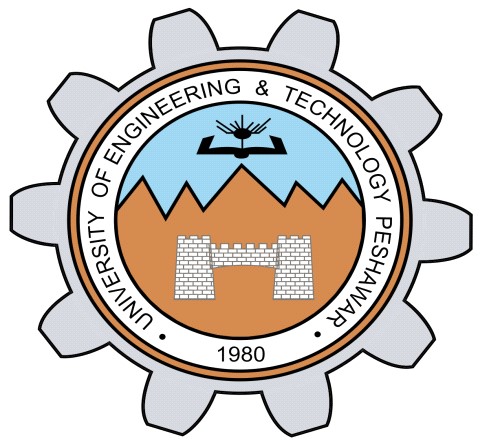
**Task no 7**

****

**Microprocessor Based System Design**

**Spring 2022**

**Submitted by**

**Name Registration no**

**Muhammad Ali 19pwse1801**

**Submitted to: Dr Bilal Habib**

**Data: 20/6/2022**

**Department of Computer System Engineering**

**Code: -**

#include <reg51.h>

#define input P1;

double newtemp;

char result\_char;

int x=0;

//LCD pins

sbit rs = P3^7; //register select pin

sbit e = P3^6; //enable pin

//ADC pins

sbit rd=P3^4; //defines rd pin of ADC use for reading purposes

sbit wr=P3^5; // define wr pin of ADC use for writing purposes

sbit intr=P3^2; //defines intr pin use for sending interrupts

sbit buz=P3^0; //used for buzzer on off

//functions

void reg\_no\_print();

void delay(unsigned int time); //Function to provide time delay in msec.

double adc(); // Function to read the values from ADC and send to controller.

void writecmd(unsigned char item); //Function to send commands to LCD see command tables in LCD Link

void writedata(double item); //Function to send data to LCD

void disp\_temp(double num); //displays number on LCD

void read(); // Displays "READING" while controller reads from ADC

void timer\_init();

void timer0(void) interrupt 1{

//Interupt service routine

x++;

if(x==1){

buz=1;

}

else if(x==7){

buz=0;

}

else if (x== 10)

{

x=0;

TR0=0; //timer stop

}

TH0=0xFE; //for delay

TL0=0x0C;

}

void main()

{

P0=0x00; //intialize port 0

timer\_init(); //initialize the timer

while(1)

{ //enters in the permanent loop

buz=1;

reg\_no\_print(); //print reg\_no

read();

newtemp=adc(); //reads first value from ADC

disp\_temp(newtemp); //show temperature

if(newtemp>25.0 || newtemp<10.0)

{

TR0=1; //timer run

buz=0;

while(TR0==1);

buz=1;

}

else

{

buz=1;

TR0=0;

}

delay(300);

}

}

void delay(unsigned int time) //Function to provide time delay in msec.

{

int i,j ;

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

for(j=0;j<1275;j++);

}

double adc() // Function to read the values from ADC and send to controller.

{

double temp;

rd=1; //high to low to read from adc

wr=0; //low to high to write on adc

delay(1);

wr=1;

while(intr==1); //interrupt is low active

rd=0;

temp=input; //getting the converted digital value

delay(3);

return temp;

}

void writecmd(unsigned char item)

{

P2 = item; //Data transfer to P2

rs= 0; // This is for command

e=1;

delay(1);

e=0;

delay(1);

}

void writedata(double item) //Function to send data to LCD

{

P2 = item;

rs= 1; // This is for data

e=1;

delay(1);

e=0;

}

void disp\_temp(double num) //displays number on LCD

{

unsigned char UnitDigit = 0;

unsigned char TenthDigit = 0;

unsigned char HundDigit = 0;

unsigned char decimal=0;

int point;

point=num\*10;

HundDigit=(num/100);

if( HundDigit != 0)

writedata(HundDigit+0x30);

TenthDigit = num - HundDigit\*100;

TenthDigit = TenthDigit/10;

if (HundDigit==0 && TenthDigit==0){}

else

writedata(TenthDigit+0x30);

UnitDigit = num - HundDigit\*100;

UnitDigit = UnitDigit - TenthDigit\*10;

writedata(UnitDigit+0x30);

writedata('.');

decimal=(point%10);

writedata (decimal+0x30);

writedata(' '); writedata('C');

