IOT Project Report

Temperature Controller Using Arduino Microcontroller

A Project Report Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Bachelor of Technology

In

Computer Science Department

By

Indradhar Paka
Sai Venkat Lakamsani
Raghavendra Naga Varma

CSE-III
3rd year

SCHOOL OF ENGINEERING AND TECHNOLOGY

BML MUNJAL UNIVERSITY GURGAON

November,2020

TABLE OF CONTENTS

SUBMITTED TO

Dr. Goldie Gabrani
Professor
Computer Science and Engineering Department
School of Engineering and Technology

PARTICULARS OF STUDENTS:

S No.	Name	Roll No.
1.	Indradhar Paka	1800315C203
2.	Raghavendra Naga Varma	1800241C203
3.	Sai Venkat Lakamsani	1800254C203

ABSTRACT

Internet of Things is an essential course in computer science curriculum, which helps students to develop a mental model of how networks and connecting devices work. Interfacing Arduino with daily life applications and large-scale plants are often complex, non-deterministic which makes them difficult for students to understand. One such concept is temperature controlling using Arduino microcontroller.

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

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

PROBLEM STATEMENT AND MOTIVATION

“Temperature controlling using Arduino microcontroller”

In our project we are trying to achieve an optimal environment for home/plant working conditions so that the temperature we wish to work in is always achieved and maintained irrespective of changes in the external environment. The system is implemented using the Arduino microcontroller, a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

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

INTRODUCTION AND DESCRIPTION

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

Industrial and control application may require automation of the process such as temperature, pressure, liquid flow, etc., in order to minimize manual intervention. To automate any application an intelligent processor plays a major role. One such processor proposed for the project is Arduino. The temperature controller can be used to control the temperature of any plant. Typically, it contains a Processor unit, Temperature input unit and Control output unit. The Arduino based motherboard forms the processing unit. In the temperature unit, according to program it processes the analog signal into digital and forms a particular voltage level for a particular temperature. The relay driver forms the control output unit. Electric power to the heating element (coil) is supplied through relay contacts. The switching ON/OFF the relay controls the heat supplied to the plant.

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

PROBLEMS FACED

The main problem that we faced during the initial stage was programming in Arduino according to our requirement and no physical interaction with the components of Arduino. We could have used any laboratory work for better experience of the work.

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

DETAILED EXPLANATION

The plant may be an industry or an equipment or a furnace for which the temperature must be monitored and maintained continuously at a particular temperature. But Arduino has in built with many components like analog to digital converter, clock of 16 MHz, shift registers. In this system we use temperature sensor DHT22, to use to detect temperature into appropriate voltage. This voltage is given to Arduino. According to program it processes the analog signal into digital and forms a particular voltage level for a particular temperature. Used 16x2 LCD is used to display the output i.e. surrounding temperature of DHT22 in both degree centigrade and Fahrenheit units. At the same time, it also sends the data to Relay, if the temperature becomes maximum from set point relay becomes activate and it switches on the cooling device like fan. In this manner it monitors and controls the temperature.

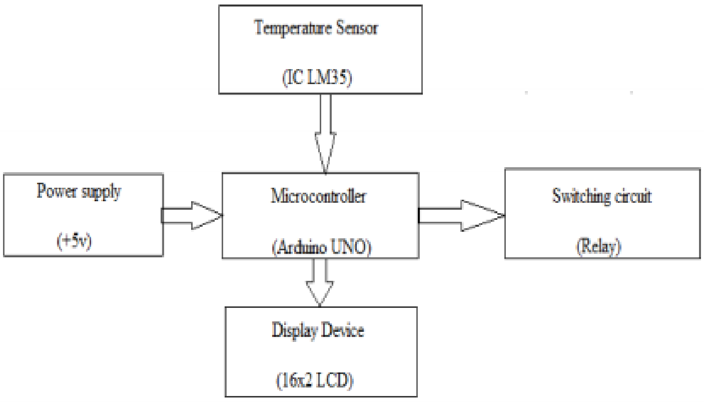

Figure 1 Block diagram for the model

Arduino

The Arduino is a microcontroller board based on the ATmega8. It has 14 digital -input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

Figure 2 Arduino UNO board

Temperature Sensor

The DHT-22 (also named as AM2302) is a digital-output, relative humidity, and temperature sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and sends a digital signal on the data pin.

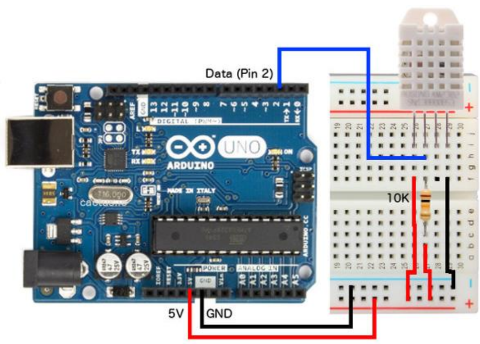

Figure 3 Configuring DHT 22 to arduino

Interfacing 16x2 LCD to Arduino

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs.

The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on. A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix.

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

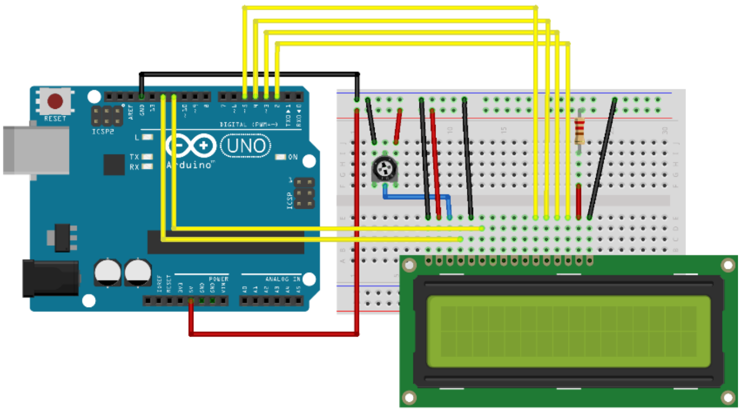

Figure 4 Interfacing LCD to Arduino

Relay Circuit

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

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

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

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

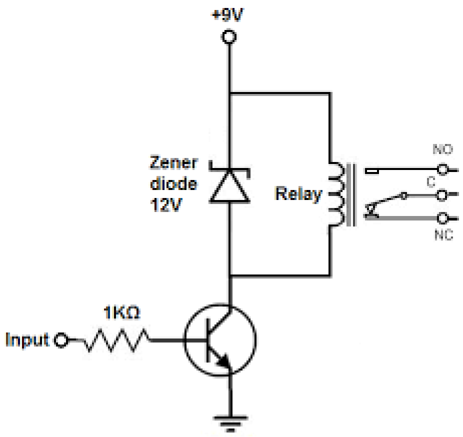

Figure 5 Relay Circuit

Keypad for Manual Input

The keypad is a set of buttons arranged in rows and columns (called matrix). Each button is called key Keypad has various types. Keypad pins are divided into two groups: row and column.

8 pins: 4 row-pins (R1, R2, R3, R4) and 4 column-pin (C1, C2, C3, C4). The process of detecting the key pressing is called scanning keypad. It is called “scanning” because it checks one key by one key. Row-pins are connected to Arduino's output pins. Column pins are connected to Arduino's input pins, in this state; the value of the input pin is HIGH if the key is not pressed). If we each key as an independent button, it requires 16 Arduino pins for 16 keys plus GND pin.

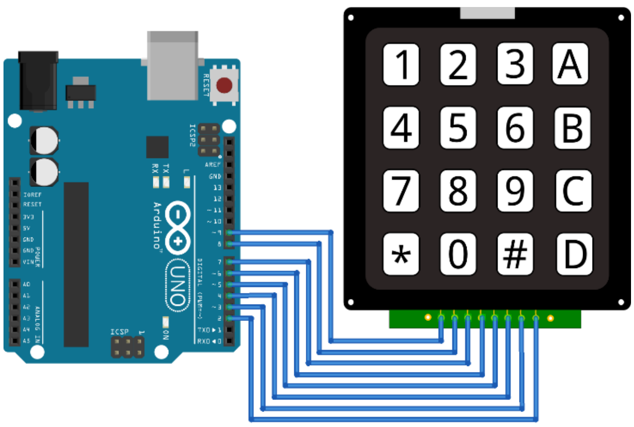

Figure 6 Interfacing 4x4 keypad to Arduino

Combining Everything for Implementation

We used temperature sensor DHT22. It generates a small voltage corresponds to the temperature. This generated voltage is in the continuous, analog form. This voltage is fed to the controller unit. Here we use Arduino microcontroller as a controller.

This voltage is given to the Analog port 0 (A0) of the Arduino UNO. Arduino UNO reads analog input and converts this analog voltage into digital bits form using inbuilt A to D converter. It converts analog voltage level in any number between 0 to 1023. It uses 10 bits for processing.

This is given to the microcontroller; it multiplies the digital data with coefficient 0.488 and converts this voltage in particular value. This value is nothing but the temperature in degree centigrade. Similarly, we multiply the data with 1.8 and the add 32 to convert voltage level into Fahrenheit unit.

This data sending out on the digital port (2, 3, 4, 5). Consequently, Arduino Uno sends out data for displaying on 16x2 LCD. Pin 1 is connected to ground and pin 2 is connected to V_{cc} through Arduino for activating or switching ON the LCD. On pin 3 a 10k ohms for adjusting the brightness of LCD screen. Register Select (RS) and Enable pin is connected to pin 12 and 11 respectively for communication with Arduino.

LCD displays temperature in both unit's degree Celsius and also in degree Fahrenheit. At the same time Arduino sends control bit 0 or 1 on the digital port 6. This bit is used for controlling part. For bit 0 Arduino sends 0V and for bit 1 it sends 5V at the output.

A relay is connected with relay circuit to digital port 6. If the temperature is less than desired or set temperature Arduino gives logic low level to the pin 6. But when current temperature goes just or more above the set level it sends logic high level to the digital pin 6.

Input of the temperature value is taken from the keypad which is interfaced with board. The program runs continuously and updates the microcontroller which changes the relay outputs.

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

TECHNICAL FEATURES AND ELEMENTS

We can divide the hardware broadly into:

Temperature Input Unit	Comprises of DHT22
Processing Unit	The Main Processing Unit i.e. the Arduino
Control Output Unit	Relays and Output Control and Read-Out

The Hardware used is comprises of the following:

- Arduino microcontroller motherboard.
- Jumping wires
- 16x2 LCD
- Temperature Sensor DHT22
- SPDT Relay
- BC557 PNP BJT Transistor
- 10k Resistor and potentiometer
- Diodes (1N4004)

TEMPERATURE SENSORS:

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

- Thermistor
- Thermocouple
- Solid State Temperature Sensor
- IC temperature sensor (LM-135) (voltage output)
- LM-336
- DHT11, DHT22

Features of DHT22:

- Temperature Range from -40°C-125°C
- Humidity Range 0-100%
- Operating Voltage: 3.5V to 5.5V.
- Operating current: 0.3mA (measuring) 60uA (standby)
- Output: Serial data.
- Resolution: Temperature and Humidity both are 16-bit.
- Accuracy: ±0.5°C and ±1%

EXISTING STATE OF THE ART

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

Problem Associated with Existing System

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

- It takes comparatively more time to process.
- It requires additional devices for operation.
- It requires external clock.
- Programming for microcontroller 8051 is difficult.
- For programming it requires development system.
- Circuit size becomes large.
- PCB making becomes complex, difficult and tedious.

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

S No.	Existing State of the art	Drawbacks in existing state of the art	Overcome
1	Control by temperature change response fan based on Arduino CN205533372U.	The utility model relates to a technical field of fan discloses a control by temperature change response fan based on Arduino, including the controller, by power source.	We can implement the same procedure for the large-scale plants such as nuclear reactors or temperature dependent explosives, with the help of Wi-Fi Sensor connected to a webserver.
2	Automatic Temperature Based Fan Speed Controller Using Arduino. Mohite et al., 2020	Non-Patent Literature, Controlling the rpm or speed of the fan based on the temperature around the sensor. Human intervention is needed partially.	Similarly, it can be used in many areas such as nuclear reactors or thermal plants etc., Where the program runs in the web server which completely removes human intervention. We can also perform ML/AI algorithm on the data provided by the Wi-Fi sensor and predict the severe situation if occurs any. And can be cautious about the impact which will get occur in the future.

