

BAB 7 KESIMPULAN DAN SARAN

7.1. Kesimpulan

Dari penelitian yang telah dilakukan, maka dapat disimpulkan bahwa alat yang dirancang mampu melakukan proses pengukuran, monitoring (dengan menggunakan 3 alternatif yang ada), dan sekaligus mampu melakukan proses *data logging* secara *real time*. Selain itu, penambahan sistem otomasi pada alat dapat digunakan untuk mengurangi pemakaian energi listrik pada lampu yang tidak efisien. Untuk penggunaan komponen seperti sensor PIR mampu mendeteksi aktivitas atau gerakan manusia hingga 7 meter. Dari hasil uji coba dan kelayakan menunjukkan bahwa alat yang dirancang sudah baik dan layak untuk digunakan oleh konsumen atau pengguna. Selain itu, untuk hasil pengukuran, monitoring dan *data logging*, dapat digunakan oleh Universitas Atma Jaya Yogyakarta untuk melihat pola pemakaian dan dapat dijadikan dasar kebijakan untuk melakukan efisiensi yang berkelanjutan.

7.2. Saran

Terdapat beberapa saran yang ingin peneliti berikan pada hasil perancangan *prototype* alat ini:

- a. Untuk mendapatkan hasil pengukuran sensor arus yang lebih akurat, dapat mengganti dengan jenis sensor arus yang mampu mengukur arus AC dengan kinerja yang lebih baik misalnya *non invasive AC 100A current sensor*
- b. Komunikasi alat dapat diganti dari kabel menjadi *wireless* menggunakan modul *ESP8266* sehingga untuk melakukan proses monitoring dan *data logging* dapat dilakukan pada jarak yang jauh dari alat (tidak dibatasi panjang kabel lagi).
- c. Proses *data logging* yang dilakukan dapat ditambahkan sebuah sistem *database* yang dapat diakses dari berbagai komputer sehingga lebih terintegrasi.

DAFTAR PUSTAKA

- Adam, I., Mohamed, A., & Sanusi, H. (2011). Sensors & Transducers Improved Web-based Power Quality Monitoring Instrument, 132(9), 89–99.
- Akao, Y. (1990), Quality Function Deployment: Integrating Customer Requirements into Product Design. Translated by Mazur, G.H. New York: Productivity Press.
- Alciatore, D. G., & Hiestand, M. B. (2007). *Introduction to Mechatronics and Measurement Systems*.
- Barnes, R. M. (1980). Motion and Time Study: Design and Measurement of Work. John Wiley and Sons. New York.
- Boonsong, W., & Ismail, W. (2014). Wireless monitoring of household electrical power meter using embedded RFID with wireless sensor network platform. International Journal of Distributed Sensor Networks, 2014.
- Bossert, J.L. (1991), Quality Function Deployment: A Practitioner's Approach, ASQC Quality Press, Milwaukee, WI.
- Cross, N. (1994). *Engineering Design Methods, 2nd Ed*. Chichester: John Wiley and Sons Ltd.
- Dieter, G.E. & Schmidt, L.C. (2013). *Engineering Design (5th Edition)*. Singapore: Mc Graw-Hill.
- Dyer, R. F., Forman, E. H. (1991). An Analytic Approach To Marketing Decisions. USA : Prentice-Hall International Editions
- Fallo, F. C. (2017). Perancangan Prototype Sistem Otomasi Berbasis Mikrokontroler Untuk Lampu Penerangan Ruangan Di UAJY. Yogyakarta: Universitas Atma Jaya Yogyakarta.
- Giancoli, D.C. (2014). *Physics : Principles with Applications (7th Edition)*. US: Pearson Education, Inc.
- Gogineni, V. R., Matcha, K., & K, R. R. (2015). Real Time Domestic Power Consumption Monitoring using Wireless Sensor Networks, 5(4), 685–694.
- Gosavi, P. R. (2016). A Review on Industrial Energy Monitoring System Using PLC and SCADA, 5(1), 1699–1701.

- Groover, M. (2001). *Automation, Production Systems and Computer Integrated Manufacturing* (Vol. 2nd Edition). New Jersey: Prentice Hall.
- Haryanto, H., Permata, E., & Nainggolan, N. R. U. (2014). Sistem Monitoring Proses Produksi pada Mesin Bardi di PT . Tirta Investama (Danone Aqua) Sukabumi Berbasis Web, 3(October), 25–34.
- Kholiq, I. (2015). Pemanfaatan Energi Alternatif Sebagai Energi Terbarukan Untuk Mendukung Substitusi BBM. *Jurnal IPTEK*, 75-91.
- Mnati, M. J., Van den Bossche, A., & Chisab, R. F. (2017). A smart voltage and current monitoring system for three phase inverters using an android smartphone application. *Sensors (Switzerland)*, 17(4), 1-16
- Mundel, M. E. & Danner, D. L. (1994), *Motion & Time Study: Improving Productivity*, Seventh edition. USA : Prentice-Hall Publishing Company
- Muñoz, D. R., Pérez, D. M., Moreno, J. S., Berga, S. C., & Montero, E. C. (2009). Design and experimental verification of a smart sensor to measure the energy and power consumption in a one-phase AC line. *Measurement: Journal of the International Measurement Confederation*, 42(3), 412–419.
- Nunnally & Bernstein, I.H. (1994). *Psychometric Theory*, Edisi ke 3. New York : McGraw Hill.
- Patro, Chandra Sekhar; Prasad, M. V. (2013). a Study on Implementation of Quality Function Deployment Technique in Product Design Stage. *International Journal of Management Research and Reviews*, 3(6), 2966–2974.
- Peng, C., & Qian, K. (2014). Development and application of a ZigBee-based building energy monitoring and control system. *The Scientific World Journal*,
- Srividyadevi P., Pusphalatha D. V., & Sharma P.M. (2013). Measurement of Power and Energy Using Arduino. *Research Journal of Engineering Sciences ISSN Res. J. Engineering Sci*, 2(10), 2278–9472.
- Suryaningsih, S., Hidayat, S., & Abid, F. (2016). Rancang Bangun Alat Pemantau Penggunaan Energi Listrik Rumah Tangga Berbasis Internet, V(October 2016), SNF2016-ERE-87-SNF2016-ERE-90.

