

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Untuk mendukung penulisan penelitian ini diperlukan suatu referensi atau kajian pustaka tentang penelitian sebelumnya yang hampir sama dengan penelitian ini antara lain : (Wibowo, 2011) tentang Perancangan Karakteristik Sistem Suspensi Semi Aktif Untuk Meningkatkan Kenyamanan Kendaraan dengan kesimpulan dengan melaksanakan analisa persamaan pada model sistem suspensi diperoleh perbandingan sistem suspensi semi aktif dengan sistem suspensi pasif menghasilkan percepatan vertikal pada jalan sinusoida, bump dan random, dengan demikian sesuai dengan standart kenyamanan ISO-2631. Nilai yang dihasilkan lebih rendah dari pada sistem suspensi pasif, yang mengakibatkan kriteria yang dimiliki bertambah nyaman.

Menurut penelitian Putra, dkk (2016) tentang perancangan dan implementasi sistem kendali gerak pada skuter dua roda seimbang otomatis menggunakan metode PID berbasis mikrokontroler diperoleh hasil penggunaan mikrokontroler sebagai pusat kendali dan PID sebagai kendali dalam pengolahan kecepatan yang dikombinasikan juga dengan beberapa sensor. Kemudian data diproses dengan kontroler PID sehingga dapat mengatur arah dan kecepatan yang mengakibatkan sistem *self balance scooter* dapat bergerak maju dan mundur. Hasil yang didapat dari parameter PID yaitu $K_p = 8,0$, $K_i = 3,0$ dan $K_d = 6,0$.

Buana, dkk (2018) tentang Desain dan Analisa Aerodinamis Bodi Mobil Minimalis Roda Tiga Terhadap Tingkat Kestabilan Kendaraan Ditinjau dari Kondisi Skid dan Rolling, disimpulkan model desain bodi mobil dengan roof yang memiliki sudut buang aerodinamis sebesar 10 derajat dengan nilai $C_d = 0,28$, $C_l = -0,03$ dan $C_m = 0,03$. Setelah melihat pengaruh aerodinamis terhadap tingkat kestabilan kendaraan ditinjau dari kondisi skid dan rolling maka didapat kecepatan yang aman ketika berbelok jika tidak terdapat kemiringan sebesar 28 km/jam. Tetapi saat mobil berbelok pada kemiringan 6 derajat maka kecepatan aman yang

di dapat sebesar 34 km/jam. Jadi kondisi mobil ketika bergerak miring pada saat berbelok berpengaruh pada gaya sentrifugal yang mengecil, sehingga ketika mobil berbelok kecepatannya dapat lebih tinggi.

Biyanto (2006) dengan penelitian perancangan sistem suspensi aktif pada kendaraan roda empat menggunakan pengendali jenis robust proposional, integral dan derivative, didapatkan hasil dengan perancangan sistem suspensi aktif mampu meredam getaran yang diakibatkan gangguan di jalan sebesar 99% lebih serta mampu menjadikan nilai ITAE dan IAE lebih kecil dari pada suspensi pasif yang mengakibatkan sistem ini cocok untuk diterapkan.

Olviani & Guntur (2014) tentang Analisa Kenyamanan Kendaraan Roda Dua dengan Pemodelan Pengendara sebagai Sistem Multi DOF, di simpulkan suatu kendaraan bermotor yang dikendarai maka merasakan gangguan berupa getaran dari kondisi jalan, yang berakibat tidak baik bagi pengendara bila getaran tersebut sangat kencang. Perlu adanya faktor kenyamanan agar pengendara selalu aman, nyaman dan tidak terjadi suatu kecelakaan. Penelitian memberikan input eksitasi yang bervariasi, dengan pemodelan sistem 8° kemudian setelah mendapat model matematis yang diinginkan didapatkanlah persamaan gerak sistem sehingga didapatkan hasil beserta analisisnya. Nilai percepatan yang didapat dari simulasi dengan nilai input harmonik minimum yaitu 0,01m pada kecepatan 80km/jam (kecepatan maksimal) dan didapat nilai RMS percepatan dari tubuh pengendara di mulai dari paha ($0,4906 \text{ m/s}^2$), nilai lower torso ($0,5451 \text{ m/s}^2$), nilai upper torso ($0,5092 \text{ m/s}^2$) dan nilai kepala ($0,5145 \text{ m/s}^2$). Selanjutnya diberi input harmonik maksimal yaitu 0,03m dengan kecepatan yang sama yakni 80 km/jam, diperoleh nilai RMS dari paha ($1,472 \text{ m/s}^2$), nilai lower torso ($1,635 \text{ m/s}^2$), nilai upper torso ($1,528 \text{ m/s}^2$) dan nilai kepala ($1,544 \text{ m/s}^2$). Hasil yang diperoleh dari simulasi untuk kecepatan 20km/jam sesuai ISO 2631 nilai dari semua input pengendara memiliki standar kenyamanan yang dibawah rata-rata. Pada kecepatan 50 km/jam dan 80 km/jam nilai dari semua input masih berada dibawah 2 m/s^2 yang berarti masih dalam wajar standar kenyamanan, meskipun masih berasa ada ketidanyamanan.

Setiawan (2006) tentang Pengaruh Pemasangan Stabiliser Terhadap Kestabilan Guling Kendaraan Jalan 4 Roda, disimpulkan pemasangan stabiliser berefek pada sudut rolling kendaraan dimana sudut rolling menjadi lebih kecil berbeda ketika kendaraan sebelum di pasang satbiliser. Pada kemiringan kendaraan efek dari pemasangan stabiliser yaitu mengurangi kemiringan body kendaraan yang berakibat kendaraan tersebut seimbang.

Wicaksono, dkk (2017) tentang Simulasi Pendeteksi Ambang Batas Getaran Vertikal Berbasis Mikrokontroler sebagai Indikator Kenyamanan dan Keamanan Pengendara pada Kendaraan Bermotor, disimpulkan rancang bangun yang memanfaatkan software simulasi elektronika mampu bekerja ketika suatu terdapat getaran melampaui standar yang berpengaruh terhadap kondisi kesehatan penumpang. Sensor yang digunakan yaitu sensor getaran selanjutnya terdapat LCD sebagai sistem informasi ketika terjadi getaran yang melampaui kriteria kewajaran maka diatur otomatis oleh sebuah *mikroontroler Atmega16*. Pemakaian software *ISIS Profesional 7* bekerja secara baik dalam mendeteksi getaran sehingga kedepannya dapat digunakan sebagai piranti keselamatan dalam berkendara.

2.2 Dasar Teori

2.2.1 Kinematika

Kinematika adalah ilmu yang mempelajari tentang kecepatan, percepatan dan lintasan dari suatu mesin atau suatu mekanisme yang tidak mempertimbangkan penyebabnya (Syam, 2015). Pengertian lain dari kinematika ilmu yang merupakan bagian dari mekanika berkaitan studi tentang gerak suatu benda atau partikel tanpa disertai penyebab geraknya. Jika terdapat suatu gaya yang diperhatikan maka yang dipelajari bagian dari dinamika. Secara gampang dijabarkan, kinematika yaitu sebuah benda yang bergerak yang memiliki lintasan, kecepatan dan percepatan tanpa disertai penyebabnya. Sedangkan dinamika suatu benda yang bergerak memiliki gaya atau studi tentang gerak suatu benda atau partikel dengan disertai penyebab geraknya.

Menurut Syam (2015) terdapat beberapa istilah dalam konsep kinematika, yaitu :

1. Mekanisme

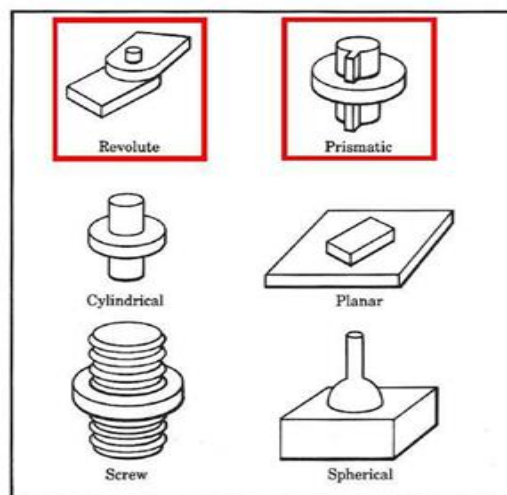
Mekanisme adalah suatu rangkaian batang penghubung (*linkage*) yang jika salah satu batang tetap ditahan dan batang lainnya digerakkan, maka gerakan batang yang lainnya dapat diperkirakan. Mekanisme sendiri terdiri dari rangkaian batang dan terdapat adanya suatu gerakan.

2. Rantai Kinematis

Rantai Kinematis atau *Kinematic Chain* yaitu, gabungan antara batang penghubung yang saling bersentuhan yang mengakibatkan gerakan sama satu dengan yang lain.

3. Pasangan

Pasangan atau *pair* adalah saling berhubungannya antara dua permukaan suatu komponen. Pada gambar 2.1 dibawah menunjukkan macam-macam pasangan.



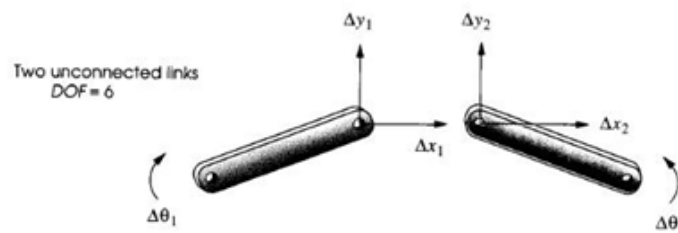
Gambar 2.1 Macam-macam pasangan atau *pair*

(Sumber : Syam, 2015)

4. Derajat Kebebasan

Derajat kebebasan atau *Degree Of Freedom* (DOF) adalah derajat independensi yang diperlukan untuk menyatakan posisi suatu sistem pada

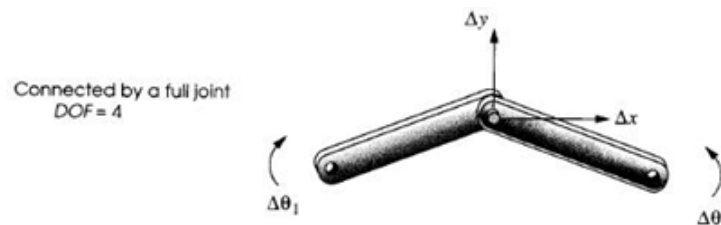
setiap saat. Suatu angka yang memberikan jumlah masukan untuk suatu mekanisme dalam melakukan suatu gerakan disebut juga derajat kebebasan. Sebagai contoh: jika *link* pada robot memiliki 3 DOF, terdapat juga *link* yang tidak terhubung ke *link* sama memiliki 3 DOF. Jika terdapat 2 *link* maka keseluruhan menjadi 6 DOF seperti ditunjukkan oleh gambar 2.2 dibawah ini.



