CHAPTER – 1

INTRODUCTION

1.1 Introduction And Motivation

The project “Temperature Monitoring and Controlling Using Arduino”, controls cooling system automatically according to the room temperature. The system is designed with Arduino (microcontroller) and Arduino are increasingly being used to implement control systems. Since the system is intended to control the cooling system, it is therefore important to understand Arduino controlled system well. Nowadays with the advancement of technology particularly in the field of micro-controllers, all the activities in our day-to-day living have become part of information technology and we find controllers in each and every application. Thus, the trend is directing towards micro-controller based project works. A micro-controller contains a CPU, clock circuitry, ROM, Ram and I/O circuitry on a single integrated circuit package.

The purpose of this project work is to present control theory that is relevant to the analysis and design of controlled systems, with an emphasis on basic concepts and ideas. It is assumed that a digital micro-controller chip with reasonable software is available for computations and simulations so that many tedious details can be left to the micro-controller. The control system design is also carried out up to the stage of implementation in the form of micro-controller programs.

In this project work, the program is written in Arduino IDE and facilitates the display of temperature in degree centigrade and also in Fahrenheit. And according to the temperature the Arduino gives input to the Relay, thereby cooling system on/off automatically depending up on the temperature. The most fundamental parameter of an industry is temperature. Monitoring and control of temperature of any oven, furnace, broiler, etc.; (normally belongs to an Industry) is very essential, otherwise; the material inside the oven or furnace may spoil because of temperature variations. The temperature sensing probe can be placed at exact location where the condition of the temperature to be monitored continuously. The exact location where the sensing probe is to be installed should be determined on the case of access and the degree of accuracy obtainable at the given point. The steps to be taken to check the accuracy during and after the test are also of extreme importance.

Temperature is the most often-measured environmental quantity. This might be expected since most physical, electronic, chemical, mechanical and biological systems are affected by temperature. Some processes work well only within a narrow range of temperatures. Certain chemical reactions, biological processes, and even electronic circuits perform best within limited temperature ranges. When these processes need to be optimized, control systems that keep temperature within specified limits or constant are often used. The field of process control has grown rapidly since its inception in the 1959s. It has become one of the core areas of chemical engineering. One of the most important process variables to be controlled is temperature of liquid in many chemical engineering Plants. In brief, how to control the temperature of a water bath is one of the industrial problems. Hence suitable temperature controller for controlling temperature of a water bath is very common requirement of the industries.

Temperature rise due to resultant increase in molecular activity of substances on application of heat, which also connotes an increase in internal energy of the material. Heat exchange, heat balances, safe temperature limits of various equipment's and multitude of other problems involving temperature are important in all phases of power generation and process industries. The means of indicating, recording and controlling temperatures are important in design, fabrication, operation, and testing power generating and utilizing equipment and associated apparatus. Correct measurement of temperature is of paramount importance to power and process engineers and great strides have been made in this branch. The ultimate aim is to measure temperatures correctly and reliably, at the same time keeping the cost of instruments down.

1.2 Existing System

Many existing system for temperature monitoring and controlling generally uses micro-controller ATMEL 89C51 (µc 8051). It does the same job by using additional devices. The microcontroller-controlled system contains essentially four parts, i.e., the process, the analog to digital converter, the control algorithm, and the clock. The times when the measured signals are converted to digital form are called the sampling instants; the time between successive samplings is called the sampling period and is denoted by h. The output from the process is a continuous time signal. The output is converted into digital form by the A – D converter. The conversion is done at the sampling times.

1.3 Problem Associated With Existing System

Many existing system for temperature monitoring and controlling generally uses micro-controller ATMEL 89C51 (µc 8051). Due to using micro controller 8051 the process of making whole device becomes not only very complex but also difficult and tedious. For operation it requires A-D converter, external clock, microcontroller development board.

Consequently, the problems are as follows :-

a) It takes comparatively more time to process.
b) It requires additional devices for operation.
c) It requires external clock.
d) Programming for microcontroller 8051 is difficult .
e) For programming it requires development system.
f) Circuit size becomes large.
g) PCB making becomes complex, difficult and tedious.

1.4 Objectives

The main objective of this project is to display the temperature and when it goes beyond certain limit then control it to bring it back into desired level and reduce waste of energy. And also to assist people who are disabled and are unable to control the speed of fan. It may also be used for monitoring changes in environment. In near future, it can also be used in different industries and electronic devices. Another objective is to study and build an automatic system using microcontroller and its interfacing with other device. Nowadays

Arduino becomes more popular because of its many advantages like simple programming and compact in size. It supports many device, so that our goal is to achieve ability to do programming and get idea about the Arduino system.

1.5 Proposed System

Existing system uses ATMEL 89C51 has many disadvantages as seen above in the section 1.3 above, to overcome this problems we use another advanced microcontroller called Arduino (ATmega8). It has in built with many components like analog to digital converter, clock of 16 MHz, shift registers.

In this system we uses temperature sensor LM35, to use to detect temperature into appropriate voltage. This voltage is given to Arduino. According to program it process the analog signal into digital and forms an particular voltage level for a particular temperature.16x2 LCD is used to display the output i.e. surrounding temperature of LM35 in both degree centigrade and Fahrenheit units.

At the same time it also sends the data to Relay, if the temperature becomes maximum from set point relay becomes activate and it switches on the cooling device like fan. In this manner it monitors and controls the temperature.

