#include <mega16.h>

#include <delay.h>

#asm

 .equ \_\_lcd\_port=0x18 //LCD дисплей подключен к порту B

#endasm

#include <lcd.h>

#include <stdio.h> //заголовочный файл функций ввода вывода

#asm

 .equ \_\_i2c\_port=0x15 //шина I2C подключена к порту С микроконтроллера

 .equ \_\_sda\_bit=1

 .equ \_\_scl\_bit=0

#endasm

#include <i2c.h> // функции шины I2C

#include <math.h> //заголовочный файл использования математических функций

#include <float.h>

eeprom unsigned char nomer=0;

eeprom unsigned char ad[201]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

eeprom unsigned char opred[201]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

eeprom float kr[8]={0,0,0,0,0,0,0,0};

unsigned char koor[10];

unsigned char g;

unsigned char ykaz;

unsigned char address\_1;

unsigned char address\_2;

unsigned char r;

unsigned char l;

unsigned char nomer\_1;

unsigned char chn;

unsigned char buk[20];

unsigned char kod;

unsigned char n;

unsigned char ld;

unsigned char g;

unsigned int j;

unsigned char x;

unsigned char x1;

unsigned char x2;

unsigned char z;

unsigned char flag;

unsigned char data;

unsigned char i;

unsigned int gps[100];

unsigned char y;

unsigned char a;

unsigned char has;

unsigned char mu;

unsigned char sek;

unsigned char gr\_sh;

unsigned char min\_sh;

unsigned int sek\_sh;

unsigned char gr\_dol;

unsigned char min\_dol;

unsigned int sek\_dol;

unsigned char nahalo;

unsigned int h;

unsigned char sputnik;

unsigned char sputnik\_r;

unsigned char bl;

unsigned char blokir;

unsigned long int rasst;

unsigned int yg2;

unsigned char oper;

unsigned char oper1;

unsigned char oper2;

unsigned char oper3;

unsigned char oper4;

unsigned char oper5;

unsigned char oper6;

unsigned char oper7;

unsigned char d;

unsigned char ik[8];

unsigned char ind;

unsigned char het;

unsigned int indik;

unsigned long int rez;

flash char string[]= {0x30,0x00}; //0

flash char string\_1[]= {0x31,0x00}; //1

flash char string\_2[]= {0x32,0x00}; //2

flash char string\_43[]= {0x2E,0x00}; //.

flash char string\_4[]= {0x34,0x00}; //4

flash char string\_5[]= {0x35,0x00}; //5

flash char string\_7[]= {0x37,0x00}; //7

flash char string\_8[]= {0x38,0x00}; //8

flash char string\_3[]= {0x33,0x00}; //3

flash char string\_6[]= {0x36,0x00}; //6

flash char string\_9[]= {0x39,0x00}; //9

flash char string\_10[]= {0xB8,0x00}; //и

flash char string\_11[]= {0xA6,0x00}; //й

flash char string\_12[]= {0xBA,0x00}; //к

flash char string\_13[]= {0xBB,0x00}; //л

flash char string\_14[]= {0xE4,0x00}; //ф

flash char string\_15[]= {0x78,0x00}; //х

flash char string\_16[]= {0xE5,0x00}; //ц

flash char string\_17[]= {0xC0,0x00}; //ч

flash char string\_18[]= {0x61,0x00}; //а

flash char string\_19[]= {0xB2,0x00}; //б

flash char string\_20[]= {0xB3,0x00}; //в

flash char string\_21[]= {0xB4,0x00}; //г

flash char string\_22[]= {0xBC,0x00}; //м

flash char string\_23[]= {0xBD,0x00}; //н

flash char string\_24[]= {0x6F,0x00}; //о

flash char string\_25[]= {0xBE,0x00}; //п

flash char string\_26[]= {0xC1,0x00}; //ш

flash char string\_27[]= {0xE6,0x00}; //щ

flash char string\_28[]= {0xC2,0x00}; //ъ

flash char string\_29[]= {0xC3,0x00}; //ы

flash char string\_30[]= {0x80,0x00}; //пусто

flash char string\_31[]= {0xE3,0x00}; //д

flash char string\_32[]= {0x65,0x00}; //е

flash char string\_33[]= {0xB6,0x00}; //ж

flash char string\_34[]= {0xB7,0x00}; //з

flash char string\_35[]= {0x70,0x00}; //р

flash char string\_36[]= {0x63,0x00}; //с

flash char string\_37[]= {0xBF,0x00}; //т

flash char string\_38[]= {0x79,0x00}; //у

flash char string\_39[]= {0xC4,0x00}; //ь

flash char string\_40[]= {0xC5,0x00}; //э

flash char string\_41[]= {0xC6,0x00}; //ю

flash char string\_42[]= {0xC7,0x00}; //я

flash char string\_44[]= {0x54,0x6F,0xC0,0xBA,0x61,0x00}; //точка

flash char string\_45[]= {0x54,0x6F,0xC0,0xBA,0x61,0x80,0xB7,0x61,0xBE,0xB8,0x63,0x61,0xBD,0x61,0x00}; //точка записана

