LAMPIRAN

LAMPIRAN

Perhitungan Seri dan Pararel Baterai

Perhitungan koonfigurasi sel seri dan pararel baterai.

V baterai = 3,6 V

V total = 11,8 V

$$\Sigma bat = \frac{Vtot}{Vbat}$$

$$= \frac{11{,}8V}{3{,}6V}$$

$$= 3\ Sel\ Li - Ion$$

I baterai = 2,5 Ah

I total = 10 Ah

$$Itot = Icell1 + Icell2 + Icell3 + Icell4$$

$$= 2{,}5\ \text{Ah} + 2{,}5\ \text{Ah} + 2{,}5\ \text{Ah} + 2{,}5\ \text{Ah}$$

$$= 10\ \text{Ah}$$

Perhitungan Pembagi Tengangan

Perhitungan pembagi tegangan pada sensor tegangan.

Vin = 12,6 V (tegangan baterai)

R1 = 30000 **Ω**

R2 = 7500 **Ω**

Vo =...? (*Analog output*)

$$Vo = \frac{R2}{R2 + R1} \times Vi$$

$$Vo = \frac{7500}{7500 + 30000} \times 12{,}6$$

$$Vo = \frac{7500}{37500} \times 12{,}6$$

$$Vo = 2{,}52\ V$$

Listing Program pada Arduino IDE

```
#include <SPI.h>
#include <SD.h>
#include "max6675.h"
#include <DS3231.h>
#include <Wire.h>
#include <Fuzzy.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4);
DS3231  rtc(SDA, SCL);

//Deklarasi Sensor Tegangan
float vOUT1 = 0.0,vOUT2 = 0.0,vOUT3 = 0.0;
float vIN1 = 0.0,vIN2 = 0.0,vIN3 = 0.0;
float Cell1 = 0.0,Cell2 = 0.0,Cell3 = 0.0;
float R1 = 30000.0;
float R2 = 7500.0;
int value = 0;
const int Teg1 = A0;
const int Teg2 = A1;
const int Teg3 = A2;

//Hitung WATT
float watt = 0.0;
//Hitung SoC
float SoC = 0.0;
//////////////////Deklarasi Sensor Suhu
//const int lm35_data = A10; //pin data ke A0
//float nilai_analog_suhu; //int: bil.bulat contoh 1-10
//float nilai_suhu; //float: nilai sampai koma c: 29,44
byte Simbol_derajat=B11011111;

// Temperature Sensor1
int thermo1D0 = 24; // so
int thermo1CS = 23; //nilai suhu setiap sensor thermo
int thermo1CLK = 22; // sck
float temp1 = 0;
MAX6675 thermocouple1(thermo1CLK, thermo1CS, thermo1D0);
```

```
// Temperature Sensor2
int thermo2D0 = 27; // so
int thermo2CS = 26; //nilai suhu setiap sensor thermo
int thermo2CLK = 25; // sck
float temp2 = 0;
MAX6675 thermocouple2(thermo2CLK, thermo2CS, thermo2D0);

// Temperature Sensor3
int thermo3D0 = 30; // so
int thermo3CS = 29; //nilai suhu setiap sensor thermo
int thermo3CLK = 28; // sck
float temp3 = 0;
MAX6675 thermocouple3(thermo3CLK, thermo3CS, thermo3D0);

//byte tanda1=B11111111;
byte battery[8] = { //icon for battery
  B01110,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111,
};
byte battery1[8] = { //icon for each battery
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
  B10000,
  B11111,
};
byte battery2[8] = { //icon for charge
  B01010,
  B11111,
  B10001,
  B10001,
```

```
 B11011,
 B00100,
 B00100,
};
byte termometru[8] = //icon for termometer
{
 B00100,
 B01010,
 B01010,
 B01110,
 B01110,
 B11111,
 B11111,
 B01110
};

//////////////////////Deklarasi Sensor Arus
//const int pinADC = A11;
//int sensitivitas = 66; //tegantung sensor arus yang digunakan 30A
//int nilaiadc = 0;
//int teganganoffset = 2500; //nilai pembacaan offset saat tidak ada arus yang lewat
//double teganganarus = 0;
