//#include "can.h"

#include "p18f25k80.h"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#pragma config IESO = ON , FCMEN = ON, PLLCFG = OFF, FOSC = HS1 /\* CONFIG1H\*/

#pragma config XINST = OFF, SOSCSEL = DIG, INTOSCSEL = LOW, RETEN = OFF /\* CONFIG1L\*/

#pragma config BORPWR = MEDIUM, BORV = 0, PWRTEN = OFF /\* CONFIG2L\*/

#pragma config WDTEN = OFF, BOREN = SBORDIS, WDTPS = 2048 /\* CONFIG2H\*/

#pragma config MCLRE = ON, CANMX = PORTC /\* CONFIG3H\*/

#pragma config STVREN = OFF, BBSIZ = BB2K /\* CONFIG4L\*/

#pragma config CP0 = OFF, CP1 = OFF, CP2 = OFF, CP3 = OFF /\* CONFIG5L\*/

#pragma config CPB = OFF, CPD = OFF /\* CONFIG5H\*/

#pragma config WRT0 = OFF, WRT1 = OFF, WRT2 = OFF, WRT3 = OFF /\* CONFIG6L\*/

#pragma config WRTB = OFF, WRTC = OFF, WRTD = OFF /\* CONFIG6H\*/

#pragma config EBTR0 = OFF, EBTR1 = OFF, EBTR2 = OFF, EBTR3 = OFF /\* CONFIG7L\*/

#pragma config EBTRB = OFF

#define POWER PORTBbits.RB1

#define INT8U unsigned char

#define INT16U unsigned int

#define INT32U unsigned long

#define C\_TIMER\_1MS 0xF84c

#define CAN\_ID 0x05FB

unsigned int Timer\_Count\_Second;

unsigned int LPWM\_Timer\_Count;

unsigned int LPWM\_SUM\_timer;

unsigned char LPWM\_Timer\_Flag;

unsigned int LPWM\_Duty\_Buffer[60]={0};

//unsigned int PM25\_Value\_Data[30]={0};

unsigned char Buffer\_Flag;

unsigned char Buffer\_Flag\_Old=31;

unsigned int PM25\_Value;

unsigned int LPWM\_Duty;

unsigned char CAN\_Timer;

unsigned char CAN\_send\_flag=0;

unsigned char num;

void isr\_high(void);

void Init\_MCU\_Clock(void)

{

OSCTUNEbits.PLLEN=0;

}

void Init\_MCU\_Timer(void)

{

PMD1bits.TMR1MD=0; //enable the timer1

T1GCON = 0X00;

T1CON = 0X06;

RCONbits.IPEN = 1;

PIE1bits.TMR1IE = 1;

IPR1bits.TMR1IP = 1;

INTCONbits.GIEH =1;

T1CONbits.TMR1ON =1;

}

void Init\_interrupt(void)

{

INTCONbits.GIEL =1;

INTCONbits.RBIE =1;

// IOCBbits.IOCB4 =1;

IOCBbits.IOCB5 =1;

}

void Init\_MCU\_IO(void)

{

TRISA = 0xff;

TRISB = 0xfd;

TRISC = 0x3f;

PORTA = 0x00;

PORTB = 0x00;

PORTC = 0x00;

}

void CAN\_Time\_Init()

{

OSCTUNEbits.PLLEN = 1;

OSCCONbits.IRCF = 7; //add

}

void InitCAN\_Init(void)

{

TRISC = (TRISC | 0X80) & 0Xbf;

CANCON = 0x80; //请求配置模式

while(!(CANSTATbits.OPMODE ==0x04)); //确认CAN处于配置模式

//CIOCON = 0X00; //CAN I/O口控制寄存器，设置PLL时钟

ECANCON= 0x00;

CIOCON=0x21;

BRGCON1=0X00;

BRGCON2=0X90;

BRGCON3=0X42;

BSEL0=0x00;

RXB0CON = 0B00100000;

RXB1CON = 0x00;

RXB0DLC = 0X08;

RXB0D0 = 0X00;

RXB0D1 = 0X00;

RXB0D2 = 0X00;

RXB0D3 = 0X00;

RXB0D4 = 0X00;

RXB0D5 = 0X00;

RXB0D6 = 0X00;

RXB0D7 = 0X00;

RXM0SIDH = 0X98;

RXM0SIDL = 0X00;

RXFCON0 = 0x01;

//RXFCON1 = 0x00;

RXF0SIDH = 0x67;

RXF0SIDL = 0xE0;

CANCON=0x00;

while(CANSTATbits.OPMODE==0x00);

PIR5 = 0X00;

PIE5 = PIE5 | 0X00;

IPR5 = IPR5 | 0X00;

PIR5bits.RXB0IF=0;

PIE5bits.RXB0IE=0;

IPR5bits.RXB0IP=0;

RXB0CONbits.RXFUL=0;

}

void CAN\_Transmit(unsigned int Duty,unsigned int Value)

{

TXB0CON = TXB0CON & 0xF7; //发送缓冲寄存器

TXB0SIDH = CAN\_ID>>3; //发送缓冲区标准标识符

TXB0SIDL = (CAN\_ID<<5)%256;

