Ashton Alston

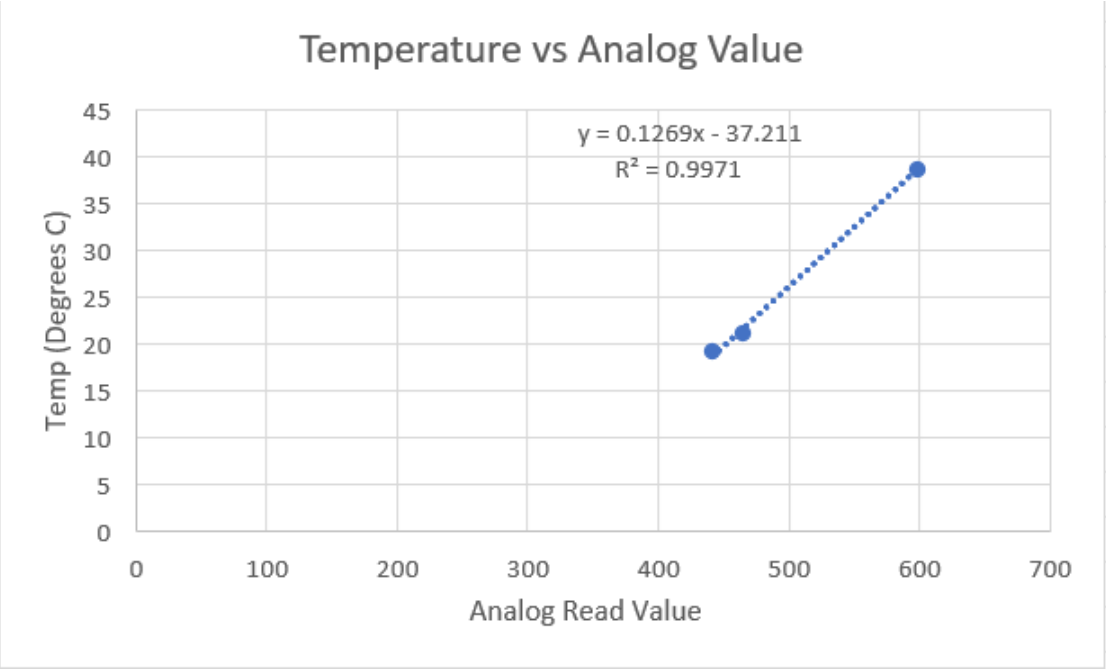
Engr.121.007

Dr. Easley

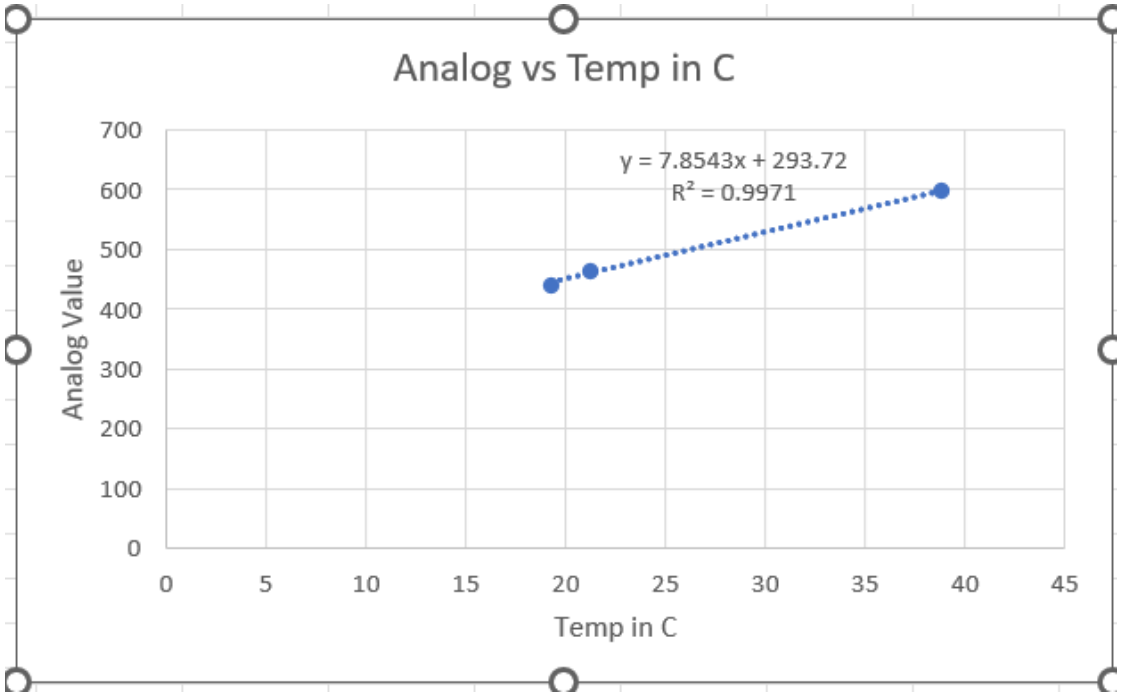
1/25/2024

Calibration Equation (Analog in terms of Temperature)		7.8543x + 293.72
Inverted Calibration Equation (Temperature in terms of Analog)		0.1269x -37.211
