Thermal Regulation in a PCR Chamber

Elisha Safa Saphira
Biomedical Engineering Student
Swiss German University
elisha.saphira@student.sgu.ac.id

Katherine Kimberly Sebastiane
Biomedical Engineering Student
Swiss German University
katherine.sebastiane@student.sgu.ac.id

***Abstract*—Polymerase chain reaction (PCR) is a commonly used process for rapidly producing millions to billions of copies (complete or partial copies) of a given DNA sample. PCR is usually done by the thermal cycler in the laboratory to amplify DNA segments. The major step of PCR is denaturation, annealing, and extension, which each of them require different time and temperature during the process. The proportional–integral–derivative controller (PID controller) is a feedback-based control loop commonly used in industrial control systems and other applications that require continuously modulated control. This paper explains the PCR chamber project from a circuit perspective. The project is inspired by the concept of PCR and its thermal regulation. Thermal regulation plays a key part in PCR. This project's main purpose is to regulate the temperature of the PCR chamber, by heating the chamber and maintaining specific temperature. A PID controller system is used in this project as a means to manage the temperature of the chamber while considering its temperature and timing.**

***Index Terms*—PCR, Thermal regulation, PID Controller**

I. INTRODUCTION

A. PCR

PCR is a method used to make millions, even billions of a specific DNA sample rapidly. This method was developed in the 1980s by Kary B. Mullis. This fast and relatively inexpensive technique is sometimes called "molecular photocopying". Once amplified, the DNA produced by PCR can be used in many different laboratory procedures. For example, most mapping techniques in the Human Genome Project (HGP) relied on PCR.

PCR is also valuable in a number of laboratory and clinical techniques, including DNA fingerprinting, detection of bacteria or viruses (particularly AIDS), and diagnosis of genetic disorders. PCR is based on using the ability of DNA polymerase to synthesize new strands of DNA complementary to the offered template strand. Because DNA polymerase can add a nucleotide only onto a preexisting 3'-OH group, it needs a primer to which it can add the first nucleotide. This requirement makes it possible for a preexisting 3'-OH group, it needs a primer to which it can add the first nucleotide. This requirement makes it possible to delineate a specific region of template sequence that the researcher wants to amplify. At the end of the PCR reaction, the specific sequence will be accumulated in billions of copies (amplicons).

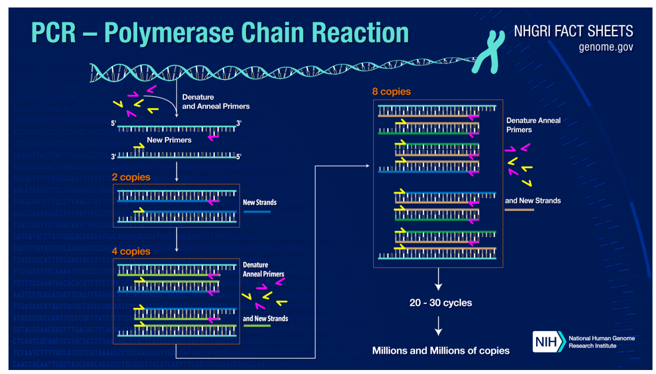


Fig. 1. Steps of Polymerase Chain Reaction

PCR happens in a total of 6 steps, which could be described as follows.

1. Initialization: This step is only required for DNA polymerases that require heat activation by hot-start PCR. It consists of heating the reaction chamber to a temperature of 94–96 °C (201–205 °F), or 98 °C (208 °F) if extremely thermostable polymerases are used, which is then held for 1–10 minutes.
2. Denaturation: This step is the first regular cycling event and consists of heating the reaction chamber to 94–98 °C (201–208 °F) for 20–30 seconds. This causes DNA melting, or denaturation, of the double-stranded DNA template by breaking the hydrogen bonds between complementary bases, yielding two single-stranded DNA molecules.
3. Annealing: In the next step, the reaction temperature is lowered to 50–65 °C (122–149 °F) for 20–40 seconds, allowing annealing of the primers to each of the single-stranded DNA templates.
4. Extension/elongation: The temperature at this step depends on the DNA polymerase used; the optimum activity temperature for the thermostable

DNA polymerase of Taq polymerase is approximately 75–80 °C (167–176 °F), though a temperature of 72 °C (162 °F) is commonly used with this enzyme. In this step, the DNA polymerase synthesizes a new DNA strand complementary to the DNA template strand by adding free dNTPs from the reaction mixture that is complementary to the template in the 5′-to-3′ direction, condensing the 5′-phosphate group of the dNTPs with the 3′-hydroxy group at the end of the nascent (elongating) DNA strand.

