

```

#include <avr/io.h>
#include <avr/delay.h>

//ST7565R 128*32
//4L-SPI ATMEGA16A 4M

//#define SCK          7          //PB7          SCK
//#define SDI          5          //PB5          MOSI

#define DC            3          //DC=0 DC=1          A0
#define RST          2          //RESET=0          /RES
#define CS            4          //CS=0          /CS

#define DC_SET        (PORTB|=(1<<DC))
#define RST_SET        (PORTB|=(1<<RST))
#define CS_SET        (PORTB|=(1<<CS))

#define DC_CLR        (PORTB &~(1<<DC))
#define RST_CLR        (PORTB &~(1<<RST))
#define CS_CLR        (PORTB &~(1<<CS))

#define NOP()          asm("nop")
#define uchar          unsigned char

unsigned char ascii[];

//
void Initial(void)
{
    RST_SET;
    _delay_ms(1);
    RST_CLR;
    _delay_ms(2);
    RST_SET;
    _delay_ms(12);

    write_ins(0xE2); //
    _delay_us(10);

    write_ins(0xA2); //SET LCD BIAS(#0A2 1/9;1/8;1/6 or #0A3 1/7;1/6;1/5)
    _delay_us(10);

    write_ins(0xA0); //ADC SELECT ON (#A0H #A1H) segment
    _delay_us(10);

    write_ins(0xC8); //SHL SELECT COM1-COM64(#0C0H #0C8) common
    _delay_us(10);

    write_ins(0x22); //REGULATOR RESISTOR SELECT (R2 R1 R0) RR 0X20-0X27(3.0-6.5)
    _delay_us(10);

    write_ins(0x81); //ELECTRONIC VOLUME SET (LCD)
    _delay_us(10);

    write_ins(0x12); //SET EV(0X00-0X3F)
    _delay_us(10);

    write_ins(0x2F); //SET POWER CONTROL (VB VR VF)=111
    _delay_us(10);
}
//V0=RR*[(99+EV)/162]*2.1 = (4.0*(99+18)/162*2.1)=6.06V
/*****

```

Function Name: Clear screen  
Function: Write DAT data to CGRAM  
Entry Parameters: dat  
Export Parameters:  
Remarks: DDRAM 128\*32

```
*****/
void clear_display(uchar dat)
{
    uchar page,column;
    write_ins(0xAF);           //
    write_ins(0x40);          //
    for(page=0;page<4;page++)
    {
        write_ins(0xB0 | page); //
        write_ins(0x10);        //
        write_ins(0x00);        //
        for(column=0;column<128;column++)
        {
            write_dat(dat);
        }
    }
}
*****/
```

Function Name: Display Frame  
Function:  
Entry Parameters:  
Export Parameters:  
Remarks: DDRAM 128\*32

```
*****/
box_display()
{
    uchar x,y;
    write_ins(0x40);
    write_ins(0xB0 );        //
    write_ins(0x10);         //
    write_ins(0x01);         //
    for(x=0;x<126;x++)
        write_dat(0x01);

    write_ins(0x40);
    write_ins(0xB3 );        //
    write_ins(0x10);         //
    write_ins(0x01);         //
    for(x=0;x<126;x++)
        write_dat(0x80);

    for(y=0;y<4;y++)
    {
        write_ins(0x40);
        write_ins(y|0xB0 );    //
        write_ins(0x10);       //
        write_ins(0x00);       //
        write_dat(0xFF);

        write_ins(0x40);
        write_ins(y|0xB0 );    //
        write_ins(0x17);       //
        write_ins(0x0F);       //
        write_dat(0xFF);
    }
}
*****/
```

```
void write_ins(unsigned char cmd)
{
```

```

        DC_CLR;
        CS_CLR;
        SPDR = cmd;
        while(!(SPSR & (1<<SPIF)))
        ;
        CS_SET;
    }

void write_dat(unsigned char dat)
{
    DC_SET;
    CS_CLR;
    SPDR = dat;
    while(!(SPSR & (1<<SPIF)))
    ;
    CS_SET;
}

/*****
Function Name: SPI Initialization
Function:
Entry Parameters:
Export Parameters:
Remarks:
*****/
void spi_init(void)
{
    DDRB |= (1<<DDB2)|(1<<DDB3)|(1<<DDB5)|(1<<DDB7)|(1<<DDB4);
    //MOSI SCK AND SS
    SPCR = (1<<SPE)|(1<<MSTR);
}

/*****
int main(void)
{
    uchar page, x;
    unsigned int i;

    spi_init();
    Initial();
    while(1)
    {
        clear_display(0x00);
        box_display();
        _delay_ms(500);
        clear_display(0xff);
        _delay_ms(500);

        i=0;
        for(page=0;page<4;page++)
        {
            write_ins(0x40);
            write_ins(page|0xB0);
            write_ins(0x10);
            write_ins(0x00);
            for(x=0;x<128;x++)
            {
                i++;
                write_dat(ascii[i]);
            }
        }
        _delay_ms(500);
    }
}