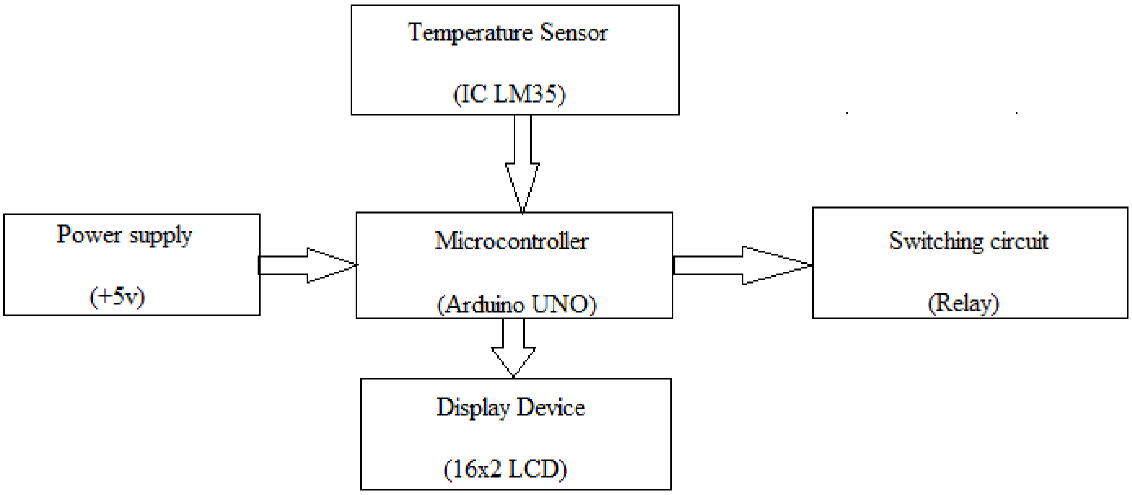


Fig 1.5 :- Block diagram for proposed system

CHAPTER – 2

REQUIREMENT ANALYSIS

2.1 Hardware Requirement

This project is based on both hardware and software. The hardware requirements are as follows :-

2.1.1 Arduino

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board – you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

The Arduino is a microcontroller board based on the ATmega8. It has 14 digital - input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a

16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode. Revision of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible with both the board that uses the AVR, which operates with 5V and with the Arduino Due that operates with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- ATmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform.

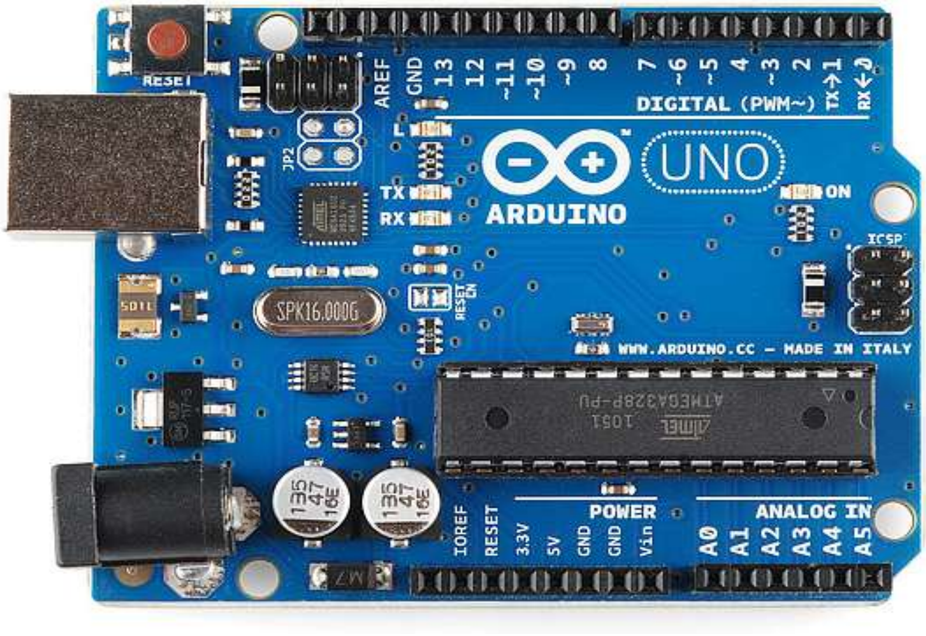


Fig. 2.1.1 :- Real Arduino UNO board

Parameters For Arduino UNO	Description
Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Table 2.1.1 :- Specifications of Arduino

2.1.2 IC LM35 Temperature Sensor

The LM35 is a popular and inexpensive temperature sensor. It provides an output voltage of 10.0mV for each degree Centigrade of temperature from a reference voltage. The output of this device can be fed to A/D Converter; any microcontroller can be interfaced with any A/D Converter for reading and displaying the output of LM35. The circuit should be designed, so that output should be at 0V when the temperature is 0 degrees Centigrade and would rise to 1000mV or 1.0V at 100 degrees Centigrade. To get the temperature value accurately, output voltage must be multiplied with 100. For example if you read 0.50V that would be 50 degrees Centigrade.

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, linear output, and precise inherent calibration of the LM35 device makes interfacing to readout or control circuitry especially easy. The device is used with single power 2 Applications supplies, or with plus and minus supplies. As the LM35 device draws only 60 µA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 device is rated to operate over a −55°C to 150°C temperature range, while the LM35C device is rated for a −40°C to 110°C range (−10° with improved accuracy). The LM35-series devices are available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D devices are available in the plastic TO-92 transistor package.