flash char string\_46[]= {0xA8,0x61,0xBC,0xC7,0xBF,0xC4,0x80,0xB7,0x61,0xBD,0xC7,0xBF,0x61,0x00}; //память занята

flash char string\_47[]= {0x48,0x65,0xBF,0x80,0xE3,0x61,0xBD,0xBD,0xC3,0x78,0x00}; //нет данных

flash char string\_48[]= {0x43,0xBF,0x70,0x3F,0x00}; //стр?

flash char string\_49[]= {0x42,0x63,0xB5,0x3F,0x00}; //все?

flash char string\_50[]= {0x80,0x80,0x80,0x80,0x00}; //пустая строка

flash char string\_51[]= {0xA8,0x61,0xBC,0xC7,0xBF,0xC4,0x80,0xBE,0x79,0x63,0xBF,0x61,0x00}; //память пуста

char string\_52[]={0x48,0x65,0xBF,0x80,0x63,0xB8,0xB4,0xBD,0x61,0xBB,0x61,0x00};//нет сигнала

char string\_53[]={0xE7,0x00};//мин

char string\_54[]={0x22,0x00};//сек

char string\_55[]={0x43,0xAC,0x00}; //сш

char string\_56[]={0x42,0xE0,0x00}; //вд

char string\_57[]={0x48,0x61,0xB3,0x65,0xE3,0x65,0xBD,0xB8,0x65,0x00}; //наведение

char string\_58[]= {0x54,0x65,0xBA,0x79,0xE6,0x61,0xC7,0x80,0xBF,0x6F,0xC0,0xBA,0x61,0x3F,0x00}; //текущая точка?

char string\_59[]= {0xAC,0xB8,0x70,0x6F,0xBF,0x61,0x3F,0x00}; //широта?

char string\_60[]= {0xE0,0x6F,0xBB,0xB4,0x6F,0xBF,0x61,0x3F,0x00}; //долгота?

char string\_61[]= {0x70,0x61,0xB7,0x70,0xC7,0xB6,0x65,0xBD,0x00}; //разражен

char string\_62[]= {0x41,0xBA,0xBA,0x79,0xBC,0x79,0xBB,0xC7,0xBF,0x6F,0x70,0x00}; //аккумулятор

char lcd\_buffer[10];

char lcd\_buffer\_1[10];

char lcd\_buffer\_2[10];

char lcd\_buffer\_3[10];

char lcd\_buffer\_4[10];

char lcd\_buffer\_5[10];

char lcd\_buffer\_6[10];

char lcd\_buffer\_7[10];

char lcd\_buffer\_8[10];

char lcd\_buffer\_9[10];

char lcd\_buffer\_10[10];

char lcd\_buffer\_11[10];

char lcd\_buffer\_12[10];

char lcd\_buffer\_13[10];

float f1;

float b;

float f2;

float lim1;

float lim2;

float cf1;

float cf2;

float sf1;

float sf2;

float dlim;

float cd;

float sd;

float ygol;

float yg;

interrupt [TIM1\_OVF]void timeR1(void){kod=0;TCCR1B=0;} // функция прерывания по переполнению таймера

interrupt [USART\_RXC] void usart\_rx\_isr(void){if((UCSRA&=0x18)==0){data=UDR;};if(flag==0){

 if(data==36){nahalo=1;};

if(nahalo==1){gps[i]=data; if(data!=13){++i;}else{if((gps[3]==82)&&(gps[4]==77)&&(gps[5]==67))

{a=gps[18];};

if((gps[3]==71)&&(gps[4]==71)&&(gps[5]==65)){ y=1;};

if((gps[3]==71)&&(gps[4]==83)&&(gps[5]==86)){sputnik=(gps[11]-48)\*10+(gps[12]-48);}

;i=0;nahalo=0;};};};} // функция прерывания по завершению приема

 interrupt [ADC\_INT] void adc\_isr (void) {rez+=ADCW;++het;if(het==100){indik=rez/100;ADCSRA=0xAE;

rez=0;het=0; ind=1;};} // функция прерывания по завершению преобразования АЦП

void main(void){

PORTA=0xFF;

DDRA=0x00;

PORTB=0xFF;

DDRB=0x00;

PORTC=0xFF;

DDRC=0x00;

PORTD=0xFF;

DDRD=0x38;

ACSR=0x80;// аналог. компаратор откл.

