

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah dilakukan pengujian dari hasil Perancangan Proteksi Ketidakseimbangan Tegangan (*Unbalance Voltage*) Relay Digital Berbasis Arduino Uno, maka diperoleh kesimpulan sebagai berikut :

1. Dalam skripsi ini telah dilakukan penelitian tentang Perancangan Proteksi Ketidakseimbangan Tegangan (*Unbalance Voltage*) Relay Digital. Penelitian ini bertujuan untuk merancang sistem yang dapat mendeteksi dan melindungi peralatan listrik dari *unbalance voltage* dan tegangan yang tidak sesuai. Hasil penelitian ini menunjukkan bahwa penggunaan Arduino Uno sebagai dasar sistem proteksi memberikan sebuah solusi yang efektif dan ekonomis.
2. Pada kondisi *unbalance voltage*, gangguan akan terdeteksi ketika tegangan turunnya 10% dari 220 Volt dengan nilai gangguan 198 Volt, tegangan naik 5% dari 220 dengan nilai gangguan maksimal terjadi pada 131 Volt. Sementara itu, untuk tegangan turun gangguan akan terdeteksi ketika tegangan dalam kisaran antara 200 Volt hingga 220 Volt dianggap sebagai kondisi stabil.
3. Persentase hasil Rata-Rata Tegangan turun 10% dari setpoint (220) dinyatakan dalam persentase sebesar 0,8% dan Rata-Rata Tegangan naik 5% dari setpoint (220) sebesar 0,1%

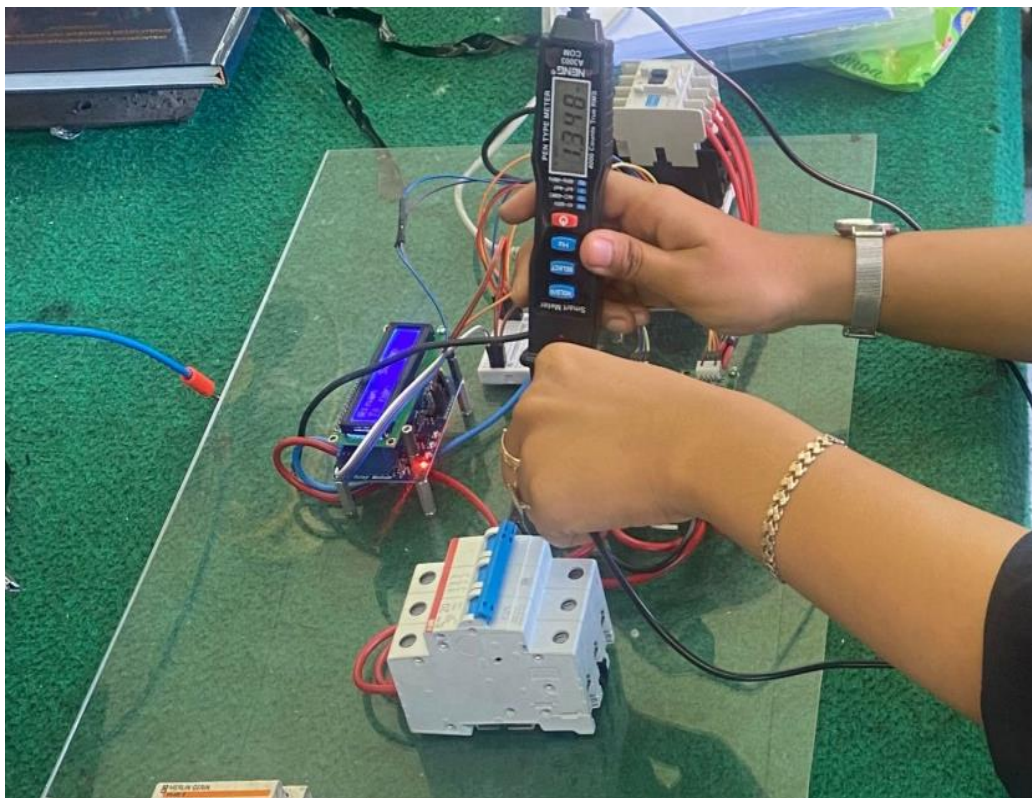
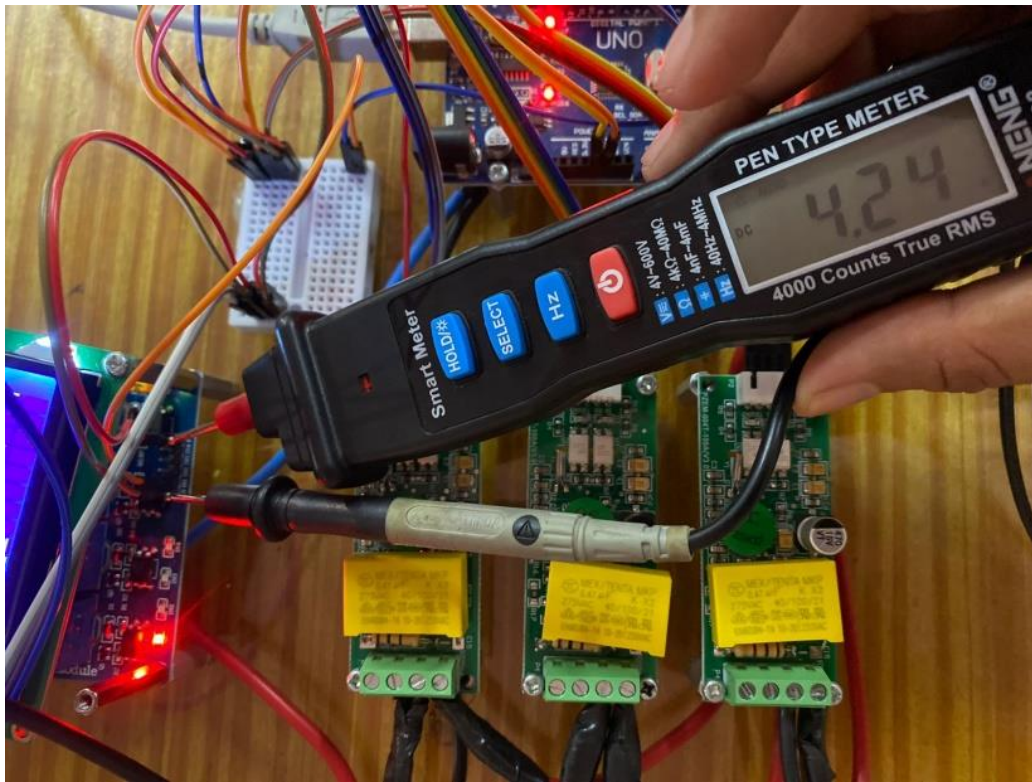
#### 5.2 Saran