5. Final elongation: This single step is optional, but is performed at a temperature of 70–74 °C (158–165 °F) (the temperature range required for optimal activity of most polymerases used in PCR) for 5–15 minutes after the last PCR cycle to ensure that any remaining single-stranded DNA is fully elongated.
6. Final hold: The final step cools the reaction chamber to 4–15 °C (39–59 °F) for an indefinite time, and may be employed for short-term storage of the PCR products.

B. PID Temperature Controller System

PID temperature control is a loop control feature found on most process controllers to improve the accuracy of the process. The purpose is to maintain the process temperature at the setpoint for the desired period of time, avoiding any severe changes from lag, overshoot or disturbances. PID temperature controllers work using a formula to calculate the difference between the desired temperature setpoint and current process temperature, then predict how much power to use in subsequent process cycles to ensure the process temperature remains as close to the setpoint as possible by eliminating the impact of process environment changes. PID temperature controllers differ from On/Off temperature controllers where 100% power is applied until the setpoint is reached, at which point the power is cut to 0% until the process temperature again falls below the setpoint. PID temperature controllers system formula:

$$u(t) = K_\mathrm{p} e(t) + K_\mathrm{i} \int_0^t e(t)\,dt + K_\mathrm{d} \frac{de(t)}{dt},$$

Where,
u(t) = PID Control Variable
Kp = Proportional Gain (denoted as P (proportional))
e(t) = Error Value
Ki = Integral Gain (denoted as I (integral))
Kd = Derivative Gain (denoted as D (derivative))
de = Change in Error Value
dt = Change in Time

II. System Plan

A. Equipments

In preparation of building this circuit, several components were chosen before-hand. These sets of components will then be used to build a circuit to which will act as a thermal regulator for a PCR chamber.

1. Arduino Uno R3(5V)
Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). 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.

2. Jumper Wire
A jump wire (also known as jumper, jumper wire, jumper cable) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

3. Arduino Relay
A relay is a programmable electrical switch, which can be controlled by Arduino or any micro-controller. It is used to programmatically control on/off the devices, which use the high voltage and/or high current. It is a bridge between Arduino and high voltage devices.

4. Breadboard
Named after a literal bread board due to its appearance, breadboard is a rectangular plastic board with a bunch of tiny holes in it. The holes are used to attach wire and other components that will connect them to the Arduino.

5. Adapter (12V 10A)
An adapter or adaptor is a device that converts attributes of one electrical device or system to those of an otherwise incompatible device or system. Some modify power or signal attributes, while others merely adapt the physical form of one connector to another.

6. LCD Monitor (16 x 2 I2C)
Abbreviated from Liquid Crystal Display, LCD is a flat panel display that is used as a monitor. In this

case, to monitor the temperature of the system.

7. Temperature Sensor (DS18B20)
   DS18B20 were chosen as alternatives for thermistor as a heater being that both components don't require external calibration or require a calibration that is much simpler than what a thermistor will need. DS18B20 is a digital temperature sensor chip. Its component outputs a digital signal that would require a Wire library (from Arduino) to communicate with it. DS18B20 requires a form of calibration. This calibration is relatively simpler than that of a thermistor. To calibrate, it only needs to measure things with temperature that is known, could be done so by giving it a “triple-point” bath. In this bath, boiling water and melting ice is used. DS18B20 is a quite precise sensor with small mean of error, as long as it is calibrated properly. Another of its advantages is that in 1-wire, multiple can be used.

8. Peltier Plate (TEC1-12706)
   Thermoelectric cooling uses the Peltier effect to create a heat flux at the junction of two different types of materials. A Peltier cooler, heater, or thermoelectric heat pump is a solid-state active heat pump which transfers heat from one side of the device to the other, with consumption of electrical energy, depending on the direction of the current. Such an instrument is also called a Peltier device, Peltier heat pump, solid state refrigerator, or thermoelectric cooler (TEC) and occasionally a thermoelectric battery. It can be used either for heating or for cooling, although in practice the main application is cooling. It can also be used as a temperature controller that either heats or cools.

9. Resistor (4.7k Ω)
   A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. In electronic circuits, resistors are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses. High-power resistors that can dissipate many watts of electrical power as heat, may be used as part of motor controls, in power distribution systems, or as test loads for generators. Fixed resistors have resistances that only change slightly with temperature, time or operating voltage. Variable resistors can be used to adjust circuit elements (such as a volume control or a lamp dimmer), or as sensing devices for heat, light, humidity, force, or chemical activity.