Standard Deviation Value	1	
	Analog	**°C**
LCL	463	21.54
Setpoint	466	21.92
UCL	469	22.31
Deadband (UCL - LCL)	6	0.76

Plot of Temperature vs Analog :



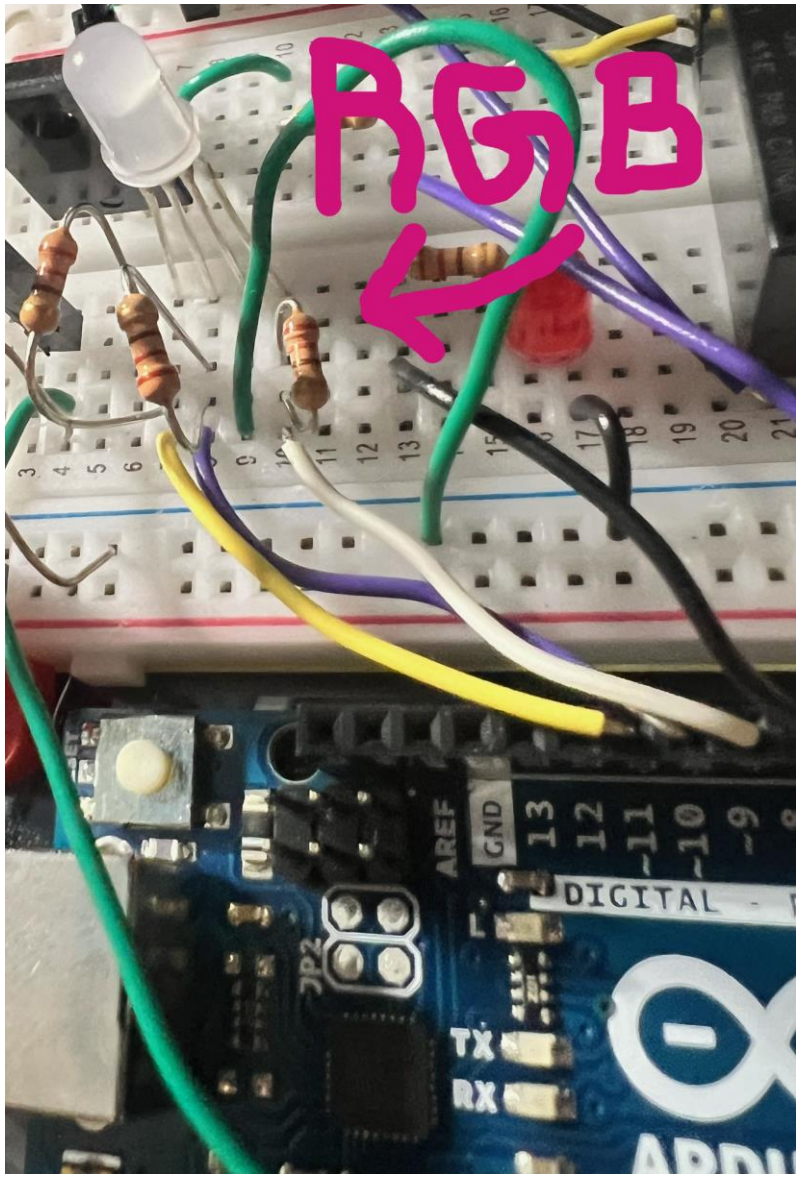
Plot of Analog vs Temperature:



