 EMBEDDED SYSTEM PROJECT

Project report - Embedded Systems Design Lab [ICT 3111] Write an Embedded C program to simulate full fledged calculator by interfacing

4 X 4 matrix keyboard and LCD to 32 bit ARM microcontroller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Team Members | Reg no | Course | Semester | Section |
| Ishan jain | 140911300 | IT | 5 TH sem | B (b2-batch) |
| Devansh ojha | 140911314 | IT | 5 TH sem | B (b2-batch) |
| Anand | 140911308 | IT | 5 TH sem | B (b2-batch) |

The Calculator made can perform arthimetic ,logrithmic and trignometric operation. Intially the first digit is taken as input this is followed by single click of the (‘=’) button to siginfy that operator can now be inserted .If single operand function is to be carried out then a double click of (‘=’) button gives required result otherwise second operand is entered .After the required input is given the result can now be calculated .This is calculation carried out by the double click of the (‘=’) button .

ALGORTHIM :-

Initialize LCD

Function to send command

Function to send data

Function to write data

Main function()

{

Check for button click on rows

Scan cols to find the clicked button

Increment the clickcount

If (clickcount==1){

Compare ASCII values of digit(input)

Increment the clickcount

If(clickcount==2)

{

take operator as input (‘+’,’-’,’\*’,’/….)

increment the clickcount

}

If(key press=”==”)

{

One operand operation performs

}

If(clickcount==3)

{ take second operator as input

Increment the count

}

If(count==4)

{

Calculate function called

Display the output

Increment the count

}

If(count==5)

{

Take the result as the first input for new operation

Display on LCD

}

Decrement count and redo loop

}

Delay function()

Function to clear display()

Calculate function(){

Arithmetic, trignometric and logarithmic operations

Convert operation result to char for lcd display

}

WORKING CODE :-

#include <lpc17xx.h>

#include <stdio.h>

#include <math.h>

#define RS\_CTRL 0x08000000

#define EN\_CTRL 0x10000000

#define DT\_CTRL 0x07800000

void scan(void);

void lcd\_init(void);

void wr\_cn(void);

void clr\_disp(void);

void delay\_lcd(unsigned int);

void lcd\_com(void);

void wr\_dn(void);

void lcd\_data(void);

void clear\_ports(void);

void lcd\_puts(char \*);

void calcu();

unsigned char col,row,flag;

char a[100];

unsigned long int i,var1,temp,temp3,x,c=0,op,d=0; // for keyboard interfacing

unsigned long int temp1=0, temp2=0,count=0,first\_dig,second\_dig,op\_flag=0,k;

char tempz[5],num1[10]={0},num2[10]={0},result[10],b[100];

unsigned char seven\_code[4][4]={{0x30,0x31,0x32,0x33}, {0x34,0x35,0x36,0x37}, {0x38,0x39,0x2e,0x2e}, {0x2e,0x2e,0xFE,0xFF}};

unsigned char op\_code[4][4]={{0x2b,0x2d,0x2a,0x2f},{'%','t','c','s'},{'l','L','f','i'},{'p','S','C',0x3d}};

int main(void)

{

SystemInit();

SystemCoreClockUpdate();

LPC\_GPIO2->FIODIR=0x3C00;

lcd\_init();

delay\_lcd(3200);

temp1 = 0x80;

lcd\_com();

delay\_lcd(800);

while(1)

{

for(row = 0; row<4; row++)

{

LPC\_GPIO2->FIOPIN = 0x1<<(row+10);

flag= 0;

scan();

if(flag == 1)

{ // tempz[0]=count+0x30;

//lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

count++;

//if(!x)

tempz[0]=seven\_code[row][col];

if(count==1)

{

if((tempz[0]>=0x30 && tempz[0]<=0x39) ||(tempz[0]==(0x2e))) //checking if first input is digit

{

if(tempz[0]!=0x2e)

{

num1[c++]=tempz[0]-0x30;

count--;