//double nilaiarus = 0;

const int pinADCcharge = A3;
const int pinADCdischarge = A4;
int sensitivitasIN = 185; //tegantung sensor arus yang digunakan, yang ini 5A
int sensitivitasOUT = 66; //tegantung sensor arus yang digunakan, yang ini 30A
int nilaiadc1= 00,nilaiadc2= 00;
float teganganoffset = 2492.6; //nilai pembacaan offset saat tidak ada arus yang lewat
double teganganarus = 00;
double nilaiarus1 = 00, nilaiarus2 = 00;

//PWM MOSFET
int pwmpin =3;
//int pwmawal = 0;
//int pwmstep = 15;

//Deklarasi Relay
```

```
int relay1 = 4;//pemutus fully charge (Relay 1 NC to NO)
int relay2 = 5;//discharge (Relay 2 NO to NC)
int relay3 = 6;//pemutus suhu max
int relay4 = 7;//

// SD Card
const int CS = 53;
File dataku;
int suhu, humd, count = 1;

// DEKLARASI FUZZY
  int tempinputfuzzy = (temp2);
  int teginputfuzzy = (vIN1);
#include <Fuzzy.h>
Fuzzy *fuzzy = new Fuzzy();

// FuzzyInput suhu
FuzzySet *dingin          = new FuzzySet(0, 0, 25, 30);
FuzzySet *hangat          = new FuzzySet(25, 30, 20, 35);
FuzzySet *panas           = new FuzzySet(30, 35, 45, 45);

// FuzzyInput tegangan
FuzzySet *rendah          = new FuzzySet(10, 10, 12, 12.2);
FuzzySet *sedang          = new FuzzySet(12, 12.3, 12.3, 12.6);
FuzzySet *tinggi          = new FuzzySet(12.4, 12.6, 13, 13);

// FuzzyOutput pwm
FuzzySet *low             = new FuzzySet(220, 225, 225, 230);
FuzzySet *medium          = new FuzzySet(235, 240, 240, 245);
FuzzySet *high            = new FuzzySet(250, 255, 255, 260);

void setup()
{
 rtc.begin(); // Initialize the rtc object
  Serial.begin(9600);
  Serial.println("Datalogger Sensor");

  //pesan status SDcard
  Serial.print("Membaca SDcard...");
```

```
  if (!SD.begin(CS))
  {
   Serial.println("GAGAL/SDcard rusak!");
   while (1);
  }
  Serial.println("Berhasil");

// Print a message to the LCD.
  lcd.init();
  lcd.backlight();

  lcd.setCursor(0,0);
  lcd.print("  MONITORING TERMAL ");
  lcd.setCursor(0,1);
  //lcd.print("Tegangan (V)");
  lcd.print("BATERAI LITHIUM ION");
  lcd.setCursor(0,2);
  //lcd.print("Arus (A)");
  lcd.print("Khen Nuhfus Sanjaya");
  lcd.setCursor(0,3);
  //lcd.print("Suhu (C)");
  lcd.print("     3332150054");
  delay(5000);
  lcd.clear();

   //pinMode(0, INPUT);
   //pinMode(1, INPUT);
   //pinMode(2, INPUT);
   //pinMode(A1, INPUT);//baterai1
   //pinMode(A2, INPUT);//baterai2
   //pinMode(A3, INPUT);//baterai3
   //pinMode(A10, INPUT);//Suhu
   //pinMode(A11, INPUT);//arus
   pinMode(pwmpin, OUTPUT);//PWM
   pinMode(relay1, OUTPUT);//relay
   pinMode(relay2, OUTPUT);
   Serial.begin(9600);
}
void bacanilaisuhu(){
```

```
  temp1 = thermocouple1.readCelsius()*0.99;
  delay(5);
  temp2 = thermocouple2.readCelsius()*0.99;
  delay(5);
  temp3 = thermocouple3.readCelsius()*0.99;
  delay(5);
}

void bacanilaitegangan(){
 value = analogRead(Teg1);//(B2,B+)
  vOUT1 = (value * 4.515) / 1024.0;
  vIN1 = vOUT1 / (R2/(R1+R2));
    delay(5);
 value = analogRead(Teg2);//(B1,B2)
