```
//Note: For the best effect, we suggest you view the program with the Source
Insight software.

//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
//  Prorgam:         79661
//  Author:          XING TAI
//  Date:            2024
//  Rev:             1.0
//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
//
//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

/*************************************************************************
****************************************
*Header File
*************************************************************************
****************************************/



#include "msp430x22x4.h"
#include "image.h"


/******EPD Resolution******/
// EPD Resolution Settings
#define GATES            240
#define SOURCES          240

#define SDA_H             (P1OUT |=BIT7)
// P1.7
#define SDA_L             (P1OUT &=~BIT7)
#define SCLK_H            (P1OUT |=BIT6)
// P1.6
#define SCLK_L            (P1OUT &=~BIT6)
#define nCS_H             (P1OUT |=BIT5)
// P1.5
#define nCS_L             (P1OUT &=~BIT5)
#define nDC_H             (P1OUT |=BIT4)
// P1.4
#define nDC_L             (P1OUT &=~BIT4)
#define nRST_H            (P1OUT |=BIT3)
// P1.3
#define nRST_L            (P1OUT &=~BIT3)
#define R_SDA            0x80
//P1.7

#define VDD_OFF           (P2OUT |=BIT3)
// P2.0
#define VDD_ON            (P2OUT &=~BIT3)

#define PIC_YELLOW        253
#define PIC_RED          254
#define PIC_WHITE         255
#define PIC_BLACK         0
```

```
#define PIC_NULL            1
#define PIC_A               2
#define PIC_B          3
#define PIC_G           4
#define PIC_F          5      //VLINE
#define PIC_C               6
#define PIC_D          7
#define PIC_E              8
#define PIC_DOT            9
#define PIC_VLINE          10
#define PIC_HLINE             11


enum Vol_Value_e
{
    _2_3V = 0,
    _2_4V,
    _2_5V,
    _3_0V,
    _3_6V

};

#define SOURCE_SCAN_NUM           (SOURCES/4)

#define GATE_FIRST_REGION          (GATES/4)
#define GATE_SECOND_REGION         (GATES/2)
#define GATE_THIRD_REGION          (GATES*3/4)

#define SOURCE_FIRST_REGION        (SOURCE_SCAN_NUM/4)
#define SOURCE_SECOND_REGION       (SOURCE_SCAN_NUM/2)
#define SOURCE_THIRD_REGION        (SOURCE_SCAN_NUM*3/4)


/*************************************
*Function-Delay Function (mS)
*Parameters-time: mS
*Back-void
*************************************/
void Delay_ms(int time)
// 1ms
{
    int i,j;

    for(i=0; i<time; i++)
        for(j=0; j<1600; j++);
}

/*************************************
*Function-Delay Function (S)
*Parameters-time: S
*Back-void
*************************************/
void Delay_S(int time)
// 1s
{
```

```
    int i,j,k;

    for(i=0; i<time; i++)
        for(j=0; j<1000; j++)
            for(k=0; k<1600; k++);
}

/***************************************
*Function-IC Busy Read Signal
*Parameters-void
*Back-void
***************************************/
void ReadBusy(void)
{
    while(1)
    {
        _NOP();
        if(P1IN & 0x04) break;
    }
}

/***************************************
*Function-IC Reset Signal
*Parameters-void
*Back-void
***************************************/
void Reset(void)
{
    nRST_L;
    Delay_ms(20);
    nRST_H;
    Delay_ms(10);
   nRST_L;
    Delay_ms(20);
    nRST_H;
    Delay_ms(10);
   ReadBusy();
   Delay_ms(10);
}

/***************************************
*Function-IC_SPI Enter command
*Parameters-init_com: Command Code
*Back-void
***************************************/
void SPI4W_WriteCom(unsigned char init_com)
{
  unsigned char temp_com;
  unsigned char scnt;

   P1DIR |= R_SDA;
   temp_com = init_com;
   nCS_H;
   nCS_L;
//SPI chip select the signal, It will be active at a low level.
   SCLK_L;
   nDC_L;
```

```
    for(scnt=0; scnt<8; scnt++)
    {
        if(temp_com & 0x80)
            SDA_H;
        else
            SDA_L;
        SCLK_H;
        SCLK_L;
        temp_com = temp_com << 1;
    }
    nCS_H;
}

/*************************************
*Function-IC_SPI Enter data
*Parameters-init_data: Data Content
*Back-void
*************************************/
void SPI4W_WriteData(unsigned char init_data)
{
    unsigned char temp_data;
    unsigned char scnt;

    P1DIR |= R_SDA;
    temp_data = init_data;
    nCS_H;
    nCS_L;
//SPI chip select the signal, It will be active at a low level.