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

}

else

{

num1[c++]='.';

count--;

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

}

}

// tempz[0]=count+0x30;

// lcd\_puts(tempz);

// for(i=0;i<50000;i++);

// clear\_ports();

}

else if(count==2)

{

tempz[0]=op\_code[row][col] ;

op=tempz[0];

if(op!=0x2b&&op!=0x2d&&op!=0x2a&&op!=0x2f&&op!='p'&&op!='%')

{

op\_flag=1;

count=3;

}

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

//tempz[0]=count+0x30;

// lcd\_puts(tempz);

// for(i=0;i<50000;i++);

// clear\_ports();

}

else if(count==3) //checking if third input is digit

{

if((tempz[0]>=0x30 && tempz[0]<=0x39) ||(tempz[0]==(0x2e)))

{

if(tempz[0]!=0x2e)

{

num2[d++]=tempz[0]-0x30;

count--;

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

continue;

}

else

{

num2[d++]='.';

count--;

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

}

}

}

else if(count==4)

{

tempz[0]=op\_code[row][col];

if(tempz[0]==0x3d)

{

calcu();

temp1 = 0xc0;

lcd\_com();

delay\_lcd(800);

lcd\_puts(tempz);

lcd\_puts(result);

}

}

else if(count==5)

{

tempz[0]=seven\_code[row][col];

if(tempz[0]==0xFE)

{

clr\_disp();

for(i=0;i<10;i++)

{

num1[i]='0';

}

for(c=0;c<k;c++)

{

if(result[c]!='.')

num1[c]=(result[c]-0x30);

else

num1[c]='.';

}

c=k;

lcd\_puts(result);

for(i=0;i<50000;i++);

clear\_ports();

for(d=0;d<10;d++)

{

num2[d]=0x30-0x30;

result[d]=0x30-0x30;

}

op\_flag=0;

d=0;

count=1;

}

else

{

clr\_disp();

for(d=0;d<10;d++)

{ num1[d]=0x30-0x30;

num2[d]=0x30-0x30;

result[d]=0x30-0x30;

}

for(i=0;i<50000;i++);

clear\_ports();

op\_flag=0;

c=0;

d=0;

count=0;

}

}

}

}

}

}

void lcd\_init()

{

LPC\_PINCON->PINSEL1 &= 0xFC003FFF;

LPC\_GPIO0->FIODIR |= DT\_CTRL;

LPC\_GPIO0->FIODIR |= RS\_CTRL;

LPC\_GPIO0->FIODIR |= EN\_CTRL;

clear\_ports();

delay\_lcd(3200);

temp2 = (0x30<<19);

wr\_cn();

delay\_lcd(30000);

temp2 = (0x30<<19);

wr\_cn();

delay\_lcd(30000);

temp2 = (0x30<<19);

wr\_cn();

delay\_lcd(30000);

temp2 = (0x20<<19);

wr\_cn();

delay\_lcd(30000);

temp1 = 0x28;

lcd\_com();

delay\_lcd(30000);

temp1 = 0x0C;

lcd\_com();

delay\_lcd(800);

temp1 = 0x06;

lcd\_com();

delay\_lcd(800);

temp1 = 0x01;

lcd\_com();

delay\_lcd(10000);

temp1 = 0x80;

lcd\_com();

delay\_lcd(800);

return;

}

void scan(void)

{

temp3=LPC\_GPIO1->FIOPIN;

temp3=temp3&0x07800000;

if(temp3!=0x0)

{

flag=1;

if(temp3==0x00800000)

col=0;

else if(temp3==0x01000000)

col=1;

else if(temp3==0x02000000)

col=2;

else if(temp3==0x04000000)

col=3;

}

}

void lcd\_com(void)

{

temp2 = temp1 & 0xf0;

temp2 = temp2 << 19;

wr\_cn();

temp2 = temp1 & 0x0f;

temp2 = temp2 << 23;

wr\_cn();

delay\_lcd(1000);

return;

}

void wr\_cn(void)

