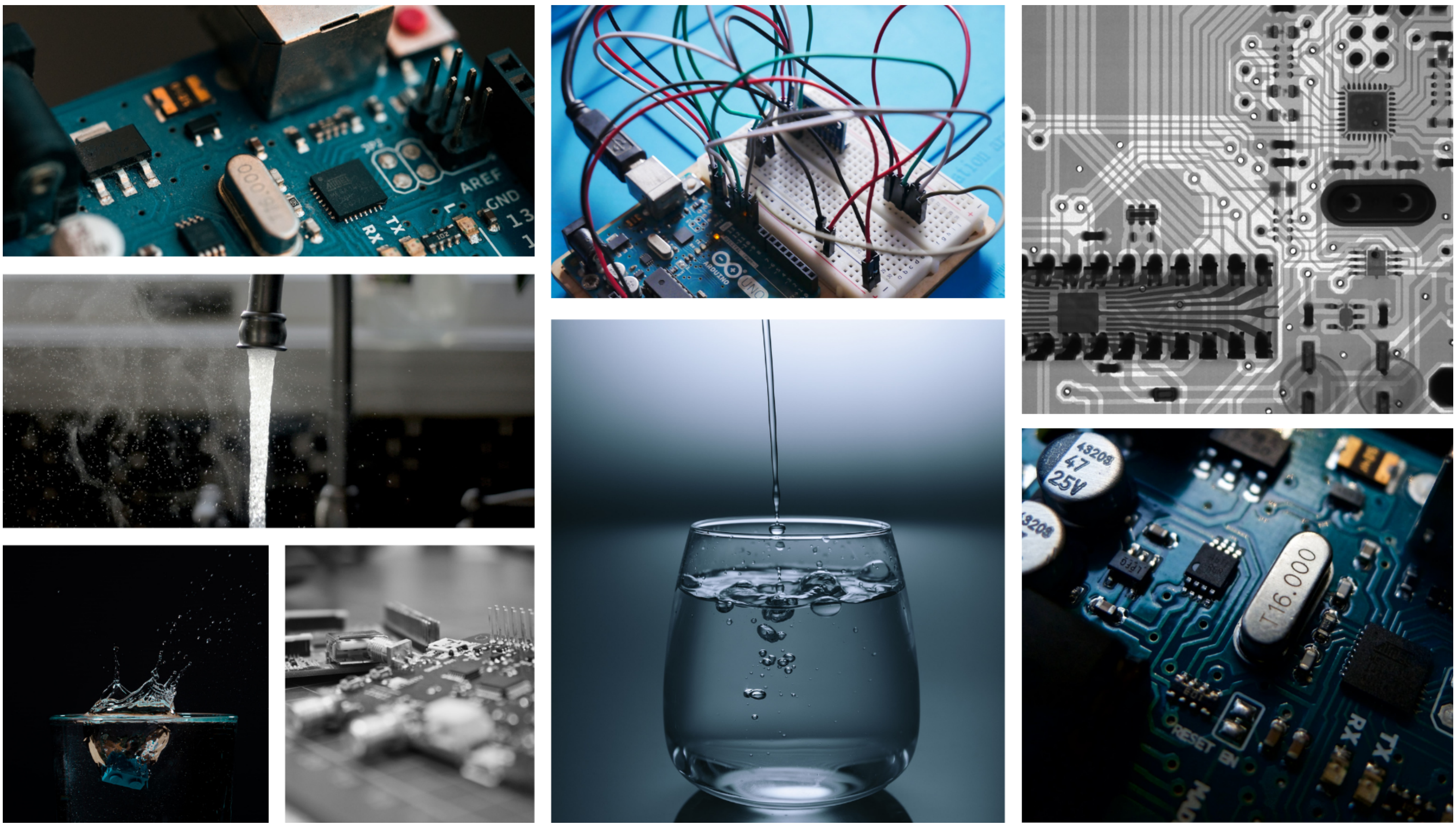
AS2019907

G.L.T.T.DASANAYAKA

SMART WATER DISPENSER

FINAL YEAR PROJECT REPORT

ICT 305 2.0
EMBEDDED SYSTEMS



ACKNOWLEDGMENTS

I would like to place on record my deep sense of gratitude to Dr.M.D.R.Perera , Senior lecturer in Computer Science, Department of Computer Science, Faculty of Applied Science, University of Sri Jayewardenepura, Gangodawila, Nugegoda, Sri Lanka for his generous guidance, help and useful suggestions. I express my sincere support for the embedded system project.

CONTENT

1. ABSTRACT

The Internet of Things (IoT) is transforming human life into a smart world due to its rapid expansion. To make people's lives easier, physical objects that are connected to smart sensors give data. Project presents a case study of smart water dispensers with the help of NTC thermistor temperature sensors, and Arduino is designed to dispense water according to the user's preferred temperature. The smart water dispenser reads the two temperatures of the hot water tank and the cold water tank of the dispenser and checks that the desired temperature is within the range of the hot water tank temperature and the cold water tank temperature. It calculates the volume of cold water and the volume of hot water to be delivered in relation to the required temperature of the water.

