REPORT ON INTERFACING A TEMPERATURE SENSOR TO LPC1768 AND DISPLAYING IT ON A 7 SEGMENT DISPLAY.

Batch No	Team No	Name	Reg No	Section	Branch
A2	7	Shrey Deshmukh	190953242	A	CCE
A2	7	K. Sai Vaibhav	190953246	A	CCE
A2	7	Nikhil Vinnakota	190953236	A	CCE

Problem Statement:

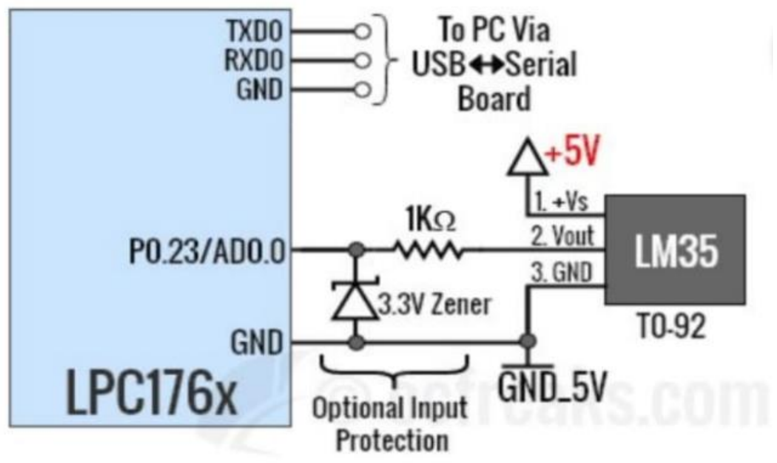
Write a program to interface a temperature sensor to LPC1768 and display the temperature on 7-segment display

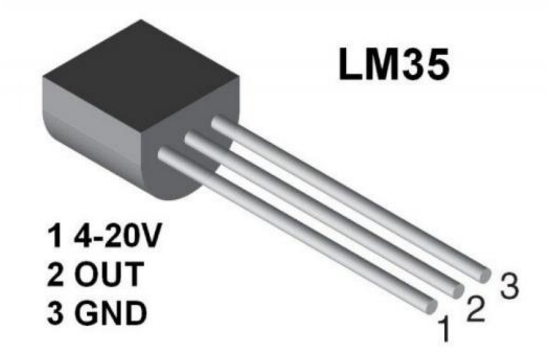
Hardware Components Used:

Components Name	Quantity
ALS-SDA-ARMCTXM3-01	1
Power Supply (+5V)	1
Cross Cable	2
LM35 Temperature Sensor	1
Jumper Cables (Female-Female)	3
USB port in the computer and PC for downloading the software	1

Circuit Diagram:

<u>CIRCUIT DIAGRAM</u>





Code:

```
#include<LPC17xx.h>
float x,y,temp;
unsigned long a,b,temp2,r1,i;
unsigned char
seven_seg[16]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F,0x77,0x7C,0x39,0X5
E,0X79,0X71};//store hex values of each digit from 0 to F
unsigned char digits[]={0,0,0,0};//will store the digits to be displayed
unsigned int dig_sel[]={0<<23,1<<23,2<<23,3<<23};//stores value for selection of 7 segment
void display(void);
void timer_init(void);
int main(void)
{
SystemInit();
SystemCoreClockUpdate();
timer_init();//delay
LPC_PINCON->PINSEL0&=0XFF0000FF;//P0.4 to P0.11 as data lines to 7 segment
LPC_PINCON->PINSEL1|=1<<16;//P0.24 as ADC input (ADD0.1)
LPC_PINCON->PINSEL3|=0x00<<14;//configuring P1.23 to P1.26 as decoder lines
LPC_GPIO0->FIODIR=0XFF<<4;//Data lines are output lines
LPC_GPIO1->FIODIR=0XF<<23;//Decoder lines are also output lines
LPC_SC->PCONP|=1<<12;//Power to the ADC by enabling the 12th pin of PCONP (Power
Control for Peripheral)
LPC_ADC->ADCR=(1<<1|1<<16|1<<21);//Enable channel 1 (ADD0.1) in burst mode and
enable power down (PDN)
NVIC_EnableIRQ(ADC_IRQn);//Enable the NVIC
LPC_ADC->ADINTEN=(1<<1);//Enable interrupt on channel 1 (ADD0.1)
while(1);
}
void ADC_IRQHandler()
{
a=(LPC_ADC->ADSTAT) & 1<<1;//Check if channel 1's DONE bit is high
if(a)
{
	b=(LPC_ADC->ADDR1);//if DONE bit high, read the data in ADDR1 register (this also
clears the DONE bit)
}
temp2=LPC_ADC->ADGDR;//Read the data in ADGDR register to clear the DONE bit of
ADGDR
b= b & 0xFFFF;//The data is present on 4th to 15th bit
b>>=4;//to get the digital value in lower bit positions
y=((float)b*(330.0/4096));//Conversion of result in the register to temperature in °C as 10mV of
input represents 1°C
digits[3]=((int)y/10);//MSB of the calculated temp should be displayed on the 3rd 7 segment
digits[2]=((int)(y)%10);//LSB of the calculated temp should be displayed on the 2nd 7 segment
digits[1]=((int)(y*10)%10);//decimal digit of the calculated temp should be displayed on the 1st
7 segment
while(LPC_TIM0->EMR & 0X01)
{
        display();//display same value for the next 3s
}
LPC_TIM0->EMR=0X011;//reset the EMR value as in timer_init()
}
void display(void)
{
int x=0,i;
/* display 4 segments values one by one */
for(x=0;x<4;x++)
{
LPC_GPIO1->FIOPIN=dig_sel[x];//enable the decoder lines according to the x value
if(x==2)
{
r1=(seven_seg[digits[x]] |0x80);//third segment should have a decimal point(since room-temp is
2 digit)
}
else
if(x==0)
{
r1=0x39;//0x39 is the 7 segment value for "C" so this is to display °C in the 0th 7 segment
}
else
{
r1=(seven_seg[digits[x]]);//for other segments get the 7 segment values of the digits from
seven_seg[]
}
LPC_GPIO0->FIOPIN=r1<<4;//Put the 7 segment value into data lines(P0.4 to P0.11)
for(i=0; i<500;i++);//Wait for some time (small delay)
LPC_GPIO0->FIOPIN=00<<4;//clear the data lines
}
}
void timer_init()
{
LPC_TIM0->CTCR=0X00;//timer mode
LPC_TIM0->TCR=0X02;//reset TC and PC
LPC_TIM0->MCR=0X02;//reset the TC and PC on match
LPC_TIM0->PR=0X02;//TC will increment for every 3 PCLK
LPC_TIM0->MR0=2999999;//calculated using formula "MR=(PCLK*DELAY)/PR+1" where
the delay is 3s
LPC_TIM0->EMR=0X011;//initially EMC0 is HIGH when there is a match it is configured to
become LOW
LPC_TIM0->TCR=0X01;//start the timer
}
```

Result:

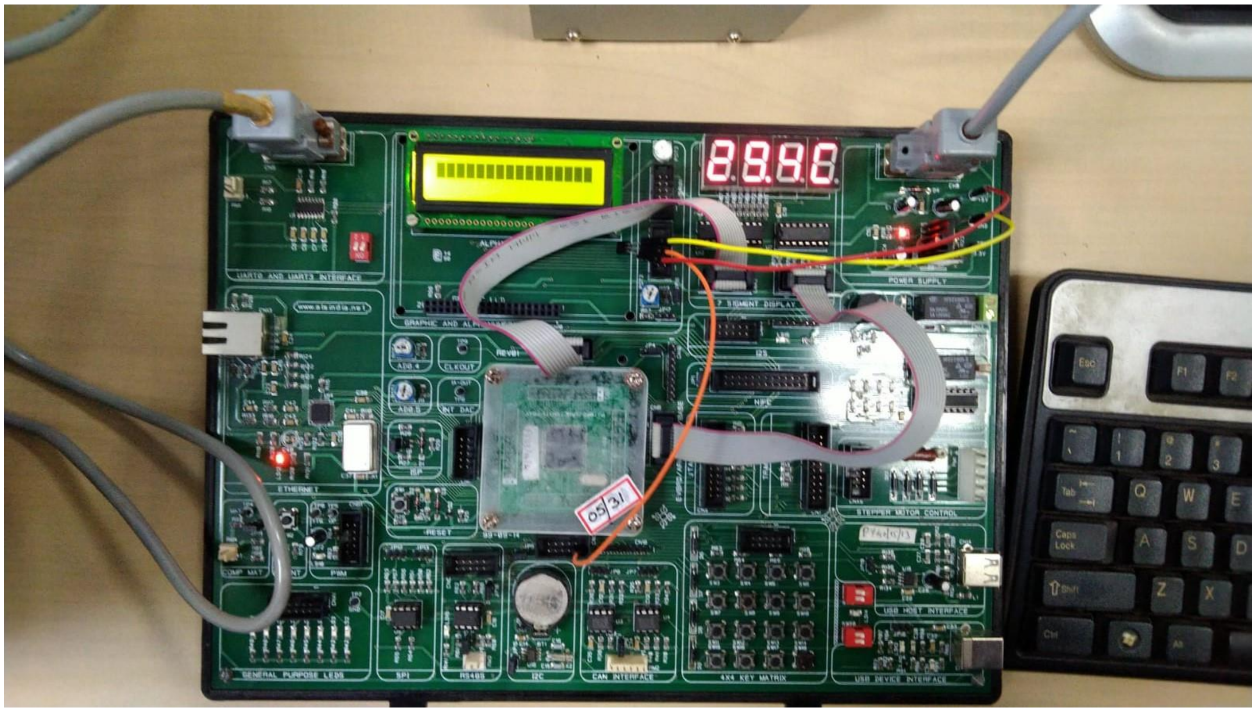
We used a hair drier to change the temperature.

