**РЕФЕРАТ**

Цель работы: произвести разработку микропроцессорной системы АОНа на базе микроконтроллера.

Содержание работы: в работе выполнено построение структурной схемы, построение функциональной схемы, выбор элементной базы, оптимальной для реализации поставленных задач по диапазону характеристик, разработана программа, разработана принципиальная схема устройства, соответствующая стандартам эксплуатации телефонных сетей Украины.

**Содержание**

Введение

1. Описание объекта и функциональная спецификация

2. Описание структуры системы

3. Описание ресурсов МК

3.1 Характеристики микроконтроллера Z86E0812PSC

3.2 ЖК индикатор PANAPHONE

4. Ассемблирование и разработка алгоритма работы устройства

5. Программирование микроконтроллера

6. Управление АОН и описание функциональных узлов МПС

7. Технические характеристики АОН

8. Описание выбора элементной базы и работы принципиальной схемы

Заключение

Список использованных источников

Приложение А

Приложение Б

**ВВЕДЕНИЕ**

Современную микроэлектронику трудно представить без такой важной составляющей, как микроконтроллеры. Микроконтроллеры незаметно завоевали весь мир. Микроконтроллерные технологии очень эффективны. Одно и то же устройство, которое раньше собиралось на традиционных элементах, будучи собрано с применением микроконтроллеров, становится проще, не требует регулировки и меньше по размерам. С применением микроконтроллеров появляются практически безграничные возможности по добавлению новых потребительских функций и возможностей к уже существующим устройствам. Для этого достаточно просто изменить программу.

Однокристальные (однокорпусные) микроконтроллеры представляют собой приборы, конструктивно выполненные в виде БИС и включающие в себя следующие составные части: микропроцессор, память программ и память данных, а также программируемые интерфейсные схемы для связи с внешней средой.

Мировая промышленность выпускает огромную номенклатуру микроконтроллеров. По области применения их можно разделить на два класса: специализированные, предназначенные для применения в какой-либо одной конкретной области (контроллер для телевизора, контроллер для модема, контроллер для компьютерной мышки ) и универсальные, которые не имеют конкретной специализации и могут применяться в самых различных областях микроэлектроники, с помощью которых можно создать как любое из перечисленных выше устройств, так и принципиально новое устройство.

В данном курсовом проекте производится разработка микропроцессорной системы АОНа на базе микроконтроллера.

1. **Описание объекта и функциональная спецификация**

АОН – Автоматический Определитель Номера.

АОН определяет номер звонящего абонента, благодаря устройствам, смонтированным на телефонных станциях. Прямое назначение АОН, определение номера звонящего абонента, для включения тарификатора.

АОН, который мы будем рассматривать, предназначен для встраивания в телефоны производства юго-восточных стран и известных под названиями «Panasoanic», «Panaphone» и др. и которые имеют в наличии ЖК индикатор.

Функциональная спецификация

1. Входы
   1. 15 кнопок
   2. Телефонная линия
   3. Дополнительное питание от адаптера или другого источника
2. Выходы
   1. Жидкокристаллический индикатор
   2. 2 светодиода («Удержание», «Громкая связь»)
   3. Звуковой динамик излучатель
3. Функции

а. Определение номера звонящего абонента после поднятия трубки на собственном аппарате или на параллельном телефоне

b. Занесение определившегося номера в список с емкостью до 9-и номеров

c. Определение номера набираемого на параллельном телефоне пульсовым методом

1. Просмотр номеров из списка с помощью двух кнопок CLOCK и SET.
2. Выбор представления выводимых из списка номеров.
3. Сохранение возможности настройки времени у часов с помощью кнопок CLOCK и SET и сохранение всех функций телефона в который встраивается АОН.

**2. Описание структуры системы**

После определения входов и выходов устройства разработана структурная схема устройства.

Структурная схема телефона.

**3. Описание ресурсов МК**

### 3.1 Характеристики микроконтроллера Z86E0812PSC

Микроконтроллер Z86E0812PSC примененный в данной разработке имеет следующие характеристики:

* разрядность 8бит;
* корпус DIP18;
* рабочее напряжение от 3.0 до 5.5 В;
* потребляемая мощность (типовая) 50мВт;
* время выполнения инструкции типа регистр-регистр 1мкс;
* два режима пониженного энергопотребления STOP и HALT;
* 14 линий ввода/вывода;
* однократно электрически программируемое ПЗУ ёмкостью 2 Кбайта;
* 124 байтарегистрового ОЗУ;
* два 8-и битных программируемых таймера/счетчика с 6-битными программируемыми делителями;
* 6-и векторная система прерываний с программируемой 6-уровневой приоритетностью;
* внешняя тактовая частота 12 МГц;
* система Watch-Dog и сброс по включению питания без внешних цепей;
* два встроеных компаратора;
* температура эксплуатации 0о C-70 оC

Расположение выводов и общий вид микроконтроллера имеет вид изображенный на рис. 1.

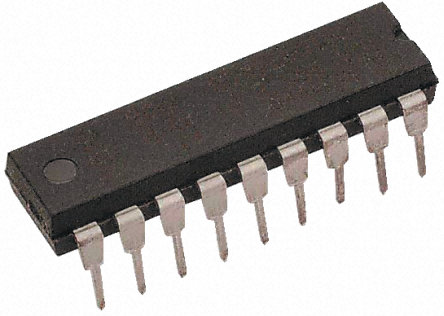


Рис. 1 - Общий вид микроконтроллера Z86E0812PSC

Вспомогательным элементом АОН служит ЖК индикатор PANAPHONE.

**3.2 ЖК индикатор PANAPHONE**

Назначение выводов ЖК индикатора:

* 12/24H - лог.0 поддерживает 24-х часовой счет времени лог.1 поддерживает 12-и часовой счет времени
* KEYCLOCK - подключение клавиши CLOCK (лог.0-выкл., лог.1-вкл.) вход в режим коррекции времени
* KEYSET - подключение клавиши SET (лог.0 - выкл., лог.1 -вкл.) перебор значений часов и минут в режиме коррекции времени
* RESET - подключение клавиши RESET (лог.0-выкл., лог.1-вкл.) первое включение переводит индикатор в режим измерения временного интервала с начальным обнулением счетчика второе нажатие останавливает счет и через 10 сек индикатор переходит в режим счета времени.
* SEL –сигнал выбора контроллера (лог.0) индикатор доступен для ввода информации, ввод производится сдвигом информации на одно знакоместо влево с появлением в крайнем правом знакоместе нового символа.
* DATA - линия передачи данных.

CLK - линия стробирования данных, минимальный период следования сигнала - 4 мкс.

**4. Ассемблирование и разработка алгоритма работы устройства**

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

В pезультате pаботы ассемблеpа создаются файлы со следующими pасшиpениями:\* HEX - объектный файл\* LST - файл листинга\* ERR - файл ошибок и пpедупpеждений\* COD Объектный файл создается в 16-pичном фоpмате и содеpжит код, котоpый должен быть записан в микpосхему. Файл листинга содеpжит полный листинг пpогpаммы вместе с загpузочным кодом. В файл ошибок и пpедупpеждений записываются все ошибки и пpедупpеждения, возникающие в пpоцессе ассемблиpования. Они также пpисутствуют и в файле листинга.После обpаботки нашей пpогpаммы ассемблеp должен был выдать сообщение "Assembly Successful", означающее, что ошибок обнаpужено не было. Файл ошибок не должен был создаться.

