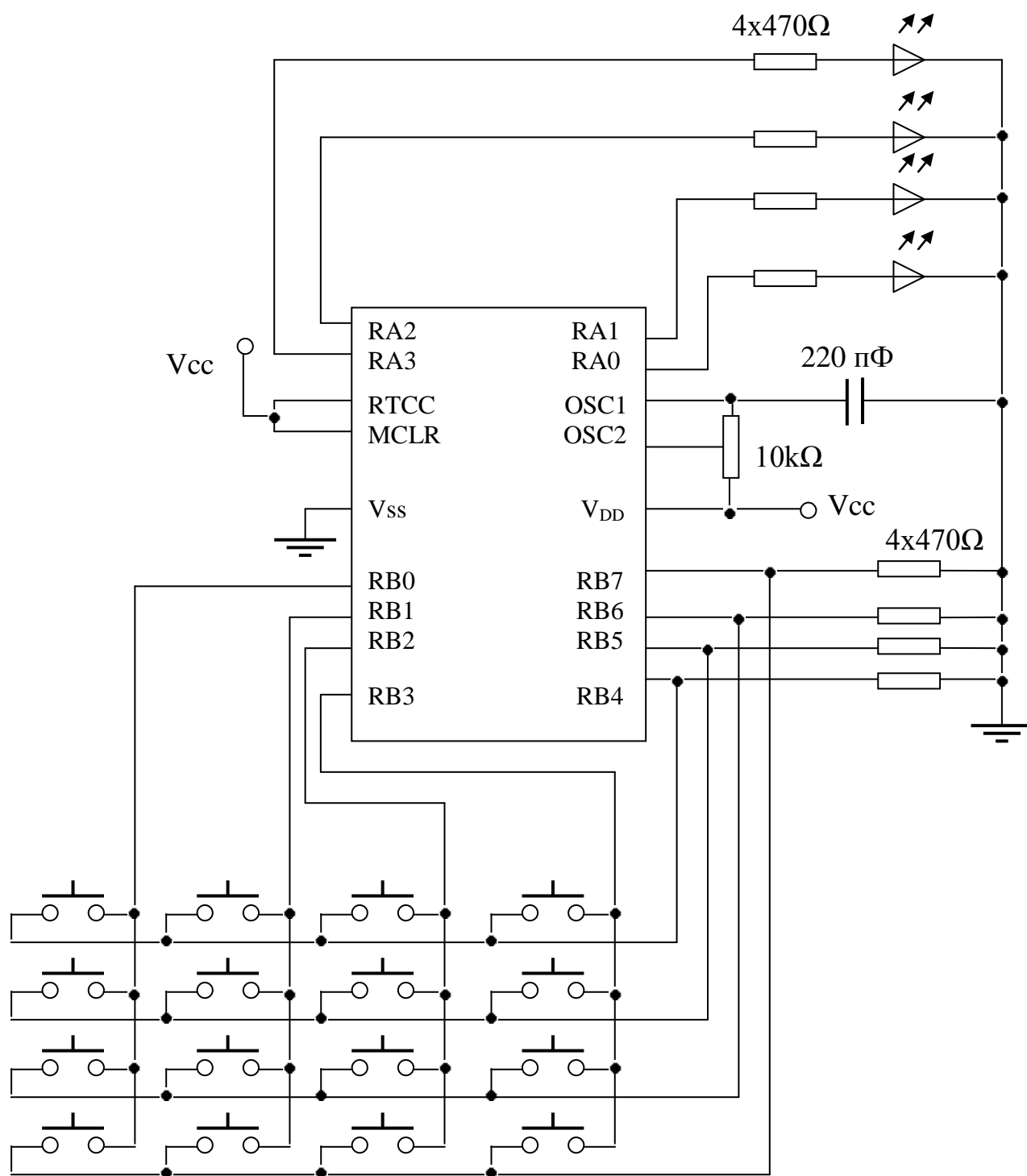


Рисунок 12 – Схема подключения матричной клавиатуры

Клавиши находятся на пересечении строк и столбцов матрицы. При нажатии на клавишу происходит замыкание соответствующей строки и столбца. Программа по номерам строки и столбца может определить, какая клавиша была нажата. Программа работает следующим образом. Линии столбцов соединены с портами RB0-RB3 микроконтроллера, являющимися выходами, линии строк - напротив подключены к входным портам RB4-RB7. Программа осуществляет сканирование клавиатуры по строкам, определяя

момент появления логической единицы на одном из входных портов. В процессе сканирования непрерывно инкрементируется переменная *key*. При обнаружении логической единицы значение *key* определяет номер нажатой клавиши.