The system is capable of fully automated water dispensing using DC water pumps and sensors. It uses an NTC thermistor temperature sensor to read the temperatures of hot water tank and cold water tank. The system uses a relay to control two dc water pumps and an electric water heater.

2. INTRODUCTION

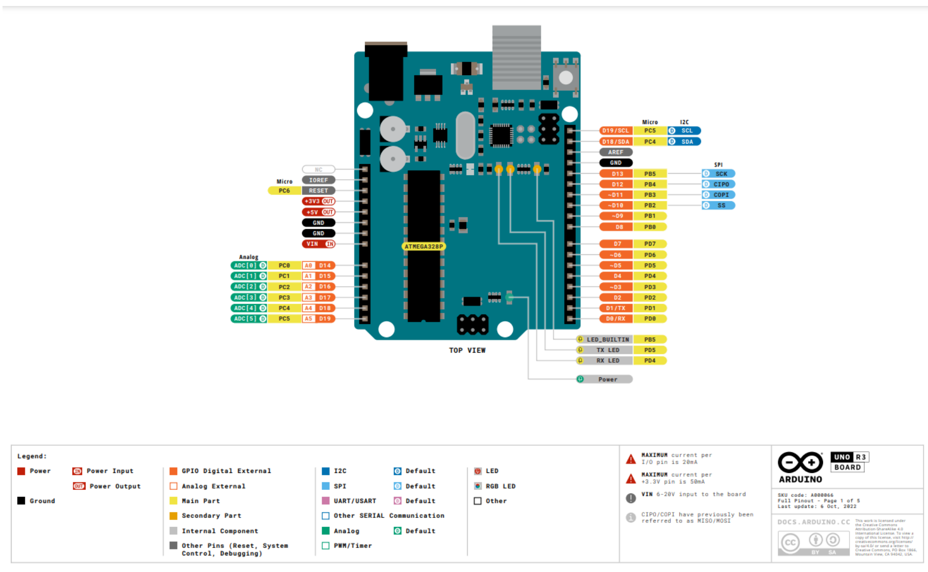
The Internet of things, or IoT, is a system that consists of networked, uniquely identifiable devices, including people and animals, that can send data across a network without human participation. Transportation, healthcare, and smart homes are a few systems that are related. Environmental smart devices can function based on contextual awareness, which is not possible with manual monitoring.

The ideal water temperature is typically chilled cold (6C°/43F) for optimum refreshment or room temperature (20C°/68F) for maximum flavour. Nevertheless, this temperature may vary depending on our preferences and why we use it or drink it. It may be 100C° for coffee, 90C° for green tea ,70C° for baby formula milk, 40C° for bakery etc. Sometimes waiting for the water to boil or waiting for the water to cool is a waste of time. Consequently, we may reduce this time to a few seconds using an instantaneous water supply. Even though there are several instant water dispensers on the market that can deliver a variety of temperatures and volumes, each dispenser only offers a limited range of temperature and volume options. It frequently has a temperature range of only 5 or 6 values. The user's demand is therefore restricted to the provided values. But there are many different user requirements. An automatic instant water dispenser will be built as a fix for this issue. The user is free to select the desired temperature because it can supply any temperature.

3. COMPONENTS

1. Arduino UNO R3

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



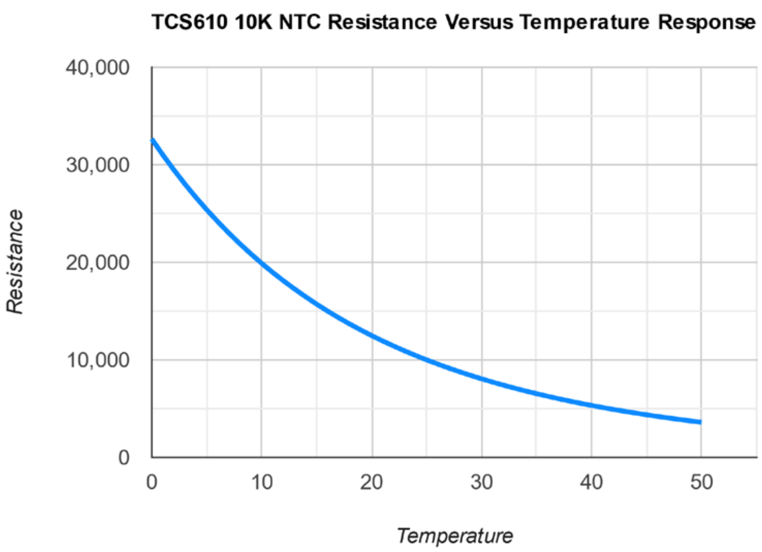
2. NTC 10K ohm 1% Thermistor Temperature Sensor with 1m Cable (MD037)