Gambar 2.2 Terdapat 2 *link* yang tidak terhubung

(Sumber : Syam, 2015)

Setelah ke-2 link dihubungkan maka, Y_1 dan $Y_2 = Y$ kemudian X_1 dan $X_2 = X$. Setelah penggabungan tersebut maka tersisa 4 DOF, karena menghilangkan 2 DOF seperti ditunjukkan pada gambar 2.3 dibawah ini.



Gambar 2.3 Terdapat 2 *link* yang terhubung

(Sumber : Syam, 2015)

2.2.2 Percepatan

Percepatan adalah perubahan kecepatan persatuan waktu atau terhadap fungsi waktu. Percepatan merupakan hasil turunan pertama dari kecepatan dan merupakan turunan kedua dari perpindahan. Percepatan disimbolkan dengan (a), percepatan juga merupakan besaran vektor.

Percepatan rata-rata yaitu hasil bagi antara perubahan kecepatan (Δv) dengan selang waktu (Δt). Secara matematis dirumuskan sebagai berikut :

$$\bar{a} = \frac{\Delta v}{\Delta t} = \frac{v_2 - v_1}{t_2 - t_1} \dots\dots\dots 2.1$$

Keterangan :

- \bar{a} = percepatan rata-rata (m/s²)
- Δv = perubahan kecepatan (m/s)
- v_1 = kecepatan awal (m/s)
- v_2 = kecepatan akhir

Percepatan sesaat yaitu perubahan kecepatan dalam selang waktu yang singkat. Secara matematis dirumuskan sebagai berikut :

$$a = \lim_{\Delta t \rightarrow 0} \frac{\Delta v}{\Delta t} \dots\dots\dots 2.2$$

Keterangan :

- a = percepatan (m/s²)
- Δv = perubahan kecepatan (m/s)
- Δt = selang waktu (s)

2.2.3 Kecepatan

Kecepatan adalah perpindahan yang dialami objek persatuan waktu. Kecepatan merupakan besaran vektor yang disimbolkan (v). Kecepatan dibagi menjadi dua yaitu kecepatan rata-rata dan kecepatan sesaat.

Kecepatan rata-rata adalah perpindahan suatu benda selama selang waktu. Kecepatan rata-rata merupakan besaran vektor dengan arahnya sama dengan arah vektor perpindahannya. Secara matematis dirumuskan sebagai berikut :

$$\bar{v} = \frac{\Delta r}{\Delta t} = \frac{r_2 - r_1}{t_2 - t_1} \dots\dots\dots 2.3$$

Keterangan :

- \bar{v} = kecepatan rata-rata (m/s)
- Δr = perpindahan (m)
- Δt = selang waktu (s)

Kecepatan sesaat adalah kecepatan rata-rata pada limit selang waktu mendekati nol. Secara matematis dirumuskan sebagai berikut :

$$v = \lim_{\Delta t \rightarrow 0} \frac{\Delta r}{\Delta t} = \frac{dr}{dt} \dots\dots\dots 2.4$$

Keterangan :

- v = kecepatan sesaat (m/s)
- Δr = perpindahan (m)
- Δt = selang waktu (s)

2.2.4 Rotasi

Rotasi atau perputaran adalah transformasi dengan memutar sebuah titik terhadap titik pusat perputaran. Rotasi (perputaran) merupakan suatu transformasi yang memindahkan titik ke himpunan titik lainnya dengan cara memutar (Septiana, 2015). Rotasi (perputaran) ditentukan oleh 3 hal yaitu :

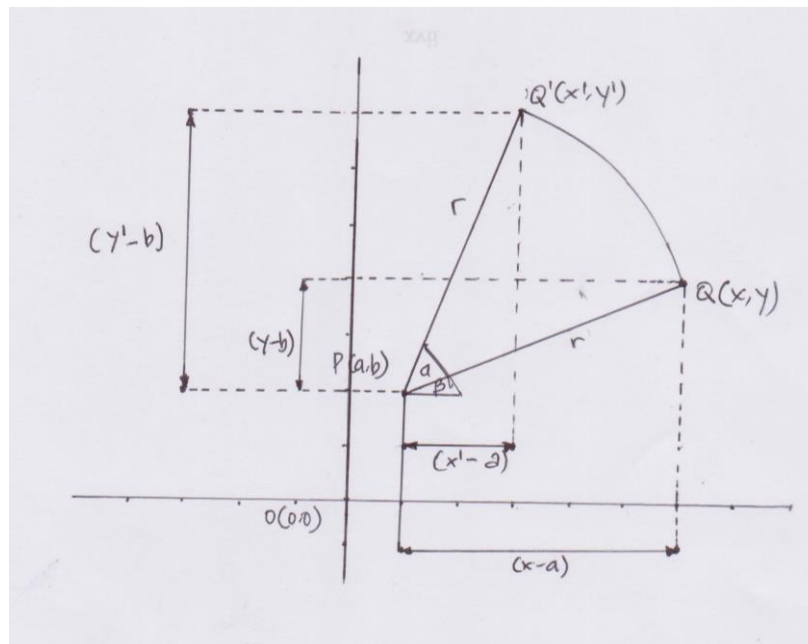
1. Titik pusat putaran.
2. Besar sudut putaran
3. Arah putaran

Pada arah rotasi dibedakan menjadi dua yaitu : arah rotasi positif (berlawanan dengan arah jarum jam) dan arah rotasi negative (searah dengan jarum jam).



Gambar 2.4 Gerak rotasi

Sumber : (API, 2018)



Gambar 2.5 Grafik rotasi

Sumber : (API, 2018)

Dari gambar 2.5 diatas jika titik $Q(x, y)$ di putar oleh sudut α maka menjadi titik $Q^1(x^1, y^1)$. Jika dilihat pada garis PQ maka $(x - a) = r \cos \beta$ dan $(y - b) = r \sin \beta$.

Untuk PQ^1 $(x - a) = r \cos \beta$

$$(x^1 - a) = r \cos(\alpha + \beta)$$

$$(x^1 - a) = r(\cos \beta \cos \alpha - \sin \beta \sin \alpha)$$

$$(x^1 - a) = r \cos \beta \cos \alpha - r \sin \beta \sin \alpha$$

$$(x^1 - a) = (x - a) \cos \alpha - (y - b) \sin \alpha$$

Untuk PQ^1 $(y - b) = r \sin \beta$

$$(y^1 - b) = r \sin(\alpha + \beta)$$

$$(y^1 - b) = r(\sin \beta \cos \alpha + \cos \beta \sin \alpha)$$

$$(y^1 - b) = r \sin \beta \cos \alpha + r \cos \beta \sin \alpha$$

$$(y^1 - b) = (y - b) \cos \alpha + (x - a) \sin \alpha$$

$$(y^1 - b) = (x - a) \sin \alpha + (y - b) \cos \alpha$$

Dengan metode aljabar matrik maka ditulis sebagai berikut;

$$\begin{pmatrix} x^1 - a \\ y^1 - a \end{pmatrix} = \begin{pmatrix} [x - a]\cos\alpha - [y - b]\sin\alpha \\ [x - a]\sin\alpha + [y - b]\cos\alpha \end{pmatrix}$$

$$\begin{pmatrix} x^1 - a \\ y^1 - a \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x - a \\ y - b \end{pmatrix}$$

Hasil titik (x, y) dirotasi dengan (a, b) dengan sudut α berlawanan arah jarum jam adalah :

$$\begin{pmatrix} x^1 - a \\ y^1 - a \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x - a \\ y - b \end{pmatrix} \text{ atau}$$

$$\begin{pmatrix} x^1 \\ y^1 \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x - a \\ y - b \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix} \dots\dots\dots 2.5$$

Jika rotasi dengan pusat $O(0, 0)$ maka didapat :

$$\begin{pmatrix} x^1 - a \\ y^1 - b \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x - a \\ y - b \end{pmatrix}$$

$$\begin{pmatrix} x^1 - 0 \\ y^1 - 0 \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x - 0 \\ y - 0 \end{pmatrix}$$

$$\begin{pmatrix} x^1 \\ y^1 \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Hasil titik (x, y) dirotasi dengan $(0, 0)$ dengan sudut α berlawanan arah jarum jam adalah :

$$\begin{pmatrix} x^1 \\ y^1 \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \dots\dots\dots 2.6$$

Hubungan dari gerak translasi (gerak lurus) dan gerak rotasi (gerak melingkar). Sebagai pembeda antara gerak translasi dan gerak rotasi yaitu jika pada gerak rotasi terdapat gaya sentripetal. Gaya sentripetal adalah gaya yang membuat benda bergerak melingkar. Kedudukan benda ditentukan dari sudut θ dan r (jari-jari lintasan). Jarak tempuh linear dan sudut tempuh memiliki hubungan yaitu $s = \theta r$. Sedangkan hubungan antara kecepatan sudut dan kecepatan linear adalah

$$v = \frac{x}{t}$$

$$v = \frac{\theta r}{t}$$

$$v = \omega \cdot r$$

$$v = r \cdot \omega \quad \dots\dots\dots 2.7$$

$$a = \frac{v}{t}$$

$$a = \frac{\omega r}{t}$$

$$a = a \cdot r \quad \dots\dots\dots 2.8$$

Benda bergerak translasi dengan kecepatan v berotasi dengan kecepatan sudut ω maka energy kinetic benda yang menggelinding dirumuskan $E_k = E_k \text{ translasi} + E_k \text{ rotasi} = \frac{1}{2}mv^2 + \frac{1}{2}I\omega^2$.