{

clear\_ports();

LPC\_GPIO0->FIOPIN = temp2;

LPC\_GPIO0->FIOCLR = RS\_CTRL;

LPC\_GPIO0->FIOSET = EN\_CTRL;

delay\_lcd(25);

LPC\_GPIO0->FIOCLR = EN\_CTRL;

return;

}

void lcd\_data(void)

{

temp2 = temp1 & 0xf0;

temp2 = temp2 << 19;

wr\_dn();

temp2= temp1 & 0x0f;

temp2= temp2 << 23;

wr\_dn();

delay\_lcd(1000);

return;

}

void wr\_dn(void)

{

clear\_ports();

LPC\_GPIO0->FIOPIN = temp2;

LPC\_GPIO0->FIOSET = RS\_CTRL;

LPC\_GPIO0->FIOSET = EN\_CTRL;

delay\_lcd(25);

LPC\_GPIO0->FIOCLR = EN\_CTRL;

return;

}

void delay\_lcd(unsigned int r1)

{

unsigned int r;

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

;

}

return;

}

void clr\_disp(void)

{

temp1 = 0x01;

lcd\_com();

delay\_lcd(10000);

return;

}

void clear\_ports(void)

{

LPC\_GPIO0->FIOCLR = DT\_CTRL;

LPC\_GPIO0->FIOCLR = RS\_CTRL;

LPC\_GPIO0->FIOCLR = EN\_CTRL;

return;

}

void lcd\_puts(char \*buf1)

{

unsigned int i=0;

while(buf1[i]!='\0')

{

temp1 = buf1[i];

lcd\_data();

i++;

}

return;

}

void calcu()

{

float res1=0,res2=0,resf,temp;

int l;

for(k=0;k<c;k++)

{

if(num1[k]!='.')

res1=res1\*10.0+(num1[k]);

else

{

for(l=1;l+k<c;l++)

{

temp=(float)((num1[k+l]))/(10.0\*l);

res1=res1+temp;

}

k=c;

}

}

sprintf(b,"%.3f",res1) ;

lcd\_puts(b);

for(i=0;i<50000;i++);

clear\_ports();

if(op\_flag!=1)

{

for(k=0;k<d;k++)

{

if(num2[k]!='.')

res2=res2\*10.0+num2[k];

else

{

for(l=1;l+k<d;l++)

{

temp=((float)((num2[k+l])))/(10.0\*l);

res2=res2+temp;

}

k=d;

}

}

}

k=0;

if(op==0x2b)

{

resf=res1+res2;

}

else if(op==0x2d)

{

resf=res1-res2;

}

else if(op==0x2a)

{

resf=res1\*res2;

}

else if(op==0x2f)

{

resf=res1/res2;

}

else if(op=='s')

{

res1=res1\*3.14/180;

resf=sin(res1);

}

else if(op=='c')

{ res1=res1\*3.14/180;

resf=cos(res1);

}

else if(op=='t')

{ res1=res1\*3.14/180;

resf=tan(res1);

}

else if(op=='p')

{

resf=pow(res1,res2);

}

else if(op=='f')

{ int p;

resf=1;

for(p=1;p<=res1;p++)

{

resf=resf\*p;

}

}

else if(op=='%')

{

resf=(int)res1%(int)res2;

}

else if(op=='l')

{

resf=log10(res1);

}

else if(op=='L')

{

resf=log(res1);

}

else if(op=='C')

{

resf=pow(res1,3);

}

else if(op=='S')

{

resf=pow(res1,2);

}

else if(op=='i')

{

resf=pow(res1,-1);

}

op\_flag=0;

sprintf(result,"%.3f",resf) ;

k=sizeof(result)/sizeof(result[0]);

/\*while(resf!=0)

{

result[k] = (resf%10) + 0x30; //ascii conversion saath me

tempz[0]=result[k];

lcd\_puts(tempz);

for(i=0;i<50000;i++);

clear\_ports();

resf = resf/10 ;

k++;

} }