**ADC PROGRAM:**

#include <LPC17xx.H>

#include "GLCD.h"

#include "Serial.h"

#define \_\_FI 1

unsigned int Channel\_No;

unsigned char result,Digital;

/\* To Display the Channel number in the ASCII format \*/

void Decimal\_Adjust(unsigned long Channel\_No)

{

switch(Channel\_No)

{

case 0xa: SER\_PutChar('0');

GLCD\_DisplayChar(6,12, \_\_FI,'0');

break;

case 0xb: SER\_PutChar('1');

GLCD\_DisplayChar(6,12, \_\_FI,'1');

break;

case 0xc: SER\_PutChar('2');

GLCD\_DisplayChar(6,12, \_\_FI,'2');

break;

case 0xd: SER\_PutChar('3');

GLCD\_DisplayChar(6,12, \_\_FI,'3');

break;

case 0xe: SER\_PutChar('4');

GLCD\_DisplayChar(6,12, \_\_FI,'4');

break;

case 0xf: SER\_PutChar('5');

GLCD\_DisplayChar(6,12, \_\_FI,'5');

break;

}

}

/\* Routine to convert the Analog Data \*/

void Convert(void)

{

unsigned long temp = 0,i = 0,temp1 = 0,res;

temp1 = (Channel\_No << 15);

LPC\_GPIO0->FIOPIN = (LPC\_GPIO0->FIOPIN & 0xFF807FFF) | temp1;

LPC\_GPIO0->FIOPIN = (LPC\_GPIO0->FIOPIN & 0xFF807FFF) | ((1 << 20) | temp1); /\* Assert the START signal \*/

temp1 = 0;

LPC\_GPIO0->FIOPIN = ((LPC\_GPIO0->FIOPIN & 0xFF807FFF) | (Channel\_No | temp1)); /\* Pull down the START signal \*/

for(i = 0; i <= 10000; i ++);

temp = LPC\_GPIO0->FIOPIN & 0x00800000;

temp >>= 23;

temp = temp & 0x01; /\* Wait for the End of Conversion \*/

while(temp != 0x1)

{

temp = LPC\_GPIO0->FIOPIN & 0x00800000;

temp >>= 23;

temp = temp & 0x01;

}

LPC\_GPIO0->FIOPIN = ((LPC\_GPIO0->FIOPIN & 0x007F8000) | ((1<<21) | Channel\_No));

for(i = 0; i <= 10000; i ++);

temp = LPC\_GPIO1->FIOPIN & 0x07F80000;

temp >>= 19;

/\* Read the Digital Value \*/

temp1 = 0;

LPC\_GPIO0->FIOPIN = ((LPC\_GPIO0->FIOPIN & 0x007F8000) | (Channel\_No | temp1));

res = LPC\_GPIO1->FIOPIN & 0x07F80000;

res >>= 19;

result = (unsigned char)res;

}

void Display(void)

{

unsigned char temp = 0;

SER\_SendString("\nChannel Number : ");

GLCD\_DisplayString(6, 0, \_\_FI,"Channel No=");

if(Channel\_No <= 9) /\* Check whether the channel no is < than 9 \*/

{

SER\_PutChar('0');

SER\_SendHex(Channel\_No);

GLCD\_DisplayChar(6,11, \_\_FI, '0');

GLCD\_DisplayChar(6,12, \_\_FI, '0');

GLCD\_DisplayChar(6,12, \_\_FI,Channel\_No+ 0x30);

}

else if((Channel\_No >= 0x0a) && (Channel\_No <= 0x0f))

{

SER\_PutChar('1');

GLCD\_DisplayChar(6,11, \_\_FI, '1');

}

/\* If > 9, send the ASCII values depending \*/

Decimal\_Adjust(Channel\_No); /\* upon the value of the channel number \*/

SER\_SendString("\tDigital Value : ");

GLCD\_DisplayString(7, 0, \_\_FI,"Digital Value=0x");

temp = (result & 0xf0) >> 4;

SER\_SendHex(temp);

temp = (result & 0xf0) >> 4;

if( temp <= 9) /\* Displays the corresponding ASCII value of 2nd digit \*/

temp = temp + 0x30;

else

temp = temp + 0x37;

GLCD\_DisplayChar(7,16, \_\_FI,temp);

SER\_SendHex(result & 0x0f);

temp=(result & 0x0f);

if( temp <= 9) /\* Displays the corresponding ASCII value of 2nd digit \*/

temp = temp + 0x30;

else

temp = temp + 0x37;

GLCD\_DisplayChar(7,17, \_\_FI,temp);

SER\_SendString("\n\r");

}

void Disp\_Update(void) /\* Routine to update the Channel numbers \*/

{

char c;

c = SER\_GetChar(); /\* Read a character from UART \*/

if(c == ',')

{

if(Channel\_No == 15) /\* If typed ',',increment the channel number \*/

Channel\_No = 0;

else

Channel\_No += 1;

}

else if(c == '-') /\* If typed '-',decrement the channel number \*/

{

if(Channel\_No == 0)

Channel\_No = 15;

else

Channel\_No -= 1;

}

}

main()

{

LPC\_SC->PCONP |= (1 << 15); /\* enable power to GPIO \*/

LPC\_GPIO0->FIODIR |= 0x007F8000; /\* PortA output PortC input \*/

LPC\_GPIO1->FIODIR &= ~(0x07F80000); /\* PortB input \*/

SER\_Init();

#ifdef \_\_USE\_LCD

GLCD\_Init(); /\* Initialize graphical LCD \*/

GLCD\_Clear(White); /\* Clear graphical LCD display \*/

GLCD\_SetBackColor(Blue);

GLCD\_SetTextColor(White);

GLCD\_DisplayString(0, 0, \_\_FI, " ESA ");

GLCD\_DisplayString(1, 0, \_\_FI, " Bangalore ");

GLCD\_DisplayString(2, 0, \_\_FI, " www.esaindia.com ");

GLCD\_SetBackColor(White);

GLCD\_SetTextColor(Blue);

GLCD\_DisplayString(5, 0, \_\_FI, " 16 Channel ADC");

#endif

Channel\_No = 0;

while(1)

{

Convert(); /\* Convert the analog to digital value \*/

Display(); /\* Display the correspoding Channel number \*/

/\* and it's equivalent digital value \*/

Disp\_Update(); /\* Update the display and channel number \*/

}

}

**ADC INTERFACING:**