Сканирование столбцов осуществляется путем изменения позиции единицы в четырехразрядном позиционном коде, то есть путем последовательной выдачи на линии столбцов кодов 0001, 0010, 0100 и 1000. При каждом коде производится сканирование строк. Если какая-либо клавиша была нажата, то в одной из строк будет обнаружена единица. Сканирование в этот момент завершается, а значение переменной *key* идентифицирует клавишу.

До начала сканирования значение переменной *key* устанавливается равной нулю. Если ни одна клавиша не нажата, программа возвращает в качестве значения переменной *key* число 16 (10h). Программа осуществляет индикацию номера клавиши в двоичном коде с помощью четырех светодиодов, подсоединенных к порту А.

Программа матричной клавиатуры:

```
include <p16F877.inc>
;Описание переменных программы
        KEYPAD    EQU    RB    ;
        ROW1      EQU    RB4    ;
        ROW2      EQU    RB5    ;
        ROW3      EQU    RB6    ;
        ROW4      EQU    RB7    ;
;Определение переменных величин
        COLS      EQU    H'20'
        KEY       EQU    H'21'
        INDEX     EQU    H'22'
        ORG 00h
        nop
        nop
        nop
        ORG 05h
START
        CLRF      STATUS
        BSF       STATUS,RP0
        MOVLW     B'11110000'
        MOVWF     TRISB
        CLRF      TRISA
KEYS
        CALL      SCANKEYS
        MOVLW     H'10'
        SUBWF     KEY,0
```


	BTFC	STATUS,Z
	GOTO	DELAY
	MOVF	KEY,0
	MOVWF	PORTA
DELAY		
	nop	
	nop	
	DECFSZ	INDEX,1
	GOTO	DELAY
	GOTO	KEYS
	;Подпрограмма сканирования клавиатуры	
	SCANKEYS	
	CLRF	KEY
	CLRF	KEYPAD
	MOVLW	D'4'
	MOVWF	COLS
	BSF	STATUS,C
SCAN		
	RLF	KEYPAD
	BCF	STATUS,C
	BTFSC	PORTB, ROW1
	GOTO	PRESS
	INCF	KEY
	BTFSC	PORTB, ROW2
	GOTO	PRESS
	INCF	KEY
	BTFSC	PORTB, ROW3
	GOTO	PRESS
	INCF	KEY
	BTFSC	PORTB, ROW4
	GOTO	PRESS
	INCF	KEY
	DECFSZ	COLS,1
	GOTO	SCAN
PRESS	RETLW	
	END;	

1.7 Составить алгоритмическую схему управления следует согласно словесному описанию алгоритма, выполненному в пункте 1.2. Сначала составляется блок-схема решения задачи в укрупненном виде, а затем детализируется решение задачи алгоритмами вида, приведенного на рисунке 8. Детализированные алгоритмы решения задач следует составлять с учетом выбранного контроллера и его программного обеспечения. Здесь имеет

значение, какие именно операторы имеются в составе программного обеспечения микроконтроллера, как настраиваются порты микроконтроллера, какие настройки дополнительно необходимо установить, какие режимы работы микроконтроллера следует предусмотреть (сброс, прерывание, режимы низкого энергопотребления и другое).

1.8 *Написать программное обеспечение* следует в точном соответствии с результатами предыдущего пункта задания. Программное обеспечение представляется в виде листингов программ. Примеры программ приведены в методических указаниях к лабораторным работам для комплекса УМК-7 [1].