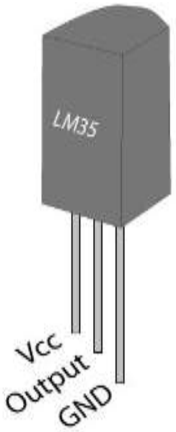


Fig. 2.1.2 :- Pin diagram of IC LM35

2.1.3 Liquid Crystal Display (LCD)

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

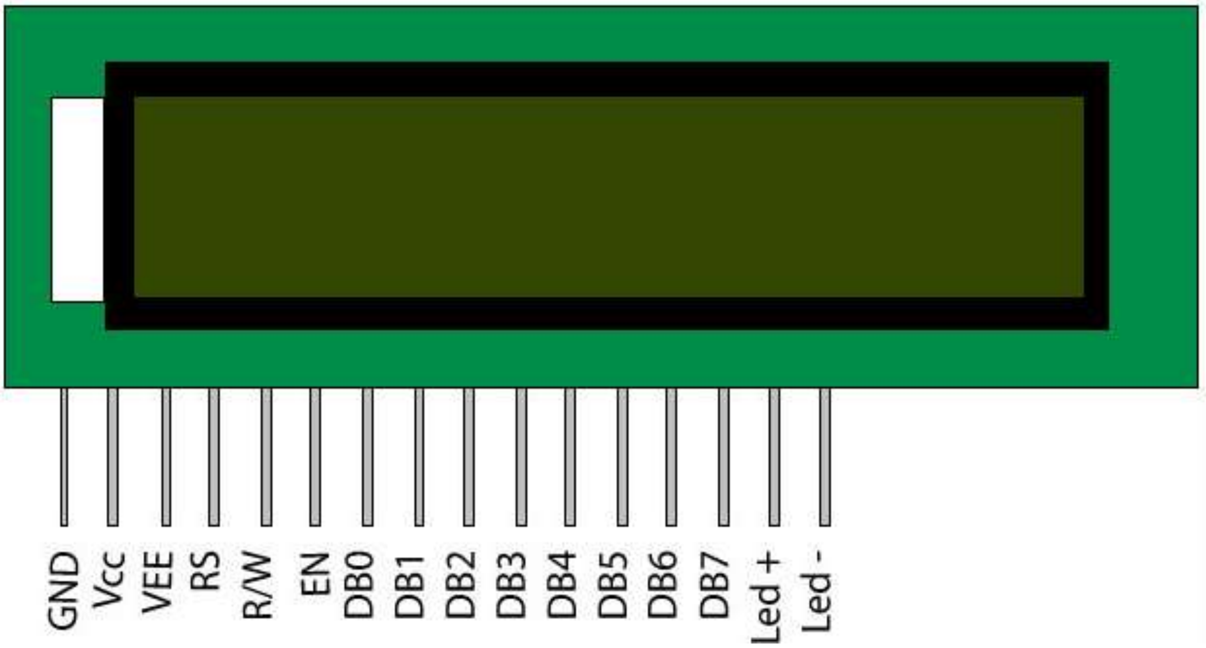


Fig 2.1.3 :- pin diagram of 16x2 LCD

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V_{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

Table 2.1.3 :- Pin description for 16x2 LCD

2.1.4 Relay

A relay is an electrically operated switch. Many relays use an electromagnet to mechanically operate a switch, but other operating principles are also used, such as solid-state relays. Relays are used where it is necessary to control a circuit by a low-power signal (with complete electrical isolation between control and controlled circuits), or where several circuits must be controlled by one signal. The first relays were used in long distance telegraph circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations.

When electrical relays used to allow low power electronic computer type circuits to switch relatively high voltage and current both on and off some form of relay switching circuitry is required to control it. In this project we used NPN relay switch circuit.

The relay coil is not only an electromagnetic but al also an inductor. When power is applied to coil due to the switching action of transistor, a maximum current will flow as a result of DC resistor coil. Some of this energy is stored within the relay coil's magnetic field. When transistor turns off, the current flowing through relay coil decreases and magnetic field collapses. However stored energy within magnetic field has to go somewhere and reverses voltage is developed across the coil as it tries to maintain the current in the relay coil.

This action produces a high voltage spikes across the relay coil that can damage the switching NPN transistor (BC 548). So in order to prevent damage to transistor a fly wheeling diode is connected across the relay coil. This diode clamps the reverse voltage across the coil about 0.7V dispatching the stored energy and prevents the transistor.

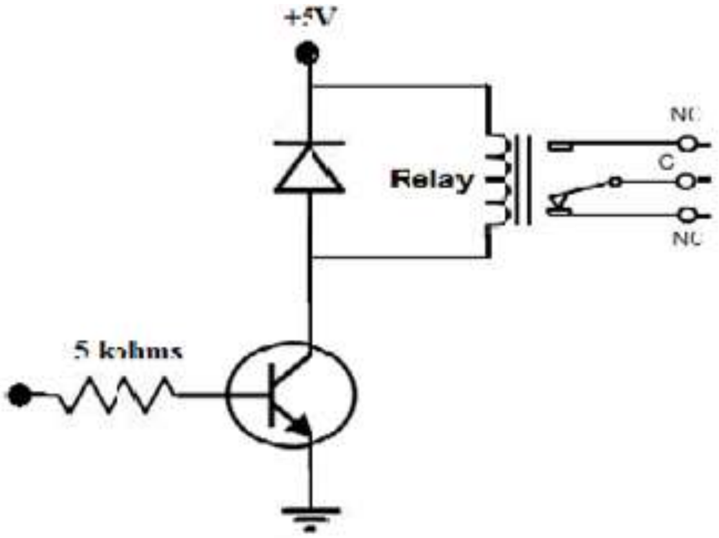


Fig 2.1.4 :- Relay Circuit

2.1.7 Power Supply

A power supply is an electronic device that supplies electric energy to an electrical load. The primary function of a power supply is to convert one form of electrical energy to another and, as a result, power supplies are sometimes referred to as electric power converters. Some power supplies are discrete, stand-alone devices, whereas others are built into larger devices along with their loads. Here, we use 5v dc power or sometimes power is given to the circuit directly from computer.

