

// This example program drives single HT16D31A via SPI interface and display the LED matrix in checkerboard pattern.

```
#include "HT66F2390.h"
```

```
#define csb      _pa6
```

```
#define DIO      _pa3
```

```
#define DIOC     _pac3
```

```
#define clk      _pa5
```

```
void IO_INIT();
```

```
void LED_INIT();
```

```
void LED_on();
```

```
void clear_ramdata();
```

```
void Software_Reset();
```

```
void option_led_command(unsigned char CMD,unsigned char ADDR);
```

```
void func_led_command(unsigned char CMD,unsigned char ADDR,unsigned char DATA);
```

```
void main()
```

```
{
```

```
    unsigned char i,j;
```

```
    _wdtc=0xAF;
```

```
    //close WDT
```

```
    IO_INIT();
```

```
    LED_INIT();
```

```
        for(i=0;i<144;i=i+32)
```

```
        //
```

```
write RAM data for Checkerboard
```

```
{
```

```
    func_led_command(0x80,i,255);
```

```
    func_led_command(0x80,i+2,255);
```

```
    func_led_command(0x80,i+4,255);
```

```
    func_led_command(0x80,i+6,255);
```

```
    func_led_command(0x80,i+0x11,255);
```

```
    func_led_command(0x80,i+0x13,255);
```

```
    func_led_command(0x80,i+0x15,255);
```

```
    func_led_command(0x80,i+0x17,255);
```

```
}
```

```
}
```

```
void IO_INIT()
```

```
{
```

```
    _pas0=0x00;
```

```
    _pcc=0xff;
```

```
    _pcpu=0xff;
```

```
    _pac=0x00;
```

```
    csb=1;
```

```
    DIO=0;
```

```
    clk=1;
```

```

}
void func_led_command(unsigned char CMD,unsigned char ADDR,unsigned
char DATA)
{
    unsigned char i,Send_temp = 0;

    csb=0;
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = CMD&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        CMD = CMD<<1;
    }
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = ADDR&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        ADDR = ADDR<<1;
    }
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = DATA&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        DATA = DATA<<1;
    }

    csb=1;

```

```

}
void option_led_command(unsigned char CMD,unsigned char ADDR)
{
    unsigned char i,Send_temp = 0;

    csb=0;
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = CMD&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        CMD = CMD<<1;
    }
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    GCC_NOP();
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = ADDR&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        ADDR = ADDR<<1;
    }
    csb=1;
}
void LED_INIT()
{
    Software_Reset();
    unsigned char i;
    clear_ramdata();

    option_led_command(0x31,0x00); //
gray mode
    option_led_command(0x35,0x02); //
SYS ON、DISPLAY OFF
    option_led_command(0x36,0x0A); //
Constant Current Ratio

    LED_on(); //LED ON

```

```

        opction_led_command(0x39,0x8F); //
Blanking Voltage Control
        opction_led_command(0x37,0xff); //
Luminance Control

```

```

        clear_ramdata();
        opction_led_command(0x35,0x03); //
SYS ON \ DISPLAY ON
}

```

```

void clear_ramdata()
{
    unsigned int i;
    opction_led_command(0xFD,0x00);
    for(i=0;i<=144;i++)
    {
        func_led_command(0x80,i,0x00);
    }
    opction_led_command(0xFD,0x01);
    for(i=0;i<=144;i++)
    {
        func_led_command(0x80,i,0x00);
    }
    opction_led_command(0xFD,0x00);
}

```

```

void Software_Reset()
{
    unsigned char i,CMD,Send_temp = 0;
    CMD=0xcc;
    csb=0;
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = CMD&0x80;
        DIO = 1;
        if(Send_temp == 0)
            DIO = 0;
        GCC_NOP();
        clk=1;
        CMD = CMD<<1;
    }
    GCC_NOP();
    csb=1;
}

```

```

void LED_on()
{
    unsigned char i,j,CMD,ADDR,DATA,Send_temp;
    csb=0;

```

```

        CMD=0x84;
        ADDR=0;
        DATA=255;
for(i =0;i<8;i++)
{
    clk=0;
    Send_temp = CMD&0x80;
    DIO = 1;
    if(Send_temp == 0)
    DIO = 0;
    GCC_NOP();
    clk=1;
    CMD = CMD<<1;
}
GCC_NOP();
GCC_NOP();
GCC_NOP();
GCC_NOP();
for(i =0;i<8;i++)
{
    clk=0;
    Send_temp = ADDR&0x80;
    DIO = 1;
    if(Send_temp == 0)
    DIO = 0;
    GCC_NOP();
    clk=1;
    ADDR = ADDR<<1;
}
GCC_NOP();
GCC_NOP();
GCC_NOP();
GCC_NOP();

for(j=0;j<9;j++)
{
    DATA=255;
    for(i =0;i<8;i++)
    {
        clk=0;
        Send_temp = DATA&0x80;
        DIO = 1;
        if(Send_temp == 0)
        DIO = 0;
        GCC_NOP();
        clk=1;
        DATA = DATA<<1;
    }
}

```

```
    csb=1;  
}
```