    SCLK_L;
    nDC_H;
    for(scnt=0;scnt<8;scnt++)
    {
    if(temp_data & 0x80)
          SDA_H;
       else
          SDA_L;
    SCLK_H;
    SCLK_L;
    temp_data = temp_data<<1;
    }
    nCS_H;
}


/*************************************
*Function-IC_SPIRead Data
*Parameters-void
*Back-unsigned char one byte
*************************************/
unsigned char SPI4W_ReadData(void)
{
    unsigned char scnt,temp;

    P1DIR &= ~R_SDA;
    nCS_H;
    nCS_L;
//SPI chip select the signal, It will be active at a low level.
```

```
    //SCLK_H;
    SCLK_L;
    nDC_H;
    for(scnt=0; scnt<8; scnt++)
    {
      if(P1IN&R_SDA)
        temp = (temp<<1) | 0x01;
      else
        temp = temp << 1;
      SCLK_H;
      SCLK_L;
    }
    nCS_H;

    return temp;
}



/*************************************
*Function-Read the IC Temperature
*Parameters-void
*Back-unsigned char one byte
*************************************/
unsigned char ReadTemptr(void)
{
    unsigned char temptr_intgr = 0;
    unsigned char temptr_decml = 0;

    Reset();

    SPI4W_WriteCom(0x40);
    Delay_ms(100);
    temptr_intgr = SPI4W_ReadData();
    temptr_decml = SPI4W_ReadData();

    return temptr_intgr;

}



/*************************************
*Function-Send Display Command
*Parameters-None
*Back-None
*************************************/
void Display_Cmd(void)
{
    SPI4W_WriteCom(0x04);
// Power ON
    ReadBusy();

    SPI4W_WriteCom(0x12);
// Display Refresh
    SPI4W_WriteData(0x00);
    ReadBusy();
```

```
    SPI4W_WriteCom(0x02);
// Power OFF
    SPI4W_WriteData(0x00);
    ReadBusy();
}

/************************************
*Function-Enter Deep Sleep Mode
*Parameters-None
*Back-None
************************************/
void EnterDeepsleep(void)
{
     SPI4W_WriteCom(0x07);
// Deep sleep
     SPI4W_WriteData(0xA5);
}


void Basic_Display(unsigned char dis_type)
{
    unsigned int col = 0, row = 0;
    unsigned int pcnt = 0;



    SPI4W_WriteCom(0x10);
// DTM1 Write

    for(col=0; col<240; col++)
    {
        for(row=0; row<120; row++)
        {
            switch(dis_type)
            {
            case PIC_A:
//                SPI4W_WriteData(gImage_Font[pcnt]);
               break;

            case PIC_VLINE:
               if(col < 16)
                   SPI4W_WriteData(0x00);
               else if(col < 32)
                   SPI4W_WriteData(0x11);
               else if(col < 48)
                   SPI4W_WriteData(0x22);
               else if(col < 64)
                   SPI4W_WriteData(0x33);
               else if(col < 80)
                   SPI4W_WriteData(0x44);
               else if(col < 96)
                   SPI4W_WriteData(0x55);
               else if(col < 112)
                   SPI4W_WriteData(0x66);
               else
                   SPI4W_WriteData(0x77);
```

```
            break;

        case PIC_YELLOW:
            SPI4W_WriteData(0xAA);
            break;
        case PIC_RED:
            SPI4W_WriteData(0xFF);
            break;
        case PIC_BLACK:
            SPI4W_WriteData(0x00);
            break;
        default:
        case PIC_WHITE:
            SPI4W_WriteData(0x11);
            break;
        }
        pcnt++;
      }
   }


   Display_Cmd();
   return;
}



/*************************************
*Functions - System Initialization
*Parameters-void
*Back-void
*************************************/
void Sys_init(void)
{
   WDTCTL = WDTPW + WDTHOLD;
// Stop watchdog timer to prevent time out reset
   BCSCTL1 = CALBC1_8MHZ;
// set DCO frequency 8MHZ
   DCOCTL = CALDCO_8MHZ;

   P1DIR |=0xF8;
// set P1.3~7 output
   P4DIR |=0x0A;
// set  P4.3 P4.1 output   0A should berevised
}

/*************************************
*Function-IC Initialization Settings
*Parameters-void: None
*Back- void
*************************************/
void Init_IC(void)
{

   //FITI cmd.
   SPI4W_WriteCom(0x4D);
   SPI4W_WriteData(0x78);
```

```
    SPI4W_WriteCom(0xE9);
    SPI4W_WriteData(0x01);
    ReadBusy();

}



/*************************Complete IC
Initialization*****************************/


//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxx
//xx   Main Function
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxx
int main(void)
{

    Sys_init();



    Reset();
    ReadBusy();
    Init_IC();
    Basic_Display(PIC_WHITE);//Font
    EnterDeepsleep();
    Delay_S(5);


    while (1)
    {

    };


}
```