

BAB II

TINJAUAN PUSTAKA

II.1 Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang mengubah pengukuran sinyal getaran dari domain waktu ke domain frekuensi dengan menggunakan analisis fast fourier transform (FFT) dari sensor accelerometer “Analisa Getaran Pada Mesin Sepeda Motor Berbasis Labview. Getaran adalah suatu proses gerak bolak - bolik secara berkala melalui suatu titik keseimbangan. Rancang Bangun Sistem Pendeteksian Getaran Mesin Kendaraan Bermotor untuk Menganalisis Keadaan Performa Mesin Berbasis Mikrokontroler Arduino Nano ini untuk mengukur performa getaran mesin sepeda motor untuk membantu para mekanik untuk mengukur kestabilan mesin sepeda motor (Sunarko : 2019).

Berdasarkan penelitian “Sistem Identifikasi Kondisi Sepeda Motor dengan Algoritma Fast Fourier Transform serta Bantuan Android dan Google Maps”. mengubah pengukuran sinyal getaran dari domain waktu ke domain frekuensi dengan menggunakan analisis FFT dan menggunakan metode Predictive Maintenance serta tambahan fitur aplikasi mobile yang bisa dibawa kemana-mana. Penelitian yang telah dilakukan menggunakan prinsip kontak langsung dimana objek yang diukur kontak langsung dengan instrumen yang digunakan. Pengukuran frekuensi getaran menggunakan serat optik telah banyak dilakukan oleh beberapa peneliti, diantaranya Spillman dkk (1998) dengan melakukan penginderaan getaran

menggunakan deteksi perubahan dalam distribusi spasial energi dalam output dari serat optik multimode (Risa : 2016)

Berdasarkan penelitian “Pengukuran Frekuensi Getaran Menggunakan Serat Optik”. memanfaatkan serat optik sebagai sensor yang mengubah besaran fisik getaran yaitu simpangan menjadi besaran tegangan. Jika objek yang akan diukur berada pada suhu yang tinggi atau terlalu rendah maka instrumen yang digunakan bisa rusak. Sensor yang digunakan juga sangat sensitif pada suhu yang tinggi dan kemungkinan bisa rusak. Dalam penelitian ini digunakan sensor serat optik dengan metode non kontak sehingga instrumen yang digunakan lebih tahan lama ketika melakukan pengukuran. (Zulaicah : 2019)

II.2. Uraian Teoritis

II.2.1. mesin kendaraan Bermotor

Kendaraan bermotor saat ini telah menjadi sarana mobilitas yang penting bagi manusia untuk mempermudah kegiatan manusia berpindah lokasi, mengangkut barang, dan lain-lain. Sepeda motor adalah kendaraan beroda dua yang ditenagai oleh sebuah mesin yang berbahan bakar bensin. Kedua rodanya sejajar depan dan belakang, pada kecepatan tinggi sepeda motor tetap stabil dan tidak terjatuh disebabkan oleh gaya giroskopik, pada kecepatan rendah pengaturan dikendalikan oleh pengguna melalui stang sepeda motor. Sepeda motor sangat diminati karena lebih cepat dalam kondisi kemacetan, lebih ringkas, irit bahan bakar, dan harga lebih murah dibandingkan kendaraan lainnya (Setiawan : 2013)

II.2.2 . Getaran

Getaran adalah sebuah Gerakan bolak balik suatu partikel dari titik acuannya (osilation) yang lemahnya getaran dipengaruhi oleh gaya yang diberikan. Kondisi suatu mesin dan masalah-masalah kerusakan mekanik yang terjadi dapat diketahui dengan

mengukur karakteristik sinyal getaran pada mesin tersebut dengan mengacu pada gerakan pegas. Karakteristik suatu getaran dengan memetakan gerakan dari pegas tersebut terhadap waktu. Pada gambar dibawah ini menjelaskan tentang karakteristik getaran. Parameter getaran dalam pengukuran amplitudo dapat dinyatakan dalam simpangan getaran (vibration displacement), kecepatan getaran (vibration velocity), dan percepatan getaran (vibration acceleration). Perpindahan getaran (vibration displacement) adalah jarak yang ditempuh dari kondisi awal hingga simpangan maksimum (peak). Perpindahan tersebut dinyatakan dalam satuan micron (μm) atau mils. Dimana : $1 \mu\text{m} = 0.001\text{mm}$ dan $1 \text{ mils} = 0.001 \text{ inch}$. Kecepatan getaran (vibration velocity) adalah simpangan getaran setiap satuan waktu, dinyatakan dalam mm/s atau inch/s. Percepatan getaran (vibration acceleration) adalah kecepatan getaran setiap satuan waktu yang dinyatakan dalam mm/s^2 , inch/s^2 , dan satuan gravitasi (g) (Sunarko : 2010)