- Tan, H., Chen, S., Shi, Q., & Wang, L. (2014). Development of green campus in China. *Journal of Cleaner Production*, 64, 646–653.
- Thaung, H. N., Tun, Z. M., & Tun, H. M. (2016). Automatic Energy Control And Monitoring System For Building, 5(6), 125–129.
- Ulrich, K. T. & Eppinger, S. D. (2008). *Product design and development*(4th Edition). New York: Mc Graw-Hill.
- Widodo, I. D. (2005). *Perencanaan dan Pengembangan Produk*. Yogyakarta : UII Press Yogyakarta.
- Yuwono, T. (2015). Sistem Kendali Dan Monitoring Beban-Beban Listrik Rumah Berbasis Processing Line Comunication (PLC). *SimetriS*, 9(1), 34–42.
- Zare Mehrjerdi, Y. (2010). Quality function deployment and its extensions. *International Journal of Quality & Reliability Management*, 27(6), 616–640.
- Meilani. (2013). Teori Warna : Penerapan Lingkaran Warna dalam Berbusana. *Humaniora*, 4(9), 326–338.
- <https://www.arduino.cc/> diakses tanggal 5 Februari 2018
- www.qfdonline.com diakses tanggal 8 Maret 2018
- www.fritzing.org diakses tanggal 20 Maret 2018
- <https://www.farnell.com/datasheets/1682209.pdf> diakses tanggal 13 Maret 2018
- <https://www.allegromicro.com> diakses tanggal 14 Maret 2018
- <https://www.mpja.com/download/31227sc.pdf> diakses tanggal 14 Maret 2018
- <http://www.farrellf.com/TelemetryViewer/> diakses tanggal 27 Maret 2018

LAMPIRAN

Lampiran 1. Kuesioner Kebutuhan Konsumen

Saya Eric Gunarto selaku mahasiswa Teknik Industri Universitas Atma Jaya Yogyakarta memohon kesediaan waktu bapak/ibu atau saudara/i untuk mengisi kuesioner berikut ini yang bertujuan untuk mengetahui kebutuhan konsumen terhadap *prototype* alat ukur dan monitoring daya listrik pada lampu ruangan kampus UAJY yang akan dirancang penulis untuk pengerjaan Tugas Akhir.

Bagian A

Pada bagian A, jawaban diisi sesuai dengan pertanyaan yang telah diajukan.

1. Nama : _____
2. Pekerjaan : _____
3. Apakah bapak/ibu mengetahui *tool* yang dapat digunakan untuk melakukan pengukuran dan monitoring terhadap penggunaan listrik? Jika iya sebutkan yang bapak/ibu ketahui. (Jika tidak, dijawab “tidak” dan langsung ke no. 6)

4. Menurut bapak/ibu apakah kelebihan dari *tool* yang telah disebutkan di atas

5. Menurut bapak/ibu apakah kekurangan dari *tool* yang telah disebutkan di atas.

6. Apakah dibutuhkan sebuah alat atau *tool* untuk melakukan pengukuran dan monitoring penggunaan daya listrik di UAJY agar dapat dijadikan dasar dalam penghematan dan efisiensi energi listrik di UAJY? ____

7. Urutkan kriteria di bawah ini dari prioritas yang paling penting pada *tool*/monitoring yang akan dirancang (Berikan prioritas 1 sampai 6)

- | | |
|-----------------|--------------------------|
| A. Dimensi alat | D. Keakuratan pengukuran |
| B. Kemanan | E. Ketahanan/Keawetan |
| C. Warna | F. Kemudahan penggunaan |

Bagian B

Pada bagian B, jawaban diisi melingkari angka pada kolom ke 2 yang telah diberikan. Berikut keterangannya:

- 1 – Sangat tidak setuju
- 2 – Tidak setuju
- 3 – Netral
- 4 – Setuju
- 5 – Sangat Setuju

1. Alat monitoring listrik diperlukan di kampus UAJY	1	2	3	4	5
2. Material penyusun alat murah tetapi beresiko tidak awet	1	2	3	4	5
3. Alat tidak menimbulkan bahaya listrik atau konslet	1	2	3	4	5
4. Alat memiliki display yang menarik	1	2	3	4	5
5. Alat memiliki warna yang menarik	1	2	3	4	5
6. Hasil pengukuran alat yang akurat dan teliti	1	2	3	4	5
7. Alat memiliki dimensi yang mudah digunakan	1	2	3	4	5
8. Penggunaan alat dapat bertahan dalam jangka waktu panjang	1	2	3	4	5
9. Adanya petunjuk manual penggunaan alat	1	2	3	4	5
10. Alat didesain agar mudah digunakan oleh semua kalangan pengguna	1	2	3	4	5

Bagian C

Pada bagian ini, pilih dan lingkari salah satu jawaban yang telah disediakan pada opsi.

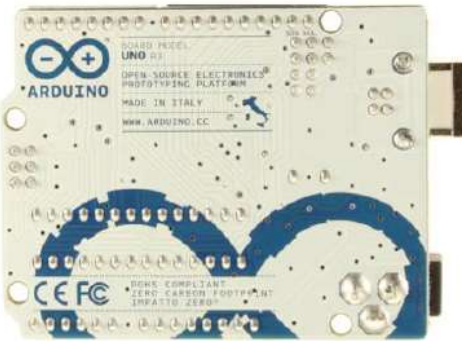
1. Dimensi keseluruhan alat yang diinginkan
 - a. Panjang \leq 30 cm & lebar \leq 20 cm
 - b. Panjang \leq 40 & lebar \leq 25 cm
 - c. Lainnya, panjang _____ & lebar _____
2. Alat monitoring yang dirancang diharapkan mampu bertahan selama
 - a. < 6 bulan
 - b. 6 - 12 bulan
 - c. 1 - 2 tahun
 - d. > 2 tahun
3. Toleransi hasil pengukuran dan monitoring
 - a. 2%
 - b. 3%
 - c. 5%
 - d. >5%
4. Warna yang diharapkan pada alat monitoring
 - a. Neutral (hitam & putih)
 - b. Warna Primer (merah, kuning, biru)
 - c. Warna Sekunder (orange, hijau, ungu)
 - d. Lainnya _____
5. Berapa pilihan/kombinasi warna yang diharapkan pada alat monitoring
 - a. 1 warna
 - b. 2 warna
 - c. 3 warna
 - d. 4 warna
6. Apakah dibutuhkan tambahan sistem terotomasi pada alat monitoring untuk mengendalikan nyala/mati lampu secara otomatis sesuai dengan kebutuhannya, sehingga dapat mereduksi biaya pemakaian daya listrik pada lampu ruangan di kampus UAJY?
 - a. Iya
 - b. Tidak

Lampiran 2. Datasheet Mikrokontroler Arduino Uno

Arduino Uno



Arduino Uno R3 Front



Arduino Uno R3 Back



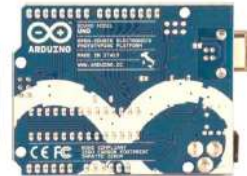
Arduino Uno R2 Front



Arduino Uno SMD



Arduino Uno Front



Arduino Uno Back

Summary

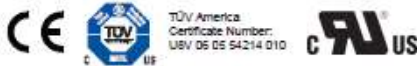
Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Lampiran 3. Datasheet Sensor Arus ACS712 30A

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Features and Benefits

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 μ s output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at $T_A = 25^\circ\text{C}$
- Small footprint, low-profile SOIC8 package
- 1.2 m Ω internal conductor resistance
- 2.1 kVRMS minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratiometric output from supply voltage



Package: 8 Lead SOIC (suffix LC)



Approximate Scale 1:1



Description

The Allegro™ ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switch-mode power supplies, and overcurrent fault protection. The device is not intended for automotive applications.