2.1.8 Connecting Wires

A Wire is a single usually cylindrical, flexible strand or rod of metal. Wires are used to bear mechanical loads or electric and telecommunication signals. Wire is formed by drawing the metal through a hole in a die or draw plate.

2.2 Software Requirement

As explained earlier our project requires two part hardware and software. Hardware parts are explained above and software requires as follows :-

2.2.1 Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. The Arduino development environment contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

Software written using Arduino are called sketches. These sketches are written in the text editor. Sketches are saved with the file extension .ino. It has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino environment including complete error messages and other information. The bottom right-hand corner of the window displays the current board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

CHAPTER – 3

DESIGN AND PLANNING

3.1 Process Model

In this section we design structure of the system before implementation of circuit. we use advanced microcontroller called Arduino (ATmega8). It has in built with many components like analog to digital converter, clock of 16 MHz, shift registers.

In this system we uses temperature sensor IC LM35, to use to detect temperature into appropriate voltage. This voltage is given to Arduino. According to program it process the analog signal into digital and forms an particular voltage level for a particular temperature.16x2 LCD is used to display the output (i.e. surrounding temperature of LM35) in both degree centigrade and Fahrenheit units. At the same time it also sends the data to Relay, if the temperature becomes maximum from set point relay becomes activate and it switches on the cooling device like fan. In this manner it monitors and controls the temperatur

3.2 Data Flow Diagram

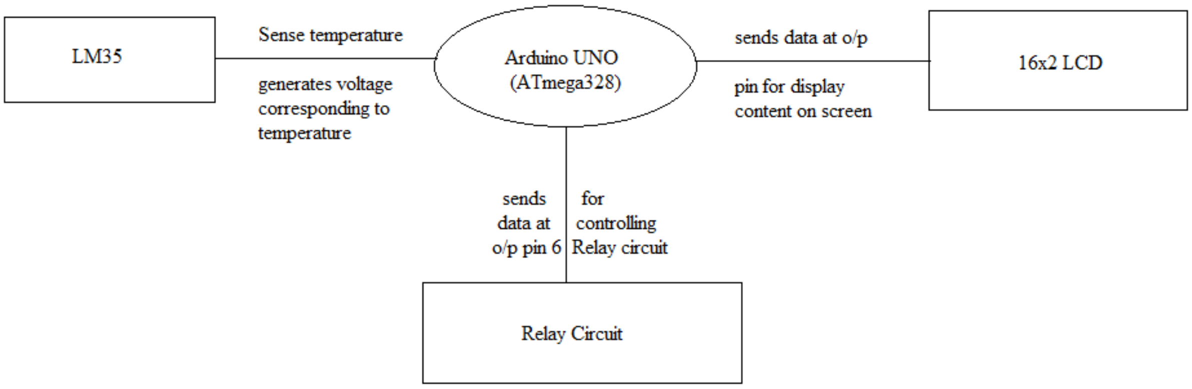

LM35
Sense temperature
generates voltage corresponding to temperature
Arduino UNO (ATmega328)
sends data at o/p
pin for display content on screen
16x2 LCD
sends data at o/p pin 6
for controlling Relay circuit
Relay Circuit

Fig. 3.2 :- Data flow diagram

CHAPTER – 4

IMPLEMENTATION

4.1 Hardware Implementation

In this section we design our project temperature monitoring and controlling. We used temperature sensor IC LM35. LM35 generates a small voltage corresponds to the temperature across IC. This generated voltage is in the continuous, analog form.

This voltage is fed to the controller unit. Here we use Arduino (ATmega8) as a controller. This voltage is given to the Analog port 0 (A0) of the Arduino UNO. Arduino UNO reads analog input and converts this analog voltage into digital bits form using inbuilt A to D converter. It converts analog voltage level in any number between 0 to 1023. It use 10 bits for processing. This is given to the ATmega328 microcontroller , it multiplies the digital data with coefficient 0.488 and converts this voltage in particular value. This value is nothing but the temperature in degree centigrade. Similarly, we multiplies the data with 1.8 and the add 32 to convert voltage level into Fahrenheit unit. This data sending out on the digital port (2,3,4,5). Consequently, Arduino uno sends out data for displaying on 16x2 LCD. Corresponding program for working of project is given in section 4.1.

An 16X2 LCD is connected with microcontroller as shown in circuit diagram (fig.3.1). Pin 1 is connected to ground and pin 2 is connected to vcc through Arduino for activating or switching ON the LCD. On pin 3 a 10k ohms for adjusting the brightness of LCD screen. RS and Enable pin is connected to pin 12 and 11 respectively for communication with Arduino. LCD displays temperature in both units degree Celsius and also in degree Fahrenheit. At the same time Arduino sends control bit 0 or 1 on the digital port 6. This bit is used for controlling part. For bit 0 Arduino sends 0V and for bit 1 it sends 5V at the output. A relay is connected with relay circuit to digital port 6. If the temperature is less than desired or set temperature Arduino gives logic low level to the pin 6. But when current temperature goes just or more above the set level it sends logic high level to the digital pin 6.

According to the logic level of digital pin 6 relay circuit gets input. According this relay circuit switches ON/OFF the relay. Consequently, the cooling device connected to relay also turns ON/OFF respectively. Once, temperature goes below the set point, relay switches off. Thus temperature gets monitored and controlled by this project.