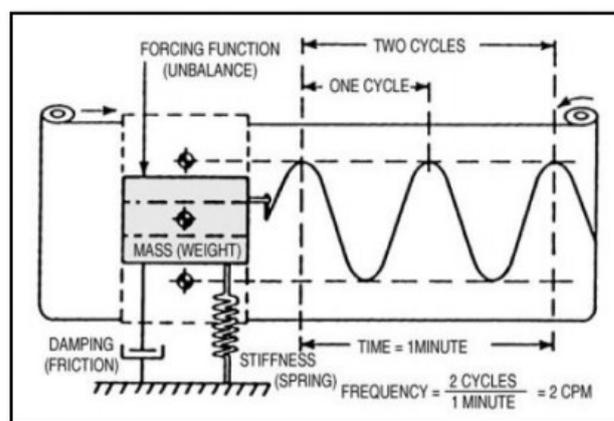
Untuk menentukan apakah getaran yang terjadi masih memenuhi syarat atau tidak, dilakukan pengukuran dengan diagnosis :

- a. Comparative
- b. Descriptive
- c. Trending

II.3. Analisis Getaran

Getaran secara teknik didefinisikan sebagai gerak osilasi dari suatu objek terhadap posisi awalnya. Semua mesin memiliki tiga sifat fundamental yang berhubungan untuk menentukan bagaimana mesin akan bereaksi terhadap kekuatan-kekuatan yang menyebabkan getaran-getaran, seperti sistem pegas-massa, yaitu :

- 1) Massa (m) Merupakan inersia untuk tetap dalam keadaan semula atau gerak. Sebuah gaya mencoba untuk membawa perubahan dalam keadaan istirahat atau gerak, yang ditentang oleh massa dan satuannya dalam kg.
- 2) Kekakuan atau stiffness (k) Ada kekakuan tertentu yang dipersyaratkan membengkokkan atau membelokkan struktur dengan jarak tertentu. Ini mengukur gaya yang diperlukan untuk memperoleh defleksi tertentu disebut kekakuan, satuan dalam N/m.
- 3) Damping atau redaman (c) Setelah memaksa set bagian atau struktur ke dalam gerakan, bagian atau struktur akan memiliki mekanisme inheren untuk memperlambat gerak (kecepatan). Karakteristik ini untuk mengurangi kecepatan gerak disebut redaman, satuannya dalam N/(m/s).



Gambar II.1 Sistem Getaran Sederhana

(Sumber : D. Petruzella, Frank. 2001. Elektronik Industri. Yogyakarta : Andi)

II.4. Mikrokontroler

Mikrokontroler adalah piranti elektronik berupa IC (Integrated Circuit) yang memiliki kemampuan manipulasi data (informasi) berdasarkan suatu urutan instruksi (program) yang dibuat oleh programmer. Mikrokontroler atau kadang dinamakan pengontrol tertanam

(embeddedcontroller) adalah suatu sistem yang mengandung masukan atau keluaran, memori, dan prosesor yang digunakan pada produk seperti mesin cuci, pemutar video, mobil dan telepon. Pada prinsipnya, Mikrokontroler adalah sebuah komputer berukuran kecil yang dapat digunakan untuk mengambil keputusan, melakukan hal-hal bersifat berulang dan dapat berinteraksi dengan peranti -peranti eksternal, seperti sensor ultrasonik untuk mengukur jarak terhadap suatu objek, penerima GPS untuk memperoleh data posisi kebumihan dari satelit dan motor untuk mengontrol gerak pada robot. Sebagai komputer yang berukuran kecil, mikrokontroler cocok diaplikasikan pada benda yang berukuran kecil (Kadir, A : 2015).

Agar sebuah mikrokontroler dapat berfungsi, maka mikrokontroler tersebut memerlukan komponen eksternal yang kemudian disebut dengan sistem minimum. Untuk membuat sistem minimal paling tidak dibutuhkan sistem clock dan reset, walaupun pada beberapa mikrokontroler sudah menyediakan sistem clock internal, sehingga tanpa rangkaian eksternal pun mikrokontroler sudah beroperasi. Yang dimaksud dengan sistem minimal adalah sebuah rangkaian mikrokontroler yang sudah dapat digunakan untuk menjalankan sebuah aplikasi. Sebuah IC mikrokontroler tidak akan berarti bila hanya berdiri sendiri. Pada dasarnya sebuah sistem minimal mikrokontroler AVR memiliki prinsip yang sama. Dimana mikrokontroler atmega 328 sebagai pengendalinya (Agung R : 2012).