Main Features

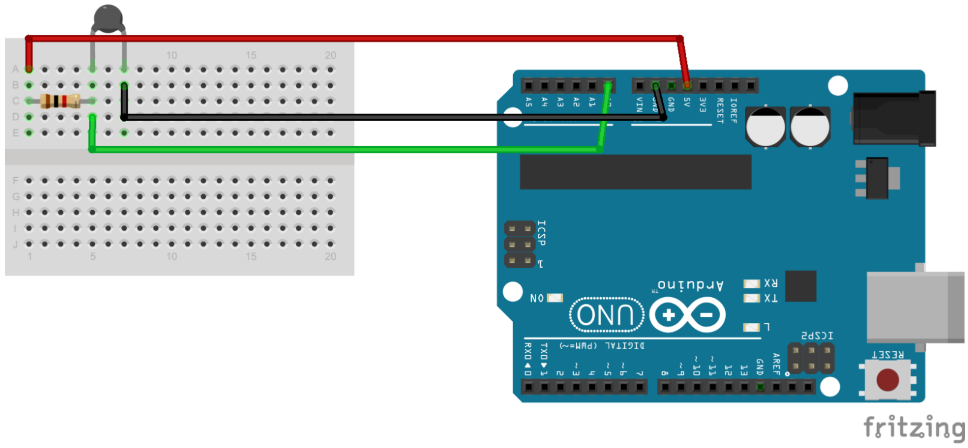
- Stainless steel sheath and waterproof
- Measurement range: -20 to 105 °C
- Length of wire: 1 metre
- Size of probe: 5 x 25 mm
- Output: 2 wires
- Type: NTC 10k±1950

- measure environmental temperatures using Arduino ADC and this 10k NTC LINK
- Resistance to temperature conversion table LINK
- B-constant : 3380K -/+ 1%
- Typical Dissipation Constant 5mW/ °C
- Probe insulation: >100MOhm
- Peak Voltage sustain time: 2 seconds, AC1800V 1mA 2 seconds
- Stress sustain: 9.8N (1kgF) for 1 minute no deformation

With NTC thermistors, resistance decreases as temperature rises, due to an increase in the number of conduction electrons energised by the thermal agitation from the valence band. NTC are commonly used as temperature sensors or in series with circuits like power supplies as an inrush current limiter.



Here is the Resistance Versus Temperature Response graph for the TCS610 10K NTC from wavelength electronics. You can see that at 25ºC the Resistance of the NTC thermistor is 10k Ohms. When the temperature increases the resistance decreases.

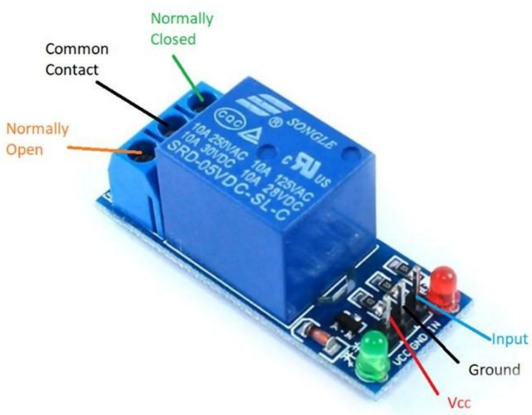


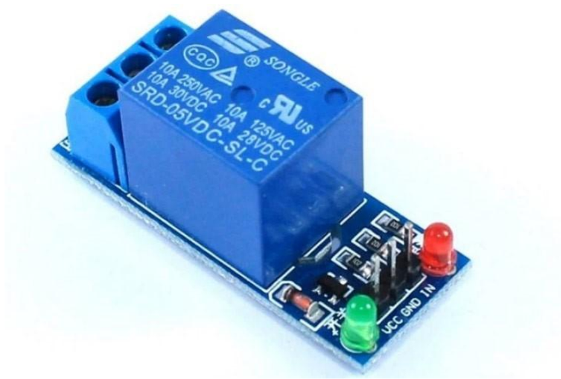
3. 3-6V DC Micro Submersible Mini Water Pump

Main Features

- Voltage: DC 3-6V
- Current: 100-200mA
- Lift: 0.3-0.8 metres
- Flow: 1.2-1.6L/ minutes
- Outside diameter of water outlet: 7.45mm / 0.3″
- Inside diameter of water outlet: 4.7mm / 0.18″
- Diameter: Approx. 24mm / 0.95″
- Length: Approx. 45mm / 1.8″
- Height: Approx. 33mm / 1.30″

