

Name:-
Roll_no:-
Subject:-
Class:-

EXPERIMENT NO:-10

ADC Program

```
#include <p18f4550.h>

#include<stdio.h>

#define LCD_EN LATAbits.LA1
#define LCD_RS LATAbits.LA0
#define LCDPORT LATB

void lcd_delay(unsigned int time)
{ unsigned int i , j ;
    for(i = 0; i < time; i++)
    {      for(j=0;j<50;j++);
    }
}

void SendInstruction(unsigned char command)
{
    LCD_RS = 0;          // RS low : Instruction

    LCDPORT = command;

    LCD_EN = 1;          // EN High

    lcd_delay(10);

    LCD_EN = 0;          // EN Low; command sampled at EN falling edge

    lcd_delay(10);
}

void SendData(unsigned char lcddata)
```

```

{
    LCD_RS = 1;          // RS HIGH : DATA

    LCDPORT = lcddata;

    LCD_EN = 1;          // EN High

    lcd_delay(10);

    LCD_EN = 0;          // EN Low; data sampled at EN falling edge

    lcd_delay(10);
}

void InitLCD(void)
{
    ADCON1 = 0x0F;

    TRISB = 0x00; //set data port as output

    TRISAbits.RA0 = 0; //RS pin

    TRISAbits.RA1 = 0; // EN pin

    SendInstruction(0x38); //8 bit mode, 2 line,5x7 dots

    SendInstruction(0x06); // entry mode

    SendInstruction(0x0C); //Display ON cursor OFF

    SendInstruction(0x01); //Clear display

    SendInstruction(0x80); //set address to 0
}

void ADCInit(void)
{
    TRISEbits.RE1 = 1;          //ADC channel 6 input

    TRISEbits.RE2 = 1;          //ADC channel 7 input

    ADCON1 = 0b00000111;        //Ref voltages Vdd & Vss; AN0 - AN7 channels Analog

```

```

    ADCON2 = 0b10101110;          //Right justified; Acquisition time 4T; Conversion clock Fosc/64
}

unsigned short Read_ADC(unsigned char Ch)
{
    ADCON0 = 0b00000001 | (Ch<<2); //ADC on; Select channel;
    GODONE = 1;           //Start Conversion
    while(GO_DONE == 1 ); //Wait till A/D conversion is complete
    return ADRES;         //Return ADC result
}

void DisplayResult(unsigned short ADCVal)
{
    unsigned char i,text[16];
    unsigned short tempv;
    tempv = ADCVal;
    SendInstruction(0x80);          //set to 1st line
    for(i=0;i<10;i++)              //Display the 10 bit ADC result on LCD
    {
        if(tempv & 0x200)
        {
            SendData('1');
        }
        else
        {
            SendData('0');
        }
    }
}

```

```

    tempv = tempv<<1;
}

ADCVal = (5500/1024)*ADCVal;          //Convert binary data to mV; 1 bit <=> (5500/1024)mV

sprintf(text,"ADC value=%4dmv",ADCVal); //Convert integer data to string

SendInstruction(0xC0);                //set to 2nd line

for(i=0;i<16;i++)                    //Display string on LCD
{
    SendData(text[i]);
}
}

void main()
{
    unsigned short Ch_result;

    TRISB = 0x00;                    //PORTB connected to LCD is output

    ADCInit();

    InitLCD();

    while(1)
    {
        Ch_result = Read_ADC(7);

        DisplayResult(Ch_result);

        lcd_delay(1000);
    }
}

```