II.5. Arduino Nano

Arduino merupakan sebuah platform dari physical computing yang bersifat open source. Pertama-tama perlu dipahami bahwa kata “platform” di sini adalah sebuah pilihan kata yang tepat. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari hardware, bahasa pemrograman dan Integrated Development Environment (IDE) yang canggih. IDE adalah sebuah software yang sangat berperan untuk menulis program, meng-compile menjadi kode biner dan meng-upload ke dalam memory

microcontroller Arduino Nano adalah salah satu papan pengembangan mikrokontroler yang berukuran kecil, lengkap dan mendukung penggunaan breadboard. Arduino Nano diciptakan dengan basis mikrokontroler ATmega328 (untuk Arduino Nano versi 3.x) atau ATmega 168 (untuk Arduino versi 2.x). Arduino Nano kurang lebih memiliki fungsi yang sama dengan Arduino Duemilanove, tetapi dalam paket yang berbeda. Arduino Nano tidak menyertakan colokan DC berjenis Barrel Jack, dan dihubungkan ke komputer menggunakan port USB Mini-B. Arduino Nano dirancang dan diproduksi oleh perusahaan Gravitech (Kadir, A : 2015).



Gambar II.2 Arduino Nano

(Sumber : <http://family-cybercode.blogspot.com/mengenal-arduino-nano.html/10/10/2021/>)

II.6. Konfigurasi Pin Arduino Nano

Konfigurasi pin Arduino Nano. Arduino Nano memiliki 10 Pin. Berikut Konfigurasi pin Arduino Nano.

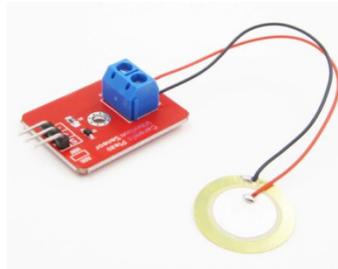
1. VCC merupakan pin yang berfungsi sebagai pin masukan catu daya digital.
2. GND merupakan pin ground untuk catu daya digital.
3. AREF merupakan Referensi tegangan untuk input analog. Digunakan dengan fungsi `analogReference()`.

4. RESET merupakan Jalur LOW ini digunakan untuk me-reset (menghidupkan ulang) mikrokontroler. Biasanya digunakan untuk menambahkan tombol reset pada shield yang menghalangi papan utama Arduino
5. Serial RX (0) merupakan pin yang berfungsi sebagai penerima TTL data serial.
6. Serial TX (1) merupakan pin yang berfungsi sebagai pengirim TT data serial.
7. External Interrupt (Interupsi Eksternal) merupakan pin yang dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau perubahan nilai.
8. Output PWM 8-Bit merupakan pin yang berfungsi untuk analogWrite().
9. SPI merupakan pin yang berfungsi sebagai pendukung komunikasi.
9. LED merupakan pin yang berfungsi sebagai pin yang diset bernilai HIGH, maka LED akan menyala, ketika pin diset bernilai LOW maka LED padam.

II.7. Sensor Getar Piezoelectric

Merupakan salah satu sensor yang dapat mengukur getaran suatu benda yang nantinya dimana data tersebut akan diproses untuk kepentingan percobaan ataupun digunakan untuk mengantisipasi sebuah kemungkinan adanya mara bahaya. Salah satu jenis sensor getaran yang saat ini sering digunakan adalah accelerometer, alat ini merupakan alat yang dapat berfungsi untuk mengukur percepatan dari sebuah benda. Percepatan tersebut diukur bukan dengan menggunakan koordinat dari percepatan tersebut, melainkan dengan mengukur percepatan berdasarkan fenomena pergerakan benda yang di hubungkan dengan perubahan massa yang terjadi di dalam alat pengukur tersebut. Contohnya adalah sensor SW-420. Sensor module SW-420 adalah sensor untuk mendeteksi getaran, cara kerja sensor ini adalah dengan menggunakan 1 buah pelampung logam yang akan bergetar ditabung yang

berisi 2 elektroda ketika modul sensor menerima getaran / shock. Terdapat 2 output yaitu digital output (0 dan 1) dan analog output (tegangan). (Kadir, A : 2015)



