Small size model of Temperature sensing Liquid cooling System

(One page write up)

Section : A **Table no.** : 11

Names and roll numbers of group members :

Gyarsi Lal Meena(190352)
Sree Preetham.B(190863)
Abhimithra Ganta (190331)

1. What problem are you trying to solve and why is it important / interesting ?

Problem Statement:

Overheating of some appliances during summer/ due to high usage operations, cause timely repairs of those devices which are used continuously. There may be expensive devices which need good maintenance during such seasons. This system is an attempt to detect the temperature of the device while it's working and act accordingly if the temperature exceeds its optimal working conditions.
This model can also be extended into laboratories where machines require quick discharge of heat during their operation for its safety. We are trying to work on a mini coolant that cools up devices once the device is heated up to a specific temperature.

2. What are the existing solutions? Describe a few of them and list any shortcomings in them. Is your solution approach unique in some way?

Devices usually are embedded with a system to solve this problem. The existing solution is by using a mini exhaust fan for the devices. We can see it everywhere. But it may not cool the device quickly and the damage may be caused during its attempt. Liquid cooling is quick as the heat transfer is between device and water, which is a better coolant. We use water as a simple yet an effective solution because it is one of the best choices for liquid cooling applications due to its high heat capacity and thermal conductivity.

3. What resources do you require for completing the project? Give a breakup of tasks that you need to accomplish week by week to complete the project.

For our task we require,

1. **Water Pump** - To pump the water from the tank around the device
2. **Temperature Sensor** - We use it measure the temperature at which the pump should be switched on.
3. **Arduino NANO and Power Cable** - Programming it to function as an intermediate between Water Pump and Temp Sensor.

4. **Relay -** We use this as a switch to test the functioning of the pump before putting the Temperature sensor.
5. **Jumper Wires**
6. **Breadboard**

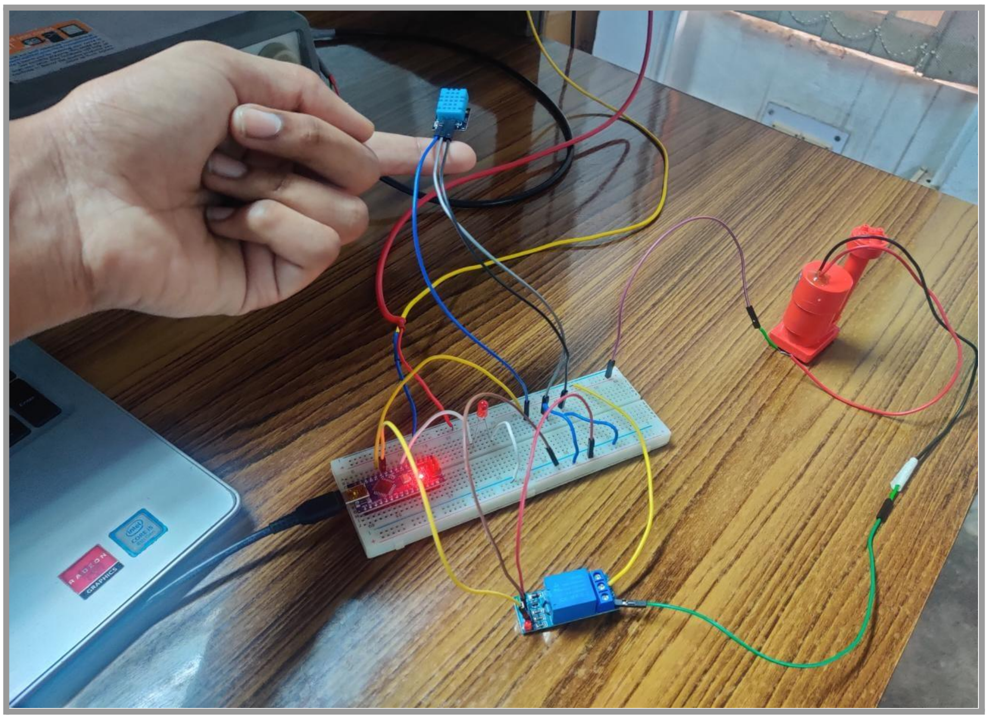
Implementation of the project :

Our task is divided into three subsections

1. Taking an input from the temperature sensor
2. Making a decision as to whether the water should pump or not by the received reading
3. If the answer is yes, the DC water pump has to pump the water until the device is cooled.

In order to achieve this, as discussed earlier we are using an arduino nano that acts as the point of taking inputs and giving outputs, decision making.

Here's the circuit for the temperature sensor working:



In addition, we also used Relay to act as a switch once the motor starts to pump water to prevent continuous flow of water. In our task we used a time delay of n seconds (n can be set in the code) => after every n seconds the water keeps pumping for a while and then stops so as to keep the pump away from overheating. This process continues until the target device cools down below to a certain temperature level (in our case room temperature).

Results :

In order to demonstrate the working of our project, we used a metal object and heated it up with a lighter, so as to increase the temperature of the object.
When we place the object closer to the DHT-11, as the air around the object also gets heated up, the sensor senses it and gradually on the screen we can see the temperature increasing.

When the temperature of the localized air, according to the sensor, reaches a threshold value(which we had set up while writing the code), as discussed the DC water motor starts to pump the water. The water would start to flow from the pipe and pass through the water blocks which are attached to the device.
The pump runs for a set time and waits for a set time while running to be efficient while sensing and pumping water.
When the device and (closer to the DHT) is cooled to a certain safe temperature the pump shuts down.
In this way we can cool the object gradually to room temperature.

This model can be designed and extended into devices which are overused and require maintenance or laboratories with expensive equipment, using better and accurate sensors and optimal design of waterblock.

The code we used for the implementation is attached below. We used the DHT11 library to read the signal from the DHT and feed it to the variable as per our requirement.

```
#include <dht.h>

#include <dht11.h>

dht DHT;

#define DHT11_PIN 7

int ledPin = 3;
int  ledpinvalue;
int RelayPin = 6;
int  temp_data;
void setup() {
  // Set RelayPin as an output pin
   pinMode(ledPin, OUTPUT);
   pinMode(RelayPin, OUTPUT);
   pinMode(DHT11_PIN, INPUT);
Serial.begin(9600);
}


void loop() {

 int chk = DHT.read11(DHT11_PIN);
   //Serial.print("Temperature = ");
    //Serial.println(DHT.temperature);
   //Serial.print("Humidity = ");
    //Serial.println(DHT.humidity);
   delay(5000);
   temp_data = DHT.temperature;
```

```
  Serial.print("Temperature=  ");
   Serial.println(temp_data);
   //Serial.println("degrees");


   if(temp_data!=(0,50)){
     digitalWrite(RelayPin,HIGH);
  }

   if(temp_data>=32){

        ledpinvalue=255;
         analogWrite(ledPin,ledpinvalue);
        delay(1000);
        ledpinvalue=0;
         analogWrite(ledPin,ledpinvalue);
        delay(30);

  // Let's turn on the relay...
  digitalWrite(RelayPin, LOW);    //Switch on
the motor
  delay(10000);

  // Let's turn off the relay...
  digitalWrite(RelayPin, HIGH);   //Switch off
the motor
  delay(8000);
  }

   if(temp_data<32){
```

```
        digitalWrite(RelayPin,HIGH);
      }
}
```