BLOCK DIAGRAM AND FLOW CHART

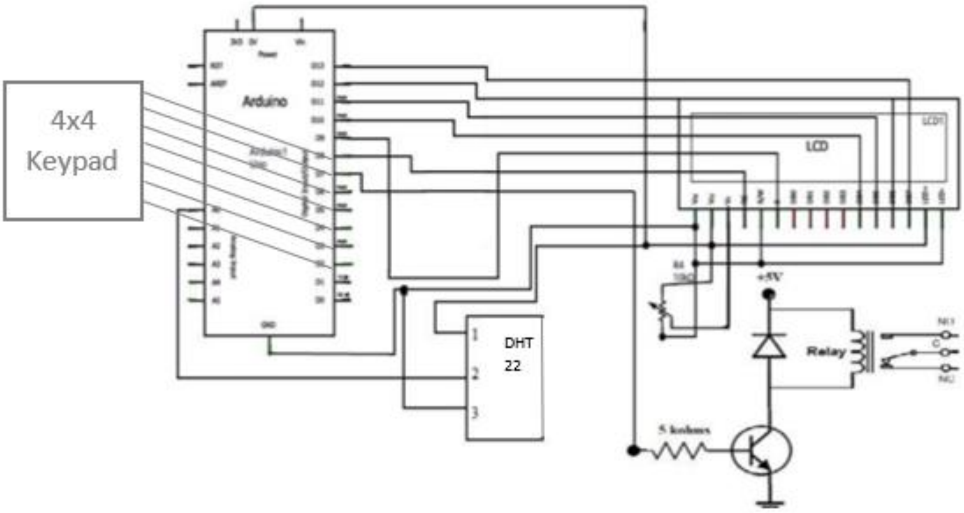

Figure 7 Circuit diagram for temperature Controller system

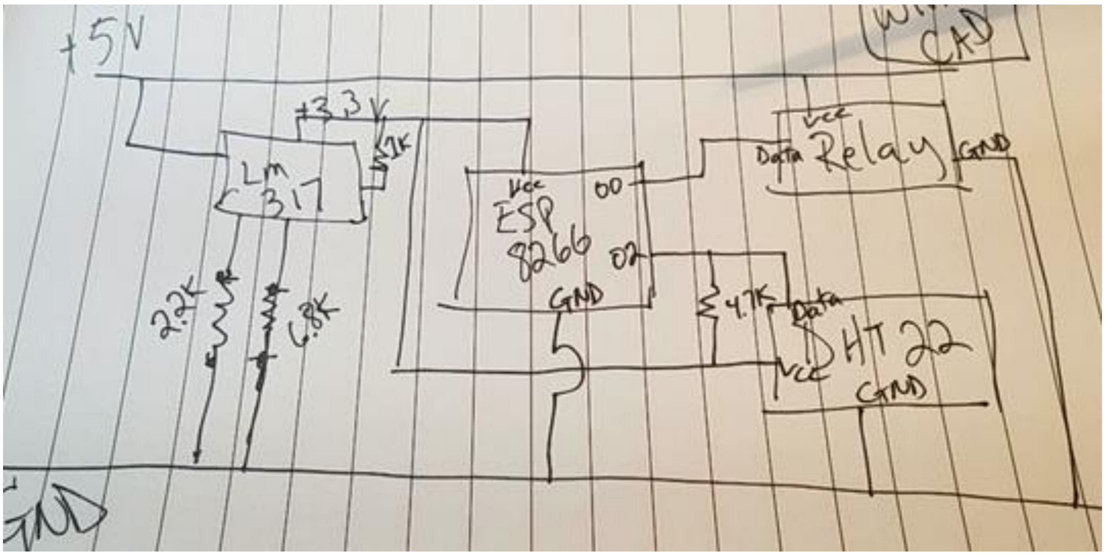

Figure 8 Interfacing ESP8266 (NodeMCU) with temperature sensors and Relay circuits

