

BAB II

TINJAUAN PUSTAKA

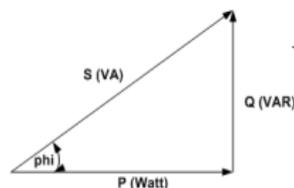
2.1 Daya Listrik²

Daya ialah banyaknya perubahan energi terhadap waktu dalam besaran tegangan dan arus. Daya listrik dapat dibagi menjadi 3 yaitu daya nyata (P), daya reaktif (Q), dan daya semu (S).

- Daya nyata P merupakan daya sebenarnya yang dibutuhkan oleh beban-beban listrik/peralatan rumah tangga. Satuan daya nyata adalah watt (W).
- Daya reaktif Q adalah daya yang timbul karena adanya pembentukan medan magnet pada beban-beban induktif. Satuan dari daya reaktif adalah volt ampere reaktif (VAR).
- Daya semu merupakan resultan antara daya nyata dan daya reaktif. Satuan dari daya semu adalah volt ampere (VA).
- Faktor daya ($\cos \phi$) merupakan suatu konstanta pengali dengan nilai 0 sampai 1, yang menunjukkan seberapa besar daya nyata yang diserap oleh beban resistif dari daya semu yang ada pada suatu beban total.

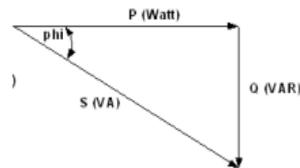
2.2 Segitiga Daya²

Daya semu (S) merupakan resultan dari dua komponen, yaitu daya nyata (P) dan komponen daya reaktif (Q). Hubungan ini disebut dengan segitiga daya dan dalam bentuk vektor dapat digambarkan beserta rumus sebagai berikut:



Gambar 2.1 Segitiga daya beban listrik bersifat kapasitif ²

³Cekmas Cekdin dan dan Taufik Barlian. (2013) *Rangkaian Listrik*. Yogyakarta: Andi Yoyakarta, hlm. 92-93



Gambar 2.2 Segitiga daya beban listrik bersifat induktif²

$$P = V \cdot I \cos \varphi \dots\dots\dots (2.1)$$

$$Q = V \cdot I \sin \varphi \dots\dots\dots (2.2)$$

$$S = \sqrt{P^2 + Q^2} \text{ atau } S = V \cdot I \dots\dots\dots (2.3)$$

$$\text{Faktor Daya} = \frac{P}{S} = \cos \varphi \dots\dots\dots (2.4)$$

Dimana:

P = Daya nyata (Watt)

Q = Daya reaktif (VAR)

S = Daya semu (VA)

φ = Beda sudut fasa (Rad/Deg)

V = Tegangan (Volt)

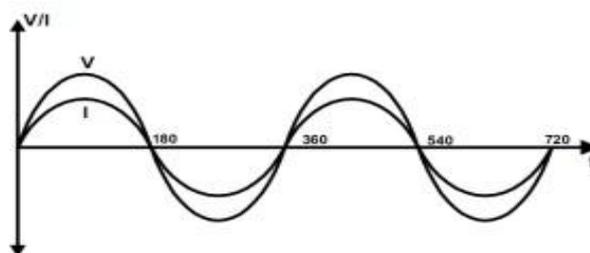
I = Arus (Amper)

2.3 Hubungan Fasa²

Ada tiga kemungkinan hubungan fasa antara arus dan tegangan dalam satuan rangkaian, yaitu arus dan tegangan sefasa seperti pada gambar 2.3.

2.3.1 Elemen Tahanan Murni (R)

Pada elemen tahanan murni R untuk arus dan tegangan dengan fungsi sinusoidal, sudut fasanya adalah sefasa. Plot untuk arus dan tegangan pada tahanan murni R adalah seperti Gambar 2.3.

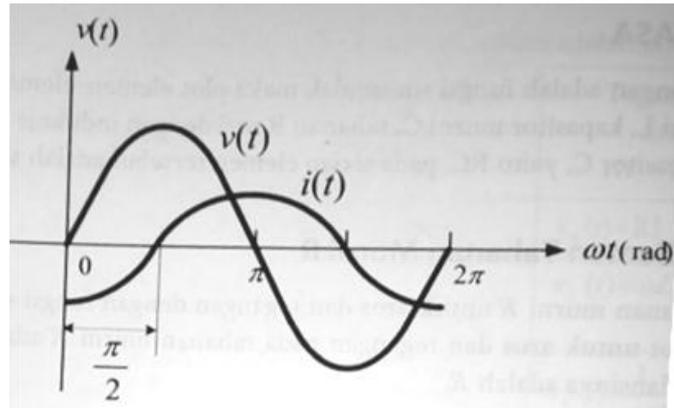


Gambar 2.3 Arus dan tegangan sefasa²

³Cekmas Cekdin dan dan Taufik Barlian. (2013) *Rangkaian Listrik*. Yogyakarta: Andi Yoyakarta, hlm. 92-93

2.3.2 Elemen Induktor Murni (L)

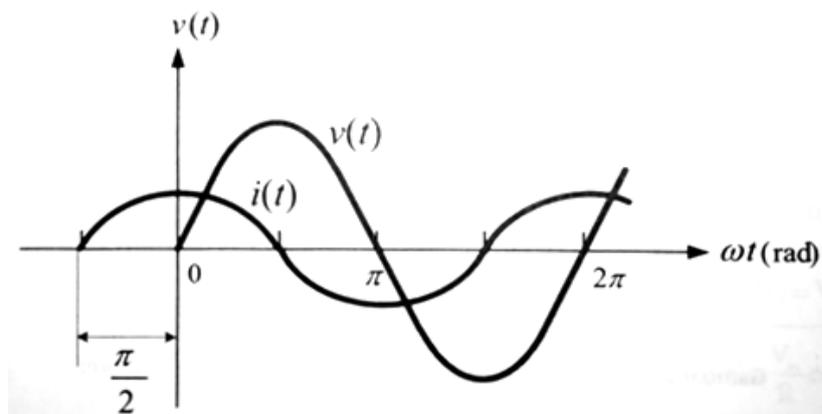
Pada elemen induktor murni L untuk arus dan tegangan dengan fungsi sinusoidal, arus ketinggalan terhadap tegangan sebesar sudut fasa 90° atau $\pi/2$ (lagging). Plot untuk arus dan tegangan pada induktor murni L adalah seperti Gambar 2.4. Magnitude impedansinya adalah ωL .