Gambar II.3 Sensor Getar Piezoelectric

sumber : www.samrasyid.com/pengertian-sensor-piezoelektrik.html/30/12/2021

II.8. Regulator

Regulator adalah rangkaian regulasi atau pengatur tegangan keluaran dari sebuah catu daya agar efek darinaik atau turunnya tegangan jala-jala tidak mempengaruhi tegangan catu daya sehingga menjadi stabil. Rangkaian penyearah sudah cukup bagus jika tegangan ripple - nya kecil, tetapi ada masalah stabilitas. Jika tegangan PLN naik/turun, maka tegangan outputnya juga akan naik/turun. Seperti rangkaian penyearah di atas, jika arus semakin besar ternyata tegangan dc keluarannya juga ikut turun. Untuk beberapa aplikasi perubahan tegangan ini cukup mengganggu, sehingga diperlukan komponen aktif yang dapat meregulasi tegangan keluaran ini menjadi stabil (Kadir, A : 2015).

1. Perlunya Regulator

Ada beberapa alasan yang mungkin diperlukannya sebuah regulator

- a. Fluktuasi tegangan jala-jala
- b. Perubahan tegangan akibat beban (loading)
- c. Perlu pembatasan arus dan tegangan untuk keperluan tertentu

Sekarang tidak perlu susah payah lagi mencari op-amp, transistor dan komponen lainnya untuk merealisasikan rangkaian regulator seperti di atas karena rangkaian semacam ini sudah dikemas menjadi satu IC regulator tegangan tetap.

Saat ini sudah banyak dikenal komponen seri 78XX sebagai regulator tegangan tetap positif dan seri 79XX yang merupakan regulator untuk tegangan tetap negatif. Bahkan komponen ini biasanya sudah dilengkapi dengan pembatas arus (current limiter) dan juga pembatas suhu (thermal shutdown). Komponen ini hanya tiga pin dan dengan menambah beberapa komponen saja sudah dapat menjadi rangkaian catu daya yang ter-regulasi dengan baik. Misalnya 7805 adalah regulator untuk mendapat tegangan 5 volt, 7812 regulator tegangan 12 volt .



Gambar II.4 Ic Regulator 7805

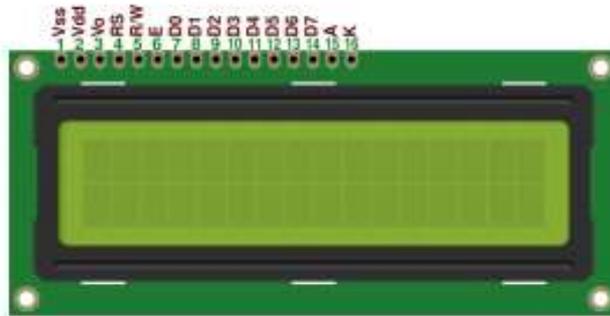
(Sumber : <http://family-cybercode.blogspot.com/mengenal-arduino-nano.html/10/10/2021/>)

II.9. LCD 16 x 2

LCD (Liquid Crystal Display) adalah suatu jenis media tampil yang menggunakan kristal cair sebagai penampil utama. Adapun fitur yang disajikan dalam LCD ini adalah:

- a. Terdiri dari 16 karakter dan 2 baris
- b. Mempunyai 192 karakter tersimpan.
- c. Terdapat karakter generator terprogram.
- d. Dapat dialamati dengan mode 4-bit dan 8-bit.
- e. Dilengkapi dengan back light.

Proses inialisasi pin arduino yang terhubung ke pin LCD RS, Enable, D4, D5, D6, dan D7, dilakukan dalam baris LiquidCrystal (2, 3, 4, 5, 6, 7), dimana lcd merupakan variable yang dipanggil setiap kali intruksi terkait LCD akan digunakan.

