ABSTRACT

Temperature control refers to the processes that are aimed at maintaining the temperature in a given area at certain maximum/minimum level or within a certain range. This process is commonly used in most areas of the world. Recently, globalization and industrialization has further necessitated the need for Temperature Control applications in various daily activities, especially with the advent of the green house effect.

Many Homes and Industries among other areas maintain certain sections of operation that must be maintained within a certain temperature for process to work successfully. In research laboratories, the lack of use of Temperature Control Systems has lead to the purchase of chambers of various sizes where temperature specific research work would be kept. This has also lead to an increase in overhead cost. In areas that have electronic activities or machinery functioning constantly, such as in server rooms and production plants. These are places where heavy machinery and computers work continuously 24 hours every day. During these processes, the temperature needs to be monitored frequently in order to ensure that it doesn't rise or fall below a value that would accelerate wearing out of the systems.

It is important also to monitor the level of temperature various other places such as morgues, hospitals, aircrafts, living rooms, etc, to ensure that **thermal comfort** is maintained. Thermal comfort is generally defined as that condition of mind or functionality which expresses satisfaction with the thermal environment (e.g. in ISO 1984). Dissatisfaction may be caused by the body / equipment being too warm or cold as a whole, or by unwanted heating or cooling of a particular part of the body (local (functional) discomfort).

Automatic temperature control is certified as the best method in any application because the temperature is usually controlled automatically (no human intervention involved) throughout the process. The results obtained from various applications of the process across

different regions and timelines shows the temperature is controlled effectively and more accurately. In addition, this finding also makes human work easier as an automatically controlled system worries about other contingent weather issues for you.

The major objective of this project would be to create a Temperature Control System that would be able to automatically control the temperature of the environment it is placed in by the timely activation of the effector devices to influence the temperature in relation to the set-point.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER ONE

BACKGROUNG OF THE STUDY

1.1 WHAT IS A TEMPERATURE CONTROL SYSTEM?

Temperature: This is the degree of hotness or coldness of a body or an environment.

Control System: A control system is a device or set of devices that manage, command, direct or regulate the behaviour of other devices or systems.

Thus we can literally say that a Temperature Control System is a device or set of devices that manage, command, direct or regulate the behaviour of other devices or systems in order to influence the degree of hotness or coldness of a body or an environment.

A Temperature Control System is a more like a **programmable thermostat** that can keep the environment (home or office) at a desired temperature regardless of fluctuating exterior weather conditions. The advantage of having a temperature control system over a common thermostat is that it saves energy and money by automatically maintaining different temperatures at different times of the day and night. It is usually a feedback system having a control loop, including sensors, control algorithms and actuators/effectors, and is arranged in such a fashion as to try to regulate a variable at a set point or reference value. An example of this may increase the fuel supply to a furnace when a measured temperature drops.

A programmable thermostat is a digital device that replaces the regular (automatic) thermostat located in older homes and apartments. A thermostat measures the temperature of a room, turning the heating / cooling unit on or off in order to maintain the setting indicated on the thermostat. One of the drawbacks of the traditional thermostat is that it is commonly left at a single setting out of sheer convenience. This translates to higher energy bills because the home is kept warmer than required when people are tucked away in bed, or even off at work (when it is not needed). What would be far more efficient is to have a thermostat that knows when you need it and when you don’t, so that it could vary the temperature and save energy. This is exactly what a Temperature Control System offers.

A temperature control system consists of a small programmable digital logic controller device, wired to a heating and/or cooling system. About the size of a typical wall-mounted thermostat, a temperature control system contains a small circuit board and a memory chip(s). After setting the temperature control system to a desired temperature, known as a ***set point,*** the system will utilize the heater and/or air conditioning unit (as needed) as **effectors**, to maintain that setting for the duration programmed.

A Programmable Logic Controller (Micro-controller) is an electronic device used for automation of industrial processes, such as control of machinery on factory assembly lines. It is an example of a real time system since output results must be produced in response to input conditions within a bounded time. It can thus be said to be a collection of relays in series.

Let’s consider this instance. In winter months you might like your home heated to 71 degrees Fahrenheit (21.6 Celsius) in the mornings when rising. If the house is empty during the day there is no need to maintain this temperature and the temperature control system can be set to allow it to naturally fall to another preset temperature. It can be preset to kick back on about 30 minutes before you or other family members normally arrive home. When the household sleeps, the temperature control system can maintain a cooler setting, warming up just before the hous|435©ehold awakens. All of these various temperatures and times or set points are preset by the user to automate the process without having to manually adjust the thermostat.

1.2 HISTORY OF TEMPERATURE CONTROL SYSTEMS

The use of Automatic Temperature Control Systems began way back in the 18th Century. The idea was conceived by Warren S. Johnson while he was teaching at Norman School, Oklahoma. Before then, Janitors had to enter each classroom to determine if it was too hot or too cold, and then adjust the dampers in the basement accordingly. Johnson sought a way to end, or at least minimize the classroom interruptions of the janitors and increase the comfort level of the students. The Automatic Temperature Control System was to meet this very need.

In 1883 Warren Johnson gave up teaching to fully devote his time to researching and developing his ideas. He moved to Milwaukee and formed the Johnson Electric Service Company in 1885. In 1895, Johnson patented the pneumatic temperature control system. This allowed for temperature control on a room by room basis in buildings and homes. It was the first such device of its kind. By the early 20th century the Automatic Temperature Control System was being used in many notable places including the New York Stock Exchange, Palaces of Spain and Japan, West Point, the Smithsonian, the US Capitol Building, and the home of Andrew Carnegie. The use of this system has increased continuously to this day.

1.3 WHY DO WE NEED A TEMPERATURE CONTROL SYSTEM?

The 21st Century was greeted with very unpredictable and unfavourable temperature conditions. The Green House effect has left our world exposed and this resulted in a lot of uncertainties in our weather conditions and climate generally. There has been a growing need for the temperature of certain areas to be kept within a certain range. This has necessitated the need for Temperature Control Systems:

IN THE HOMES: In many modern day homes, the wastage margin of food stuff has increased greatly. This is due to the fact that the temperature of the storage area of the home rose above or fell below a certain allowable maximum or minimum value respectively, leading to the accelerated decay of the food materials.

In addition to this, some areas of the home have to be regulated within certain habitable temperatures (i.e. not too high and not too low). This ensures that life processes can be carried out by people conveniently in those areas.

IN THE INDUSTRIES: Many Industries (especially Manufacturing and Pharmaceutical Industries) have growing concerns for the need to store certain production materials within a specific temperature range. Some of these materials could be highly inflammable or explosive at certain extreme temperatures. This necessitates the need for a Temperature Control system.

IN MORGUES: In morgues and mortuaries, dead bodies have to be preserved at a certain temperature to prevent them from accelerated decay. This temperature must be monitored and maintained regardless of the presence/absence of mortuary staff, and it also has to be managed in such an efficient manner that it doesn't generate enormous energy bills for the management. This problem also necessitates the need for a Temperature Control System.

IN JETS AND AIRCRAFTS: Aircrafts are an important area where safety of passengers is mainly guaranteed by the efficient management and regulation of weather elements such as temperature, air pressure and humidity. These elements must be kept at a certain quantity / degree within the aircraft in order to sustain its weight. Practically, such weather elements as pressure and humidity are factors of adequate temperature. This also necessitates the need for a Temperature Control System.

NATIONAL ENERGY SUPPLY: In developed countries, the dream to conserve enough energy for future use has gradually become a nightmare owing to a decline in the use or lack of use of Temperature Control Systems. In many homes, offices and industries, many heating and cooling devices are accidentally left functioning even when there is no need of them. Occasionally, these mistakes have resulted in municipal infernos that have destroyed lots of lives and properties. In underdeveloped countries the governments are being buried beneath extreme debts of energy bills because of wastage of energy resources.

By using a Temperature Control System you never have to worry about wasting money or electrical energy by forgetting to turn the air conditioning or heating unit off. This greatly optimizes the cost of production in Industrial processes and the cost of living in Homes. Also, you never have to worry about the temperature at which your living or storage area must be maintained. Just let the Temperature Control System worry about that for you. Programming the system only takes a few minutes, and weekends can have separate set points to accommodate alternate schedules (in more deluxe systems). It's also easy to override the set point with the touch of a button, in case you want the area to temporarily be warmer or cooler at any time.

1.4 OBJECTIVE OF THE PROJECT

The main objective of this project is to design a Temperature Control System that helps to optimize Costs of production and living both in the homes and industries. It also serves to eliminate hazards that result from the accidental neglect of heating and cooling appliances in the homes and industries, even when they are not needed. To achieve this, a highly sensitive Temperature sensor detects the current temperature and feeds it as input to the Micro-controller. The Micro-controller then initiates a sequence of control procedures based on the configuration of the control program it contains. These control procedures would include: turning on/off a heating or cooling system and activating a buzzer/alarm unit.

1.5 SCOPE OF THE PROJECT

Owing to inevitable constraints of time and finance, the scope of the project for the purpose of this research work would be limited to its home application only. The Temperature Control System would detect the temperature changes within the home environment and regulate it by triggering the appropriate equipments to influence the temperature. To successfully implement a Temperature Control System of this Capacity, knowledge about the following is needed:

1. Knowledge about the output voltages from the temperature sensor and how to convert them to byte values.
2. The particular formula that will be used to convert the byte values to Centigrade scalar (for LCD display).
3. The programming of the Micro-controller and development of the Control Program.
4. Driver circuit operation and function of each component

1.6 TEMPERATURE CONTROL SYSTEM TERMINOLOGIES

I. Controlled Variable: This refers to what is being controlled by the Control System. In this case, it is the temperature.

II. Sensor: This is the device or unit that measures/detects the temperature of the environment at a given time.

III. Effectors: These refer to the output/control element and related devices or units that are used to affect the temperature of the environment (It could be a heating/cooling unit). In this case, it is the alarm unit.

IV. Controller: The Controller is the device that processes the temperature reading from the sensor and uses the results generated to activate the appropriate effectors. In this case, it is a microcontroller.

V. Set-point: This is used as a reference point for the Controller. It is set or input by an external operator. The Controller compares the readings received from the sensor with this reference point in order to determine which effector is appropriate.

VI. Thermostat Function: The processes that involves comparing the current temperature status received by the sensor with the set-point

VII. Temperature Breach: Any instance in which the set-point has been compromised.

1.7 BASIC COMPONENTS OF A TEMPERATURE CONTROL SYSTEM

Temperature Control Systems down through the years have been made up of the following five major units:

I. The Power Supply Unit: This Unit provides the Temperature Control System with the Electrical Energy that drives it. In this case, the Power Supply Unit consists of a Step-down transformer which works based on the principle of induction. The transformer steps down the voltage received from the power outlet from the national rating of 230V to 15V, which is all the voltage needed to drive the system. This voltage is further rectified (using a bridge rectifier) and filtered (using a power capacitor) to give a perfect and undistorted voltage to the system. Of this 15V input voltage, about 5V drives the microcontroller. The rest are needed to drive the other units of the circuit.

II. The Sensor Unit: This Module consists of devices (thermometers in traditional systems) that detect the current temperature status. These devices sense the current room/surface temperature, and provide its result to be used as input in the Control unit and in the Display Unit.

III. The LCD/Display Unit: This displays the current temperature status of the environment as received from the Sensor Unit. In this case it consists of a 7-bit graphic large-digit display device that reveals the results/reading of the temperature sensor to the external user.

IV. The Control Unit: The Control unit houses the Controller and related devices (thermostats in automatic systems) that process information to produce effects/action by the system. In this case, this unit houses the microcontroller (and control program/algorithm) that stores the set-point temperature. The control program receives temperature status from the sensor unit and ensures that it doesn't compromise the set-point by initiating the appropriate sequence of action(s).

V. The Menu/Function Unit: This unit consists of input buttons that are used to give commands to the control program and also to program the set-point for the system. In this case, a variable resistor which changes the set-point temperature when its resistance is varied.

VI. The Alarm Unit: This unit consists of an alarm system that alerts the inhabitants of the environment of a temperature breach. This is an optional component of Temperature Control Systems. It comes mostly with those systems that are built to specifications (custom systems). Most commercial Temperature Control Systems prefer to maintain a silent profile in the environment where they function.

1.8 HOW DOES IT WORK?

To enable us successfully understand how a Temperature Control System Works, let's go back to the roots by taking a look at the first ever and the certified "most efficient" Temperature Control System that was ever made – The Human Body Temperature Control System.

The body regulates its temperature continuously. It may increase or decrease its temperature when it finds that it is too cold or too hot. In this case, temperature is being regulated by a **control system**, and the control is called **homeostasis**. Somewhere in the brain, perhaps the Hypothalamus, the optimum temperature of the body (**set point**) is stored (about 37°C). That information is continuously available to some structure, we call the comparator. The comparator sends signals to:

1. Heat gain mechanisms in the pre-optic area or anterior hypothalamus leading to:
    - Shivering
    - increased thyroid hormone output
    - increased activity in the sympathetic nervous system
    - piloerection
    - cutaneous vasoconstriction
2. Heat loss mechanisms in the posterior hypothalamus leading to:
    - decreased thyroid hormone output
    - sweating
    - cutaneous vasodilation

The output of these mechanisms will end as either a net increase or a net decrease in body temperature. The body temperature is sensed by thermal receptors (thermo-receptors) in the brain and peripherally in the body, and the value is sent to the comparator where it is compared with the set point. If the value is less than the set point, then signals go mainly to the heat gain mechanisms; if it is greater than the set point, then they go mainly to the heat loss mechanisms. In this way, body temperature is constantly sensed and maintained constant (i.e., homeostasis).

The block diagram of the system for the control of body temperature is given below:

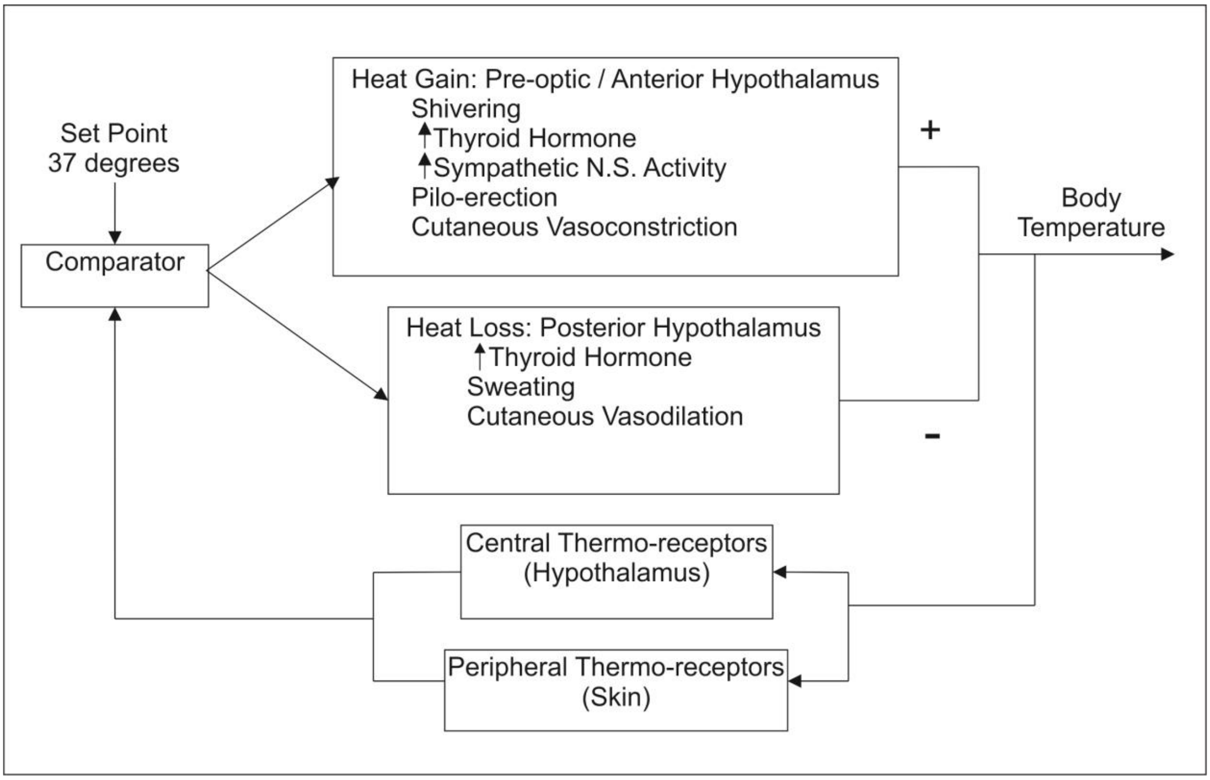


FIGURE 1.1 DIAGRAM SHOWING THE PROCESS OF HOMEOSTASIS

From this we can deduce that in order to accurately control process temperature without extensive operator involvement, a temperature control system relies upon a Control Unit, which accepts the reading of a temperature sensor such as a thermocouple or RTD as input. The set point is programmed into the Control Unit using the Menu/Function Unit, and the start button is pressed. Once the start button is pressed, the Thermostat Function is initiated. The Sensor monitors the current external temperature status and sends its reading to the

Control Unit. The temperature reading is also displayed on the Display Unit. In the Control Unit, the control program constantly compares the temperature reading with the set-point to ensure that the reading doesn’t breach the set-point. In the event of a Temperature breach, the Control Unit sends off a signal to trigger on the Alarm and inform the individuals present of the breach. Depending on the complexity and robustness of the system, the Control Unit with the aid of the Control Program/Algorithm then determines which sequence of actions would be most appropriate to correct the breach; these are then sent off as interrupts to the appropriate effectors. Sequence of Actions in this case would include either turning on/off the heater or turning on/off the cooling system or other installed control elements. However, the Control Unit/Controller is just one part of the entire control system; in selecting an appropriate controller, the following items should be considered:

- Type of input temperature sensor (thermocouple, thermistor, RTD) and temperature range
- Type of output required (electromechanical relay, SSR, or analog output)
- Control Algorithm (On/Off, proportional, or PID (proportional–integral–derivative))
- The number and type of outputs (heating system, cooling system, alarm system and limit)

There are three types of Controller / Control Algorithms for use in the construction and design of most Temperature Control Systems. These include:

A. The On/Off Control – an on/off controller is the simplest form of temperature control device. The output from the device is either on or off, with no middle state. An on-off controller will switch the output only when the temperature crosses the set-point. For heating control, the output is on when the temperature is below the set-point and off above the set-point. Since the temperature crosses the set-point to change the output state, the process temperature will be cycling continually, going from below the set-point to above, and back below. In cases where this cycling occurs rapidly, and to prevent damage to contactors and valves, and on-off differential, or “hysteresis” is added to the controller operations. This differential requires that the temperature exceed the set-point by a certain amount before the output will turn off or on again. On-off differential prevents the output from “chattering” or making fast, continual switches if the cycling above and below the set-point occurs very rapidly. On-off control is usually used where a precise control is not necessary such as in systems which cannot handle having the energy turned on and off frequently, where the mass of the system is so great that temperatures change extremely slowly, or for a temperature alarm. One special type of on-off control used for alarm is a limit controller. This controller uses a latching relay, which can be manually reset, and is used to shut down a process when a certain temperature is reached.

B. Proportional Control – proportional controls are designed to eliminate the cycling associated with the on/off control. A proportional controller decreases the average power supplied to the effector as the temperature approaches the set-point. This has the effect of slowing down the heater/cooler so that it will not overshoot the set-point, but will approach the set-point and maintain a stable temperature. This proportioning action can be accomplished by turning the effectors on/off for short time intervals. This “time proportioning” varies the ratio of “on” to “off” time to control the temperature. The proportioning action occurs within a “proportional band” around the set-point temperature. Outside this band, the controller functions as an on-off unit, with the output either fully on (below the band) or fully off (above the band). However, within the band, the output is turned on and off in the ratio of the measurement difference from the set-point. At the set-point (the midpoint of the proportional band), the output on: off ratio is 1:1; that is, the on-time and the off-time are equal. If the temperature is further from the set-point, the on-and-off times vary in proportion to the temperature difference. However, if the temperature is below the set-point, the output will be on longer; if the temperature is too high, the output will be off longer.

C. PID Control (proportional–integral–derivative controller) – The third controller type provides proportional with integral and derivative control, or PID. This controller combines proportional control with two additional adjustments, which helps the unit to automatically compensate for changes in the system. These adjustments, integral and derivative, are expressed in time-based units; they are also referred to by their reciprocals, RESET and RATE, respectively. The proportional, integral and derivative terms must be individually adjusted or "tuned" to a particular system using trial and error. It provides the most accurate and stable control of the three controller types, and is best used in systems which have a relatively small mass, those which react quickly to changes in the energy added to the process. It is recommended in systems where the load changes often and the controller is expected to compensate automatically due to frequent changes in the set-point, the amount of energy available, or the mass to be controlled. Some other controllers exist which are designed to automatically tune themselves. These are known as **auto-tune controllers**.

1.9 PROJECT METHODOLOGY

The research methodology included the following steps:

I. Study of previous literatures on the project to better understand the concept and functionality of the project.

II. Understanding the whole system of hardware and software sequences.

III. Designing the system circuit and developing the control algorithm.

IV. Testing the functionality of the various sections of the system.

V. Combining the both hardware and software components of the system.

VI. Documenting the Research/Project

The steps above are further explained using the flowchart below:

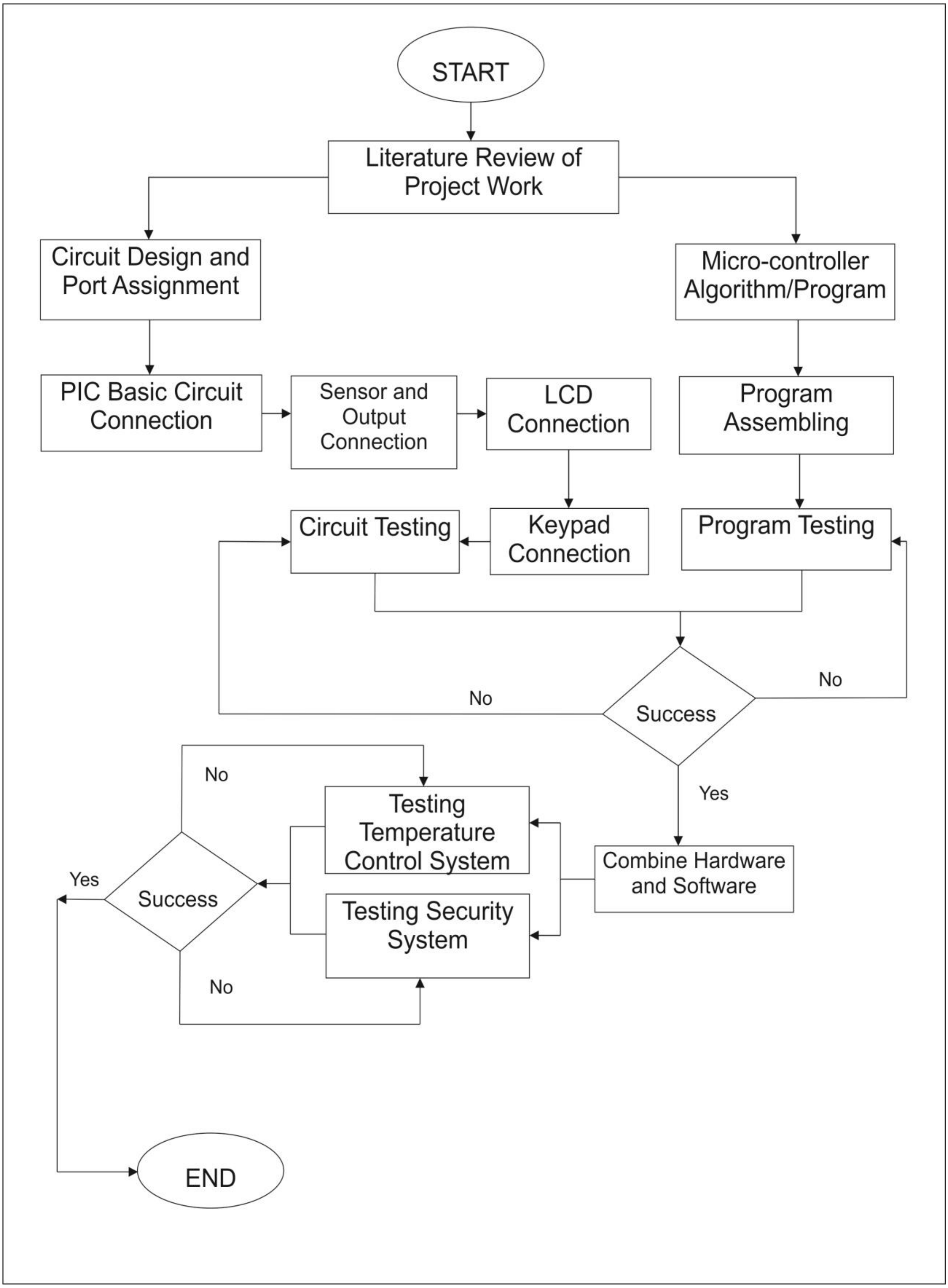


FIGURE 1.2 FLOWCHART OF PROJECT METHODOLOGY

1.10 PROJECT CHANNEL

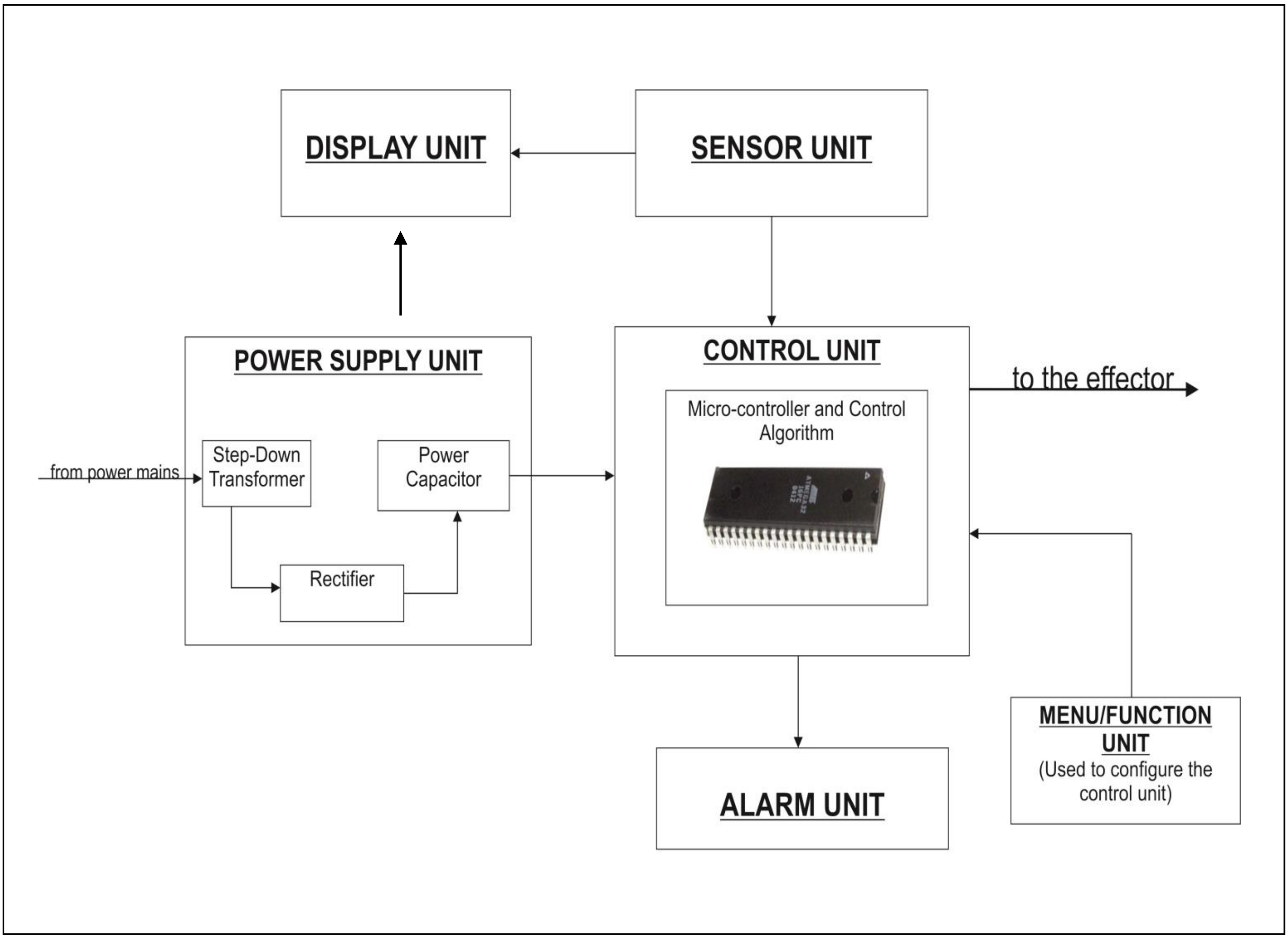


FIGURE 1.3 PROJECT CHANNEL FOR A TYPICAL TEMPERATURE CONTROL SYSTEM

1.10.1 **PROJECT BLOCK DIAGRAM**

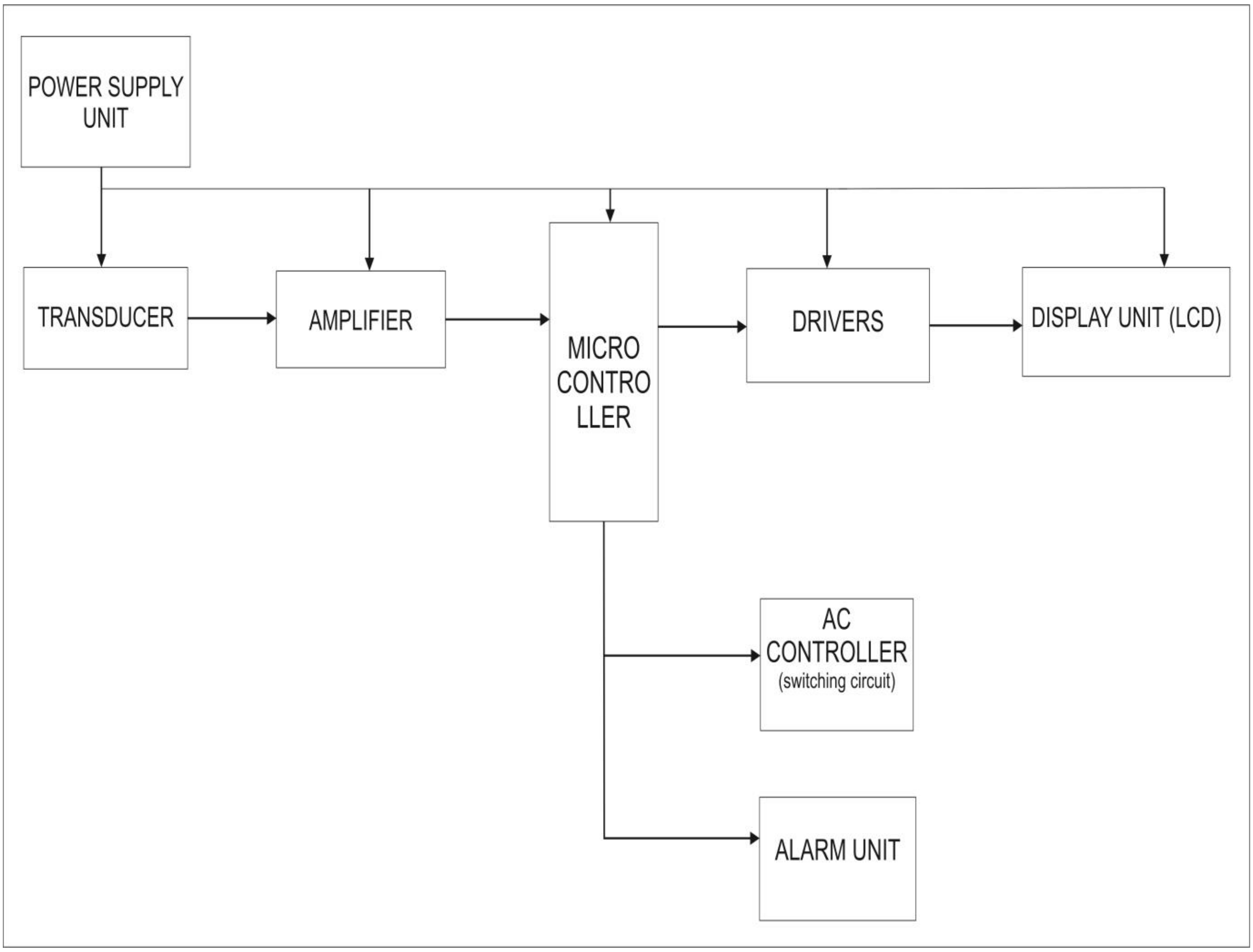


FIGURE 1.4 BLOCK DIAGRAM OF THE TEMPERATURE CONTROL SYSTEM

1.10.2 **PROJECT FLOWCHART**

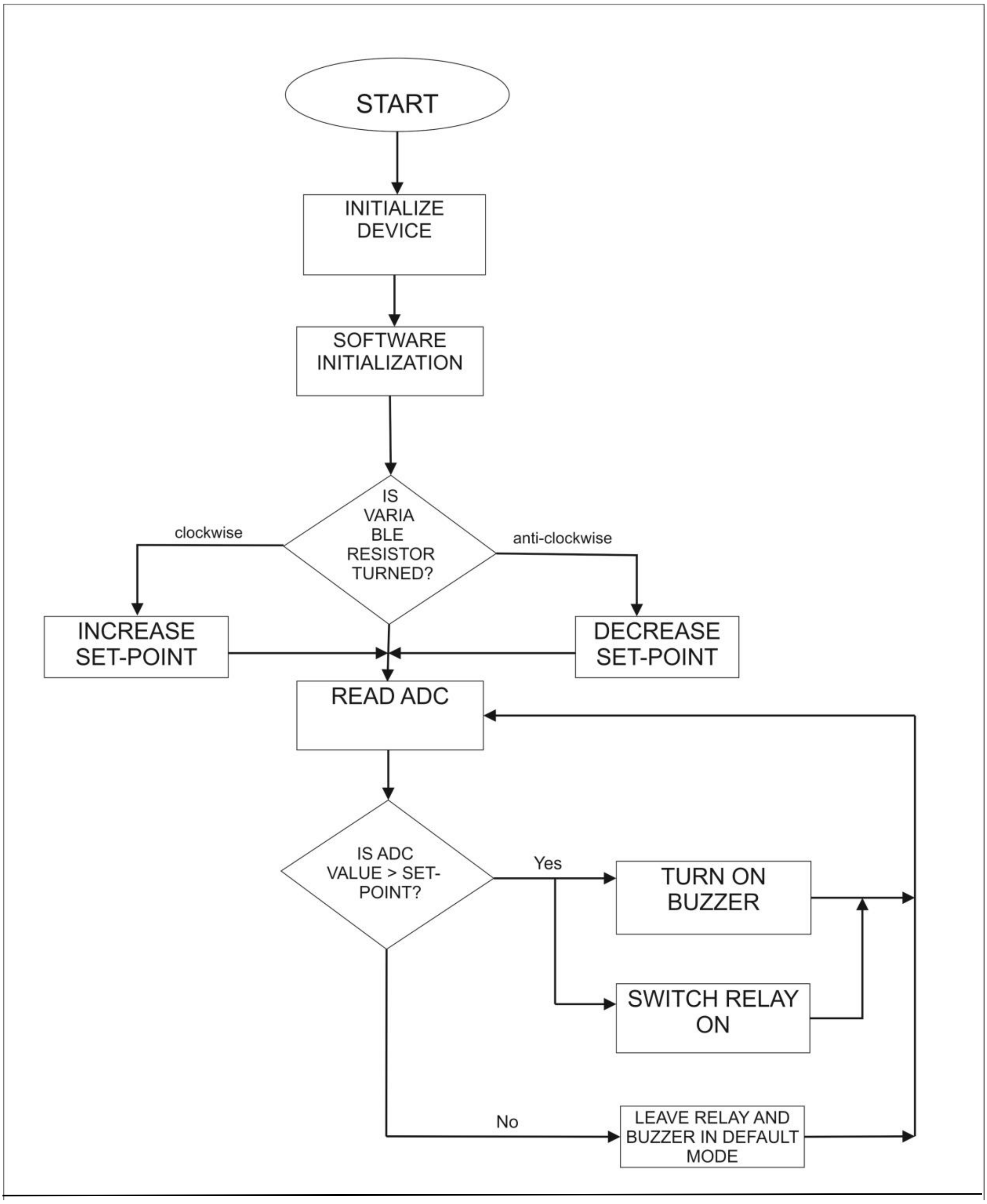


FIGURE 1.5 FLOWCHART OF THE TEMPERATURE CONTROL SYSTEM

1.11 IMPORTANCE OF A TEMPERATURE CONTROL SYSTEM

The Importance of a Temperature Control System can be grouped into three basic areas:

1. Economic Importance – Temperature Control Systems are economically important as they have the duty of ensuring that the energy supply reaching the house/office/industry is economized and managed as much as possible. This it achieves by making sure that the heating and cooling units are only functioning when they are needed. The Control System further ensures that the energy bills that would be paid by the company or individual are "efficient", because the bill would only cover energy that was efficiently utilized
2. Safety Objective – Temperature Control Systems have saved the day in many places where they are being used. Electrical Fires have been minimized because these Control Systems turn off heating and cooling units when they are not necessarily in use, thus preserving lives and property.
3. Wastage of Resources Prevention – In Homes and Industries alike, much wastage of useful resources have been prevented as a result of the use of Temperature Control Systems. These Systems efficiently manage the temperature of storage areas, thus keeping stored items at temperatures that extend their value period.

1.12 TEMPERATURE CONTROL SYSTEM LIMITATIONS

Most Temperature Control Systems in use today do not come with built-in heating and cooling systems. Therefore, they have to be connected to third-party heaters and cooling systems. Coping with the difficulties that are posed by some of these old legacy heaters and cooling systems and they connectors/configurations have been a major hindrance to the use of Temperature Control Systems. Most Temperature Control Systems have been bought by users, only to discover at installation point that the System cannot work efficiently with the existing heating and cooling system installations. This means that the existing heating and cooling systems have to be abandoned and completely uninstalled and replaced with more compatible systems.

The cost of Temperature Control Systems increases proportionally with the amount of operational flexibility and accuracy provided by the system. This also affects the installation; because the more sophisticated Temperature Control Systems are more difficult to install.

Temperature Control Systems have to be able to respond efficiently and adapt to rapid temperature swings. Temperature Swings are instances when the exterior temperature rises or falls rapidly (e.g. swings of the order of ±80°C/second) from the current state with a large difference margin. The Industrial Age has resulted in very unstable and unpredictable temperature conditions as a result of the green house effect. Modern Temperature Control Systems must be able to efficiently manage these unforeseen contingencies.

Temperature Control Systems these days are designed for use indoor use only or within small enclosed areas. This is due to the fact that the design of such systems is not robust enough to regulate the temperature of larger areas or outdoor environment as a result of excessive external thermal influences. This has been the major limitation to the use of Temperature Control Systems in our world.

1.13 ORGANIZATION OF PROJECT CHAPTERS

A. **Chapter One: (Introduction)** – This chapter briefly introduces the concepts, origin of Temperature Control Systems and Control Systems in general. It also outlines the objectives and scope of the project.

B. **Chapter Two: (Literature Review)** – In this chapter, previous Temperature Control Systems designed by various Engineers in the past would be reviewed and their opinions on the project would be properly documented, alongside their various design and implementation methodologies and techniques, in a bid to strike a comparison between the past and the current project timeline.

C. **Chapter Three: (System Analysis and Design)** – In this chapter, the various existing timelines of the project would be analyzed with an aim to more clearly define the project objectives and carve a niche for the project in the engineering world.

D. **Chapter Four: (Implementation and Documentation)** – This Chapter would include the complete implementation of the project work and a detailed documentation to guide the prospective clients of the project, alongside a detailed user guide to guide various users on how to work with the system.

E. **Chapter Five: (Summary, Conclusion and Recommendation) –** In This Chapter, being the final chapter, the entire project would be reviewed in a bid to represent it in a more concise manner, future milestones for the project would also be outlined, as well as recommendations on all aspects of the project work.

CHAPTER TWO

LITERATURE REVIEW

2.1 HOME ENVIRONMENT OVERVIEW

The Home is usually the most inhabited place in any society. The need to keep the home environment thermally conducive should be of paramount concern in any society that wants to maintain happy and healthy citizens. Areas in the home that are usually occupied by people, such as the living room and bedrooms need to be maintained within habitable temperature ranges. The human body has a set-point temperature of about 37°C. Extremely higher or lower temperatures can result in damage to some body organs or tissues and eventual death. These issues become more pertinent in areas of the home that are occupied by infants. Adults could possibly find their way around "thermal discomforts", but infants may not.

Other areas of the home that are used as storage areas for perishable food items also need to be thermally regulated in order to prevent accelerated decay of such items. This makes necessary the need for a Temperature Control System within the home.

For instance, during winter in most parts of Europe, the atmospheric / environmental temperature sometimes drops to as low as -15°C during the day. This temperature implies that few liquids can exist under such conditions (body fluids inclusive). Therefore, a Temperature Control System is needed to act as a "watchdog" to make sure that such a thermal condition never exists especially when people are in the house.

2.2 AN AUTOMATIC ROOM TEMPERATURE CONTROL WITH SECURITY SYSTEM

This project was carried out by Ahmad Faris Bin Zulkifli, a student of the University of Malaysia in May 2009. The aim of the project was to implement an automatic room temperature control system with an added security system for controlling the temperature in server rooms, especially those that are poorly ventilated and have no cooling units. The automatic room temperature control system utilized temperature sensors to detect the temperature of the server room. When the current temperature exceeds the set-point temperature, the Controller triggers on a cooling system made up of a set of brushless fans. These fans would cool the server room until the current temperature fell below the set-point temperature. The added Security System is perceived as an auxiliary system that regulates access to the server room door by demanding an access password to open the door. The system is built with a temperature sensor that is placed in the server room that detects the current temperature and displays the value on the LCD. A PIC Microcontroller reads the data from the temperature sensor which is in output voltage. The system will operate in three different conditions depending on the range of temperature. When the current temperature value reaches higher than the desired value, the fan will start functioning and the LED indicator for high temperature turns on. As the current temperature reaches the desired value, the fan stops functioning and the LED indicator for normal state temperature will come on. Finally, if the current temperature reaches lower than the desired value, the fan also stops functioning and the LED indicator for cold temperature comes on. Any changes in temperature in the room are continuously displayed on the LCD and the LEDs are used to indicate the current state and range of temperature in the server room.

With the security system (which also acts as a door lock security), the user has to provide the correct access code/password to gain access to the server room. If the correct password is inserted, the door unlocks. However, if the password provided happens to be the wrong password, the door remains locked and a buzzer alarm system is activated. The Temperature Control System and the Security System are designed to function independent of each other. Thus, the failure of one of the systems does not affect the functionality of the other.

The hardware comprised a PIC Microcontroller Circuit, A Sensory Input Circuit, A Driver Circuit, An LCD Display Module, LEDs and an Output Circuit. The system board was designed using a Bootstrap Mode Connection due to the constraints of size and finance.

The Microcontroller is the Microchip PIC18F4550 owing to its ease of use, built-in timers, and many digital inputs and outputs. The Temperature Sensor used was the LM35DZ Sensor. An Alphanumeric LCD was chosen having 2 lines of 16 characters each.

The block diagram of the project is given below:

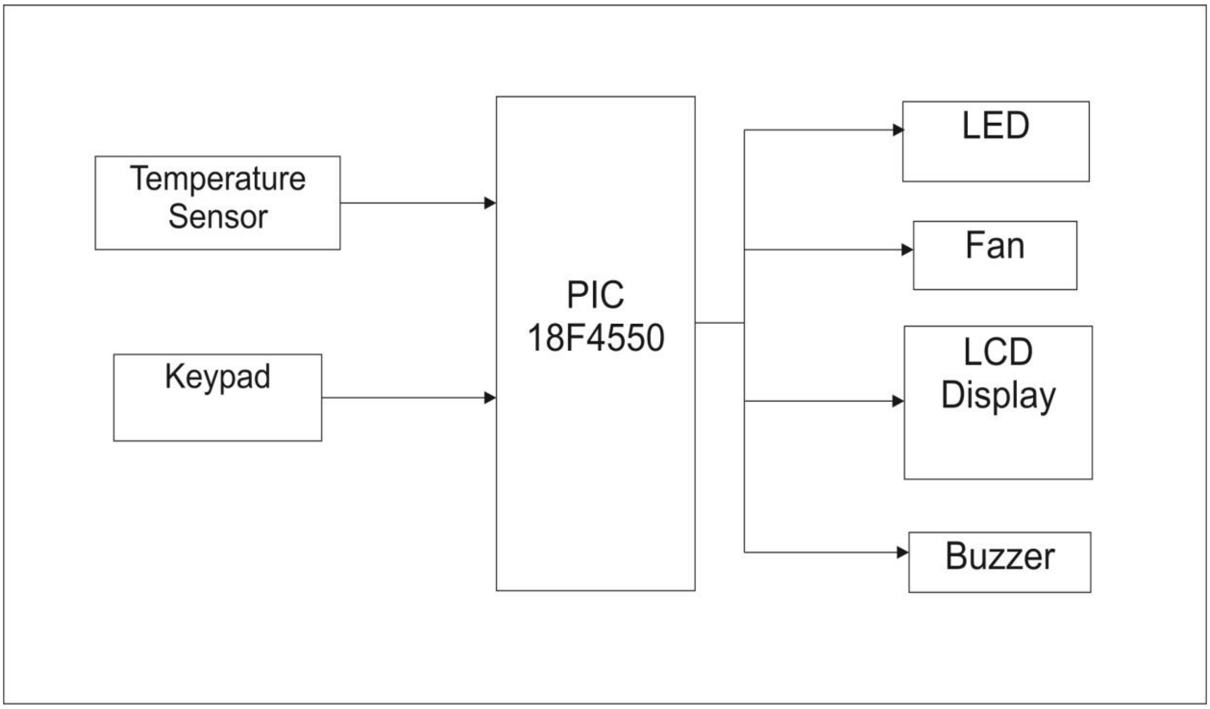


FIGURE 2.1 BLOCK DIAGRAM OF AN AUTOMATIC ROOM TEMPERATURE CONTROL WITH SECURITY SYSTEM

2.3 AN AVR LM92 TEMPERATURE SENSOR SYSTEM

The project was built using an LM92 Temperature Sensor, and a microcontroller AVR was used as the main processor. The control program was compiled using BASCOM.

The system is comprised of two main parts: the half-sphere contained four LM92 Temperature Sensors, which were connected to a small box containing an ATTiny2313 controller. The controller reads the input from the four sensors and then sends the temperature string over a low-speed RS232 cable to a display box which is close to the DMX light controllers. The display box has an ATMega32 which reads the temperature string and displays the result on a 240x128 graphics display using large digits. The ATMega32 also reads a potmeter to be used as a trip value. When one of the temperature readings exceeds this value, the display is repeatedly switched from normal to inverse as an alarm signal.

2.4 TEMPERATURE CONTROL SYSTEM USING LM35

In January 2008, Cytron Technologies Limited built a commercial Temperature Control System using 2 LM35 Temperature Sensors. Other components in the system included a PIC16F876A Microcontroller, Dc brushless Fans, LEDs, Buzzer and a BD135 power transistor. A difference in the design of this system over previously designed models by the company was that: in previous versions, the PIC was used to control the LEDs and Buzzers. In this project, the PIC doesn’t have enough current to perform this function; hence, an NPN power transistor (BD135) is used to power the brushless fans.

Two LM35 Temperature sensors are used to detect the temperatures of two different areas. The Control Program in the Microcontroller compares the temperature readings against a set-point that has been programmed into the micro-controller at the push of a button. When this set-point is exceeded, the buzzers are triggered on and the DC brushless fans are powered on to begin cooling the room. When the temperature normalizes, the DC Brushless fans switch off and the buzzers go off too.

The System Overview is shown in the diagram below:

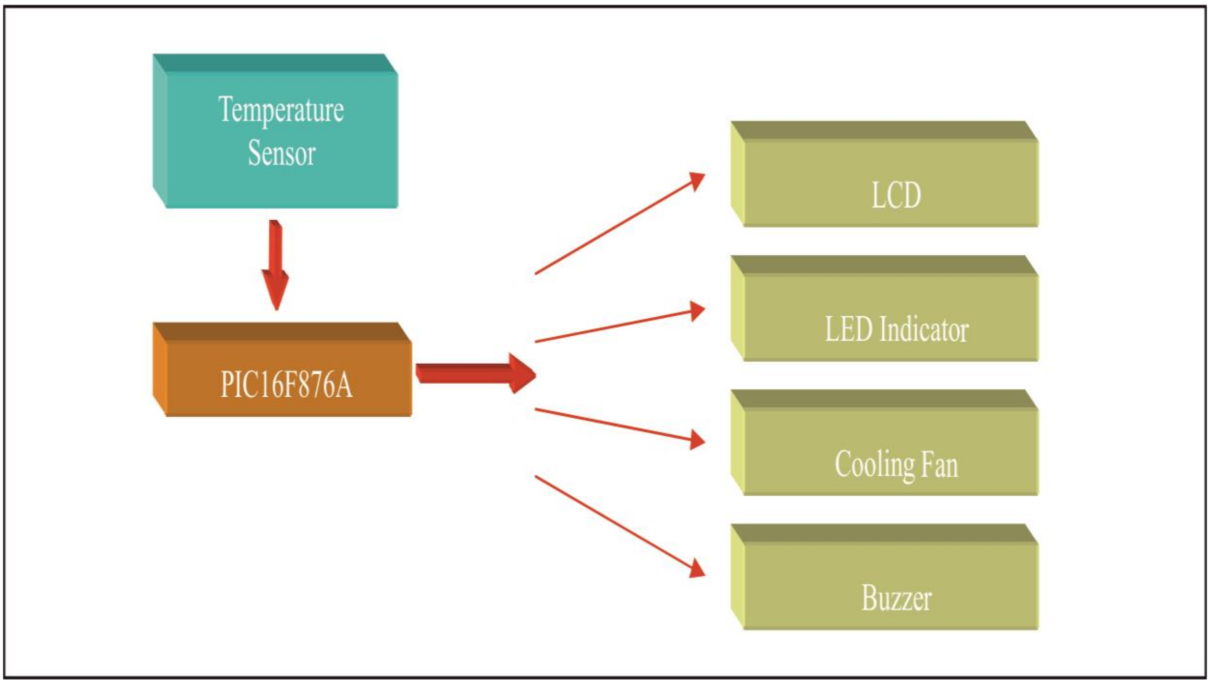


FIGURE 2.2 BLOCK DIAGRAM OF A TEMPERATURE CONTROL SYSTEM USING LM35

The Control Program was written in C Language. The Flowchart for the control program is given below:

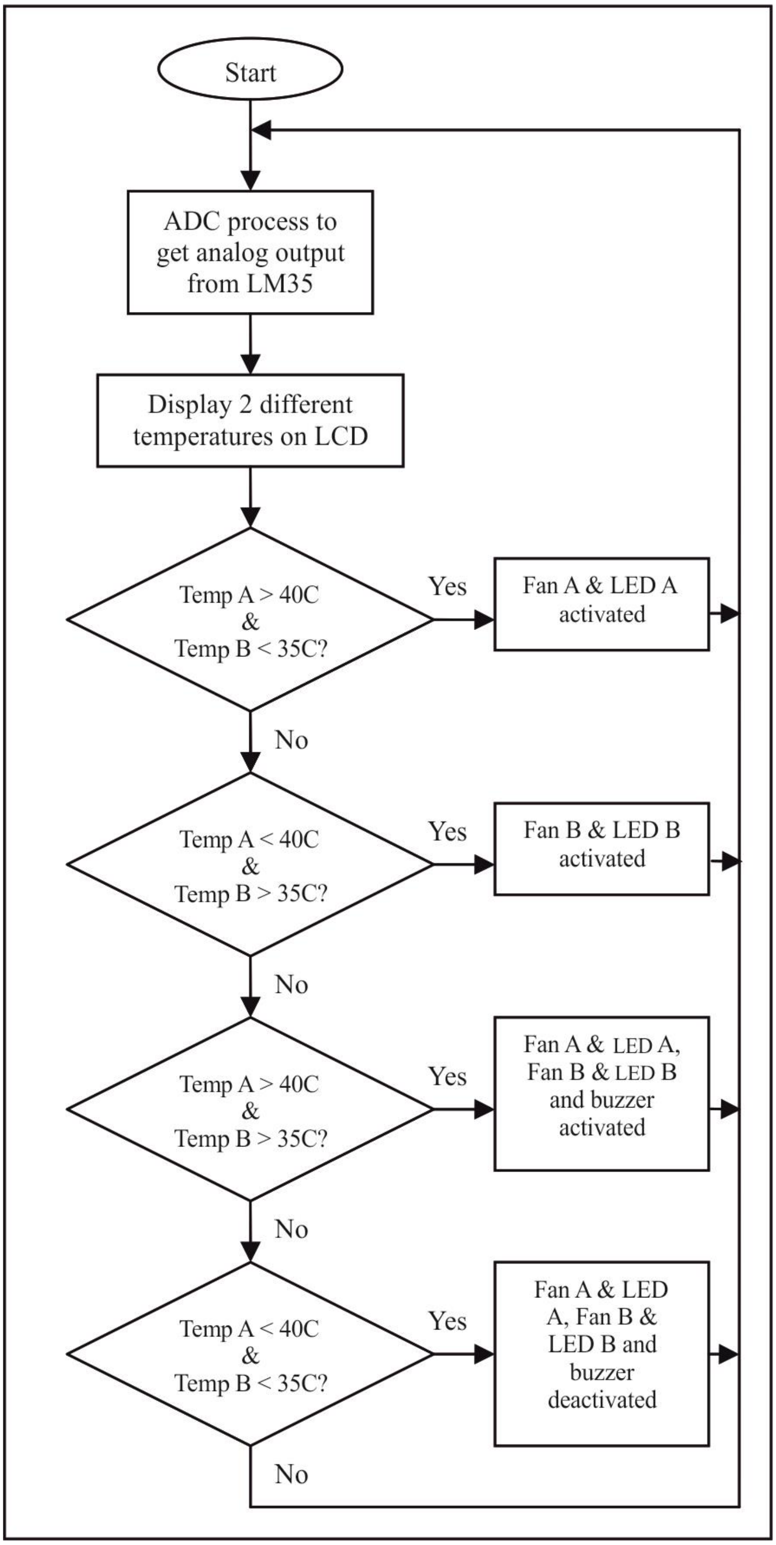


FIGURE 2.3 CONTROL PROGRAM FLOWCHART FOR A TEMPERATURE CONTROL SYSTEM USING LM35

2.5 TEMPERATURE ACQUISITION AND CONTROL SYSTEM

The design group at "North Controllers" created a temperature control system intended to solve the problem of temperature variation in an incubator facility. This system is to efficiently keep the temperature inside an incubator within a required range (10°C to 35°C). The required temperature range is set via a user friendly graphical interface from a supervisory/monitoring computer, thus, enabling the user to specify the minimum and maximum temperatures in the range. The current temperature within the incubator is measured using a temperature sensor. When the current temperature is below the lower limit of the desired range, the incubator must be heated using an air heater, and if it is above the upper limit of the desired range, it must be cooled by using a DC fan. When it is within the desired range, no control action is needed. The current temperature of the incubator must be continuously displayed on the supervisory computer screen, to one decimal place significance (for instance, 26.40C), and updated at least every tenth of a second. In addition, the controller should use LEDs to indicate the current state of the temperature in the incubator (within the range, below the low limit or above the high limit). The whole system is controlled using an MC68HC11.

Compared to the proposed project, there are a few differences. The proposed project is supposed to solve thermal overruns within the home and save energy cost by switching on the correct thermal regulator at the right time. The maximum habitable temperature within a home is about 37 °C; the system is to work at maintaining this habitable temperature for individuals living in the room. The proposed project has only a single upper limit temperature, above which the cooling system should be activated and below which the heater should be activated. The current temperature is displayed on the LCD Display every few seconds.

2.6 WATER LEVEL AND TEMPERATURE CONTROL USING A PROGRAMMABLE LOGIC CONTROLLER (PLC)

A proposal was submitted for the construction of the above project in November 2008, leading to the development of the project by Norhaslinda Binti Hasim. The objective of the project was to develop a simple process plant that can control water level and temperature in a single tank using a Programmable Logic Controller (PLC) Omron C200HS. A ladder diagram was constructed that can control the desired system by entering the mnemonic code into the programming console, PR027. The system could also be controlled via a remote access by using a CX-Programmer simulation. The flowchart below describes the process:

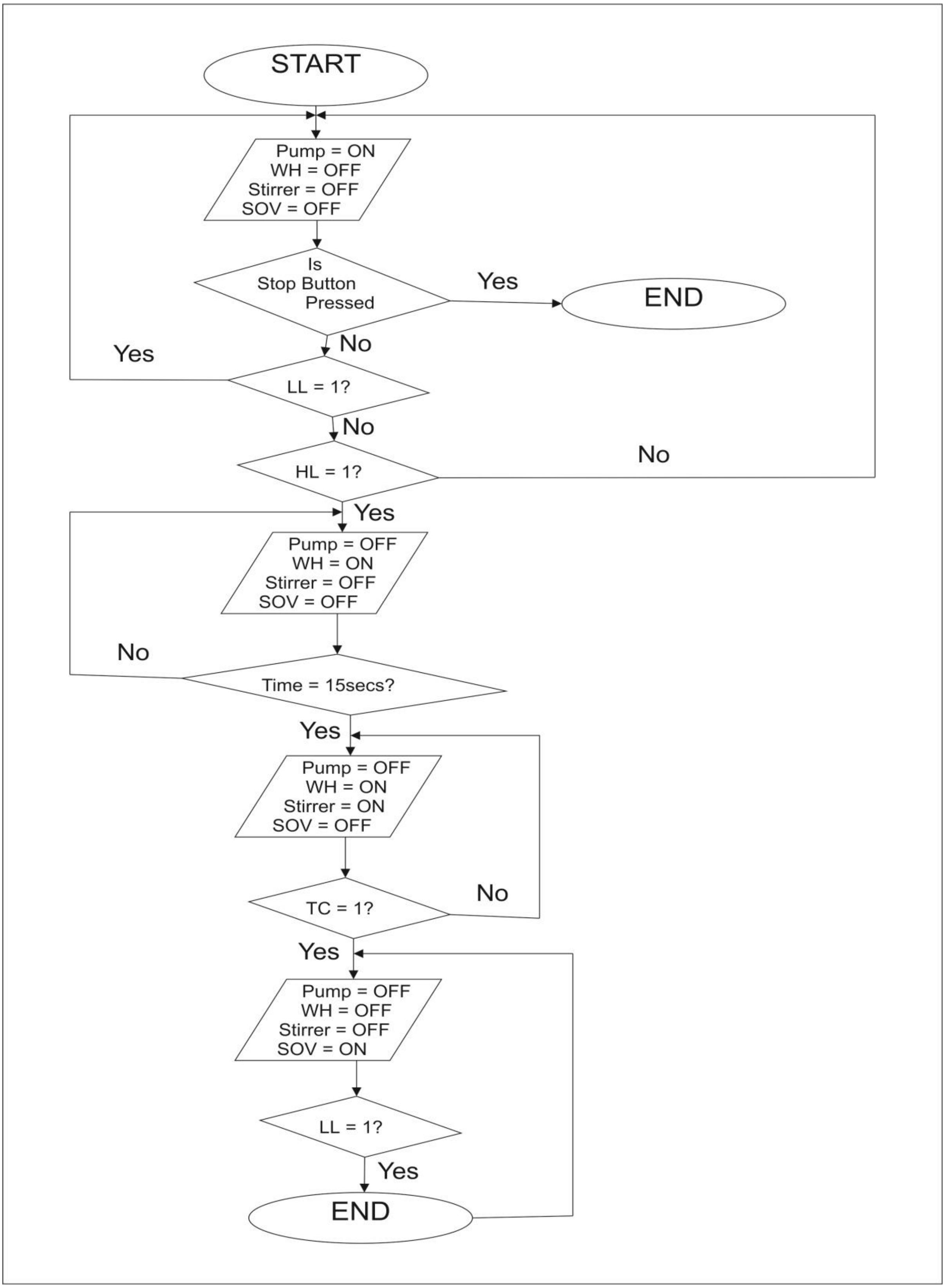


FIGURE 2.4 FLOWCHART FOR WATER LEVEL AND TEMPERATURE CONTROL USING A PLC

Programmable Logic Controllers were chosen in preference to PID Controllers due to the fact that PLCs are more accurate in their operations than PIDs. This is due to the fact that PID loops in operation require continuous monitoring and adjustments since they can easily become improperly tuned as a result of process parameter variations or changes in the operating conditions. The System was made of five input devices and four output devices:

INPUT DEVICES

A. Low level float
B. High level float
C. Temperature Controller
D. Start and Stop buttons

OUTPUT DEVICES

A. Pump
B. Heater
C. Stirrer
D. Solenoid Valve

At initial condition, regardless of whether the water level is at low level or below low level, the pump starts as soon as the start button is pressed; so as to allow water from the reservoir into the tank until the water reaches high level.

At this point, the pump will automatically stop and the heater switches on. After the water has been heated for 15 seconds, a stirrer is activated that stirs the water in the tank so as to get a consistent temperature.

When the temperature has reached 30°C, the heater and the stirrer stops automatically and the solenoid valve opens, allowing the heated water from the tank back into the reservoir. When the water level inside the tank gets to the low level, the solenoid valve closes automatically. This ends the whole process.

2.7 AN AUTOMATIC TEMPERATURE CONTROL SYSTEM USING RZK

Zilog Technologies implemented an Automatic Temperature Control System to demonstrate the possibility of an application running on Zilog's Real-Time Kernel (RZK) to be used to control various devices to maintain a certain temperature. This Temperature Control System reads a value from a temperature sensor and determines when to switch a fan (for cooling) or bulb (for heating) off or on according to minimum and maximum temperature limits settings. These settings are manipulated using upper and lower limit set switches.

The RZK is a real-time, pre-emptive, multitasking kernel designed for time-critical embedded applications. RZK objects used for real-time application development are Threads, message queues, event groups, semaphores, Timers, partitions and regions (memory objects), and interrupts.

A Real-time multitasking kernel, also called a real-time operating system (RTOS) is software that ensures that time-critical events are processed efficiently. The use of an RTOS generally simplifies the design process of a system by allowing the application to be divided into multiple independent elements called tasks.

The Block diagram of the hardware architecture is given below: It has a temperature sensor for reading temperature, a fan for cooling the sensor, a bulb for heating the sensor, switches for setting the upper and lower temperature limits, and the Character LCD module for displaying the current temperature, upper and lower limits.

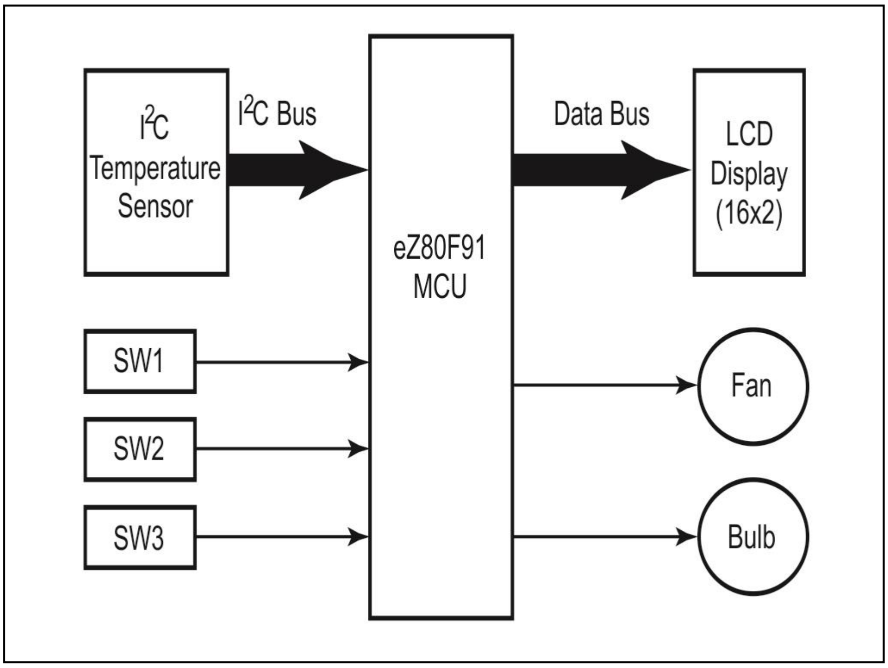


FIGURE 2.5 BLOCK DIAGRAM OF AN AUTOMATIC TEMPERATURE CONTROL SYSTEM USING RZK

The figure below displays the connections between the eZ80F91 Micro-Controller that was used in the design and a thermostat board. The data bus is connected to the Character LCD Module. Port pins PB0, PB1, and PB2 are connected to switches SW1, SW2 and SW3. Pins PB3 and PB7 are connected to the lamp and fan respectively. The MAX6625 temperature sensor used in the design is connected to the I^2C bus.

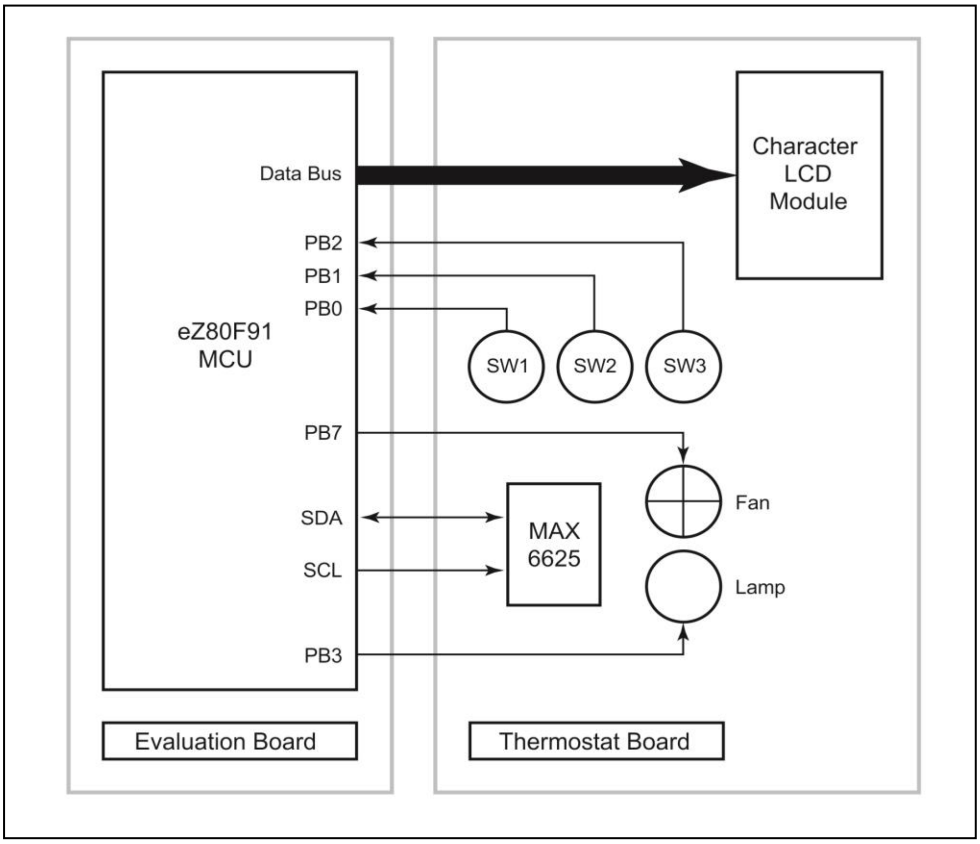


FIGURE 2.6 SCHEMATIC DIAGRAM OF AN AUTOMATIC TEMPERATURE CONTROL SYSTEM USING RZK

The software implementation for the automatic temperature control system provides the functionality to maintain a temperature within a specified limit. The main functions provided by this application are listed below:

- Automatic fan ON/OFF
- Automatic bulb ON/OFF
- Set lower and upper limits by pressing a switch
- Read a temperature from a temperature sensor
- Display the current temperature and the lower and upper limits on LCD

Complete functionality is managed by the following four functions, in order of priority from high (#1) to low (#4):

1. RZKTempReadTask (4)

2. TempControlTask (8)
3. TempDisplayTask (12)
4. RZKKeyControlTask (16)

These functions are executed according to their priority. The automatic ON/OFF of the bulb and fan, and the setting of the lower and upper limits is controlled by the **TempControlTask**. If the temperature read by **RZKTempReadTask** is greater than the set upper limit, then this task switches off the bulb and switches on the fan. If the read temperature is lower than the set limit, **RZKTempRead-Task** switches the bulb on and switches off the fan.

RZKTempReadTask (): This function reads the current temperature from I2C temperature sensor.

TempControlTask (): This function performs the following functions:

1. Set the upper and lower limit.
2. Upload the current temperature, upper, and lower limit to display array.
3. Compare the upper and lower set limit with current temperature and Switch ON/OFF the fan/bulb accordingly.

TempDisplayTask (): This function reads and updates the temperature on LCD display. It displays the current temperature, lower, and upper limit of temperature. The main operations performed by this function are read display buffer and update the display with current temperature with lower and upper limit.

RZKKeyControlTask (): This function scans the switches for setting the lower and upper limit of temperature. The main operations performed by this function are:

1. Scan the switches.
2. If switch SW1 is pressed, decrease the lower limit (LL).
3. If switch SW2 is pressed, decrease the upper limit (UL).
4. If switch SW1 and SW3 are pressed, increase the lower limit (LL).

5. If switch SW2 and SW3 are pressed, increase the upper limit (UL).

RZKApplicationEntry (): In addition to the four tasks described above, this fifth RZK function is the main entry point for any application into RZK. The application program entry function performs the following operations:

1. Initializes all peripherals.
2. Creates a function for reading the temperature from the I2C temperature sensor, controlling the temperature within a specified limit, and displaying the current temperature and limits on the LCD panel.
3. Resumes all functions.

The System key scan tasks are illustrated in the diagram below:

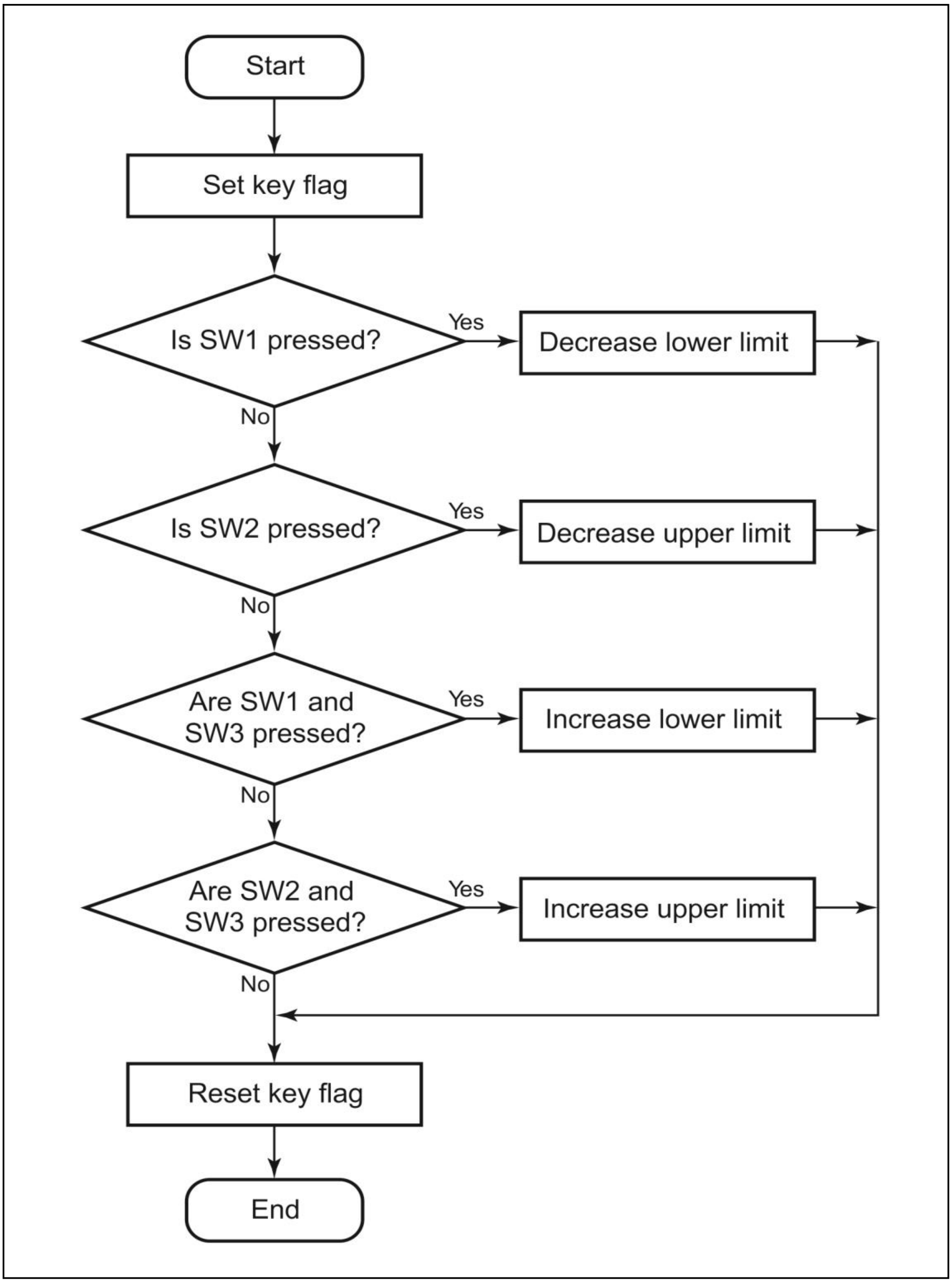


FIGURE 2.7 FLOWCHART FOR SYSTEM KEY SCAN TASKS OF AN AUTOMATIC TEMPERATURE CONTROL SYSTEM USING AN RZK

The Flowchart below illustrates the application entry thread flow:

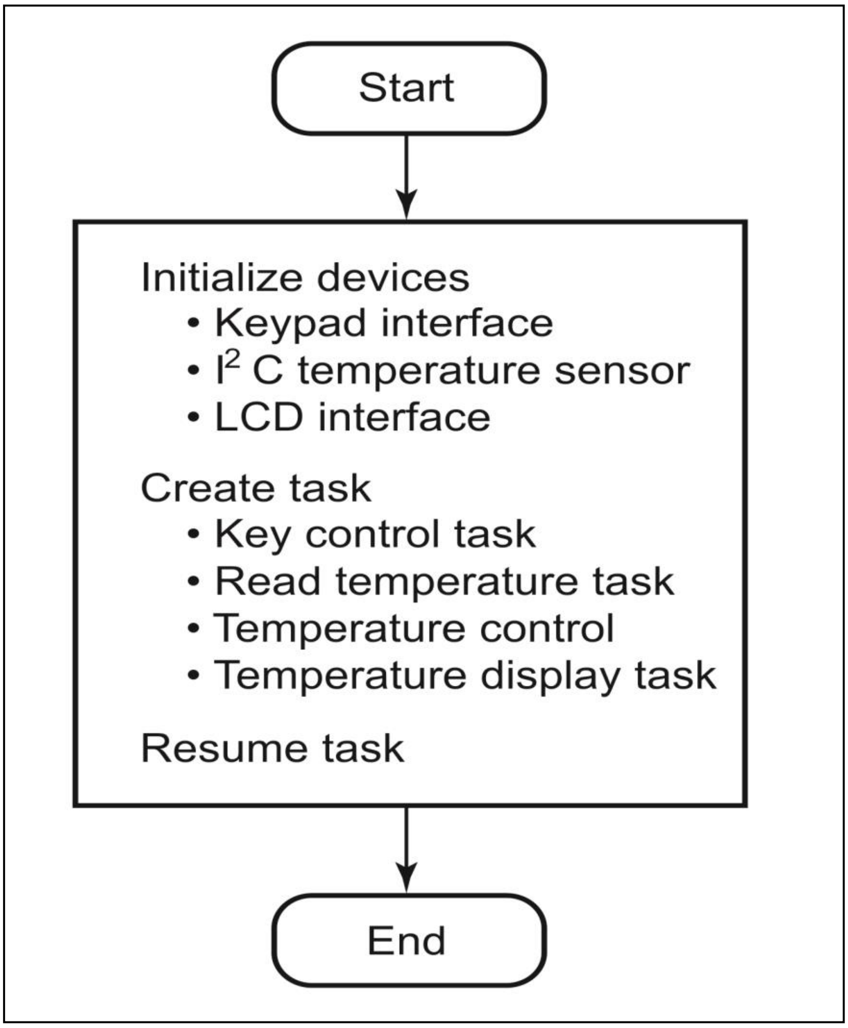


FIGURE 2.8 FLOWCHART OF AN AUTOMATIC TEMPERATURE CONTROL SYSTEM USING AN RZK

2.8 **SUMMARY OF PREVIOUS LITERATURE**:

From the review of the above literature, it can be concluded that temperature control and sensor systems follow a seeming conventional pattern. The sensor devices sense the temperature and pass it into the control unit for the micro-controller to test and determine that the temperature is within the set-point. When this test fails, an alarm system is activated and/or an effector system that would move to correct the temperature and return it back within range. With this conclusion it is safe to proceed with the project.

CHAPTER THREE

SYSTEM DESIGN AND ANALYSIS

3.1 **PREAMBLE**

This chapter deals with the design and analysis of this Temperature Control System, and sheds more light on how the variable power supply is built and other various aspects of the system and their respective components. The basic units of the system are given below:

- POWER SUPPLY UNIT
- TEMPERATURE SENSING UNIT
- TEMPERATURE CONTROL UNIT
  - ✓ MENU/FUNCTION UNIT
- DISPLAY (LCD) UNIT
- SWITCHING CIRCUIT
- SYSTEM ALARM UNIT

3.2 **POWER SUPPLY UNIT**

All stages in the system require a +12volts DC supply except for the microcontroller and the sensor ICs which require a +5volts supply. Hence, a stabilized DC voltage is required for the microcontroller and the sensor Ics. The method used is to create a linear power supply which includes a 12v step-down transformer, a bridge rectifier, filter capacitor, and a 5volts voltage regulator of positive output kind.

3.2.1 **TRANSFORMER**

Transformers convert AC electricity from one voltage to another with little power losses. Step-up Transformers increase voltage, while step-down transformers reduce voltage. Most power supplies such as the one used in this project use a step-down transformer to reduce the dangerously high mains voltage (230V in Nigeria) to a safer low voltage. The input coil is called the **primary** and the output coil is called the **secondary**. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. This is shown below:

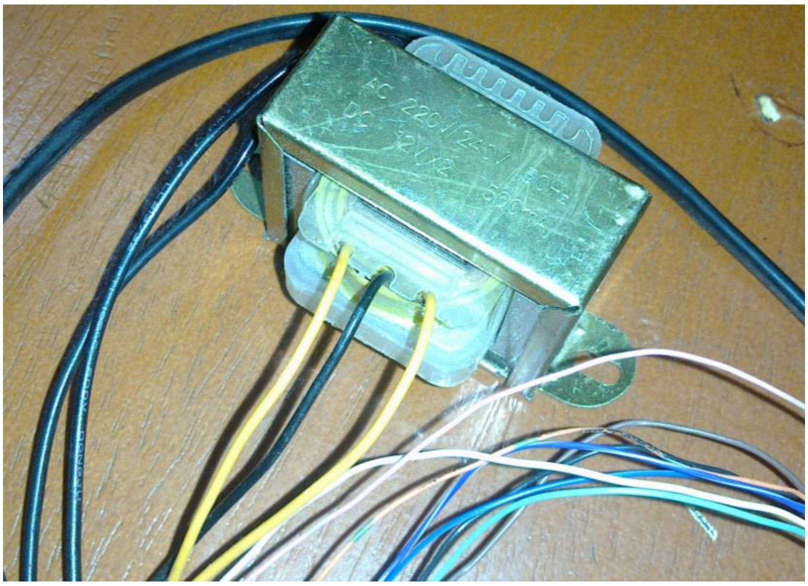


FIGURE 3.1 DIAGRAM SHOWING THE 12V STEP-DOWN TRANSFORMER

3.2.2 **RECTIFIER**

Rectifiers convert AC to DC. There are several ways of connecting diodes to make rectifiers that convert AC to DC. The bridge rectifier is the most important and it produces a **full-wave** varying DC. A full-wave rectifier can also be made from just two diodes if a centre-tap transformer is used, but this method is rarely used now that diodes are cheaper. A single diode can be used as a rectifier but it only uses the positive (+) parts of the AC wave to produce **half-wave** varying DC.

3.2.3 **FILTRATION**

Smoothening is performed by a high valued electrolytic capacitor connected across the DC supply to act as a reservoir, supplying current to the output when the varying DC voltage from the rectifier is falling. The capacitor charges quickly near the peak of the varying DC, and then discharges as it supplies current to the output.

3.2.4 **REGULATION**

Voltage regulator Ics are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current ('overload protection') and overheating ('thermal protection').

Many of the fixed voltage regulator Ics has 3 leads and look like power transistors, such as the 7805 +5V 1A regulator shown below. They include a hole for attaching a heat sink if necessary.

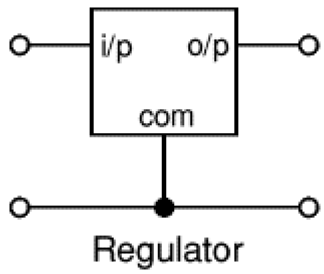


FIGURE 3.2 DIAGRAMS SHOWING VOLTAGE REGULATORS

The Schematic application of the L7805CV Voltage Regulator IC used in this project is also given below:

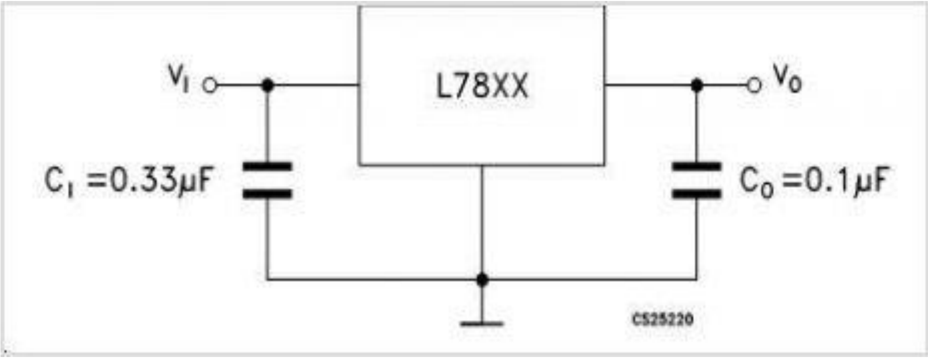


FIGURE 3.3 SCHEMATIC DIAGRAM OF THE L7805CV VOLTAGE REGULATOR

(Refer to Manufacturer's Website for Complete Datasheet)

The entire power supply circuit is shown below:

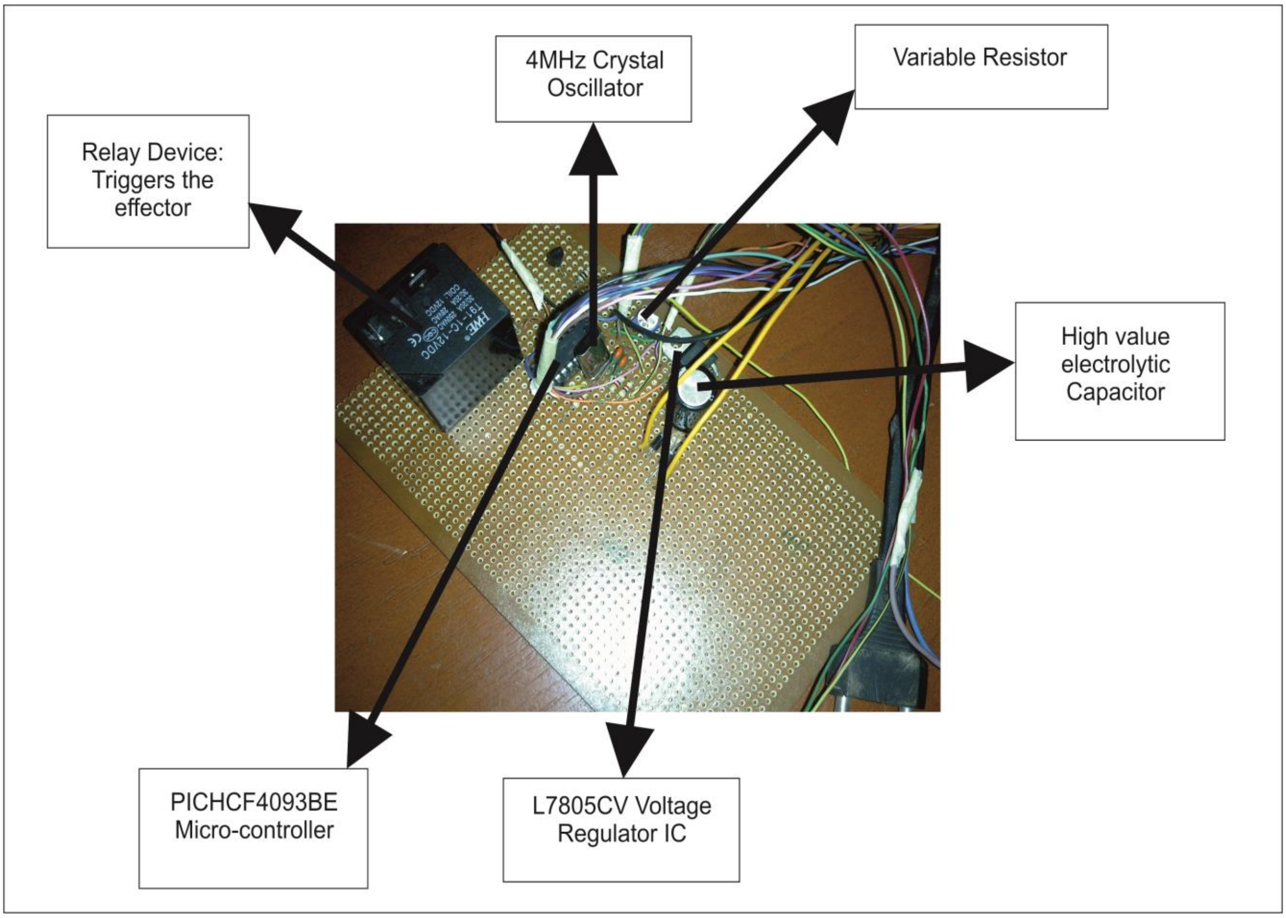


FIGURE 3.4 DETAILED DIAGRAM SHOWING COMPONENTS OF THE POWER CIRCUIT

3.3 **TEMPERATURE SENSING UNIT**

This unit senses or detects the temperature in a room. The unit consists of a ceramic Negative Temperature Coefficient Thermally Sensitive Resistor (NTC Thermistor). This type of temperature sensor exhibits a decrease in electrical resistance with increasing temperature. It is a semi-conductor based ceramic device. It generally has an operating temperature range of –50°C to +150°C and is accurate to ±0.1°C. The NTC Thermistor has a relatively large change in resistance with respect to temperature – of the order of -3% per °C to -6% per °C. This provides an order of magnitude of greater sensitivity or signal response than most other temperature sensors such as Thermocouples and RTDs. This sensor is well suited for sensing temperature at remote locations via long two-wire cables, because the resistance of the long wires is insignificant compared to its relatively high resistance. The diagram below shows the NTC Thermistor used in the construction of this project:

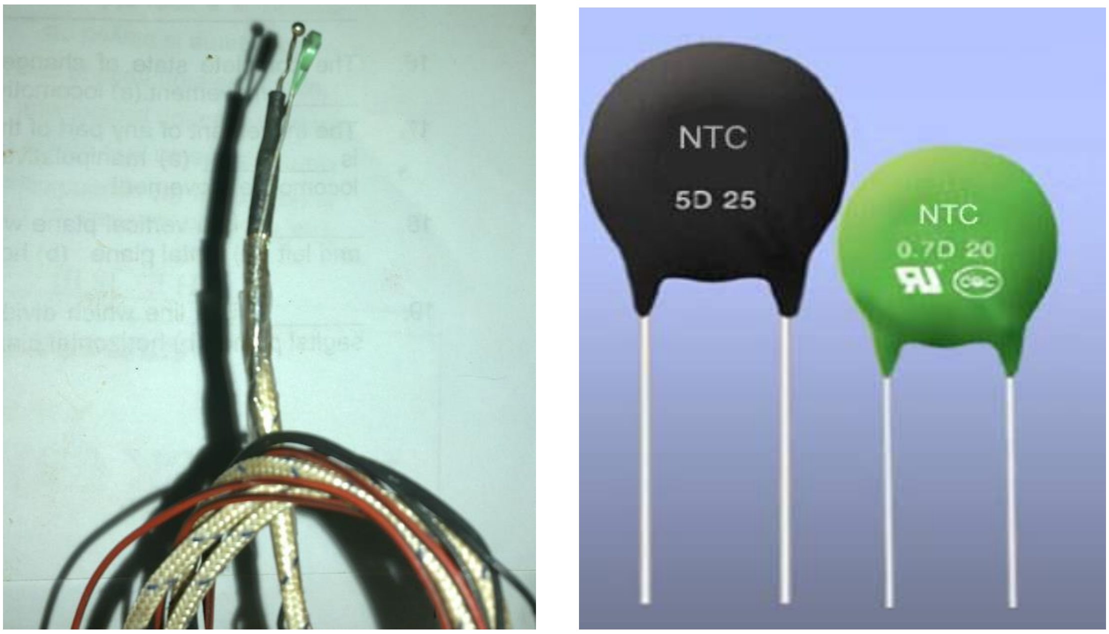


FIGURE 3.5 DIAGRAMS SHOWING THE NEGATIVE TEMPERATURE COEFFICIENT THERMISTOR

(Refer to Manufacturer's Website for Complete Datasheet)

3.4 **TEMPERATURE CONTROL UNIT**

This unit processes the temperature detected by the NTC THERMISTOR IC, and controls the overall operation of the system. It consists of an ATMEL AT89C52 microcontroller IC which contains a non-volatile FLASH program memory that is parallel programmable.

A **microcontroller** (sometimes abbreviated **μC**, **uC** or **MCU**) is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications.

The ATMEL AT89C52 microcontroller IC is shown below:

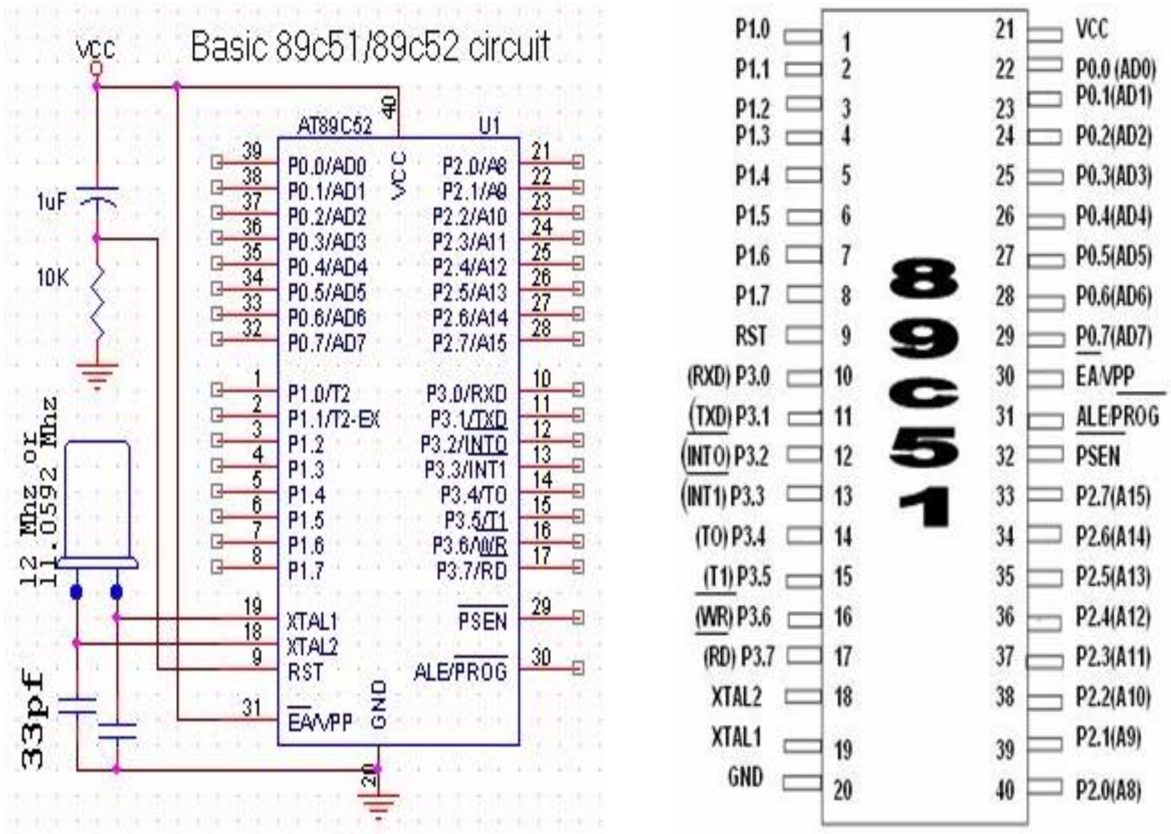


FIGURE 3.6 SCHEMATIC DIAGRAM OF THE ATMEL AT89C52

MICROCONTROLLER

(Refer to Manufacturer's Website for Complete Datasheet)

The basic features of the 89C52 are given below:

- 80C51 Central Processing Unit
- On-chip FLASH Program Memory
- Speed up to 33 MHz
- Full static operation

- RAM expandable externally to 64 k bytes
- 4 level priority interrupt
- 6 interrupt sources
- Four 8-bit I/O ports
- Full-duplex enhanced UART
    1. Framing error detection
    2. Automatic address recognition
- Power control modes
    1. Clock can be stopped and resumed
    2. Idle mode
    3. Power down mode
- Programmable clock out
- Second DPTR register
- Asynchronous port reset
- Low EMI (inhibit ALE)
- 3 16-bit timers
- Wake up from power down by an external interrupt

3.4.1 **USING THE MICROCONTROLLER**

A microcontroller is a good-natured “genie in the bottle” and no extra knowledge is required to use it. In order to create a device controlled by the microcontroller, it is necessary to provide the simplest PC program for compiling and simple device to transfer that code from PC to chip itself. Even though this process is quite logical, there are often some queries, not because it is complicated, but for numerous variations. A Crystal Oscillator is necessary when using the Micro-controller. The Crystal Oscillator helps in regularizing and stabilizing the voltage signal that is being fed into the Micro-controller. In this project, a 4MHz Crystal Oscillator is used, and this is shown in the diagram below:

FIGURE 3.7 DIAGRAM SHOWING A 4MHZ CRYSTAL OSCILLATOR

(Refer to Manufacturer’s Website for Complete Datasheet)

3.4.2 **WRITING THE CONTROL PROGRAM**

Writing the Control Program for the Microcontroller requires a specialized program in the Windows environment. Any text editor can be used for this purpose. The program is written in Assembly Language. The bone of contention here is to write all instructions in such an order that they should be executed sequentially by the microcontroller; observing the rules of assembly language programming and writing instructions exactly as they are defined. In other words, you just have to follow the program idea! That's all! When using custom software, there are numerous tools which are also installed to aid in the development process. One such tool is the Simulator. This enables the user to simulate/test the code prior to burning it to the MCU.

```
Loop      button PORTA,0,0,Increment
          button PORTA,1,0,Decrement
          goto Loop

Increment incf cnt,f
          movf cnt,w
          movwf PORTB
          goto Loop

Decrement decf cnt,f
          movf cnt,w
          movwf PORTB
```

To enable the Assembly Language Compiler to successfully run the program, the source file must have the extension, `.asm` in its name, for example: Program.asm

3.4.3 **TRANSLATING THE CONTROL PROGRAM**

The microcontroller does not understand assembly language. Hence, it is necessary to translate the program into machine language. It is made easier when using a custom program (*such as MPLAB in this case*) because a translator is built into it and it is just a click away! The Machine Language code is generated with a `.hex` ("hex code") extension.

The Machine Code is then compiled and simulated for error checking to ensure that all functions and parameters are performing the correct tasks. The report screen is shown below:

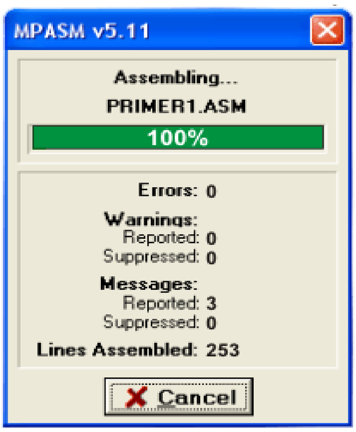


FIGURE 3.8 DIAGRAM SHOWING RESULT OF MICROCONTROLLER PROGRAM ERROR CHECKING

3.4.4 **PROGRAMMING THE MICROCONTROLLER**

To move the "hex code" from PC to the microcontroller, a cable for Serial, Parallel or USB communication and a special device called "*programmer*" is used with appropriate software.

The steps are simple: Insert the Micro-chip onto the "*programmer*" device and connect the device to the PC, then load/open the "hex code" document; set a few parameters on the burning software (*PIC Flash* in this case) such as the type of the microcontroller, frequency and clock oscillator etc., and click on the "WRITE" icon for burning, Then, *Voila!*. After a while, a sequence of 0's and 1's are burned onto the microcontroller through the Serial, Parallel or USB connection cable and the "*programmer*" hardware.

Refer to Appendix I for the source Assembly Language program in the ATMEL AT89C52 Microcontroller of This Temperature Control System.

3.4.5 **MENU/FUNCTION UNIT**

This unit is made up of a variable resistor that is used to increase and decrease the set-point temperature in the microcontroller. Turning the variable resistor in a clockwise direction increases the maximum temperature while turning it in an anti-clockwise direction decreases the maximum temperature. The figure below shows this unit:

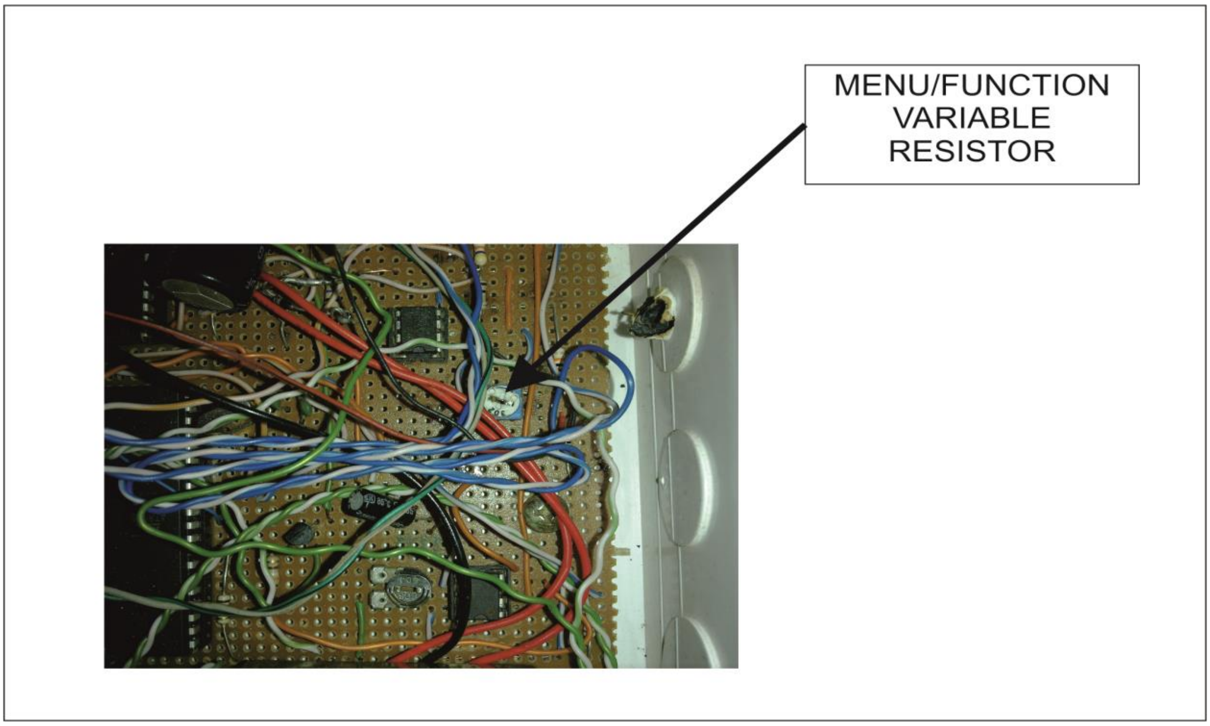


FIGURE 3.9 DIAGRAM SHOWING THE MENU/FUNCTION VARIABLE RESISTOR

3.4.5.1 **COMPARATION**

An op-amp configuration without a feedback is used as a comparator. The purpose of the comparator is to compare two voltages and produce a signal that indicates which voltage is greater. Any difference in the two voltages, no matter how small drives the Op-Amp into Saturation, but if the voltages supplied by the two inputs are of the same magnitude and polarity, the output from the Op-Amp is 0 volts. An Operational Amplifier is used here for comparing the sensed temperature value (in form of voltage signals) from the thermistor with the set-point temperature that has been set using the variable resistor. This is shown in the diagram below:

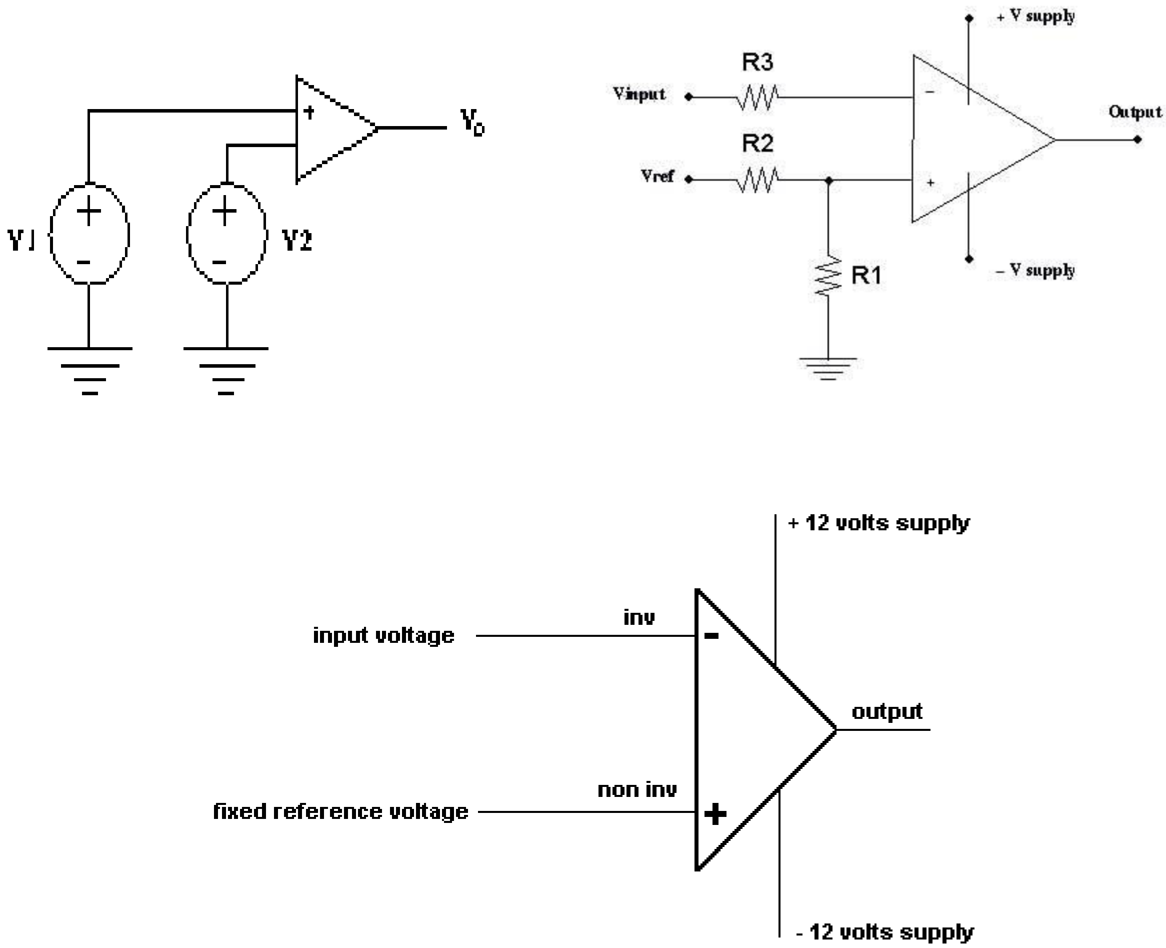


FIGURE 3.10 DIAGRAMS SHOWING AN OPERATIONAL AMPLIFIER USED AS A COMPARATOR

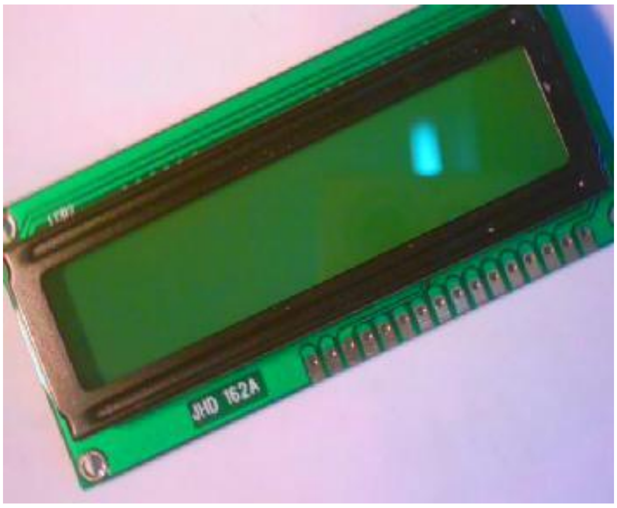
(Refer to Manufacturer's Website for Complete Datasheet)

The Op-Amp Comparator will produce a negative voltage at its output when the voltage at the non-inverting input is more positive than the voltage at the inverting input.

Owing to the fact that one of the inputs of the Op-Amp is inverted, the output from the Op-Amp is again passed through a NAND Gate that inverts the signal back to its original state.

3.5 **OUTPUT/DISPLAY UNIT**

The display unit is made of a 16x1 Liquid Crystal Display (LCD). This is used to display the reading of the NTC THERMISTOR Temperature Sensor device as it changes in response to the current environmental temperature. This is shown below:



Dimensions in inches
Scale: None

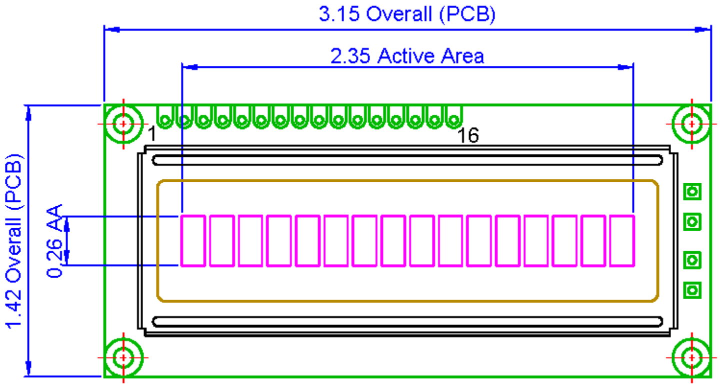


FIGURE 3.11 DIAGRAMS SHOWING THE LIQUID CRYSTAL DISPLAY

(Refer to Manufacturer's Website for Complete Datasheet)

3.6 **SWITCHING CIRCUIT**

The output from the micro-controller pin 24 triggers the switching circuit. The signal is passed through a MOSFET. The **Metal–Oxide–Semiconductor Field-Effect Transistor** (MOSFET) is a transistor used for amplifying or switching electronic signals. The basic principle of this kind of transistor is based on the fact that a voltage on the oxide-insulated gate electrode can induce a conducting channel between the two other contacts called source and drain. The channel can be of n-type or p-type, and is accordingly called an nMOSFET or a pMOSFET. The MOSFET activates the relay which turns on the external heating or cooling system. The diagram of the MOSFET connected to the relay is given below:

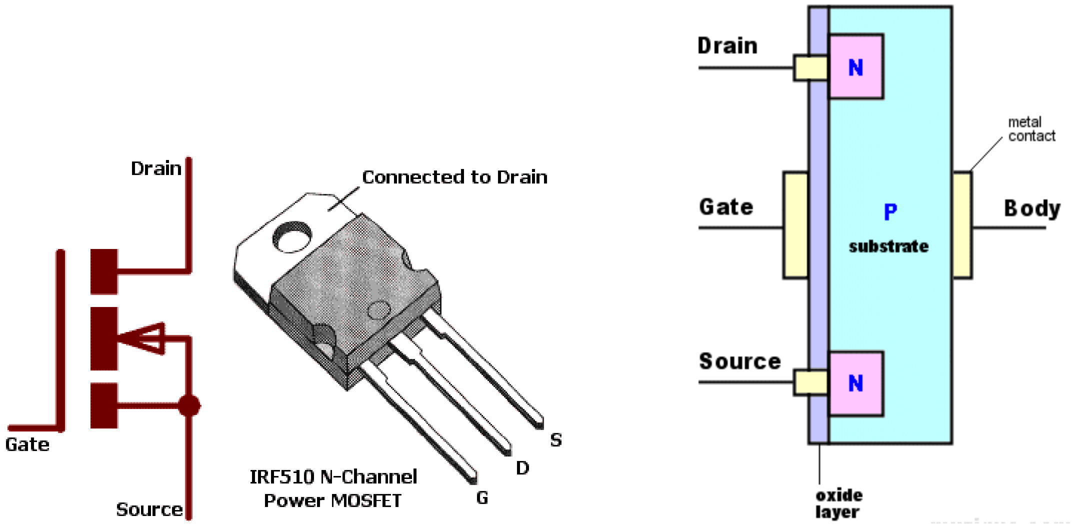


FIGURE 3.12 SCHEMATIC DIAGRAM OF A MOSFET

(Refer to Manufacturer's Website for Complete Datasheet)

3.7 **SYSTEM ALARM UNIT**

This consists basically of a buzzer or a wailer alarm sounder. A buzzer is a signalling device, usually electronic, typically used in automobiles, household appliances such as microwaves, or game shows to pass alerts on system events.

It most commonly consists of a number of switches or sensors connected to a control unit that determines which system event or breach has occurred. The buzzer in this project is used as a warning/alerting device which informs that the existing room temperature has risen above the preset temperature value. The buzzer is wired to a "driver" circuit contained in the 89C52 microcontroller IC, and to a 555 Timer which pulses the sound on and off. This is connection is shown below:

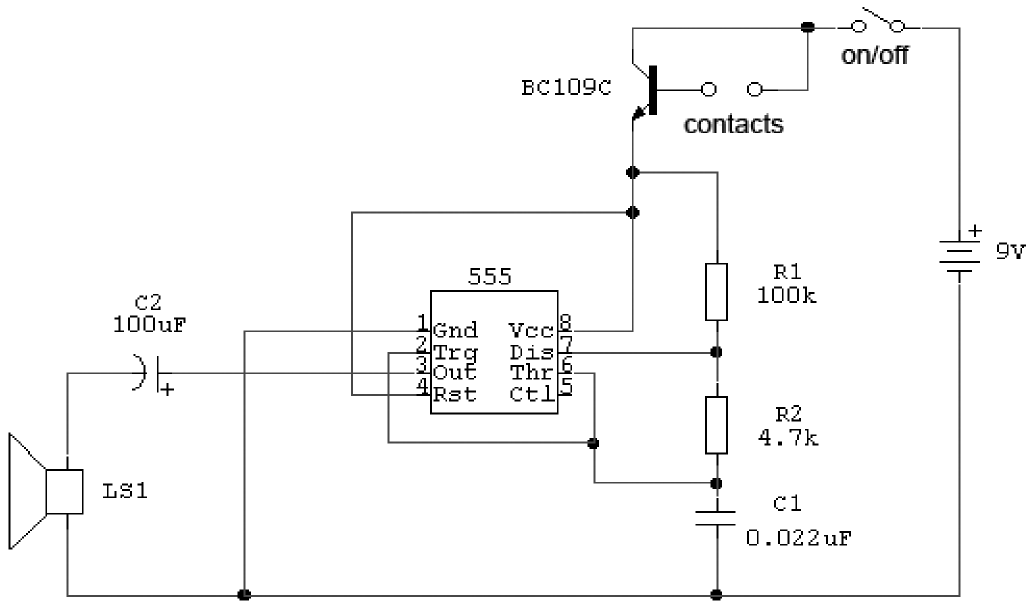


FIGURE 3.13 DIAGRAM SHOWING A 555 TIMER CONNECTED TO A BUZZER

(Refer to Manufacturer's Website for Complete Datasheet)

3.8 **SYSTEM DESIGN MODEL**

The Design Model used in the designing of this particular system is the On/Off Control Model. As explained in Chapter One, using this method means that the output from the device is either on or off, with no middle state/position. An on-off controller switches the output only when the temperature crosses the set-point. For heating control, the output is on when the temperature is below the set-point and off above the set-point, and vice-versa for cooling control. Since the temperature crosses the set-point to change the output state, the process temperature will be cycling continually, going from below the set-point to above, and back below. In cases where this cycling occurs rapidly, and to prevent damage to contactors and valves, and on-off differential, or "hysteresis" is added to the controller operations. This differential requires that the temperature exceed the set-point by a certain amount before the output will turn off or on again. On-off differential prevents the output from “chattering” or making fast, continual switches if the cycling above and below the set-point occurs very rapidly. On-off control is usually used where a precise control is not necessary such as in systems which cannot handle having the energy turned on and off frequently, where the mass of the system is so great that temperatures change extremely slowly, or for a temperature alarm. One special type of on-off control used for alarm is a limit controller. This controller uses a latching relay, which can be manually reset, and is used to shut down a process when a certain temperature is reached.

CHAPTER FOUR

CONSTRUCTION AND TESTING

4.1 PREAMBLE

This chapter seeks to look at the workability and practicability of the temperature sensor and control system when the system is fully operational.

4.2 SYSTEM IMPLEMENTATION

The temperature control system consists of temperature sensor, a microcontroller and a display module.

The whole system requires stabilized 5V electricity to function and the power supply unit reduces the 220V from the mains to the above mentioned voltage through a step down transformer.

The system functionality is described below:

When power is given to the system the temperature sensor detects the room temperature and sends whatever it detects to the microcontroller to process the temperature. The sensor is wired to the microcontroller's input port but it is exposed to atmosphere so that it can have direct contact with the environment. The microcontroller cannot work on its own because it is like a storage device which is empty at upon purchase of it. Therefore, some data in the form of computer codes have to be programmed into the microcontroller to make it work with its associated circuits.

The microcontroller after processing the signal transmitted by the temperature sensor sends the final process to the display module so that it is easily understood by lay men.

The set-point is programmed such that when the temperature of the environment rises to the pre-set value an alarm/buzzer is sounded. The complete circuit diagram of the project is given below:

COMPLETE CIRCUIT DIAGRAM FOR THE 89C52 CONTROLLED CIRCUIT

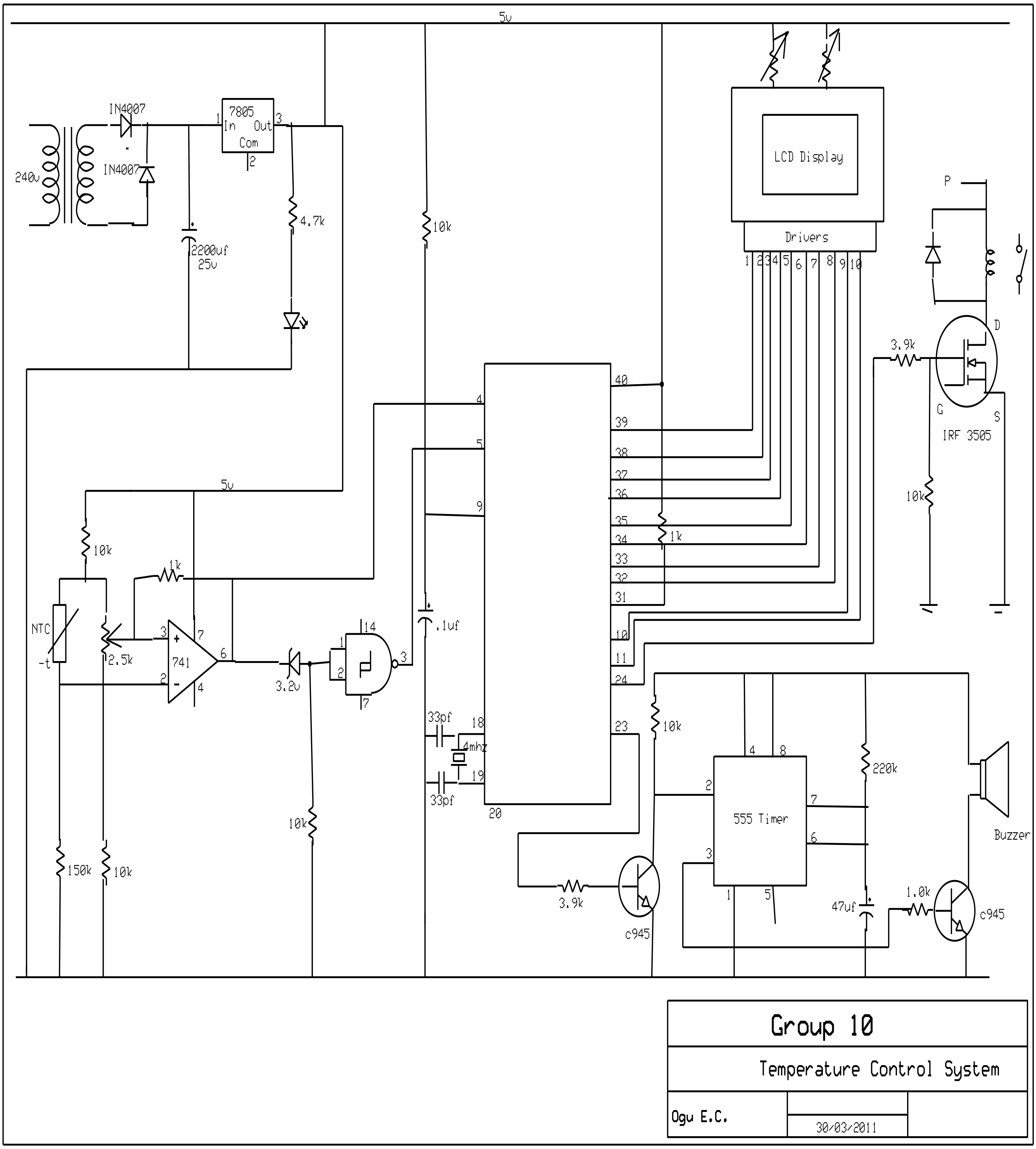


FIGURE 4.1 COMPLETE CIRCUIT DIAGRAM

The Negative Temperature Co-efficient Thermistor is connected to the IC 741 which is an Operational Amplifier using a Bridge Network. The Operational functions as a Comparator. It compares the input voltage coming from the Thermistor to the Maximum possible voltage that has been pre-set into the Micro-controller using the variable Resistor. The Operational Amplifier is connected also to a Zener Diode which cuts off all irregular input voltages and signal noise from the sensor, thus, ensuring that the signal proceeding towards the Micro-controller is a clean undistorted signal. This signal is further passed through a NAND Gate IC which inverts the signal before sending it to the Micro-controller. The actual comparison is done inside the Micro-controller which sends the input temperature signal to the LCD for display, and sends an interrupt to the buzzer to turn it on when the sensed temperature exceeds the set-point.

4.3 TESTING AND ANALYSIS

The circuit was tested as it was being developed. During the construction, the following faults were initially encountered and corrected as explained further below;

- Microcontroller not functioning.
- Mal-functioning of the alarm system.
- Display reading not bright and clear enough.

4.3.1 MICROCONTROLLER NOT FUNCTIONING

The system did not work at all when power was given to it at the first time of testing the project. The microcontroller had to be re-programmed being the heart of the project and every other circuit depending on it to function. The problem was solved after the re-programming of the microcontroller was done.

4.3.2 MAL-FUNCTIONING OF THE ALARM SYSTEM

The alarm is meant to sound only when the room temperature is higher than the set-point value. It was discovered, however, that the alarm started sounding immediately the system was provided with power. After troubleshooting, it was discovered to be as a result of bugs in the control program. The bug was corrected in the program and it worked normal.

4.3.3 DISPLAY READING NOT BRIGHT AND CLEAR ENOUGH

The display module was not clear enough to be seen even under a dark condition. The driver transistors used to buffer the display were replaced with higher valued ones and the current limiting resistors to each transistor were replaced with higher resistance.

4.4 CONSTRUCTION AIDS

The following tools and materials were used during the construction of this project:

- **Vero-Board:** The circuits were built on this board. It requires soldering and it has high connectivity which does not allow or enable two or more circuits to be placed on board.
- **Connecting cables:** These connectors are used to connect the various components together in the circuit.
- **Pliers and wire strippers:** These were used to cut the cables and remove their outer covering to prepare them for soldering and easy connection on the Vero-board.
- **Soldering iron and lead:** These were used to connect the cables together and to connect components to the copper layer of the Vero-board.
- **Multi-meter:** This was used to test values of the components used to ensure that certain components were working efficiently and to test continuity in cables and components.

4.5 LIMITATIONS

The limitations encountered in this project are listed below:

- This system is microcontroller based and therefore must be handling with care to prevent shifting of the microcontroller.
- Forgetting to activate the system while going out will be a great risk.
- System may be useless if power back-up is not provided in form of batteries, generators or inverters.

4.6 SYSTEM FUNCTIONAL REQUIREMENTS

For the temperature control system to operate optimally, there are some requirements that have to be met. These requirements include the following:

- The temperature sensor should be well positioned.
- The system should be protected from rain and even excessive heat that is generated from fire.
- The alarm should be positioned where it will be heard.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

Controlling the temperature is a major problem in our rapidly evolving world and it needs cost-efficient solutions. This Temperature Control System shows a way to get the temperature value and displaying the value on a graphical LCD via 89C52 microcontroller. In this Project temperature values are measured in analog form, and then it is converted to digital by the Microcontroller. Digital data is used for driving the graphical LCD by the microcontroller unit. The user can configure a set-point temperature value and control an external heating and/or cooling device by using the Temperature Control System. The system can be used as the basis for developing custom solutions for networked and stand alone data collection and control equipment. It can be centrally powered due to its low current requirement and its small size makes it more portable, allowing it to be placed almost anywhere.

5.2 LIMITATIONS IN MOST TEMPERATURE CONTROL SYSTEMS

Most Temperature Control Systems use either of the On/Off, Proportional or PID Control techniques/Algorithms as discussed earlier in Chapter One. All these individual Algorithms have their unique defects and limitations. Some of these are discussed below:

5.2.1 LIMITATIONS OF THE ON/OFF CONTROL TECHNIQUE

The On-off control will work where the overall control system has a relatively long response time (the time between the instances the temperature sensor provides its output to the comparator when the comparator activates the effectors). This means that the On/Off Control Algorithm would be inefficient and will result in instability if the system being controlled has a rapid response time. The on-off control can be better understood as driving a car by applying either full power or no power (to the accelerator pedal) and varying the duty cycle, to control speed. The power would be on until the target speed is reached, and then the power would be removed, so the car reduces speed. When the speed falls below the target, with a certain hysteresis, full power would again be applied. It can be seen that this looks like **pulse-width modulation**, but would obviously result in poor control and large variations in speed. The more powerful the engine is, the greater the instability; the heavier the car, the greater the stability. Stability may be expressed as correlating to the power-to-weight ratio of the vehicle.

5.2.2 LIMITATIONS OF THE PROPORTIONAL CONTROL TECHNIQUE

Proportional control overcomes the limitations of the On/Off Control Technique. This it achieves by modulating the output to the controlling device, such as a continuously variable valve. Let's go further to see proportional control as the way most drivers control the speed of a car. If the car is at target speed and the speed increases slightly, the power is reduced slightly, or in proportion to the error (the actual versus target speed), so that the car reduces speed gradually and reaches the target point with very little, if any, "overshoot", so the result is a much smoother control than the on-off control.

5.2.3 LIMITATIONS OF THE PROPORTIONAL INTEGRAL – DIFFERENTIAL (PID) CONTROL TECHNIQUE

Further refinements to the Proportional Control Technique like the PID control would help compensate for additional variables like hills, where the amount of power needed for a given speed change would vary, which would be accounted for by the integral function of the PID control. PID controllers are applicable to many control problems, and most times they perform satisfactorily without any improvements or even tuning; they can perform poorly in some applications, and do not in general provide *optimal* control. **The fundamental difficulty with PID control is that it is a *feedback* system, with *constant* parameters, and no direct knowledge of the process, and thus overall performance is reactive and a compromise – while PID control is the best controller with no model of the process, better performance can be obtained by incorporating a model of the process.** The most significant improvement is to incorporate feed-forward control with knowledge about the system, and using the PID only to control error. Alternatively, PIDs can be modified in more minor ways, such as by changing the parameters (either gain scheduling in different use cases or adaptively modifying them based on performance), improving measurement (higher sampling rate, precision, and accuracy, and low-pass filtering if necessary), or cascading multiple PID controllers.

PID controllers, when used alone, can give poor performance when the PID loop gains must be reduced so that the control system does not overshoot, oscillate or *hunt* about the control set-point value. They also have difficulties in the presence of non-linearities, may trade off regulation versus response time, do not react to changing process behaviour (say, the process changes after it has warmed up), and have lag in responding to large disturbances.

Another problem faced with PID controllers is that they are linear, and in particular symmetric. Thus, performance of PID controllers in non-linear systems is variable. In temperature control, a common use case is active heating (via a heating element) but passive cooling (heating off, but no cooling), so overshoot can only be corrected slowly – it cannot be forced downward. In this case the PID should be tuned to be over-damped, to prevent or reduce overshoot, though this reduces performance (it increases settling time).

5.3 RECOMMENDATIONS

PID Controller Algorithms are considered the most efficient and effective of all control Algorithms having successfully overcome the limitations of both the On/Off Control and the Proportional Control Algorithms. But even with its efficiency and effectiveness, we can still see some limitations evident in its control structure. What then is the problem?

The major problem lies with the derivative term: this is due to the fact that small amounts of measurement or process noise can cause large amounts of change in the output. Thus, a low-pass filter can be used in order to remove higher-frequency noise components. However, low-pass filtering and derivative control can cancel each other out, so reducing noise by instrumentation means can be considered a much better choice. Alternatively, a nonlinear median filter may be used, which improves the filtering efficiency and practical performance. In some cases, the differential band can be turned off in many systems with little loss of control. This is equivalent to using the PID controller as a *PI* controller.

Artificial Intelligence may be the key to solving the problems/limitations of the PID Controller and making it the perfect method. A little bit of artificial intelligence added to the PID controlled system that would perform an uninformed search with the aim of gaining some prior knowledge of the model of the process would perfect the PID Controller methodology. The major difficulty and defect of the PID Controller is that is acts or performs instinctively and this could sometimes compromise its role because its parameters are constant and precise and hence making it impossible for it to implement knowledge gained of the process because it has no knowledge of the model. Thus, an Artificial Intelligence system would completely remove this defect and perfect the PID Controller methodology.

REFERENCES

Ahmad Faris Bin Zulkifli, A Project on Automatic Room Temperature Control with Security System, University of Malaysia (May 2009)

American Society of Mechanical Engineers (ASME) http://www.asme.org

Ana Sayfa (2009), Digital Thermometer Project, http://sites.google.com/site/digitalthermometerusingds1620/

Automatic Temperature Control System using RZK, Zilog Technologies and Zilog Developer Studio, http://www.zilog.com

Cytron Technologies: Temperature Control System http://www.youtube.com/watch?v=WsRHuc-SC9w&feature=player_detailpage

Data Sheets, http://www.datasheetarchive.com

Google Search Engine (Temperature Control System and related resources) http://www.google.com

J.L.M. Hensen, Thermal Comfort in transient Conditions, Eindhoven University of Technology (2000)

May Wong, Edward Hettiaratchi, Gautham Jayachandran, Ian Cathers (2001) Temperature Control System; http://www3.fhs.usyd.edu.au/bio/homeostasis/Temp_Control_System.html

Mike Mann (August 2002); Control Systems and Homeostasis; http://www.mann2.wpd

Norhaslinda Binti Hasim, Water Level and Temperature Control using a Programmable Logic Controller (PLC), University of Technology, Malaysia (November 2008)

Nor Mazlee Bin Norazmi, Temperature Control System, University of Malaysia (May 2009)

Omega Technologies Ltd. (Temperature Control System), http://www.omega.com

Proportional Control, http://www.amswers.com

Wise GEEK Encyclopaedia (Temperature Control System and related resources) http://www.wisegeek.com

Wikipedia: The free Encyclopaedia (Temperature Control System and related resources) http://www.wikipedia.com

APPENDIX I

MICROCONTROLLER SOURCE ASSEMBLY LANGUAGE PROGRAM FOR THE TEMPERATURE CONTROL SYSTEM

```
INCLUDE REG_51.PDF
DIS_A  EQU    P0.2
DIS_B  EQU    P0.3
DIS_C  EQU    P0.4
DIS_D  EQU    P0.6
DIS_E  EQU    P0.5
DIS_F  EQU    P0.1
DIS_G  EQU    P0.0
DIS1   EQU    P0.7
DIS2   EQU    P2.7
DIS3   EQU    P2.6
DIS4   EQU    P2.5
ALARM  EQU    P2.4
PLUS   EQU    P1.0
MINUS  EQU    P1.1
SW1    EQU    P1.4
SW2    EQU    P1.5
DQ     EQU    P1.2
CLK    EQU    P1.3
RST    EQU    P1.6
RB0    EQU    000H   ; Select Register Bank 0
RB1    EQU    008H   ; Select Register Bank 1  ...poke to PSW to use

DSEG                 ; this is internal data memory
ORG           20H    ; Bit addressable memory
COUNT: DS     1
SPEED:        DS     1
VALUE_1:      DS     1
VALUE_2:      DS     1
VALUE_3:      DS     1
VALUE_4:      DS     1
NUMB1:        DS     1
NUMB2:        DS     1
NUMB3:        DS     1
NUMB4:        DS     1
TEMP:         DS     1
ALRMTEMP:     DS     1
STACK:        DS     1
CSEG          AT     0        ; RESET VECTOR
```

```
                     PROCESSOR INTERRUPT AND RESET VECTORS
```

```
ORG    00H           ; Reset
JMP    MAIN
ORG    000BH         ; Timer Interrupt0
JMP    REFRESH
```

```
MAIN:

MOV PSW, #RB0        ; Select register bank 0
MOV SP, STACK
MOV SPEED, #00H
MOV COUNT, #00H
MOV NUMB1, #00H
MOV NUMB2, #04H
```

```
MOV NUMB3, #00H
MOV NUMB4, #00H

CLR ALARM
MOV VALUE_1, #15H          ; Switch off all displays
MOV VALUE_2, #15H
MOV VALUE_3, #15H
MOV VALUE_4, #15H
CLR DIS1
CLR DIS2
CLR DIS3
CLR DIS4
MOV TMOD, #01H             ; enable timer0 for scanning
MOV TL0, #00H
MOV TH0, #0FDH
SETB ET0
SETB EA
SETB TR0                   ; Start the Timer

SETB CLK                   ; Start with CLK equal to 1
ACALL CONFIGURE            ; Configure NTC THERMISTOR

; wait 10 MS for Configuration to be written
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS
       ACALL DELAYMS

UPP:   ACALL START_CONVERT        ; Send command to start temperature conversion
       ACALL DELAYS
       ACALL READ_TEMPERATURE     ; Get Temperature Reading111

MOV A, R3
MOV B, #02H
DIV AB
MOV R4, B
CJNE R4, #01H, GFG1
MOV VALUE_4, #05H
AJMP GFG2

GFG1:  MOV VALUE_4, #00H
GFG2:  MOV R2, A
       MOV R1, #00H

       MOV R3, #00D
       MOV R4, #00D
       MOV R5, #00D
       MOV R6, #00D
       MOV R7, #00D

       CALL HEX2BCD

              MOV VALUE_3, R3
              MOV VALUE_2, R4
              MOV VALUE_1, R5

              MOV A, R3
              XRL A, NUMB3
              JNZ EDE
              MOV A, R4
              XRL A, NUMB2
```

```
        JNZ EDE
        MOV A, R5
        XRL A, NUMB1
        JNZ EDE
        MOV A, NUMB4
        XRL A, VALUE_4
        JNZ EDE
        SETB ALARM
        AJMP SXZ

  EDE:  MOV A, VALUE_1
        SWAP A
        ORL A, VALUE_2
        MOV R1, A
        MOV A, VALUE_3
        SWAP A
        ORL A, VALUE_4
        MOV R5, A

        MOV A, NUMB1
        SWAP A
        ORL A, NUMB2
        MOV R3, A
        MOV A, NUMB3
        SWAP A
        ORL A, NUMB4
        MOV R6, A

        MOV A, R3
        CLR C
        SUBB A, R1
        JZ DFD1
        JNC DFD
        SETB ALARM
        AJMP SXZ
  DFD1: MOV A, R6
        CLR C
        SUBB A, R5
        JNC DFD

        SETB ALARM
        AJMP SXZ
  DFD:  CLR ALARM
  SXZ:
        SETB SW1
        JNB SW1, SHOW_TEMP
        AJMP UPP
```

```
SHOW_TEMP:
        CLR ALARM
        CALL DELAY
        JNB SW1, $
  SXD1: MOV VALUE_1, NUMB1
        MOV VALUE_2, NUMB2
        MOV VALUE_3, NUMB3
        MOV VALUE_4, NUMB4
        SETB SW2
        JNB SW2, UPP1
        SETB PLUS
        SETB MINUS
        JNB PLUS, INC_TEMP
        JNB MINUS, DEC_TEMP
        AJMP SXD1
UPP1:
        CALL DELAY
```

```
                JNB SW2, $
                AJMP UPP
INC_TEMP:
                CALL DELAY
                JNB PLUS, $
                MOV R5, NUMB4
                CJNE R5, #00H, FGG1
                MOV R5, NUMB3
                CJNE R5, #05H, FGG12
                MOV R5, NUMB2
                CJNE R5, #02H, FGG12
                MOV R5, NUMB1
                CJNE R5, #01H, FGG12
                AJMP SXD1
FGG12:
                MOV NUMB4, #05H
                AJMP SXD1
FGG1:
                MOV NUMB4, #00H
                INC NUMB3
                MOV R5, NUMB3
                CJNE R5, #0AH, SXD1
                INC NUMB2
                MOV NUMB3, #00H
                MOV R5, NUMB2
                CJNE R5, #0AH, SXD1
                INC NUMB1
                MOV NUMB2, #00H
                AJMP SXD1

DEC_TEMP:
                CALL DELAY
                JNB MINUS, $
                MOV R5, NUMB4
                CJNE R5, #00H, FG1
                MOV R5, NUMB3
                CJNE R5, #00H, FG12
                MOV R5, NUMB2
                CJNE R5, #00H, FG12
                MOV R5, NUMB1
                CJNE R5, #00H, FG12
                AJMP SXD1
FG1:
                MOV NUMB4, #00H
                AJMP SXD1
FG12:
                MOV NUMB4, #05H
                MOV R5, NUMB3
                CJNE R5, #00H, DXC1
                MOV NUMB3, #09H
                MOV R5, NUMB2
                CJNE R5, #00H, DXC2
                MOV NUMB2, #09H
                MOV NUMB1, #00H
                AJMP SXD1
DXC1:
                DEC NUMB3
                AJMP SXD1
DXC2:
                DEC NUMB2
                AJMP SXD1
```

```
HEX2BCD:
                MOV B, #10D
                MOV A, R2
```

```
            DIV AB
            MOV R3, B
            MOV B, #10                    ; R7, R6, R5, R4, R3
            DIV AB
            MOV R4, B
            MOV R5, A
            CJNE R1, #0H, HIGH_BYTE       ; CHECK FOR HIGH BYTE
            SJMP END
HIGH_BYTE:
            MOV   A, #6
            ADD   A, R3
            MOV   B, #10
            DIV   AB
            MOV   R3, B
            ADD   A, #5
            ADD   A, R4
            MOV   B, #10
            DIV   AB
            MOV   R4, B
            ADD   A, #2
            ADD   A, R5
            MOV   B, #10
            DIV   AB
            MOV   R5, B
            CJNE R6, #00D, ADD_IT
            SJMP CONTINUE
ADD_IT:
            ADD A, R6
CONTINUE:
            MOV R6, A
            DJNZ R1, HIGH_BYTE
            MOV B, # 10D
            MOV A, R6
            DIV AB
            MOV R6, B
            MOV R7, A
END:        RET
```

```
; This routine writes the value in A to the NTC THERMISTOR

            WRITE1620:
            MOV R0, #08H        ; Set Counter for 8 bits

NEXTBITWRITE:
            CLR CLK             ; Start clock cycle
            RRC A               ; Rotate 'A' Right into Carry Bit (Lowest Bit in A goes to C)
            MOV DQ, C           ; Move outgoing bit to DQ
            SETB CLK            ; Rising Edge of Clock makes one clock cycle
            DJNZ R0, NEXTBITWRITE
            RET
```

```
; This routine reads a value from the NTC THERMISTOR and puts it in A

            READ1620:
            MOV R0, #08H        ; Set Counter for 8 bits
            SETB DQ             ; Set DQ to 1 to enable it as an input pin

NEXTBITREAD:
            CLR CLK             ; Start clock cycle
            MOV C, DQ           ; Move incoming bit to DQ
            SETB CLK            ; Rising Edge of Clock makes one clock cycle
            RRC A               ; Rotate 'A' Right through Carry Bit(C goes to Highest Bit of A)
            DJNZ R0, NEXTBITREAD
            CLR DQ
            RET
```

```
; Routine to Configure NTC THERMISTOR

CONFIGURE:
             SETB RST            ; Make 1620 reset go 'high' to start transfer
             MOV A, #0CH         ; Send the 'Write Config' command to 1620
             ACALL WRITE1620
             MOV A, #00001010B   ; CPU = 1, 1Shot = 0, THF = 0, TLF = 0
             ACALL WRITE1620     ; Send the Configuration Byte
             CLR RST             ; Make 1620 reset go Low to signal end of transfer
             RET
```

```
; Routine to Start Temperature Conversion on 1620

START_CONVERT:
             SETB RST            ; Make 1620 reset go 'high' to start transfer
             MOV A, #0EEH        ; Send the "START CONVERT" command to 1620
             ACALL WRITE1620
             CLR RST             ; Make 1620 reset go Low to signal end of transfer
             RET
```

```
; Routine to Read Temperature from 1620

READ_TEMPERATURE:
             SETB RST
             MOV A, #0AAH        ; Send the "Read Temperature" command
             ACALL WRITE1620
             ACALL READ1620      ; Get first byte of temperature
             MOV R3, A           ; Store Byte in R1
             ACALL READ1620      ; Get second byte of temperature
             MOV R1, A           ; Store Byte in R2
             CLR RST             ; End Transfer
             RET
```

```
; Routine to Write Temperature High 1620

WRITE_HIGH:
             SETB RST
             MOV A, #01H         ; Send the "Read Temperature" command
             ACALL WRITE1620
             MOV A, R3           ; Store Byte in R1
             ACALL WRITE1620
             MOV A, R1           ; Store Byte in R1
             ACALL WRITE1620
             CLR RST             ; End Transfer
             RET
```

```
; Routine to Read Temperature from 1620

READ_HIGH:
             SETB RST
             MOV A, #0A1H        ; Send the "Read Temperature" command
             ACALL WRITE1620
             ACALL READ1620      ; Get first byte of temperature
             MOV R3, A           ; Store Byte in R1
             ACALL READ1620      ; Get second byte of temperature
             MOV R1, A           ; Store Byte in R2
             CLR RST             ; End Transfer
             RET
```

```
; One second delay routine

DELAYS:
              MOV R6, #00H         ; put 0 in register R6 (R6 = 0)
              MOV R5, #004H        ; put 5 in register R5 (R5 = 4)
LOOPB:
              INC R6               ; increase R6 by one (R6 = R6 +1)

ACALL DELAYMS                      ; call the routine above. It will run and return to
here.
              MOV A, R6            ; move value in R6 to A
              JNZ LOOPB            ; if A is not 0, go to LOOPB
              DEC R5               ; decrease R5 by one. (R5 = R5 -1)
              MOV A, R5            ; move value in R5 to A
              JNZ LOOPB            ; if A is not 0 then go to LOOPB.
              RET
```

```
; Millisecond delay routine

DELAYMS:

              MOV R7, #00H                ; put value of 0 in register R7
LOOPA:
              INC R7                      ; increase R7 by one (R7 = R7 +1)
              MOV A, R7                   ; move value in R7 to Accumulator (also known as A)
              CJNE A, #0FFH, LOOPA        ; compare A to FF hex (256). If not equal go to LOOPA
              RET                         ; return to the point that this routine was called from
```

INTRRUPT ROUTINE TO REFRESH THE DISPLAY

```
REFRESH:
              PUSH   PSW                  ; save current register set
              MOV    PSW, #RB1
              PUSH   ACC
              INC COUNT
              MOV R4, COUNT
QA1:
              CJNE R4, #01H, QA2
              MOV SPEED, VALUE_1
              SETB DIS1
              CLR DIS2
              CLR DIS3
              CLR DIS4
              CALL DISP
              AJMP DOWN
QA2:
              CJNE R4, #02H, QA3
              MOV SPEED, VALUE_2
              CLR DIS1
              SETB DIS2
              CLR DIS3
              CLR DIS4
              CALL DISP
              AJMP DOWN
QA3:
              CJNE R4, #03H, QA4
              MOV SPEED, VALUE_3
              CLR DIS1
              CLR DIS2
              SETB DIS3
              CLR DIS4
              CALL DISP
              AJMP DOWN
```

```
QA4:
             CJNE R4, #04H, QA5
             MOV SPEED, VALUE_4
             CLR DIS1
             CLR DIS2
             CLR DIS3
             SETB DIS4
             CALL DISP
             AJMP DOWN
QA5:
             MOV COUNT, #01H
             MOV R4, COUNT
             AJMP QA1
DOWN:
             MOV TL0, #0FFH
             MOV TH0, #0F0H
             POP    ACC
             POP    PSW
             RET
```

```
FLASHING:
             CALL DELAY                  ; Display on/off for 2 times
             CALL DELAY
             MOV VALUE_1, #16H
             MOV VALUE_2, #16H
             MOV VALUE_3, #16H
             MOV VALUE_4, #16H
             CALL DELAY
             CALL DELAY
             MOV VALUE_1, NUMB1
             MOV VALUE_2, NUMB2
             MOV VALUE_3, NUMB3
             MOV VALUE_4, NUMB4
             CALL DELAY                  ; Display on-off for 2 times
             CALL DELAY
             MOV VALUE_1, #16H
             MOV VALUE_2, #16H
             MOV VALUE_3, #16H
             MOV VALUE_4, #16H
             CALL DELAY
             CALL DELAY
             MOV VALUE_1, NUMB1
             MOV VALUE_2, NUMB2
             MOV VALUE_3, NUMB3
             MOV VALUE_4, NUMB4
```

```
DELAY:
             MOV R1, #0FFH
REPA2:
             MOV R2, #0FFH
REPA1:
             NOP
             DJNZ R2, REPA1
             DJNZ R1, REPA2
             RET
             END
```

APPENDIX II

LIST OF PROJECT MATERIALS

- A Thermistor
- Resistors
- An Operational Amplifier
- Diodes
- Transistors
- A NAND Gate
- Capacitors
- Wires
- ATMEL AT89C52 Microcontroller
- A Crystal Oscillator
- 555 Timer
- A Buzzer
- Inductors
- A MOSFET
- An LCD Display
- An LED

- A Voltage Regulator
- A 12v Transformer
- A 240v Relay
- Variable Resistors
- Soldering Wire and Iron