```

}

```
uchar ascii[]={0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00, // " "=00
0x00,0x00,0x00,0x4F,0x00,0x00,0x00,0x00, // "! "=01
0x00,0x00,0x07,0x00,0x07,0x00,0x00,0x00, // ". "=02
0x00,0x14,0x7F,0x14,0x7F,0x14,0x00,0x00, // "# "=03
0x00,0x24,0x2A,0x7F,0x2A,0x12,0x00,0x00, // "$ "=04
0x00,0x23,0x13,0x08,0x64,0x62,0x00,0x00, // "% "=05
0x00,0x36,0x49,0x55,0x22,0x50,0x00,0x00, // "& "=06
0x00,0x00,0x05,0x03,0x00,0x00,0x00,0x00, // "' "=07
0x00,0x00,0x1C,0x22,0x41,0x00,0x00,0x00, // "( "=08
0x00,0x00,0x41,0x22,0x1C,0x00,0x00,0x00, // ") "=09
0x00,0x14,0x08,0x3E,0x08,0x14,0x00,0x00, // "* "=0A
0x00,0x08,0x08,0x3E,0x08,0x08,0x00,0x00, // "+ "=0B
0x00,0x00,0x50,0x30,0x00,0x00,0x00,0x00, // "/ "=0C
0x00,0x08,0x08,0x08,0x08,0x08,0x00,0x00, // "- "=0D
0x00,0x00,0x60,0x60,0x00,0x00,0x00,0x00, // ". "=0E
0x00,0x20,0x10,0x08,0x04,0x02,0x00,0x00, // "/" "=0F
0x00,0x3E,0x51,0x49,0x45,0x3E,0x00,0x00, // "0 "=10
0x00,0x00,0x42,0x7F,0x40,0x00,0x00,0x00, // "1 "=11
0x00,0x42,0x61,0x51,0x49,0x46,0x00,0x00, // "2 "=12
0x00,0x21,0x41,0x45,0x4B,0x31,0x00,0x00, // "3 "=13
0x00,0x18,0x14,0x12,0x7F,0x10,0x00,0x00, // "4 "=14
0x00,0x27,0x45,0x45,0x45,0x39,0x00,0x00, // "5 "=15
0x00,0x3C,0x4A,0x49,0x49,0x30,0x00,0x00, // "6 "=16
0x00,0x01,0x01,0x79,0x05,0x03,0x00,0x00, // "7 "=17
0x00,0x36,0x49,0x49,0x49,0x36,0x00,0x00, // "8 "=18
0x00,0x06,0x49,0x49,0x29,0x1E,0x00,0x00, // "9 "=19
0x00,0x00,0x36,0x36,0x00,0x00,0x00,0x00, // ":" "=1A
0x00,0x00,0x56,0x36,0x00,0x00,0x00,0x00, // "/" "=1B
0x00,0x08,0x14,0x22,0x41,0x00,0x00,0x00, // "<" "=1C
0x00,0x14,0x14,0x14,0x14,0x14,0x00,0x00, // "=" "=1D
0x00,0x00,0x14,0x22,0x14,0x08,0x00,0x00, // ">" "=1E