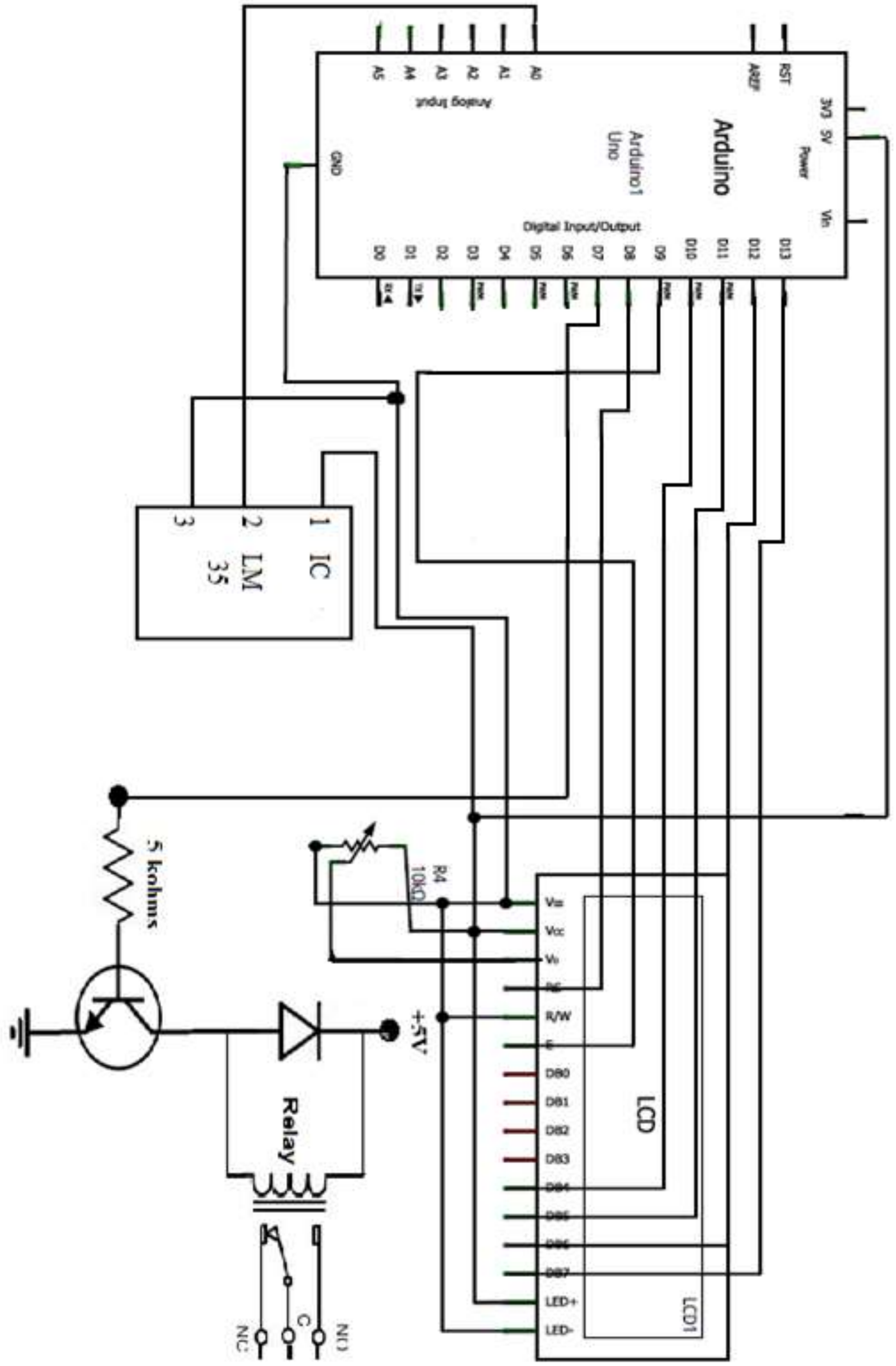
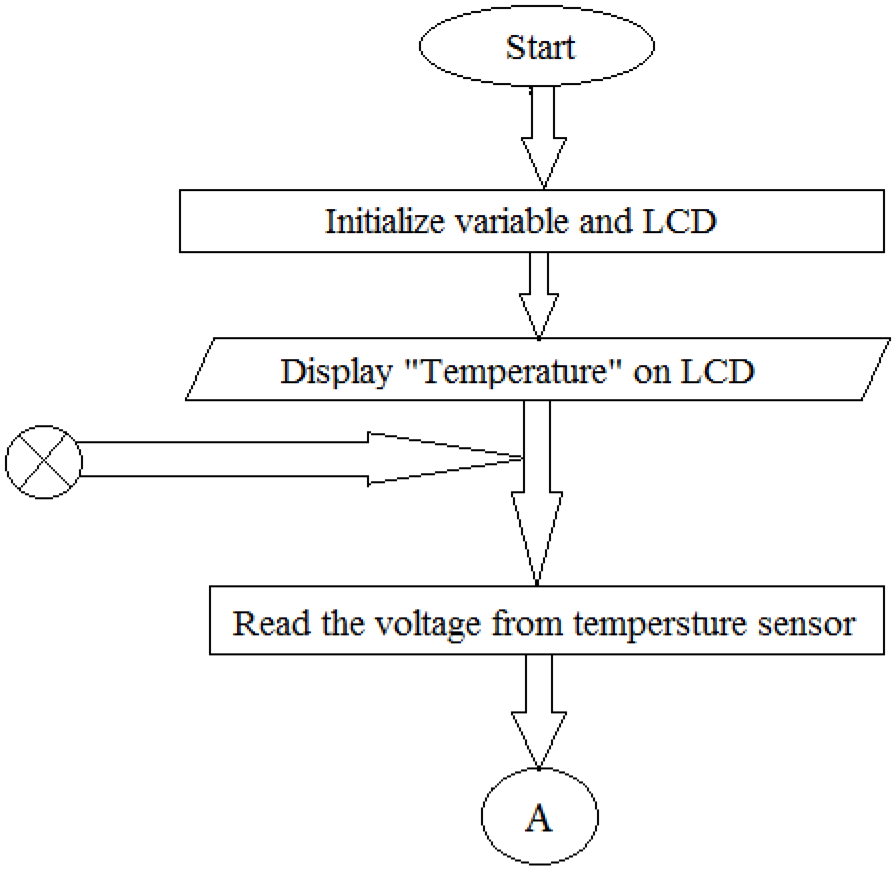


Fig 4.1 :- Circuit diagram of temperature monitoring and controlling system

4.2 Software Implementation

For software implementation we require a software Arduino IDE. This software enables us to load the program in Arduino board. Information about Arduino IDE is given in section 2.2.1.