B. Schematic Circuit

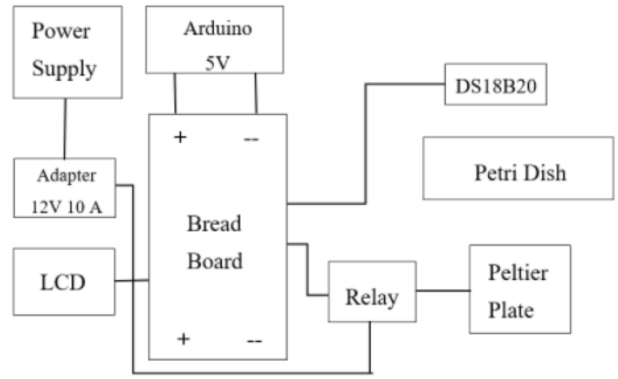


Fig. 2. Schematic of the Circuit

An Arduino with 5V is chosen in this circuit. The Arduino was then connected to a Breadboard. The power supply was connected to the adapter which was directly connected to the relay. Then the LCD Monitor was connected to the Breadboard as well. The LCD Monitor acted as a display for the Temperature. A Temperature Sensor (DS18B20) is connected to the Breadboard as well. The Temperature Sensor will be a sensor to monitor the temperature. A system of Peltier Plate also connected to the relay which also connected to Breadboard. The Peltier plate will act as a heater. Both Heater and Temperature Sensor are connected directly to the PCR Chamber (Petri Dish). This will then provide heat to the Chamber which has been regulated.

C. *System Flow*

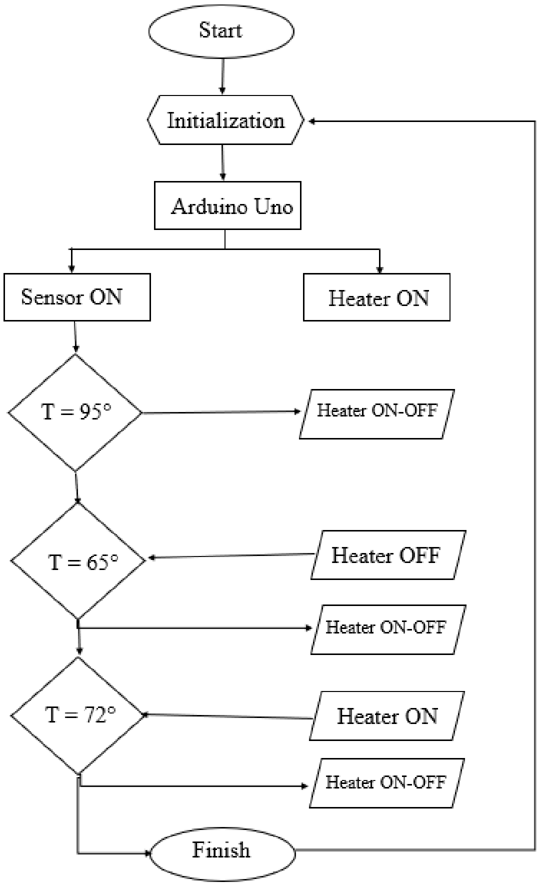


Fig. 3. System Flow Diagram

The system is started which will initialize the heating system. The system is heated until the temperature of 95°C is achieved. After that, the PID system is triggered ON to maintain the temperature. Then, the heater is turned on-off 10 seconds after the temperature of 95°C is achieved. With the heater system OFF, the temperature will drop to 65°C. To which then the PID system is turned back ON and the heater will turn on-off for 10 seconds. With the heater ON, the temperature will rise to 72°C. Then the PID system will be turned ON for 10 seconds to maintain the temperature. Basically, 10 seconds after the temperature of 72°C is achieved, the system is finished. But due to the heater which is still on, the temperature will rise and the whole system will repeat and make it circle.