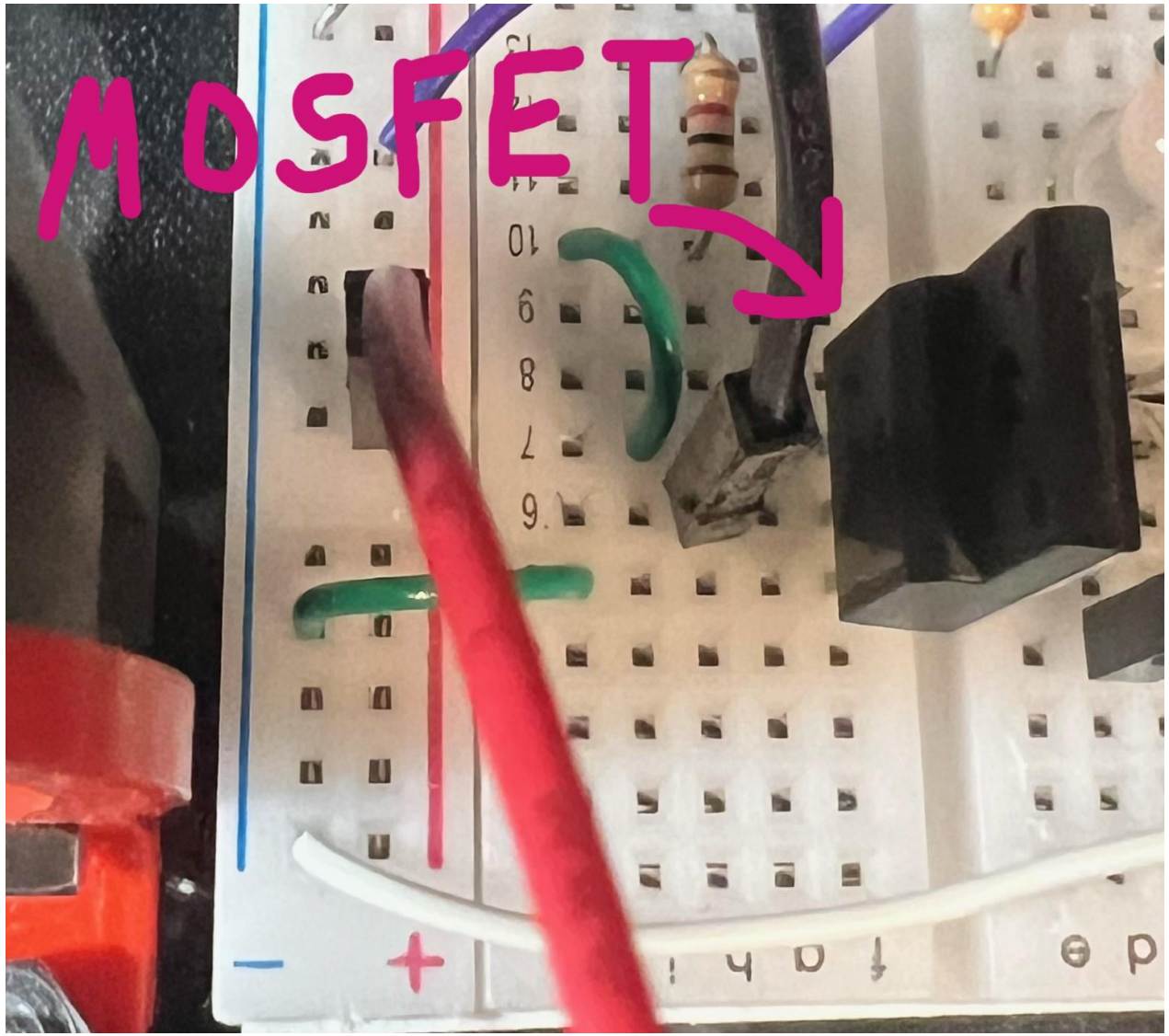
20 data points and the result of my standard deviation:

Analog	Standard Dev
467	0.933302
467	
468	
468	
469	
468	
467	
468	
469	
470	
468	
467	
467	
468	
468	
468	
468	
469	
469	
470	