Tabel 2.1 Hubungan gerak translasi dan gerak rotasi

Translasi		Rotasi		Hubungan
Jarak	s	Sudut	θ	$s = \theta \cdot r$
Kecepatan	v	Kecepatan	ω	$v = \omega \cdot r$
Percepatan	a	Percepatan	a	$a = a \cdot r$

2.2.5 Mikrokontroler Arduino

Mikrokontroler merupakan suatu sistem komputer bersifat fungsional yang terdapat didalam suatu chip. Arduino merupakan sebuah platform prototipe yang bersifat *open source*, berdasarkan pada perangkat keras dan lunak yang bersifat fleksibel serta mudah digunakan. Arduino bukan hanya sekedar alat pengembangan lebih kepada sebuah alat kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment (IDE)* yang canggih. Sebuah software yang sangat berperan dalam suatu penulisan program, meng-*compile* menjadi sebuah

kode biner yang kemudian mengunggah ke memori mikrokontroler dinamakan IDE. Terdapat banyak proyek atau pengembangan yang dilakukan oleh akademisi menggunakan arduino ini, juga terdapat banyak modul-modul pendukung lainnya agar dapat dihubungkan atau dikombinasikan dengan arduino (Djuandi, 2011). Yang membuat arduino digemari banyak orang karena sifat *open source*, baik untuk *hardware* dan *software*. Didalam papan arduino terdapat komponen utama yaitu mikrokontroler 8 bit bernama Atmega yang diproduksi oleh Atmel Corporation. Tipe Atmega yang digunakan pada setiap papan arduino berbeda-beda tergantung pada spesifikasi dari arduino tersebut, contoh pada arduino uno memakai Atmega328 tetapi pada arduino mega 2560 R3 menggunakan Atmega2560 yang lebih kompleks (Utami, 2015). Arduino sangat diperuntukan bagi para seniman, desainer, dan siapapun yang tertarik untuk belajar didalamnya. Ada beberapa faktor yang membuat arduino dengan cepat diterima oleh orang-orang yaitu: harganya yang murah, arduino sangat mudah digunakan dan dipelajari, sistemnya yang terbuka, dan software arduino dapat dijalankan di beberapa sistem operasi.

2.2.6 Jenis – jenis Arduino

Saat ini terdapat bermacam – macam jenis papan Arduino yang disesuaikan dengan peruntukannya, seperti di bawah ini :

2.2.6.1 Arduino USB

Arduino jenis ini menggunakan jenis port USB sebagai penghubung pemrograman atau komunikasi komputer. Karena menggunakan port USB sebagai penghubung, jenis arduino ini dapat digunakan secara pasang dan pakai. Arduino tipe USB ditunjukkan oleh gambar 2.6 dibawah ini (Ahlina, 2015).



Gambar 2.6 Board Arduino USB

(Sumber: Ahlina, 2015)

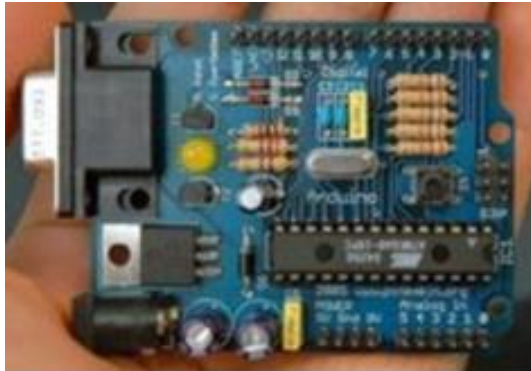
Jenis-jenis dari arduino USB antara lain :

1. Arduino Uno
2. Arduino Duemilanove
3. Arduno Leonardo
4. Arduino Mega2560
5. Arduino Pro Micro AT
6. Arduino Nano R3
7. Arduino mini Atmega

2.2.6.2 Arduino Serial

Menggunakan RS232 sebagai antar muka pemrograman atau komunikasi komputer. Contohnya adalah arduino serial dan arduino serial v2.0.

Arduino serial ditunjukkan pada gambar 2.7.



Gambar 2.7 Arduino Serial

(Sumber : Djuandi, 2011)

2.2.6.3 Arduino Fio

Arduino Fio ini menggunakan ATmega328P sebagai basis kontrolernya. Berfungsi untuk penggunaan nirkabel. Arduino Fio ditunjukkan oleh gambar 2.8 dibawah ini.



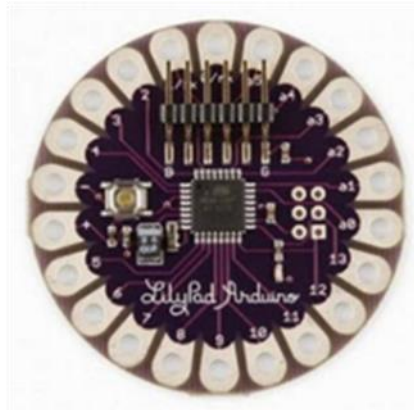
Gambar 2.8 Arduino Fio

(Sumber : (Djuandi, 2011)

2.2.6.4 Arduino Lylipad

Arduino Lylipad memiliki bentuk yang melingkar. Memiliki 14 pin I/O digital, dan 6 pin input analognya.

Arduino lylipad ditunjukkan pada gambar 2.9 dibawah ini.



Gambar 2.9 Arduino LilyPad

(Sumber : Djuandi, 2011)

2.2.6.5 Arduino BT

Arduino jenis ini terpasang modul Bluetooth untuk komunikasi nirkabel (Wireless).

Arduino BT ditunjukkan pada gambar 2.10 berikut ini.



Gambar 2.10 Arduino BT

(Sumber : Djuandi, 2011)

2.2.6.6 Arduino Nano dan Mini

Arduino jenis ini berbentuk papan *compact* dan digunakan bersama *breadboard*. Contoh: Arduino Nano 3.0, Arduino Nano 2.x, Arduino Mini 04, Arduino Mini 03, Arduino Stamp 02.

Arduino Nano dan Mini ditunjukkan pada gambar 2.11 dibawah ini.



Gambar 2.11 Arduino Nano dan Mini

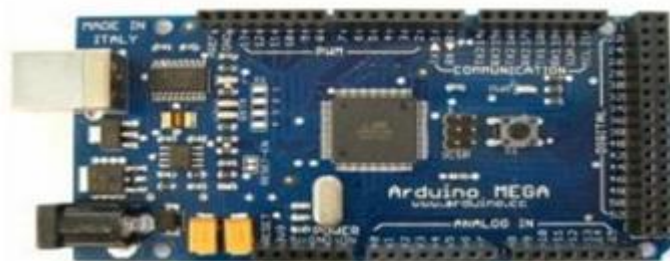
(Sumber : Djuandi, 2011)

2.2.6.7 Arduino Mega

Arduino mega memiliki spesifikasi yang lebih canggih ketimbang arduino jenis lainnya, dengan adanya tambahan pin digital, pin analog, port serial dan lain-lain. Jenis arduino mega yaitu :

- Arduino Mega
- Arduino Mega 2560

Arduino mega ditunjukkan pada gambar 2.12 dibawah ini.



Gambar 2.12 Arduino Mega

(Sumber : Djuandi, 2011)

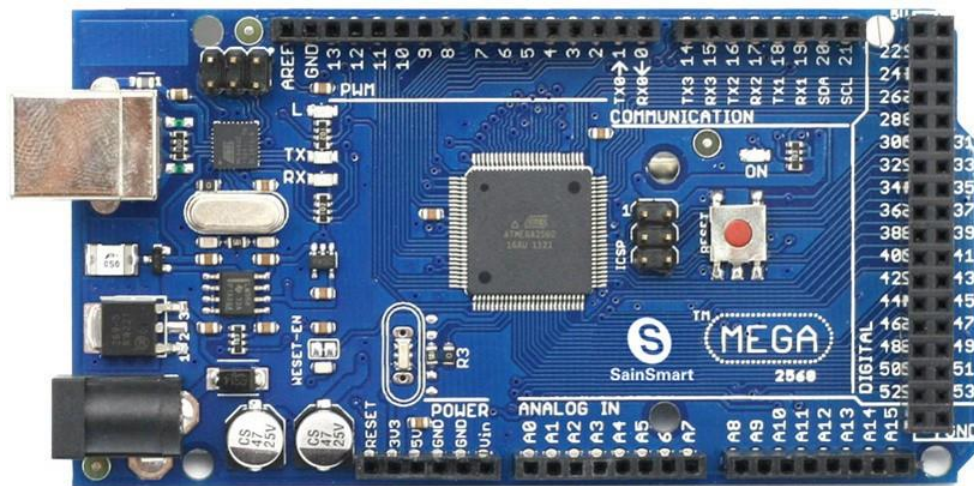
2.2.7 Arduino Mega 2560

2.2.7.1 Pengenalan Arduino Mega 2560

Arduino Mega 2560 merupakan pengembangan dari papan Arduino Mega sebelumnya. Pada awalnya Arduino Mega menggunakan chip Atmega1280 yang

kemudian diubah menjadi chip Atmega2560, karena penggantian nama tersebut maka sekarang lebih dikenal dengan nama Arduino Mega 2560. Sampai saat ini Arduino Mega 2560 telah sampai pada revisi yang ke 3 (R3). Terdapat pula perbedaan lainnya selain dari chip ATmega yang di gunakan, yaitu sudah tidak lagi menggunakan chip FTDI sebagai fungsi USB to Serial Converter, tetapi menggunakan chip ATmega16u2 pada revisi ke 3 (R3), sedangkan pada revisi 1 dan 2 di gunakan chip ATmega8u2 sebagai fungsi USB to Serial Converter.