0x00,0x02,0x01,0x51,0x09,0x06,0x00,0x00, // "?" "=1F
0x00,0x32,0x49,0x79,0x41,0x3E,0x00,0x00, // "@" "=20
0x00,0x7E,0x11,0x11,0x11,0x7E,0x00,0x00, // "A" "=21
0x00,0x41,0x7F,0x49,0x49,0x36,0x00,0x00, // "B" "=22
0x00,0x3E,0x41,0x41,0x41,0x22,0x00,0x00, // "C" "=23
0x00,0x41,0x7F,0x41,0x41,0x3E,0x00,0x00, // "D" "=24
0x00,0x7F,0x49,0x49,0x49,0x49,0x00,0x00, // "E" "=25
0x00,0x7F,0x09,0x09,0x09,0x01,0x00,0x00, // "F" "=26
0x00,0x3E,0x41,0x41,0x41,0x49,0x7A,0x00,0x00, // "G" "=27
0x00,0x7F,0x08,0x08,0x08,0x7F,0x00,0x00, // "H" "=28
0x00,0x00,0x41,0x7F,0x41,0x00,0x00,0x00, // "I" "=29
0x00,0x20,0x40,0x41,0x3F,0x01,0x00,0x00, // "J" "=2A
0x00,0x7F,0x08,0x14,0x22,0x41,0x00,0x00, // "K" "=2B
0x00,0x7F,0x40,0x40,0x40,0x40,0x00,0x00, // "L" "=2C
0x00,0x7F,0x02,0x0C,0x02,0x7F,0x00,0x00, // "M" "=2D
0x00,0x7F,0x06,0x08,0x30,0x7F,0x00,0x00, // "N" "=2E
0x00,0x3E,0x41,0x41,0x41,0x3E,0x00,0x00, // "O" "=2F
0x00,0x7F,0x09,0x09,0x09,0x06,0x00,0x00, // "P" "=30
0x00,0x3E,0x41,0x51,0x21,0x5E,0x00,0x00, // "Q" "=31
0x00,0x7F,0x09,0x19,0x29,0x46,0x00,0x00, // "R" "=32
0x00,0x26,0x49,0x49,0x49,0x32,0x00,0x00, // "S" "=33
0x00,0x01,0x01,0x7F,0x01,0x01,0x00,0x00, // "T" "=34
0x00,0x3F,0x40,0x40,0x40,0x3F,0x00,0x00, // "U" "=35
0x00,0x1F,0x20,0x40,0x20,0x1F,0x00,0x00, // "V" "=36
0x00,0x7F,0x20,0x18,0x20,0x7F,0x00,0x00, // "W" "=37
0x00,0x63,0x14,0x08,0x14,0x63,0x00,0x00, // "x" "=38
0x00,0x07,0x08,0x70,0x08,0x07,0x00,0x00, // "Y" "=39
0x00,0x61,0x51,0x49,0x45,0x43,0x00,0x00, // "Z" "=3A
0x00,0x00,0x7F,0x41,0x41,0x00,0x00,0x00, // "[" "=3B
0x00,0x02,0x04,0x08,0x10,0x20,0x00,0x00, // "\" "=3C
0x00,0x00,0x41,0x41,0x7F,0x00,0x00,0x00, // "]" "=3D
```