The device consists of a precise, low-offset, linear Hall circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which the Hall IC converts into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging.

The output of the device has a positive slope ($>V_{IOUT(Q)}$) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sampling. The internal resistance of this conductive path is 1.2 m Ω typical, providing low power loss. The thickness of the copper conductor allows survival of

Continued on the next page...

COMMON OPERATING CHARACTERISTICS¹ over full range of T_A , $C_F = 1$ nF, and $V_{CC} = 5$ V, unless otherwise specified

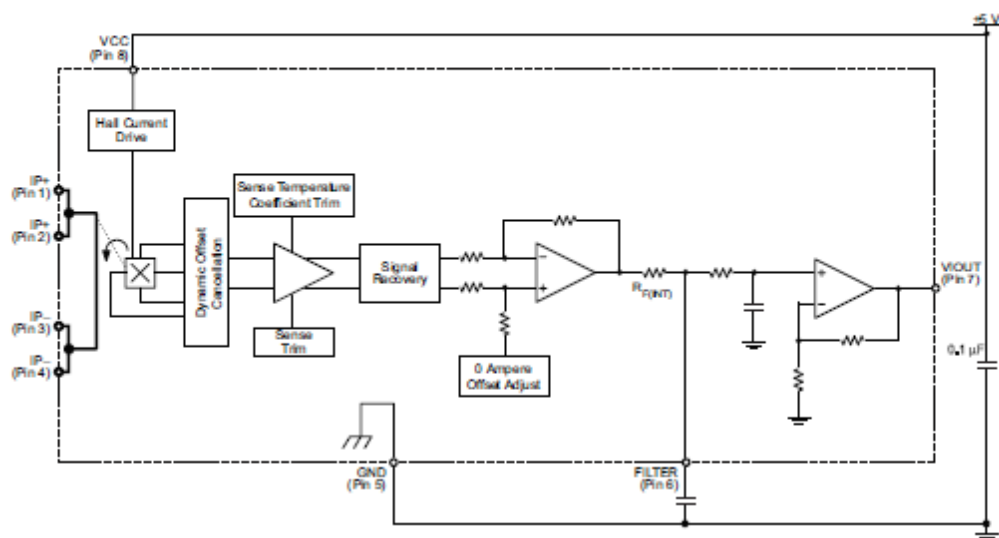
Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
ELECTRICAL CHARACTERISTICS						
Supply Voltage	V_{CC}		4.5	5.0	5.5	V
Supply Current	I_{CC}	$V_{CC} = 5.0$ V, output open	–	10	13	mA
Output Capacitance Load	C_{LOAD}	V _{IOUT} to GND	–	–	10	nF
Output Resistive Load	R_{LOAD}	V _{IOUT} to GND	4.7	–	–	k Ω
Primary Conductor Resistance	$R_{PRIMARY}$	$T_A = 25^\circ\text{C}$	–	1.2	–	m Ω
Rise Time	t_r	$I_P = I_P(\text{max})$, $T_A = 25^\circ\text{C}$, $C_{OUT} = \text{open}$	–	3.5	–	μ s
Frequency Bandwidth	f	–3 dB, $T_A = 25^\circ\text{C}$; I_P is 10 A peak-to-peak	–	80	–	kHz
Nonlinearity	E_{LIN}	Over full range of I_P	–	1.5	–	%
Symmetry	E_{SYM}	Over full range of I_P	98	100	102	%
Zero Current Output Voltage	$V_{IOUT(Q)}$	Bidirectional; $I_P = 0$ A, $T_A = 25^\circ\text{C}$	–	$V_{CC} \times 0.5$	–	V
Power-On Time	t_{PO}	Output reaches 90% of steady-state level, $T_J = 25^\circ\text{C}$, 20 A present on leadframe	–	35	–	μ s
Magnetic Coupling ²			–	12	–	G/A
Internal Filter Resistance ³	$R_{F(INT)}$			1.7		k Ω

¹Device may be operated at higher primary current levels, I_P , and ambient, T_A , and internal leadframe temperatures, T_A , provided that the Maximum Junction Temperature, $T_J(\text{max})$, is not exceeded.

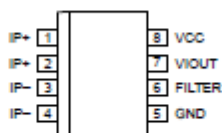
²IG = 0.1 mT.

³ $R_{F(INT)}$ forms an RC circuit via the FILTER pin.

Functional Block Diagram



Pin-out Diagram



Terminal List Table

Number	Name	Description
1 and 2	IP+	Terminals for current being sampled; fused internally
3 and 4	IP-	Terminals for current being sampled; fused internally
5	GND	Signal ground terminal
6	FILTER	Terminal for external capacitor that sets bandwidth
7	VIOUT	Analog output signal
8	VCC	Device power supply terminal

x30A PERFORMANCE CHARACTERISTICS¹ $T_A = -40^\circ\text{C}$ to 85°C , $C_F = 1\text{ nF}$, and $V_{CC} = 5\text{ V}$, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_p		-30	-	30	A
Sensitivity	Sens	Over full range of I_p , $T_A = 25^\circ\text{C}$	63	66	69	mV/A
Noise	$V_{\text{NOISE(PP)}}$	Peak-to-peak, $T_A = 25^\circ\text{C}$, 66 mV/A programmed Sensitivity, $C_F = 47\text{ nF}$, $C_{\text{OUT}} = \text{open}$, 2 kHz bandwidth	-	7	-	mV
Zero Current Output Slope	$\Delta V_{\text{OUT(Q)}}$	$T_A = -40^\circ\text{C}$ to 25°C	-	-0.35	-	mV/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.08	-	mV/°C
Sensitivity Slope	ΔSens	$T_A = -40^\circ\text{C}$ to 25°C	-	0.007	-	mV/A/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.002	-	mV/A/°C
Total Output Error ²	E_{TOT}	$I_p = \pm 30\text{ A}$, $T_A = 25^\circ\text{C}$	-	± 1.5	-	%

¹Device may be operated at higher primary current levels, I_p , and ambient temperatures, T_A , provided that the Maximum Junction Temperature, $T_{j(\text{max})}$, is not exceeded.