2 Перечень заданий к курсовой работе

Таблица 2 – Варианты заданий

Вариант	Задание
1	<p>На входной двери установлен блокиратор, световой датчик и таймер. Реализовать следующий алгоритм:</p> <ul style="list-style-type: none"> - дверь блокируется и разблокируется по сигналу оператора; - при отключенном таймере он включается вновь на определенное время при отсутствии сигнала блокировки оператора; - пока включен таймер, идет подсчет количества посетителей; - если за установленное время количество посетителей превосходит 10, таймер отключается и дверь блокируется.
2	<p>Датчик температуры стоит внутри помещения. На кондиционере выставляется значение температуры. Реализовать следующий алгоритм:</p> <ul style="list-style-type: none"> - если температура в помещении ниже заданной на 5°C, отключить кондиционер; - если температура в помещении выше или равна заданной температуре, включить кондиционер; - держать включенное состояние кондиционера не больше 30 минут; - повторно включить кондиционер через 10 минут.
3	<p>Осуществить регулирование освещением от трех ламп по следующему алгоритму:</p> <ul style="list-style-type: none"> - при поступлении с датчика на входе включить первую лампу через 5 секунд; - при поступлении сигнала с середины комнаты включить вторую лампу и оставить включенным обе лампы до срабатывания сигнала выключить; - при поступлении дополнительно трех сигналов с первого датчика оставить включенным первую лампу на час; - при поступлении дополнительно трех сигналов со второго датчика оставить включенным вторую лампу не менее часа.

4	<p>Реализовать управление светофором поезда на пути с тремя ветвями. Датчики стоят на каждом пути за 100 м до светофора. Реализовать следующие алгоритмы:</p> <ul style="list-style-type: none"> - зажечь зеленый, если нет сигналов с датчиков двух других ветвей; - зажечь красный, если есть хотя бы один сигнал с датчиков других ветвей; - разрешить переключение с красного света на зеленый только в случае сброса с панели оператора.
5	<p>На двери автомобиля установлен блокиратор. Установить контроллер со следующими входными данными: кнопки от реле включения двигателя, кнопка пульта водителя. Реализовать следующий алгоритм:</p> <ul style="list-style-type: none"> - блокировка двери автоматически при включении двигателя; - блокировка двери от пульта оператора; - автоматическое открывание двери через 20 с после отключения двигателя.
6	<p>Разработка замка для подъезда. Кодовый замок содержит 9-ти клавишную клавиатуру. Устройство позволяет хранить 99 кодовых ключей, кодовые комбинации представляет собой 4-х значные комбинации десятичных чисел.</p>
7	<p>Имеется семисегментный индикатор. Подключить его к контроллеру и реализовать вывод на нем цифр 0-9 при наборе с клавиатуры цифр.</p>
8	<p>Реализовать управление с дистанционного пульта, на котором имеется четыре кнопки движения: «вперед», «назад» «влево», «вправо» – управление движением небольшого манипулятора мощностью двигателя 50Вт по следующему алгоритму:</p> <ul style="list-style-type: none"> - при движении «вперед» блокируется движение «назад» и наоборот; - при движении «вправо» блокируется движение «влево» и наоборот; - при прокручивании колес, которое определяется датчиком движения на ободе колеса, блокируются все кнопки.
9	<p>Регулировать движением подачи станка по следующему алгоритму:</p> <ul style="list-style-type: none"> - включить двигатель подачи, если на подаче есть деталь, нет аварийного сигнала питания и включено питание двигателя сверла; - отключить подачу при совпадении сигнала датчика положения подачи с заданным сигналом оператора; - разрешить включить двигатель подачи детали повторно, если сверло остановилось.

