Smart Coaster

By

Phùng Quang Nhật

Phạm Thanh Quang

Nguyễn Đặng Phúc

1 Introduction

1.1 Objective

In order to be fully enjoyed by a consumer, many drinks, like hot coffee, need to beprepared to an optimal temperature. The problem is that the initial temperature of a drink is correlated to the temperature it is stored in, without prior preparation. The storage temperaturerange can vary from frozen to chilled to normal room temperature.

In order to address the problem of inaccuracy and inefficiency in controlling drink temperatures, we propose the coaster that will emit a user-defined temperature in our cup. We believe that our product would be a simple and efficient way to modify the temperature of theirdrink. In addition, the portable nature of the coaster affords users the luxury of taking it on-the-go to heat and cool their drinks, something that is not afforded by major household appliances.

1.2 Visual Aid

The design of the coaster depicted in Figure 1 will allow for users to place their standard-sized mugs upon the silver heating/cooling pad featured in the middle of the design. This pad will feature a module that can be used to then manipulate the temperature of the drinkto the temperature.

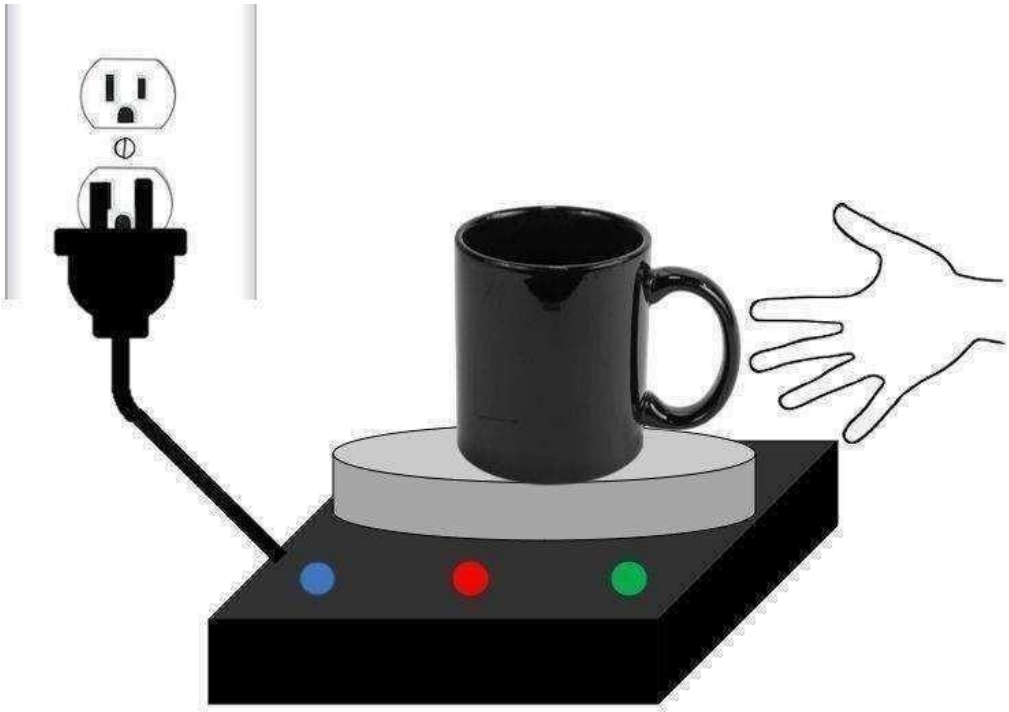


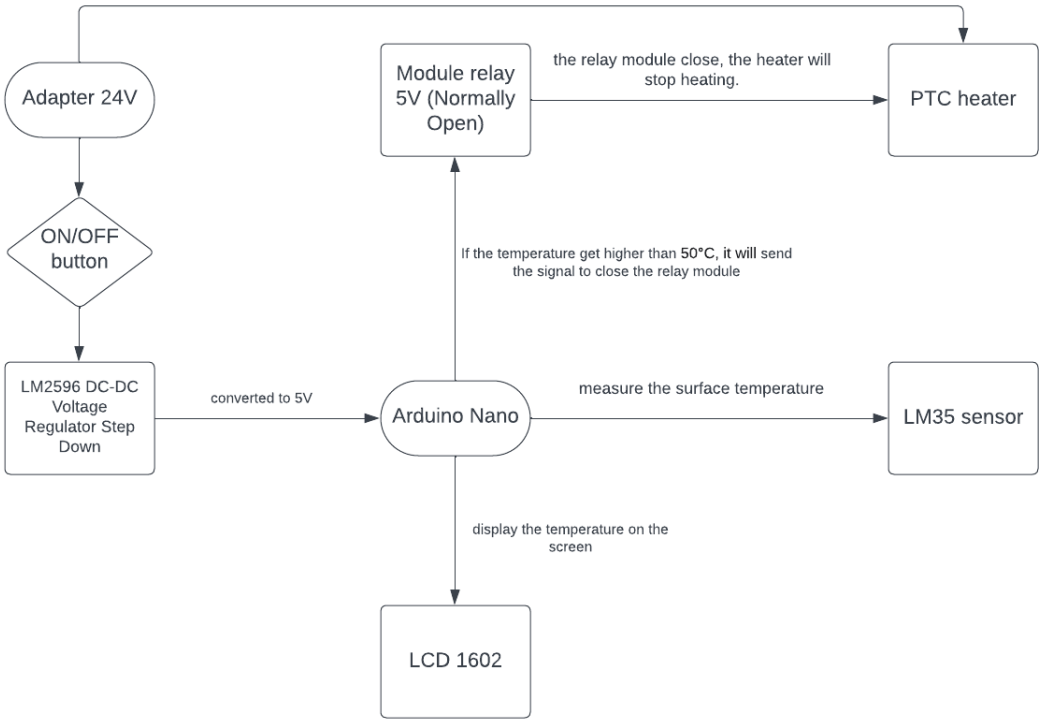
Figure 1. Visual Aid for Scale

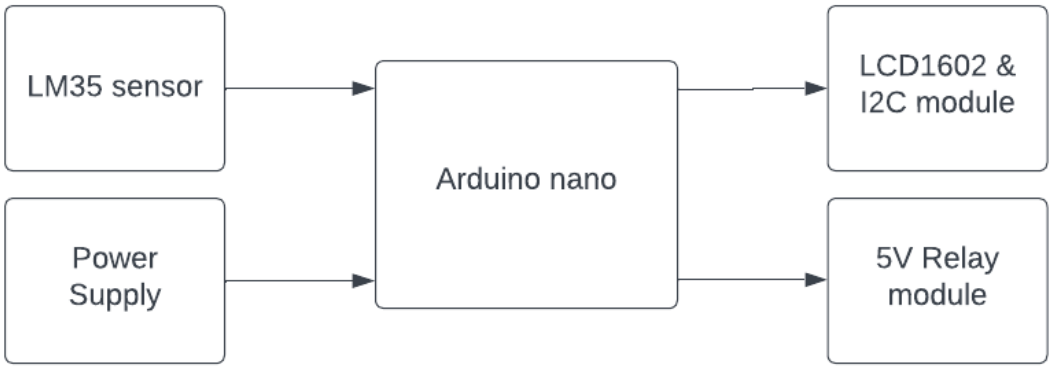
1.3 High-Level Requirements

- A portable drink coaster that includes a temperature sensor, pressure sensor, and power supply that can heat 1 cup of a liquid. LEDs must also be present on the coaster to indicate when the coaster is heating, cooling, or inactive.
- Given the drink is initially at room temperature (~23° C), the coaster must heat it within 50° C .