4. 5V Single-Channel Relay Module

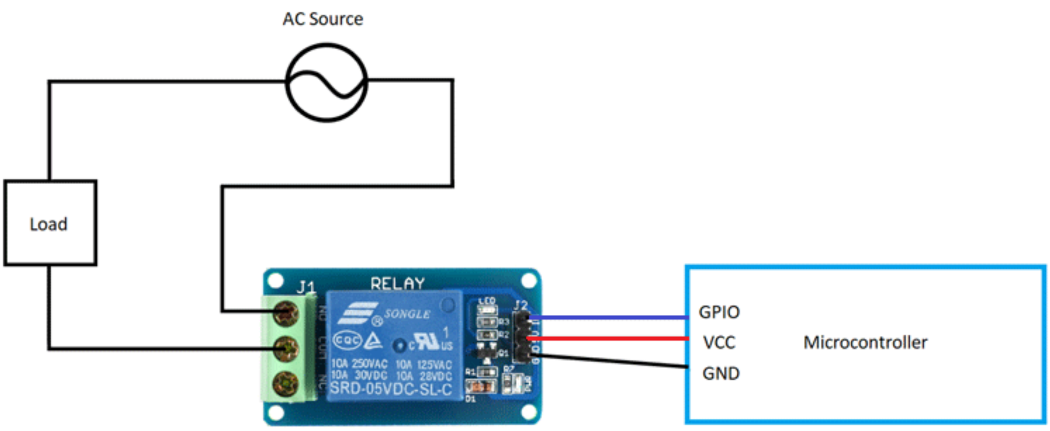




Relay is an electromechanical device that uses an electric current to open or close the contacts of a switch. The single-channel relay module is much more than just a plain relay, it comprises components that make switching and connection easier and act as indicators to show if the module is powered and if the relay is active or not.

Main Features

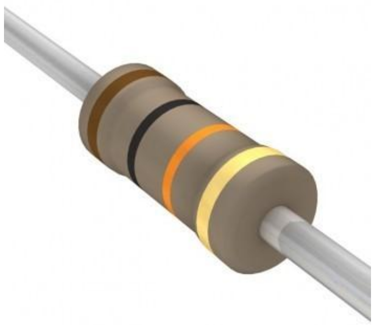
- Supply voltage – 3.75V to 6V
- Quiescent current: 2mA
- Current when the relay is active: ~70mA
- Relay maximum contact voltage – 250VAC or 30VDC
- Relay maximum current – 10A



5. Electric Water Heater

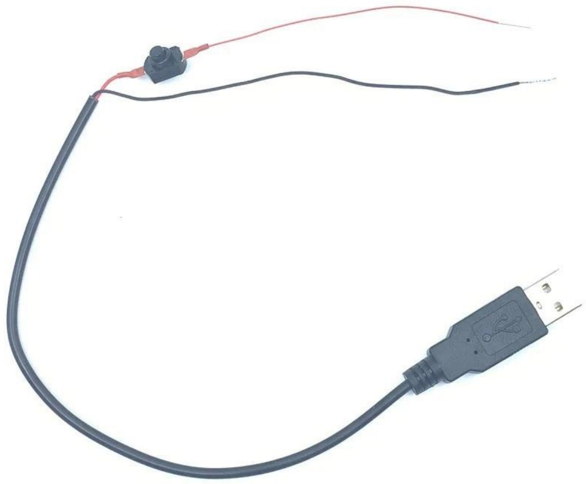
It heats water quickly and has a corrosion resistive body that keeps it away from rusting. Save energy, it is electric, but it does not need to provide you with continuous hot water. When your water is boiled, you can unplug the immersion water heater. The Electric heater is so portable that it is an easy instalment that gives you the temperature that you want, anytime anywhere.

6. Resistors



The resistor is a passive electrical component that creates resistance in the flow of electric current. In all electrical networks and electronic circuits, they can be found. The resistance is measured in ohms (Ω). An ohm is the resistance that occurs when a current of one ampere (A) passes through a resistor with a one volt (V) drop across its terminals. The current is proportional to the voltage across the terminal ends. Resistors are used for many purposes. A few examples include limiting electric current, voltage division, heat generation, matching and loading circuits, gain control, and setting time constants. They are commercially available with resistance values over a range of more than nine orders of magnitude.

7. 5V USB Cable



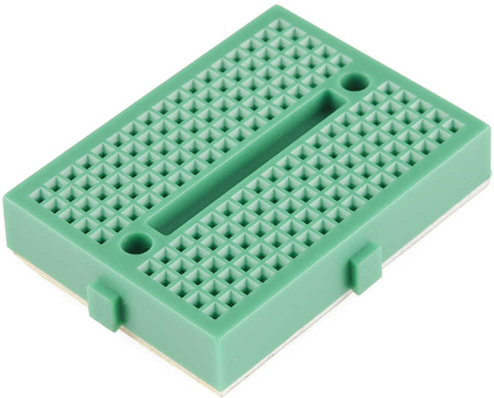
Most computer USB ports supply 5V of electricity with a maximum current of 0.5A. This amount of current is standard for most computers and means the overall power output will be 2.5 Watts at best.

8. Jumper Wire

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn't get much more basic than jumper wires.

Though jumper wires come in assorted colours, the colours do not mean anything. This means that a red jumper wire is technically the same as a black one. But the colours can be used to your advantage to differentiate between connections, such as ground or power.