TIMSK=0x04;

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

UCSRA=0x00;

UCSRB=0x90;

UCSRC=0x86;

UBRRH=0x00;

UBRRL=0x03; // скорость обмена USART 115200

ADCSRA =0xAE;

ADMUX=0x00;

i2c\_init();

lcd\_init(16);

#asm("sei")

flag=1;

ind=0;

ADCSRA=0xEE; // запуск АЦП

while(ind==0);

if(indik<=328){lcd\_gotoxy(2,1);lcd\_puts(string\_62);

lcd\_gotoxy(2,2);lcd\_puts(string\_61);

delay\_ms(2000);lcd\_clear();};

if(PIND.2==0){blokir=1;};

m:while(1){flag=0; y=0;

while(y==0);

switch (a){

case 86:lcd\_clear();lcd\_gotoxy(3,0);lcd\_puts(string\_52);

break;

case 65:

has=(gps[7]-48)\*10+ (gps[8]-48);

mu=(gps[9]-48)\*10+(gps[10]-48);

sek=(gps[11]-48)\*10+ (gps[12]-48);

sprintf(lcd\_buffer,"%u:%u:%u",has,mu,sek);

gr\_sh=(gps[18]-48)\*10+ (gps[19]-48);

min\_sh=(gps[20]-48)\*10+(gps[21]-48);

sek\_sh=(gps[23]-48)\*600+(gps[24]-48)\*60+(gps[25]-48)\*6+(gps[26]-48)\*0.6;

 sprintf(lcd\_buffer\_1,"%u\xdf",gr\_sh);

 sprintf(lcd\_buffer\_2,"%u",min\_sh);

 sprintf(lcd\_buffer\_3,"%u.%02u",sek\_sh/100,sek\_sh%100);

gr\_dol=(gps[31]-48)\*10+ (gps[32]-48);

min\_dol=(gps[33]-48)\*10+(gps[34]-48);

sek\_dol=(gps[36]-48)\*600+(gps[37]-48)\*60+(gps[38]-48)\*6+(gps[39]-48)\*0.6;

 sprintf(lcd\_buffer\_4,"%u\xdf",gr\_dol);

 sprintf(lcd\_buffer\_5,"%u",min\_dol);

 sprintf(lcd\_buffer\_6,"%u.%02u",sek\_dol/100,sek\_dol%100);

 sputnik\_r=gps[45]-48;

 sprintf(lcd\_buffer\_7,"%u",sputnik\_r);

 h=(gps[52]-48)\*1000+(gps[53]-48)\*100+(gps[54]-48)\*10+(gps[56]-48);

 sprintf(lcd\_buffer\_8,"%u.%uM",h/10,h%10);

 sprintf(lcd\_buffer\_9,"%u",sputnik);

 lcd\_clear();

 lcd\_gotoxy(0,0);

 lcd\_puts(lcd\_buffer\_1);

 lcd\_gotoxy(3,0);

lcd\_puts(lcd\_buffer\_2);

lcd\_gotoxy(5,0);

lcd\_puts(string\_53);

lcd\_gotoxy(6,0);

lcd\_puts(lcd\_buffer\_3);

lcd\_gotoxy(11,0);

lcd\_puts(string\_54);

lcd\_gotoxy(12,0);

lcd\_puts("N");

lcd\_gotoxy(0,1);

 lcd\_puts(lcd\_buffer\_4);

 lcd\_gotoxy(3,1);

lcd\_puts(lcd\_buffer\_5);

lcd\_gotoxy(5,1);

lcd\_puts(string\_53);

lcd\_gotoxy(6,1);

lcd\_puts(lcd\_buffer\_6);

lcd\_gotoxy(11,1);

lcd\_puts(string\_54);

lcd\_gotoxy(12,1);

lcd\_puts("S");

lcd\_gotoxy(0,2);

lcd\_puts(lcd\_buffer);

lcd\_gotoxy(9,2);

lcd\_puts(lcd\_buffer\_8);

lcd\_gotoxy(14,0);

lcd\_puts(lcd\_buffer\_7);

lcd\_gotoxy(14,1);

lcd\_puts(lcd\_buffer\_9);

if((PIND.2==0)&&(blokir==1)){f1=((float)sek\_sh/360000+(float)min\_sh/60+(float)gr\_sh)\*0.01745; // если нажата кнопка

lim1=((float)sek\_dol/360000+(float)min\_dol/60+(float)gr\_dol)\*0.01745; //наведение.