²Percentage of I_p , with $I_p = 30\text{ A}$, Output filtered.

Lampiran 4. Datasheet Sensor PIR HC-SR501

HC-SR501 PIR MOTION DETECTOR

Product Discription

HC-SR501 is based on infrared technology, automatic control module, using Germany imported LH778 probe design, high sensitivity, high reliability, ultra-low-voltage operating mode, widely used in various auto-sensing electrical equipment, especially for battery-powered automatic controlled products.

Specification:

- Voltage: 5V – 20V
- Power Consumption: 65mA
- TTL output: 3.3V, 0V
- Delay time: Adjustable (3->5min)
- Lock time: 0.2 sec
- Trigger methods: L – disable repeat trigger, H enable repeat trigger
- Sensing range: less than 120 degree, within 7 meters
- Temperature: – 15 ~ +70.
- Dimension: 32*24 mm, distance between screw 28mm, M2, Lens dimension in diameter: 23mm

Application:

Automatically sensing light for Floor, bathroom, basement, porch, warehouse, Garage, etc; ventilator, alarm, etc.

Features:

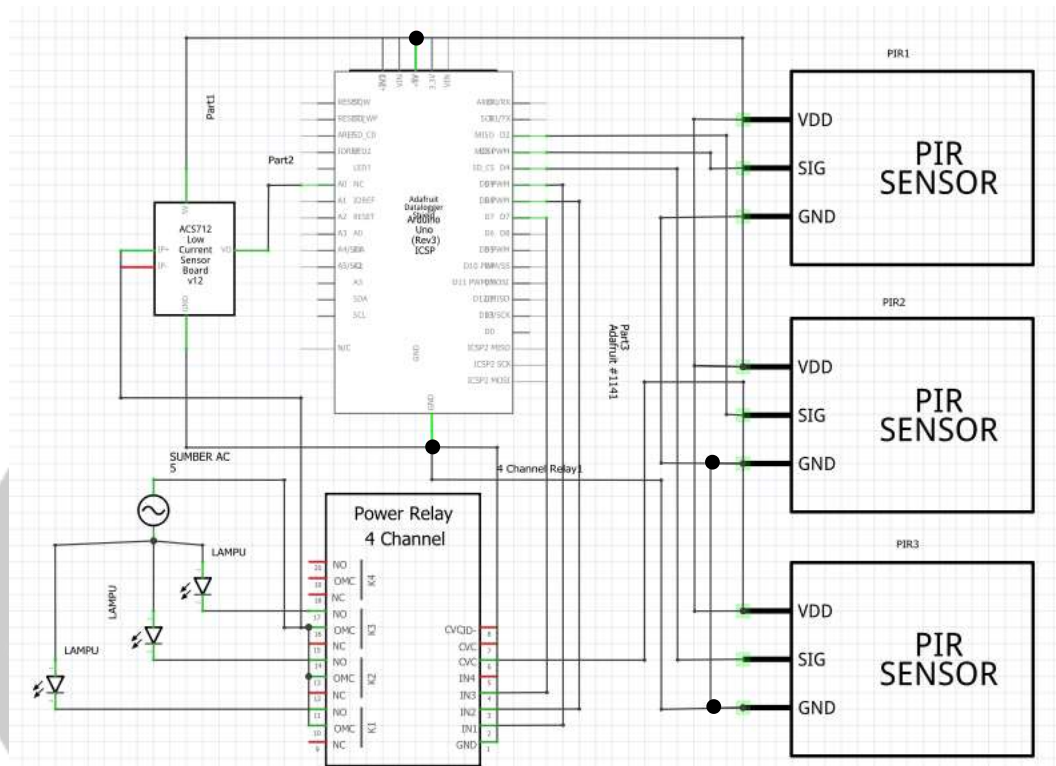
- Automatic induction: to enter the sensing range of the output is high, the person leaves the sensing range of the automatic delay off high, output low.
- Photosensitive control (optional, not factory-set) can be set photosensitive control, day or light intensity without induction.
- Temperature compensation (optional, factory reset): In the summer when the ambient temperature rises to 30 ° C to 32 ° C, the detection distance is slightly shorter, temperature compensation can be used for performance compensation.
- Triggered in two ways: (jumper selectable)
 - non-repeatable trigger: the sensor output high, the delay time is over, the output is automatically changed from high level to low level.
 - repeatable trigger: the sensor output high, the delay period, if there is human activity in its sensing range, the output will always remain high until the people left after the delay will be high level goes low (sensor module detects a time delay period will be automatically extended every human activity, and the starting point for the delay time to the last event of the time).
- With induction blocking time (the default setting: 2.5s blocked time): sensor module after each sensor output (high into low), followed by a blockade set period of time, during this time period sensor does not accept any sensor signal. This feature can be achieved sensor output time "and" blocking time "interval between the work can be applied to interval detection products; This function can inhibit a variety of interference in the process of load switching. (This time can be set at zero seconds – a few tens of seconds).
- Wide operating voltage range: default voltage DC4.5V-20V.
- Micropower consumption: static current <50 microamps, particularly suitable for battery-powered automatic control products.
- Output high signal: easy to achieve docking with the various types of circuit.

Adjustment:

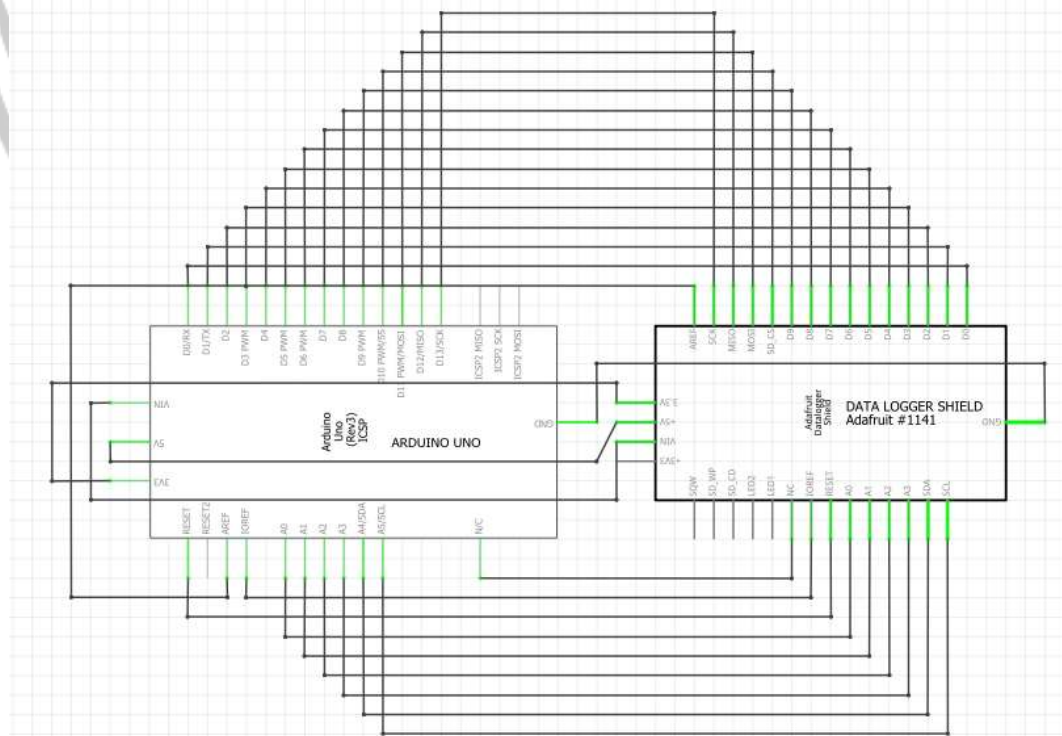
- Adjust the distance potentiometer clockwise rotation, increased sensing distance (about 7 meters), on the contrary, the sensing distance decreases (about 3 meters).
- Adjust the delay potentiometer clockwise rotation sensor the delay lengthened (300S), on the contrary, shorten the induction delay (5S).

Lampiran 5. Skematik Alat yang Dirancang

Skematik secara keseluruhan



Skematik Arduino dan Shield Data Logger



Lampiran 6. Program alternatif 1 (monitoring pada serial monitor arduino)

```
#include <SD.h>          // Memasukkan library untuk module SD card
#include <SPI.h>
#include <Wire.h>
#include "RTClib.h"      // Memasukkan library untuk Real Time Clock

File myFile; // variabel untuk file data logger
int pinCS = 10; // Pin 10 on Arduino Uno
RTC_DS1307 rtc; // Membangkitkan variabel Real Time Clock
char namaHari[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat", "Sabtu"};

int Lampu1 = 5; // Lampu 1 pada pin 5
int Lampu2 = 7; // Lampu 2 pada pin 6
int Lampu3 = 8; // Lampu 3 pada pin 7
int PinSensor1 = 2; // Sensor PIR-1 pada pin 2
int PinSensor2 = 3; // Sensor PIR-2 pada pin 3
int PinSensor3 = 4; // Sensor PIR-3 pada pin 4
int NilaiPin1; // Variabel untuk membaca nilai sinyal sensor PIR-1
int NilaiPin2; // Variabel untuk membaca nilai sinyal sensor PIR-2
int NilaiPin3; // Variabel untuk membaca nilai sinyal sensor PIR-3

//Pembangkitan variabel awal untuk pengukuran sensor arus ACS712
const int Sensor_Arus = A0; // Inisialisasi pin A0 untuk sensor arus
int sensitivitas = 66; // Nilai sensitivitas 66mV untuk modul sensor arus 30A
float NilaiAdc= 0; // Variabel nilai pembacaan sinyal analog sensor arus
float Arus = 0; // Variabel untuk nilai arus
double ArusRMS = 0; // Variabel untuk nilai arus efektif

void setup() {
  Serial.begin(9600); // Inisialisasi awal komunikasi serial 9600 baud rate
  pinMode(pinCS, OUTPUT);
  pinMode(Lampu1, OUTPUT); // Deklarasi Lampu 1 sebagai Output
  pinMode(Lampu2, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Lampu3, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Sensor_Arus, INPUT);
  pinMode(PinSensor1, INPUT); // Deklarasi pin PIR-1 sebagai Input
  pinMode(PinSensor2, INPUT); // Deklarasi pin PIR-2 sebagai Input
  pinMode(PinSensor3, INPUT); // Deklarasi pin PIR-3 sebagai Input
  digitalWrite(Lampu1, HIGH); // Inisialisasi kondisi awal relay Lampu 1 yaitu mati)
  digitalWrite(Lampu2, HIGH); // Inisialisasi kondisi awal relay Lampu 2 yaitu mati)
  digitalWrite(Lampu3, HIGH); // Inisialisasi kondisi awal relay Lampu 3 yaitu mati)
```

```

pins_init();
// Untuk kondisi Data Logger
if (SD.begin()      // Kondisi ketika SD Card dapat diakses
    {
        Serial.println("SD card is ready to use.");
        Serial.println("");
    }
else                // Kondisi ketika SD Card tidak dapat diakses
    {
        Serial.println("SD card initialization failed");
        Serial.println("");
        return;
    }
// untuk kondisi RTC
if (! rtc.begin())  // Kondisi Ketika RTC tidak dapat dibaca
    {
        Serial.println("RTC Tidak Bisa Dibaca");
        while (1);
    }

if (! rtc.isrunning()) // Kondisi Ketika RTC tidak dapat dijalankan
    {
        Serial.println("RTC Tudak dapat Dijalankan");
        rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); // Meng-update rtc dari waktu komputer
    }
}

void loop()
{
    NilaiAdc = getMaxValue(); // NilaiAdc mengambil nilai maks pembacaan sensor
    Arus= abs((float)(NilaiAdc-513)/1024*5/sensitivitas*1000000); // mencari nilai arus
    ArusRMS = Arus/( sqrt(2) ); // mencari nilai arus efektif
    double Power = ArusRMS * 216 /1000; // Menghitung Daya

    myFile = SD.open("Data.txt", FILE_WRITE); // Membuka dan membuat file Data.txt pada SD Card
    if (myFile) // Kondisi ketika myFile terbuka
    {
        DateTime now = rtc.now();
        //Menampilkan hasil pengukuran dan perhitungan serta RTC pada serial monitor
        Serial.print(namaHari[now.dayOfTheWeek()]); // menampilkan hari pada serial monitor
        Serial.print(','); // Menampilkan pemisah "-" pada serial monitor
        Serial.print(now.day()); // Menampilkan tanggal pada serial monitor
    }
}

```

```

Serial.print('/'); // Menampilkan pemisah "/" pada serial monitor
Serial.print(now.month()); // Menampilkan bulan pada serial monitor
Serial.print('/'); // Menampilkan pemisah "/" pada serial monitor
Serial.print(now.year()); // Menampilkan tahun pada serial monitor
Serial.print(" "); // Menampilkan pemisah " " pada serial monitor
Serial.print(now.hour()); // Menampilkan jam pada serial monitor
Serial.print(':'); // Menampilkan pemisah ":" pada serial monitor
Serial.print(now.minute()); // Menampilkan menit pada serial monitor
Serial.print(':'); // Menampilkan pemisah ":" pada serial monitor
Serial.print(now.second()); // Menampilkan detik pada serial monitor
Serial.println(); // Membuat proses enter pada serial monitor
Serial.print("Arus = "); // Menampilkan tulisan "Arus = " pada serial monitor
Serial.print(ArusRMS,3); // Menampilkan nilai dari variabel ArusRMS dengan 3 digit belakang
koma
Serial.print(" mA"); // Menampilkan tulisan " mA" pada serial monitor
Serial.print("\t\t Daya = "); // Menampilkan 2 kali tab dan tulisan "Daya = " pada serial monitor
Serial.print(Power,3); // Menampilkan nilai dari variabel Power dengan 3 digit belakang koma
Serial.println(" W"); // Menampilkan tulisan " W" pada serial monitor
Serial.println(""); // Melakukan proses enter pada serial monitor

//Menyimpan hasil pengukuran dan perhitungan serta RTC ke SD Card
myFile.print(namaHari[now.dayOfTheWeek()]); // menampilkan hari dan disimpan ke SD Card
myFile.print('-'); // Menampilkan pemisah "-" dan disimpan ke SD Card
myFile.print(now.day()); // Menampilkan tanggal dan disimpan ke SD Card
myFile.print('/'); // Menampilkan pemisah "/" dan disimpan ke SD Card
myFile.print(now.month()); // Menampilkan bulan dan disimpan ke SD Card
myFile.print('/'); // Menampilkan pemisah "/" dan disimpan ke SD Card
myFile.print(now.year()); // Menampilkan tahun dan disimpan ke SD Card
myFile.print(" "); // Menampilkan pemisah " " dan disimpan ke SD Card
myFile.print(now.hour()); // Menampilkan jam dan disimpan ke SD Card
myFile.print(':'); // Menampilkan pemisah ":" dan disimpan ke SD Card
myFile.print(now.minute()); // Menampilkan menit dan disimpan ke SD Card
myFile.print(':'); // Menampilkan pemisah ":" dan disimpan ke SD Card
myFile.print(now.second()); // Menampilkan detik dan disimpan ke SD Card
myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
myFile.print(ArusRMS,3); // Menampilkan nilai variabel ArusRMS dan disimpan ke SD Card
myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
myFile.println(Power,3); // Menampilkan nilai variabel Power dan disimpan ke SD Card
myFile.println(""); // Membuat enter
myFile.close(); // Menutup myFile
}
else // Kondisi ketika myFile tidak terbuka
{

```



```

Serial.println("Eror dalam membuka Data.txt"); // Menampilkan tulisan "Eror dalam membuka
Data.txt"
}

NilaiPin1 = digitalRead(PinSensor1); //pembacaan sensor PIR-1
NilaiPin2 = digitalRead(PinSensor2); //pembacaan sensor PIR-2
NilaiPin3 = digitalRead(PinSensor3); //pembacaan sensor PIR-3

digitalWrite(Lampu1, !NilaiPin1); // Nyala/mati Lampu1 tergantung dari pembacaan NilaiPin1
digitalWrite(Lampu2, !NilaiPin2); // Nyala/mati Lampu2 tergantung dari pembacaan NilaiPin2
digitalWrite(Lampu3, !NilaiPin3); // Nyala/mati Lampu3 tergantung dari pembacaan NilaiPin3
}

void pins_init()
{
    pinMode(Sensor_Arus, INPUT);
}

int getMaxValue()
{
    int sensorValue;           // Nilai Pembacaan dari sensor
    double sensorMax = 0;
    uint32_t start_time = millis();
    while((millis()-start_time) < 1000) // Pengambilan sampel selama 1 detik
    {
        sensorValue = analogRead(Sensor_Arus);
        if (sensorValue > sensorMax)
        {
            sensorMax = sensorValue; // Menyimpan nilai maksimum dari hasil pembacaan
        }
    }
    return sensorMax;
}
}

