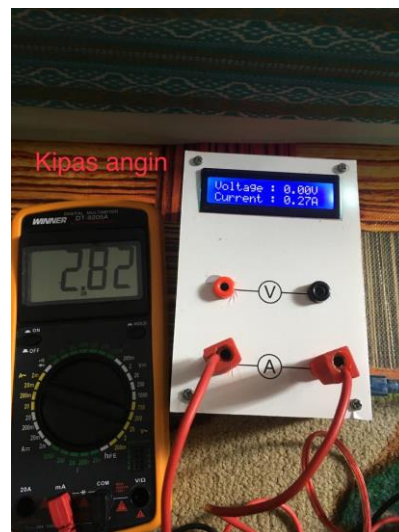


LAMPIRAN

Lampiran 1. Proses Pemasangan Komponen



Lampiran 2. Proses Pengambilan Data Sensor Arus

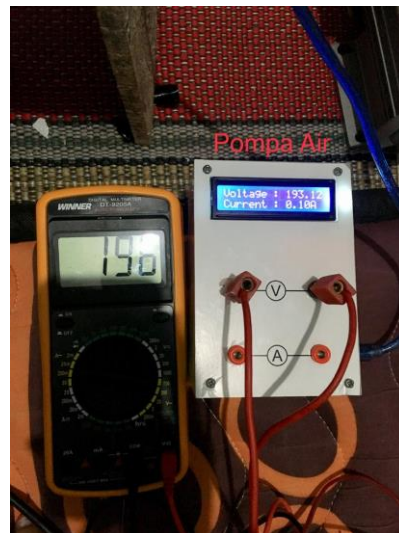
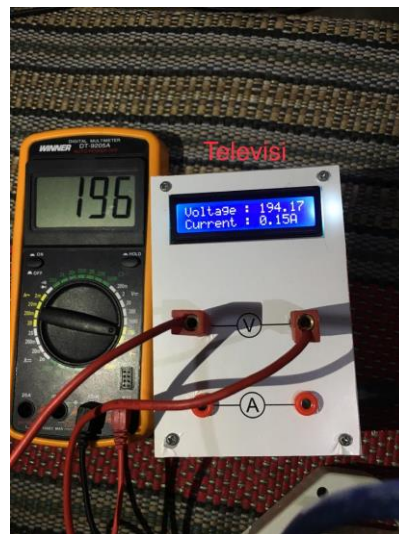
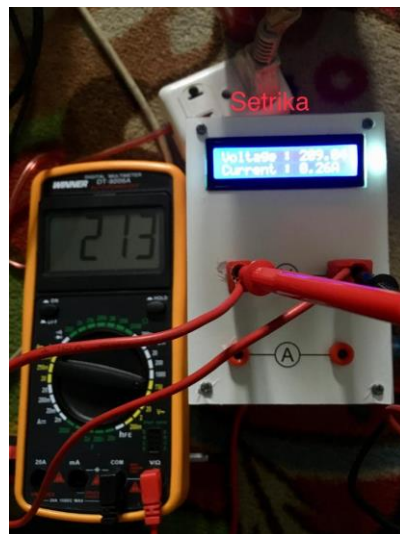




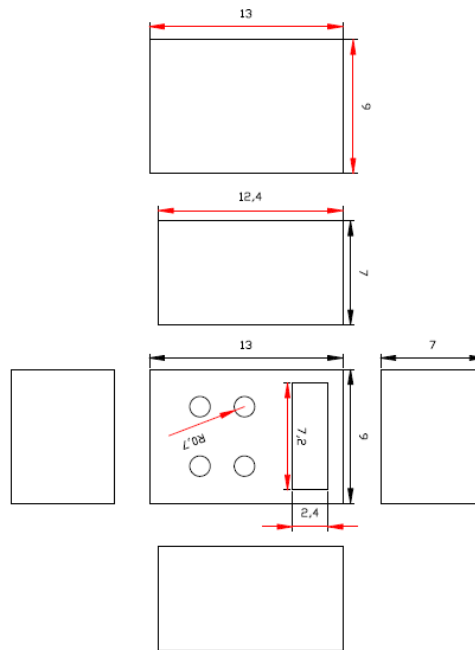


Lampiran 3. Proses Pengambilan Data Sensor Tegangan

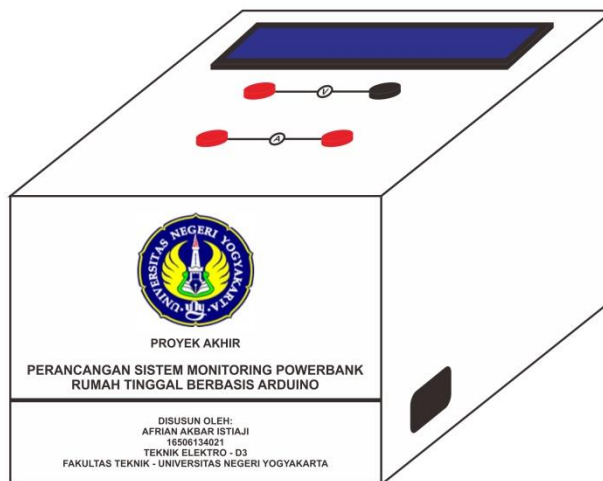




Lampiran 4. Desain 2D Akrilik



Lampiran 5. Desain 3D Akrilik



Lampiran 6. Kode Program

```
#include <LiquidCrystal_I2C.h>
#include <ACS712.h>
#include <SoftwareSerial.h>
String AP = "Maybach";          // CHANGE ME
String PASS = "mercedezbenz123"; // CHANGE ME
String API = "E861TI2HXJSDJCGE"; // CHANGE ME
String HOST = "api.thingspeak.com";
String PORT = "80";
String field = "field1";
String field1 = "field2";
int countTrueCommand;
int countTimeCommand;
boolean found = false;
SoftwareSerial esp8266(3,2);
LiquidCrystal_I2C lcd(0x27, 16, 2);
ACS712 sensorac(ACS712_20A, A1);

void setup() {
  lcd.begin();
  lcd.backlight();
  Serial.begin(9600);
  esp8266.begin(115200);
  sendCommand("AT",5,"OK");
  sendCommand("AT+CWMODE=1",5,"OK");
  sendCommand("AT+CWJAP=\"" + AP + "\",\"" + PASS
+"\"",20,"OK");
}

void loop() {

  //Tegangan AC
  double a = analogRead(A0);
  double VAC = a * (220/1023.0);
  double c = a * (4.87/1023.0);
  delay(1500);

  //arus AC
  float I = sensorac.getCurrentAC();

  //Serial Monitor
  Serial.print(I);
  lcd.setCursor(0,0);
  lcd.print("Voltage :");
  lcd.setCursor(10,0);
```

```

    lcd.print(VAC);
    lcd.print("V");
    lcd.setCursor(17,0);
    Serial.println(a);
    Serial.println(I);
    lcd.setCursor(0,1);
    lcd.print("Current :");
    lcd.setCursor(10,1);
    lcd.print(I);
    lcd.print("A");
    lcd.setCursor(17,1);

    //Upload Data
    Serial.println(String("V AC = ") + VAC + " V");
    String getData = "GET /update?api_key="+ API +"&"+
    field +"="+String(VAC)
    +""+"GET /update?api_key="+ API +"&"+ field1
    +""+String(I);
    sendCommand("AT+CIPMUX=1",5,"OK");
    sendCommand("AT+CIPSTART=0,\"TCP\", \""+ HOST +"\", "+
    PORT,15,"OK");
    sendCommand("AT+CIPSEND=0,"
    +String(getData.length()+4),4,">");

    esp8266.println(getData);delay(1500);countTrueCommand++
    ;
    sendCommand("AT+CIPCLOSE=0",5,"OK");
}
int getSensorData()
{
    return random(1000); // Replace with
}
void sendCommand(String command, int maxTime, char
readReplay[]) {
    Serial.print(countTrueCommand);
    Serial.print(". at command => ");
    Serial.print(command);
    Serial.print(" ");
    while(countTimeCommand < (maxTime*1))
    {
        esp8266.println(command);//at+cipsend
        if(esp8266.find(readReplay))//ok
        {
            found = true;
            break;
        }
    }
}

```

```
    }

    countTimeCommand++;
}

if(found == true)
{
    Serial.println("DONE");
    countTrueCommand++;
    countTimeCommand = 0;
}

if(found == false)
{
    Serial.println("FAIL");
    countTrueCommand = 0;
    countTimeCommand = 0;
}

found = false;
}
```