TXB0DLC = 0x08; //发送缓冲区数据长度

TXB0D0 = 0xFF; //发送缓冲区0数据字段字节0

TXB0D1 = 0x3E&num<<1;

TXB0D1 = TXB0D1|(0xc0&Value<<6);

TXB0D0 = 0xff&Value>>2;

TXB0D3 = 0x80&Duty<<7;

TXB0D2 = 0xff&Duty>>1;

TXB0D1 |= 0x01&Duty>>9;

TXB0D6 = 0x00;

TXB0D7 = 0x00;

TXB0CON = TXB0CON | 0x08; //请求发送报文

}

#pragma code isrhighcode = 0x0008 /\*the address is 0x008h\*/

void isr\_high\_direct(void)

{ /\*go to the interupt \*/

\_asm /\*begin in-line assembly \*/

goto isr\_high /\*go to isr\_high function\*/

\_endasm /\*end in-line assembly\*/

}

#pragma code

#pragma interrupt isr\_high /\*this is a interupt function\*/

void isr\_high(void)

{

if(PIR1bits.TMR1IF==1)

{

PIR1bits.TMR1IF=0;

TMR1H=C\_TIMER\_1MS/256;

TMR1L=C\_TIMER\_1MS%256;

if(LPWM\_SUM\_timer) LPWM\_SUM\_timer--;

else

{

LPWM\_SUM\_timer=500;

CAN\_Timer =1;

Buffer\_Flag++;

if(Buffer\_Flag>60)Buffer\_Flag =0;

LPWM\_Duty\_Buffer[Buffer\_Flag] =LPWM\_Timer\_Count;

LPWM\_Timer\_Count=0;

}

if(LPWM\_Timer\_Flag==1) LPWM\_Timer\_Count++;

}

if(INTCONbits.RBIF == 1)

{

INTCONbits.RBIF =0;

if(PORTBbits.RB5 == 0)

LPWM\_Timer\_Flag =1;

else LPWM\_Timer\_Flag =0;

}

}

#pragma code /\*end the program \*/

void main(void)

{

long int LPWM\_SUM;

unsigned char i;

unsigned int Bias\_PM25;

unsigned int LPWM\_Duty\_old;

unsigned int PM25\_Value\_old;

//CAN\_Init( );

Init\_MCU\_Clock();

Init\_MCU\_Timer();

Init\_MCU\_IO();

//CAN\_Time\_Init();

InitCAN\_Init();

Init\_interrupt();

WDTCON = 0x01;

while(1)

{

//CAN\_pro( );

POWER=1;

if(Buffer\_Flag\_Old!=Buffer\_Flag)

{

LPWM\_SUM =0;

for(i=0;i<60;i++)

{

LPWM\_SUM+=LPWM\_Duty\_Buffer[i];

}

LPWM\_Duty =(float)LPWM\_SUM/30000\*1000;

Buffer\_Flag\_Old = Buffer\_Flag;

//CAN\_Transmit();

}

if (LPWM\_Duty>=500) {LPWM\_Duty=500;} //Remove the peaking Value

if(LPWM\_Duty<13) PM25\_Value =LPWM\_Duty\*46;

else if(LPWM\_Duty<23) PM25\_Value =LPWM\_Duty\*60-180;

else if(LPWM\_Duty<46) PM25\_Value =LPWM\_Duty\*13+900;

else if(LPWM\_Duty<58) PM25\_Value =LPWM\_Duty\*58-1180;

else if(LPWM\_Duty<86) PM25\_Value =LPWM\_Duty\*36+130;

else if(LPWM\_Duty<120) PM25\_Value =LPWM\_Duty\*44-590;

else if(LPWM\_Duty<139) PM25\_Value =LPWM\_Duty\*68-3510;

else if(LPWM\_Duty<168) PM25\_Value =LPWM\_Duty\*38+730;

else if(LPWM\_Duty<195) PM25\_Value =LPWM\_Duty\*52-1610;

else if(LPWM\_Duty<220) PM25\_Value =LPWM\_Duty\*60-3200;

//else if(LPWM\_Duty<236) PM25\_Value =LPWM\_Duty\*80-8880;

//else if(LPWM\_Duty<25) PM25\_Value =LPWM\_Duty\*143-23710;

//else if(LPWM\_Duty<270) PM25\_Value =LPWM\_Duty\*60-3080;

//else if(LPWM\_Duty<280) PM25\_Value =LPWM\_Duty\*100-14000;

else{PM25\_Value=10000;}

if(PM25\_Value>10000) PM25\_Value=10000;

PM25\_Value=PM25\_Value/10;

if(CAN\_Timer){

CAN\_send\_flag++;

if(CAN\_send\_flag==1){

num++;

if(num>29)num=0;

CAN\_Transmit(LPWM\_Duty,PM25\_Value);

PM25\_Value\_old=PM25\_Value;

LPWM\_Duty\_old=LPWM\_Duty;

}

else if (CAN\_send\_flag==2)

{

CAN\_Transmit(LPWM\_Duty\_old,PM25\_Value\_old);

CAN\_send\_flag=0;

}

// while(TXB0CONbits.TXBIF!=1);

CAN\_Timer=0;

}

}

}