}

void timer\_init(){

TMOD=0x01; //16 bit timer mode of timer0

TH0=0xFE; //for delay

TL0=0x0C; //for 500microsec delay

IE=0x82; //Only the timer0 interupts is acknowbuzged

}

void read(){ // Displays "READING" while controller reads from ADC

writecmd(0x95);

writedata('R');writedata('E');writedata('A');writedata('D');writedata('I');writedata('N');writedata('G');writedata(' ');

}

void reg\_no\_print(){

writecmd(0x0E); //turn display ON for cursor blinking

writecmd(0x01); //clear screen

writecmd(0x06); //increment cursor

writedata('T');

writedata('E');

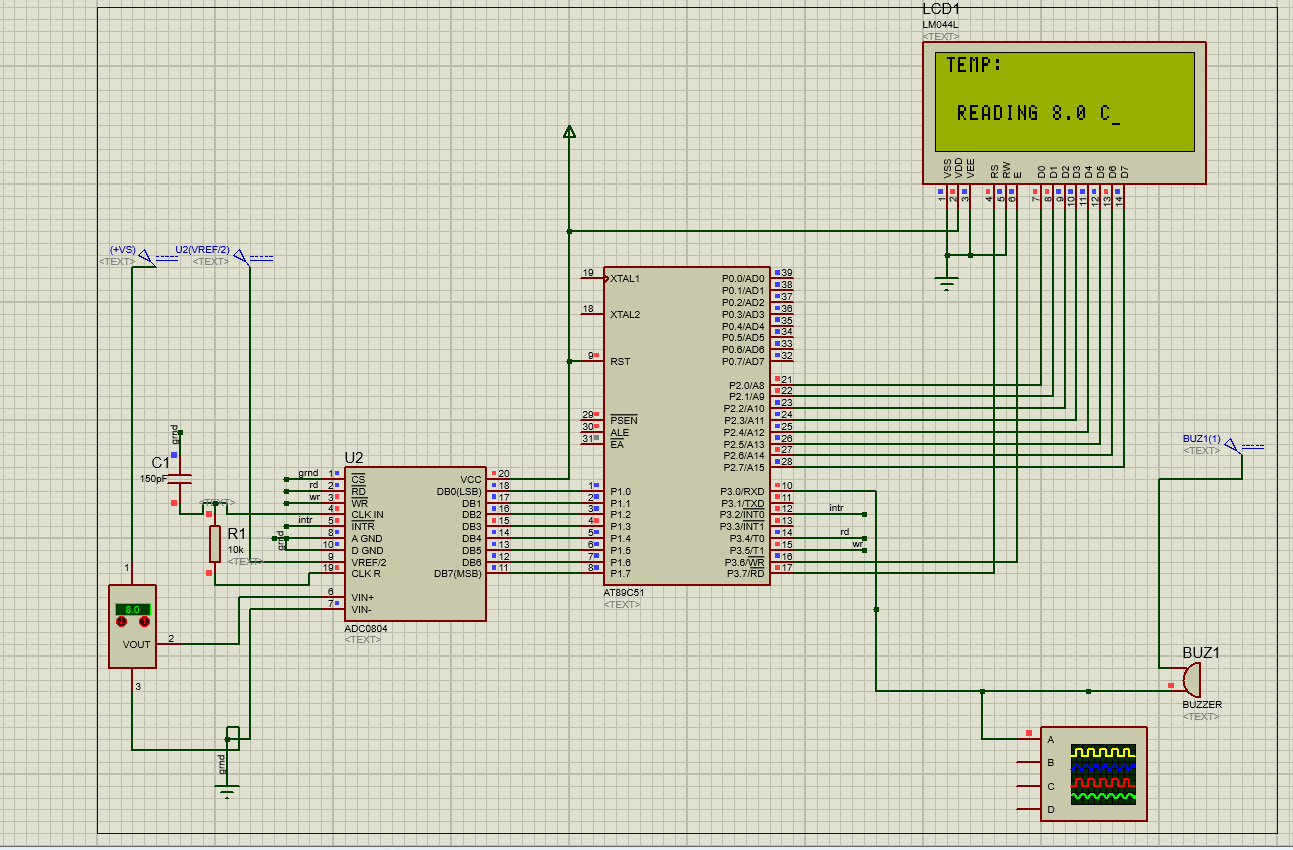
writedata('M');

writedata('P');

writedata(':');

}

**Output result on LCD: -**

****