Листинг программы и объектный файл приведен в Приложении А.

**5. Программирование микроконтроллера**

После ассемблирования имеется объектный файл EXAMPLE.HEX, котоpый должен быть записан в микpосхему. Запись осуществляется пpи помощи пpогpамматоpа и пpогpаммы Pic-prog. Микросхему микроконтроллера вставляется в панель программатора. Программатор подключается к порту LPT1. Необходимо запустить программу Pic\_prog.exe.

Подать питание на программатор. Выполнить команду «ЗАПИСАТЬ / ПАМЯТЬ ПРОГРАММ». В течении следующих нескольких секунд будит выполняться процесс программирования, а затем проверка правильности записанных в микроконтроллер данных.

### 6. Управление АОН и описание функциональных узлов МПС

Используя клавиши CLOCK и SET пользователь может просмотреть список номеров звонивших абонентов а также установить удобный для себя образ представления номеров на индикаторе. Просмотр списка возможен когда не производятся операция вода времени и набора номера. По нажатию клавиши SET на индикаторе появляется номер из первой ячейки списка, нажимая клавишу CLOCK пользователь может просматривать следующие номера из списка, после девятого номера происходит возврат к первому номеру из списка. Номера в список записываются в порядке поступления, последний поступивший номер хранится в первой ячейке списка. Нажатие клавиши SET в режиме просмотра списка приводит к выходу из этого режима и восстановлению режима счета времени. В режиме счета времени может производится счет текущего времени или счет времени разговора, что означает занятость или незанятость телефонной линии соответственно. Счет времени разговора начинается с нуля после каждого выхода из режима просмотра списка. С целью экономии энерго потребления в режиме просмотра списка и режиме установок после отсутствия нажатия клавиш CLOCK и SET более 10 с индикатор переходит в режим счета времени разговора, следующее нажатие клавиши обновляет информацию на индикаторе. В случаях когда пользователь поднимает трубку, нажимает клавишу «SP-PHONE», поднимают трубку на параллельном телефоне или идет звонок, производится автоматический выход из режимов просмотра списка, установок, ввода времени и на индикатор выводится счет времени разговора. При освобождении телефонной линии абонентом на индикаторе возобновляется счет текущего времени. В связи с присутствием специфической информации в номерах абонентов принимаемых АОН пользователь может исключить ее индикацию при просмотре списка. Исключатся может индикация категории абонента и седьмой цифры номера АТС звонящего абонента. В случае междугородного звонка 7-я цифра будет индицироваться если она не совпадает с маской в веденной в установках. В режим установки входят из режима просмотра списка последовательно нажимая CLOCK и не отпуская SET. На индикаторе должны появится две цифры первая из которых выделена кавычками. Изменение значения цифры производится клавишей CLOCK переход к следующей цифре клавишей SET, третье нажатие SET приводит к выходу в режим просмотра списка. Первая цифра в установках принимает значения 0 и 1, что означает: 0 - показ категории абонента, 1 - запрет показа категории абонента, категория абонента выводится в 8-ом знакоместе справа в режиме просмотра списка. Вторая цифра в установках является маской 7-ой цифры номера абонента она принимает значения 0, 1,..., 9, пробел. В случае совпадения 7-ой цифры с маской первая не индицируется. В случае когда маска имеет значение «пробел» 7-я цифра номера будет всегда выводится на индикацию. Для быстрого выхода из любого режима пользователь может поднять и опустить трубку. Следует отметить что, когда выход из режима просмотра списка производится при поднятой трубке на своем или параллельном аппарате возможна задержка обновления информации на индикаторе на 10 с.

### 7. Технические характеристики АОН

Таблица 1 - Технические характеристики АОН

|  |  |
| --- | --- |
| напряжение питания схемы | 3.5-6.5 В |
| потребляемый схемой ток в активном режиме | < 1.1 мА |
| ток потребляемый схемой от телефонной линии | < 300 мкА |
| длительность вызывного сигнала АОН | 0.2 сек |
| максимальное количество повторов запроса сигнала АОН АТС | 4 |
| необходимый уровень напряжения звонка (пиковое значение) | 100 В |
| необходимый уровень сигнала ответа АТС при отношении сигнал/шум 12дБ | > 30мВ |
| необходимое напряжение линии при положенной трубке | 25-65В |

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

Схема работает по принципу активизации процессора в момент внешнего воздействия и пребывания его в режиме пониженного энергопотребления (режим STOP) в остальных случаях. Для вывода процессора из режима STOP служит сигнал логического 0 на выводе P2.7. Если на этом выводе сохраняется логический 0, то процессор не переходит в режим STOP. Активизация процессора вызывается следующими ситуациями: нажатие клавиши CLOCK или SET, падение напряжения в линии ниже уровня 20 В. Таким образом процессор активизируется после поступления звонка, при снятии трубки или просто при отключении от линии, вследствии чего при питании от батарей телефон не следует оставлять надолго отключенным от линии из-за достаточно большого тока потребления в активном режиме. Датчиком напряжения линии служит цепь собранная на элементах VD1, VD2, VD3, R1, R2, VD4, VD5, VD24. На элементах R21, VD21, VD22, VD23 собрано логическое.

И по активизации процессора от разных событий. Цепь C1, R5 служит для передачи аналогового сигнала на вход компаратора процессора AN1. На элементах VD6, VD7, R3, R4, VD8, C4, R7 собран датчик звонка формирующий напряжение на входе второго компаратора AN2. Элементы C5, R8, R9 формируют опорное напряжение для компараторов. Запросный сигнал синусоидальной формы формируется схемой на элементах R17, C8, C9. Для формирования запросного сигнала использован принцип дельта-модуляции с частотой дискретизации около 44 КГц. Исходная цифровая последовательность проходя интегрирующую цепь R17, C8 приобретает синусоидальную форму. Для передачи сигнала в линию используется схема на элементах R18, VT3, R19, R20, VT4, VD19, VD20. Для подстройки амплитуды сигнала изменяется сопротивление резистора R18. Режим по постоянному току задается резистором R20. Включение запросной цепи производится ключом DD3. Замыкание перемычкой контактов B2 и B3 приводит к выключению функции определения набора номера с параллельного телефона, что может понадобиться на блокированных линиях. Замыкание перемычкой контактов B2 и B1 включает функцию определения набора номера с параллельного телефона. Оставлять контакт B2 неподключеным нельзя. Цепи на VT1 и VT2 предназначены для перехвата сигналов нажатия клавиш SET и CLOCK. Через диоды VD15, VD16, VD17 процессор получает доступ к управлению ЖКИ параллельно с управляющей микросхемой телефона. Через резистор R11 и диод VD14 схема получает питание с разговорной части телефона. Через резистор R10 и диод VD12 схема получает питание от батареи которой снабжен телефон.