Gambar 2.4 Arus ketinggalan terhadap tegangan sebesar sudut fasa 90° atau $\pi/2$ (lagging)²

2.3.3 Elemen Kapasitor Murni (C)

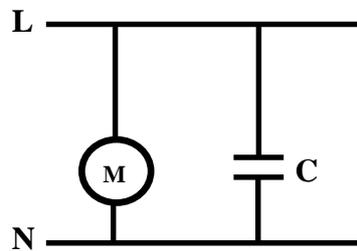
Pada elemen kapasitor murni C untuk arus dan tegangan dengan fungsi sinusoidal, arus mendahului terhadap tegangan sebesar sudut fasa 90° atau $\pi/2$ (leading). Plot untuk arus dan tegangan pada kapasitor murni C adalah seperti Gambar 2.5. Magnitude impedansinya adalah $1/\omega L$.



Gambar 2.5 Arus mendahului terhadap tegangan sebesar sudut fasa 90° atau $\pi/2$ (leading)²

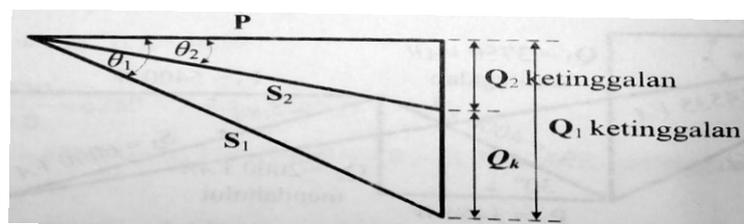
2.4 Perbaikan Faktor Daya²

Perbaikan faktor daya atau *power factor* ($p.f$) atau $\cos \theta$ dalam sistem tenaga listrik pada suatu industri adalah untuk meningkatkan faktor daya dari sistem tenaga listrik yang sebelumnya memiliki kualitas rendah. Beban-beban motor pada industri adalah beban induktif. Untuk mengubah faktor daya yang rendah ke faktor daya yang lebih baik pada beban motor adalah dengan menambahkan kapasitor (Q_k) pada beban motor tersebut yang dipasang secara paralel. Gambar 2.6 adalah pemasangan kapasitor (Q_k) untuk perbaikan faktor daya dengan sebuah motor (M) yang dipasang secara paralel.



Gambar 2.6 Pemasangan kapasitor (Q_k) untuk perbaikan faktor daya dengan sebuah motor (M) yang dipasang secara paralel²

Besar harga kapasitor yang harus dipasang secara paralel dengan sebuah motor dapat ditentukan dengan, misalnya, daya aktif pada sebuah motor adalah P dalam *kilowatt* (kW) dengan faktor daya pada kondisi pertama adalah $p.f_1 = \cos \theta_1$ ketinggalan. Motor ini akan diperbaiki dengan faktor daya menjadi $p.f_2 = \cos \theta_2$ ketinggalan. Untuk menentukan besar harga kapasitor yang harus dipasang, perhatikan Gambar 2.7.



Gambar 2.7 Untuk menentukan besar harga kapasitor yang harus dipasang²

³Cekmas Cekdin dan dan Taufik Barlian. (2013) *Rangkaian Listrik*. Yogyakarta: Andi Yoyakarta, hlm. 92-93

Besar harga daya kompleks dan daya reaktif pada faktor daya kondisi pertama adalah:

$$S_1 = \frac{P}{p.f_1} = \frac{P}{\cos \theta_1} \dots\dots\dots (2.5)$$

$$Q_1 = S_1 \sin \theta_1 = \left(\frac{P}{\cos \theta_1} \right) \sin \theta_1 = P \tan \theta_1 \dots\dots\dots (2.6)$$

Besar harga daya kompleks dan daya reaktif pada faktor daya untuk perbaikan adalah:

$$S_2 = \frac{P}{p.f_2} = \frac{P}{\cos \theta_2} \dots\dots\dots (2.7)$$

Sehingga besar harga kapasitor yang harus dipasang adalah:

$$Q_k = Q_1 - Q_2 = P (\tan \theta_1 - \tan \theta_2) \dots\dots\dots (2.8)$$

$$C = \frac{Q_k}{\text{supplyVoltage}^2 \times 2 \times \pi \times \text{frequency}} \dots\dots\dots (2.9)$$

Dimana:

S_1	= Daya semu awal (sebelum dikompensasi) (VA)
P	= Daya aktif beban (Watt)
$\cos \varphi_1$	= Faktor daya awal (sebelum dikompensasi) (Rad/Deg)
Q_1	= Daya reaktif (sebelum dikompensasi) (VAR)
S_2	= Daya semu yang diinginkan (setelah dikompensasi) (VA)
$\cos \varphi_2$	= Faktor daya yang diharapkan (Rad/Deg)
Q_2	= Daya reaktif yang setelah dikompensasi (VAR)
Q_c	= Rating kapasitor (KVAR)
C	= Nilai kapasitor (uF)

2.5 Transformator Ukur⁵

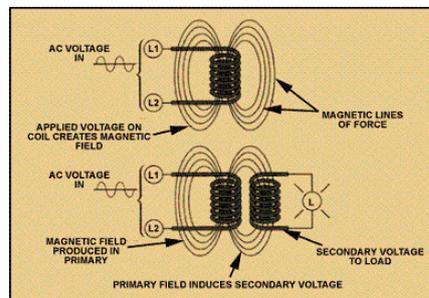
Transformator ukur didisain secara khusus untuk pengukuran dalam sistem daya. Transformator ini banyak digunakan dalam sistem daya karena mempunyai keuntungan, antara lain:

- Memberikan isolasi elektrik bagi sistem daya
- Tahan terhadap beban untuk berbagai tingkatan
- Secara fisik lebih sederhana bentuknya

⁵ PT PLN (PERSERO), 2014, *Buku Pedoman Pemeliharaan Transformator Tenaga*, Jakarta Selatan, PT. PLN, hlm.1.