10	<p>Реализовать алгоритм движения электрода сварочного аппарата по квадрату при следующих условиях:</p> <ul style="list-style-type: none"> - температура электрода не превышает 200°C; - сигнал оператора совпадает с сигналом датчика; - установлен зажим электрода.
11	<p>Создать контроллер для управления дверью гаража, реализующего следующий алгоритм:</p> <ul style="list-style-type: none"> - при въезде идет подсчет количества автомобилей; - при выезде количество уменьшается; - если количество автомобилей больше 50, запрет на въезд и блокировка дверей гаража.
12	<p>Имеется четыре кнопки набора номера, датчик открывания двери и дверной замок. Необходимо реализовать следующий алгоритм:</p> <ul style="list-style-type: none"> - при правильном наборе номера из двух кнопок дверь открывается; - при ошибочном номере дверь блокируется; - при открывании двери без набора номера дверь снова блокируется.
13	<p>Реализовать модель автоматического нагрева воды в титане по следующему алгоритму:</p> <ul style="list-style-type: none"> - в определенное время включить питание нагрева; - при поступлении сигнала от датчика верхнего значения температуры отключить питание; - при поступлении сигнала с нижнего значения температуры включить питание нагрева.
14	<p>Манипулятор выполняет операции с тремя степенями свободы: вперед - назад; влево - вправо; вращение. На пульте оператора расположить эти кнопки и 4 кнопки готовых алгоритмов:</p> <ul style="list-style-type: none"> - алгоритм 1: вправо – поворот – назад; - алгоритм 2: вперед – поворот – вперед; - алгоритм 3: влево – назад – вправо. - алгоритм 4: назад – поворот – назад. <p>Обеспечить управление в ручном режиме и в автоматическом по четырем алгоритмам.</p>
15	<p>На выход контроллера подключить катушки реле на 5 В. Реализовать алгоритм генератора импульсов с требуемой длительностью с шагом 10 мс от трех кнопок:</p> <ul style="list-style-type: none"> - кнопка задания длительности импульса 10 мс; - копка задания длительности импульса 1 с; - копка задания длительности импульса 10 с; - кнопка вывода импульсов в порт для управления двигателем; кнопка сброса импульсов на выходе.

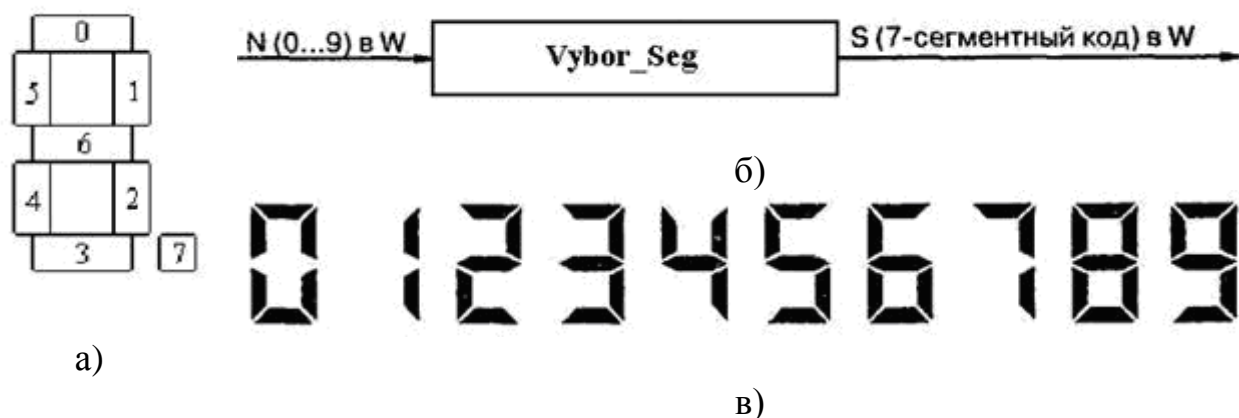
16	<p>Включить насос при следующих условиях:</p> <ul style="list-style-type: none"> - включена кнопка «пуск»; - отключена кнопка «стоп»; - нет аварийного сигнала с датчика двигателя; - нет сигнала с датчика уровня жидкости. <p>Отключить насос при обратных значениях этих сигналов.</p>
17	<p>Имеется резервуар для заполнения жидкости. В резервуаре установлены датчики температуры и уровня. Реализовать с помощью контроллера следующий алгоритм:</p> <ul style="list-style-type: none"> - при превышении уровня открыть задвижку для слива; - при уменьшении уровня ниже контрольного открыть задвижку заполнения жидкости; - при охлаждении температуры жидкости ниже заданной открыть задвижку заполнения жидкости и одновременного слива.
18	<p>Включить аварийную лампу сигнализации при следующих условиях:</p> <ul style="list-style-type: none"> - выход значения давления за заданный уровень; - при отключении двигателя насоса; - при превышении температуры среды заданного значения; - при появлении одновременно сигналов задания температуры и давления. <p>Отключить аварийную лампу при обратных значениях этих сигналов.</p>
19	<p>Реализовать модель тренажера со следующим алгоритмом:</p> <ul style="list-style-type: none"> - при включении кнопки «нагрузка» включается полная нагрузка; - через 10 минут выключается половина нагрузки; - через 20 минут ещё четверть; - через 30 минут нагрузка снимается и может быть включена вновь только через 10 минут.
20	<p>Светофор работает в обычном режиме, зажигая каждый цвет в течении 3 минут и при наличии следующих условий:</p> <ul style="list-style-type: none"> - нет разрыва в цепи светофора, работающего в противоположном режиме; - нет сигнала оператора на включение красного сигнала; - есть сигнал наличия питания на соседних светофорах.
21	<p>Создать индикатор для стиральной машины, на который будет выводиться название режима стирки и соответствующие этому режиму параметры: время стирки и температура воды. Выбор режима осуществляется с помощью клавиатуры.</p> <p>4 кнопки клавиатуры</p>
22	<p>Систем жизнеобеспечения космонавтов.</p> <p>Небольшое помещение, в котором установлен датчик давления воздуха, индикатор в виде светодиода.</p>