Подключение контактов. K1 - Подключение к плюсу диодного моста на плате телефона, выпрямляющего напряжение линии. K2 - Подключение +4,5 В от батареи питания телефона. K3 - свободный.

K4 - подключается к сигналу DATA ЖКИ.

K5 - подключается к сигналу CLK ЖКИ. K6 - подключается к сигналу SEL ЖКИ. K7 - подключается к сигналу KEYCLOCK ЖКИ. K8 - подключается к сигналу KEYSET ЖКИ. K9 - подключается после ключа включающего разговорную часть, но до сглаживающего дросселя. K10 и K11 подключаются непосредственно к двум проводам телефонной линии. K12 - подключается к общему проводу телефона.

Схема принципиальная электрическая АОН разработанная в *Accel EDA* представлена на рис. 2.

Номера записанные в память телефона для быстрого набора никак не связаны с номерами хранящимися в списке АОН и манипуляции с одними не отражаются на других.

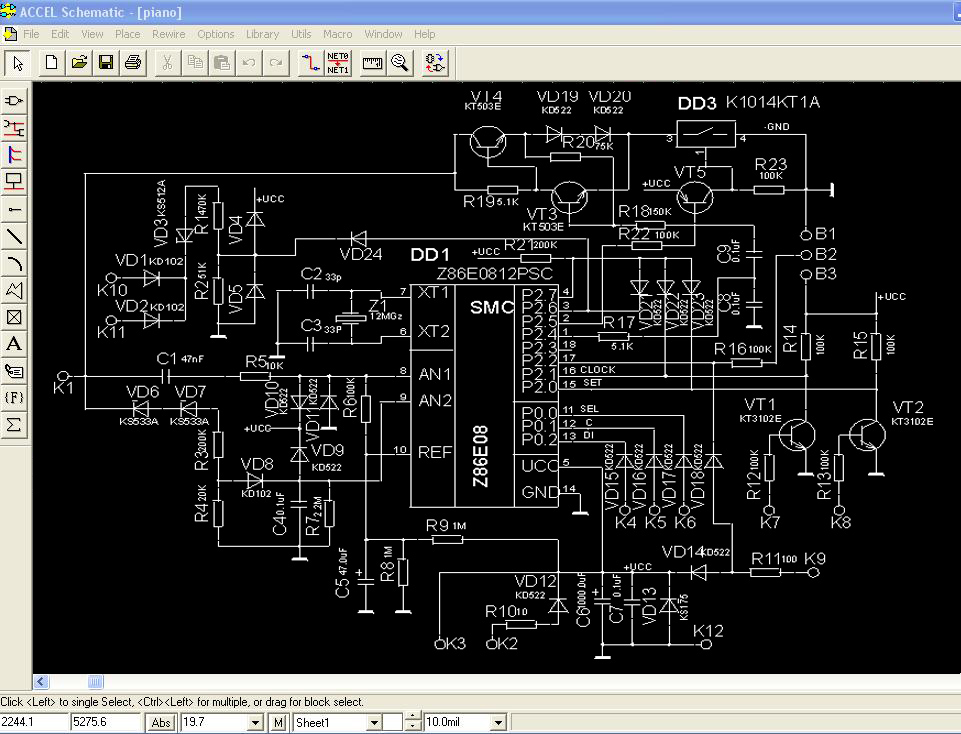


Рис. 2 - Принципиальная схема АОНа в Accel EDA

Принципиальная схема АОНа приведена в Приложении Б.

Спецификация элементов приведена в Приложении В.

**ЗАКЛЮЧЕНИЕ**

В данном курсовом проекте разработано устройство - Автоматический Определитель Номера. Разработана схема электрическая принципиальная этого устройства и программа для микроконтроллера. В результате ассемблирования 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. Существенно сократились размеры устройства, относительно аналогичных, собранных на базе п/п элементов и соответственно сократилось энергопотребление.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Техническая документация на микроконтроллеры Z86E0812PSC компании Microchip Technology Incorporated ООО«Микро-Чип», Москва, 2002.-184 с.

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. Ульрих В.А. Микроконтроллеры PIC16х\17хx / В.А. Ульрих. Изд. 2-е, перераб. и доп. – СПб.: Наука и техника, 2002. – 320 с.

5. Александр Елисеев (E-mail: ea@eunet.lt)

**ПРИЛОЖЕНИЕ А**

**Листинг программы и объектный файл**

TITLE Программа АОН "Panaphone"

Z8REGDEFS; Назначение символических имен

TIM\_BL\_H = 10H; Таймер звонка

TIM\_BL\_L = 11H; Таймер звонка

N\_RECORD = 12H; Номер записи в списке

BLC\_CLOCK = 13H; Счетчик блокировки по нажатию клавиши CLOCK

CNT\_SET = 14H; Счетчик нажатия SET

CNT\_CLOCK = 15H; Счетчик нажатия CLOCK

CNT\_TWIX = 16H; Счетчик нажатия обоих клавиш

F\_KEYS = 17H; XXXXXXXX Флаги клавиш

CONST\_KEY = 10; Константа минимального количества тактов между нажатиями

CONST\_BELL = 1000; Константа звонка

TET1 = 18H; 1-я ячейка тетрады

TET2 = 19H; 2-я ячейка тетрады

TET3 = 1AH; 3-я ячейка тетрады

TET4 = 1BH; 4-я ячейка тетрады

TET1\_ = R8; 1-я ячейка тетрады

TET2\_ = R9; 2-я ячейка тетрады

TET3\_ = R10; 3-я ячейка тетрады

TET4\_ = R11; 4-я ячейка тетрады

CNT\_ONL = 18H; Счетчик низкой линии

PRE\_N = 1CH; Предыдущая цифра

CNT\_REP = 1DH; Счетчик повторов цифры

COND = 1EH; Состояние линии

MASK1 = 1FH; Маска первой цифры номера

BUF\_NUM = 20H; Начало буфера номера

BEG\_ST = 40H; Начало стека

BEG\_COD = 40H; Начало области записи принятого кода

BEG\_LST = 59H; Начало списка

END\_LST = 7CH; Конец списка

BFLAG = 7FH; XXXXXXXX

; ¦¦¦¦¦¦¦¦

; ¦¦¦¦¦¦¦+------

; ¦¦¦¦¦¦+-------

; ¦¦¦¦¦+-------- Флаг неопределившегося номера

; ¦¦¦¦+--------- флаг блокировки звонка

; ¦¦¦+---------- флаг индикации категории

; ¦¦+----------- Флаг активности своего телефона

; ¦+------------ Флаг 1-й цифры

; +------------- Флаг звонка

TESTB = 7EH; Тест-байт целостности информации

CNT\_B = 7DH; Счетчик длительности сигнала звонка

;+------------------------------------------------------------------+

;¦ Начало программного блока ¦

;+------------------------------------------------------------------+

CODE

GLOBAL

ABS

ORG 0

Область хранения векторов прерываний

WORD INT\_P3\_2L;P3.2 F-edge

WORD INT\_P3\_3L;P3.3 F-edge

WORD INT\_P3\_1L;P3.1 F-edge

WORD INT\_P3\_2H;P3.2 R-edge

WORD INT\_T0;T0

WORD Sample;T1

; После сброса: Счетчики запрещены

; Прерывания запрещены

; Приоритеты не определены

; Указатель стека не определён

; Регистровый указатель = 0

; Порт 2 открытый сток,все входы

; Порт 3 цифровой

; Порт 0 вход

INT\_P3\_2L:

INT\_P3\_3L:

INT\_P3\_1L:

INT\_P3\_2H:

START:

; LD RP,#0FH; Активизация управления по P2.7

; LD 0BH,#00010100B; для эмулятора

LD RP,#00

LD P01M,#00000100B; Порт P0 - выход

LD P3M,#00000011B; P2 - Pull-Ups Activ

; P3 - Analog

LD P2, #11111111B; Вывод в порт 2

; ¦¦¦¦¦¦¦¦

; ¦¦¦¦¦¦¦+------ Вход клавиши SET

; ¦¦¦¦¦¦+------- Вход клавиши CLOCK

; ¦¦¦¦¦+-------- -

; ¦¦¦¦+--------- -

; ¦¦¦+---------- Датчик активности телефона

; ¦¦+----------- Управление ключом

; ¦+------------ Датчик линии

; +------------- Вход прерываний

;

LD P2M,#11001111B; Определение направления выводов порта P2

LD P0,#00000111B;

LD PRE0,#11001001B; Коэффицент деления 50

LD TMR,#00001011B; Разрешение обоих таймеров T0 с загрузкой

LD T0,#150; Переполнение таймера каждые 5 мс

LD IMR,#10010000B; Разрешение T0 и запрещение T1

BEGIN:

LD SPL,#BEG\_ST

EI

CP TESTB,#55H; Проверка целостности информации

JR Z,NOT\_CLEAR\_RAM;

LD R4,#5

CLEAR\_RAM:; Очистка ОЗУ контроллера

LD @R4,#0

INC R4

CP R4,#127

JR C,CLEAR\_RAM

LD TESTB,#55H

LD MASK1,#2; Выставить маску первой цифры номера

LD CNT\_B,#0

NOT\_CLEAR\_RAM:

AND BFLAG,#00000111B; Сброс флага звонка,определения

; пульсового набора,набранной цыфры и

; очистки буфера

CALL GO\_HALT

;

;------------------------------------------------------------------------

; Основной блок

;------------------------------------------------------------------------

MAIN:

TCM P2,#00000011B

JP NZ,M1

M2\_2:

TM P2,#01000000B

JR Z,M2

TM P3,#00000100B

JR NZ,M2

LD P2M,#11001111B

NOP

STOP

M2:

LD R4,#10

LD CNT\_ONL,#0

M2\_0: CALL GO\_HALT

INC CNT\_ONL; Счетчик низкой линии

TM P2,#01000000B

JR Z,M2\_1

LD CNT\_ONL,#0

M2\_1:

DJNZ R4,M2\_0

CP CNT\_ONL,#8

JR UGE,M3; Переход если трубку сняли

TM BFLAG,#10000000B

JR NZ,M2

JR M2\_2

M3:

AND BFLAG,#10110011B

TM BFLAG,#10000000B

JR Z,M4

AND P0,#11111110B; Захват LCD

PUSH IMR

CALL ID\_NUM

OR P2,#00100000B

POP IMR

LD R7,#BUF\_NUM; Вывод номера на LCD

LD R6,#0

LD R8,#8

CALL SEND\_BYTE

CALL SEND\_BYTE

M3\_1: LD R6,@R7

CALL SEND\_BYTE

INC R7

DJNZ R8,M3\_1

TM BFLAG,#00000100B; Не сохранять номер если он не определился

JR NZ,M4

LD R7,#END\_LST; Сдвиг списка на одну запись

LD R8,#END\_LST-4

M3\_2: LD R4,@R8

LD @R7,R4

DEC R7

DEC R8

CP R8,#BEG\_LST

JR UGE,M3\_2

LD R7,#BEG\_LST; Упаковка номера в список

LD R8,#BUF\_NUM

LD R9,#4

M3\_3: LD R4,@R8

SWAP R4

AND R4,#0F0H

INC R8

AND @R8,#0FH

OR R4,@R8

LD @R7,R4

INC R8

INC R7

DJNZ R9,M3\_3

OR P0,#00000001B; Снять захват LCD

M4: AND BFLAG,#00110011B

LD BLC\_CLOCK,#0

TM P2,#00000100B

JR NZ,M4\_1

AND P0,#11111110B; Захват LCD

M4\_1:

CALL GO\_HALT

TM P2,#01000000B; Проверка тока в линии

JR Z,M5; Переход если нет импульса в линии

LD R9,#0; Счетчик цыфры

MN4\_1: LD R4,#20; Счетчик импульса

MN4\_2: CALL GO\_HALT

TM P2,#01000000B; Проверка тока в линии

JR Z,MN4\_3; Переход если прервался импульс

DJNZ R4,MN4\_2

; Выход если импульс слишком длинный

OR P0,#00000001B; Снять захват LCD

LD P2M,#11001111B

NOP

STOP

MN4\_3:

LD R4,#12; Счетчик паузы

INC R9

MN4\_4: CALL GO\_HALT

TM P2,#01000000B; Проверка тока в линии

JR NZ,MN4\_1; Переход если прервалась пауза

DJNZ R4,MN4\_4

TM P2,#00000100B

JR NZ,M5

MN4\_6: TM BFLAG,#01000000B; Проверить флаг первой цифры

JR NZ,MN4\_7

LD R6,#00

LD R8,#10

MN4\_8:CALL SEND\_BYTE

DJNZ R8,MN4\_8

OR BFLAG,#01000000B; Выставить флаг первой цифры

MN4\_7: LD R6,R9

CALL SEND\_BYTE

M5:

TM F\_KEYS,#00001000B; Проверка клавиши SET

JR Z,M4\_1

AND F\_KEYS,#11110111B;

CALL LIST

JR M4

M1:

TM F\_KEYS,#00010000B; Проверка клавиши CLOCK

JR Z,M1\_1

AND F\_KEYS,#11101111B; Сброс флага клавиши CLOCK

CP BLC\_CLOCK,#00; Пуст ли счетчик нажатий клавиши CLOCK

JR NZ,M1\_0

LD BLC\_CLOCK,#2

JR M1\_1

M1\_0: DEC BLC\_CLOCK

M1\_1: TM F\_KEYS,#00001000B; Проверка клавиши SET

JP Z,MAIN

;

; Режим просмотра списка

;

AND F\_KEYS,#11110111B

CALL LIST

JP MAIN

;+----------------------------------------------------------------------+

;¦ Режим просмотра списка ¦

;+----------------------------------------------------------------------+

LIST:

AND F\_KEYS,#11110111B

CP BLC\_CLOCK,#00

JP Z,LIST1

RET

LIST1: AND P0,#11111110B; Захват LCD

LD R4,#10

LIST2: CALL GO\_HALT

DJNZ R4,LIST2

LIST3: LD N\_RECORD,#1

LIST4: LD R6,N\_RECORD

CALL SEND\_BYTE; Вывод на LCD номера записи

LD R6,#0FH; '-'

CALL SEND\_BYTE

LD R7,N\_RECORD; Вычисление адреса записи

DEC R7

RL R7; Умножить на 4 (длину записи)

RL R7

ADD R7,#BEG\_LST; Получить адрес начала записи

LD R8,#2

;

; Вывод номера

;

LIST5:

LD R6,@R7; Вывод 8-и байтов номера

SWAP R6

AND R6,#0FH;

TM BFLAG,#00010000B; Проверить флаг индикации категории

JP NZ,LIST6

LD R6,#0; Не показывать категорию если не выставлен

LIST6:; флаг

CALL SEND\_BYTE

LD R6,@R7; Вывод первой цифры номера

AND R6,#0FH

CP R6,MASK1

JP NZ,LIST7

LD R6,#0

LIST7:

CALL SEND\_BYTE

INC R7; Вывод второй цифры номера

LD R6,@R7

SWAP R6

CALL SEND\_BYTE

SWAP R6; Вывод третьей цифры номера

CALL SEND\_BYTE

LIST8:

INC R7; Вывод остальных цифр номера

LD R6,@R7

SWAP R6

CALL SEND\_BYTE

SWAP R6

CALL SEND\_BYTE

DJNZ R8,LIST8

LIST9: TM F\_KEYS,#00100000B; Проверка одновременного нажатия CLOCK,SET

JR Z,LIST10

AND F\_KEYS,#11000111B;

JR LIST12

LIST10: TM F\_KEYS,#00010000B; Проверка клавиши CLOCK

JR Z,LIST11

AND F\_KEYS,#11101111B;

INC N\_RECORD

CP N\_RECORD,#9

JR UGT,LIST3

JR LIST4

LIST11: TM F\_KEYS,#00001000B; Проверка клавиши SET

JP Z,LIST23

AND F\_KEYS,#11110111B;

OR P0,#00000001B

CALL GO\_HALT

CALL SEND\_BYTE

RET

;

; Режим установок

;

LIST12:

; Установка флага показа категории

LD R6,#0CH

CALL SEND\_BYTE

TM BFLAG,#00010000B; Проверить флаг индикации категории

LD R6,#0AH

JP Z,LIST13; Показать флаг индикации категории

LD R6,#01H

LIST13:

CALL SEND\_BYTE

LD R6,#0DH

CALL SEND\_BYTE

LD R6,#0

CALL SEND\_BYTE

CALL SEND\_BYTE

LD R6,MASK1

CALL SEND\_BYTE

LD R6,#0

CALL SEND\_BYTE

CALL SEND\_BYTE

CALL SEND\_BYTE

CALL SEND\_BYTE

LIST14:

TM F\_KEYS,#00001000B; Проверка клавиши SET

JR Z,LIST15

AND F\_KEYS,#11110111B;

JP LIST17

LIST15:

TM F\_KEYS,#00010000B; Проверка клавиши CLOCK

JR Z,LIST16

AND F\_KEYS,#11101111B;

XOR BFLAG,#00010000B;

JP LIST12

LIST16: CALL GO\_HALT

JP NZ,BEGIN

JR LIST14

LIST17:; Установка маски первой цифры

LD R6,#00H

CALL SEND\_BYTE

TM BFLAG,#00010000B; Проверить флаг индикации категории

LD R6,#0AH

JP Z,LIST18; Показать флаг индикации категории

LD R6,#01H

LIST18:

CALL SEND\_BYTE

LD R6,#00H

CALL SEND\_BYTE

CALL SEND\_BYTE

LD R6,#0CH

CALL SEND\_BYTE

LD R6,MASK1

CALL SEND\_BYTE

LD R6,#0DH

CALL SEND\_BYTE

LD R6,#00H

CALL SEND\_BYTE

CALL SEND\_BYTE

CALL SEND\_BYTE

LIST19:

TM F\_KEYS,#00001000B; Проверка клавиши SET

JR Z,LIST20

AND F\_KEYS,#11110111B;

JP LIST4

LIST20:

TM F\_KEYS,#00010000B; Проверка клавиши CLOCK

JR Z,LIST22

AND F\_KEYS,#11101111B;

INC MASK1

CP MASK1,#0AH

JR ULE,LIST21

LD MASK1,#00

LIST21:

JP LIST17

LIST22: CALL GO\_HALT

JP NZ,BEGIN

JR LIST19

;

;

;

LIST23: CALL GO\_HALT

JP NZ,BEGIN

JP LIST9

;

;----------------------------------------------------------------------

; END LIST

;+--------------------------------------------+

;¦ Подпрограмма HALT режима ¦

;+--------------------------------------------+

GO\_HALT:

PUSH R4

NOP

HALT

EXIT\_STOP:

;

; Поддержка клавиши SET

;

TM P2,#00000001B;

JR NZ,K1;

TM F\_KEYS,#00000001B;

JR NZ,K3;

OR F\_KEYS,#00001001B;

JR K3;

K1: AND F\_KEYS,#11110110B;

;

; Поддержка клавиши CLOCK

;

K3:;

TM P2,#00000010B;

JR NZ,K4;

TM F\_KEYS,#00000010B;

JR NZ,K5;

OR F\_KEYS,#00010010B;

JR K5;

K4: AND F\_KEYS,#11101101B;

;

; Поддержка клавиш CLOCK+SET

;

K5:;

TCM F\_KEYS,#00000011B;

JR NZ,K6;

TM F\_KEYS,#00000100B;

JR NZ,K7;

OR F\_KEYS,#00100100B;

JR K7;

K6: AND F\_KEYS,#11011011B;

K7:;

;

; Отслеживание звонка

;

TM BFLAG,#00001000B; Проверить блокировку звонка

JR NZ,K12

TM P3,#00000100B

JR Z,K10

INC CNT\_B

CP CNT\_B,#60; 0.3 сек для звонка

JP ULT,K12

OR BFLAG,#10000000B

LD TIM\_BL\_H,#^hb(CONST\_BELL)

LD TIM\_BL\_L,#^lb(CONST\_BELL)

JR K12

K10:

CLR CNT\_B

LD R4,TIM\_BL\_H

OR R4,TIM\_BL\_L

JR NZ,K11

AND BFLAG,#01111111B

JR K12

K11: DECW TIM\_BL\_H

K12:

LD R4,COND; Сравнить предыдущее и настоящее состояния

LD COND,P2; линии

XOR R4,COND; Z - если состояния одинаковы

AND R4,#01000000B; NZ - если состояния разные

POP R4

RET

;+----------------------------------------------------------------------+

;¦ Определение номера звонящего абонента ¦

;+----------------------------------------------------------------------¦

;¦ Выход: флаг C = 0 - определение произведено ¦

;¦ C = 1 - сбой определения ¦

;¦ ¦

;+----------------------------------------------------------------------+

ID\_NUM:

LD TET1,0FFH; Заполнение тетрады

LD TET2,0FFH; кодом неопределенности

LD TET3,0FFH

LD TET4,0FFH

CALL INIT\_T1\_500

LD IMR,#10100000B; Разрешение T1 и запрещение T0

LD R5,#4; Число запросов 3

AND BFLAG,#01111111B; Сброс звонка

LD TIM\_BL\_H,#0; Сбросить таймер звонка

LD TIM\_BL\_L,#0; Сбросить таймер звонка

AND P2,#11011111B; Занять линию

;---------------------------------------------------------------------------

; Запрос 500 Гц

;

REQ0:

CALL INIT\_T1\_44117

EI

LD IMR,#10100000B; Разрешение T1 и запрещение T0

LOOP1:

LD R9,#0; 6

LD R11,#BEG\_COD

LOOP1\_0:

LD R12,#^hb(BIT\_MAP); 6

LD R13,#^lb(BIT\_MAP); 6

ADD R13,R9; 6

ADC R12,#0; 10

LDC R10,@RR12; 12

LD @R11,R10

INC R11

INC R9; 6

CP R9,#11; 10

JP ULT,LOOP1\_0; 12/10 MAX = 131

LD R10,#100

LOOP4:

LD R11,#BEG\_COD; 6

LOOP4\_0:

LD R8,#8; 6

LOOP3:

LD R12,@R11; 10

AND R12,#11011111B

LD R2,R12

RL @R11; 10

NOP; 6

HALT; 7

LOOP2: DJNZ R8,LOOP3; 12/10

INC R11; 6

CP R11,#BEG\_COD+11; 10

JP ULT,LOOP4\_0; 12/10 MAX = 131

; JP LOOP4

DJNZ R10,LOOP4; 12

;----------------------------------------------------------------------------

TM P2,#00000100B; Снять захват линии если активен свой телефон

JR Z,ID\_NUM0;

OR P2,#00100000B;

ID\_NUM0:

LD R4,#20

ID\_NUM1:

PUSH R4; Ожидать ответа 0.2 сек

PUSH R5

CALL AON\_processor

POP R5

POP R4

CP R8,#30

JP NC,ID\_NUM2

DJNZ R4,ID\_NUM1

DJNZ R5,REQ0

ID\_NUM2:

LD CNT\_REP,#0; Обнулить счетчик повторов

LD PRE\_N,#0FFH; Предыдущая цифра не определена

LD R4,#25; Количество принимаемых кодов

LD R5,#BEG\_COD; Адрес буфера принимаемых цифр

LD TET1,R9; Запись в тетраду первого кода

ID\_NUM3:

PUSH R4

PUSH R5

CALL AON\_processor

POP R5

POP R4

;

; Вписать цифру в тетраду

;

LD RP,#10H; Сменить банк регистров

LD TET4\_,TET3\_; Сместить цыфры в тетраде

LD TET3\_,TET2\_;

LD TET2\_,TET1\_;

LD TET1\_,9; Записать текущую цифру

;

; Обработка тетрады

;

LD 9,TET1\_

CP 9,TET2\_

JP Z,ID\_NUM4

LD 9,TET2\_

CP 9,TET3\_

JP Z,ID\_NUM4

LD 9,TET3\_

CP 9,TET4\_

JP Z,ID\_NUM4

LD 9,#0FFH

ID\_NUM4:

LD RP,#00

CP PRE\_N,R9; Цифра равна предыдущей ?

JP Z,ID\_NUM6

CP CNT\_REP,#00; Записывать цифру только после второго

JP Z,ID\_NUM6; повтора

ID\_NUM5:

LD @R5,R9

LD PRE\_N,R9; Сохранить текущую цифру

INC R5

LD CNT\_REP,#00; Обнулить счетчик повторов

DJNZ R4,ID\_NUM3; Продолжать ввод пока не будет записано

JR ID\_NUM8\_; 25 цифр

ID\_NUM6:

INC CNT\_REP;

; CP R9,#0FFH; Не повторять запись цифр отличных от 0FFH

; JR NZ,ID\_NUM3;

CP CNT\_REP,#7;

JP UGT,ID\_NUM5;

JR ID\_NUM3

;

; Сборка номера

;

ID\_NUM8\_:

LDR4,#BEG\_COD

LD R5,#6; Поиск двух кодов 0AH в области принятых

ID\_NUM9:; кодов

CP @R4,#0AH; Адрес первого хранится в рег. R6

JR NZ,ID\_NUM7; второго в рег. R7

LD @R5,R4;

INC R5;

CP R5,#8;

JR UGE,ID\_NUM8;

ID\_NUM7:;

INC R4;

CP R4,#BEG\_COD+25

JR ULT,ID\_NUM9

ID\_NUM8:

CP R5,#8

JR Z,ID\_NUM10

CP R5,#7

JR Z,ID\_NUM11

;

; Заполнение буфера кодом неопределенности если не найдено ни одного

; кода 0AH

;

LDR4,#BUF\_NUM

LDR5,#8

ID\_NUM12:

LD @R4,#0FH

INC R4

DJNZ R5,ID\_NUM12

JP ID\_NUM\_ERR

ID\_NUM10:

LDR8,R6

LD R9,R7

JR ID\_NUM13

ID\_NUM11:

LD R8,R6

LD R9,R8

; ADD R9,#9

ID\_NUM13:

LDR4,#8

LD R10,#BUF\_NUM

DEC R8

DEC R9

ID\_NUM20:

CP R8,#BEG\_COD

JR ULT,ID\_NUM14

CP R8,#BEG\_COD+24

JR UGE,ID\_NUM14

LDR6,@R8

JR ID\_NUM15

ID\_NUM14:

LD R6,#0FFH

ID\_NUM15:

CP R9,#BEG\_COD

JR ULT,ID\_NUM16

CP R9,#BEG\_COD+24

JR UGE,ID\_NUM16

LDR7,@R9

JR ID\_NUM17

ID\_NUM16:

LD R7,#0FFH

ID\_NUM17:

CP R7,#0FFH

JR Z,ID\_NUM18

LD @R10,R7

JR ID\_NUM19

ID\_NUM18:

LD @R10,R6

ID\_NUM19:

DEC R8

DEC R9

INC R10

DJNZ R4,ID\_NUM20

;

; Удаление повторов и нулей

LD R4,#BUF\_NUM

LD R5,@R4

LD R6,#8

ID\_NUM22:

CP @R4,#0BH

JR NZ,ID\_NUM21

LD @R4,R5

ID\_NUM21:

CP @R4,#0

JR NZ,ID\_NUM23

LD @R4,#10

ID\_NUM23:

LDR5,@R4

INC R4

DJNZ R6,ID\_NUM22

; Циклический сдвиг на одну позицию вправо

LD R4,#BUF\_NUM+7

LD R5,#BUF\_NUM+6

LD R8,@R4

LD R7,#7

ID\_NUM24:

LD R6,@R5

LD @R4,R6

DEC R4

DEC R5

DJNZ R7,ID\_NUM24

LD @R4,R8

RCF

RET

ID\_NUM\_ERR:

OR BFLAG,#00000100B; Флаг неопределения

RET

;+-------------------------------------------------+

;¦ Подпрограмма инициализации T1 на 500 ¦

;+-------------------------------------------------+

INIT\_T1\_500:

LD PRE1,#11110011B; Коэффицент деления 60

LD TMR,#00001110B; Разрешение обоих таймеров T1 с загрузкой

LD T1,#50; Переполнение таймера каждые 2 мс

RET

;+-------------------------------------------------+

;¦ Подпрограмма инициализации T1 на 44117 Гц ¦

;+-------------------------------------------------+

INIT\_T1\_44117:

LD PRE1,#01000111B; Коэффицент деления 17

LD TMR,#00001110B; Разрешение обоих таймеров T1 с загрузкой

LD T1,#2; Переполнение таймера каждую

RET

;+-------------------------------------------------+

;¦ Подпрограмма вывода байта на ЖКИ 'Panasoanic' ¦

;¦ Ввод: R6 - выводимый байт ¦

;¦ ¦

;¦ Изменяются R4,R5 ¦

;+-------------------------------------------------+

SEND\_BYTE:

SWAP R6

LD R5,#4

SB1: RL R6

JP C,SB2; Передача 1

AND P0,#11111011B; Передача 0

JR SB3

SB2: OR P0,#00000100B

SB3:

LD R4,#15

DJNZ R4,$

AND P0,#11111101B; Передача строба

LD R4,#15

DJNZ R4,$

OR P0,#00000010B

LD R4,#15

DJNZ R4,$

OR P0,#00000100B; Снятие сигнала

LD R4,#15

DJNZ R4,$

DJNZ R5,SB1

RET

;+-------------------------------------------------------------------------+

;¦ Подпрограмма ввода информации АОН ¦

;+-------------------------------------------------------------------------+

Fr1\_1 = 20H

Fr1\_2 = 21H

Fr2\_1 = 22H

Fr2\_2 = 23H

Fr3\_1 = 24H

Fr3\_2 = 25H

Fr4\_1 = 26H

Fr4\_2 = 27H

Fr5\_1 = 28H

Fr5\_2 = 29H

Fr6\_1 = 2AH

Fr6\_2 = 2BH

AON\_processor:

LD PRE1,#00010111B; Коэфф. дел. =5

LD TMR,#00001110B

LD T1,#15; Коэфф. дел. =15

; При частоте 12 МГц период счетчика 50 мкс

LD IMR,#10100000B; Разрешение T1 запрещение T0

; Производится 200 отсчетов

; На каждую эпюру отводится в таблице 25 байт

LD R4,#Fr1\_1

LD R6,#0

LD R7,#6

LD R8,#8

LD R14,#^hb(TABLE); @RR14 - Адрес таблицы эпюр

LD R15,#^lb(TABLE);

LD R12,#^hb(NUM\_TABLE); @RR12 - Адрес таблицы сумм

; Формирование первого байта отсчетов

AONp2: NOP

HALT;7 Ожидание времени отсчета

SCF;6

TM P3,#00000010B;10

JR NZ,AONp0;10

CCF;6

AONp0: RLC R6;6 Формирование байта отсчетов

CP R7,#0;10

JR Z,AONp1;10

LD @R4,#0;10

INC R4;6

LD @R4,#0;10

INC R4;6

DEC R7;6

AONp1:

DJNZ R8,AONp2;12 -> 103 T. 17.17 мкс

; Прием и обработка следующих байтов отсчетов

LD R11,#25;10

AONp5:

LD R4,#Fr1\_1;10 Адрес буфера идентификаторов

LD R7,#6;10 Счетчик принимаемых бит

LD R8,#8;10 Счетчик циклов ввода

LD R9,R6;6 Сохранить предыдущий байт отсчетов

LD R6,#0;10 Байт отсчетов

AONp4: NOP

HALT;7

SCF;6

TM P3,#00000010B;10

JR NZ,AONp6;10

CCF;6

ONp6: RLC R6;6 Формирование байта отсчетов

CP R7,#00;10

JR Z,AONp3;10

LDC R13,@RR14;12

XOR R13,R9;6

LDC R10,@RR12;12

ADD R10,@R4;6

LD @R4,R10;10

INCW RR14;10

INC R4;6

LDC R13,@RR14;12

XOR R13,R9;6

LDC R10,@RR12;12

ADD R10,@R4;6

LD @R4,R10;10

INCW RR14;10

INC R4;6

DEC R7;6

AONp3:

DJNZ R8,AONp4;12 ->207 Продолжение цикла ввода байта

; отсчетов

DJNZ R11,AONp5;12 ->275 Продолжение ввода байтов отсчетов

; Приведение массива идентификаторов к нормальной форме

LD R13,#01

LD R14,#00

LD R15,#00

LD R4,#Fr1\_1;10 Адрес буфера идентификаторов

LD R6,#6;10 Счетчик нормальных идентификаторов

LD R7,#00

AONp11:

SUB @R4,#101

JR NC,AONp7

COM @R4

AONp7: LD R5,@R4

INC R4

SUB @R4,#101

JR NC,AONp8

COM @R4

AONp8: ADD R5,@R4; Получение нормального идентификатора

INC R4

; Определение пары максимальных идентификаторов

CP R8,R5

JR NC,AONp9

LD R7,R8

LD R8,R5

LD R14,R15

LD R15,R13

JR AONp10

AONp9: CP R7,R5

JR NC,AONp10

LD R7,R5

LD R14,R13

AONp10: RL R13

DJNZ R6,AONp11

; Вычисление цифры

OR R14,R15

LD R9,#12

AONp13:

LD R12,#^hb(TABLE\_NUM\_AON-1)

LD R13,#^lb(TABLE\_NUM\_AON-1)

ADD R13,R9

ADC R12,#0

LDC R10,@RR12

CP R14,R10

JR Z,AONp12

DJNZ R9,AONp13

AONp12: DEC R9

DI

RET

;Выход \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; R8 - величина первого идентификатора

; R7 - величина второго идентификатора

; R9 - принятая цифра {0..Bh}, если R9=FFh то цифра не определена

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;

; Последовательность цифр в посылке:

; 1. Категория абонента

; 2. Единицы номера

; 3. Десятки номера

; 4. Сотни номера

; 5. Тысячи номера

; 6. 3-я цифра индекса станции

; 7. 2-я цифра индекса станции

; 8. 1-я цифра индекса станции

; 9. Начало/конец передачи

; 10. Категория абонента

;----------------------------

; Подпрограмма ввода выборки

;----------------------------

Sample:

IRET

INT\_T0:

IRET

;----------------------------

TABLE\_NUM\_AON:;усл.номер цифра

.BYTE 18H; 0

.BYTE 03H; 1

.BYTE 05H; 2

.BYTE 06H; 3

.BYTE 09H; 4

.BYTE 0AH; 5

.BYTE 0CH; 6

.BYTE 11H; 7

.BYTE 12H; 8

.BYTE 14H; 9

.BYTE 24H; 0AH - код начала/конца

.BYTE 28H; 0BH - код повтора

BIT\_MAP:

.BYTE 11111111B

.BYTE 01111111B

.BYTE 01111101B

.BYTE 00101010B

.BYTE 00001000B

.BYTE 00000000B

.BYTE 00000000B

.BYTE 00000010B

.BYTE 01110100B

.BYTE 11110110B

.BYTE 11111110B

.ORG 05D4H

;----------------------------

; Таблица АОН

TABLE:

.BYTE 0FFH,0FFH,0FFH,0FCH,0FFH,0F8H,0FFH,0F0H,0FEH,0F0H,0FCH,0E0H

.BYTE 0FEH,000H,0F0H,000H,0C0H,003H,000H,00FH,003H,01FH,00FH,07EH

.BYTE 000H,003H,001H,07FH,01FH,0FEH,0FFH,0F0H,0F8H,080H,0C0H,007H

.BYTE 007H,0FFH,0FFH,0F0H,0F0H,000H,001H,01FH,01FH,0FEH,0FCH,0E0H

.BYTE 0FFH,0F0H,0C0H,001H,007H,0FFH,0FEH,0E0H,0C0H,007H,00FH,07EH

.BYTE 0E0H,000H,007H,0FFH,0FCH,080H,001H,01FH,07EH,0F0H,0C0H,007H

.BYTE 000H,01FH,0FFH,0E0H,001H,01FH,0FCH,0E0H,003H,01FH,0F8H,0E0H

.BYTE 03FH,0FFH,000H,003H,0FFH,0F0H,003H,03FH,0F8H,080H,01FH,0FCH

.BYTE 0FFH,080H,01FH,0FFH,000H,007H,0FCH,0C0H,01FH,0FEH,081H,00FH

.BYTE 000H,001H,0FCH,080H,07FH,0FCH,007H,03FH,0C0H,007H,0F8H,0C0H

.BYTE 003H,0FFH,000H,00FH,0C0H,001H,0F8H,080H,07EH,0F0H,01FH,0FCH

.BYTE 0FFH,0F8H,07FH,0FEH,01FH,0FFH,007H,07FH,003H,01FH,081H,00FH

.BYTE 0F8H,000H,0F8H,000H,0F8H,000H,0F8H,080H,0F8H,080H,0F8H,0C1H

.BYTE 000H,00FH,000H,03FH,003H,07FH,00FH,0FFH,01FH,0FEH,03FH,0F8H

.BYTE 01FH,0FFH,0FFH,0F8H,0FEH,0C0H,0F0H,000H,0C0H,007H,003H,01FH

.BYTE 0FFH,0C0H,0E0H,000H,000H,01FH,00FH,0FEH,07EH,0F0H,0F0H,081H

.BYTE 080H,000H,003H,0FFH,0FFH,0F0H,0E0H,001H,003H,01FH,03FH,0F8H

.BYTE 001H,0FFH,0FFH,0E0H,080H,007H,01FH,0FEH,0F8H,080H,003H,01FH

.BYTE 0FFH,0FEH,080H,001H,03FH,0FEH,0E0H,001H,01FH,0FEH,0F0H,081H

.BYTE 0FCH,000H,00FH,0FFH,0E0H,000H,03FH,0FCH,0C0H,007H,03EH,0F0H

.BYTE 000H,007H,0FEH,0C0H,00FH,0FFH,0C0H,003H,07EH,0F0H,007H,03FH

.BYTE 00FH,0FFH,000H,007H,0F8H,080H,03FH,0FCH,003H,01FH,0E0H,003H

.BYTE 0FFH,0E0H,03FH,0FFH,003H,03FH,0C0H,007H,0F8H,080H,07EH,0F0H

.BYTE 0C0H,000H,0F8H,000H,0FFH,0E0H,07FH,0F8H,01FH,0FEH,007H,03FH

.BYTE 000H,07FH,000H,01FH,000H,00FH,080H,007H,0C0H,007H,0E0H,003H

.ORG 700H

;

; ТАБЛИЦА СООТВЕТСТВИЯ БАЙТА И КОЛИЧЕСТВА

; ЕДИНИЦ В НЕМ

NUM\_TABLE:

.BYTE 00H,01H,01H,02H,01H,02H,02H,03H,01H,02H,02H,03H,02H,03H,03H,04H

.BYTE 01H,02H,02H,03H,02H,03H,03H,04H,02H,03H,03H,04H,03H,04H,04H,05H

.BYTE 01H,02H,02H,03H,02H,03H,03H,04H,02H,03H,03H,04H,03H,04H,04H,05H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 01H,02H,02H,03H,02H,03H,03H,04H,02H,03H,03H,04H,03H,04H,04H,05H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 03H,04H,04H,05H,04H,05H,05H,06H,04H,05H,05H,06H,05H,06H,06H,07H

.BYTE 01H,02H,02H,03H,02H,03H,03H,04H,02H,03H,03H,04H,03H,04H,04H,05H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 03H,04H,04H,05H,04H,05H,05H,06H,04H,05H,05H,06H,05H,06H,06H,07H

.BYTE 02H,03H,03H,04H,03H,04H,04H,05H,03H,04H,04H,05H,04H,05H,05H,06H

.BYTE 03H,04H,04H,05H,04H,05H,05H,06H,04H,05H,05H,06H,05H,06H,06H,07H

.BYTE 03H,04H,04H,05H,04H,05H,05H,06H,04H,05H,05H,06H,05H,06H,06H,07H

.BYTE 04H,05H,05H,06H,05H,06H,06H,07H,05H,06H,06H,07H,06H,07H,07H,08H

.END

**Приложение Б**

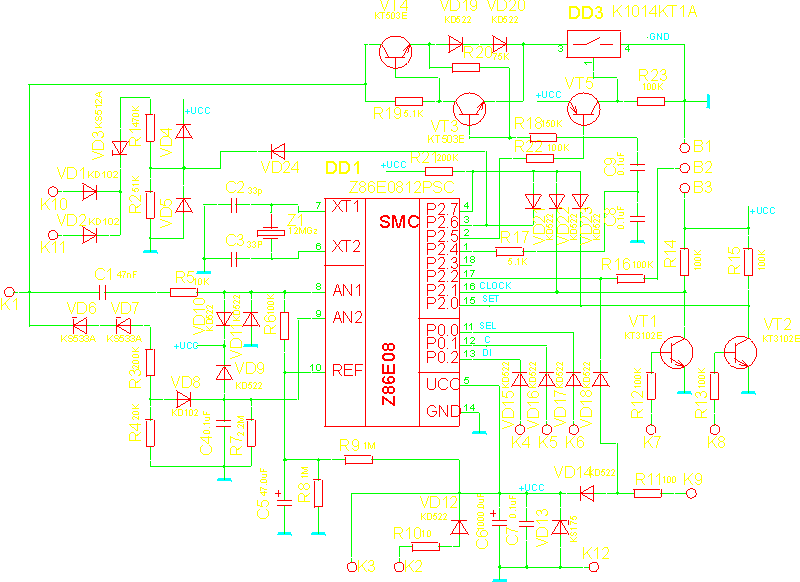


Схема принципиальная электрическая АОН