4.2.1 Flowchart


Start
Initialize variable and LCD
Display "Temperature" on LCD
Read the voltage from tempersture sensor
A

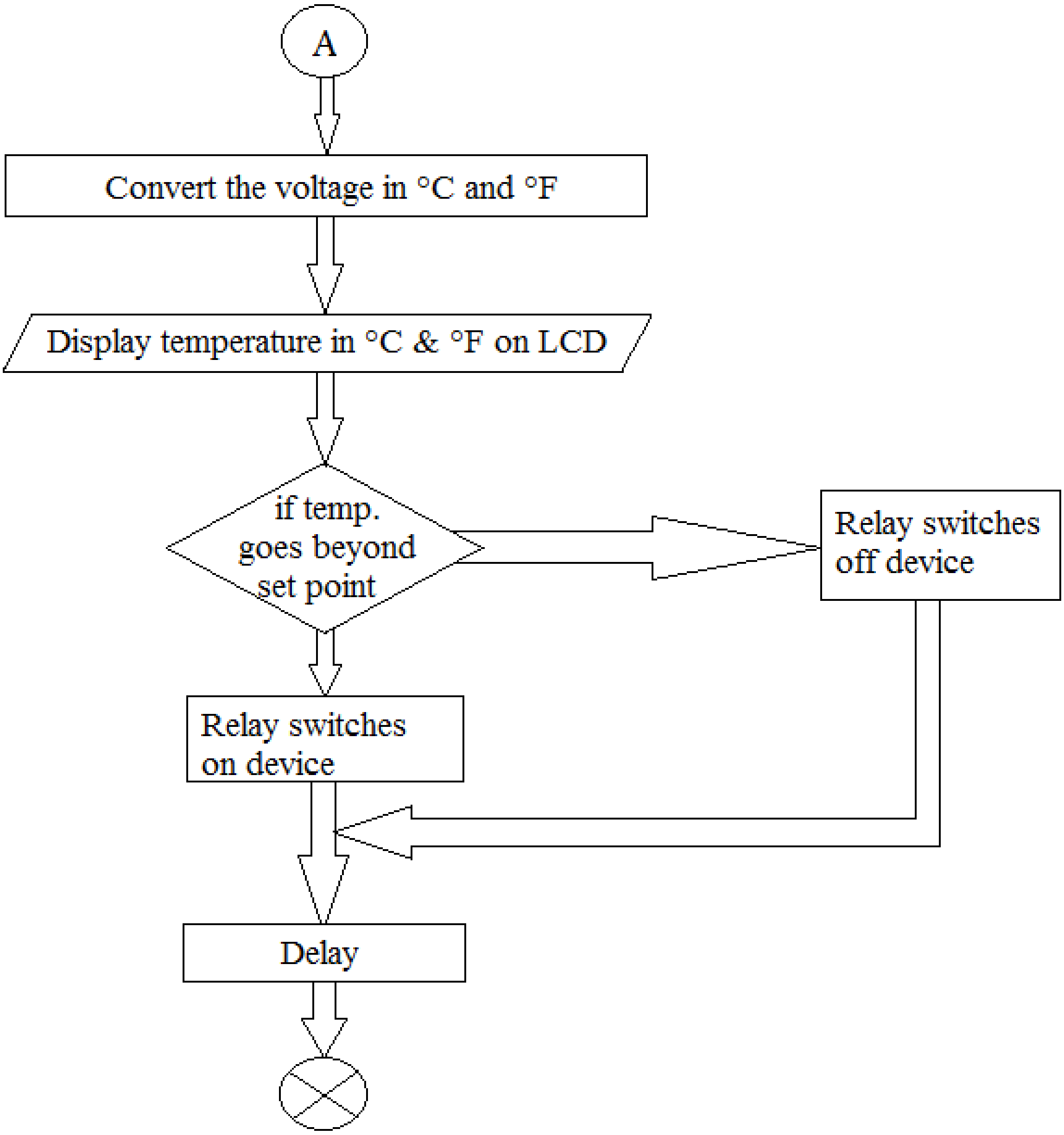

A
Convert the voltage in °C and °F
Display temperature in °C & °F on LCD
if temp. goes beyond set point
Relay switches off device
Relay switches on device
Delay