23	<p>Включить конвейер при следующих условиях:</p> <ul style="list-style-type: none"> - включена кнопка «пуск»; - отключена кнопка «стоп»; - имеется хоть одна деталь на конвейере; - количество деталей на конвейере не больше 8. <p>Отключить конвейер при обратных значениях этих сигналов.</p>
24	Автоматические ворота перед въездом в туннель. Проезд по туннелю односторонний (въезд машин)
25	Автоматический шлагбаум. Проезд односторонний (въезд машин)

Приложение А

Применение сегментного индикатора

Большинство цифровых индикаторов работают по принципу выборочного включения требуемых сегментов, как показано на рисунке А.1, а. Цифрами обозначены разряды порта передающего информацию. Обычно эти сегменты представляют собой светодиоды или электроды элементов на жидких кристаллах.



а – выборочное включение сегментов; б – системное представление; в – знаки семисегментного индикатора.

Рисунок А.1 – Семи сегментный индикатор

Системное представление разработанной подпрограммы приведено на рисунке А.1, б. Входным сигналом в данном случае является 4-битный двоичный код, находящийся в рабочем регистре W. Выходным значением, также возвращаемым в W, является соответствующий 7-битный код, необходимый для отображения соответствующей цифры (подпрограмма **Vyb_Seg**). В микроконтроллерах PIC гарвардская архитектура делает невозможным использование значений памяти программ в виде данных. Вместо этого таблицы преобразования реализуются в виде наборов команд **retlw**, каждая из которых возвращает однобайтную константу (подпрограмма **Vybor_Seg**). Седьмой бит используется для точки.

Применяемые символьные имена регистров показаны на рисунке А.2.

Подпрограмма **Vybor_Seg** осуществляет выборку строки таблицы, прибавляя число N, передаваемое через рабочий регистр W, к младшему байту счетчика команд (регистр PCL, который расположен по адресу h'02'). Поскольку PCL при вызове подпрограммы **VYBOR_SEG** уже указывает на первую команду **retlw** с кодом для цифры «0», то даже при W=0 будет выбрана цифра «0». После прибавления N к W, PLC будет указывать на N-ю команду, что нам и требуется.

Address	Symbol	Value
200	w	B'10101101'
03	status	B'00011000'
07	PORTC	D'2 '
26	K_ED	D'5 '
25	K_DES	D'3 '
21	K_SOT	D'2 '
51	Kod_seg_ed	B'01101101'
52	Kod_seg_des	B'01001111'
53	Kod_seg_sot	B'01011011'
40	CHISLO	B'11101011'
41	VYCH	D'10 '
02	PCL	D'2 '
25	K_DES	B'00000011'
42	R_D_ED	D'53 '
43	Rab_Reg	D'5 '

Рисунок А.2 – Окно наблюдения

Фрагменты программы.

Стандартные инструкции настройки МК и обнуление регистров составляются самостоятельно.

Регистры K_SOT, K_DES, K_ED предназначены для хранения соответственно количества сотен, десятков и единиц.