I. Building Circuit

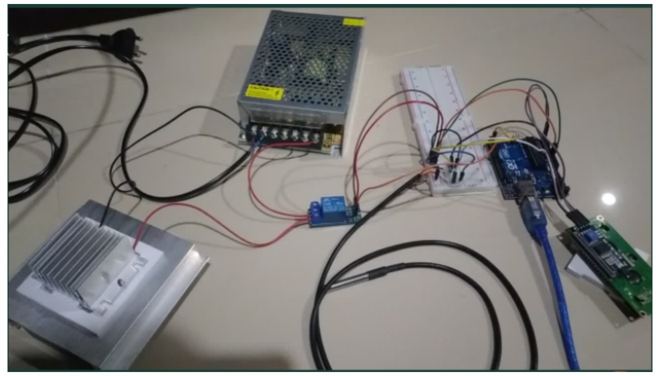
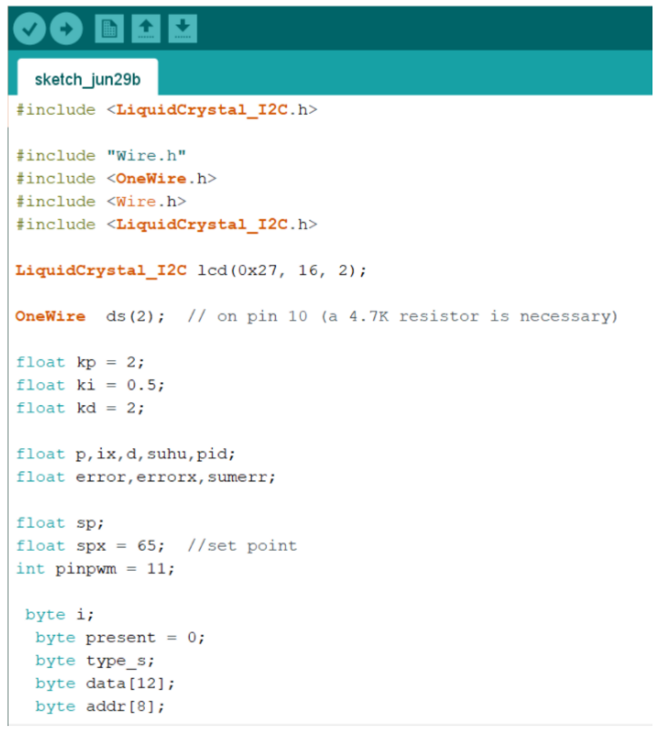


Fig 4. Circuit Building

II. Code



```
sketch_jun29b
#include <LiquidCrystal_I2C.h>

#include "Wire.h"
#include <OneWire.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

OneWire  ds(2);  // on pin 10 (a 4.7K resistor is necessary)

float kp = 2;
float ki = 0.5;
float kd = 2;

float p,ix,d,suhu,pid;
float error,errorx,sumerr;

float sp;
float spx = 65;  //set point
int pinpwm = 11;

 byte i;
  byte present = 0;
  byte type_s;
  byte data[12];
  byte addr[8];
```

Fig 5. Code Part 1

```
byte i;
 byte present = 0;
 byte type_s;
 byte data[12];
 byte addr[8];
 float celsius, fahrenheit;


void setup() {

  pinMode(pinpwm,OUTPUT);
  Serial.begin(9600);
  lcd.init();
  lcd.clear();
  lcd.noCursor();

}

void loop() {

  if ( !ds.search(addr)) {
    ds.reset_search();
    delay(250);
    return;
  }
```

Fig 6. Code Part 2

```
for( i = 0; i < 8; i++) {

}

if (OneWire::crc8(addr, 7) != addr[7]) {
    return;
}

switch (addr[0]) {
  case 0x10:
    type_s = 1;
    break;
  case 0x28:
    type_s = 0;
    break;
  case 0x22:
    type_s = 0;
    break;
  default:
    return;
}

ds.reset();
ds.select(addr);
ds.write(0x44, 1);        // start conversion, with parasite power on at the end
```

Fig 7. Code Part 3

```
ds.reset();
ds.select(addr);
ds.write(0x44, 1);        // start conversion, with parasite power on at the end

delay(1000);     // maybe 750ms is enough, maybe not

present = ds.reset();
ds.select(addr);
ds.write(0xBE);         // Read Scratchpad

for ( i = 0; i < 9; i++) {           // we need 9 bytes
  data[i] = ds.read();
}

int16_t raw = (data[1] << 8) | data[0];
if (type_s) {
  raw = raw << 3; // 9 bit resolution default
  if (data[7] == 0x10) {
    raw = (raw & 0xFFF0) + 12 - data[6];
  }
} else {
  byte cfg = (data[4] & 0x60);
  if (cfg == 0x00) raw = raw & ~7;  // 9 bit resolution, 93.75 ms
  else if (cfg == 0x20) raw = raw & ~3; // 10 bit res, 187.5 ms
```

Fig 8. Code Part 4

```
} else {
  byte cfg = (data[4] & 0x60);
  if (cfg == 0x00) raw = raw & ~7;  // 9 bit resolution, 93.75 ms
  else if (cfg == 0x20) raw = raw & ~3; // 10 bit res, 187.5 ms


  else if (cfg == 0x40) raw = raw & ~1; // 11 bit res, 375 ms

}

celsius = (float)raw / 16.0;
fahrenheit = celsius * 1.8 + 32.0;
suhu = celsius;


analogWrite(pinpwm,pid);

sp = spx + 30;  //atur range overlap kalibrasi

error = sp - suhu;
p = error * kp;
sumerr = error + errorx;
ix = ki * sumerr;
d = kd * (error - errorx);
pid = p + ix + d;
//pid = 255.0 - pid;
```

