



DIGITAL THERMOMETER BY USING AT89C51 MICROCONTROLLER

Rajdip Das

Department of Electronics and Communication Engineering Supreme Knowledge Foundation Group of Institutions 1, Khan Road Mankundu, Chandannagar West Bengal, India, 712123

Soumen Khatua

Associate Professor, Department of Electronics and Communication Engineering Supreme Knowledge Foundation Group of Institutions 1, Khan Road Mankundu, Chandannagar West Bengal, India, 712123

Article DOI: <https://doi.org/10.36713/epra5235>

ABSTRACT

Automation is nowadays a new trend of Electronics. This automation has been implemented by the help of microcontrollers. If a microcontroller is programmed efficiently then it is capable to work efficiently as human. This era of automation has reduced so much human efforts. In this century some changes has taken place in the field of technology. The most important of those is the use of digital circuits in everywhere. The world of digitization has made things easier than before or we can say it has reduced human error in most of the areas. In our research we have made a digital thermometer by using AT89C51 microcontroller. The other components we have used are LM35 temperature sensor, Analog to Digital converter IC (ADC0804), MAX-232 IC chip for serial communication with the microcontroller and LCD display to display the temperature. We hope that this research paper will be helpful to scientific improvement in the field of making Digital thermometers to a great extent.

KEYWORDS— Digital Thermo, Microcontroller, Timer, Temperature Sensor.

I. INTRODUCTION

The invention of microprocessors and hence microcontrollers were a big invention in the fields of technology. Throughout the years the microcontroller family has evolved to provide us with better opportunities of application. Now-a-days, dual core, 64-bit controllers are widely used all over the world. But, the need for basic 8bit microcontrollers have always been at a high because of their small size, simplicity, ease of use and most importantly they are quite cheap as well. Simple yet providing the opportunity to spread the wings of innovativeness, the basic microcontrollers have found applications in various fields and thus created the concept of Embedded Systems. In our paper we have used the 89C51 microcontroller to make a digital thermometer [1], providing us with temperature as accurate as 0.5 °C variation. Combining the hardware and software concepts together and interfacing the microcontroller with an analog to digital converter (ADC) chip and A LCD module to display the temperature we have implemented our project. By using a temperature sensor ranging from 0-100 °C scale, after this program the microcontroller has been built. Eventually to test its precision, performance and reliability, a virtual microcontroller was simulated. We have been able to make it

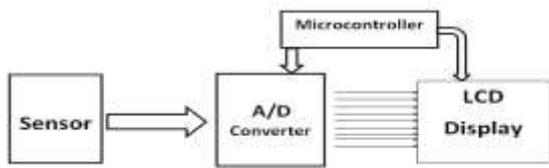
work as a Digital Thermometer. The rest of the paper is organized as follows: The block diagram is presented in Section III, the circuit diagram in Section IV, followed by description of electronic components in Section V, description of software design in Section VI, and experimental results in Section VII. Finally, the conclusion are discussed.

II. AIMS AND OBJECTIVES

Digital room temperature is very crucial, since a certain room or atmosphere needs to be digitally controlled which means that the analog temperature reader uses a tension overwriting which may require more measurements to obtain the actual ambient temperature. With the LM35 temperature sensor this device helps the consumer to achieve a more accurate temperature representation of the room.

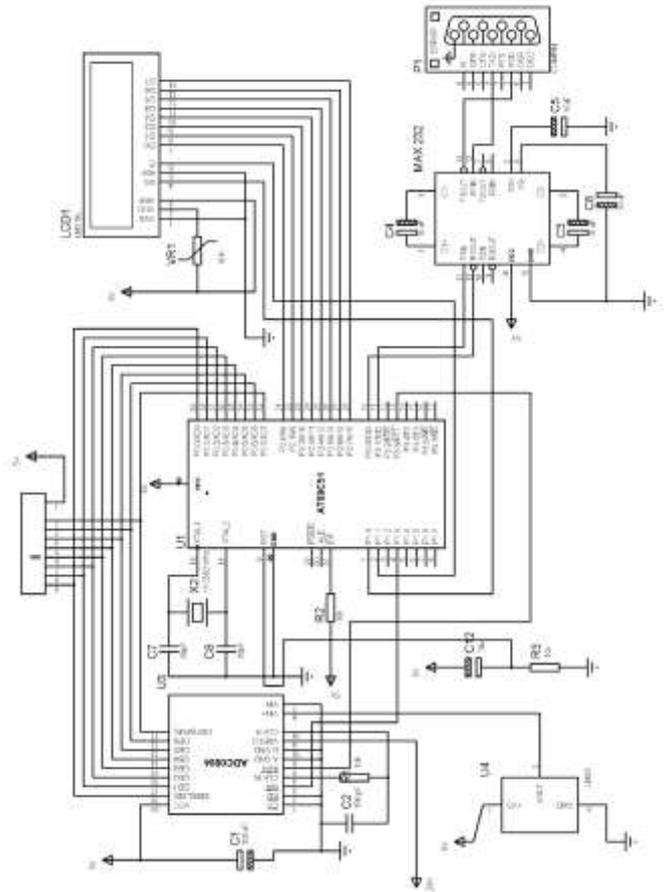
III. BLOCK DIAGRAM

The temperature sensor LM35 is used in the digital thermometer that takes input as analog signal [2]. The signal is converted into digital by ADC, and by the help of microcontroller consisting of software which has been simulated to check its precision. Fig 1. illustrates the block diagram of the whole system.

**Fig 1. Block Diagram of Digital Thermometer**

IV. CIRCUIT DIAGRAM

The LM35 series are circuit-temperature [3], [7] sensors with a precise circuit integrated output voltage that coincides linearly with centigrade. Since the consumer is needless to eliminate a too large constant voltage from the output for comfortable centigrade scaling, has a benefit over the linear temperature sensors tuned to ° Kelvin. Moreover, the LM35 sensor does not require any external calibration or trimming to supply typical accuracies of $\pm 0.1^\circ\text{C}$ over a full- 55°C to $+150^\circ\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The low output impedance, linear output, and precise inherent calibration of the LM35 make interfacing to read out or control circuitry especially easy. The device is employed with single power supplies, or with plus and minus supplies. As the LM35 draws only $60\mu\text{A}$ [4] from the provision, it has very low self-heating of less than 0.1°C in still air. LM35D is obtainable with an 8-lead, small-scale surface assembly and a TO-220 plastic kit (Texas, 2016). We have used the LM35 IC as the temperature sensor in our project. With the variations of temperature, the IC generates varying voltages which are fed to the analog input of the ADC which then sends the digital data to the 8051 microcontroller. The microcontroller recognizes the different digital inputs and thus displays the corresponding temperatures on the LCD. The circuit diagram consists of temperature a sensor, ADC, microcontrollers and LCDs. LM35 is used for measuring temperature, as a temperature sensor. In the shape of an analogue signal, the temperature sensor output is. ADC0804 is used to digitalize this analogue signal. The Microcontroller is supplied with the ADC output. The controller output is seen on the LCD. Hence, the positive temperature from this circuit configuration is determined from 0°C to 100°C . An ADC of AT89C51 microcontroller converts the sensor output voltage to digital voltage of 10 bit [5], [6]. Fig 2. illustrates the circuit diagram of the digital thermometer.

**Fig 2. Circuit simulation of digital thermometer.**

V. ELECTRONIC COMPONENTS

A. AT89C51 Microcontroller and MAX-232

The AT89C51 is a low power, high performance CMOS 8-bit microcontroller with a 4 Kilobytes of Flash Programmable Memory and Erasable Read Only Memory (PEROM). It also consists of 128 bytes of RAM, 32 I/O lines, two 16-bit timers/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. This microcontroller is used for this thermometer implementation. It consists of a 40 pin package which has 5 ports namely port A, port B, port C, port D, and port E. This microcontroller is commercially available everywhere which makes it more economic and simple. The MAX232 is used to transform signals of RS232, which would be suitable to 8051 'S TxD and RxD pins, into TTL voltage values. One bonus of MAX232 is the +5V energy supply, which is similar to the 8051 source voltage. So we can power both 8051 and MAX232 with a single +5V power supply without the need for dual power supplies typical to many older systems. The MAX232 has two lines for data transmission and reception. The TxD line drivers are called T1 and T2, while RxD is called R1 and R2. T1 and R1 for TxD and RxD 8051 in certain cases are used together, although the second pair is not used. The T1 line driver has a T1 in and T1out of pin numbers 11 and 14 respectively. The T1 in pin is the component of the TTL and associated with the TxD. The T1out is an RS232 side attached



to the RxD RS232 pin of the microcontroller. The bin plug of the R1 line driver also has a pin numbers classification for R1 in and R1out. The R1 in (pin 13) is the side of the RS232 attached to the TXD pin. The TTL side attached to the RxD pin of the RS232 DB and R1out (pin 12) microcontroller.

B. Display device

In our temperature sensor model, we have used the HD44780 controller based LCD module to display the temperature. Before using the LCD, we first have to initialize the LCD in the following manner mentioned in assembly language programming:

1. LCD_data equ P2 ;LCD Data port
2. LCD_D7 equ P2.7 ;LCD D7/Busy Flag
3. LCD_rsequ P1.0 ;LCD Register Select
4. LCD_rwequ P1.1 ;LCD Read/Write
5. LCD_enequ P1.2 ;LCD Enable
- 6.
7. LCD_init:
8. mov LCD_data,#38H ;Function set: 2 Line, 8-bit, 5x7 dots
9. clrLCD_rs;Selected command register
10. clrLCD_rw ;We are writing in instruction register
11. setbLCD_en ;Enable H->L
12. clrLCD_en
13. acallLCD_busy;Wait for LCD to process the command
14. mov LCD_data,#0FH ;Display on, cursor blinking command
15. clrLCD_rs;Selected instruction register
16. clrLCD_rw ;We are writing in instruction register
17. setbLCD_en ;Enable H->L
18. clrLCD_en
19. acallLCD_busy;Wait for LCD to process the command
20. mov LCD_data,#01H ;Clear LCD
- 21.
22. clrLCD_rs;Selected command register
23. clrLCD_rw;We are writing in instruction register
24. setbLCD_en ;Enable H->L
25. clrLCD_en
- 26.
27. acallLCD_busy;Wait for LCD to process the command
28. mov LCD_data,#06H ;Entry mode, auto increment with no shift
29. clrLCD_rs ;Selected command register
30. clrLCD_rw;We are writing in instruction register
- 31.
32. setbLCD_en;Enable H->L
33. clrLCD_en
34. acallLCD_busy;Wait for LCD to process the command
35. ret;Return from routine.

After initializing the LCD we can then use it for displaying the temperature by sending proper instructions to the LCD via microcontroller.

VI. SOFTWARE DESIGN

The software function of this digital thermometer design is achieved by Assembly Language. Basically the program highlights the data acquisition from the temperature sensor and display real-time data of temperature. The C code for our project (correct to 0.5 degree Celsius).

```
sbit LCD_RS at P1_0_bit; //initializing P1.0 as LCD_RS
sbit LCD_EN at P1_1_bit; //initializing P1.1 as LCD_EN
```

```
sbit ADC_SC at P1_3_bit; //initializing P1.3 as ADC_SC
unsigned int ADC_DATA=0,x,fnew; //initializing integer values
unsigned long double f; //initializing long double values
unsigned short ADC_READ=0,interrupt_status=0;
unsigned char ch; //initializing character value
void LCD_INI(); //declaring the functions void
LCD_DATA(unsigned short);
void LCD_COMM(unsigned short);
void conversion(unsigned int);
void convf(unsigned int);
void display();
void main() //main function
{
    IE=0x94; //setting the IE register with 94H value
    P0=0xff; //setting the port 0 as the output port
    P1=0x00; //setting the port 1 as the input port
    P2=0x00; //assigning values to port 1 and port 2
    LCD_INI(); //calling the function
    while(1) {
        delay_ms(5000); //delaying the process
        ADC_SC=0; /*sending a low to high pulse to the
        ADC_SC=1; start conversion bit of ADC*/
        if(interrupt_status==1)//checking the value of the flag
        {
            display(); //calling the function display()
        }
    }
}
//*****
//*****//
void LCD_INI() //initializing the LCD
{
    LCD_COMM(0x38);
    LCD_COMM(0x38);
    LCD_COMM(0x38);
    LCD_COMM(0x0c);
    LCD_COMM(0x01);
    LCD_COMM(0x06);
    LCD_COMM(0x80);
    delay_ms(100);
    return;
}
//*****
//*****//
void LCD_DATA(unsigned short x) //function LCD_DATA
{
    P2=x;
    LCD_RS=1;
    LCD_EN=1; //sending a high to low pulse to LCD enable pin
    LCD_EN=0;
    delay_ms(5); // delaying the program by 5 ms
    return;
}
//*****
//*****//
void LCD_COMM(unsigned short y) //function LCD_COMM
{
    P2=y; //port 2 is sent the value y
    LCD_RS=0;
    LCD_EN=1; // high to low pulse given to LCD enable
    LCD_EN=0;
    delay_ms(5);
    return;
}
```



```

}
//*****
//*****//
void interrupt() org IVT_ADDR_EX1 //function for interrupt
{ interrupt_status=1; //setting the flag
ADC_READ= P0 ; //reading the content of P0
ADC_DATA=ADC_READ*5; //multiplying the value with step size
return;
}
//*****
//*****//
void display() //the function to display
{
interrupt_status=0; //resetting the flag
x=ADC_DATA; conversion(ADC_DATA); //calling the function
LCD_DATA('C'); //display the character C on LCD
LCD_COMM(0x80); //displaying on the first line of LCD
return;
} void conversion(unsigned int a) //function to convert into
{
ASCII ch=(a/100)+48; //the first digit is converted to ASCII
LCD_DATA(ch); //calling the function LCD_DATA
ch=((a/10)%10)+48; //the second digit is converted LCD_DATA(ch);
LCD_DATA('.'); //displaying the decimal point ch=(a%10)+48;
//displaying the digit after the decimal LCD_DATA(ch);
LCD_DATA(223); //displaying the character ° (degree)
return;
}

```

VII. EXPERIMENTAL RESULTS

The circuit was tested after the final assembly and Fig 3 and Fig 4 revealed final figures. The final result reveals the temperature. The output of the sensor is linear. Timing and temperature system calculation dimensions are carried out successfully.



Fig 3. The LCD displaying the room temperature of the lab. The DC power supply is supplying the 5V 0.03A current to the circuit for it to operate.

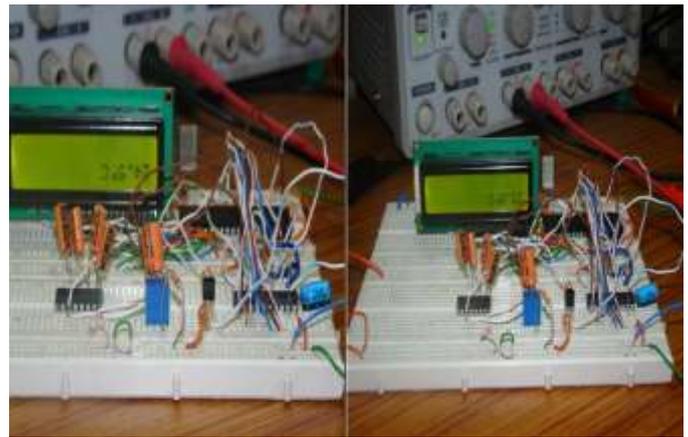


Fig 4. The two clips shows the temperatures at two instants. The left one shows 24.5°C and the right one shows 25.0°C

CONCLUSION

In this paper we got a clear idea about the functions of the 8051 microcontroller. We used C code to program the microcontroller. Secondly, we learnt about RS232 for serial communication with the microcontroller and also about the MAX232 IC chip to convert the voltage levels of RS232 to the corresponding TTL voltage level. We used the analog to digital converter ADC0804 to convert the analog voltage signals to its corresponding digital output and how to use this output in 8051 for various purposes. We learnt to interface the LCD with the microcontroller and hence displaying data on it. In our paper we used LM35 temperature sensor which gives varying voltage levels upon variations of temperature. These varying voltage levels are fed into the ADC and the digital output is fed into the microcontroller which is also programmed to display the corresponding temperature on the LCD. Thus we made a digital thermometer which is correct upto 0.5°C ranging from 0°C to 100°C which is capable enough to be used in practical day to day activities. Last, but not the least, the knowledge of the microcontroller has inspired us to portray our innovations and expand our thought process in new fields, which will be beneficial to human beings.

ACKNOWLEDGEMENTS

I want to acknowledge my deep sense of gratitude to my research guide Mr. Soumen Khatua who directed and guided me with her timely advice and constant inspiration which eased the task of my research work.

REFERENCES

1. "Digital Thermometers", *Global Spec, the Engineering Search Engine*, available online at <http://material-handling.globalspec.com/>, accessed on 16/12/2011.
2. Theophilus Wellem, Bhudi Setiawan, "A Microcontroller-based Room Temperature Monitoring System", *International Journal of Computer Applications* (0975 – 8887), Volume 53– No.1, September 2012.
3. Xing Guo Quan, "Temperature Characteristics and Application of LM 35 Sensor", *Biomedical Engineering*



Department of Xianmng College, Xianning Hubei-437100, China, Article id 1007— 7510(2007)11—0049—02, 2007.

4. *Gao Mei Zhen, “Principle and application of the LM 35 series temperature sensor”, Hubei normal university, China Article Id: 1671-1041(2005)01-0114-O 1, 2005.*
5. *Theophilus Wellem, Bhudi Setiawan, “A Microcontroller-based Room Temperature Monitoring System”, International Journal of Computer Applications (0975 – 8887), Volume 53– No.1, September 2012.*
6. *Xing Guo Quan, “Temperature Characteristics and Application of LM 35 Sensor”, Biomedical Engineering*

Department of Xianmng College□Xianning Hubei-437100□C hina , Article id 1007— 7510(2007)11—0049—02, 2007.

7. *B. O. and Odueso V. T. (2017), Design and Development of Microcontroller Based Temperature and Humidity Controller for Infant Incubator. Journal of Industrial Technology, 2 (4): 78 –87.*