Fig 4.2.1 :- Flowchart

4.2.2 Source Code

```
#include <LiquidCrystal.h>    // Include the library to use a LCD display

#define tempSensor 0 // Define the A0 pin as "sensor"

int Vin;     // Variable to read the value from the Arduino's pin A0

float TC;  // Variable that receives the converted voltage value to ºCtemperature

float TF;  // Variable to receive the converted value from ºC to ºF

LiquidCrystal lcd( 8, 9, 10, 11, 12, 13);
```

```
/* The function above declares which Arduino’s pins will be used for controlling the LCD */
void setup()
{
  lcd.begin(16, 2);                // It tells the Arduino that the display is a 16x2 type
  lcd.print("*Temperature*");  // Send the text to the screen of the display.
  pinMode(7,OUTPUT);
}
void loop()
{
  Vin = analogRead (tempSensor); /* Tells the Arduino to read A0 and stores the value in
“Vin” */
  TC=(500.0*Vin)/1023.0; /* Converts the voltage value into temperature and stores it into
the “TC” variable (in ºC)*/
  TF = ((9.0*TC)/5.0)+32.0;     // Converts ºC to ºF
  lcd.setCursor(0, 1);             // Moves the cursor of the display to the next line
  lcd.print(TC);                   // Exhibits the value of the temperature on the display
  lcd.print((char)223);     //degree symbol
  lcd.print("C ");          // Writes “F” to indicate that it is in Fahrenheit scale.
  lcd.print (TF);
  lcd.print((char)223);  // degree symbol
  lcd.print("F ");        // Writes “C” to indicate that it is in Centigrade scale.
  if(TC < 32.00){
   digitalWrite(7,LOW);
  }else{
    digitalWrite(7,HIGH);
   }
  delay(1000);        // Waits for one second to read the sensor pin again
}
```

CHAPTER – 5

SYSTEM TESTING

5.1 Test Approach

The project “temperature monitoring and controlling” is made as explained in above chapters. It is necessary to check the system is working properly or not. It can be tested in two method. The system should displays the current temperature and if it goes above set point it controls it and bring back into desired level. The system should controls the temperature. It should operate properly in the given operating range. If the system does not work properly then it is necessary to troubleshoot the problem. The system should complete all the objectives.

5.2 Test Plan

For testing the project we make two parts. First part is used to check the program, in this step we check the program is working properly or not. It is done by using Arduino IDE. Second part is used to check hardware component like LCD and Relay.

5.2.1 Features To Be Tested

After building the whole circuit we test it, testing procedure is given in section 5.3.3. This project should satisfy some features. Features to be tested as follows :-

a) LM35 should detect temperature properly.

b) Arduino should give required (according to program) outputs on output port.

c) LCD should display the real time temperature.

d) Relay should be OFF below set point and it should be ON above set point.

e) System should be stable and work proper in the given range of temperature.

5.2.2 Testing Tools And Environment

For testing of the project we require some tools, like to test Arduino program we require a software called Arduino IDE using this we can check the program that program is working properly or not. For hardware checking we require power supply and proper range of temperature. Testing environment for project is important criteria as the project is based on the environment temperature. The environment temperature should be in the range of - 40 to 85 degrees Celsius. While for only IC LM 35 range is −55°C to 150°C.

5.3 Test Cases

In this section we discuss about the inputs, expected output, testing procedure. Testing tools required for the circuit is explained above.

5.3.1 Inputs

The project requires two inputs. The inputs are as follows :-

a) Power supply :-

Power supply is the basic need of any electronic circuit. Here we use 5v dc battery to give power Arduino and sometimes we can give power directly from the computer.

b) Temperature :-

As we know that the project is mainly based on the temperature. It uses environmental temperature as input and controls the relay circuit.

5.3.2 Expected Output

After building the whole circuit we test it, testing procedure is given in section 5.3.3. This project should give some desired outputs. Expected outputs are as follows :-

a) LCD should display the real time temperature.

b) Relay should be OFF below set point.

c) Relay should be ON above set point.

5.3.3 Testing Procedure

For testing first connect the circuit as shown in fig. 4.1. then power supply is given to the Arduino using computer and it can be done by using battery. In this way the whole testing circuit is built. Now we gives input to the LM35 as varying its surrounding temperature. Change in temperature should be shown on the 16x2 LCD screen. System should work according to program. Output of the circuit is observed.

Summary of testing procedure :-

1) Connect the circuit as shown in fig 4.1

2) Give power to the system.

3) Vary temperature across LM35.

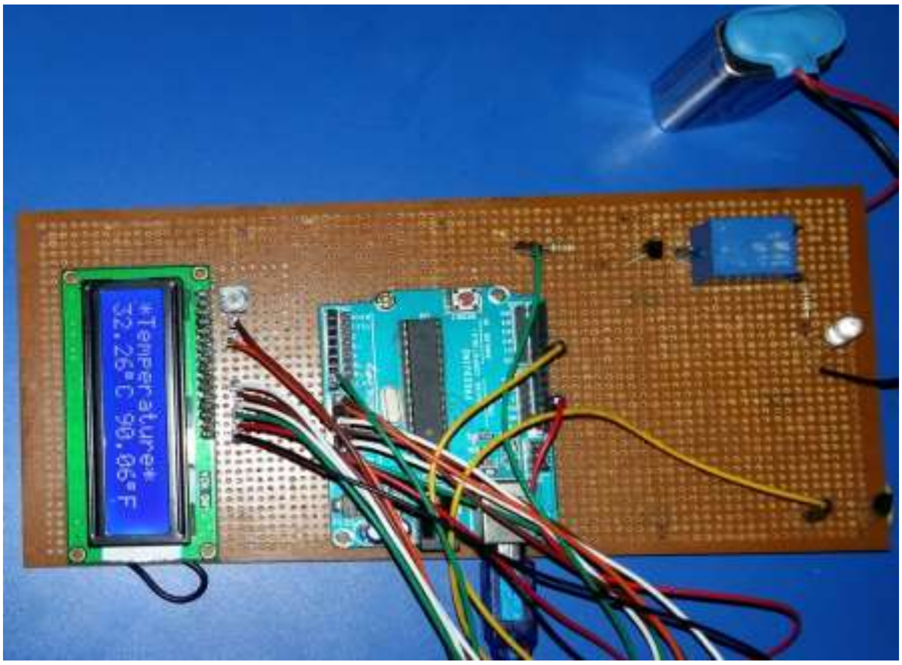
4) Working of LCD and Relay is observed.