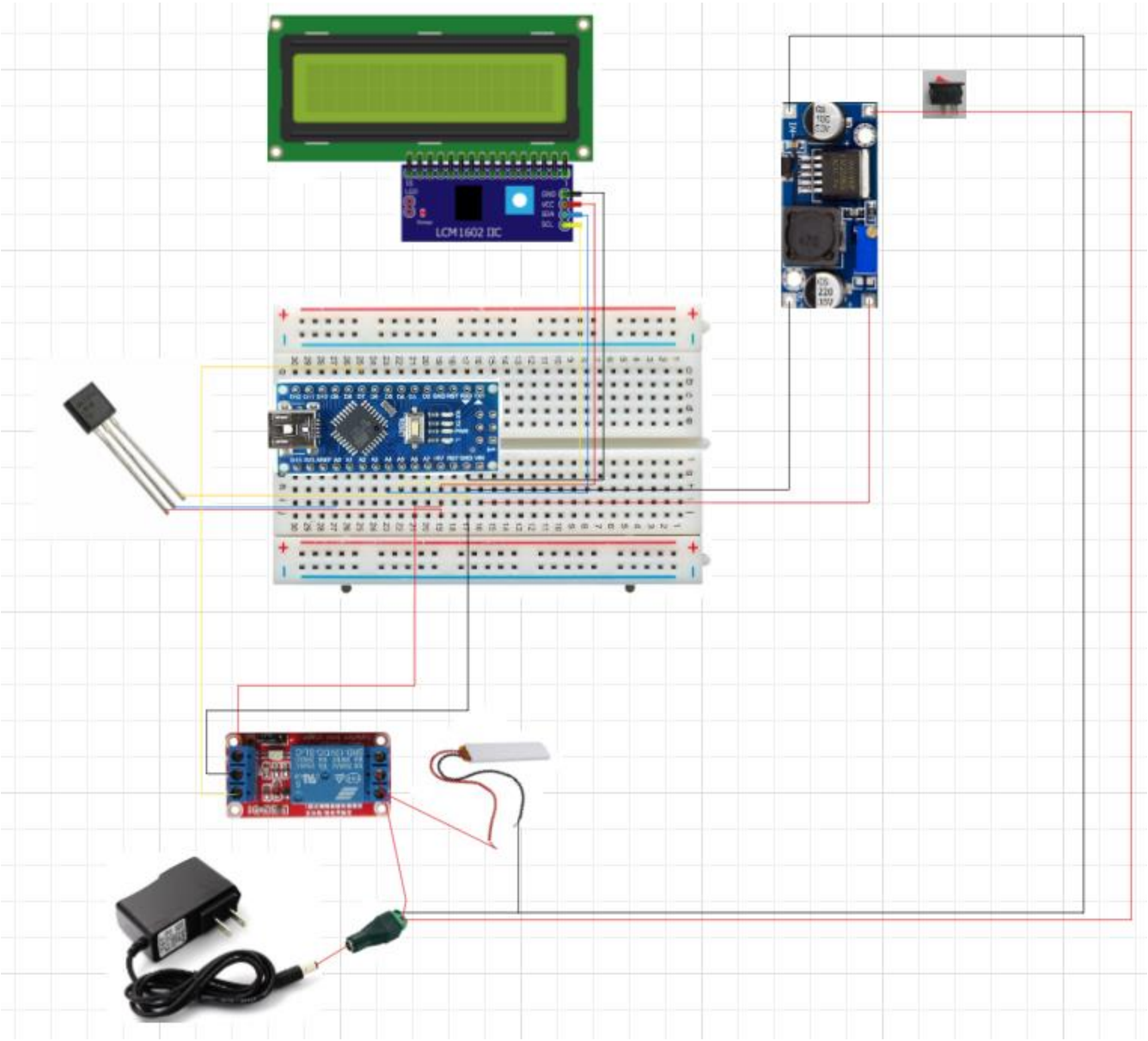
2/ Design

2.1 / Block diagram:





2.2 / Diagram



2.3/ Code

Pseudo Code

```
Initialize the OneWire instance with the sensor pin
Initialize the DallasTemperature instance with the OneWire instance
Initialize the LiquidCrystal_I2C instance with the LCD address and
dimensions

Setup:
  Initialize serial communication
  Begin the sensors
  Set the relay pin as an output
  Initialize the LCD and turn on the backlight

Loop:
  Read the analog value from the sensor pin
  Convert the analog value to temperature in Celsius and Fahrenheit

  Display the temperature on the LCD
  Print the temperature values to the serial monitor

  Check if the temperature in Celsius is below the threshold
    If true, turn off the relay
    If false, turn on the relay

Delay for a certain period
```

Arduino code:

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

#define SENSOR_PIN  A0  // Analog pin connected to LM35 sensor's output
#define RELAY_PIN   7   // Digital pin connected to the relay module
```

```
const float TEMP_THRESHOLD_UPPER = 60;  // Upper threshold of temperature, change to your desired value
OneWire oneWire(SENSOR_PIN);         // Setup a oneWire instance
DallasTemperature sensors(&oneWire); // Pass oneWire to DallasTemperature library
```

```
void setup() {
  Serial.begin(9600); // Initialize serial communication
  sensors.begin();    // Initialize the sensor
  pinMode(RELAY_PIN, OUTPUT); // Initialize digital pin as an output
  lcd.init();
  lcd.backlight();
}
```

```
void loop() {
  int value1 = analogRead(SENSOR_PIN);
  double tempC = (value1 * (5.0 / 1023.0) * 100);
  double tempF = (tempC * 9 / 5 + 32);
  lcd.setCursor(0, 0);
  lcd.print("..Temperature..");
  lcd.setCursor(0, 1);
  lcd.print("C:");
  lcd.print(tempC);
  lcd.setCursor(8, 1);
  lcd.print(" F:");
  lcd.print(tempF);
  Serial.print("Temperature C: ");
  Serial.print(tempC);
  Serial.print("'C");
  Serial.print("\t");
  Serial.print("Temperature F: ");
  Serial.print(tempF);
  Serial.println("'F");
  if (tempC < TEMP_THRESHOLD_UPPER) {
    Serial.println("The heating element is turned off");
    digitalWrite(RELAY_PIN, HIGH ); // Turn off the relay
  } else  {
    Serial.println("The heating element is turned on");
    digitalWrite(RELAY_PIN, LOW); // Turn on the relay
  }
```

```
  delay(500);
}
```

3 Ethics and Safety

There is a potential ethical issue that the durability of the coaster would be compromised in situations that involve direct contact with objects or liquids that may damage or destroy the integrity of the circuit. Our coaster will have water resistance for the outer structure of the coaster. By doing so, we avoid potential issues where the user is shocked when spilling a liquid onto the device, or even further shorting the entire circuit design. Wires becoming exposed within our design are also a safety issue, especially given our design's use of high current to power the heating module. To combat this issue, our coaster's exterior should be composed of a durable and shock-absorbent material like plastic to nullify minor physical collisions that may damage the internal circuit or expose wires within the design.

Another possible damage and safety concern that could arise from the project include the heat generated. To deal with the safety concern of burning an individual there will be LED indicators for heating modes that are clear and easy to see. Another potential concern here is damage to the surface that the user has placed the device on. For this issue there will be a layer of insulating material between the heating device and the surface it is resting on.

4 Cost

Components	Quantity	Cost (VND)	Total (VND)
Adapter 24V	1	68.000	68.000
Arduino Nano	1	85.000	85.000
Temperature sensor lm35	1	45.000	45.000
Breadboard	1	19.000	19.000
20CM 40Pcs Male to Male, Male to Female and Female to Female Jumper Wire	2	28.000	56.000
LCD 1602 5V + I2C	1	48.900	48.900
LM2596 DC-DC Buck Converter Step-Down Power Module	1	21.314	21.314
Relay 5V	1	21.200	21.200
PTC Heater	1	45.000	45.000
DC Power Jack 5.5x2.1mm	1	3.400	3.400
Switch KCD1 size 15X21mm	1	1.900	1.900
Total			414.714

5 Result