LAMPIRAN A
Program Sistem Arduino Nano

```
//Arduino Nano
//Perhitungan Nilai DoD, SoC, SoH Baterai
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

#include "ACS712.h"
#define pinvolt    A0 // sensor tegangan baterai
#define pincurrent  A1 // sensor arus pengisian
#define pincurrentl A2 // sensor arus beban
#define pinrelay    12 // relay charger otomatis
#define pinled      13

// konfigurasi sensor
ACS712 sensor(ACS712_05B, pincurrent);
ACS712 sensorl(ACS712_30A, pincurrentl);

// variable sensor
float current,currentl,voltage;
int soc,dod,soh;
floatcharger_ah,discharger_ah,counter_current,counter_current_load
;
int tanda_charger=0;

// kalibrasi sensor sensor
float zero_factor = -0.1; // zero untuk sensor current (default 0.0)
sensor arus pengisian
float zero_factorl = -0.2; // zero untuk sensor currentl (default 0.0)
sensor arus beban
float factorcurrent = 1.0; // factor kalibrasi arus (default 1.0) sensor
arus pengisian
float factorcurrentl = 1.0; // factor kalibrasi arusl (default 1.0) sensor
arus beban
float factorvoltage = 1.0; // factor kalibrasi tegangan (default 1.0)
sensor tegangan baterai
```

```arduino
float kapasitas_battery = 5.0; // ah
float kapasitas_awal = 0.0; // ah
float set_penuh = 12.5; // volt

unsigned long previousMillis = 0;
const long interval = 1000;
int counter=0;
int timer_send=0;
int timer_display=0;

void hitung_charger_ah(){
   counter_current += current;
   charger_ah = (counter_current / 3600.0) + kapasitas_awal;

}
void hitung_discharger_ah(){
   counter_current_load += currentl;
   discharger_ah = counter_current_load / 3600.0;

}
void hitung_soc(){
  soc = ((kapasitas_battery-charger_ah)/kapasitas_battery)*100.0;
}
void hitung_dod(){
  dod = (100.0-soc);
}
void hitung_soh(){
  soh = (kapasitas_battery /kapasitas_awal)*100.0;
}
void algoritma_charger(){
  if(voltage > 5 ){
     if(voltage >= set_penuh || charger_ah >= kapasitas_battery) {
       if(tanda_charger==0)
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Baterai Penuh");
delay(1000);
hitung_soh();
}
```

```cpp
digitalWrite(pinrelay, 1); // charger off
tanda_charger=1;
    }

  }
  if(voltage <= 5 ) { // kurang dari 5v
   digitalWrite(pinrelay, 1); // charger off
  }
}

// voltage sensor
float read_voltage(int pinsensor){
float adc_voltage = 0.0;
float in_voltage = 0.0;
float R1 = 30000.0;
float R2 = 7500.0;
float ref_voltage = 5.0;
int adc_value = 0;
adc_value = analogRead(pinsensor);
adc_voltage  = (adc_value * ref_voltage) / 1024.0;
in_voltage = adc_voltage / (R2/(R1+R2)) ;
return in_voltage;
}

void setup() {
  pinMode(pinrelay, OUTPUT);
  digitalWrite(pinrelay, 0); // on
  pinMode(pinled, OUTPUT);
  Serial.begin(9600);

  lcd.begin();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("DoD Baterai");
  delay(1000);

  Serial.println(" ready ");
```

```
}

void loop() {

  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    digitalWrite(pinled, digitalRead(pinled)^1);

    voltage=0;
    current=0;
    currentl=0;
    for(int i=0; i<100; i++){
    current += sensor.getCurrentDC();
    currentl += sensorl.getCurrentDC();
    voltage += read_voltage(pinvolt);
