#include<lpc17xx.h>

#include<stdio.h>

#include<stdlib.h>

#define RS\_CTRL 1<<27

#define EN\_CTRL 1<<28

#define DT\_CTRL 0XF<<23

unsigned long int temp1=0,temp2=0,i,j;

unsigned char flag1=0,flag2=0,flag3=0;

unsigned int f,a,sw=0,count=0;

double x;

unsigned int num[] = {1,2,3,4,5,6};

char dispval[100];

void lcd\_write(void);

void port\_write(void);

void delay\_lcd(unsigned int);

void delay(void);

void lcd\_init(void);

void pwm\_init(void);

void PWM\_IRQHandler(void);

unsigned long int cmd,data,flag=0,init\_command[]={0x30,0x30,0x30,0x20,0x28,0x0C,0x06,0x01,0x80};

int main(void)

{

SystemInit();

SystemCoreClockUpdate();

LPC\_GPIO0->FIODIR = RS\_CTRL|EN\_CTRL|DT\_CTRL;//input lines enabled

LPC\_PINCON->PINSEL3=3<<28;//assigning port pin p1.30 function 3 for ADC 0.4

LPC\_ADC->ADCR=(1<<4)|(1<<21)|(1<<24);//enabling ADCR

lcd\_init();//lcd initialization function

pwm\_init();//pulse width modulator initalization function

while(1)

{

int j1;

flag1=0; //indicates command is being written

temp1 = 0x80;// clear lcd screen

lcd\_write(); //

flag1=1; //indicates data is being written

LPC\_ADC->ADCR = 0x01200010;

for(j1=0;j1<2000;j1++);//delay

if((LPC\_ADC->ADGDR & 1<<31))//checking if done bit for ADC0.4 is high

{ flag3=1;

x = (LPC\_ADC->ADGDR & 0xFFF0) >> 4; //digital value -->analog value

x = (float) x \* 0.0008 \* 2.909; //analog input-->radio frequency (mapping)

x+=90.0; //getting in range of 90 to 99

sprintf(dispval,"Radio: %2.1fMHz",x);//to string conversion and store in buffer for display

i=0;

while(dispval[i]!='\0')

{ temp1=dispval[i];//write date on to LCD

lcd\_write();

i+=1;

}

}

delay();

}

}

void lcd\_init(void){

flag1=0;//command being written

for(i=0;i<9;i++){

temp1=init\_command[i];//full basic initializations onto LCD

lcd\_write();

}

flag1=1; //data being written

}

void lcd\_write(void){

flag2 = (flag1==1) ? 0:((temp1==0x30)||(temp1==0x20)) ? 1 : 0; //flag2 is high for 0x30 and 0x20

temp2 = temp1 & 0xF0;//extracting most significant 4 bits

temp2 <<= 19;//assigning to P0.23-26

port\_write();

if(!flag2){

temp2 = temp1 & 0x0F;//extracting least significant 4 bits

temp2 <<= 23;//assigning to P0.23-26

port\_write();

}

}

void port\_write(){

LPC\_GPIO0->FIOPIN = 0; //clearing all i/o lines

LPC\_GPIO0->FIOPIN = temp2;

if(flag1==0)

LPC\_GPIO0->FIOCLR = RS\_CTRL; //write commands

else LPC\_GPIO0->FIOSET = RS\_CTRL;//write data

LPC\_GPIO0->FIOSET = EN\_CTRL;// creating a -ve edge

delay\_lcd(25);

LPC\_GPIO0->FIOCLR = EN\_CTRL;

delay\_lcd(5000);

}

void delay\_lcd(unsigned int r1){

unsigned int r;

for(r=0;r<r1;r++);

}

void delay(void){

for(i=0;i<30000;i++);

}

void pwm\_init(void)

{ LPC\_PINCON->PINSEL3&= ~(0x0000C000);

LPC\_PINCON->PINSEL3 |= 0x00008000; //P1.23 is assigned function 2

LPC\_PWM1->PR=0;//count frequency of FPLCK

LPC\_PWM1->MCR=0x03;//reset and interrupt PWMMR0

LPC\_PWM1->PCR=0x00001000;//select PWM1 single edge

LPC\_PWM1->MR0=30000;

LPC\_PWM1->LER=0xFF;//shadow copy register enabled

LPC\_PWM1->TCR=0x02;//reset counter

LPC\_PWM1->TCR=0x09;//enable PWM and counter

NVIC\_EnableIRQ(PWM1\_IRQn);//global enabling

return;

}

void PWM1\_IRQHandler(void)

{ LPC\_PWM1->IR=0xFF; //clear interrupts

f=x\*10; //multiplied by 10 to remove decimal point

f=f&0xFFF;

if(f>902 | f<925 & f>929 | f<933 & f>937 | f<981 & f>985 | f<993 ){

LPC\_PWM1->MR4=0;//switch off PWM led

LPC\_PWM1->LER=0xFF;//enable shadow copy register

}

if(f==902||f==925|f==929|f==933|f==937|f==981|f==985|f==993){

LPC\_PWM1->MR4=9900;//intensity of one third

LPC\_PWM1->LER=0xFF;

}

else if(f==901|f==926|f==928|f==934|f==936|f==982|f==984){

LPC\_PWM1->MR4=19800;//intensity of two third

LPC\_PWM1->LER=0xFF;

}

else if(f==900|f==927|f==935|f==983|f==995){

LPC\_PWM1->MR4=29700;//intensity max at radio stations and start and end of spectrum

LPC\_PWM1->LER=0xFF;

}

}