9. Breadboard – Mini Modular



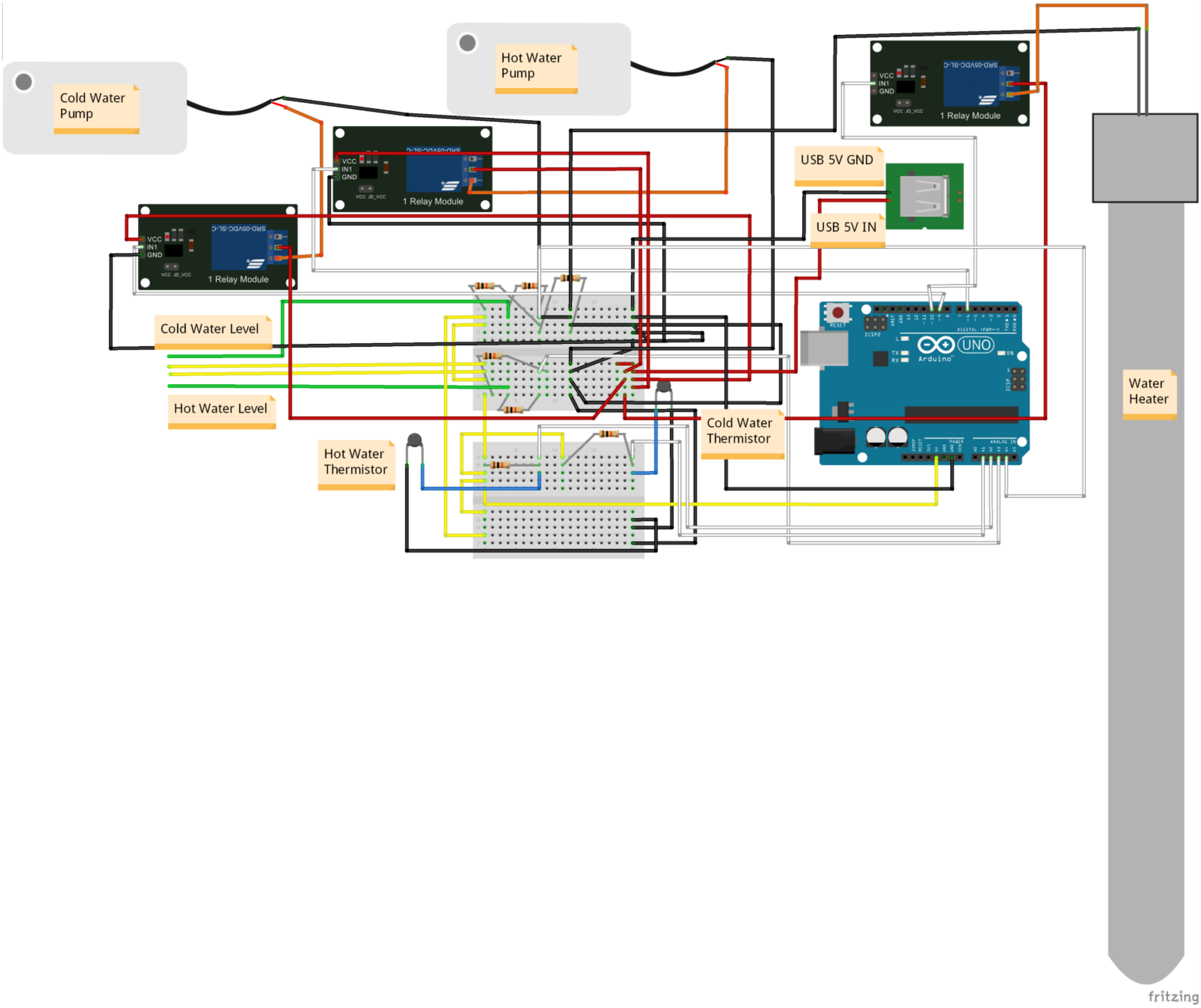
A breadboard is a solderless device for temporary prototypes with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connects the holes on the top of the board. This Mini Breadboard is a great way to prototype your small projects! With 170 tie points there's just enough room to build and test simple circuits; They are also great for breaking out DIP package ICs to jumper wires.