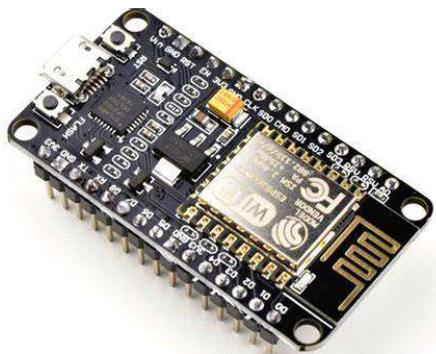


Gambar II.5 LCD 16 x 2

(Sumber : <http://family-cybercode.blogspot.com/mengenal-arduino-nano.html/10/10/2021/>)

II.10. NodeMCUESP 8699

NodeMCU ESP8266 adalah salah satu mikrokontroler yang biasa digunakan untuk kepentingan Internet of Things (IoT) dikarenakan fasilitasnya yang sudah dilengkapi dengan wifi untuk terkoneksi dengan internet



Gambar II.7 NodeMCU 8699

(Sumber : <http://family-cybercode.blogspot.com/mengenal-arduino-nano.html/10/03/2023>)

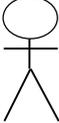
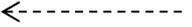
II.10. UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah bahasa spesifik standar yang dipergunakan untuk mendokumentasikan, menspesifikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak digunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Ade Hendeni : 2016)

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel II.1 dibawah ini :

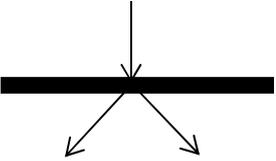
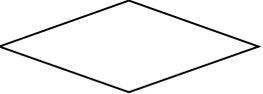
Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor,</p>
	<p><i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang atau system yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktir harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i>, merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i>, merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.</p>

Gambar II.7 Tabel Simbol *Use Case*

(Sumber: Ade Hendini: 2016: 108)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.2:

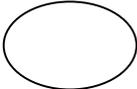
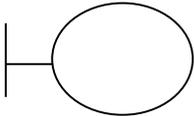
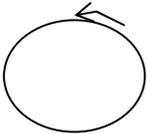
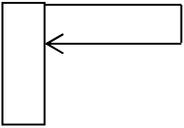
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

Gambar II.8 Tabel Simbol *Activity Diagram*

(Sumber: Ade Hendini: 2016: 109)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada Gambar tabel II.3:

Gambar	Keterangan
	<p><i>Entity Class</i>, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem</p>
	<p><i>Boundary Class</i>, berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.</p>
	<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas,</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

Tabel II.9. Simbol Sequence Diagram

(Sumber: Ade Hendini: 2016: 110)

4. Diagram Kelas (*Class Diagram*)

Class diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti (Ade Hendini, 2016: 111). Simbol *class diagram* dan *multiplicity class diagram* dapat dilihat pada Tabel II.4 dan Tabel II.5. dibawah ini :

Gambar	Keterangan			
	<p><i>Generalization</i>, untuk menghubungkan antar kelas dengan arti umum-khusus. Jadi jika ada kelas dengan arti umum-khusus. Jadi jika ada kelas bermakna umum dan kelas bermakna khusus dapat menggunakan simbol ini.</p>			
<table border="1" data-bbox="225 1294 438 1520"> <tr> <td data-bbox="225 1294 438 1368">Nama_kelas</td> </tr> <tr> <td data-bbox="225 1368 438 1442">+atribut</td> </tr> <tr> <td data-bbox="225 1442 438 1520">+operasi</td> </tr> </table>	Nama_kelas	+atribut	+operasi	<p><i>Class</i>, untuk sebuah kelas pada struktur sistem. Penulisan tidak boleh menggunakan spasi. Simbol ini memiliki 3 susunan, yaitu kotak pertama adalah nama kelas, kedua atribut dan ketiga operasi.</p>
Nama_kelas				
+atribut				
+operasi				
	<p><i>Interface</i>, untuk simbol <i>interface</i> atau dalam bahasa indonesianya antar muka. Konsep yang digunakan pun sama dengan pemrograman berorientasi object (OOP).</p>			

	<i>Association</i> , digunakan untuk menghubungkan atau merelasikan kelas satu dengan kelas yang lainnya dengan makna umum.
	<i>Directed Association</i> , adalah relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
	<i>Aggregation</i> , adalah relasi antar kelas dengan makna semua bagian.
	<i>Dependency</i> , adalah relasi antar kelas dengan makna kebergantungan antar kelas.

Gambar II.10. Tabel Simbol *Class Diagram*

(Sumber: Ade Hendini: 2016: 111)

II.2.6. Bahasa Pemrograman C

Bahasa Pemrograman C adalah sebuah bahasa pemrograman komputer yang bisa digunakan untuk membuat berbagai aplikasi (*general-purpose programming language*), mulai dari sistem operasi (seperti Windows atau Linux), antivirus, software pengolah gambar (*image processing*), hingga *compiler* untuk bahasa pemrograman, dimana C banyak digunakan untuk membuat bahasa pemrograman lain yang salah satunya adalah PHP. Meskipun termasuk *general-purpose programming language*, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi,