MBSD Project Report

Temperature Monitoring Using 8085 Microprocessor

A Project Report Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Bachelor of Technology

In

Computer Science Department

By

Indradhar Paka
Virat Chowdhary
Akhil Ram
Raghavendra Naga Varma

CSE+CSC-1
2nd year

BML MUNJAL UNIVERSITY

SCHOOL OF ENGINEERING AND TECHNOLOGY

BML MUNJAL UNIVERSITY GURGAON

May,2020

TABLE OF CONTENTS

SUBMITTED TO

Dr. Nirupama
Assistant Professor
Computer Science and Engineering Department
School of Engineering and Technology

PARTICULARS OF STUDENTS:

S No.	Name	Roll No.
1.	Indradhar Paka	1800315C203
2.	Raghavendra Naga Varma	1800241C203
3.	Virat Chowdhary	1800327C203
4.	Akhil Ram	1800262C203

ABSTRACT

Microprocessor Based Systems is an essential course in computer science curriculum, which helps students to develop a mental model of how microprocessors work. The internal architecture and interfacing of a microprocessor are often complex, non-deterministic which makes them difficult for students to understand. One such concept is temperature monitoring using 8085 microprocessor.

In practice, MBSD courses involve classroom lectures describing high-level abstractions of the concepts, and students complete programming assignments to apply the material in a more concrete way. Depending on the programming assignments, this approach may leave students with only a theoretical understanding of MBSD ideas, which may be different from the actual way these concepts are implemented in a microprocessor. What many students require is a practical knowledge of microprocessor implementation to supplement the high-level presentations of concepts taught in class or presented in a textbook.

Our project covers all the interfacing components from a microprocessor to external components using programmable peripheral interface (PPI), ADC, DAC in order to monitor the temperature.

PROBLEM STATEMENT AND MOTIVATION

<u>“Temperature monitoring using 8085 microprocessor”</u>

In our project we are trying to achieve an optimal environment for home/plant working conditions so that the temperature we wish to work in is always achieved and maintained irrespective of changes in the external environment. The system is implemented using the Intel 8085 microprocessor, an 8-bit microprocessor introduced in 1976. Eagle software was used to draw out its schematic and board files. It is implemented on a Printed Circuit Board (PCB).

The motivation for this project came to us while we were sitting and sipping coffee on a chilly winter night in our campus at Haryana. We hoped for a device which could “sense” the changes in temperature inside the room and eradicate the need for us to keep changing the intensity of heater when it got too hot or too cold.

INTRODUCTION AND DESCRIPTION

It is the process in which change of temperature of space (and objects there collectively there within) is measured or otherwise detected and the passage of heat energy into or out of the space is adjusted to achieve a desired temperature range.

Industrial and control application may require automation of the process such as temperature, pressure, liquid flow, etc., in order to minimize manual intervention. To automate any application an intelligent processor plays a major role. One such processor proposed for the project is 8085, an 8-bit microprocessor. The temperature controller can be used to control the temperature of any plant. Typically, it contains a Processor unit, Temperature input unit and Control output unit. The 8085 based motherboard forms the processing unit. The Analog-to-Digital unit together with temperature sensor forms the temperature input unit. The relay driver forms the control output unit. Electric power to the heating element (coil) is supplied through relay contacts. The switching ON/OFF the relay controls the heat supplied to the plant.

Operationally, the system requires two set points-upper and lower, to be entered by the user. Whenever the temperature of the plant exceeds the upper limit or recede the lower limit relay is turned-off, so that a temperature is maintained within limits. The software for the temperature controller is developed in 8085 assembly language programs.

PROBLEMS FACED

The main problem that we faced during the initial stage was interfacing 8255 chip with our microprocessor and programming in machine language was our primary issue. We could have used any laboratory work for better experience of the work.

Solution: We came up with a clever way of designing the entire system and doing the simulation using simulators provided by JAVA. So, we learned how to do it in a stable manner and were able to create our project.

DETAILED EXPLANATION

The plant may be an industry or an equipment or a furnace for which the temperature must be monitored and maintained continuously at a particular temperature. The transducer measures current temperature and it is amplified by the amplifier which gets converted to hexadecimal digital value in ADC. This value is compared with the already set desired value in the microprocessor and the result of the comparison is as an error signal or null. The error may be positive meaning the temperature has to be increased by a heater arrangement. It may be negative meaning the temperature has to be reduced by some cooling set up. If there is no error, then no process must be done and again the sensing activity must be done repeatedly for continuous temperature maintenance.

For interfacing temperature control system with microprocessor, 8255 (PPI) and suitable ADC are connected between microprocessor and sensor output. Figure 1 shows the interface diagram of successive approximation A/D converter to the microprocessor through 8255 (PPI).

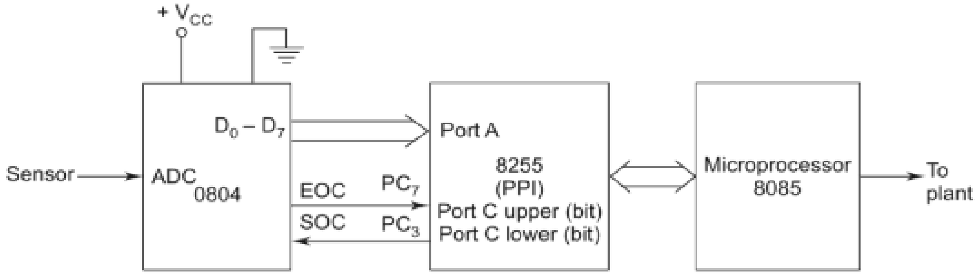


Figure 1 Interfacing A/D converter with microprocessor

The microprocessor sends a Start of Conversion (SOC) signal to the A/D converter through Port Cower of 8255. When A/D converter computes conversion, it sends as End of Conversion (EOC) signal to the microprocessor. Having received as EOC signal from A/D converter, the microprocessor reads the output of an A/D converter which is a digital quantity proportional to the temperature to be measured.

ADC Interfacing with 8085 microprocessor

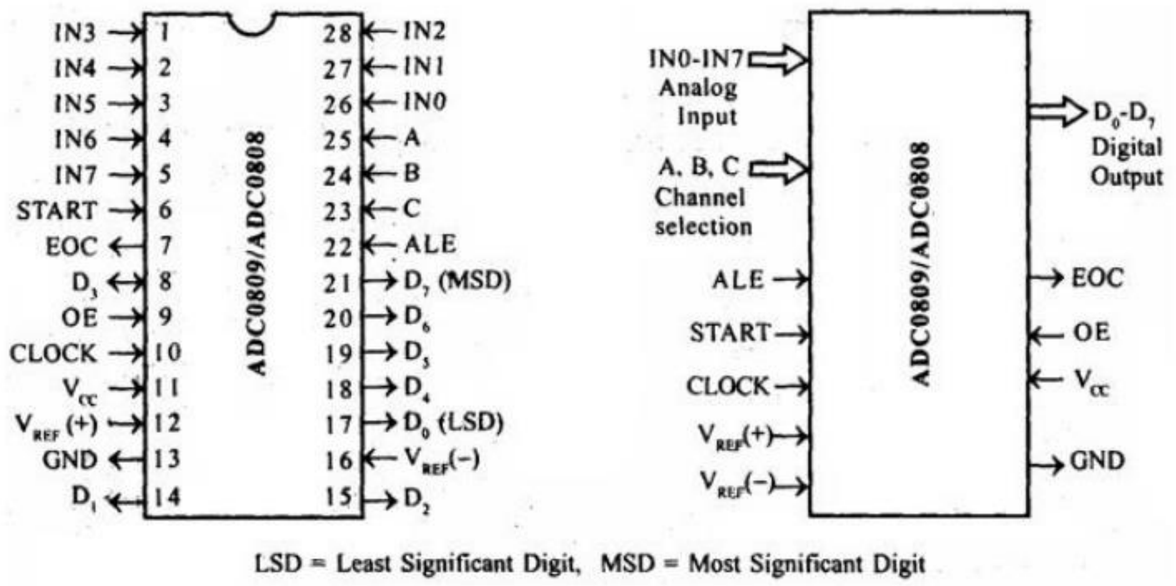


Figure 2 Pin diagram of ADC 0809

It converts the DC input signal into an AC signal and amplifies the AC sign using high gain AC amplifier. Then it converts AC signal to DC signal. This technique limits the drift component of the amplifier, because the drift is a DC component and it is not amplified/passed by the AC amp1ifier. This makes the ADC extremely insensitive to temperature, long term drift and input offset errors.

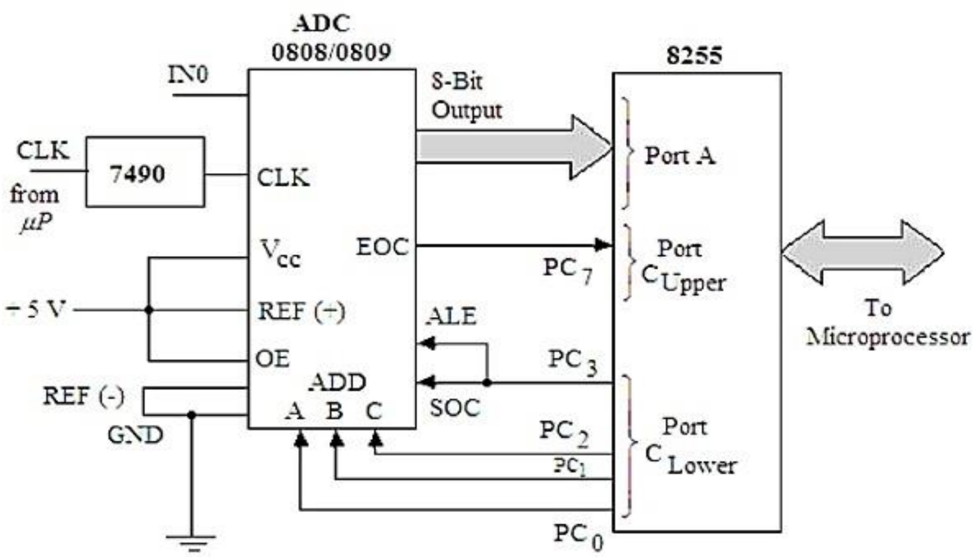


Figure 3 Interfacing ADC 0808

The Port A of 8255 chip is used as the input port. The PC_7 pin of Port C_{upper} is connected to the End of Conversion (EOC) Pin of the analog to digital converter. This port is also used as input port. The C_{lower} port is used as output port. The PC_{2-0} lines are connected to three address pins of this chip to select input channels. The PC_3 pin is connected to the Start of Conversion (SOC) pin and ALE pin of ADC 0808.

8255 interfacing with 8085 microprocessor

These signals relate to the following signals of 8085 or derived from 8085:

- Demultiplexed Do – D_7 data lines.
- A_1- A_0 lines of demultiplexed address bus.
- IOR’ or MEMR’
- IOW’ or MEMW’
- The remaining address lines of 8085. (i.e., A_2 – A_7 or A_2 – A_{15}) are used to generate a chip select signal.
- Reset outline of 8085.

8255 can be interfaced with 8085 in two different ways, i.e., **I0 mapped I0** and **memory mapped I0** techniques**.**

IO mapped IO

In this technique the control signals are IOR’ and IOW’, which are generated by a control signal generating logic. The port addresses in this case are of 8-bits, i.e., the chip select logic must decode only A_2- A_7, address lines to generate the chip select signal.

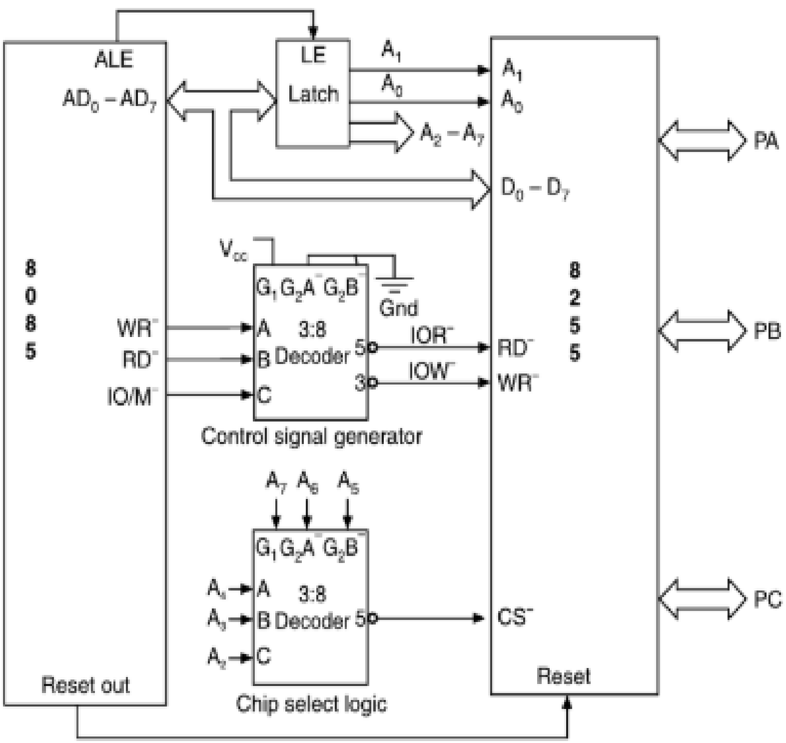


Figure 4 Interfacing of 8255 with 8085 in IO mapped IO

one such interfacing of 8085 with 8255. In Figure 4 the port and the CWR (control word addressing) addresses are:

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	HEX	Port
1	0	0	1	0	1	0	0	94H	A
1	0	0	1	0	1	0	1	95H	B
1	0	0	1	0	1	1	0	96H	C
1	0	0	1	0	1	1	1	97H	CWR

Decides by chip select logic

TECHNICAL FEATURES AND ELEMENTS

We can divide the hardware broadly into:

Temperature Input Unit : Comprises of ADC and the sensor

Processing Unit : The Main Processing Unit i.e. the 8085 M.P

Control Output Unit : Relays and Output Control and Read-Out

The Hardware used is comprises of the following:

- 8085 microprocessor motherboard.
- ADC interface using the IC 0809
- 32KB EPROM, 32KB RAM using AT28C256-28DIP
- 3*Latches 74HC573N
- 2*Decoder 74HCT138N
- 2*7-Segment LEDs
- Temperature Sensor LM135
- Output Drivers and Relays

TEMPERATURE SENSORS:

There are many types of temperature sensors that can be used:

- Thermistor
- Thermocouple
- Solid State Temperature Sensor
- IC temperature sensor (LM-135) (voltage output)
- LM-336 (current output)

Features of LM135:

- Directly calibrated to Kelvin temperature scale
- 1^oC Initial accuracy Available
- Operates from 400 µA to 5mA
- Less than 1Ω Dynamic Impedance
- Easily Calibrated
- Wide operation temperature range
- 200^oC overrange

EXISTING STATE OF THE ART

Temperature monitoring system has been developed by many sources and also available in advanced programming tools like java and latest microprocessors has been used.

To better illustrate the existing state of the art, I designed a table which consists all the information.

S No.	Existing State of the art	Drawbacks in existing state of the art	Overcome
1	Temperature monitoring using Arduino	Has dependency on Arduino	We can do it easier without Arduino
2	Temperature monitoring using 8086	Not a drawback	It would be as like 8085

DISTINGUISHABLE FEATURES

This project is also created by by using several microprocessors. But those are dependent on the technologies. Our project uses 8085 microprocessor and 8255 for interfacing. It would be lesser cost and efficient way to monitor temperature in plants etc.

ALTERNATE SOLUTION

There may be many alternate implementations of this project. People may use technologies in which they are comfortable and can implement the project using technologies such as Arduino, Raspberry, 8086 M.P etc. Even I initially thought of doing this project on 8086. But it is quite difficult to interface these components to the microprocessor.

BLOCK DIAGRAM AND FLOW CHART

Transducer:

For the measurement of physical quantities transducers are used. They convert them to electrical quantities. Here, for measuring temperature, sensors like thermocouple, thermistor, sensistor (whose resistance changes with temperature) can be used.

Amplifier:

If the electrical signal from transducer is small, it cannot be visualized or processed. Hence, it is amplified using amplifiers.

ADC:

The electrical signal from transducer is an analog signal which a microprocessor cannot process. Hence, an analog to digital converter is used.

DAC:

The signal from microprocessor will be digital signal which is going to control the analog elements like heater, cooler etc. For that the digital signal from the microprocessor has to be converted to analog by a DAC

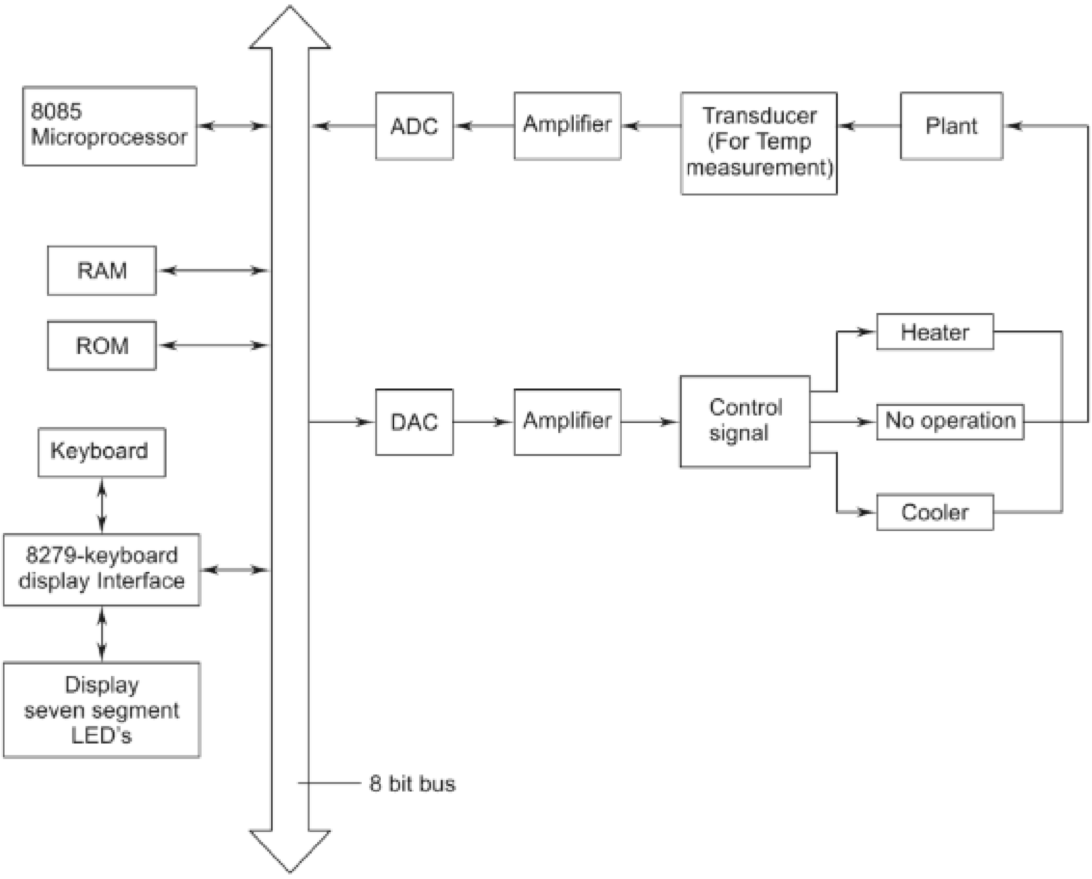
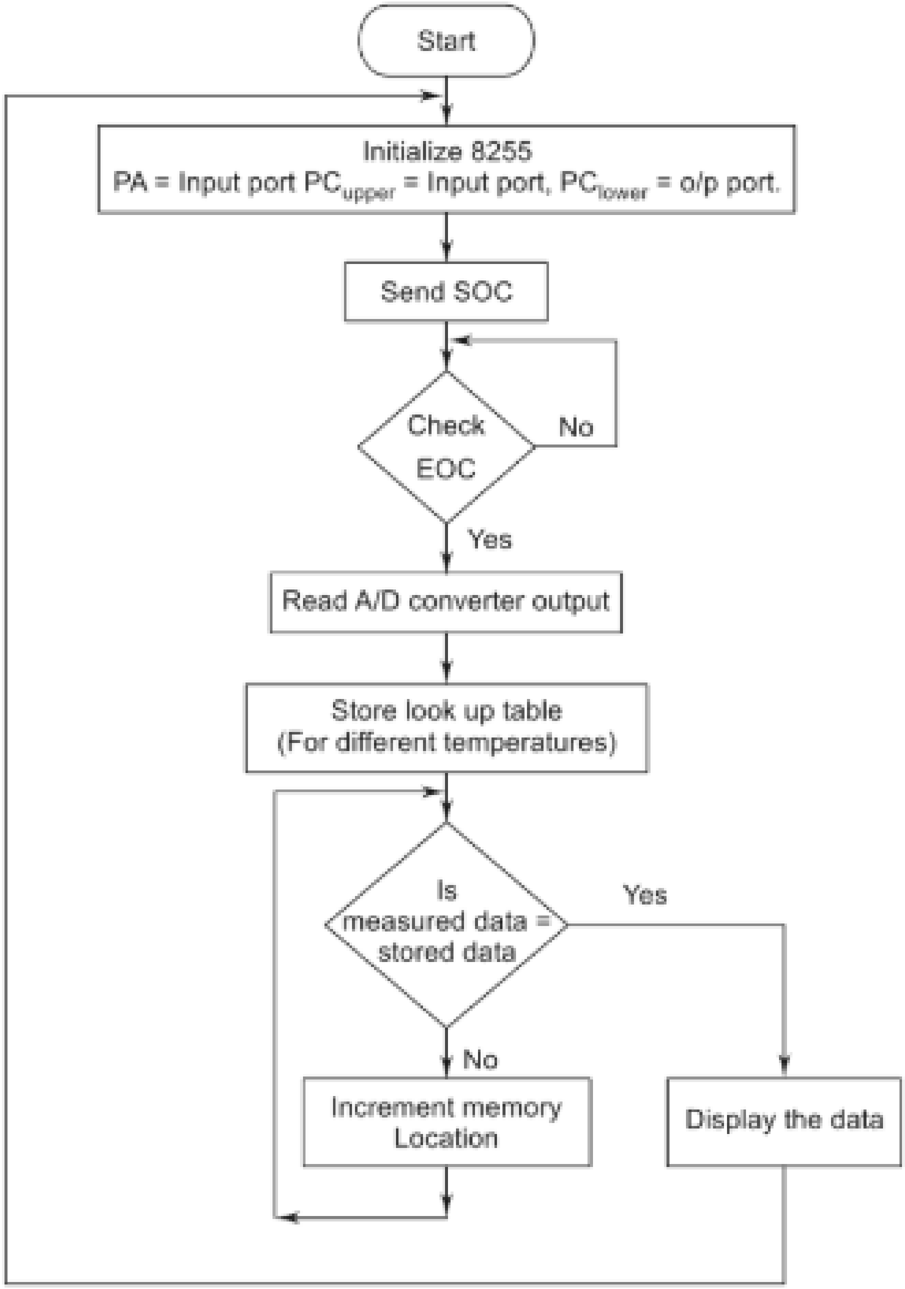


Figure 5 8085 microprocessor bases temperature monitoring system

FLOWCHART:



ALGORITHM AND PROGRAM

Step 1. Initialize 8255 port A as an input port, port C_{upper} as an input port and port C_{lower} as an output port.
Step 2. Send SOC to an A/D converter (C_L port).
Step 3. Check for EOC from A/D converter (C_U port).
Step 4. Read the data from A/D converter.
Step 5. Store the temperature (in °C) and corresponding digital voltage in memory locations.
Step 6. Compare the output of A/D converter with value in the memory location.

Step 7. Display the equivalent value (in ^oC).
Step 8. Repeat steps 2 to 7 for continuous temperature measurement and control.

Program:

```
MVI A,98
OUT[CR]
MVI A,00
OUT[PC]
MVI A,08
OUT[PC]
IN[PC]
RAL
JC READ
IN[PA]
LXI H, LOOP UP
MOV C, M
INX H
CMP M
JC GO
JZ GO
DCR C
JNZ SEARCH
INX H
MOV A, M
STA 5000
MVI B,00
CALL DELAY
JMP START
LXI D, FFFE    //DELAY LOOP
DCX D
MOV A, E
ORA D
JNZ LOOP
```

CONCLUSION

Temperature monitoring is a major concept in MBSD. Since this project has been designed exclusively as a project, certain complexities that do faced by any real-life manual problem like total interfacing the 8255 chip are considered in this project. But enhancement to the project can easily be made without changing the current design and programming structure.

This project gave also improved the understanding microprocessor concepts and to implement various functionalities of MBSD concepts.

I thank our faculty mentor for giving such opportunity for making us learn those concepts practically in the form of project work.

REFERENCES:

- Microprocessors Architecture Programming and Applications with 8085 – *by R.S. Ganokar, Fifth Edition (Penram International Publishing)*

- Introduction to Microprocessor- *by A.P. Mathur (Tata McGraw-Hill publishing)*

- https://www.scribd.com/doc/68632956/Temperature-Controller-Using-8085-Microprocessor#scribd

- http://www.ti.com/product/LM135

- https://en.wikipedia.org/wiki/Intel_8085

- http://www.daenotes.com/electronics/digital-electronics/Intel-8085-8-bit-microprocessor

- http://www.dailyfreecode.com/Tutorial/Easy-Assembly_Language-49.aspx