4. BUDGET

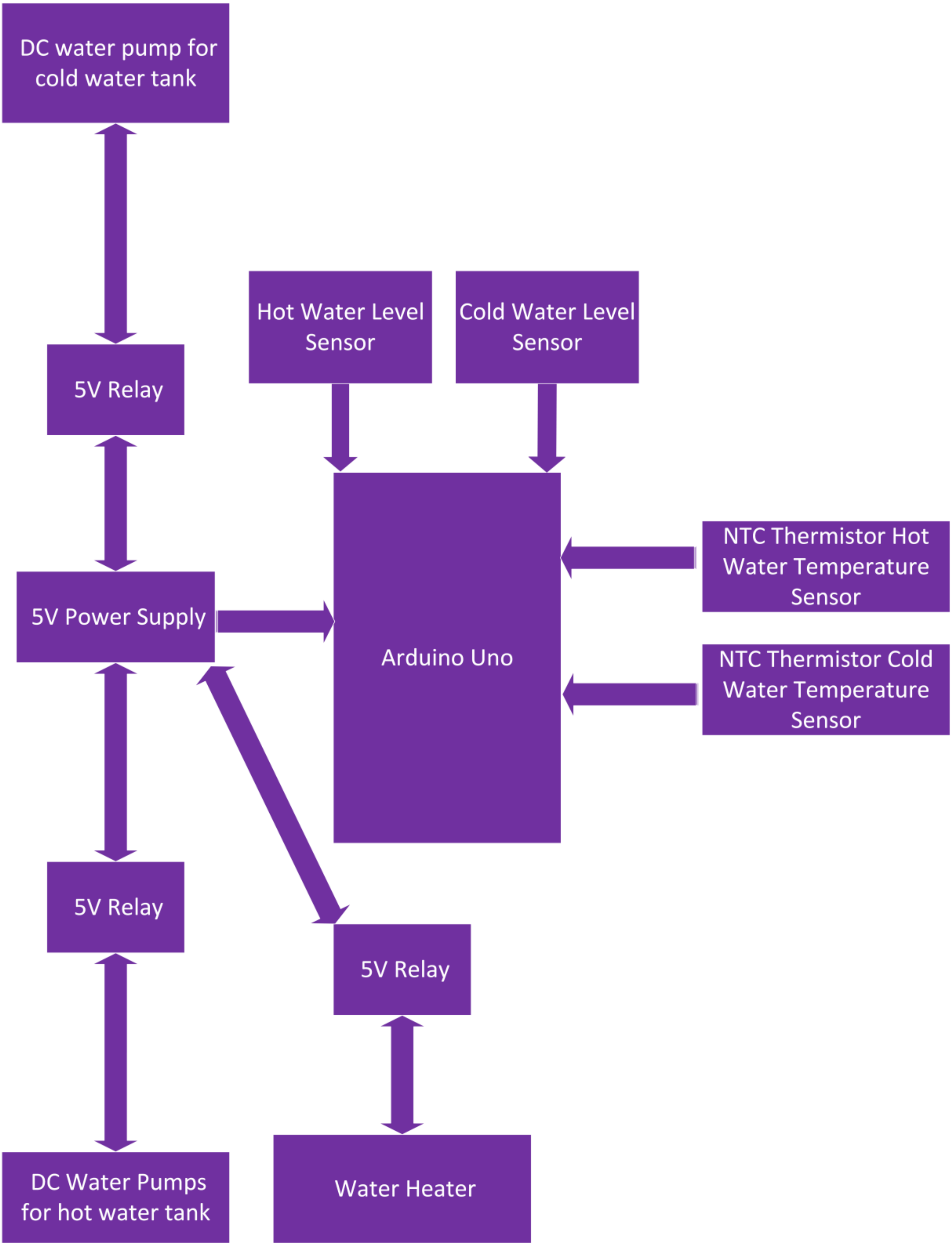
Item	Price(Rs)
Arduino Uno Original Development Board with USB Cable	3200.00
NTC 10K ohm 1% Thermistor Temperature Sensor with 1m Cable (2 parts)	380.00
3-6V DC Micro Submersible Mini Water Pump (2 parts)	560.00
5VDC 1 Way 1 Channel Relay Module (Transistor Version) (3 parts)	630.00
Electric Water Heater	1900.00
10k Resistors (10 Parts)	50.00
Jumper wire Female-to-Female (1x40-pin)	220.00
Jumper wire Male-to-Male (1x40-pin)	220.00
Jumper wire Female-to-Male (1x40-pin)	220.00
USB To Mini Type B USB Cable (30cm)	160.00
Mini breadboard (2 Parts)	160.00
Glue Gun Standard 60W	1020.00
Glue Stick Large Clear Long (5 parts)	250.00
Soldering Iron 30W	750.00
Transparent Tube for Mini Submersible Water Pump 1m	60.00
Plastic Jugs(2 parts)	1000.00
Plastic Cup	150.00

5. DESIGN OVERVIEW

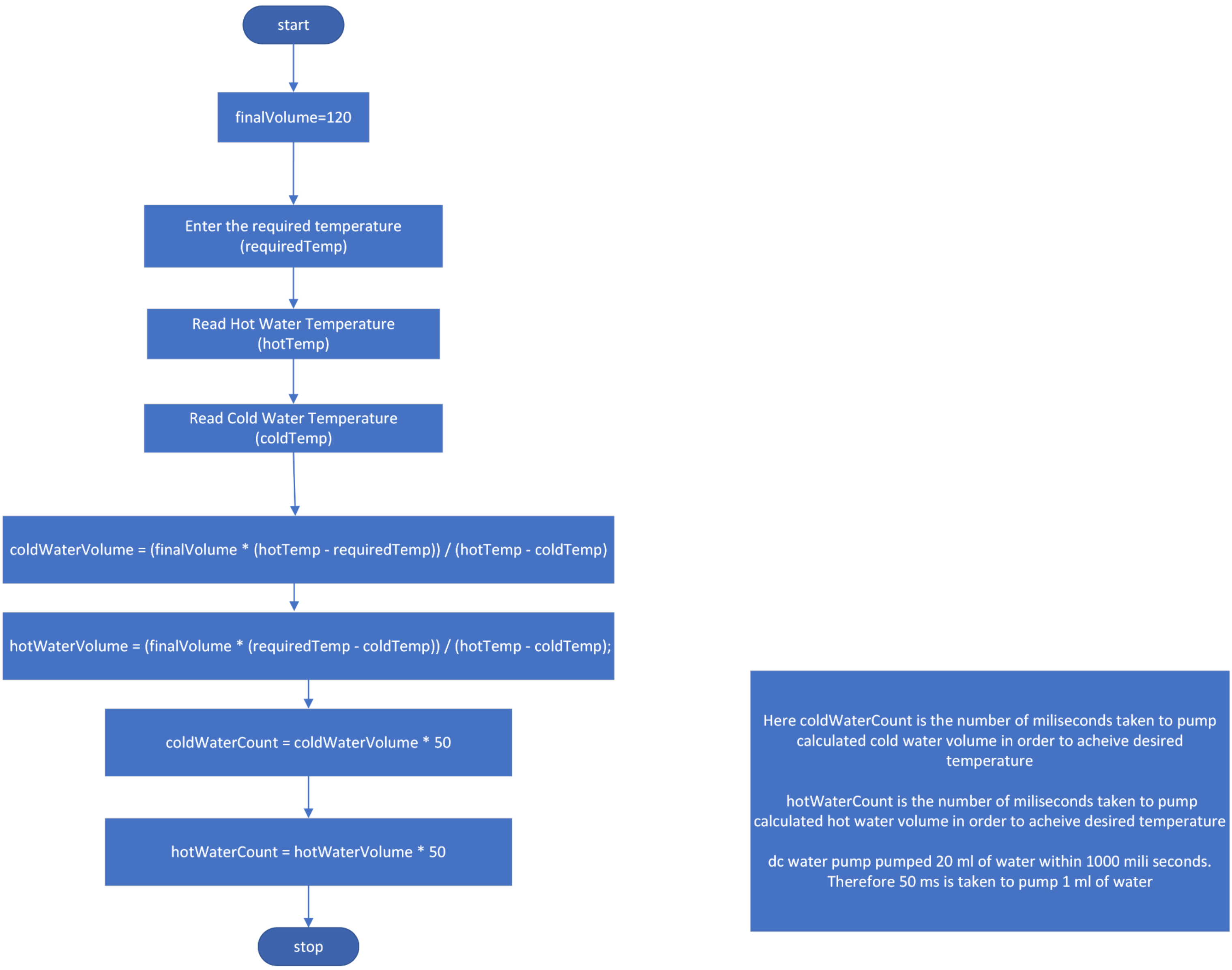
5.1 Circuit Diagram



5.2 Block Diagram



5.3 Flow Chart



5.4 Code

```
//Variables
unsigned long prev_time = millis();
int decimalPrecision = 2, hotThermistorPin = A2, coldThermistorPin = A1;
float voltageDividerR1 = 10000, BValue = 3380, R1 = 10000, T1 = 298.15, R2,
R3, hotTemp, coldTemp, a, b, c, d, e = 2.718281828 ;
float tempSampleRead = 0, tempSampleRead1 = 0, tempLastSample = 0,
tempLastSample1 = 0, tempSampleSum  = 0, tempSampleSum1 = 0,
tempSampleCount = 0, tempSampleCount1 = 0, tempMean, tempMean1;
int  hotWaterCount, coldWaterCount;
float coldWaterVolume, hotWaterVolume;
float finalVolume = 120.00;
float requiredTemp = 0;

void waterPump() {

    coldWaterCount= coldWaterVolume*50;
    digitalWrite(9, HIGH);
    delay(coldWaterCount);
    digitalWrite(9, LOW);

    hotWaterCount = hotWaterVolume*50;
    digitalWrite(6, HIGH);
    delay(hotWaterCount);
    digitalWrite(6, LOW);

}

void calVolume() {
  coldWaterVolume = (finalVolume * (hotTemp - requiredTemp)) / (hotTemp -
coldTemp);
  Serial.println(coldWaterVolume);
  hotWaterVolume = (finalVolume * (requiredTemp - coldTemp)) / (hotTemp -
coldTemp);
  Serial.println(hotWaterVolume);

}

void getAllTemp() {
  /* 1- Hot Temperature Measurement */
```

```
if (millis() >= tempLastSample + 1)
{
  tempSampleRead = analogRead(hotThermistorPin);
  tempSampleSum = tempSampleSum + tempSampleRead;
  tempSampleCount = tempSampleCount + 1;
  tempLastSample = millis();
}

if (tempSampleCount == 1000)
{
  tempMean = tempSampleSum / tempSampleCount;
  R2 = (voltageDividerR1 * tempMean) / (1023 - tempMean);

  a = 1 / T1;
  b = log10(R1 / R2);
  c = b / log10(e);
  d = c / BValue ;
  hotTemp = (1 / (a - d))-273;
  Serial.print("Hot Temperature:");
  Serial.print(hotTemp - 273.15, decimalPrecision);
  Serial.println(" Â°C");

  tempSampleSum = 0;
  tempSampleCount = 0;
}

/* 2- Cold Temperature Measurement */

if (millis() >= tempLastSample1 + 1)
{
  tempSampleRead1 = analogRead(coldThermistorPin);
  tempSampleSum1 = tempSampleSum1 + tempSampleRead1;
  tempSampleCount1 = tempSampleCount1 + 1;
  tempLastSample1 = millis();
}

if (tempSampleCount1 == 1000)
{
  tempMean1 = tempSampleSum1 / tempSampleCount1;
  R3 = (voltageDividerR1 * tempMean1) / (1023 - tempMean1);
```

```
    a = 1 / T1;
    b = log10(R1 / R3);
    c = b / log10(e);
    d = c / BValue ;
    coldTemp = (1 / (a - d))-273;
    Serial.print("Cold Temperature:");
    Serial.print(coldTemp - 273.15, decimalPrecision);
    Serial.println(" Â°C");

    tempSampleSum1 = 0;
    tempSampleCount1 = 0;
  }



}

void setup() {
  Serial.begin(9600);
  pinMode(9, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(10,OUTPUT);
     digitalWrite(6, LOW);
      digitalWrite(9, LOW);

while(1){
  getAllTemp();
  if(hotTemp>0 && coldTemp>0 ){

Serial.println("hotTemp");
          Serial.println(hotTemp);
           Serial.println("coldTemp");
          Serial.println(coldTemp);
          break;
          }
  }
}
```

```
void loop() {
    Serial.println("Required Temperature ");      //Prompt User for input
    while (Serial.available() == 0) {}
    if (Serial.available() > 0) {
      requiredTemp = Serial.parseFloat();             //Read user input and hold it in a
variable

    }