CHAPTER – 6

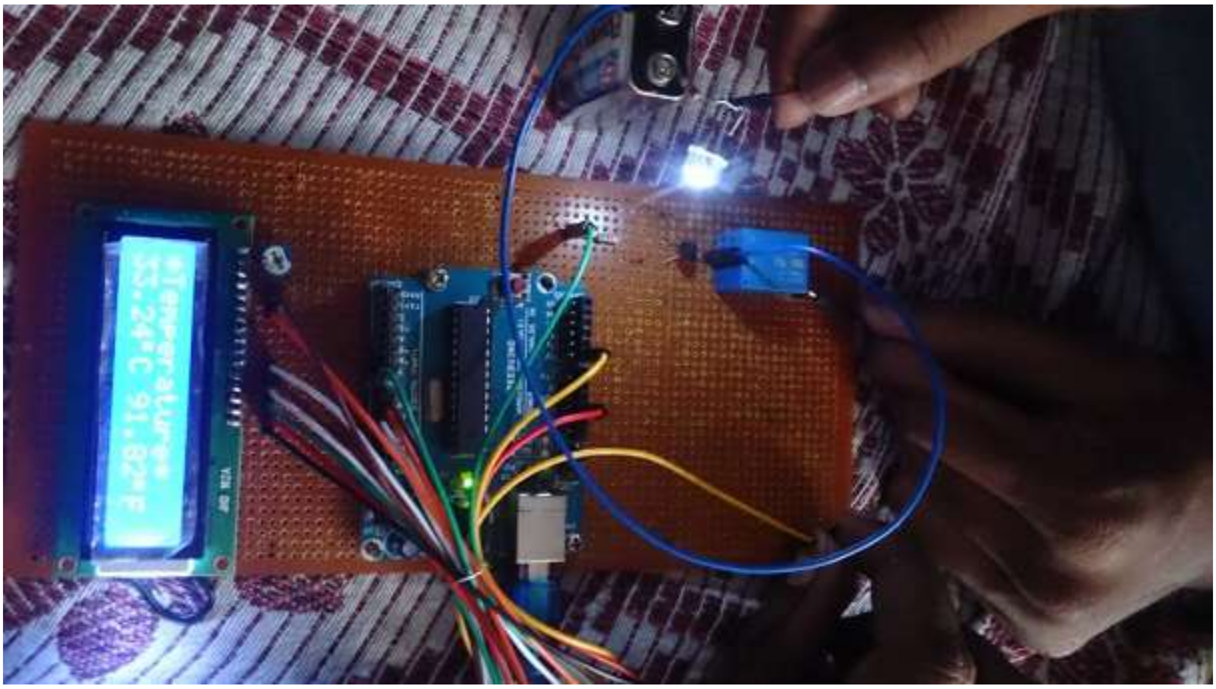
RESULTS

As project names Temperature monitoring and controlling the project gives two outputs. One is displaying temperature on LCD screen. And second on is given to relay, relay switches ON and OFF the device connected across it. Here, we used LED to show the output of relay.

a) When temperature is below set point (i.e. 33 degree celsius) the relay is off. Consequently, LED connected across it is off.



b) When temperature goes above set point (i.e.33 degree celsius) the relay is ON. Consequently, LED connected across it glows.



CHAPTER – 7

CONCLUSION

7.3 Conclusion

In our project, we designed and implemented an efficient temperature monitoring and controlling system with an Arduino board. Output was verified by setting the temperature at different levels and it was found that the led turn on and off when the device crosses the set value. It is very useful for the people who are disable. There is still much room for future development that would enhance the system and increase its business value.

7.1 Advantages

- This project can be used in Industry as well as in Home.
- To monitor the environments that is not comfortable, or possible, for humans to monitor, especially for extended periods of time.
- Prevents waste of energy when it’s not hot enough for a fan to be needed.
- To assist people who are disabled to switch on or off cooling device automatically.

7.2 Disadvantages

- It can only be maintained by technical person. Thus, it becomes difficult to be maintained.
- Due to temperature variation, after sometimes its efficiency may decrease.
- Comparatively it is costly.

.

CHAPTER – 8

FUTURE ENHANCEMENTS

Arduino based temperature controller is a simple whereas a useful circuit with which the temperature can be controlled with the aid of a LM 35 temperature sensor. As explained the circuit can be made useful in practical area where the circuit can be connected to a device whose temperature has to be controlled at a particular limit say a water tank with a heater whose temperature can be set to a particular value. Similar another application is that the circuit attached with a buzzer which can be connected to a device like an iron box so that it would help to save electricity by avoiding overheating of the device.

In future the circuit can be enhanced by connecting a GSM Module to the circuit so that in industrial area when a machine crosses the set temperature, we can inform the control room by sending a message, or else a call to control room manager so that damages to the machine can be avoided by disconnecting the equipment with GSM technology.

CHAPTER – 9

REFERENCES

Books,

[1] Jeremy Blum, 2013, Exploring Arduino: Tools and Techniques for Engineering Wizardry, Wiley PUB.

[2] Sharam, M.C., 1986, Relay And Their Application, Delhi, BPB PUB.

[3] Chaudhary, D. Roy, 2003, Linear Integrated Circuit, Second Edition, New Delhi, New Age International PUB.

[4] Massimo Banzi, 2011, Getting Started with Arduino, 2nd Edition, UK, Maker Media, Inc.

Catalogs ,

[1] TEXAS - LINEAR IC's manual

[2] OMRON CORPORATION's manual

[3] VISHAY INTERTECHNOLOGY, INC's manual