2.5.1 Transformator ukur tegangan

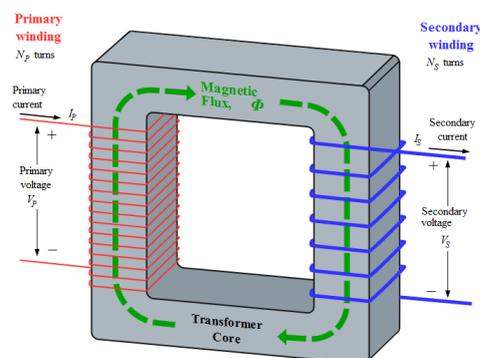
Trafo merupakan peralatan statis dimana rangkaian magnetik dan belitan yang terdiri dari 2 atau lebih belitan, secara induksi elektromagnetik, mentransformasikan daya (arus dan tegangan) sistem AC ke sistem arus dan tegangan lain pada frekuensi yang sama (IEC 60076 -1 tahun 2011). Trafo menggunakan prinsip elektromagnetik yaitu hukum ampere dan induksi *faraday*, dimana perubahan arus atau medan listrik dapat membangkitkan medan magnet dan perubahan medan magnet / fluks medan magnet dapat membangkitkan tegangan induksi.



Gambar 2.8 Prinsip hukum elektromagnetik

Arus AC yang mengalir pada belitan primer membangkitkan *flux* magnet yang mengalir melalui inti besi yang terdapat diantara dua belitan, flux magnet tersebut menginduksi belitan sekunder sehingga pada ujung belitan sekunder akan terdapat beda potensial / tegangan induksi (Gambar 2.8).

Belitan yang terdiri dari batang tembaga berisolasi yang mengelilingi inti besi, dimana saat arus bolak balik mengalir pada belitan tembaga tersebut, inti besi akan terinduksi dan menimbulkan flux magnetik.

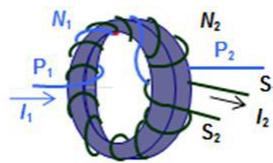


Gambar 2.9 Elektromagnetik pada trafo⁵

2.5.2 Transformator ukur arus⁵

Trafo Arus (Current Transformator - CT) yaitu peralatan yang digunakan untuk melakukan pengukuran besaran arus pada instalasi tenaga listrik disisi primer (TET, TT dan TM) yang berskala besar dengan melakukan transformasi dari besaran arus yang besar menjadi besaran arus yang kecil secara akurat dan teliti untuk keperluan pengukuran dan proteksi.

Prinsip kerja trafo arus adalah sebagai berikut:



Gambar 2.10 Rangkaian pada CT⁵

2.6 Kapasitor¹

Kapasitor atau kondensator adalah alat (komponen) yang dibuat sedemikian rupa sehingga mampu menyimpan muatan listrik yang besar untuk sementara waktu. Lambang kapasitor adalah dua garis yang parallel dan tegak lurus dengan sambungan kabel seperti yang terlihat pada gambar 2.11. Lambang ini menunjukkan bahwa kapasitor pada dasarnya dibentuk oleh dua plat logam yang terpisah oleh isolator.



Gambar 2.11 Simbol Kapasitor¹

2.6.1 Kapasitansi

Kapasitansi adalah ukuran jumlah muatan listrik yang disimpan (atau dipisahkan) untuk sebuah listrik potensial yang telah ditentukan. Bentuk paling umum dari penyimpanan muatan adalah sebuah kapasitor dua lempeng/ pelat/ keping. Ketika belum ada muatan pada plat logam, di antara plat logam belum terdapat medan listrik, maka belum juga terdapat tegangan di antara kedua plat logam tersebut. Karena belum terdapat tegangan antara kedua plat logam, jelas belum terdapat tegangan diantara kedua kaki-kaki kapasitor tersebut. Jika kapasitor

¹ Bayu, 2013, Perbaikan Faktor Daya, Palembang, Politeknik Negeri Sriwijaya, hlm. 1.

ini dihubungkan dengan suatu rangkaian, maka akan ada arus yang mengalir. Tetapi karena diantara plat logam dipasang sebuah isolator, arus tidak akan dapat melewati isolator tersebut, sehingga muatan yang bergerak (arus) ha/nya bisa bergerak sampai ke plat logam saja. Disanalah muatan berkumpul, berarti pada plat logam akan ada semakin banyak muatan.

Karena ada muatan pada plat logam, maka timbul medan listrik di antara plat logam, sehingga akan ada tegangan antara dua plat logam. Karena ada tegangan antara dua plat logam, berarti ada tegangan antara kaki sambungan kapasitor.

Jika muatan di lempeng/pelat/keping adalah $+Q$ dan $-Q$, dan V adalah tegangan listrik antar lempeng/pelat/keping, maka rumus kapasitansi adalah:

$$C = \frac{Q}{V} \dots\dots\dots 2.10$$

Dimana:

C = kapasitansi (Farad)

Q = muatan (Coulomb)

V = tegangan (Volt)

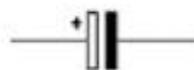
2.6.2 Jenis Kapasitor

a) Menurut polaritas

1. Kapasitor polar memiliki polaritas positif (+) dan negatif (-). Dalam pemasangannya harus diperhatikan polaritasnya dan tidak boleh dipasang terbalik. Pada bodinya terdapat tanda polaritasnya untuk menandai kaki yang berpolaritas positif (+) dan negatif (-).



Gambar 2.12 Kapasitor polar¹



Gambar 2.13 Simbol kapasitor polar¹

2. Kapasitor non-polar merupakan kapasitor yang pada kedua kutubnya tidak mempunyai polaritas artinya pada kutub-kutubnya dapat dipakai secara berbalik, dan biasanya kapasitor jenis ini mempunyai nilai kapasitansi yang kecil serta bahan dielektriknya terbuat dari kertas, keramik, mika, dll. Dapat dilihat pada gambar dibawah ini:



Gambar 2.14 Kapasitor non-polar¹



Gambar 2.15 Simbol kapasitor non-polar¹

b) Menurut bahan pembuat dielektrik

1. Kapasitor elektrostatik

Kapasitor elektrostatik adalah kelompok kapasitor yang dibuat dengan bahan dielektrik dari keramik, film, dan mika. Umumnya kapasitor kelompok ini adalah non-polar.

2. Kapasitor elektrolytik

Kapasitor elektrolytik adalah kelompok kapasitor yang dibuat dengan bahan dielektriknya dari metal-oksida. Umumnya kapasitor yang termasuk kelompok ini adalah kapasitor polar, dengan tanda positif dan negatif dibadannya. Kapasitor ini dapat memiliki polaritas adalah karena proses pembuatannya menggunakan elektrolisa sehingga terbentuk kutub positif anoda dan kutub negatif katoda.

3. Kapasitor electrochemical

Termasuk kapasitor jenis ini adalah baterai dan accu. Pada kenyataannya baterai dan accu adalah kapasitor yang sangat baik, karena memiliki kapasitansi yang sangat besar dan arus bocor (leakage current) yang sangat kecil.

4. Kapasitor Tabung

Kapasitor tabung atau silinder terdiri dari dua silinder konduktor berbeda jari-jari yang mengapit bahan dielektrik diantaranya.

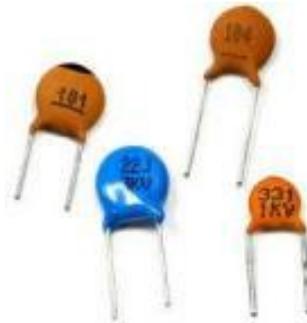
c) Menurut ketetapan nilai

1. Kapasitor tetap

Kapasitor tetap adalah suatu kapasitor yang nilainya konstan dan tidak berubah-ubah (nilai kapasitansya tetap tidak dapat diubah). Kapasitor tetap ada tiga macam:

a. Kapasitor keramik

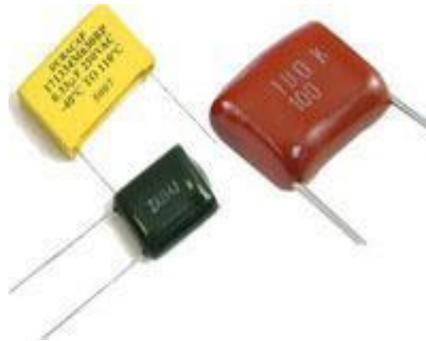
Kapasitor keramik ada yang berbentuk bulat tipis, ada yang persegi empat berwarna merah, hijau, coklat, dan lain-lain. Dalam pemasangannya dipapan rangkaian (PCB), boleh dibolak-balik karena tidak mempunyai polaritas. Mempunyai kapasitas mulai dari beberapa piko Farad sampai ratusan Kilopik Farad (KpF). Dengan tegangan kerja maksimal 25 volt sampai 1000 volt, tetapi ada juga yang sampai ribuan volt.



Gambar 2.16 Kapasitor keramik¹

b. Kapasitor polyester

Pada dasarnya sama dengan kapasitor keramik begitu juga cara menghitung nilainya. Bentuknya persegi empat seperti permen. Biasanya mempunyai warna merah, hijau, coklat, dan sebagainya.

Gambar 2.17 Kapasitor polyester¹

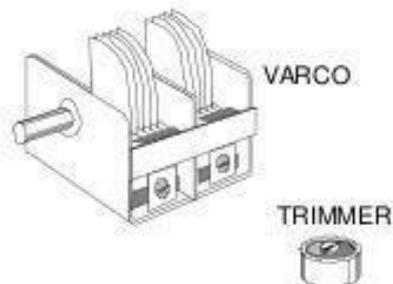
c. Kapasitor kertas

Kapasitor kertas ini sering disebut sebagai kapasitor padder. Misalnya pada radio, dipasang seri dari spul osilator ke variable kapasitor. Nilai kapasitas pada sirkuit oscillator antara lain:

Gambar 2.18 Kapasitor Kertas¹

2. Kapasitor tidak tetap (variable dan trimmer)

Kapasitor variable dan trimmer adalah jenis kapasitor yang kapasitasnya bisa diubah-ubah. Kapasitor ini dapat berubah kapasitasnya karena secara fisik mempunyai poros yang dapat diputar dengan menggunakan obeng.

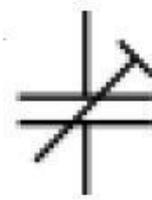
Gambar 2.19 Kapasitor variable¹

Kapasitor variabel (Varco) terbuat dari logam, mempunyai kapasitas maksimum sekitar 100 pF (pikoFarad) sampai 500 pF ($100\text{pF} = 0.0001\mu\text{F}$). Kapasitor variabel dengan spul antena dan spul osilator berfungsi sebagai pemilih gelombang frekuensi tertentu yang akan ditangkap.



Gambar 2.20 Lambang simbol Kapasitor Variable¹

Sedangkan kapasitor trimer dipasang paralel dengan variabel kapasitor berfungsi untuk menepatkan pemilihan gelombang frekuensi tersebut, Kapasitor trimer mempunyai kapasitas dibawah 100 pF (pikoFarad).



Gambar 2.21 Lambang simbol Kapasitor Trimmer¹

2.6.3 Pengkodean Nilai Kapasitansi Kapasitor¹

Kapasitansi dari kapasitor memiliki satuan Farad. Rentang kapasitansi kapasitor dari 0,015 pico farad sampai beberapa ribu Farad. Besarnya nilai kapasitansi biasanya diberikan pada kemasan kapasitor. Ada beberapa metode penulisan nilai kapasitansi dari kapasitor.

Penulisan nilai sesungguhnya pada kemasan kapasitor. Untuk kapasitor dengan kemasan besar biasanya nilai kapasitansi dituliskan nilainya dengan satuannya contoh 1 uF, 10 uF dan seterusnya. Juga biasanya dituliskan juga tegangan kerja maksimum dari kapasitor. Untuk kapasitor polar juga akan diberikan tanda elektroda mana yang negatif atau positif (biasanya hanya salah satu saja tandanya negatif; contoh elco atau positif; contoh tantalum)

¹Bayu, 2013, Perbaikan Faktor Daya, Palembang, Politeknik Negeri Sriwijaya, hlm. 1.

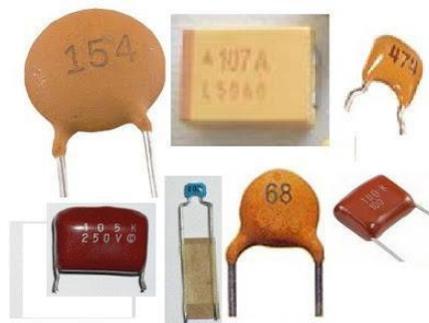


Gambar 2.22 Contoh nilai kapasitansi yang langsung ditulis di kemasan kapasitor¹

a) Penulisan dengan kode angka dua dan tiga digit

Kapasitor yang memiliki kemasan kecil biasanya nilai kapasitansinya dituliskan dengan kode tertentu. Salah satu sistem pengkodean tersebut adalah dengan kode angka dua digit atau tiga digit. Pada sistem kode dua digit, maka angka dua digit tersebut merupakan nilai kapasitansi dari kapasitor tersebut dengan satuan pico Farad. Contoh jika pada kapasitor tertulis 47 maka nilai kapasitansi dari kapasitor tersebut adalah 47 pico farad.

Jika pada kemasan kapasitor tertulis kode tiga angka, maka dua digit pertama adalah nilai awal dari kapasitor sedangkan digit ketiga adalah pengali dari kelipatan pangkat 10 dan dengan satuan yang sama yaitu pico Farad. Contoh jika pada kemasan kapasitor tertulis 332 berarti nilai kapasitansi kapasitor tersebut adalah $33 \times 10^2 = 3300 \text{ pF} = 3,3 \text{ nF}$



Gambar 2.23 Contoh kapasitor dengan penulisan kapasitansi menggunakan kode dua digit dan tiga digit¹

b) Penulisan kapasitansi dengan kode angka dan huruf

Kadang-kadang nilai kapasitansi dari kapasitor dituliskan dikemasan kapasitor dengan kode huruf dan angka. Dimana angka menunjukkan nilai kapasitansi sedangkan huruf menandakan satuan kapasitor. Contoh pada kemasan kapasitor tertulis 22n berarti nilai kapasitor tersebut adalah 22 nF.



Gambar 2.24 Contoh kapasitor dengan penulisan kapasitansi menggunakan kode huruf dan angka¹

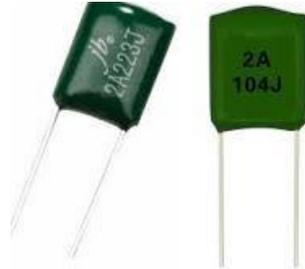
c) Penulisan Kode Toleransi dan Tegangan kapasitor

Pada gambar di atas setelah penulisan kode kapasitansi terdapat lagi kode huruf dan angka. Pada kapasitor pertama tertulis kode 100nJ 63. Kode 100n adalah nilai kapasitansi 100 nF sedangkan kode J menunjukkan nilai toleransi suhu sedangkan kode 63 menunjukkan nilai tegangan (63 V). Begitu pula pada kapasitor kedua tertulis kode 33nK 100. Kode 33n adalah nilai kapasitansi 33 nF sedangkan kode K menunjukkan nilai toleransi suhu sedangkan kode 100 menunjukkan nilai tegangan (100 V). Besarnya kode toleransi suhu dapat dilihat pada tabel dibawah ini:

Tabel 2.1 Toleransi koefisien suhu¹

Toleransi Koefisien suhu	
Simbol	PPM/ C ⁰
G	± 30
H	± 60
J	± 120
K	± 250
L	± 500

Terkadang sebelum nilai kode kapasitansi tertulis kode angka dan huruf seperti gambar dibawah ini. Kode tersebut merupakan kode nilai toleransi dari kapasitor dengan tabel kode seperti di bawah ini.



Gambar 2.25 Contoh kapasitor dengan penulisan toleransi dan tegangan kapasitor¹

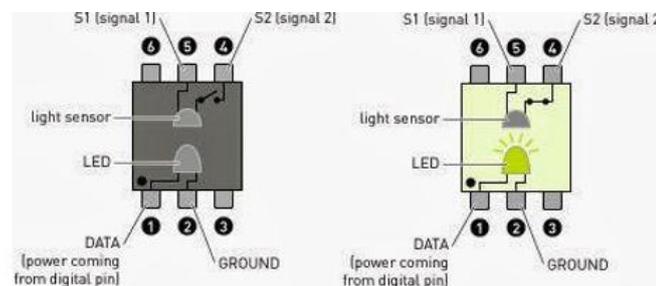
- d) Kapasitor dengan penulisan kapasitansi menggunakan kode warna
 Seperti pada resistor penulisan nilai kapasitansi ada juga yang menggunakan kode warna meskipun jarang digunakan. Terdapat dua macam kode warna yaitu dengan empat warna dan lima warna. Cara menghitung kode warna dapat dilihat pada tabel berikut ini:

Tabel 2.2 Nilai kode warna pada kapasitor¹

Warna	Warna ke-1	Warna Ke-2	Warna k-3	Warna ke-4
	Digit 1	Digit 2	Pengali 10^x	Toleransi %
Hitam	0	0	$10^0 = 1$	20
Coklat	1	1	$10^1 = 10$	
Merah	2	2	$10^2 = 100$	
Oranye	3	3	$10^3 = 1000$	
Kuning	4	4	$10^4 = 10000$	
Hijau	5	5	$10^5 = 100000$	5
Biru	6	6	$10^6 = 1000000$	
Ungu	7	7	$10^7 = 10000000$	
Abu-abu	8	8	$10^8 = 100000000$	
Putih	9	9	$10^9 = 1000000000$	10

2.7 Optocoupler

Optocoupler adalah komponen elektronika yang berfungsi sebagai penghubung berdasarkan cahaya optik. Pada dasarnya *Optocoupler* terdiri dari dua bagian utama yaitu Transmitter yang berfungsi sebagai pengirim cahaya optik dan Receiver yang berfungsi sebagai pendeteksi sumber cahaya. Masing-masing bagian *Optocoupler* (*Transmitter* dan *Receiver*) tidak memiliki hubungan konduktif rangkaian secara langsung tetapi dibuat sedemikian rupa dalam satu kemasan komponen.



Gambar 2.26 Tampilan Konstruksi *Optocoupler*¹

2.8 Relay

Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar/*Switch*). *Relay* menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat *menghantarkan* listrik yang bertegangan lebih tinggi.