Fig 8. Code Part 5

```
//  if(pid < 1){
//  pid = 0;
//  }

 lcd.setCursor(0,0);
 lcd.print("Suhu=");
 lcd.print(suhu);
 lcd.print("/");
 lcd.print(spx);
 lcd.print("   ");

 lcd.setCursor(0,1);
 lcd.print("PID=");
 lcd.print(pid);
 lcd.print("     ");

Serial.println("suhu= ");
Serial.println(suhu);
Serial.println("pid= ");
Serial.println(pid);

delay(1000);

errorx = error;

}
```

Fig 9. Code Part 6

III. Result

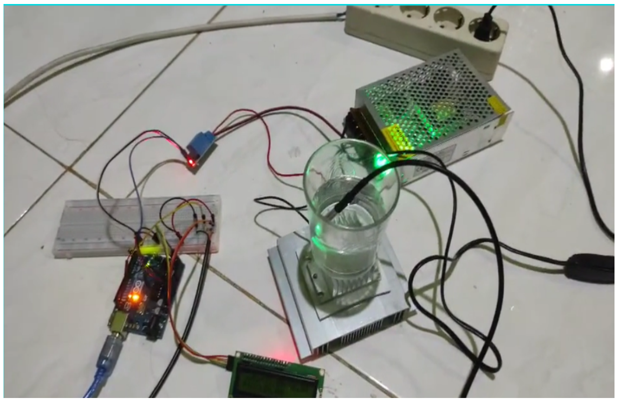


Fig 10. Result

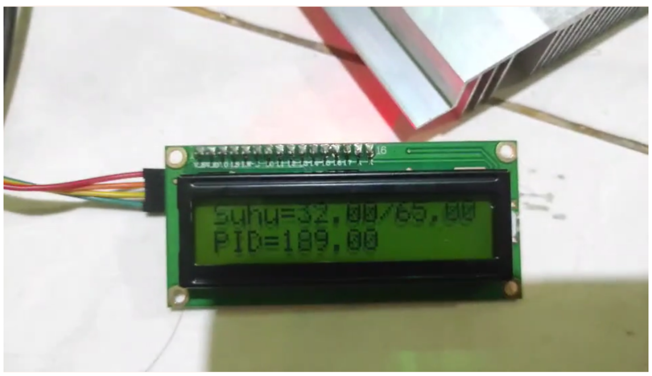


Fig 11. Temperature and PID Value

Fig 12. Temperature and PID Value at Setpoint

This is the circuit that has been made. We can see that all the components are already turned on and connected to the power supply and laptop. We can also see the water's temperature start to increase through the LCD. In the code we set the point in 65 degrees with kp ki and kd value 2, 0.5, and 2. The values were chosen based on the trial through PID control simulator. There are the obstacles we found in our code. We've looked up many references to modify our code so that we can set 3 set points. However, we are unable to do so. the code that we already have can only have 1 set point temperature. However we already tried, but the PID system in our code is only able to maintain that one set point of temperature for some amount of time.

IV. Analysis

The experiment is performed and then analyzed. Through analyzing, it is found that the experiment was ineffective. It is concluded that the problem lies in the coding of the system. the coding that was provided in the previous chapter was runned. The code could successfully initiate the LCD, DS18B20 sensor, and the Peltier Module which act as a heater. With that, it is noted that on top of the Peltier Module, a container that acts as a water chamber for the PCR is located.

When the whole system is run, the Peltier module will start to heat up and the DS18B20 sensor was put in the water chamber. The set point temperature was set at 60 C. It took about 3 hours for the water chamber to reach that temperature. This was then analyzed and through that it is concluded that the water chamber is quite large, it comes as no surprise that it took as long. A solution was suggested such as making the water chamber smaller, possibly containing only a few mm of water, and also not using the heatsink of the Peltier Module and opting for a different intermediary.

After the set point temperature is reached, the system has a hard time to maintain said temperature and has no capabilities to go to another set point. This is being that the coding of the system only has 1 set point temperature. An obstacle was found when trying to set 3 set point temperatures. Solution to this was suggested such as making the set point of temperature as a function, so that this function could be recalled to make 3 set point temperature.