Dari penelitian Tugas Akhir ini ada beberapa saran yang diajukan penulis untuk pengembangan selanjutnya antara lain sebagai berikut :

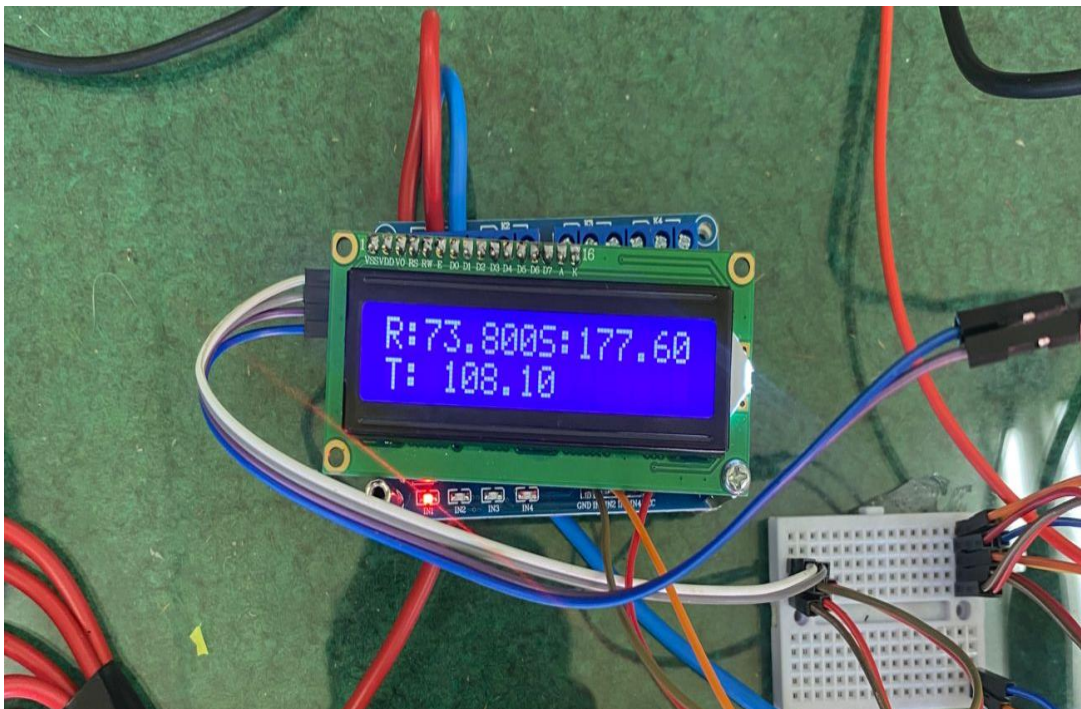
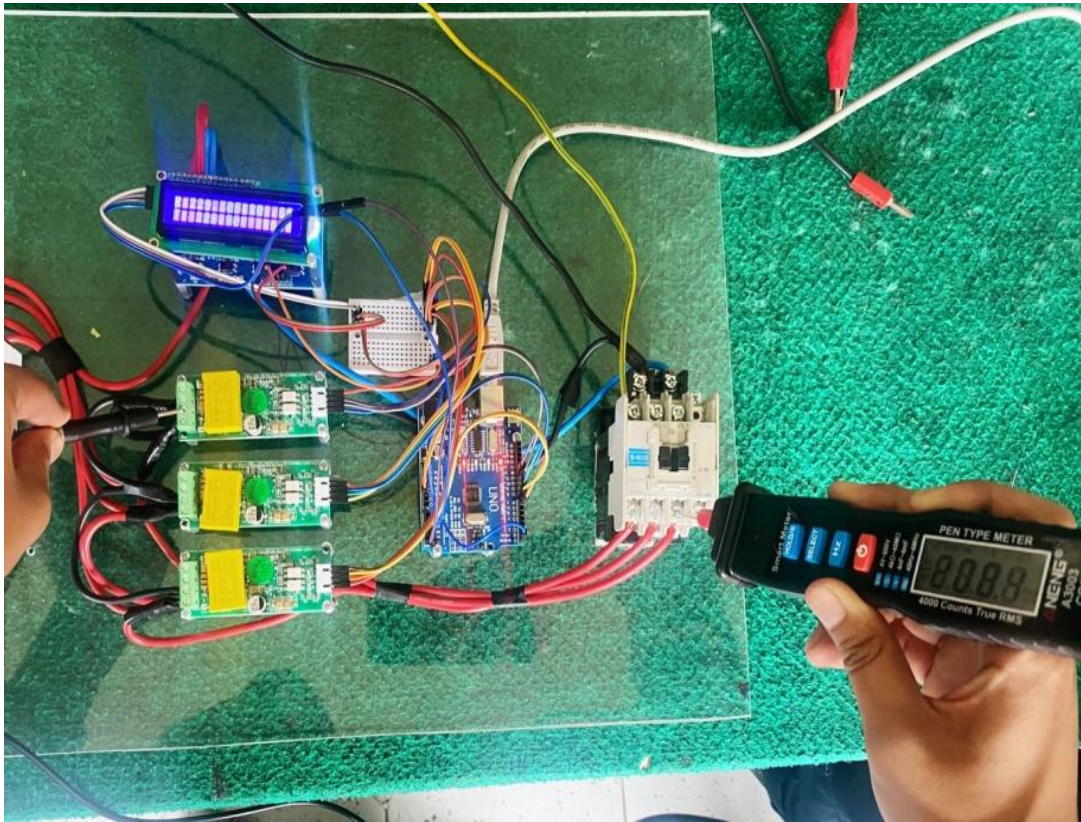
1. Diharapkan pada perancangan alat ini dapat dijadikan modul atau bahan pembelajaran kedepannya.
2. Diharapkan untuk penelitian selanjutnya agar bisa meningkatkan jika masih terdapat beberapa kekurangan

## DAFTAR PUSTAKA

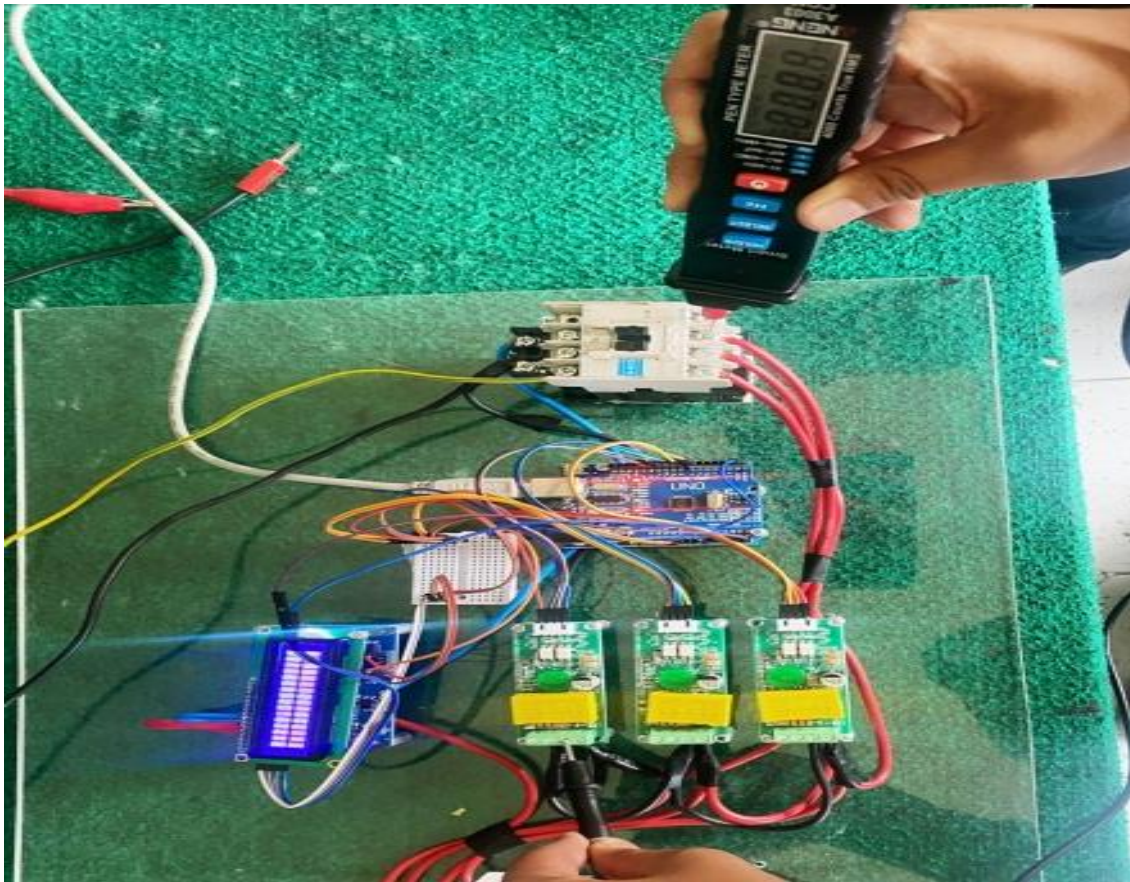
- [1] Kukuh Widarsono,dkk., 2019. Relay Protection of Over Voltage, Under Voltage and Unbalance Voltage Magnitude Based on Visual Basic Using Arduino Mega. Sampang, Indonesia.
- [2] Drs. F.J. Tasiam, M.Pd., 2017. PROTEKSI SISTEM TENAGA LISTRIK. Yogyakarta : Teknosain
- [3] Siti Khoirotun Nisa,dkk., 2015. Identifikasi Karakteristik Sinyal Arus Motor Induksi Tiga Fase Akibat Ketidakseimbangan Tegangan (Unbalance Voltage). Semarang
- [4] [https://www.researchgate.net/figure/Arduino-Uno-Microcontroller-Board\\_fig1\\_332175094](https://www.researchgate.net/figure/Arduino-Uno-Microcontroller-Board_fig1_332175094)
- [5] Paulus Setiawan, 2019. Analisis Pengaruh Tegangan Tidak Seimbang Pada Kinerja Motor Induksi Menggunakan Metode Transformasi Direct Qudrature
- [6] [https://elibrary.unikom.ac.id/id/eprint/542/8/UNIKOM\\_WILLIANA](https://elibrary.unikom.ac.id/id/eprint/542/8/UNIKOM_WILLIANA)
- [7] Pirza Hariyanto, Sepdian, Mahmud Idris, Maizal Isnen, 2019. Perancangan Alat Proteksi Tegangan Listrik Berlebih dan Menurun pada Jaringan Distribusi Tegangan Rendah
- [8] <https://www.elprocus.com/lcd-16x2-pin-configuration-and-its-working/>
- [10] Dhata Praditya, 2020. Perancangan Alat Pengaman Yang Mampu Mendeteksi Keseimbangan Suplai Tegangan, Kegagalan Fasa, Ketidakseimbangan Arus Beban, Arus Lebih, Tegangan Lebih Dan Tegangan Kurang
- [11] Ali Mashar, 2017 Pengaruh Ketidakseimbangan Tegangan Terhadap Unjuk Kerja Motor Induksi 5,5 kW
- [12] Ramadhan Firdiyanto Firdaus, 2021 Pemrograman Prototype Penstabil Tegangan Untuk Mengatasi Gangguan Over-Under Voltage Menggunakan Arduino
- [13] <https://www.khaddavi.net/2015/07/pengertian-kontaktor-dan-prinsip.html>









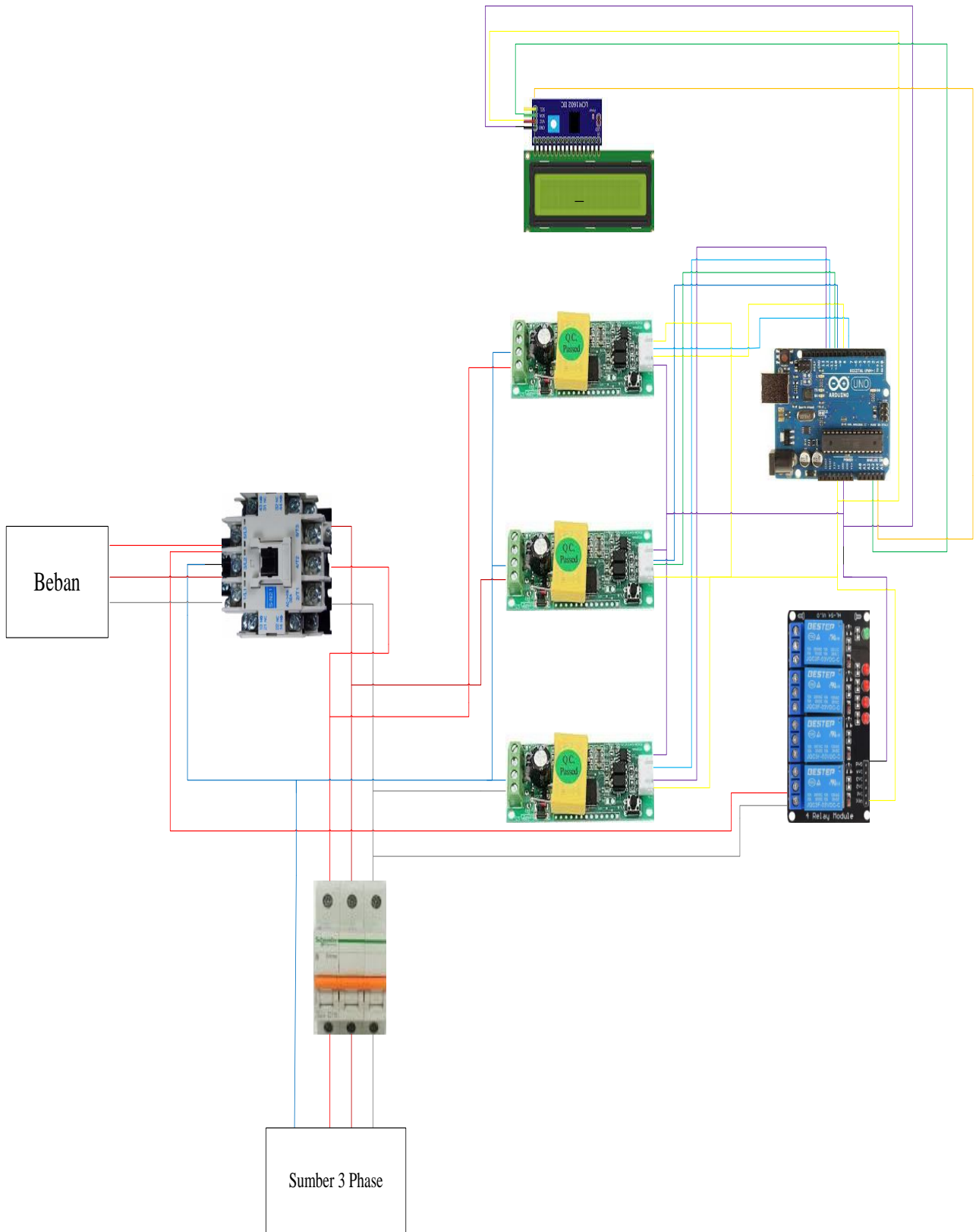


```
Anbalance_Voltage | Arduino 1.8.15
File Edit Sketch Tools Help

Anbalance_Voltage
1 #include <PZEM004Tv30.h>
2 #include <Wire.h>
3 #include <LiquidCrystal_I2C.h>
4 LiquidCrystal_I2C lcd (0x27, 16,2);
5
6 PZEM004Tv30 fasaR(12, 11); // Software Serial pin 8 (RX) & 9 (TX)
7 PZEM004Tv30 fasaS(10, 9);
8 PZEM004Tv30 fasaT(8, 7);
9
10 float fasaR, fasaS, fasaT;
11 int relay = 2;
12
13 float setPointR;
14 float setPointS;
15 float setPointT;
16
17 void setup() {
18   Serial.begin(9600);
19   lcd.init();
20   lcd.backlight();
21   pinMode(relay, OUTPUT);
22 }
23 }
```

AI Thinker ESP32-CAM, 240MHz (WiFi/BT), QIO, 80MHz on COM13

85°F Partly sunny Search 110 PM 8/15/2023



## Perancangan Program Arduino IDE

```
#include <PZEM004Tv30.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd (0x27, 16,2);

PZEM004Tv30 FasaR(12, 11); // Software Serial pin 8 (RX) & 9 (TX)
PZEM004Tv30 FasaS(10, 9);
PZEM004Tv30 FasaT(8, 7);

float fasaR, fasaS, fasaT;
int relay = 2;

float setPoinR;
float setPoinS;
float setPoinT;

void setup() {
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  pinMode(relay, OUTPUT);
}

void loop() {
  fasaR = FasaR.voltage();
  fasaS = FasaS.voltage();
  fasaT = FasaT.voltage();
```

```
Serial.print("R: ");
Serial.print(fasaR);

Serial.print(" S: ");
Serial.print( fasaS);

Serial.print(" T: ");
Serial.print(fasaT);
Serial.println(" V");
delay(1000);

lcd.setCursor(0,0);
lcd.print("R:");
lcd.setCursor(2,0);
lcd.print(fasaR);

lcd.setCursor(8,0);
lcd.print("S:");
lcd.setCursor(10,0);
lcd.print(fasaS);

lcd.setCursor(0,1);
lcd.print("T: ");
lcd.setCursor(3,1);
lcd.print(fasaT);

if (fasaR < 200) {
digitalWrite(relay, HIGH);
}
```



```
if (fasaS < 200) {  
    digitalWrite(relay, HIGH);  
}
```

```
if (fasaT < 200) {  
    digitalWrite(relay, HIGH);  
}
```

```
if (fasaR > 230) {  
    digitalWrite(relay, HIGH);  
}
```

```
if (fasaS > 230) {  
    digitalWrite(relay, HIGH);  
}
```

```
if (fasaT > 230) {  
    digitalWrite(relay, HIGH);  
}
```

```
else {  
    digitalWrite(relay, LOW);  
}
```

```
}
```