Gambar 2.27 Bentuk Fisik *Relay*

2.9 Mikrokontroler⁴

Mikrokontroler adalah salah satu dari bagian dasar dari suatu sistem komputer. Meskipun mempunyai bentuk yang jauh lebih kecil dari suatu komputer pribadi dan komputer *mainframe* (komputer besar yang digunakan untuk memproses data dan aplikasi yang besar), mikrokontroler dibangun dari elemen-elemen dasar yang sama. Secara sederhana, komputer akan menghasilkan keluaran data spesifik berdasarkan masukan data yang diterima dan program yang dikerjakan.

Seperti umumnya komputer, mikrokontroler adalah alat yang mengerjakan instruksi-instruksi yang diberikan kepadanya. Artinya, bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri yang dibuat oleh seorang *programmer*. Program ini menginstruksikan komputer untuk melakukan jalinan yang panjang dari aksi-aksi sederhana untuk melakukan tugas yang lebih kompleks yang diinginkan oleh *programmer*.

Perkembangan mikrokontroler sangat pesat. Saat ini jenis-jenis mikrokontroler memiliki berbagai macam jenis, bentuk, fungsi yang beragam. Mikrokontroler dapat diprogram sesuai dengan keinginan pengguna atau *user* dengan program tertentu yang akan dimasukkan kedalam mikrokontroler tersebut. Bahasa pemrograman untuk mikrokontroler tersebut bermacam-macam yang digunakan, sebagai contohnya bahasa C, C++, *basic*, dan *assembly*. Bahasa tersebut digunakan sesuai dengan mikrokontroler yang digunakan. Pada penelitian ini, penulis menggunakan mikrokontroler tertentu yaitu *Arduino*.

Arduino adalah sebuah *platform* elektronik yang *open source*. *Arduino* yang digunakan adalah *Arduino Mega2560* dan *Arduino Uno*. Bahasa pemrograman yang digunakan oleh *Arduino* adalah bahasa C.

Arduino memiliki pemrograman sendiri yang dikenal dengan *Integrated Development Environment (IDE)*. Pemrograman tersebut digunakan untuk menulis program yang nanti akan diunggah kedalam *Arduino*.

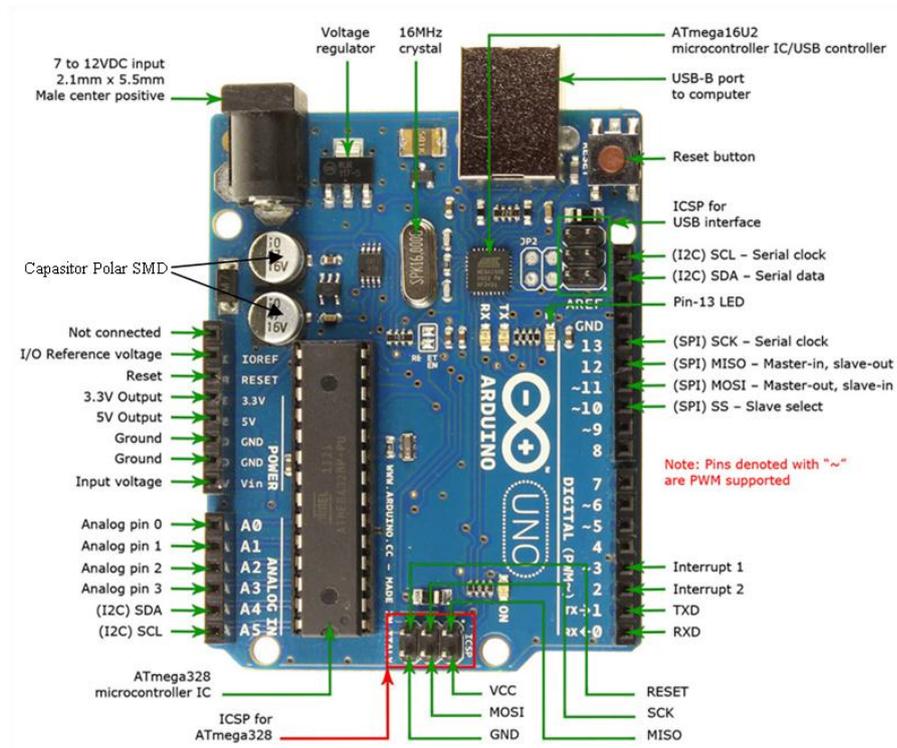
Program tersebut dapat berfungsi pada beberapa sistem operasi seperti

⁴ Kusuma, 2015, *Lengan Robot Peniru Gerakan Tangan Manusia*, Yogyakarta, Universitas Sanata Dharma, hlm. 6-12.

Windows, Macintosh, dan Linux. Selain itu *Arduino* juga dapat saling berkomunikasi antar *Arduino*. Komunikasi tersebut dapat dilakukan dengan komunikasi *Serial*. Komunikasi tersebut memanfaatkan *port Tx* dan *Rx* pada *Arduino*.

2.10 *Arduino*⁴

Arduino adalah jenis suatu papan (*board*) yang berisi mikrokontroler. Dengan perkataan lain, *Arduino* dapat disebut sebagai sebuah papan mikrokontroler. Salah satu papan *Arduino* yang terkenal adalah *Arduino Uno*. *Arduino Uno* adalah mikrokontroler menggunakan *ATmega328P*. *Uno* memiliki 14 *PIN input* atau *output* digital (dimana 6 *PIN* dapat digunakan sebagai *output PWM*), 6 *input* analog, 16 MHz osilator kristal, koneksi *USB*, *jack power*, *ICSP header*, dan tombol *reset*. Secara spesifikasi, *Arduino Uno* sama dengan *Duemilanove*, dan *Uno* diluncurkan di tahun 2010 sedangkan *Duemilanove* diluncurkan di tahun 2009.



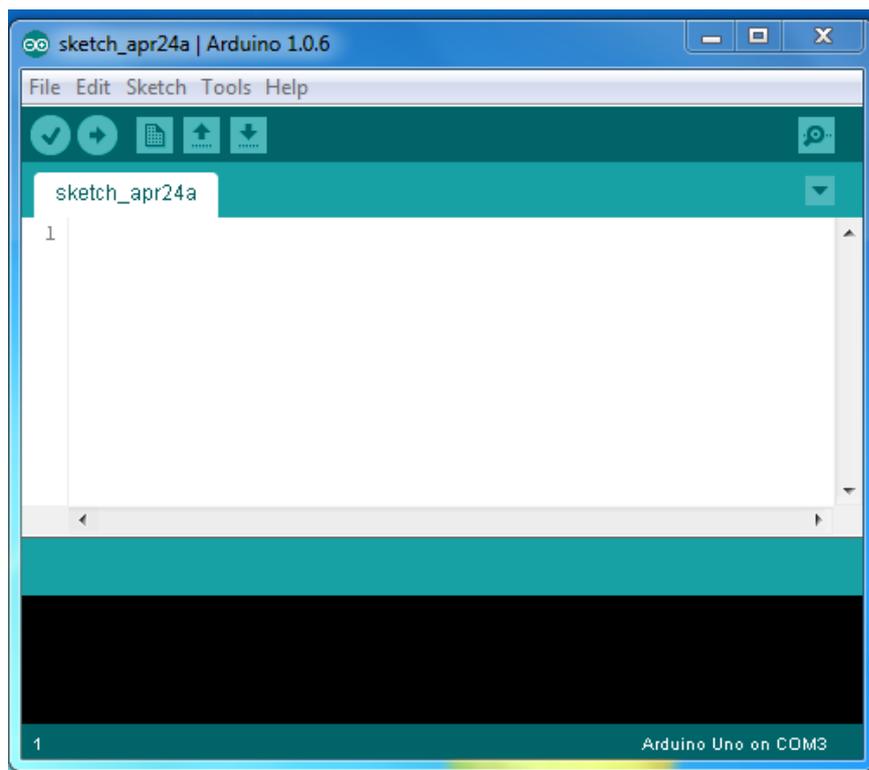
Gambar 2.28 Tampilan *Arduino Uno*⁴

Tabel 2.3 Keterangan *PIN* *Arduino Uno*⁴

No.	Parameter	Keterangan
1	<i>ATMega328P</i>	<i>IC</i> mikrokontroler yang digunakan pada <i>Arduino Uno</i> .
2	<i>Jack USB</i>	Untuk komunikasi mikrokontroler dengan <i>PC</i> .
3	<i>Jack Adaptor</i>	Masukan <i>power</i> eksternal bila <i>Arduino</i> bekerja mandiri (tanpa komunikasi dengan <i>PC</i> melalui kabel <i>Serial USB</i>).
4	Tombol <i>Reset</i>	Tombol <i>reset internal</i> yang digunakan untuk me- <i>reset</i> modul <i>Arduino</i> .
5	<i>PIN Analog</i>	Menerima <i>input</i> dari perangkat analog lainnya.
6	<i>PIN Power</i>	<ul style="list-style-type: none"> • <i>Vin</i> = Masukan tegangan <i>input</i> bagi <i>Arduino</i> ketika menggunakan sumber daya eksternal. • 5V = Sumber tegangan yang dihasilkan regulator <i>internal board Arduino</i>. • 3,3V = Sumber tegangan yang dihasilkan regulator <i>internal board Arduino</i>. Arus maksimal pada <i>PIN</i> ini adalah 50 mA. • <i>GND</i> = <i>PIN ground</i> dari regulator tegangan <i>board Arduino</i>. • <i>IOREF</i> = Tegangan Referensi. • <i>AREF</i> = Tegangan Referensi untuk <i>input</i> analog.
7	<i>PIN Digital</i>	<i>PIN</i> yang digunakan untuk menerima <i>input digital</i> dan memberi <i>output</i> berbentuk digital (0 dan 1 atau <i>low</i> dan <i>high</i>)
8	<i>PIN Serial</i>	Digunakan untuk menerima dan mengirimkan data <i>Serial TTL (Receiver (Rx), Transmitter (Tx))</i> . <i>PIN</i> 0 dan 1 sudah terhubung kepada <i>PIN Serial USB to TTL</i> sesuai dengan <i>PIN ATMega</i> .

2.11 Perangkat Lunak *Arduino*⁴

Area pemrograman *Arduino* dikenal dengan *Integrated Development Environment (IDE)*. Area pemrograman yang digunakan untuk menulis baris program dan mengunggahnya ke dalam *board Arduino*, disamping itu juga dibuat lebih mudah dan dapat berjalan pada beberapa sistem operasi seperti *Windows*, *Macintosh*, dan *Linux*. Gambar 2.29 dan Tabel 2.4 merupakan area pemrograman *Arduino* dan keterangan beberapa tombol utama. *IDE Arduino* membutuhkan beberapa pengaturan yang digunakan untuk mendeteksi *board Arduino* yang sudah dihubungkan ke komputer. Beberapa pengaturan tersebut adalah mengatur jenis *board* yang digunakan sesuai dengan *board* yang terpasang dan mengatur jalur komunikasi data melalui perintah *Serial Port*.



Gambar 2.29 Tampilan *IDE Arduino*⁴

⁴ Kusuma, 2015, *Lengan Robot Peniru Gerakan Tangan Manusia*, Yogyakarta, Universitas Sanata Dharma, hlm. 6-12.

Tabel 2.4 Keterangan tombol pada tampilan *IDE Arduino*⁴

No.	Nama	Fungsi
1	<i>Verify</i>	Menguji apakah ada kesalahan pada program atau <i>sketch</i> . Apabila <i>sketch</i> sudah benar, maka <i>sketch</i> tersebut akan dikompilasi. Kompilasi adalah proses mengubah kode program ke dalam kode mesin.
2	<i>Upload</i>	Mengirimkan kode mesin hasil kompilasi ke <i>board Arduino</i>
3	<i>New</i>	Membuat <i>sketch</i> yang baru.
4	<i>Open</i>	Membuka <i>sketch</i> yang sudah ada.
5	<i>Save</i>	Menyimpan <i>sketch</i> .
6	<i>Serial Monitor</i>	Menampilkan data yang dikirim dan diterima melalui komunikasi <i>Serial</i> .

2.11.1 Komunikasi *Serial*⁴

Komunikasi *Serial* merupakan komunikasi data dengan pengiriman data secara satu per satu pada waktu tertentu. Komunikasi data *Serial* hanya menggunakan dua kabel yaitu kabel data untuk pengiriman yang disebut *transmit* (TX) dan kabel data untuk penerimaan yang disebut *receive* (RX). Kelebihan dari komunikasi *Serial* adalah jarak pengiriman dan penerimaan dapat dilakukan dalam jarak yang cukup jauh dibandingkan dengan komunikasi parallel tetapi kekurangannya kecepatannya lebih lambat dibandingkan komunikasi parallel.

Dalam komunikasi *Serial* dikenal ada dua *mode* komunikasi *Serial*:

a. *Mode Sinkron*

Mode sinkron merupakan mode komunikasi yang pengiriman tiap *bit* data dilakukan dengan menggunakan sinkronisasi *clock*. Saat *transmitter* hendak mengirimkan data, harus disertai *clock* untuk sinkronisasi antara *transmitter* dan *receiver*.

b. *Mode Asinkron*

Komunikasi asinkron *Serial* merupakan sebuah *protocol* transmisi *asinkronous*, dimana komunikasi ini tidak menggunakan *clock*, tetapi telah memiliki *baudrate* (kecepatan aliran data)⁴ yang telah disepakati oleh masing-masing sistem yang sedang berkomunikasi. Kerja dari komunikasi ini adalah, *signal start* dikirimkan pada saat sebelum data dikirimkan dan *signal stop* dikirimkan setelah setiap data selesai dikirimkan.

Signal start digunakan untuk mempersiapkan mekanisme penerimaan untuk menerima dan memproses data yang akan dikirimkan dan *signal stop* berguna untuk mempersiapkan mekanisme penerimaan data berikutnya.

2.12 Perintah *IF* dan *IF – ELSE*³

Perintah *IF* memiliki beberapa kombinasi, bisa *IF* saja, *IF-ELSE*, *IF-ELSE IF-ELSE*, dan seterusnya. Semakin kompleks tentu logika yang dipakai akan tampak semakin rumit.

```

1 // Free Ebook Arduino
2 // www.elangsakti.com
3 // coder elangsakti
4 // Pin 8 untuk LED
5
6 const int pinLED = 8;
7
8 void setup() {
9 // pin LED sebagai output
10 pinMode(pinLED, OUTPUT);
11 }
12
13 // awal time delay 1000 | 1 detik
14 int timeDelay = 1000;
15
16 void loop() {
17 // Setiap looping, nilai timeDelay dikurangi 100
18 timeDelay = timeDelay - 100
19

```

³Hari Santoso, “Arduino Untuk Pemula”, www.elangsakti.com (diakses 27 Mei 2016)

```

20  /* Jika timeDelay bernilai 0 atau negatif
21  maka nilai timeDelay direset ke 1000
22  */
23  if(timeDelay <= 0){
24  timeDelay = 1000;
25  }

26
27  //Nyalakan dan matikan LED selama timeDelay
28  digitalWrite(pinLED, HIGH);
29  delay(timeDelay);
30  digitalWrite(pinLED, LOW);
31  delay(timeDelay);
32  }

```

Sketch di atas digunakan untuk mengontrol *LED Flip-flop* dengan prinsip kerja *LED* awalnya akan berkedip pelan, selanjutnya akan berkedip dengan cepat, dan akhirnya akan berkedip pelan kembali, berikut pembahasannya:

Ketika awal dinyalakan, maka *timeDelay* adalah 1000. Nilai tersebut diinisialisasi pada baris 14. Baris ini tidak dijadikan konstanta (*const*) sebagaimana *pinLED* karena nilai *timeDelay* akan diubah-ubah.

```

14  int timeDelay = 1000;

```

Setelah masuk ke bagian utama aplikasi, pada baris 18 nilai *timeDelay* dikurangi 100.

```

18  timeDelay = timeDelay - 100

```

Sehingga *timeDelay* pertama yang digunakan untuk menyalakan dan mematikan *LED* adalah 900. Pada perulangan selanjutnya, *timeDelay* kembali dikurangi 100, sehingga perulangan ke dua menggunakan *timeDelay* 800 (lebih singkat dari *timeDelay* yang awal), begitu seterusnya hingga pada *timeDelay* 100.

Pada waktu melewati *timeDelay* 100, ketika dikurangi dengan 100, maka *timeDelay* bernilai 0. Kondisi ini cocok dengan baris ke 23.

```

23  if(timeDelay <= 0){
24  timeDelay = 1000;

```

Jika (*IF*) *timeDelay* lebih kecil atau sama dengan 0, maka *timeDelay* akan diisi dengan 1000. Sehingga nilai *timeDelay* tidak akan pernah negatif dan hal tersebut akan

berulang terus-menerus. Hal ini lah yang membuat durasi nyala hidup-mati LED bisa berubah lebih cepat.

Perintah *IF* pasti akan diikuti dengan kondisi yang bernilai *True* yang diapit dengan tanda kurung, *if (kondisi)*. Pada Sketch di atas kondisi yang digunakan adalah *timeDelay <= 0*, *timeDelay* lebih kecil atau sama dengan nol. Artinya, jika *timeDelay* bernilai 0 atau lebih kecil dari 0, maka blok kode dalam *if* akan dieksekusi.

Berikut beberapa operator matematika yang dibutuhkan dalam bahasa pemrograman yaitu:

Tabel 2.5 Operator aritmatika³

Operator	Arti
=	Operator assignment, untuk memberi nilai pada variabel
+	Operator penambahan
-	Operator pengurangan
*	Operator perkalian
/	Operator pembagian. Sebagai catatan: - Jika tipe data yang digunakan adalah integer (<i>int</i>), maka hasil bagi adalah nilai asli, bukan desimal. Misal $5/2 = 2$, bukan 2.5 atau 3 . Tapi jika tipe data yang digunakan adalah double/float, maka hasil bagi adalah angka desimal. Misal, $5/2 = 2.5$
%	Operator modulo (sisanya pembagian). Misal: - $10\%2 = 0$, 10 dibagi 2 = 5 + 0 - $10\%3 = 1$, 10 dibagi 3 = 3 + 1 - $10\%4 = 2$, 10 dibagi 4 = 2 + 2 - $10\%5 = 0$, 10 dibagi 5 = 2 + 0

Selain operator aritmatika, maka berikut ini adalah beberapa operator relasional yang biasa digunakan dalam bentuk perbandingan:

Table 2.6 Operator relasional³

Operator	Arti
==	Sama dengan
!=	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil atau sama dengan
>=	Lebih besar atau sama dengan