  vOUT2 = (value * 4.359) / 1024.0;
  vIN2 = vOUT2 / (R2/(R1+R2));
    delay(5);
 value = analogRead(Teg3);//dekat dengan negetif (B-,B1)
  vOUT3 = (value * 3.98) / 1024.0;
  vIN3 = vOUT3 / (R2/(R1+R2));
    delay(5);
}

void hitungnilaiteganganCell(){
  Cell3  = vIN3;
  Cell2  = vIN2 - Cell3;
  Cell1  = vIN1 - Cell3 - Cell2;
}

void hitungwatt(){
  watt = (vIN1 *10);// 2.5 =2500 mAH
  delay(5);
}
void hitungSoC(){
  SoC = ((vIN1- 7.5)/5.1)*100;
  delay(5);
}
void bacanilaiarus(){
  nilaiadc1 = analogRead(pinADCcharge);
```

```
  teganganarus = (nilaiadc1 / 1024.0) * 5000;
  nilaiarus1 = ((teganganarus - teganganoffset) / sensitivitasIN);
    nilaiadc2 = analogRead(pinADCdischarge);
  teganganarus = (nilaiadc2 / 1024.0) * 5000;
  nilaiarus2 = ((teganganarus - teganganoffset) / sensitivitasOUT);
}

//void aturpwm(){
//{
  //analogWrite(10, 0);
  //delay(50);
  //analogWrite(10, 15);
  //delay(50);
  //analogWrite(10, 30);
  //delay(50);
  //analogWrite(10, 45);
  //delay(50);
  //analogWrite(10, 60);
  //delay(50);
  //analogWrite(10, 75);
  //delay(50);
  //analogWrite(10, 90);
  //delay(50);
  //analogWrite(10, 105);
  //delay(50);
  //analogWrite(10, 120);
  //delay(50);
  //analogWrite(10, 135);
  //delay(50);
  //analogWrite(10, 150);
  //delay(50);
  //analogWrite(10, 165);
  //delay(50);
  //analogWrite(10, 180);
  //delay(50);
  //analogWrite(10, 195);
  //delay(50);
  //analogWrite(10, 210);
  //delay(50);
```

```
 //analogWrite(10, 225);
 //delay(50);
 //analogWrite(10, 240);
 //delay(50);
// analogWrite(3, 255);
// delay(50);
//}
void fuzzifikasi()
{
 Serial.begin(115200);
 pinMode(pwmpin , OUTPUT);

 //--------------------------------------
 // FuzzyInput
 FuzzyInput *suhu = new FuzzyInput(1);
 suhu->addFuzzySet(dingin);
 suhu->addFuzzySet(hangat);
 suhu->addFuzzySet(panas);
 fuzzy->addFuzzyInput(suhu);
 // FuzzyInput
 FuzzyInput *tegangan = new FuzzyInput(2);
 tegangan->addFuzzySet(rendah);
 tegangan->addFuzzySet(sedang);
 tegangan->addFuzzySet(tinggi);
 fuzzy->addFuzzyInput(tegangan);
 // FuzzyOutput
 FuzzyOutput *pwm = new FuzzyOutput(1);
 pwm->addFuzzySet(low);
 pwm->addFuzzySet(medium);
 pwm->addFuzzySet(high);
 fuzzy->addFuzzyOutput(pwm);
 //--------------------------------------(RXONE Electronics)---//
 // Building FuzzyRule////////////////////////////////////////////////////////////////// 1
 FuzzyRuleAntecedent *dingin_rendah = new FuzzyRuleAntecedent();
 dingin_rendah->joinWithAND(dingin, rendah);

 FuzzyRuleConsequent *high1 = new FuzzyRuleConsequent();
 high1->addOutput(high);
 FuzzyRule *fuzzyRule1 = new FuzzyRule(1, dingin_rendah, high1);
```

```
fuzzy->addFuzzyRule(fuzzyRule1);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 2
FuzzyRuleAntecedent *dingin_sedang = new FuzzyRuleAntecedent();
dingin_sedang->joinWithAND(dingin, sedang);
FuzzyRuleConsequent *high2 = new FuzzyRuleConsequent();
high2->addOutput(high);
FuzzyRule *fuzzyRule2 = new FuzzyRule(2, dingin_sedang, high2);
fuzzy->addFuzzyRule(fuzzyRule2);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 3
FuzzyRuleAntecedent *dingin_tinggi = new FuzzyRuleAntecedent();
dingin_tinggi->joinWithAND(dingin, tinggi);
FuzzyRuleConsequent *medium3 = new FuzzyRuleConsequent();
medium3->addOutput(medium);
FuzzyRule *fuzzyRule3 = new FuzzyRule(3, dingin_tinggi, medium3);
fuzzy->addFuzzyRule(fuzzyRule3);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 4
FuzzyRuleAntecedent *hangat_rendah = new FuzzyRuleAntecedent();
hangat_rendah->joinWithAND(hangat, rendah);
FuzzyRuleConsequent *high4 = new FuzzyRuleConsequent();
high4->addOutput(high);
FuzzyRule *fuzzyRule4 = new FuzzyRule(4, hangat_rendah, high4);
fuzzy->addFuzzyRule(fuzzyRule4);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 5
FuzzyRuleAntecedent *hangat_sedang = new FuzzyRuleAntecedent();
hangat_sedang->joinWithAND(hangat, sedang);
FuzzyRuleConsequent *medium5 = new FuzzyRuleConsequent();
medium5->addOutput(medium);
FuzzyRule *fuzzyRule5 = new FuzzyRule(5, hangat_sedang, medium5);
fuzzy->addFuzzyRule(fuzzyRule5);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 6
FuzzyRuleAntecedent *hangat_tinggi = new FuzzyRuleAntecedent();
hangat_tinggi->joinWithAND(hangat, tinggi);
FuzzyRuleConsequent *low6 = new FuzzyRuleConsequent();
low6->addOutput(low);
FuzzyRule *fuzzyRule6 = new FuzzyRule(6, hangat_tinggi, low6);
fuzzy->addFuzzyRule(fuzzyRule6);
// Building FuzzyRule////////////////////////////////////////////////////////////////// 7
FuzzyRuleAntecedent *panas_rendah = new FuzzyRuleAntecedent();
panas_rendah->joinWithAND(panas, rendah);
```

```
  FuzzyRuleConsequent *medium7  = new FuzzyRuleConsequent();
  medium7->addOutput(medium);
  FuzzyRule *fuzzyRule7 = new FuzzyRule(7, panas_rendah, medium7);
  fuzzy->addFuzzyRule(fuzzyRule7);
  // Building FuzzyRule////////////////////////////////////////////////////////////////////// 8
  FuzzyRuleAntecedent *panas_sedang = new FuzzyRuleAntecedent();
  panas_sedang->joinWithAND(panas, sedang);
  FuzzyRuleConsequent *low8 = new FuzzyRuleConsequent();
  low8->addOutput(low);
  FuzzyRule *fuzzyRule8 = new FuzzyRule(8, panas_sedang, low8);
  fuzzy->addFuzzyRule(fuzzyRule8);
  // Building FuzzyRule////////////////////////////////////////////////////////////////////// 9
  FuzzyRuleAntecedent *panas_tinggi = new FuzzyRuleAntecedent();
  panas_tinggi->joinWithAND(panas, tinggi);
  FuzzyRuleConsequent *low9 = new FuzzyRuleConsequent();
  low9->addOutput(low);
  FuzzyRule *fuzzyRule9 = new FuzzyRule(9, panas_tinggi, low9);
  fuzzy->addFuzzyRule(fuzzyRule9);
}
void loop() {
  bacanilaisuhu();
  bacanilaitegangan();
  hitungnilaiteganganCell();
  hitungwatt();
  hitungSoC();
  bacanilaiarus();
  fuzzifikasi();
  //aturpwm();
dataku = SD.open("LOGDATA.txt", FILE_WRITE);
  if (dataku)
  {
    dataku.print(count);
    dataku.print("\t");
    dataku.print(rtc.getDOWStr());
    dataku.print("\t");
    dataku.print(rtc.getDateStr());
    dataku.print("\t");
    dataku.print(rtc.getTimeStr());
    dataku.print("\t");
```

```
    dataku.print(temp1);
    dataku.print("\t");
    dataku.print(temp2);
    dataku.print("\t");
    dataku.print(temp3);
    dataku.print("\t");
    dataku.print(SoC);
    dataku.print("\t");
    dataku.print(Cell1);
    dataku.print("\t");
    dataku.print(Cell2);
    dataku.print("\t");
    dataku.print(Cell3);
    dataku.print("\t");
    dataku.print(nilaiarus1);
    dataku.print("\t");
    dataku.print(nilaiarus2);
    dataku.print("\t");
    dataku.print(watt);
    dataku.print("\t");
    dataku.print(tempinputfuzzy);
    dataku.print("\t");
    dataku.println(teginputfuzzy);
    dataku.close();
    Serial.println(" | Data Tersimpan");
  }
  else
  {
    Serial.println(" | Gagal Tersimpan!");
  }
  count = count + 1;
delay(1000);

//float temp2 = analogRead(temp2);
//float vIN1 = analogRead(vIN1);
  //---------------------------------------
  fuzzy->setInput(1, tempinputfuzzy);
  fuzzy->setInput(2, teginputfuzzy);
  fuzzy->fuzzify();
```

```
  int out_pwm = fuzzy->defuzzify(1);
  analogWrite(pwmpin , out_pwm);
  //--------------------------------------
  Serial.print("pot1    ");
  Serial.print(tempinputfuzzy);  Serial.println("  C");
  Serial.print("pot2    ");
  Serial.print(teginputfuzzy);  Serial.println("  V");
  //---------------------
  Serial.print(tempinputfuzzy);   Serial.print(" ");
  Serial.println(teginputfuzzy);
  Serial.println(" ");
  Serial.println("Result: ");
  Serial.print("       pwm: ");
  Serial.println(out_pwm);
  Serial.println("");
  Serial.println("");
  delay(1000);

//LCD
lcd.clear();
 lcd.setCursor(0,0);
 lcd.print(rtc.getTimeStr());
 lcd.setCursor(10,0);
 lcd.print("SoC=");
 lcd.print(SoC);
 lcd.print("%");
 //lcd.setCursor(14,0);
 //lcd.print("");
 //lcd.print(rtc.getDateStr());
lcd.createChar(0, battery);
lcd.setCursor(0,1);
lcd.write(0);
lcd.createChar(3, battery1);
lcd.setCursor(0,2);
lcd.write(3);
//lcd.createChar(4, battery2); //add charge icon
//lcd.setCursor(15,3);
//lcd.write(4);
lcd.setCursor(2,1);
```

```
lcd.print(vIN1);
lcd.print(" V");
  //Serial.print("Baterai 1");
lcd.setCursor(2,2);
lcd.print("");
lcd.print(Cell1);
//lcd.print(vIN1);
  //Serial.print("Baterai 2");
lcd.setCursor(7,2);
lcd.print(" ");
lcd.print(Cell2);
//lcd.print(vIN2);
  //Serial.print("Baterai 3");
lcd.setCursor(12,2);
lcd.print("  ");
lcd.print(Cell3);
//lcd.print(vIN3);
  //Tampilan Suhu
lcd.createChar(1, termometru);
lcd.setCursor(0,3);
lcd.write(1);
lcd.setCursor(2,3);
lcd.print(temp1);
lcd.setCursor(8,3);
lcd.print(temp2);
lcd.setCursor(14,3);
lcd.print(temp3);
//lcd.write(Simbol_derajat);
//lcd.setCursor(9,3);
//lcd.print("C");
delay(4000);
lcd.clear();
//sensor arus
//lcd.setCursor(9,1);
//lcd.print(nilaiarus);
//lcd.print("A");
 lcd.setCursor(0,0);
 lcd.print(rtc.getTimeStr());
 lcd.setCursor(10,0);
```

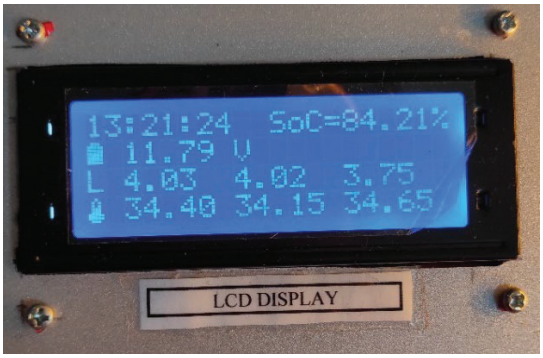
```
 lcd.print("SoC=");
 lcd.print(SoC);
 lcd.print("%");

 lcd.setCursor(1,1);
 lcd.print("Charge");
  lcd.setCursor(10,1);
 lcd.print("Discharge");
  lcd.setCursor(1,2);
  lcd.print(nilaiarus1);
  lcd.print(" A");
  lcd.setCursor(11,2);
  lcd.print(nilaiarus2);
  lcd.print(" A");
  lcd.setCursor(4,3);
  lcd.print(watt);
  lcd.print(" Watt");
  delay(4000);

}
```

Hasil Perancangan *Hardware*

Gambar 1 Tampilan utama sistem *monitoring*

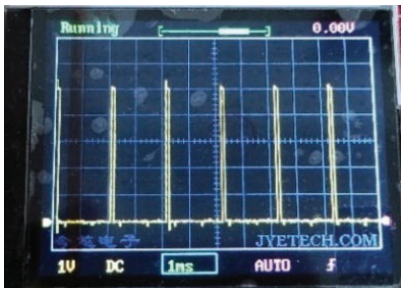


Gambar 2 Tampilan LCD sistem *monitoring*

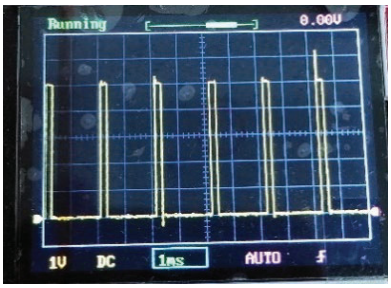
Gambar 3 Tampilan Baterai Lithium Ion

Hasil Pengujian Bentuk Gelombang pada MOSFET

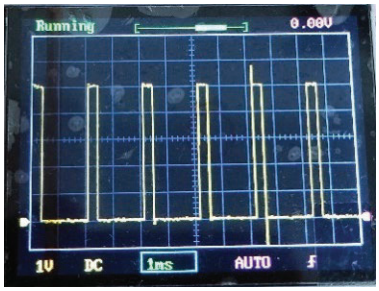
Gambar 1 Hasil Pengujian *Pulse Width Modulation* sebesar 0 atau 0%



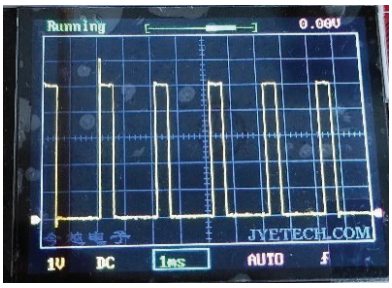
Gambar 2 Hasil Pengujian *Pulse Width Modulation* sebesar 15 atau 6%



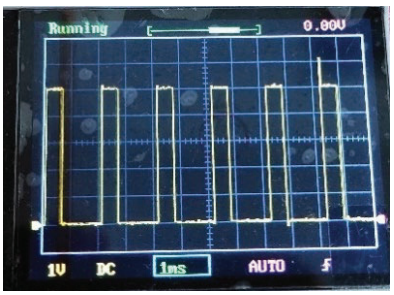
Gambar 3 Hasil Pengujian *Pulse Width Modulation* sebesar 30 atau 12%



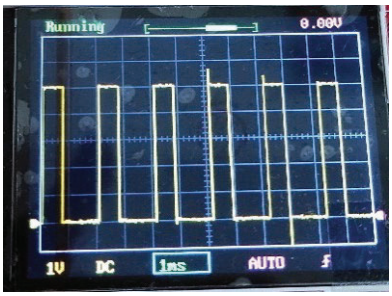
Gambar 4 Hasil Pengujian *Pulse Width Modulation* sebesar 45 atau 18%



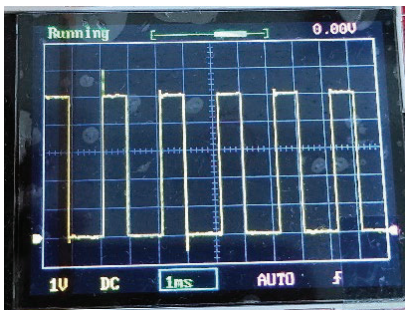
Gambar 5 Hasil Pengujian *Pulse Width Modulation* sebesar 60 atau 24%



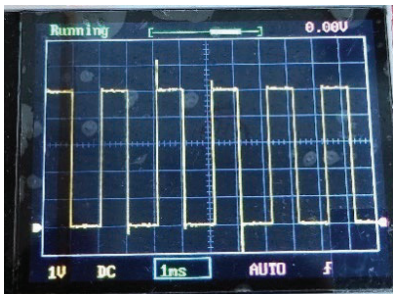
Gambar 6 Hasil Pengujian *Pulse Width Modulation* sebesar 75 atau 30%



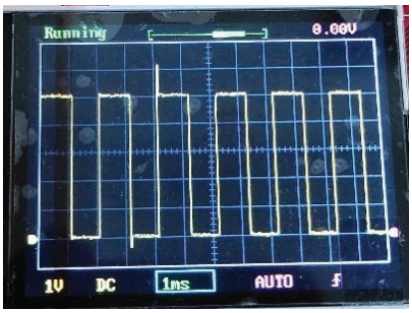
Gambar 7 Hasil Pengujian *Pulse Width Modulation* sebesar 90 atau 36%



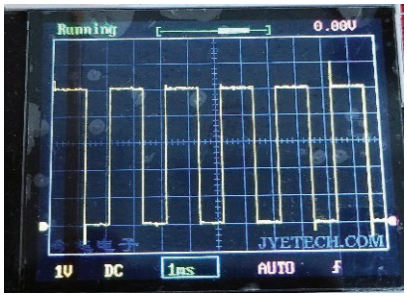
Gambar 8 Hasil Pengujian *Pulse Width Modulation* sebesar 105 atau 41%



Gambar 9 Hasil Pengujian *Pulse Width Modulation* sebesar 120 atau 47%

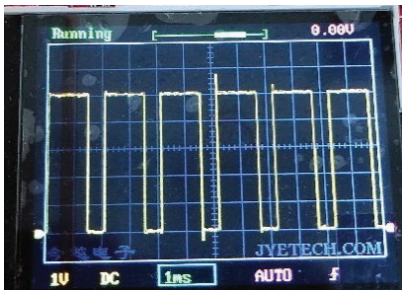


Gambar 10 Hasil Pengujian *Pulse Width Modulation* sebesar 135 atau 53%

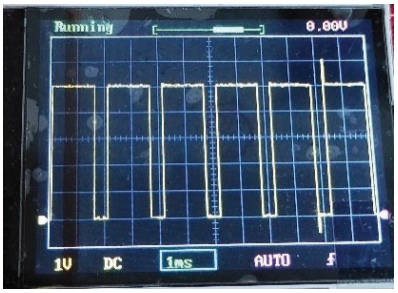


Gambar 11 Hasil Pengujian *Pulse Width Modulation* sebesar 150 atau 59%

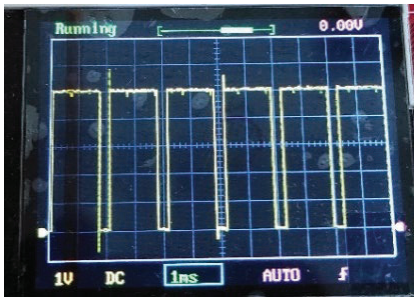
Gambar 12 Hasil Pengujian *Pulse Width Modulation* sebesar 165 atau 63%



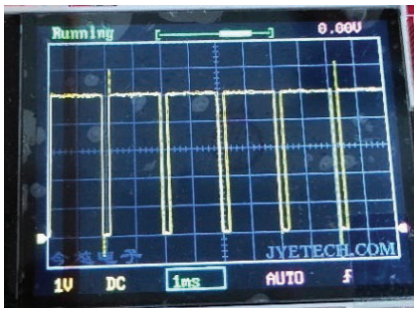
Gambar 13 Hasil Pengujian *Pulse Width Modulation* sebesar 180 atau 70%



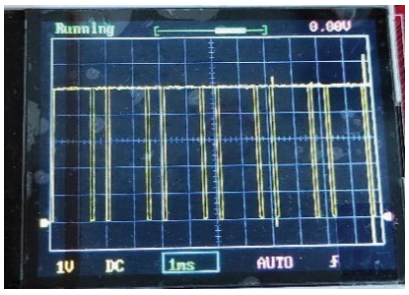
Gambar 14 Hasil Pengujian *Pulse Width Modulation* sebesar 195 atau 77%



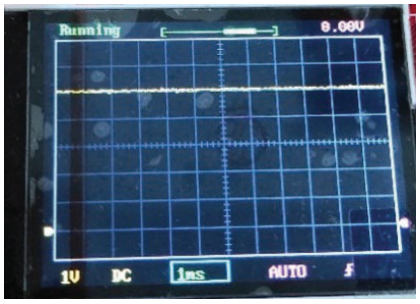
Gambar 15 Hasil Pengujian *Pulse Width Modulation* sebesar 210 atau 82%



Gambar 16 Hasil Pengujian *Pulse Width Modulation* sebesar 225 atau 88%



Gambar 17 Hasil Pengujian *Pulse Width Modulation* sebesar 240 atau 94%



Gambar 18 Hasil Pengujian *Pulse Width Modulation* sebesar 255 atau 100%