Регистры Kod_seg_ed, Kod_seg_des, Kod_seg_sot предназначены для хранения кодов сотен, десятков и единиц для сегментов цифрового индикатора.

```

MOVLW b'11101011' ; исходное двоичное число.
MOVWF CHISLO      ; вспомогательный регистр.
MOVWF Rab_Reg     ; подсчет количества сотен.
MOVLW D'100'
MOVWF VYCH        ; вычитаемое при определении количества сотен.
M_SOT INCF K_SOT, F ;
SUBWF Rab_Reg, F  ; Rab_Reg= Rab_Reg-100
BTFSC STATUS, C   ; проверка, разность отрицательна (C=0)?
GOTO M_SOT        ; цикл подсчета количества сотен повторяется
при C=1.
ADDWF Rab_Reg, F  ; был заем, выполняется возврат назад на один
шаг, добавляя 100.
DECF K_SOT, F     ; восстанавливается истинное количество сотен.
MOVLW D'10'
MOVWF VYCH        ; вычитаемое при определении количества
десятков.
Подсчёт количества десятков выполняется аналогично подсчету количества
сотен.

```

Определяется код для индикации количество единиц

MOVFW Rab_Reg ; рабочий регистр содержит единицы после
 возврата.
 MOVWF K_ED ; переписывается количество единиц в регистр.
 SWAPF K_DES, W ; перемещаются десятки в старшие разряды.
 ADDWF K_ED, W ; добавляется количество единиц в младшие
 разряды.
 MOVWF R_D_ED ; этот регистр теперь содержит десятки и
 единицы.
 MOVF K_ED, W ; W- аргумент для подпрограммы Vyb_Seg.
 CALL VYBOR_SEG ; вызов подпрограммы с аргументом W.
 MOVWF Kod_seg_ed ; запись кода для индикации единиц.
Аналогично определяются коды для индикации десятков и сотен
 GOTO \$; разделитель главной программы и
 подпрограмм.
 VYBOR_SEG ; подпрограмма, возвращающая код для
 включения сегментов.
 addwf PCL, f ; получаем новый адрес в PLC=PCL + W.
 retlw b'00111111' ; код для цифры 0. Возвращается в главную
 программу при W=0.
 retlw b'00000110' ; код для цифры 1. Возвращается в главную
 программу при W=1.
 retlw b'01011011' ; код для цифры 2. Возвращается в главную
 программу при W=2.
 retlw b'01001111' ; код для цифры 3. Возвращается в главную
 программу при W=3.
 retlw b'01100110' ; код для цифры 4. Возвращается в главную
 программу при W=4.
 retlw b'01101101' ; код для цифры 5. Возвращается в главную
 программу при W=5.
 retlw b'01111101' ; код для цифры 6. Возвращается в главную
 программу при W=6.
 retlw b'00000111' ; код для цифры 7. Возвращается в главную
 программу при W=7.
 retlw b'01111111'; код для цифры 8. Возвращается в главную
 программу при W=8.
 retlw b'01101111' ; код для цифры 9. Возвращается в главную
 программу при W= 9.
 ; конец подпрограммы.
 END ; конец программы.

Список литературы

- 1 Рудакова Л.Н. Элементы и устройства автоматики. Методические указания по выполнению лабораторных работ для студентов специальности 5В070200 - Автоматизация и управление. – Алматы: АУЭС, 2017.
- 2 Яценков В.С. Микроконтроллеры MicroCHIP. Практическое руководство. – М.: Горячая линия – Телеком, 2008.
- 3 Кохц Д. Измерение, управление и регулирование с помощью PIC-микроконтроллеров: Схемы и программы для микроконтроллеров PIC16C71, PIC16F84 и семейства PIC16C5X. МК-Пресс, 2015.
- 4 Тавернье К. PIC-контроллеры. Практика применения: Пер. с фр. М.: ДМК-Пресс, 2010.
- 5 Катцен С. PIC микроконтроллеры: полное руководство.- М.: «Додека», 2010.
- 6 Копесбаева А.А., Тарасов В. М. Программирование цифровой техники и микроконтроллеров управления. Методические указания к выполнению лабораторных работ для студентов специальности 5В070200 -«Автоматизация и управление» - Алматы: АУЭС, 2013.
- 7 Промышленные приборы и средства автоматизации. Справочник/В.Я. Боронов, Т.Х. Безновская, В.А.Бек и др.; Под общ. ред. В.В.Черенкова. – Л.: Машиностроение, 1987.
- 8 <http://www.microchip.su/>

Содержание

Введение.....	3
1 Порядок выполнения курсовой работы	4
2 Перечень заданий к курсовой работе	19
Приложение А. Применение сегментного индикатора	24
Список литературы	27