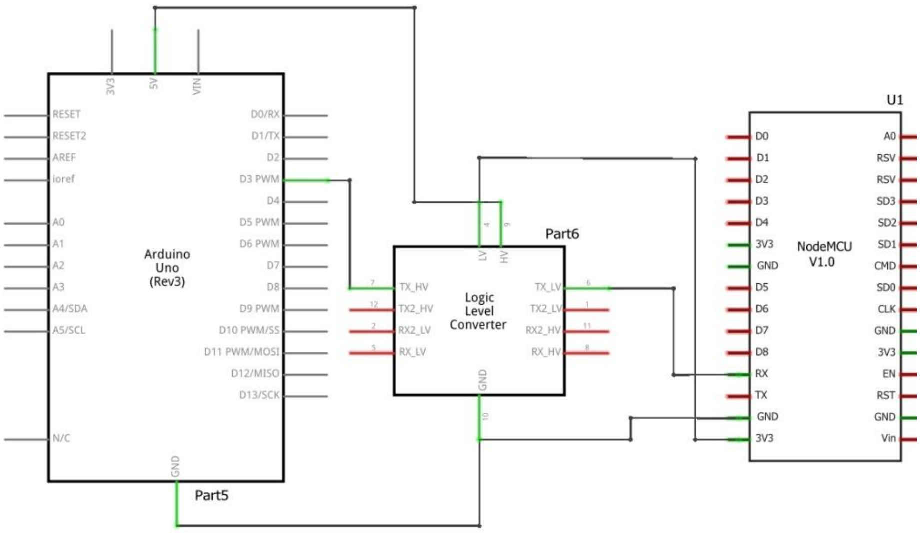

Figure 9 Interfacing NodeMCU with Arduino UNO

FLOWCHART:

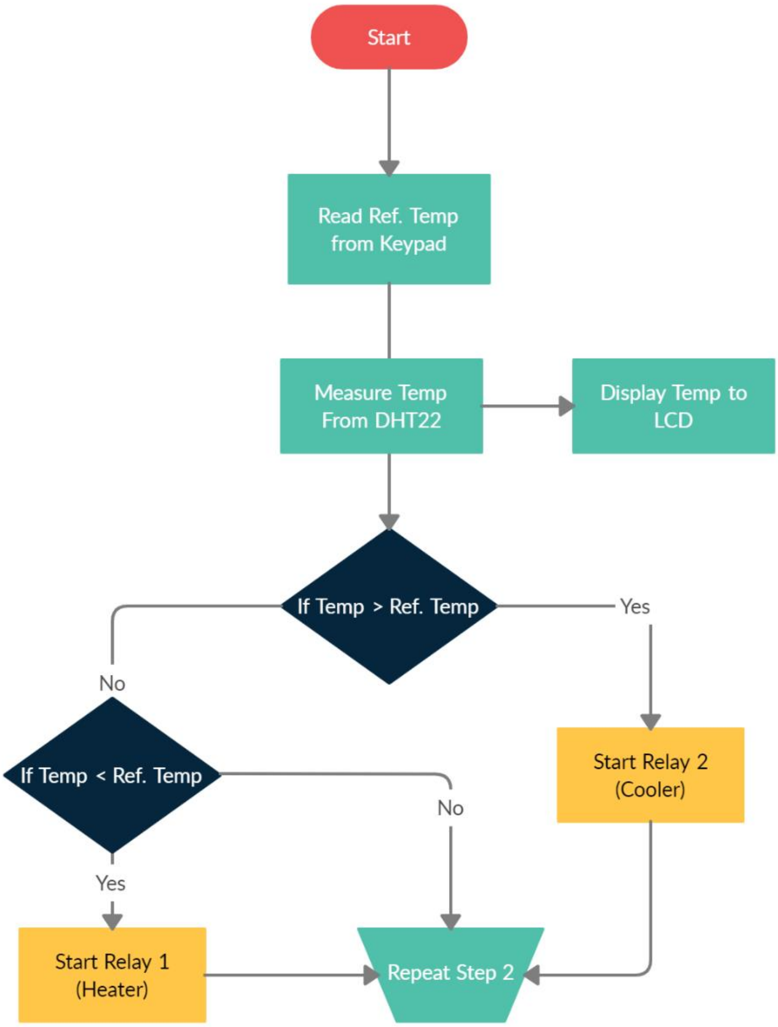

Figure 10 Flow Chart for Temperature Controller System

DISTINGUISHABLE FEATURES

This project is also created by by using several microprocessors. But those are dependent on the technologies. Our project uses Arduino microcontrolle. It would be lesser cost and efficient way to control temperature in plants etc.

ALTERNATE SOLUTION AND FURTHER SCOPE OF PROJECT

There may be many alternate implementations of this project. People may use technologies in which they are comfortable and can implement the project using technologies such as 8051, Raspberry, 8086 M.P etc. We initially thought of doing this project on microprocessors. But it is quite difficult to interface these components to the microprocessor.

We can also measure liquid flowing through a pipe or container. The Arduino flow meter works on the principle of the Hall effect According to the Hall effect, a voltage difference is induced in a conductor transverse to the electric current and the magnetic field perpendicular to it. Here, the Hall effect is utilized in the flow meter using a small fan/propeller-shaped rotor, which is placed in the path of the liquid flowing. we can use it in your gardening system, or interface it with an LCD display for other applications that require you to measure water flow rate and quantity.

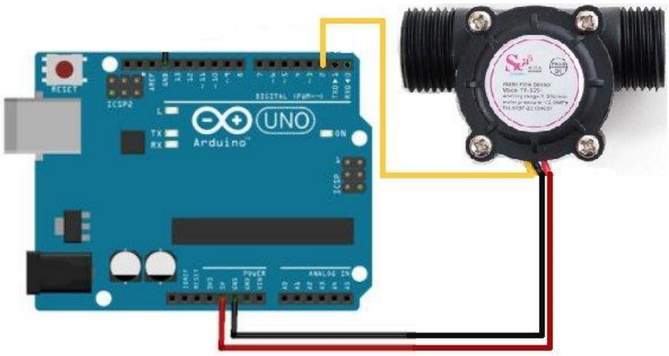

Figure 11 Arduino flow rate sensor

ALGORITHM AND PROGRAM

Algorithm:

- On start, Reference temperature is read from the keypad to Arduino.
- Measure the temperature using DHT22 and send to Arduino board.
- Run in the loop if the temperature measured is in the range of reference temperature.
- Relay1(Heater) is called if measured temperature is less than reference temperature.
- Relay2(cooler) is called if measured temperature is more than reference temperature.
- Display the temperature and Humidity to the LCD display.

Program:
Since the code is little larger and insufficient to display it here. We have uploaded the code in GitHub. To redirect to program for the project kindly refer the link.
https://github.com/indradhar/Temperature_Controller

CONCLUSION

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

This project gave also improved the understanding microcontroller concepts and to implement various functionalities of IOT concepts.

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

REFERENCES:

[1] Nwankwo Nonso Prince , Alumona Theophilus , Onwuzulike Daniel .A., Nwankwo Vincent, "Design and Implementation of Microcontroller based Automatic Fan Speed Regulator (Using Temperature Sensor)", International Journal of Engineering Research and Management (IJERM), vol. 01, no.5, pp. 202-208, August 2014.

[2] Aakanksha Pimpalgaonkar, Mansi Jha, Nikita Shukla, Kajol Asthana, "A Precision Temperature Controller using Embedded System", International Journal of Scientific and Research Publications, vol.3, no.12, pp.1-3, December 2013.

[3] A.L.Amoo, H. A. Gud, H. A. Sambo, T. L. G. Soh, "Design and Implementation of a Room Temperature Control System: Microcontroller-Based", IEEE Trans. Energy Conservation , 2014.

[4] S. V. Devika, Sk. Khamuruddeen, Sk. Khamurunnisa, J. Thota and K. Shaik, Arduino Based Automatic Plant Watering System, IJARCSSE, vol 4, issue 10, pp. 449-456, 2014

[5] James S. McDonald, "Temperature Control Using a Microcontroller: An Interdisciplinary Undergraduate Engineering Design Project", 1997 Frontiers in Education Conference, pp. 1620-1624, 1997 IEEE.