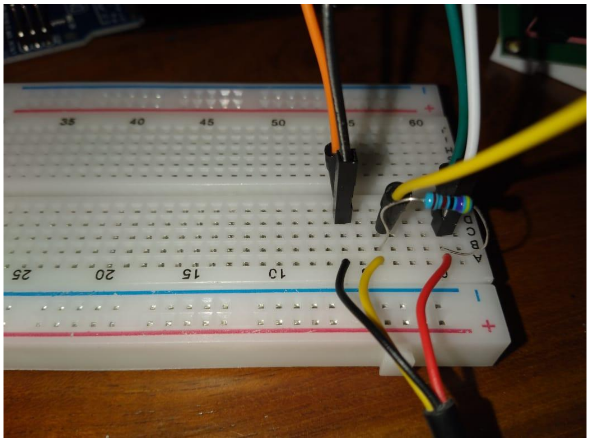
With the DS18B20 sensor, no problems were found as the sensor performed as per expectations. It could read temperature quite accurately and through coding, the specification of the temperature could also be set. The waterproof factor was also found to be quite functional. In consideration of the water chamber needing to be smaller, this sensor could still be used however it won't be as easy being that the sensor is not quite small enough to fit in the desired water chamber.

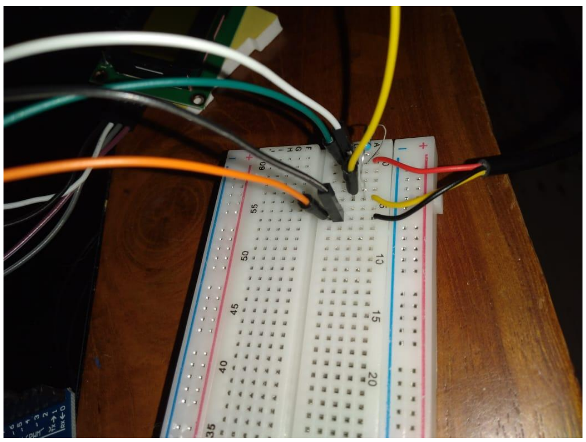
All things considered, the system worked well despite the problems it has. Improvements were unquestionably required.

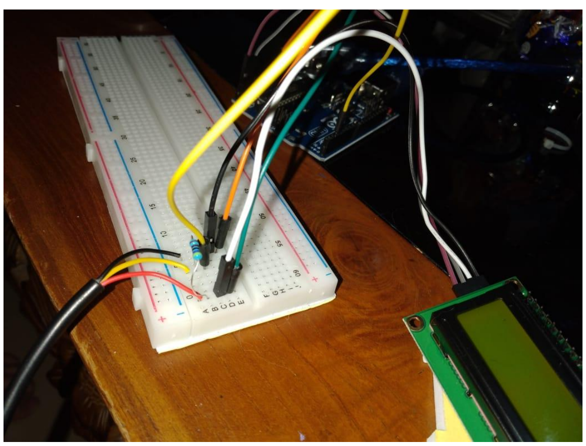
V. Conclusion

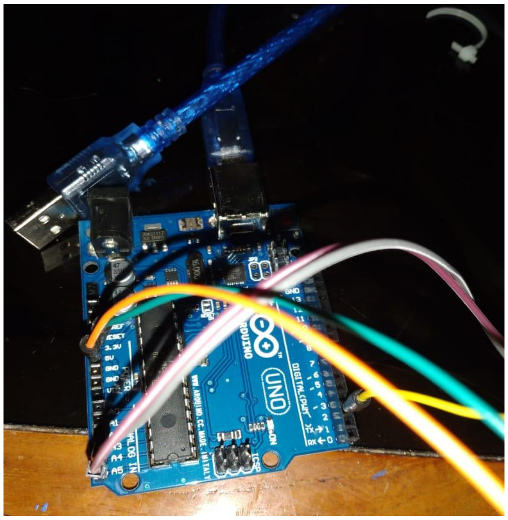
Through this project the principle of the PCR chamber can be understood. The principle of the PID controller system also can be comprehended. As the temperature closes with the set point, the PID value will decrease. Also, The procedure to build a simple PCR chamber can be known for the circuit and coding even though the code still needed to be improved.

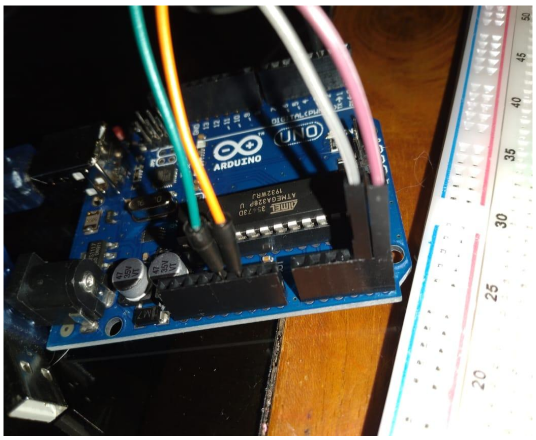
Appendix

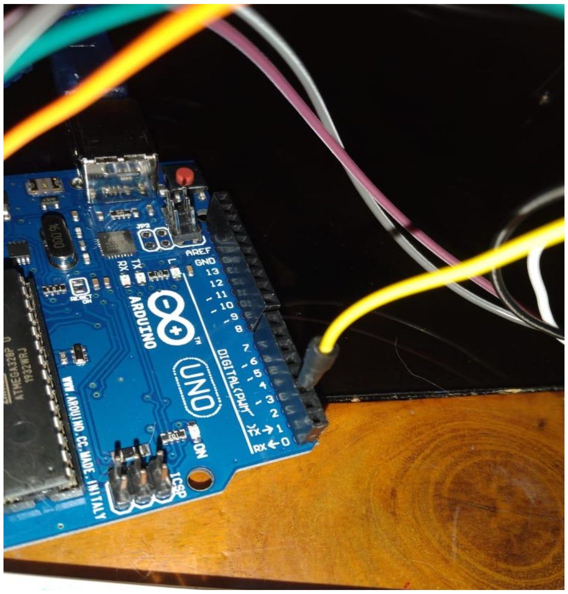


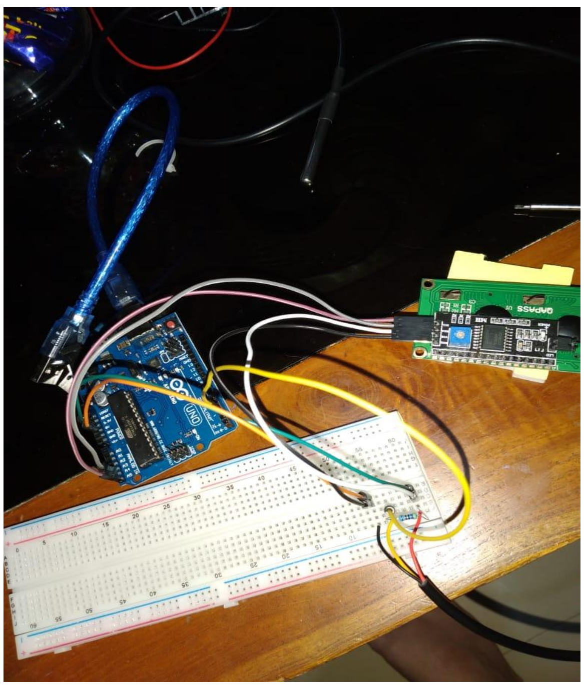


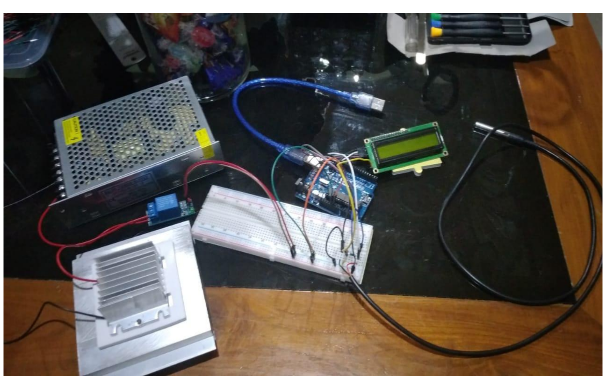


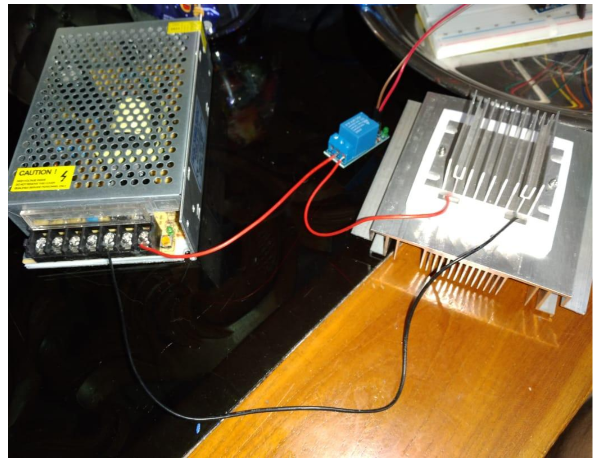

UNO
ARDUINO

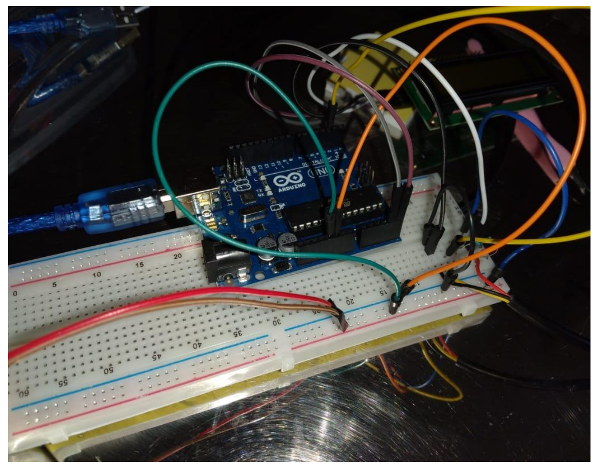

ARDUINO
UNO
20
25
30
35
40
45

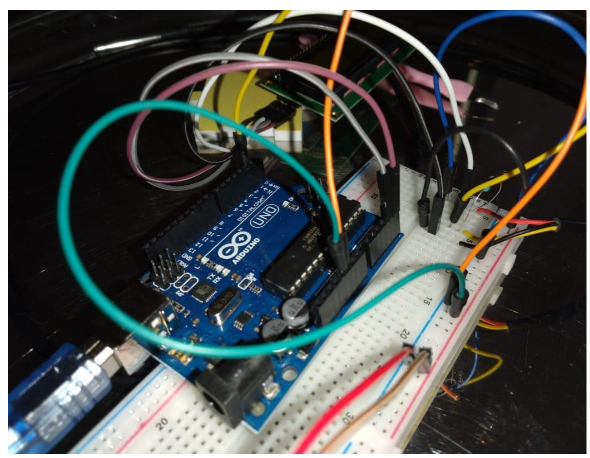












Acknowledgment

References and Footnotes

A. References

- https://arduinogetstarted.com/tutorials/arduino-relay#:~:text=A%20relay%20is%20a%20programmab le,Arduino%20and%20high%20voltage%20devices.

- https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf
- https://www.west-cs.com/products/l2/pid-temperature-controller/
- https://en.wikipedia.org/wiki/Adapter
- https://en.wikipedia.org/wiki/Arduino_Uno#:~:text=The%20Arduino%20Uno%20is%20an,(shields)%20and%20other%20circuits.
- https://en.wikipedia.org/wiki/Breadboard#:~:text=A%20breadboard%2C%20or%20protoboard%2C%20is,wood%20used%20when%20slicing%20bread.&text=Because%20the%20solderless%20breadboard%20does%20not%20require%20soldering%2C%20it%20is%20reusable.
- https://en.wikipedia.org/wiki/Jump_wire
- https://en.wikipedia.org/wiki/PID_controller
- https://en.wikipedia.org/wiki/Polymerase_chain_reaction
- https://en.wikipedia.org/wiki/Resistor#:~:text=A%20resistor%20is%20a%20passive,transmission%20lines%2C%20among%20other%20uses.
- https://en.wikipedia.org/wiki/Thermoelectric_cooling
- https://www.youtube.com/watch?v=eVRi9617Ta0
- https://www.youtube.com/watch?v=r8dqkEg5-v0
- https://www.youtube.com/watch?v=BPTb40hFN18&t=332s
- https://kursuselektronikaku.blogspot.com/2020/02/arduino-pid-control-suhu-sensor-ds18b20.html?m=1
- https://www.youtube.com/watch?v=cE3NUNh7aVI
- https://www.youtube.com/watch?v=Y1__vmkr8-g
- https://www.youtube.com/watch?v=SFPTzJZNAUY
- https://drive.google.com/file/d/1drxUHBdxFfw80-MqS21zA2HMmJ1N4KQN/view
- https://www.youtube.com/watch?v=qPYXSE-3gJY
- https://www.youtube.com/watch?v=LXhTFBGgskI
- https://miliohm.com/how-to-simply-use-ds18b20-temperature-sensor-with-arduino/
- https://www.youtube.com/watch?v=JTL3vzvTZac&t=146s
- https://www.youtube.com/watch?v=kc94lP9MFqo
- https://github.com/mishmashlabs/MAX31855_LCD_Relay/blob/main/Thermocouple_with_LCD_and_Relay/Thermocouple_with_LCD_and_Relay.ino
- https://www.youtube.com/watch?v=GW4YsLnrO2s
- http://grauonline.de/alexwww/ardumower/pid/pid.html
- https://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID
- https://www.youtube.com/watch?v=NFd4gz6TTBQ&t=1s
- https://create.arduino.cc/projecthub/mohannad-rawashdeh/arduino-control-ac-water-heater-temperature-e0712e?ref=search&ref_id=water%20heater&offset=5
- https://www.youtube.com/watch?v=kL-j2qaw_OI
- https://www.youtube.com/watch?v=QIdERbxDVuM
- https://www.youtube.com/watch?v=rVFXJl9bS1E
- https://www.youtube.com/watch?v=TbHsOgtCMDc