    delay(1);
    }
    current = current / 100.0;
    currentl = currentl / 100.0;
    current = (current - zero_factor)*factorcurrent;
    if(current<0.0)current=0;
    currentl = (currentl - zero_factorl)*factorcurrentl;
    if(currentl<0.0)currentl=0;
    voltage = (voltage / 100.0)*factorvoltage;

    algoritma_charger();

    hitung_charger_ah();
    hitung_soc();

    hitung_discharger_ah();
    hitung_dod();


    Serial.print("charger:");
    Serial.print(charger_ah);
    Serial.print("ah ");
    Serial.print("discharger:");
```

```
Serial.print(discharger_ah);
Serial.print("ah ");
Serial.println();

lcd.clear();
lcd.setCursor(0,0);
lcd.print(voltage,1);
lcd.print("V");
lcd.setCursor(6,0);
lcd.print(current,1);
lcd.print("A");
lcd.setCursor(11,0);
lcd.print(currentl,1);
lcd.print("A");

if(timer_display <= 5){
lcd.setCursor(0,1);
lcd.print(soc);
lcd.print("%");
lcd.setCursor(6,1);
lcd.print(dod);
lcd.print("%");
lcd.setCursor(11,1);
lcd.print(soh);
lcd.print("%");
}
else{
lcd.setCursor(0,1);
lcd.print(charger_ah);
lcd.print("ah");
lcd.setCursor(8,1);
lcd.print(discharger_ah);
lcd.print("ah");
}
if(++timer_display > 10) timer_display=0;

if(++timer_send>60){
Serial.print(" ");
Serial.print(voltage,1);
```

```
      Serial.print("V ");
      Serial.print(current,1);
      Serial.print("I ");
      Serial.print(currentl,1);
      Serial.print("L ");
      Serial.print(soc,1);
      Serial.print("C ");
      Serial.print(dod,1);
      Serial.print("D ");
      Serial.print(soh,1);
      Serial.print("H ");
      Serial.println();

      timer_send=0;
      }
    }
  }
```

```
//NODEMCU ESP8266
//Mengirim Data ke ThingSpeak dan Google SpreadSheet
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>
WiFiClient client1;

unsigned long myChannelNumber = 2221256;
const char * myWriteAPIKey = "W7UVMGI1RZ54DFD6";

#include <ESP8266WiFi.h>
//#include <WiFi.h>
#include <WiFiClientSecure.h>
#define ledpin    D4

const char* ssid = "mywifi"; //wifi name
const char* password = "354354354"; //password

//---------------------------------------Host & httpsPort
const char* host = "script.google.com";
const int httpsPort = 443;
//---------------------------------------

WiFiClientSecure client; //--> Create a WiFiClientSecure object.

String GAS_ID =
"AKfycby9QOFMYml564AYtp3QJaOflfEVvzvHiTEIrMLtVFmnE
YA9mKCT0Qfw1F8TsAU0Aoy4"; //--> spreadsheet script ID

unsigned long oldTime=0;
float voltage,current,currentl,soc,dod,soh;
String in="";

void test_kirim(){
  if(Serial.available()){
  char c = Serial.read();
  if(c=='a'){
```

```
      Serial.println("Test Kirim");
      send_spreadsheet(); // kirim data
      send_thingspeak();
   }
  }
}
void terima_data(){
  if(Serial.available()){
    char c = Serial.read();
    if(isDigit(c)||c=='.'||c=='-') in+=c;
    if(c==' ')in="";
    if(c=='V'){
      voltage = in.toFloat();
      Serial.print(" V:");
      Serial.print(in);
      in="";
    }
    if(c=='I'){
      current = in.toFloat();
      Serial.print(" I:");
      Serial.print(in);
      in="";
    }
    if(c=='L'){
      currentl = in.toFloat();
      Serial.print(" L:");
      Serial.print(in);
      in="";
    }
    if(c=='C'){
      soc = in.toFloat();
      Serial.print(" SOC:");
      Serial.print(in);
      in="";
    }
    if(c=='D'){
      dod = in.toFloat();
      Serial.print(" DOD:");
      Serial.print(in);
```

```cpp
      in="";
     }
    if(c=='H'){
     soh = in.toFloat();
     Serial.print(" SOH:");
     Serial.print(in);
     Serial.println();
     Serial.println();
     in="";
     send_spreadsheet(); // kirim ke spreadsheet
     send_thingspeak();
    }
  }
}

void setup() {
 // put your setup code here, to run once:
 Serial.begin(9600);
 pinMode(ledpin, OUTPUT);
 Serial.println();
 Serial.print("Connecting to ");
 Serial.println(ssid);

 ThingSpeak.begin(client1);  // Initialize ThingSpeak
 // Wi-Fi connection
 WiFi.begin(ssid, password);
 while (WiFi.status() != WL_CONNECTED) {
  delay(1000);
  Serial.print(".");
 }
 Serial.println("");
 Serial.println("WiFi connected");

 client.setInsecure();
 Serial.println(" ready ");
}
void loop() {
 terima_data();
```

```
 if ((millis() - oldTime) > 200) {
 oldTime = millis();
 digitalWrite(ledpin, digitalRead(ledpin)^1);
 }

//test_kirim();
}

void send_thingspeak(){
 ThingSpeak.setField(1, voltage);
 ThingSpeak.setField(2, current);
 ThingSpeak.setField(3, currentl);
 ThingSpeak.setField(4, soc);
 ThingSpeak.setField(5, dod);
 ThingSpeak.setField(6, soh);

 // write to the ThingSpeak channel
 int x = ThingSpeak.writeFields(myChannelNumber,
myWriteAPIKey);
 if(x == 200){
   Serial.println("thingspeak update successful.");
 }
 else{
   Serial.println("thingspeak problem updating. HTTP error code " +
String(x));
 }
}
// Subroutine for sending data to Google Sheets
void send_spreadsheet() {
 Serial.println("=========");
 Serial.print("connecting to ");
 Serial.println(host);

 //Connect to Google host
 if (!client.connect(host, httpsPort)) {
  Serial.println("connection failed");
  return;
 }
```

```
//Processing data and sending data

String _voltage =  String(voltage);
String _current =  String(current);
String _currentl =  String(currentl);
String _soc =  String(soc);
String _dod =  String(dod);
String _soh =  String(soh);
String url = "/macros/s/" + GAS_ID + "/exec?voltage=" + _voltage
+ "&current=" + _current + "&currentl=" + _currentl + "&soc=" +
_soc + "&dod=" + _dod + "&soh=" + _soh;

Serial.print("requesting URL: ");
Serial.println(url);
client.print(String("GET ") + url + " HTTP/1.1\r\n" +
    "Host: " + host + "\r\n" +
    "User-Agent: BuildFailureDetectorESP32\r\n" +
    "Connection: close\r\n\r\n");

Serial.println("request sent");
//Checking whether the data was sent successfully or not
while (client.connected()) {
  String line = client.readStringUntil('\n');
  if (line == "\r") {
    Serial.println("headers received");
    break;
  }
}
String line = client.readStringUntil('\n');
if (line.startsWith("{\"state\":\"success\"")) {
  Serial.println("successfull!");
} else {
  Serial.println("failed");
}
Serial.print("reply was : ");
Serial.println(line);
Serial.println("closing connection");
Serial.println("=========");
Serial.println();}
```

LAMPIRAN B
Data Pengujian pada Proses Pengosongan
Baterai *Lead-Acid* SE5-12 Baru

Waktu	Tegangan (V)	Arus *Discharge* (A)	SoC (%)	DoD (%)	SoH (%)
21:18:13	11.50	3.30	98.00	2.00	0.00
21:28:20	11.50	3.20	87.00	13.00	0.00
21:38:30	11.50	3.20	76.00	24.00	0.00
21:48:39	11.40	3.10	65.00	35.00	0.00
22:00:51	11.10	2.90	52.00	48.00	0.00
22:10:59	10.50	2.10	42.00	58.00	0.00
22:21:09	7.00	1.20	35.00	65.00	0.00
22:27:15	6.40	0.70	31.00	69.00	0.00
22:33:20	6.30	0.20	29.00	71.00	0.00
22:34:22	6.20	0.00	26.00	74.00	0.00

Data Pengujian pada Proses Pengisian
Baterai *Lead-Acid* SE5-12 Baru

Waktu	Tegangan (V)	Arus *Charge* (A)	SoC (%)	DoD (%)	SoH (%)
23:17:10	12.00	0.40	26.00	74.00	0.00
23:28:15	12.30	0.40	37.00	74.00	0.00
23:36:23	12.30	0.40	50.00	74.00	0.00
23:47:39	12.40	0.40	69.00	74.00	0.00
23:56:41	12.40	0.40	85.00	74.00	0.00
0:03:48	12.40	0.40	92.00	74.00	0.00
0:04:50	12.40	0.27	93.00	74.00	0.00
0:06:51	12.50	0.40	94.00	74.00	0.00
0:07:53	12.40	0.40	95.00	74.00	0.00
0:14:01	12.50	0.00	100.00	74.00	100.00

Data Pengujian pada Proses Pengosongan
Baterai *Lead-Acid* SE5-12 Bekas

Waktu	Tegangan (V)	Arus *Discharge* (A)	SoC (%)	DoD (%)	SoH (%)
14:23:23	11.50	3.30	97.00	3.00	0.00
14:34:33	11.20	2.90	77.00	23.00	0.00
14:44:42	11.20	2.20	59.00	41.00	0.00
14:53:56	11.00	1.40	44.00	56.00	0.00
14:58:56	10.90	0.90	35.00	65.00	0.00
15:02:59	10.80	0.50	29.00	71.00	0.00
15:04:00	10.80	0.40	27.00	73.00	0.00
15:06:02	10.70	0.20	24.00	76.00	0.00
15:07:02	10.60	0.10	22.00	78.00	0.00
15:08:03	10.60	0,00	20.00	80.00	0.00

Data Pengujian pada Proses Pengisian
Baterai *Lead-Acid* SE5-12

Waktu	Tegangan (V)	Arus *Charge* (A)	SoC (%)	DoD (%)	SoH (%)
18:50:32	12.10	0.20	20.00	80.00	0.00
19:06:48	12.20	0.30	28.00	80.00	0.00
19:19:02	12.30	0.30	39.00	80.00	0.00
19:26:06	12.30	0.30	42.00	80.00	0.00
19:31:14	12.30	0.30	44.00	80.00	0.00
19:49:27	12.30	0.30	56.00	80.00	0.00
20:06:42	12.40	0.30	72.00	80.00	0.00
20:15:59	12.40	0.30	80.00	80.00	0.00
20:42:16	12.40	0.30	99.00	80.00	0.00
20:43:16	12.50	0.30	100.00	80.00	68.00

LAMPIRAN C
Dokumentasi Kegiatan

Pengujian Sensor Arus ACS712 Pengisian dan Pengosongan	
Pengambilan Data Pengosongan dan Pengisian Baterai GTZ6V	
Pengambilan Data Pengosongan dan pengisian Baterai SE5-12	

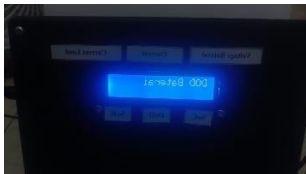| Tampilan LCD	
Pengosongan Baterai GTZ6V	Pengosongan Baterai SE5-12
Pengisian Baterai GTZ6V	Pengisian Baterai SE5-12
Tampilan awal LCD I2C	Tampilan Baterai Penuh

BIODATA PENULIS

Nama   : Shofi Alya
Tempat/Tanggal Lahir   : Bekasi, 11 Mei 2002
Agama   : Islam
Alamat   : Perum Graha Mustika Media, Blok B6
                         No.23 Rt.008/009, Kab.Bekasi, Jawa Barat
Email   : shofialya1105@gmail.com
Telepon/Hp   : 081905685703
Hobi   : Menonton Film
Motto   : Berani mencoba, bijaksana dalam
                         menghadapi, dan pantang menyerah.

Riwayat Pendidikan

Sekolah / Institusi/ Universitas	Jurusan	Periode
SD Negeri 2 Telajung	-	2008-2014
MTs Negeri 1 Bekasi	-	2014-2017
SMA Negeri 1 Setu	IPA	2017-2020
Politeknik Negeri Cilacap	Prodi D3 Teknik Listrik	2020-2023