    if(requiredTemp!=0){
      Serial.println(requiredTemp);
      while (1){
        getAllTemp();
        if(hotTemp>0 && coldTemp>0 ){
          Serial.println("hotTemp");
          Serial.println(hotTemp);
           Serial.println("coldTemp");
          Serial.println(coldTemp);
          calVolume();
          Serial.println("cold  volume");
          Serial.print( coldWaterVolume);
          Serial.println("hot  volume");
          Serial.print( hotWaterVolume);
          if(coldWaterVolume>0 && hotWaterVolume>0 ){
          waterPump();
          }
          hotTemp=0;
          coldTemp=0;
          break;

          }
        }

      }// wait for user input



}
```

6. ISSUES

- I couldn’t use a keypad to get the user input for the required temperature. Instead I used the serial monitor to enter the desired temperature.

- At first a load cell was used to measure the volume of water using mass. But some issues arose when calibrating Load Cell with HX711 with Arduino. Therefore it was removed from the system. A measuring cup was used to take the value for the final volume (volume of the cup) and to decide the flow rate of the dc water pump (time taken to dispense 1 ml of water).

- It was decided to use a heater to maintain constant hot water temperature but due to some circumstances it was unused. Instead instant hot water was used to check the operation of the developed embedded system.

7. DISCUSSION

- Calculation of cold water volume and hot water volume that is needed to obtain desired temperature is done by using an equation. The equation is constructed by following the first law of thermodynamics (conservation of energy) which implies that the amount of heat energy lost by hot water is equal to the amount of heat energy gained by cold water.

- delay function pauses the program for the amount of time (in milliseconds) specified as a parameter. It is used to calculate the time to cold water flow and hot water flow that is needed to supply the desired temperature. First,the amount of water dispensed within 1 second(1000 ms) was measured. With the aid of that reading, time taken to dispense 1 ml of water was decided. Then calculated cold water volume and hot water volume that needed to provide desired temperature was multiplied by the time taken to dispense 1 ml of water.

- All these calculations are done to obtain only one predetermined volume at the desired temperature which is the volume of the cup(final volume). The volume of the cup is measured using a measuring cup and fed into the system as a predefined value.

- An external power supply is used to power up the dc water motors of hot water tank and cold water tanks. Two relays are used to control the mechanism of dc water motors.

8. CONCLUSION

A smart water dispenser that supplies the desired temperature of water is created. The system meets the demand of high-speed production using the least mechanism requirements. The system works effectively avoiding unnecessary spill of water. The project was completed successfully.

8.1 FUTURE IMPLEMENTATION

Currently this system provides only one volume value which is the volume of the cup. It can be improved to dispense any desired volume value at the desired temperature

9. References

https://www.wikipedia.org/
https://www.youtube.com/

☐