```

Lampiran 7. Program Alternatif 2 (monitoring pada excel dengan macro PLX-DAQ)

```
#include <SD.h>           // Memasukkan library untuk module SD card
#include <SPI.h>
#include <Wire.h>
#include "RTClib.h"       // Memasukkan library untuk Real Time Clock

File myFile; // variabel untuk file data logger
int pinCS = 10; // Pin 10 on Arduino Uno
RTC_DS1307 rtc; // Membangkitkan variabel Real Time Clock
char namaHari[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat", "Sabtu"};

int Lampu1 = 5; // Lampu 1 pada pin 5
int Lampu2 = 7; // Lampu 2 pada pin 6
int Lampu3 = 8; // Lampu 3 pada pin 7
int PinSensor1 = 2; // Sensor PIR-1 pada pin 2
int PinSensor2 = 3; // Sensor PIR-2 pada pin 3
int PinSensor3 = 4; // Sensor PIR-3 pada pin 4
int NilaiPin1; // Variabel untuk membaca nilai sinyal sensor PIR-1
int NilaiPin2; // Variabel untuk membaca nilai sinyal sensor PIR-2
int NilaiPin3; // Variabel untuk membaca nilai sinyal sensor PIR-3

//Pembangkitan variabel awal untuk pengukuran sensor arus ACS712
const int Sensor_Arus = A0; // Inisialisasi pin A0 untuk sensor arus
int sensitivitas = 66; // Nilai sensitivitas 66mV untuk modul sensor arus 30A
float NilaiAdc = 0; // Variabel nilai pembacaan sinyal analog sensor arus
float Arus = 0; // Variabel untuk nilai arus
double ArusRMS = 0; // Variabel untuk nilai arus efektif

void setup() {
  Serial.begin(9600); // Inisialisasi awal komunikasi serial 9600 baud rate
  pinMode(pinCS, OUTPUT);
  pinMode(Lampu1, OUTPUT); // Deklarasi Lampu 1 sebagai Output
  pinMode(Lampu2, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Lampu3, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Sensor_Arus, INPUT);
  pinMode(PinSensor1, INPUT); // Deklarasi pin PIR-1 sebagai Input
  pinMode(PinSensor2, INPUT); // Deklarasi pin PIR-2 sebagai Input
  pinMode(PinSensor3, INPUT); // Deklarasi pin PIR-3 sebagai Input
  digitalWrite(Lampu1, HIGH); // Inisialisasi kondisi awal relay Lampu 1 yaitu mati
  digitalWrite(Lampu2, HIGH); // Inisialisasi kondisi awal relay Lampu 2 yaitu mati
  digitalWrite(Lampu3, HIGH); // Inisialisasi kondisi awal relay Lampu 3 yaitu mati
}
```

```

pins_init();
// Untuk kondisi Data Logger
if (SD.begin()) // Kondisi ketika SD Card dapat diakses
{
    Serial.println("SD card is ready to use.");
    Serial.println("");
}
else // Kondisi ketika SD Card tidak dapat diakses
{
    Serial.println("SD card initialization failed");
    Serial.println("");
    return;
}
// untuk kondisi RTC
if (! rtc.begin()) // Kondisi Ketika RTC tidak dapat dibaca
{
    Serial.println("RTC Tidak Bisa Dibaca");
    while (1);
}

if (! rtc.isrunning()) // Kondisi Ketika RTC tidak dapat dijalankan
{
    Serial.println("RTC Tudak dapat Dijalankan");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); // Meng-update rtc dari waktu komputer
}
}

void loop()
{
    NilaiAdc = getMaxValue(); // NilaiAdc mengambil nilai maks pembacaan sensor
    Arus= abs((float)(NilaiAdc-513)/1024*5/sensitivitas*1000000); // mencari nilai arus
    ArusRMS = Arus/( sqrt(2) ); // mencari nilai arus efektif
    double Power = ArusRMS * 216 /1000; // Menghitung Daya

    Serial.print("DATA,TIME, "); // Mengirim dan menampilkan data dan waktu real time ke Excel
    Serial.print (ArusRMS); // Mengirim dan menampilkan nilai arus efektif ke Excel
    Serial.print(","); // Pemisahan dengan simbol koma ","
    Serial.println (Power); // Mengirim dan menampilkan nilai daya ke Excel

    myFile = SD.open("Data.txt", FILE_WRITE); // Membuka dan membuat file Data.txt pada SD Card
    if (myFile) // Kondisi ketika myFile terbuka

```

```

{
  DateTime now = rtc.now();
  //Menyimpan hasil pengukuran dan perhitungan serta RTC ke SD Card
  myFile.print(namaHari[now.dayOfTheWeek()]); // menampilkan hari dan disimpan ke SD Card
  myFile.print("-"); // Menampilkan pemisah "-" dan disimpan ke SD Card
  myFile.print(now.day()); // Menampilkan tanggal dan disimpan ke SD Card
  myFile.print("/"); // Menampilkan pemisah "/" dan disimpan ke SD Card
  myFile.print(now.month()); // Menampilkan bulan dan disimpan ke SD Card
  myFile.print("/"); // Menampilkan pemisah "/" dan disimpan ke SD Card
  myFile.print(now.year()); // Menampilkan tahun dan disimpan ke SD Card
  myFile.print(" "); // Menampilkan pemisah " " dan disimpan ke SD Card
  myFile.print(now.hour()); // Menampilkan jam dan disimpan ke SD Card
  myFile.print(":"); // Menampilkan pemisah ":" dan disimpan ke SD Card
  myFile.print(now.minute()); // Menampilkan menit dan disimpan ke SD Card
  myFile.print(":"); // Menampilkan pemisah ":" dan disimpan ke SD Card
  myFile.print(now.second()); // Menampilkan detik dan disimpan ke SD Card
  myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
  myFile.print(ArusRMS,3); // Menampilkan nilai variabel ArusRMS dan disimpan ke SD Card
  myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
  myFile.println(Power,3); // Menampilkan nilai variabel Power dan disimpan ke SD Card
  myFile.println(""); // Membuat enter
  myFile.close(); // Menutup myFile
}
else // Kondisi ketika myFile tidak terbuka
{
  Serial.println("Error dalam membuka Data.txt"); // Menampilkan tulisan "Error dalam membuka
Data.txt"
}

NilaiPin1 = digitalRead(PinSensor1); //pembacaan sensor PIR-1
NilaiPin2 = digitalRead(PinSensor2); //pembacaan sensor PIR-2
NilaiPin3 = digitalRead(PinSensor3); //pembacaan sensor PIR-3

digitalWrite(Lampu1, !NilaiPin1); // Nyala/mati Lampu1 tergantung dari pembacaan NilaiPin1
digitalWrite(Lampu2, !NilaiPin2); // Nyala/mati Lampu2 tergantung dari pembacaan NilaiPin2
digitalWrite(Lampu3, !NilaiPin3); // Nyala/mati Lampu3 tergantung dari pembacaan NilaiPin3
}

void pins_init()
{
  pinMode(Sensor_Arus, INPUT);
}

```

```
int getMaxValue()
{
    int sensorValue;          // Nilai Pembacaan dari sensor
    double sensorMax = 0;
    uint32_t start_time = millis();
    while((millis()-start_time) < 1000) // Pengambilan sampel selama 1 detik
    {
        sensorValue = analogRead(Sensor_Arus);
        if (sensorValue > sensorMax)
        {
            sensorMax = sensorValue; // Menyimpan nilai maksimum dari hasil pembacaan
        }
    }
    return sensorMax;
}
```



Lampiran 8. Program Alternatif 3 (monitoring menggunakan TelemetryViewer_v0.4.jar)

```
#include <SD.h>           // Memasukkan library untuk module SD card
#include <SPI.h>
#include <Wire.h>
#include "RTClib.h"       // Memasukkan library untuk Real Time Clock

File myFile; // variabel untuk file data logger
int pinCS = 10; // Pin 10 on Arduino Uno
RTC_DS1307 rtc; // Membangkitkan variabel Real Time Clock
char namaHari[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat", "Sabtu"};

int Lampu1 = 5; // Lampu 1 pada pin 5
int Lampu2 = 7; // Lampu 2 pada pin 6
int Lampu3 = 8; // Lampu 3 pada pin 7
int PinSensor1 = 2; // Sensor PIR-1 pada pin 2
int PinSensor2 = 3; // Sensor PIR-2 pada pin 3
int PinSensor3 = 4; // Sensor PIR-3 pada pin 4
int NilaiPin1; // Variabel untuk membaca nilai sinyal sensor PIR-1
int NilaiPin2; // Variabel untuk membaca nilai sinyal sensor PIR-2
int NilaiPin3; // Variabel untuk membaca nilai sinyal sensor PIR-3

//Pembangkitan variabel awal untuk pengukuran sensor arus ACS712
const int Sensor_Arus = A0; // Inisialisasi pin A0 untuk sensor arus
int sensitivitas = 66; // Nilai sensitivitas 66mV untuk modul sensor arus 30A
float NilaiAdc = 0; // Variabel nilai pembacaan sinyal analog sensor arus
float Arus = 0; // Variabel untuk nilai arus
double ArusRMS = 0; // Variabel untuk nilai arus efektif

void setup() {
  Serial.begin(9600); // Inisialisasi awal komunikasi serial 9600 baud rate
  pinMode(pinCS, OUTPUT);
  pinMode(Lampu1, OUTPUT); // Deklarasi Lampu 1 sebagai Output
  pinMode(Lampu2, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Lampu3, OUTPUT); // Deklarasi Lampu 2 sebagai Output
  pinMode(Sensor_Arus, INPUT);
  pinMode(PinSensor1, INPUT); // Deklarasi pin PIR-1 sebagai Input
  pinMode(PinSensor2, INPUT); // Deklarasi pin PIR-2 sebagai Input
  pinMode(PinSensor3, INPUT); // Deklarasi pin PIR-3 sebagai Input
  digitalWrite(Lampu1, HIGH); // Inisialisasi kondisi awal relay Lampu 1 yaitu mati
  digitalWrite(Lampu2, HIGH); // Inisialisasi kondisi awal relay Lampu 2 yaitu mati
  digitalWrite(Lampu3, HIGH); // Inisialisasi kondisi awal relay Lampu 3 yaitu mati
}
```

```

pins_init();
// Untuk kondisi Data Logger
if (SD.begin())      // Kondisi ketika SD Card dapat diakses
{
  Serial.println("SD card is ready to use.");
  Serial.println("");
}
else                // Kondisi ketika SD Card tidak dapat diakses
{
  Serial.println("SD card initialization failed");
  Serial.println("");
  return;
}
// untuk kondisi RTC
if (! rtc.begin())  // Kondisi Ketika RTC tidak dapat dibaca
{
  Serial.println("RTC Tidak Bisa Dibaca");
  while (1);
}

if (! rtc.isrunning()) // Kondisi Ketika RTC tidak dapat dijalankan
{
  Serial.println("RTC Tudak dapat Dijalankan");
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); // Meng-update rtc dari waktu komputer
}
}

void loop()
{
  NilaiAdc = getMaxValue(); // NilaiAdc mengambil nilai maks pembacaan sensor
  Arus= abs((float)(NilaiAdc-513)/1024*5/sensitivitas*1000000); // mencari nilai arus
  ArusRMS = Arus/( sqrt(2) ); // mencari nilai arus efektif
  double Power = ArusRMS * 216 /1000; // Menghitung Daya

  Serial.print (ArusRMS); // Mengirim dan menampilkan nilai arus efektif kolom 0 software
  Serial.print(","); // Pemisahan dengan simbol koma "," untuk memisahkan kolom
  Serial.println (Power); // Mengirim dan menampilkan nilai daya ke kolom 1 software

  myFile = SD.open("Data.txt", FILE_WRITE); // Membuka dan membuat file Data.txt pada SD Card
  if (myFile) // Kondisi ketika myFile terbuka

```

```

{
  DateTime now = rtc.now();
  //Menyimpan hasil pengukuran dan perhitungan serta RTC ke SD Card
  myFile.print(namaHari[now.dayOfTheWeek()]); // menampilkan hari dan disimpan ke SD Card
  myFile.print("-"); // Menampilkan pemisah "-" dan disimpan ke SD Card
  myFile.print(now.day()); // Menampilkan tanggal dan disimpan ke SD Card
  myFile.print("/"); // Menampilkan pemisah "/" dan disimpan ke SD Card
  myFile.print(now.month()); // Menampilkan bulan dan disimpan ke SD Card
  myFile.print("/"); // Menampilkan pemisah "/" dan disimpan ke SD Card
  myFile.print(now.year()); // Menampilkan tahun dan disimpan ke SD Card
  myFile.print(" "); // Menampilkan pemisah " " dan disimpan ke SD Card
  myFile.print(now.hour()); // Menampilkan jam dan disimpan ke SD Card
  myFile.print(":"); // Menampilkan pemisah ":" dan disimpan ke SD Card
  myFile.print(now.minute()); // Menampilkan menit dan disimpan ke SD Card
  myFile.print(":"); // Menampilkan pemisah ":" dan disimpan ke SD Card
  myFile.print(now.second()); // Menampilkan detik dan disimpan ke SD Card
  myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
  myFile.print(ArusRMS,3); // Menampilkan nilai variabel ArusRMS dan disimpan ke SD Card
  myFile.print(","); // Menampilkan pemisah "," dan disimpan ke SD Card
  myFile.println(Power,3); // Menampilkan nilai variabel Power dan disimpan ke SD Card
  myFile.println(""); // Membuat enter
  myFile.close(); // Menutup myFile
}
else // Kondisi ketika myFile tidak terbuka
{
  Serial.println("Error dalam membuka Data.txt"); // Menampilkan tulisan "Error dalam membuka
  Data.txt"
}

NilaiPin1 = digitalRead(PinSensor1); //pembacaan sensor PIR-1
NilaiPin2 = digitalRead(PinSensor2); //pembacaan sensor PIR-2
NilaiPin3 = digitalRead(PinSensor3); //pembacaan sensor PIR-3

digitalWrite(Lampu1, !NilaiPin1); // Nyala/mati Lampu1 tergantung dari pembacaan NilaiPin1
digitalWrite(Lampu2, !NilaiPin2); // Nyala/mati Lampu2 tergantung dari pembacaan NilaiPin2
digitalWrite(Lampu3, !NilaiPin3); // Nyala/mati Lampu3 tergantung dari pembacaan NilaiPin3
}

void pins_init()
{
  pinMode(Sensor_Arus, INPUT);
}

```



```
int getMaxValue()
{
    int sensorValue;          // Nilai Pembacaan dari sensor
    double sensorMax = 0;
    uint32_t start_time = millis();
    while((millis()-start_time) < 1000) // Pengambilan sampel selama 1 detik
    {
        sensorValue = analogRead(Sensor_Arus);
        if (sensorValue > sensorMax)
        {
            sensorMax = sensorValue; // Menyimpan nilai maksimum dari hasil pembacaan
        }
    }
    return sensorMax;
}
```