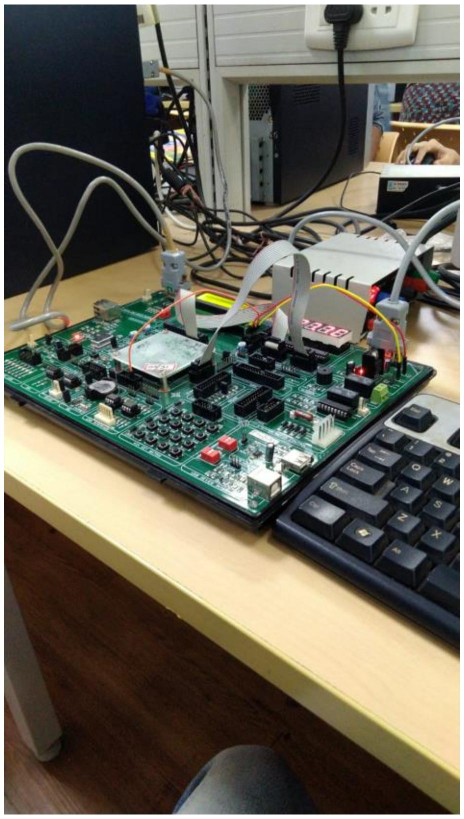
Minimum Temperature: 26.8 °C

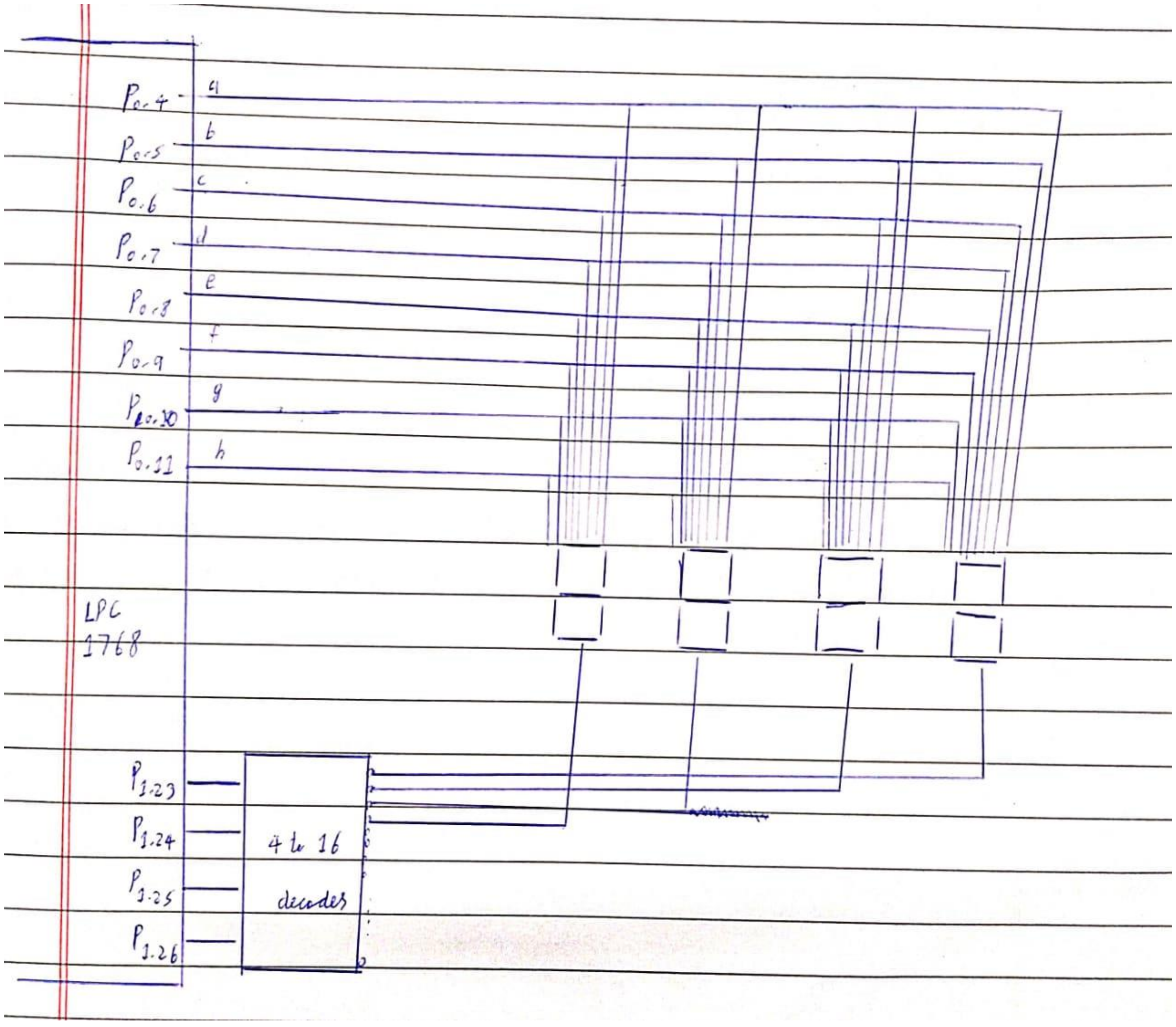
Maximum Temperature: 69 °C

Observation:

Initially the temperature on the 7 segment display was displayed as room temperature. Upon using the hair drier on the sensor, we could observe a gradual rise in temperature. After we turned it off, there was a gradual fall back to the room temperature corresponding to the sensor being back to its normal room temperature.






P0.4 a
P0.5 b
P0.6 c
P0.7 d
P0.8 e
P0.9 f
P0.10 g
P0.11 h
LPC 1768
P1.23
P1.24
P1.25
P1.26
4 to 16 decoder