0x00, 0x04, 0x02, 0x01, 0x02, 0x04, 0x00, 0x00, //"^ "=3E  
0x00, 0x40, 0x40, 0x40, 0x40, 0x40, 0x00, 0x00, //"\_ "=3F  
0x00, 0x01, 0x02, 0x04, 0x00, 0x00, 0x00, 0x00, //"|=40  
0x00, 0x20, 0x54, 0x54, 0x54, 0x78, 0x00, 0x00, //"a "=41  
0x00, 0x7F, 0x48, 0x44, 0x44, 0x38, 0x00, 0x00, //"b "=42  
0x00, 0x38, 0x44, 0x44, 0x44, 0x28, 0x00, 0x00, //"c "=43  
0x00, 0x38, 0x44, 0x44, 0x48, 0x7F, 0x00, 0x00, //"d "=44  
0x00, 0x38, 0x54, 0x54, 0x54, 0x18, 0x00, 0x00, //"e "=45  
0x00, 0x00, 0x08, 0x7E, 0x09, 0x02, 0x00, 0x00, //"f "=46  
0x00, 0x0C, 0x52, 0x52, 0x4C, 0x3E, 0x00, 0x00, //"g "=47  
0x00, 0x7F, 0x08, 0x04, 0x04, 0x78, 0x00, 0x00, //" "=48  
0x00, 0x00, 0x44, 0x7D, 0x40, 0x00, 0x00, 0x00, //"i "=49  
0x00, 0x20, 0x40, 0x44, 0x3D, 0x00, 0x00, 0x00, //"j "=4A  
0x00, 0x00, 0x7F, 0x10, 0x28, 0x44, 0x00, 0x00, //"k "=4B  
0x00, 0x00, 0x41, 0x7F, 0x40, 0x00, 0x00, 0x00, //"l "=4C  
0x00, 0x7C, 0x04, 0x78, 0x04, 0x78, 0x00, 0x00, //"m "=4D  
0x00, 0x7C, 0x08, 0x04, 0x04, 0x78, 0x00, 0x00, //"n "=4E  
0x00, 0x38, 0x44, 0x44, 0x44, 0x38, 0x00, 0x00, //"o "=4F  
0x00, 0x7E, 0x0C, 0x12, 0x12, 0x0C, 0x00, 0x00, //"p "=50  
0x00, 0x0C, 0x12, 0x12, 0x0C, 0x7E, 0x00, 0x00, //"q "=51  
0x00, 0x7C, 0x08, 0x04, 0x04, 0x08, 0x00, 0x00, //"r "=52  
0x00, 0x58, 0x54, 0x54, 0x54, 0x64, 0x00, 0x00, //"s "=53  
0x00, 0x04, 0x3F, 0x44, 0x40, 0x20, 0x00, 0x00, //"t "=54  
0x00, 0x3C, 0x40, 0x40, 0x3C, 0x40, 0x00, 0x00, //"u "=55  
0x00, 0x1C, 0x20, 0x40, 0x20, 0x1C, 0x00, 0x00, //"v "=56  
0x00, 0x3C, 0x40, 0x30, 0x40, 0x3C, 0x00, 0x00, //"w "=57  
0x00, 0x44, 0x28, 0x10, 0x28, 0x44, 0x00, 0x00, //"x "=58  
0x00, 0x1C, 0xA0, 0xA0, 0x90, 0x7C, 0x00, 0x00, //"y "=59  
0x00, 0x44, 0x64, 0x54, 0x4C, 0x44, 0x00, 0x00, //"z "=5A  
0x00, 0x00, 0x08, 0x36, 0x41, 0x00, 0x00, 0x00, //"{" "=5B  
0x00, 0x00, 0x00, 0x77, 0x00, 0x00, 0x00, 0x00, //"|" "=5C  
0x00, 0x00, 0x00, 0x41, 0x36, 0x08, 0x00, 0x00, //"}" "=5D  
0x00, 0x02, 0x01, 0x02, 0x04, 0x02, 0x00, 0x00, //"~" "=5E  
0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, //"" "=5F

};