lim2=((kr[6]\*100+kr[7])/360000+kr[5]/60+kr[4])\*0.01745;

dlim=lim2-lim1;

cf1=cos(f1); sf1=sin(f1);

cf2=cos(f2); sf2=sin(f2);

sd=sin(dlim)\*10000; cd=cos(dlim)\*10000;

ygol=atan(sqrt(pow((cf2\*sd),2)+pow((cf1\*sf2\*10000-sf1\*cf2\*cd),2))/(sf1\*sf2\*10000+cf1\*cf2\*cd));

rasst=6372795\*ygol;

b=(cf1\*sf2\*10000)-(sf1\*cf2\*cd);

yg=atan(-(sd\*cf2)/b)\*57.296;

if(b<0){yg=yg+180;};

yg=-(fmod((yg+180),360)-180);

if(yg<0){yg=360+yg;};

yg2=yg\*100;

sprintf(lcd\_buffer\_11,"%uM",rasst);

sprintf(lcd\_buffer\_12,"%u\xdf%u",yg2/100,(yg2%100\*6)/10);

sprintf(lcd\_buffer\_13,"%u",((yg2%100\*6)%10)\*6);

lcd\_gotoxy(0,3);

lcd\_puts(lcd\_buffer\_12);

lcd\_gotoxy(5,3);

lcd\_puts(string\_53);

lcd\_gotoxy(6,3);

lcd\_puts(lcd\_buffer\_13);

lcd\_gotoxy(8,3);

lcd\_puts(string\_54);

lcd\_gotoxy(9,3);

lcd\_puts(lcd\_buffer\_11);}else{blokir=0;};};

if(blokir==0){if(PIND.2==0){delay\_ms(50); if(PIND.2==0){flag=1;y=0;i=0;nahalo=0; bl=1;goto m\_3;};};}; //вход в наведение.

if(PINC.6==0){delay\_ms(50); if(PINC.6==0){flag=1;y=0;i=0;nahalo=0; //вход в режим режим чтение

++nomer;

if(nomer>200){lcd\_clear(); lcd\_gotoxy(2,0);lcd\_putsf(string\_46);delay\_ms(2000);lcd\_clear();goto m;};

sprintf(lcd\_buffer\_10,"%u",nomer);

lcd\_clear(); lcd\_gotoxy(0,0);lcd\_putsf(string\_44);//точка

lcd\_gotoxy(5,0);lcd\_puts(lcd\_buffer\_10,); lcd\_gotoxy(8,0);lcd\_puts(":");

 while(ld==0){ //набор символов из цифр и русских букв не больше 16

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xDF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(kod!=1){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00;

 buk[chn]=1;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_1);}; delay\_ms(300); kod=1; ++chn; TCCR1B=0x04;}else{

--chn; buk[chn]=43; if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_43);}; delay\_ms(300);++chn; kod=0; };};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(kod!=4){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=4;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_4);}; delay\_ms(300); kod=4; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=10;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_10);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=11;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_11);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=12;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_12);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=13;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_13);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(kod!=7){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=7;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_7);}; delay\_ms(300); kod=7; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=14;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_14);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=15;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_15);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=16;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_16);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=17;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_17);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.5==0){delay\_ms(50);

 if(PINC.5==0){ld=1;};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xEF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(kod!=2){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=2;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_2);}; delay\_ms(300); kod=2; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=18;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_18);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=19;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_19);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=20;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_20);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=21;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_21);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(kod!=5){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=5;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_5);}; delay\_ms(300); kod=5; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=22;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_22);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=23;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_23);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=24;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_24);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=25;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_25);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(kod!=8){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=8;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_8);}; delay\_ms(300); kod=8; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=26;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_26);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=27;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_27);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=28;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_28);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=29;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_29);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){if(kod!=100){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00;

 buk[chn]=100;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string);}; delay\_ms(300); kod=100; ++chn; TCCR1B=0x04;}else{

--chn; buk[chn]=30; if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_30);}; delay\_ms(300);++chn; kod=0; };};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xF7;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(kod!=3){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=3;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_3);}; delay\_ms(300); kod=3; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=31;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_31);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=32;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_32);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=33;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_33);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=34;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_34);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(kod!=6){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=6;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_6);}; delay\_ms(300); kod=6; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=35;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_35);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=36;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_36);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=37;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_37);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=38;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_38);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(kod!=9){TCCR1B=0x00; TCNT1H=0x00;TCNT1L=0x00; n=0;

 buk[chn]=9;

if(chn<=15){

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_9);}; delay\_ms(300); kod=9; n=1; ++chn; TCCR1B=0x04;}else{

switch(n){case 1: --chn; buk[chn]=39;

if(chn<=15){ lcd\_gotoxy(chn,1);

lcd\_putsf(string\_39);};delay\_ms(300);++chn; ++n;

break;

case 2: --chn; buk[chn]=40;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_40);};delay\_ms(300);++chn; ++n;

break;

case 3: --chn; buk[chn]=41;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_41);};delay\_ms(300);++chn; ++n;

break;

case 4: --chn; buk[chn]=42;

if(chn<=15){lcd\_gotoxy(chn,1);

lcd\_putsf(string\_42);};delay\_ms(300);++chn; n=0; kod=0;};};};};

if(PINC.5==0){delay\_ms(50);

 if(PINC.5==0){TCCR1B=0x00;TCNT1H=0x00;TCNT1L=0x00; --chn; buk[chn]=0;

lcd\_gotoxy(chn,1);

lcd\_putsf(string\_30);delay\_ms(300);kod=0;n=0;};};};

ykaz=1;

while(ykaz!=0){ ykaz=ad[g];if(ykaz!=0){++g;};}; //поиск адреса свободной страницы

 ad[g]=1; opred[nomer]=g; //сохранение адреса

lcd\_gotoxy(0,2); lcd\_puts(string\_58); delay\_ms(500);

l=0;

 while(l==0){

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xDF;

delay\_ms(5);

if(PINC.5==0){delay\_ms(50);

 if(PINC.5==0){d=1;l=1;};};

 PORTD|=0x38;

delay\_ms(5);

PORTD&=0xF7;

delay\_ms(5);

if(PINC.5==0){delay\_ms(50);

 if(PINC.5==0){d=2;l=1;};};};l=0; chn=0; n=0;

if(d==1){ //функция записи данных текущей точки

i2c\_start();

i2c\_write(0xA0);

i2c\_write(g>>2);

i2c\_write(g<<6);

i2c\_write (buk[0]);

i2c\_write(buk[1]);

i2c\_write(buk[2]);

i2c\_write(buk[3]);

i2c\_write(buk[4]);

i2c\_write(buk[5]);

i2c\_write(buk[6]);

i2c\_write(buk[7]);

i2c\_write(buk[8]);

i2c\_write(buk[9]);

i2c\_write(buk[10]);

i2c\_write(buk[11]);

i2c\_write(buk[12]);

i2c\_write(buk[13]);

i2c\_write(buk[14]);

i2c\_write(buk[15]);

i2c\_write(gr\_sh);

i2c\_write(min\_sh);

i2c\_write(sek\_sh/100);

i2c\_write(sek\_sh%100);

i2c\_write(gr\_dol);

i2c\_write(min\_dol);

i2c\_write(sek\_dol/100);

i2c\_write(sek\_dol%100);

i2c\_write(h/10);

i2c\_write(h%10);

i2c\_stop();

delay\_ms(10);};

if(d==2){ //набор координат с карты

m\_4:lcd\_clear(); lcd\_gotoxy(4,0); if(n==0){lcd\_puts(string\_59);delay\_ms(500);};

if(n==1){lcd\_puts(string\_60);delay\_ms(500);};

l=0;

 while(l==0){

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xDF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=1;lcd\_putsf(string\_1);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=4;lcd\_putsf(string\_4);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=7;lcd\_putsf(string\_7);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){

if(n==0){

oper=ik[0]\*10+ik[1];oper1=ik[2]\*10+ik[3];oper2=ik[4]\*10+ik[5];oper3=ik[6]\*10+ik[7];

chn=0;++n;goto m\_4;};

if(n==1){oper4=ik[0]\*10+ik[1];oper5=ik[2]\*10+ik[3];oper6=ik[4]\*10+ik[5];oper7=ik[6]\*10+ik[7];

l=1;};};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xEF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=2;lcd\_putsf(string\_2);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=5;lcd\_putsf(string\_5);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=8;lcd\_putsf(string\_8);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=0;lcd\_putsf(string);delay\_ms(300);++chn;};};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xF7;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=3;lcd\_putsf(string\_3);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=6;lcd\_putsf(string\_6);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=7){lcd\_gotoxy((4+chn),1);

 ik[chn]=9;lcd\_putsf(string\_9);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){--chn;ik[chn]=0;lcd\_gotoxy((4+chn),1);

lcd\_putsf(string\_30);delay\_ms(300);};};};

 i2c\_start(); //запись координат с карт

i2c\_write(0xA0);

i2c\_write(g>>2);

i2c\_write(g<<6);

i2c\_write (buk[0]);

i2c\_write(buk[1]);

i2c\_write(buk[2]);

i2c\_write(buk[3]);

i2c\_write(buk[4]);

i2c\_write(buk[5]);

i2c\_write(buk[6]);

i2c\_write(buk[7]);

i2c\_write(buk[8]);

i2c\_write(buk[9]);

i2c\_write(buk[10]);

i2c\_write(buk[11]);

i2c\_write(buk[12]);

i2c\_write(buk[13]);

i2c\_write(buk[14]);

i2c\_write(buk[15]);

i2c\_write(oper);

i2c\_write(oper1);

i2c\_write(oper2);

i2c\_write(oper3);

i2c\_write(oper4);

i2c\_write(oper5);

i2c\_write(oper6);

i2c\_write(oper7);

i2c\_write(0);

i2c\_write(0);

i2c\_stop();

delay\_ms(10);};

lcd\_clear(); lcd\_gotoxy(0,1);lcd\_putsf(string\_45);//точка записана

delay\_ms(2000); lcd\_clear();

for(chn=0;chn<20;++chn){buk[chn]=0;};chn=0;kod=0;n=0;g=0;l=0;d=0;ld=0;};};

 if(PINC.7==0){delay\_ms(50); if(PINC.7==0){flag=1;i=0;y=0;nahalo=0; //вход в режим чтения

m\_3: if(nomer==0){ lcd\_clear(); lcd\_gotoxy(2,1);

 lcd\_putsf(string\_47); delay\_ms(2000);lcd\_clear();goto m;};

 lcd\_clear(); lcd\_gotoxy(2,0);

 lcd\_putsf(string\_44);// точкa

 lcd\_gotoxy(7,0);lcd\_puts(":");

m\_1:l=0;

 while(l==0){ //набор номера точки

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xDF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=1;lcd\_putsf(string\_1);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=4;lcd\_putsf(string\_4);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=7;lcd\_putsf(string\_7);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){

switch(chn){

case 1:nomer\_1=buk[0];

break;

case 2:nomer\_1=buk[0]\*10+buk[1];

break;

case 3:nomer\_1=buk[0]\*100+buk[1]\*10+buk[2];};l=1;};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xEF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=2;lcd\_putsf(string\_2);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=5;lcd\_putsf(string\_5);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=8;lcd\_putsf(string\_8);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=0;lcd\_putsf(string);delay\_ms(300);++chn;};};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xF7;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=3;lcd\_putsf(string\_3);delay\_ms(300);++chn;};};};

if(PINC.3==0){delay\_ms(50); if(PINC.3==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=6;lcd\_putsf(string\_6);delay\_ms(300);++chn;};};};

if(PINC.4==0){delay\_ms(50); if(PINC.4==0){if(chn<=2){lcd\_gotoxy((8+chn),0);

 buk[chn]=9;lcd\_putsf(string\_9);delay\_ms(300);++chn;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){--chn;buk[chn]=0;lcd\_gotoxy((8+chn),0);

lcd\_putsf(string\_30);delay\_ms(300);};};};

delay\_ms(50);lcd\_clear();

 for(chn=0;chn<20;++chn){buk[chn]=0;}; chn=0;

if(nomer\_1>nomer){lcd\_gotoxy(2,1);lcd\_putsf(string\_47);//нет данных

delay\_ms(2000);lcd\_clear(); lcd\_gotoxy(2,0);

 lcd\_putsf(string\_44);// точкa

lcd\_gotoxy(7,0); lcd\_puts(":");

 goto m\_1;};

m\_2: lcd\_clear();

 sprintf(lcd\_buffer\_10,"%u",nomer\_1);

 lcd\_gotoxy(0,0); lcd\_puts("T");

 lcd\_gotoxy(1,0); lcd\_puts(lcd\_buffer\_10);

address\_1=opred[nomer\_1]>>2; //определение адреса страницы

address\_2=opred[nomer\_1]<<6;

for(chn=0;chn<16;++chn){i2c\_start(); //функция чтения

 i2c\_write(0xA0);

i2c\_write(address\_1);

i2c\_write(address\_2);

i2c\_start();

i2c\_write(0xA1);

buk[chn]=i2c\_read(0);

i2c\_stop();

switch(buk[chn]){

case 0: lcd\_gotoxy(chn,1);lcd\_putsf(string\_30);delay\_ms(5); break;

case 1: lcd\_gotoxy(chn,1);lcd\_putsf(string\_1);delay\_ms(5); break;

case 2: lcd\_gotoxy(chn,1);lcd\_putsf(string\_2);delay\_ms(5); break;

case 3: lcd\_gotoxy(chn,1);lcd\_putsf(string\_3);delay\_ms(5); break;

case 4: lcd\_gotoxy(chn,1);lcd\_putsf(string\_4);delay\_ms(5); break;

case 5: lcd\_gotoxy(chn,1);lcd\_putsf(string\_5);delay\_ms(5); break;

case 6: lcd\_gotoxy(chn,1);lcd\_putsf(string\_6);delay\_ms(5); break;

case 7: lcd\_gotoxy(chn,1);lcd\_putsf(string\_7);delay\_ms(5); break;

case 8: lcd\_gotoxy(chn,1);lcd\_putsf(string\_8);delay\_ms(5); break;

case 9: lcd\_gotoxy(chn,1);lcd\_putsf(string\_9);delay\_ms(5); break;

case 43: lcd\_gotoxy(chn,1);lcd\_putsf(string\_43);delay\_ms(5); break;

case 10: lcd\_gotoxy(chn,1);lcd\_putsf(string\_10);delay\_ms(5); break;

case 11: lcd\_gotoxy(chn,1);lcd\_putsf(string\_11);delay\_ms(5); break;

case 12: lcd\_gotoxy(chn,1);lcd\_putsf(string\_12);delay\_ms(5); break;

case 13: lcd\_gotoxy(chn,1);lcd\_putsf(string\_13);delay\_ms(5); break;

case 14: lcd\_gotoxy(chn,1);lcd\_putsf(string\_14);delay\_ms(5); break;

case 15: lcd\_gotoxy(chn,1);lcd\_putsf(string\_15);delay\_ms(5); break;

case 16: lcd\_gotoxy(chn,1);lcd\_putsf(string\_16);delay\_ms(5); break;

case 17: lcd\_gotoxy(chn,1);lcd\_putsf(string\_17);delay\_ms(5); break;

case 18: lcd\_gotoxy(chn,1);lcd\_putsf(string\_18);delay\_ms(5); break;

case 19: lcd\_gotoxy(chn,1);lcd\_putsf(string\_19);delay\_ms(5); break;

case 20: lcd\_gotoxy(chn,1);lcd\_putsf(string\_20);delay\_ms(5); break;

case 21: lcd\_gotoxy(chn,1);lcd\_putsf(string\_21);delay\_ms(5); break;

case 22: lcd\_gotoxy(chn,1);lcd\_putsf(string\_22);delay\_ms(5); break;

case 23: lcd\_gotoxy(chn,1);lcd\_putsf(string\_23);delay\_ms(5); break;

case 24: lcd\_gotoxy(chn,1);lcd\_putsf(string\_24);delay\_ms(5); break;

case 25: lcd\_gotoxy(chn,1);lcd\_putsf(string\_25);delay\_ms(5); break;

case 26: lcd\_gotoxy(chn,1);lcd\_putsf(string\_26);delay\_ms(5); break;

case 27: lcd\_gotoxy(chn,1);lcd\_putsf(string\_27);delay\_ms(5); break;

case 28: lcd\_gotoxy(chn,1);lcd\_putsf(string\_28);delay\_ms(5); break;

case 29: lcd\_gotoxy(chn,1);lcd\_putsf(string\_29);delay\_ms(5); break;

case 30: lcd\_gotoxy(chn,1);lcd\_putsf(string\_30);delay\_ms(5); break;

case 31: lcd\_gotoxy(chn,1);lcd\_putsf(string\_31);delay\_ms(5); break;

case 32: lcd\_gotoxy(chn,1);lcd\_putsf(string\_32);delay\_ms(5); break;

case 33: lcd\_gotoxy(chn,1);lcd\_putsf(string\_33);delay\_ms(5); break;

case 34: lcd\_gotoxy(chn,1);lcd\_putsf(string\_34);delay\_ms(5); break;

case 35: lcd\_gotoxy(chn,1);lcd\_putsf(string\_35);delay\_ms(5); break;

case 36: lcd\_gotoxy(chn,1);lcd\_putsf(string\_36);delay\_ms(5); break;

case 37: lcd\_gotoxy(chn,1);lcd\_putsf(string\_37);delay\_ms(5); break;

case 38: lcd\_gotoxy(chn,1);lcd\_putsf(string\_38);delay\_ms(5); break;

case 39: lcd\_gotoxy(chn,1);lcd\_putsf(string\_39);delay\_ms(5); break;

case 40: lcd\_gotoxy(chn,1);lcd\_putsf(string\_40);delay\_ms(5); break;

case 41: lcd\_gotoxy(chn,1);lcd\_putsf(string\_41);delay\_ms(5); break;

case 100: lcd\_gotoxy(chn,1);lcd\_putsf(string);delay\_ms(5); break;

case 42: lcd\_gotoxy(chn,1);lcd\_putsf(string\_42);delay\_ms(5); };++address\_2;};

for(chn=0;chn<20;++chn){buk[chn]=0;}; chn=0;

switch(bl){case 0:

for(chn=0;chn<10;++chn){i2c\_start();

 i2c\_write(0xA0);

i2c\_write(address\_1);

i2c\_write(address\_2);

i2c\_start();

i2c\_write(0xA1);

koor[chn]=i2c\_read(0);

i2c\_stop();

delay\_ms(5);

++address\_2;};

sprintf(lcd\_buffer\_1,"%u\xdf",koor[0]);

sprintf(lcd\_buffer\_2,"%u",koor[1]);

sprintf(lcd\_buffer\_3,"%u.%u",koor[2],koor[3]);

sprintf(lcd\_buffer\_4,"%u\xdf",koor[4]);

sprintf(lcd\_buffer\_5,"%u",koor[5]);

sprintf(lcd\_buffer\_6,"%u.%u",koor[6],koor[7]);

sprintf(lcd\_buffer\_8,"%u.%uM",koor[8],koor[9]);

 lcd\_gotoxy(1,2);

 lcd\_puts(lcd\_buffer\_1);

 lcd\_gotoxy(4,2);

lcd\_puts(lcd\_buffer\_2);

lcd\_gotoxy(6,2);

lcd\_puts(string\_53);

lcd\_gotoxy(7,2);

lcd\_puts(lcd\_buffer\_3);

lcd\_gotoxy(12,2);

lcd\_puts(string\_54);

lcd\_gotoxy(13,2);

lcd\_puts(string\_55);

lcd\_gotoxy(1,3);

 lcd\_puts(lcd\_buffer\_4);

 lcd\_gotoxy(4,3);

lcd\_puts(lcd\_buffer\_5);

lcd\_gotoxy(6,3);

lcd\_puts(string\_53);

lcd\_gotoxy(7,3);

lcd\_puts(lcd\_buffer\_6);

lcd\_gotoxy(12,3);

lcd\_puts(string\_54);

lcd\_gotoxy(13,3);

lcd\_puts(string\_56);

lcd\_gotoxy(5,0);

lcd\_puts(lcd\_buffer\_8);

for(chn=0;chn<10;++chn){koor[chn]=0;}; chn=0;

break;

case 1: //режим наведения. Запись координат в eeprom микроконтроллера

for(chn=0;chn<8;++chn){i2c\_start();

 i2c\_write(0xA0);

i2c\_write(address\_1);

i2c\_write(address\_2);

i2c\_start();

i2c\_write(0xA1);

kr[chn]=i2c\_read(0);

i2c\_stop();

delay\_ms(5);

++address\_2;};

 chn=0;

lcd\_gotoxy(3,2);lcd\_puts(string\_57); bl=0; blokir=1; delay\_ms(2000); lcd\_clear();goto m;};

r=0;

while(r==0){ PORTD|=0x38; //здесь данные листаются, стираются все или по точкам

delay\_ms(5);

PORTD&=0xF7;

delay\_ms(5);

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){if((z==1)||(x1==1)){z=0;x1=0;

lcd\_gotoxy(12,0);lcd\_putsf(string\_50);delay\_ms(500);}else{

lcd\_clear(); PORTD|=0x38;r=1;};};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xEF;

delay\_ms(5);

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){++nomer\_1; if(nomer\_1<=nomer){goto m\_2;}

else {--nomer\_1;};};};

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){--nomer\_1;if(nomer\_1>0){goto m\_2;}

else{++nomer\_1;};};};

PORTD|=0x38;

delay\_ms(5);

PORTD&=0xDF;

delay\_ms(5);

if(PINC.5==0){delay\_ms(50); if(PINC.5==0){++z; switch(z){

case 1:if(x1==1){x2=1;x1=0;z=0;r=1;}else{

lcd\_gotoxy(12,0);lcd\_putsf(string\_48);delay\_ms(500);};

break;

case 2:x=1;r=1;z=0;};};};

if(PINC.2==0){delay\_ms(50); if(PINC.2==0){if(z==1){x1=1;z=0;lcd\_gotoxy(12,0);

lcd\_putsf(string\_49);delay\_ms(500);};};};};

if(x==1){ad[nomer\_1]=0; for(j=nomer\_1;j<nomer;++j){opred[j]=opred[j+1];};j=0;

--nomer;if(nomer==0){lcd\_clear(); lcd\_gotoxy(2,2);lcd\_putsf(string\_51);delay\_ms(2000);

lcd\_clear();x=0;goto m;};

 if(nomer\_1>nomer){--nomer\_1;};x=0; goto m\_2;};

if(x2==1){ for(j=0;j<201;++j){ad[j]=0;};j=0;x2=0;nomer=0;

lcd\_clear(); lcd\_gotoxy(2,2);lcd\_putsf(string\_51);delay\_ms(2000);lcd\_clear();};

};};};}