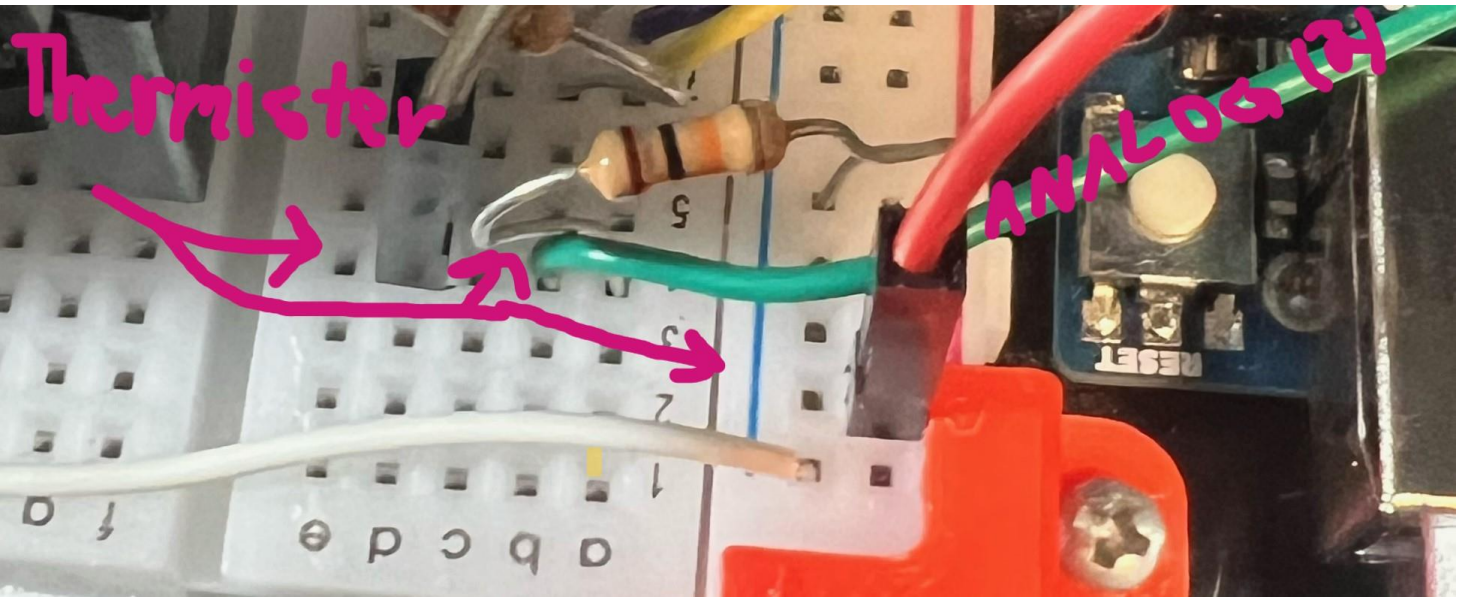
RGB Circuit:



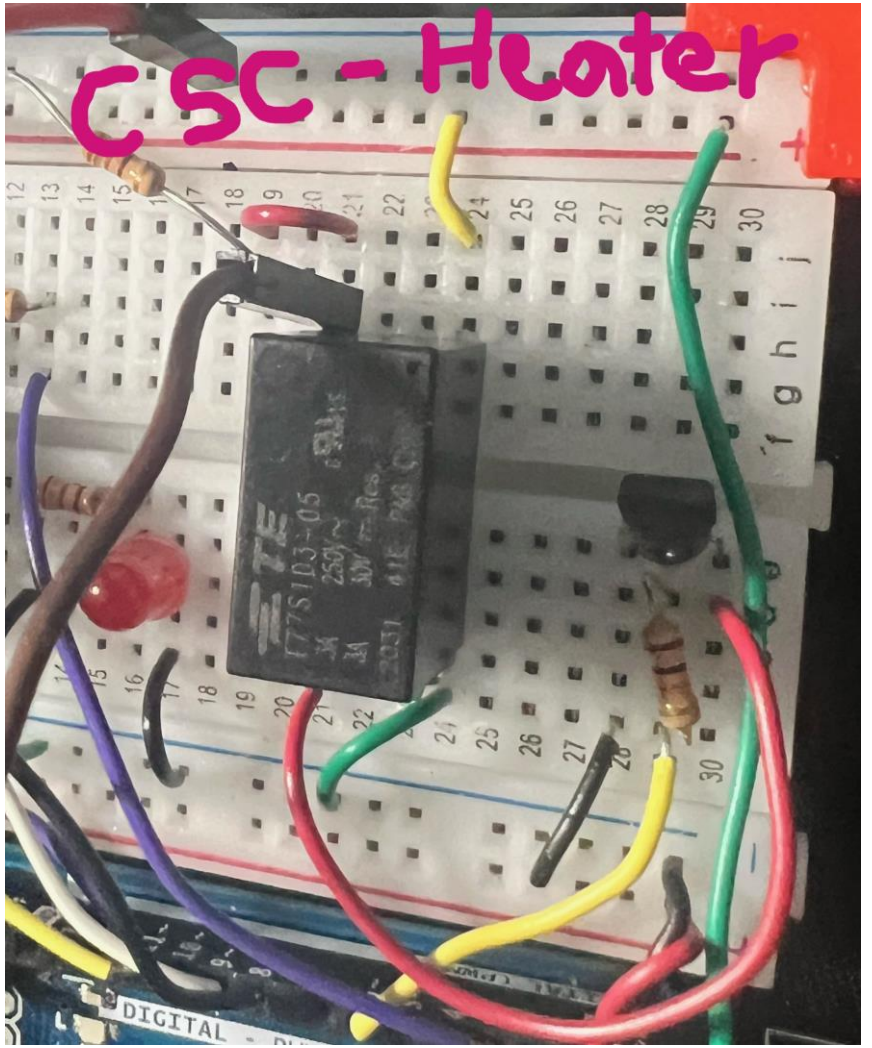
MOSFET to turn fan on:



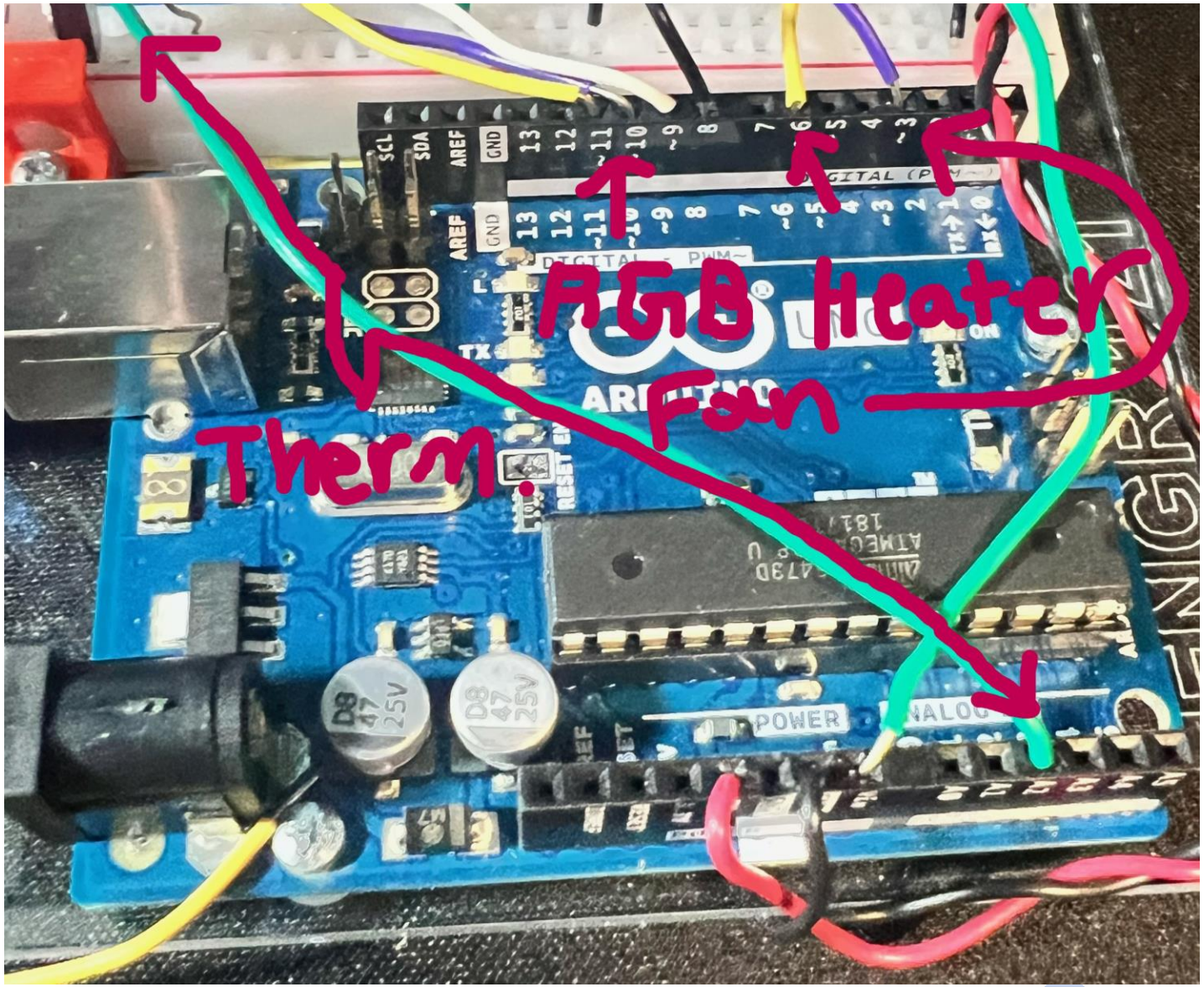
Cascading Switch Circuit:



Thermistor Circuit :



Arduino Board to show
What pins contribute
to inputs and outputs:



Final Code For Canister Oven:

```
// Initializing pins for components
const int redLED = 9;
const int blueLED = 11;
const int fan = 3;
const int heater = 6;
const int heaterLight = 8;

// Analog inputs and LED intensities
int tempA;
int redIntensity;
int blueIntensity;

// Constants for temperature control
const int stDev = 1;
float SPC = 24.4;
float tempC;

// Setpoints and control limits
int SPA = 7.8543 * SPC + 293.72;
int UCLA = SPA + 3 * stDev;
int LCLA = SPA - 3 * stDev;

// Mapping ranges for LED intensities
int fromLOW = LCLA;
int fromHIGH = UCLA;
int toLOW = 0;
int toHIGH = 255;

// Setup function
void setup() {
  Serial.begin(9600);
  pinMode(redLED, OUTPUT);
  pinMode(blueLED, OUTPUT);
  pinMode(fan, OUTPUT);
  pinMode(heater, OUTPUT);
  pinMode(heaterLight,OUTPUT);
  Serial.println("Temperature Control System Initialized");
}

// Loop function
void loop() {
  // Read analog temperature and convert to Celsius
  tempA = analogRead(2);
  tempC = 0.1269 * tempA - 37.211;

  // Map temperature to LED intensities
  redIntensity = map(tempA, fromLOW, fromHIGH, toLOW, toHIGH);
  blueIntensity = map(tempA, fromLOW, fromHIGH, toHIGH, toLOW);

  // Print headers
  Serial.println("LCLA\tSP\tUCLA\tTempA\tTempC\tHeater\tFan");

  // Print variable values
  Serial.print(LCLA); Serial.print("\t"); Serial.print(SPA); Serial.print("\t");
  Serial.print(UCLA); Serial.print("\t"); Serial.print(tempA); Serial.print("\t");
  Serial.print(tempC); Serial.print("\t");
  // Fade RGB
  analogWrite(redLED,redIntensity);
  analogWrite(blueLED,blueIntensity );

  // Control heater and print status
  if (tempA < LCLA) heaterOn();
  if (tempA > UCLA) heaterOff();
  if (tempA <= UCLA && tempA >= LCLA) Serial.print("OFF");

  Serial.print("\t");

  // Control fan and print status
  if (tempA > UCLA) fanOn();
  if (tempA < SPA) fanOff();
  if (tempA <= UCLA && tempA >= SPA) Serial.print("OFF");

  // Delay for stability
  delay(1000);
  Serial.println("");
}

// Function to turn heater on
void heaterOn() {
  Serial.print("ON");
  digitalWrite(heater, HIGH);
  analogWrite(blueLED, 255);
  analogWrite(redLED, 0);
  digitalWrite(heaterLight,HIGH);
}

// Function to turn heater off
void heaterOff() {
  Serial.print("OFF");
  digitalWrite(heater, LOW);
  analogWrite(redLED, 255);
  analogWrite(blueLED, 0);
  digitalWrite(heaterLight, LOW);
}

// Function to turn fan on
void fanOn() {
  Serial.print("ON");
  digitalWrite(fan, HIGH);
}

// Function to turn fan off
void fanOff() {
  Serial.print("OFF");
  digitalWrite(fan, LOW);
}
```

Serial Monitor:

LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	481	23.83	ON	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	482	23.95	OFF	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	482	23.95	OFF	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	481	23.83	ON	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	481	23.83	ON	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	482	23.95	OFF	OFF
LCLA	SP	UCLA	TempA	TempC	Heater	Fan
482	485	488	482	23.95	OFF	OFF