Arduino Mega 2560 merupakan papan mikrokontroler yang berbasis ATmega 2560 dimana memiliki 54 pin digital input / output (15 pin diantaranya adalah PWM), 16 pin analog input, 4 pin UART (serial port hardware). Arduino Mega 2560 juga di lengkapi oscillator 16 Mhz, sebuah port USB, power jack DC, ICSP header, dan tombol reset. Itu semua dibutuhkan untuk mendukung mikrikontroler, untuk mulai mengktifkan cukup dengan menghubungkan power dari USB ke computer atau dengan adaptor AC – DC ke jack DC. Arduino Mega 2560 juga kompatibel dengan sebagian besar *shield* yang di rancang untuk Arduino *Deumilanove* atau *Diecimila* (Utami, 2015). Arduino Mega 2560 ditunjukkan pada gambar 2.13 dibawah ini.



Gambar 2.13 Board Arduino Mega 2560

Sumber : (Arduino, 2018)

2.2.7.2 Spesifikasi Arduino Mega 2560

Spesifikasi dari Arduino Mega 2560 dapat di lihat di tabel 2.2 dibawah ini.

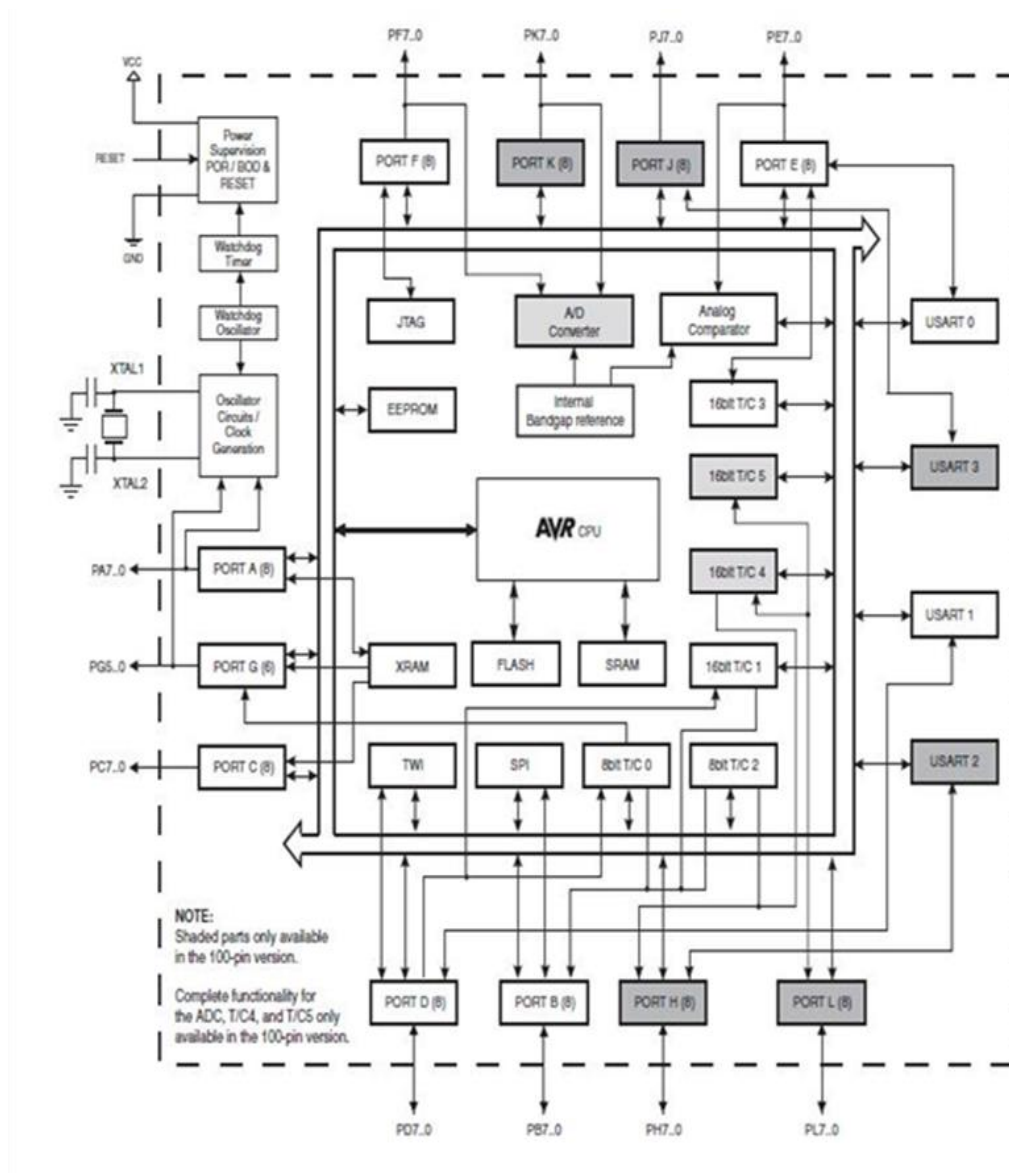
Tabel 2.2 Spesifikasi Arduino Mega 2560

Mikrokontroler	ATmega2560
Tegangan operasi	5V
Tegangan input (di sarankan)	7V – 12 V
Tegangan input (limit)	6V – 20 V
Digital pin I/O	54 buah (15 diantaranya menyediakan PWM output)
Analog pin I/O	16 buah
Arus DC per pin I/O	20 mA
Arus DC pin 3.3V	50 mA
Memori Flash	256 KB, 8KB digunakan untuk bootloader
SRAM	8 KB
EEPROM	4 KB
Waktu Kecepatan	16 Mhz
LED_BUILTIN	13
Panjang	101.52 mm
Lebar	53.3 mm
Berat	37 g

Sumber : (Arduino, 2018)

2.2.7.3 Blok Diagram Arduino Mega 2560

Blok Diagram dari Arduino Mega 2560 dapat di lihat pada gambar 2.14 di bawah ini.

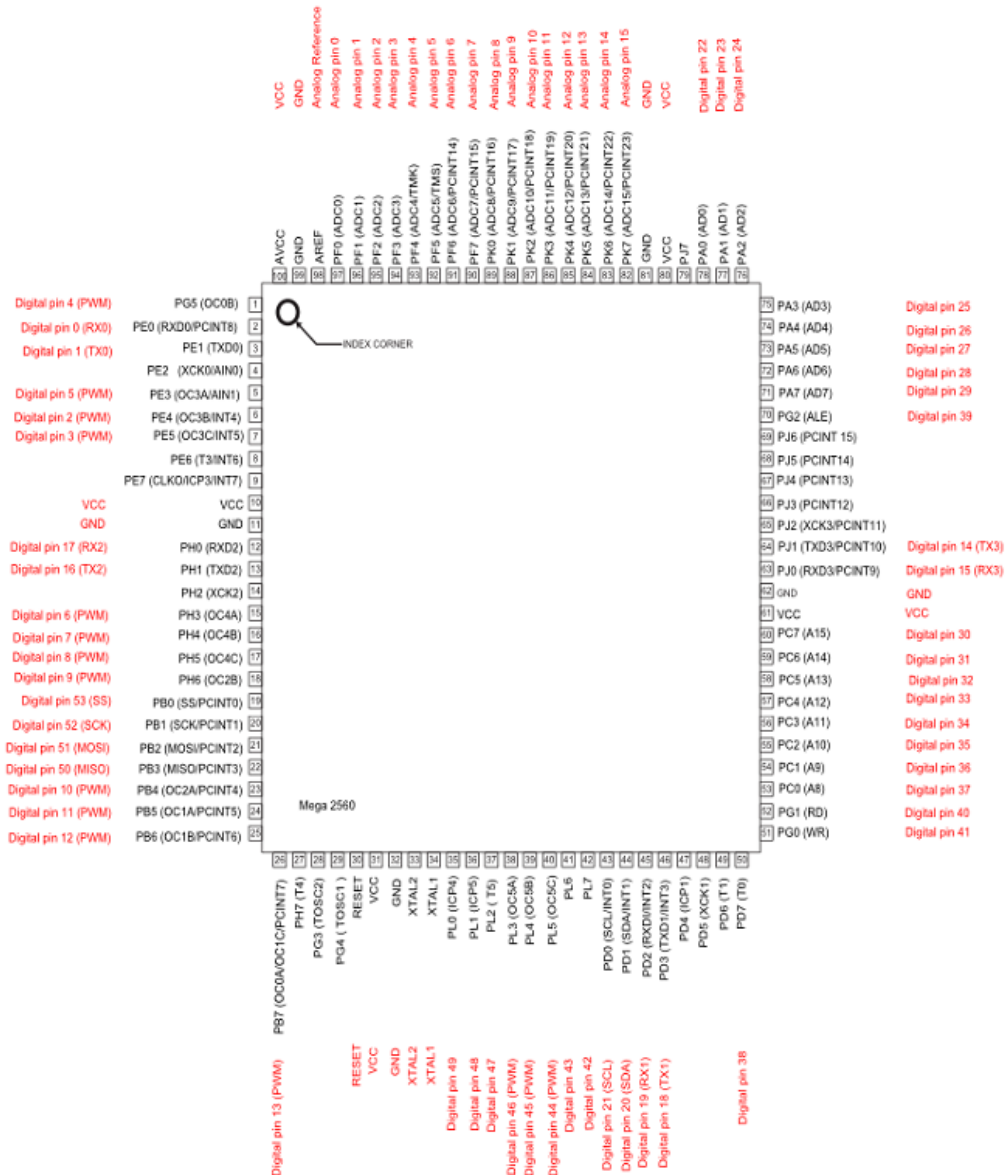


Gambar 2.14 Blok Diagram Arduino Mega 2560

(Sumber : Utami, 2015)

2.2.7.4 Konfigurasi Pin Arduino Mega 2560

Konfigurasi dari Pin Arduino Mega 2560 dapat dilihat pada gambar 2.15 di bawah ini.



Gambar 2.15 Konfigurasi Pin Arduino Mega 2560

(Sumber : Utami, 2015)

Beberapa pin tegangan yang terdapat pada papan Arduino Mega 2560 sebagai berikut :

- a. 5V : adalah pin output yang mengalirkan tegangan 5Volt yang telah ter-regulator, pada pin ini tegangan telah di atur dari regulator yang terdapat pada papan. Papan Arduino dapat rusak jika memberikan tegangan langsung tanpa melewati regulator melalui pin 5V dan 3,3V.

- b. GND : adalah pin *Ground (Massa)*.
- c. VIN : adalah pin yang digunakan untuk memberikan tegangan ke papan Arduino dengan tegangan yang disarankan.
- d. 3V3 : adalah sebuah pin output yang menghasilkan tegangan 3,3 V yang dihasilkan oleh regulator pada papan. Arus maksimum yang dihasilkan adalah 50 mA.
- e. IOREF : adalah pin yang memberikan referensi tegangan mikrokontroler. Sebuah *shield* dikonfigurasi dengan benar agar dapat membaca pin tegangan IOREF dan memilih sumber daya yang tepat atau mengaktifkan penerjemah tegangan (*voltage translator*) pada *output* untuk bekerja pada tegangan 5 Volt atau 3,3 Volt (Fernando, 2017).

Menurut Akhmadi, dkk (2014) beberapa pin memiliki fungsi spesialis tersendiri, sebagai berikut :

- a. Serial, terdapat 4 serial yang terdiri dari 2 pin. Serial 0 : pin 0 (RX) dan pin 1 (TX). Serial 1 : pin 19 (RX) dan pin 18 (TX). Serial 2 : pin 17 (RX) dan pin 16 (TX). Serial 3 : pin 15 (RX) dan pin 14 (TX). RX pakai sebagai penerima dan TX untuk transmit data serial TTL. Pin 0 dan pin 1 merupakan pin yang terhubung oleh chip USB-to-Serial TTL ATmega16U2.
- b. External Interrupts, adalah pin 2 (interrupt 0), pin 3 (interrupt 1), pin 18 (interrupt 5), pin 19 (interrupt 4), pin 20 (interrupt 3), dan pin 21 (interrupt 2). Dengan total memiliki 6 buah interrupt. Agar interrupt terpicu pada nilai rendah, tinggi atau perubahan nilai maka pin ini harus dikonfigurasi terlebih dahulu.
- c. PWM: Pin 2 sampai 13 dan pin 44 sampai 46, yang menyediakan output PWM sebesar 8-bit dengan menggunakan analogWrite.
- d. SPI : Pin 50 (MISO), pin 51 (MOSI), pin 52 (SCK), dan pin 53 (SS) untuk mendukung komunikasi SPI dengan menggunakan SPI Library.
- e. LED : Pin 13. Pada pin 13 terhubung dengan built-in LED yang terkoneksi pada digital pin 13. Ketika LED menyala (ON) berarti pin ter-set HIGH sedangkan ketika LED mati (OFF) maka pin ter-set LOW.

- f. TWI : Pin 20 (SDA) dan pin 21 (SCL) mendukung komunikasi TWI dengan memanfaatkan Wire Library.

Arduino Mega 2560 mempunyai 16 pin analog input, masing-masing pin analog input menyediakan resolusi 10 bit (memiliki 1024 nilai yang berbeda). Secara default pin-pin ini diukur dari Ground sampai dengan 5 Volt, namun dapat mengubah titik jangkauan menggunakan pin AREF dan fungsi *Analog Reference* (Akhmadi dkk, 2014).

Ada beberapa pin lain yang masih tersedia di papan ini, yaitu :

- a. AREF: adalah pin referensi tegangan input analog A/D Converter (Utami, 2015).
- b. RESET: adalah jalur LOW yang digunakan untuk menghidupkan ulang mikrokontroler (Fernando, 2017).

2.2.7.5 Proteksi atau Perlindungan Beban Berlebih pada Arduino Mega 2560

Arduino Mega 2560 dilengkapi polyfuse yang dapat direset guna melindungi port USB komputer atau laptop dari hubungan singkat dan arus lebih. Pada komputer atau laptop sebenarnya sudah memiliki suatu perlindungan internal untuk port USB di dalamnya dengan terdapat lapisan perlindungan tambahan pada sekering. Jadi secara otomatis sekering akan memutuskan sambungan sampai hubungan singkat dibuang, apabila terjadi arus yang dihubungkan ke port USB lebih dari 500 mA dan akan menyambung kembali jika batasan aman telah kembali (Fernando, 2017).

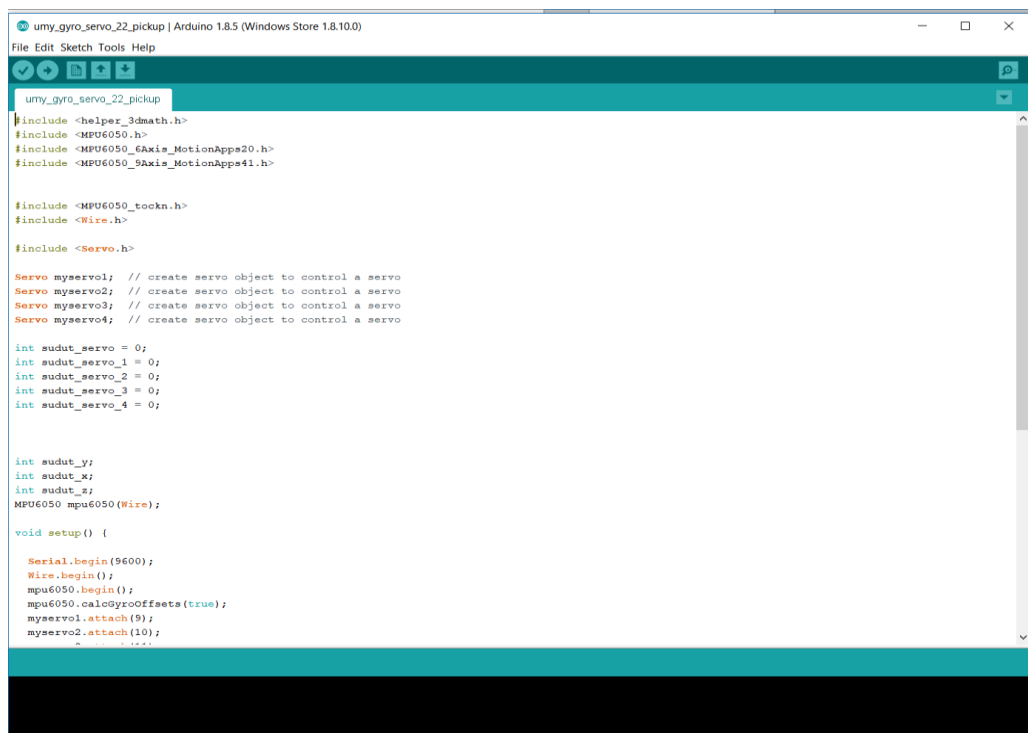
2.2.8 IDE Arduino

Integrated Development Environment (IDE) Arduino adalah aplikasi komplit yang berguna dalam pemrograman arduino meliputi editor, compiler, dan pengunggahan. Semua itu dapat menggunakan semua seri modul keluarga arduino, kecuali papan arduino yang menggunakan mikrokontroler selain seri AVR.

Menurut Utami (2015) *software* terbaik dan terancang yang ditulis dengan memakai java yaitu IDE Arduino. IDE Arduino yang dimaksud yaitu:

- Editor program, merupakan jendela yang berguna dalam menulis serta mengedit program dengan bahasa *Processing*.
- *Compiler*, merupakan mikrokontroler tidak dapat untuk memahami bahasa *processing* dikarenakan mikrokontroler hanya bisa memahami kode biner. Oleh karena itu dibutuhkan *compiler* sebagai pengubah bahasa *processing* menjadi kode biner.
- *Uploader*, sebuah modul yang berisikan kode biner dari komputer atau laptop untuk ditransfer ke dalam penyimpanan pada papan arduino.

Untuk tampilan dari IDE Arduino dapat dilihat pada gambar 2.16 dibawah ini.



```

umy_gyro_servo_22_pickup
#include <helper_3dmath.h>
#include <MPU6050.h>
#include <MPU6050_6Axis_MotionApps20.h>
#include <MPU6050_9Axis_MotionApps41.h>

#include <MPU6050_tockn.h>
#include <Wire.h>

#include <Servo.h>

Servo myservo1; // create servo object to control a servo
Servo myservo2; // create servo object to control a servo
Servo myservo3; // create servo object to control a servo
Servo myservo4; // create servo object to control a servo

int sudut_servo = 0;
int sudut_servo_1 = 0;
int sudut_servo_2 = 0;
int sudut_servo_3 = 0;
int sudut_servo_4 = 0;

int sudut_y;
int sudut_x;
int sudut_z;
MPU6050 mpu6050(Wire);

void setup() {
  Serial.begin(9600);
  Wire.begin();
  mpu6050.begin();
  mpu6050.calcGyroOffsets(true);
  myservo1.attach(9);
  myservo2.attach(10);
  ...
}

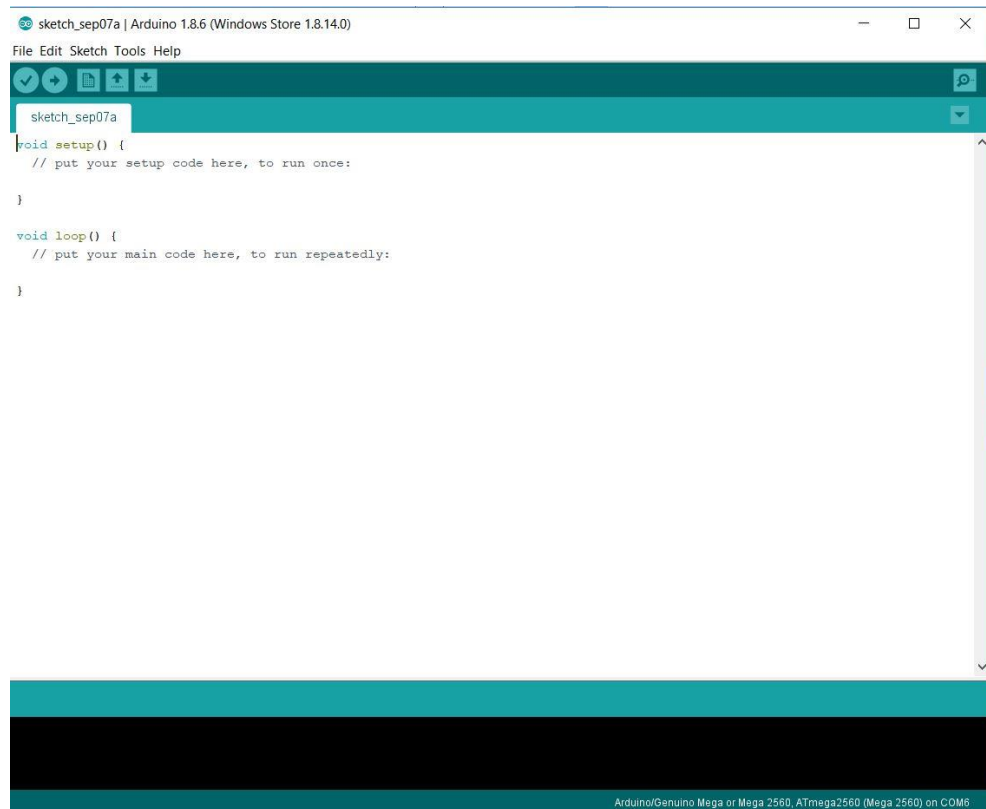
```

Gambar 2.16 Tampilan IDE Arduino

Pada umumnya kode arduino di sebut dengan istilah sketch. Dalam penggunaannya sketch ini sering digunakan bergantian dengan kode program, namun dengan demikian tidak mengubah makna atau arti karena memiliki arti yang sama (Djuandi, 2011).

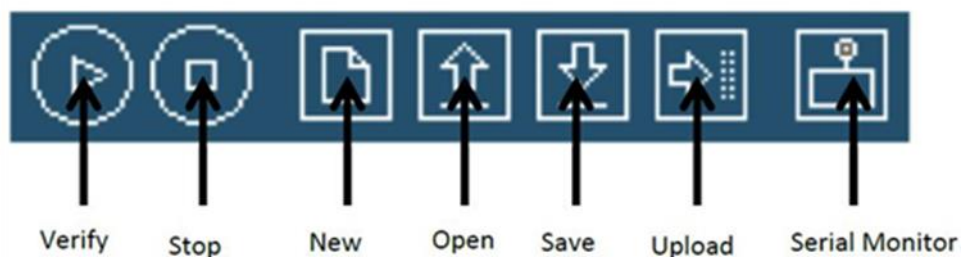
2.2.8.1 Bagian – bagian IDE Arduino

Pada gambar 2.17 dan 2.18 dibawah ini terlihat bagian – bagian dari IDE Arduino.



Gambar 2.17 Tampilan toolbar IDE Arduino

(Sumber : Akhmadi, 2014)



Gambar 2.18 Tampilan toolbar IDE Arduino

(Sumber : Utami, 2015)

Pada gambar 2.17 terdapat sebuah area kolom berwarna putih yang berfungsi untuk menulis program yang akan dibuat. Sedangkan dibagian bawah terdapat kolom berwarna hitam yang berfungsi untuk menampilkan pesan hasil

pemrograman yang berhasil atau tidak berhasil (error), kolom berwarna hitam tersebut dinamakan progress area (Akhmadi dkk, 2014).

Menurut Utami (2015) pada gambar 2.18 merupakan toolbar IDE Arduino yang mempunyai fungsi masing-masing, sebagai berikut :

- Verify : merupakan pemeriksaan apakah program yang telah dibuat sudah sesuai atau pengecekan kebenaran atau kesalahan program yang dibuat.
- Stop : digunakan untuk menghentikan berjalannya serial monitor.
- New : digunakan untuk membuka jendela pemrograman baru dalam keadaan kosong.
- Open : digunakan untuk membuka file pemrograman yang telah tersimpan sebelumnya.
- Save : digunakan untuk menyimpan data pemrograman yang sedang dibuat.
- Upload : digunakan untuk mentransfer data pemrograman yang sebelumnya telah terverify ke papan arduino.
- Serial Monitor : digunakan untuk membuka jendela serial monitor agar dapat melihat data yang dikirimkan oleh arduino dan juga untuk mengirim data kembali.

2.2.8.2 Komplikasi dan Pengunggahan Program

Sebelum melakukan proses komplikasi dan pengunggahan ada yang harus di perhatikan terlebih dahulu ketika mengkonfigurasi dalam IDE, pertama adalah jenis arduino yang digunakan, kedua terhubungnya port serial ke arduino. Dalam mengidentifikasi jenis arduino itu sangatlah mudah, karena tertulis di dalam board arduino. Kita juga dapat melihat jenis mikrokontroler Atmega yang digunakan dan dapat melihatnya di dalam board arduino tersebut. Sebagai contoh pada arduino uno menggunakan Atmega328, arduino nano r3 menggunakan Atmega168 dan arduino mega 2560 menggunakan Atmega2560 sebagai mikrikontrolernya (Utami, 2015).

Setelah melakukan langkah tersebut selanjutnya memilih *port* komunikasi serial mana yang terhubung ke arduino, *port* USB harus sesuai dengan settingan pada sketch, dengan pilih tools kemudian serial *port*. Jika sudah lanjut ke klik ke

verify (logo centang), setelah tidak ada keterangan eror kemudian klik upload (sebelah logo verify), pada saat upload lampu TX dan RX pada board arduino akan berkedip, jika terdapat keterangan *done uploading* berarti upload berhasil. Perlu diingat jika dalam proses verify terdapat keterangan pesan kesalahan biasanya disebabkan karena terjadi salah pengetikan dalam program atau salah menentukan jenis arduino yang digunakan.

2.2.9 Motor Servo

2.2.9.1 Pengenalan Motor Servo

Motor servo adalah jenis dari motor DC, motor servo adalah salah satu motor yang menggunakan sistem umpan balik tertutup untuk rangkaian kontrolnya yang terdapat didalam motor servo akan terinformasikan kembali menuju rangkaian kontrol oleh motor tersebut (Nasution, dkk 2015). Sedangkan menurut (Damayanti, 2016) motor servo adalah suatu perangkat motor yang mampu mengatur atau menentukan besarnya posisi sudut pada keluaran poros motor, dengan menggunakan sistem kontrol umpan close loop. Komponen yang menyusun motor servo antara lain, potensiometer, sebuah motor dc, rangkaian yang berupa gear dan kontrol. Potensiometer berguna pada motor servo sebagai penentu batas posisi putar pada motor (Nasution, dkk 2015).

Jika menginginkan posisi poros apakah telah sesuai yang diharapkan atau belum, penggunaan sistem kontrol close loop sangatlah membantu pada saat pengontrolan dari gerakan motor servo (Damayanti, 2016). Biasanya pada motor servo pergerakannya tidak dapat berkesinambungan seperti halnya motor DC atau motor stepper, karena pergerakan dari motor servo hanya bisa sudut tertentu. Meskipun begitu motor servo dapat dimodifikasi agar dapat bergerak berkesinambungan. Motor servo memiliki fungsi yang beragam seperti keperluan industri, penggerak dari robot (lengan robot, kaki robot dan yang lain) karena motor servo memiliki torsi yang besar sehingga dapat menopang beban yang berat. Kegunaan dari motor servo sangat beragam antara lain digunakan pada *Remote Control (RC)* yang meliputi mobil, motor, kapal, pesawat, dan digunakan pada

penggerak kamera atau kamera pengintai serta penggerak robot (Refliansyah, 2016).

Pada gambar 2.19 dibawah ini menunjukkan gambar dari motor servo standar.



Gambar 2.19 Motor Servo Standar Hitec HS – 322 HD

(Sumber : Sudarmanto, 2007)

Kelebihan dari pemakaian motor servo antara lain :

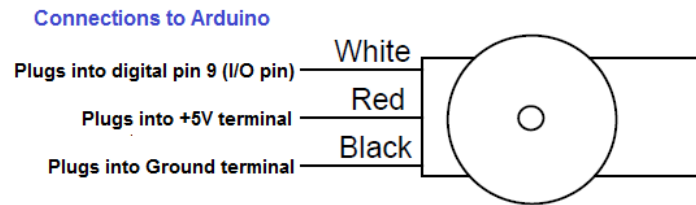
1. Tidak menimbulkan suara berisik pada saat digunakan dalam kecepatan tinggi.
2. Tidak bergetar pada saat digunakan.
3. Menghasilkan daya yang sepadan dengan berat dan ukuran motor.
4. Arus listrik yang digunakan sepadan dengan bebab yang diberikan.
5. Penggantian encoder yang digunakan dapat mempengaruhi resolusi dan akurasi.

2.2.9.2 Prinsip Kerja Motor Servo

Motor servo merupakan motor yang memiliki putaran lambat, yang ditunjukkan pada *rate* putarannya yang lambat, tetapi motor servo memiliki torsi yang kuat pada *internal gear*-nya.

Didalam motor servo terdapat 3 warna kabel yang memiliki fungsi masing-masing antara lain kabel putih merupakan I/O *pin*, VCC sendiri yaitu kabel merah dan untuk *ground* merupakan kabel hitam. Untuk pengontrolan dari motor servo yaitu melalui kabel putih atau I/O. Pada gambar 2.20 dibawah ini merupakan *pin out* kabeel servo yang duhubungkan ke rangkaian pengontrol.

Parallax Standard Servo Motor



Gambar 2.20 *Pin Out* Kabel Motor servo

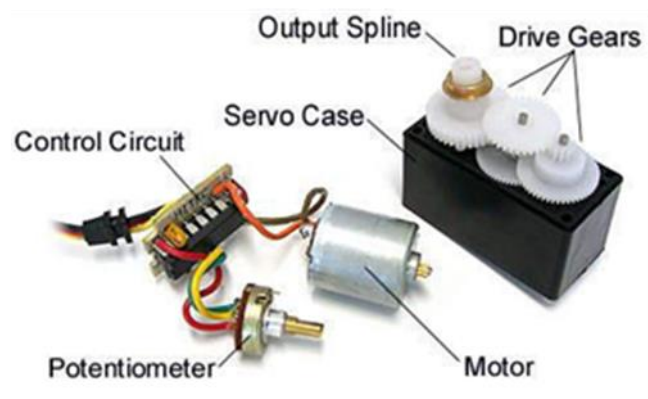
(Sumber : Refliansyah, 2016)

Karakteristik di dalam sebuah motor servo, yaitu :

1. Memiliki 3 jalur yaitu : *power*, *ground* dan *control*.
2. Posisi dikendalikan oleh sinyal *control*.
3. Di dalamnya terdapat konstruksi *internal gear*, *potensiometer* dan *feedback control*.
4. Pengaktifan dari motor servo dikendalikan oleh pulsa sebesar 20ms, dimana lebar pulsa antara 500 μ s dan 2400 μ s menyatakan akhir dari Range sudut maksimum.

Potensiometer yang terdapat di dalam motor servo berfungsi untuk sensor posisi, yang terhubung pada *output shaft* agar dapat mengetahui posisi sudut dari *output gear* pada motor servo. Saat motor berputar maka akan menggerakkan *output shaft* dan *potensiometer*. Agar dapat mengetahui posisi dari *actual shaft* maka rangkaian control akan membaca kondisi potensiometer tersebut. Motor akan berhenti jika posisi tersebut sudah sesuai dengan yang diinginkan (Ahlina, 2015).

Gambar 2.21 dibawah ini menunjukkan *internal gear* dan *control* elektronik untuk pengaturan gerakan dari motor.



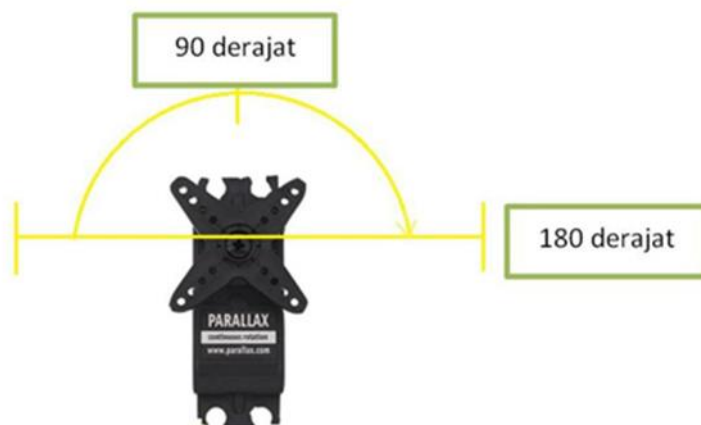
Gambar 2.21 Kontruksi Motor Servo

(Sumber : Refliansyah 2016)

2.2.9.3 Jenis – Jenis Motor Servo

1. Motor Servo Standar

Motor servo standar merupakan jenis motor servo yang putarannya hanya dapat 180° . Dengan defleksi masing-masing sudut 90° atau dengan kata lain hanya dapat bergerak 90° ke masing-masing arah (CW dan CCW) dan tidak dapat berputar satu putaran penuh (Damayanti, 2016). Motor servo jenis ini umumnya digunakan pada sistem robotika. Arah putar motor servo standar dapat dilihat pada gambar 2.22 dibawah ini.



Gambar 2.22 Arah putar motor servo standar.

(Sumber : Ahlina 2015)

2. Motor Servo Kontinyu

Motor servo kontinyu adalah jenis motor servo yang putarannya dapat mencapai 360° . Dengan kata lain servo ini dapat berputar tanpa adanya defeksi sudut putar atau dapat berputar terus menerus (Damayanti, 2016). Terdapat perbedaan lain untuk motor servo kontinyu yaitu dari prinsip kerja dimana nilai delay yang diberikan mempengaruhi untuk menggerakkan motor servo ke kiri atau ke kanan (Ahlina, 2015). Arah putar motor servo kontinyu dapat dilihat pada gambar 2.23 dibawah ini.

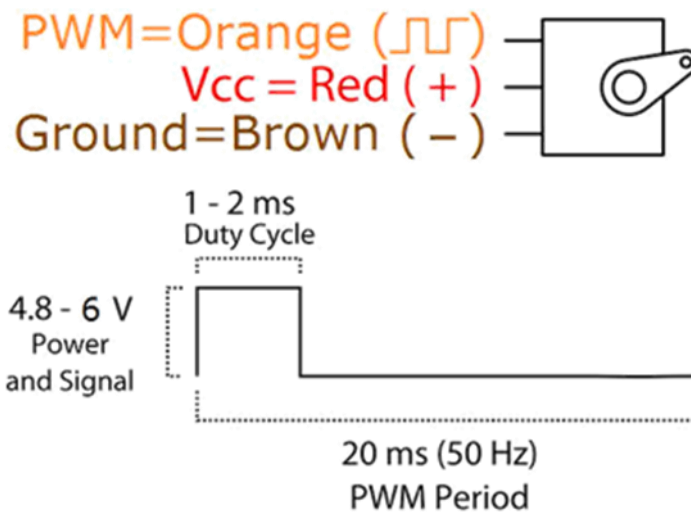


Gambar 2.23 Arah putar motor servo kontinyu

(Sumber : Ahlina 2015)

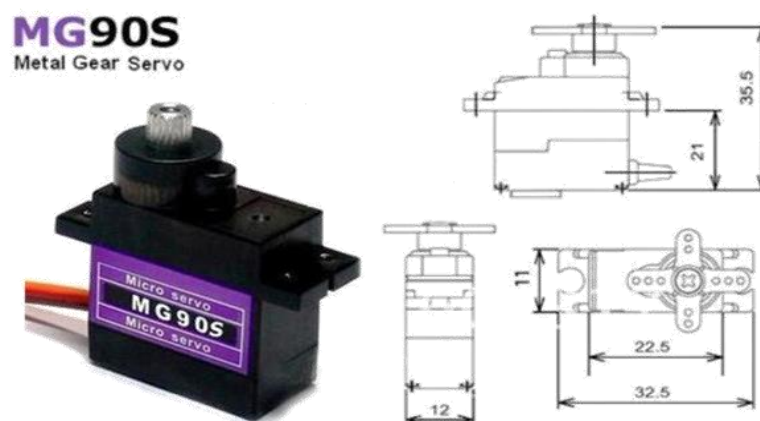
2.2.9.4 Karakteristik Motor Servo Tower Pro Micro Servo MG90S

Pada perancangan kali ini motor servo yang digunakan adalah jenis *Tower Pro Micro Servo MG90S*. Konfigurasi pin *Tower Pro Micro Servo MG90S* yaitu :
 1. Kabel merah = kutub positif pada baterai / *power rail*,
 2. Kabel coklat = kutub negatif pada baterai / *ground*,
 3. Kabel orange = sinyal masukan pengendali / *input control signal* (PWM).
 Pada gambar 2.24 dan 2.25 di bawah ini merupakan pin-pin pengkabelan dari motor servo yang dihubungkan pada rangkaian pengontrol dan menunjukkan bentuk dari *Tower Pro Micro Servo MG90S*.



Gambar 2.24 Konfigurasi pin kabel motor servo

(Sumber : datasheetspdf.com, 2018)



Gambar 2.25 Tower Pro Micro Servo MG90S

(Sumber : datasheetspdf.com, 2018)

Agar lebih jelas tentang karakteristik motor servo *MG 90S*, tabel 2.3 dibawah ini menunjukkan spesifikasi dari motor servo SG 90, sebagai berikut.

Tabel 2.3 Spesifikasi motor servo *MG90S*

Motor Servo	<i>Micro Servo MG90S</i>
Dimensi	22.5 x 12 x 35.5 mm
Berat (Hanya Motor)	13.4 gram

Kecepatan operasi	0.1s/ 60 derajat
<i>Pulse Width</i>	500 – 2400 μ s
<i>PWM Period</i>	20 ms (50Hz)
Tegangan operasi	4,8 V – 60 V
Arus	Kurang Dari 500 mA
Kisaran suhu	0 ° C Sampai 55° C
Panjang Kabel	150 mm
<i>Stall Torque</i>	1.8 Kgf/Cm
<i>Gear Type</i>	<i>Metal</i>
<i>Limit angle</i>	180° ($\pm 10^\circ$)
<i>Neutral position</i>	1500 μ s

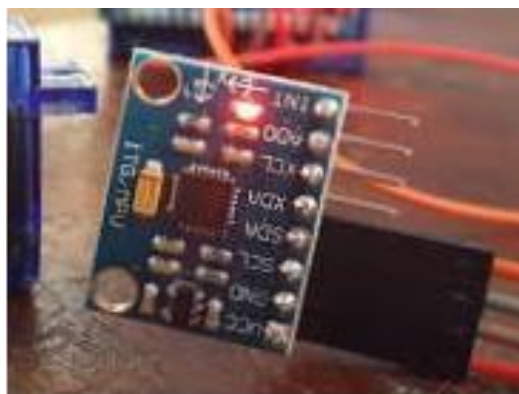
2.2.10 *Inertial Measurement Unit (IMU)*

Terdapat beberapa alat sensor yang dipakai dalam dunia elektomekanik antara lain untuk mengukur kemiringan dan getaran yaitu *accelerometer* dan *gyroscope*. Kembali lagi ke *Inertial Measurement Unit (IMU)* merupakan sebuah alat pengukur inersia yang digunakan sebagai pengukur suatu keseimbangan maupun sudut kemiringan (Raranda & Rusimamto, 2017). Pada IMU sendiri merupakan gabungan dari sensor atau alat ukur berupa *accelerometer* dan *gyroscope*. *Accelerometer* merupakan alat yang digunakan dalam mengukur getaran (vibrasi), mengukur percepatan gravitasi bumi (inklinasi) dan mengukur percepatan. Sederhananya *accelerometer* merupakan alat sensor pengukur posisi dan perpindahan. Sedangkan menurut (Indra, dkk 2016) *accelerometer* merupakan

alat ukur untuk mengukur akselerasi, dimana suatu *accelerometer* mendapatkan akselerasi ketika mengalami fenomena berat pada uji massa kerangka acuan *accelerometer*. Misalnya akselerasi yang diukur oleh *accelerometer* di permukaan bumi sebesar $g = 9.81 \text{ m/s}^2$. Prinsip kerja *accelerometer* secara umum yaitu jika konduktor digerakkan melewati suatu medan magnet atau sebaliknya, maka timbul suatu tegangan induksi pada konduktor tersebut.

Gyroscope merupakan alat yang dipakai dalam mengukur dan mempertahankan orientasi benda berdasarkan prinsip momentum sudut (Indra, dkk 2016). Sederhananya *gyroscope* dipakai sebagai sensor sudut atau gerak rotasi. Prinsip kerja *gyroscope* yaitu dengan mempertahankan orientasi rotor. Jika *gyroscope* diputar dari depan kebelakang, maka frame luar saja yang hanya bergerak karena poros berputar terdapat pada hubungan frame dengan gimbal horizontal (Indra, dkk 2016).

Pada perancangan kali ini penggunaan dari *accelerometer* dan *gyroscope* akan digantikan menggunakan sensor MPU 6050. MPU 6050 merupakan modul sensor yang berisi MEMS (Micro Electro Mechanical System) untuk *accelerometer* dan *gyro* dalam satu chip yang kecil. Dalam MPU 6050 ini terdapat 3 sumbu *accelerometer* dan 3 sumbu *gyroscope*, sumbu tersebut yaitu (x, y dan z) yang terkoneksi secara *Inter-Integrated Circuit* (I2C) (Raranda & Rusimamto, 2017). Gambar 2.28 dibawah ini menunjukkan sensor MPU 6050 dan pada tabel 2.4 dibawah ini merupakan spesifikasi dari MPU6050.



Gambar 2.26 Sensor MPU 6050 Accelerometer + Gyro

(Sumber : Muhammad, 2017)

Tabel 2.4 Spesifikasi MPU6050

Menggunakan chip MPU6050
Power supply 3V-5V
Gyroscope range ± 250 ± 500 ± 1000 ± 2000
Accelerometer range ± 2 ± 4 ± 8 ± 16
Digital Output I ² C
3 sumbu <i>accelerometer</i> dan 3 sumbu <i>gyroscope</i>
Chip built in 16 bit AD converter, 16 bit data output
GY-521
Jarak antara header 2.54mm

Mpu6050 ini sangat akurat, karena berisi 16 bit perangkat keras konversi analog kedigital pada setiap saluran. Untuk pembacaan keluaran data dari MPU6050 dilihat pada serial monitor Arduino IDE.

2.2.11 Teknik Kendali

Model matematik adalah sebuah deskripsi matematik dari sistem yang dinyatakan dalam bentuk hubungan matematik dari masukan dan keluaran sistem. Model matematik digunakan untuk memperkirakan bagaimana sistem akan memberi suatu respon psaat kondisi spesifik, tanpa menguji sistem fisik sebenarnya. Terdapat model matematik yaitu model matematik sistem mekanik dan model matematik sistem elektrik.

Model matematik sistem mekanik untuk mencari persamaan karakteristik menggunakan hokum newton. Gerak translasi $\Sigma \mathbf{F} = \mathbf{m} \mathbf{a}$ dan gerakan rotasi $\Sigma \boldsymbol{\tau} = \mathbf{J} \boldsymbol{\alpha}$. Sebagai contoh pada sistem mekanik pegas, sistem mekanik roda gigi dan sistem mekanik pengungkit. Sedangkan model matematik sistem elektrik untuk mencari persamaan karakteristik sistem menggunakan hukum ohm, persamaan tegangan kirchoff dan arus mesh. Sebagai contoh pada resistor, kapasitor dan inductor.

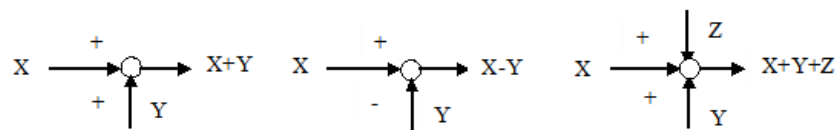
Diagram blok adalah suatu pernyataan gambar secara ringkas dari hubungan sebab dan akibat antara masukkan dan keluaran dari suatu sistem. Dapat juga

diagram blok diartikan suatu penyajian bergambar dari fungsi yang dilakukan oleh setiap komponen sistem dan aliran sinyalnya.



Gambar 2.27 Bentuk diagram blok

Ciri-ciri operasi penjumlahan dan pengurangan, agar dapat digambarkan secara khusus, maka bentuk blok seperti diatas diubah menjadi sebuah lingkaran kecil yang disebut dengan titik penjumlahan, dengan tanda *plus* (+) dan atau *minus* (-), yang tetap sesuai dengan anak-anak panah yang memasuki lingkaran. Sedangkan keluarannya adalah jumlah aljabar dari inputnya. Contoh :

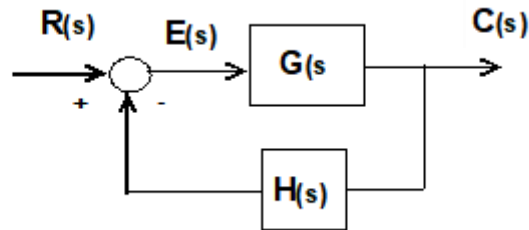


Gambar 2.28 Operasi penjumlahan dan pengurangan

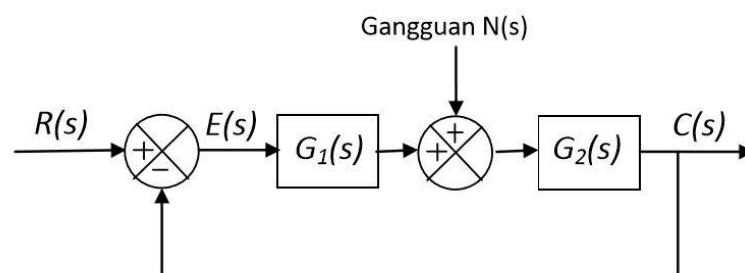
Terdapat juga sensing (*error detector*), diagram blok sistem loop tertutup dan diagram blok sistem loop tertutup dengan adanya gangguan seperti pada gambar dibawah ini.



Gambar 2.29 *Error detector*



Gambar 2.30 diagram blok sistem loop tertutup

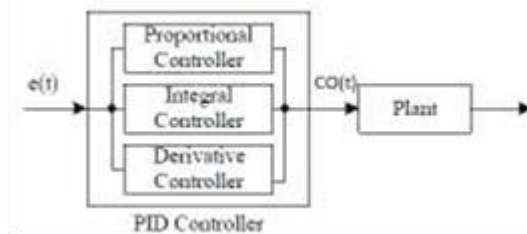


Gambar 2.31 diagram blok sistem loop tertutup dengan gangguan

Sistem kontrol PID (Proportional Integral dan Derivative) merupakan mekanisme umpan balik guna menentukan tingkat kepresisian suatu instrument yang biasanya dipakai pada sistem kontrol industri (Putra,dkk 2016). Sistem kontrol PID sangatlah mudah untuk diperoleh serta kemudahan pemakaiannya dalam suatu perancangan dikarenakan nilai redaman yang diinginkan langsung dapat dihitung pada semua kondisi jalan (Biyanto, 2006). Sistem kontrol PID merupakan jenis pengatur yang selain banyak dimanfaatkan juga sistem ini sangat mudah ketika harus dikombinasikan dengan berbagai metode seperti *Robust* dan kontrol logika *Fuzzy* (Rohmad, dkk 2015).

Kendali PID merupakan gabungan dari tiga buah kontroler pengendali, antara lain : Kontroler proposional (*Propotional Controller*), Kontroler integral (*Integral Controller*), dan Kontroler turunan (*Derivative Controller*) (Wijaya, 2018). Menurut (Santosa, dkk 2012) adapun tujuan dari penggabungan dari tiga buah kontroler pengendali yaitu jika terdapat suatu kelebihan dan kekurangan dari masing-masing pengendali maka akan saling melengkapi dan menutup agar kinerja

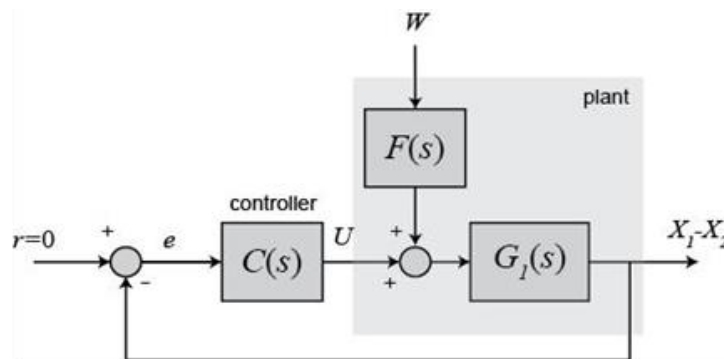
sistem menjadi lebih baik lagi. Pada gambar 2.32 dibawah ini menunjukkan diagram blok kontroler PID ideal.



Gambar 2.32 Kontrol PID ideal

(Sumber : Santoso, 2012)

Keluaran dari kontroler PID merupakan jumlah dari ketiga buah kontroler pengendali. Diagram blok digunakan agar dapat memahami respon kerja pada sistem loop tertutup (Edie, 2015). Pada gambar 2.33 menunjukkan diagram blok yang dimaksud.



Gambar 2.33 Blok diagram Unity Feedback System

(Sumber : Rohmad, 2015)

Sedangkan persamaan 2.5 dibawah ini yang menunjukkan bentuk umum PID ideal.

$$m(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \dots\dots\dots 2.9$$

Persamaan untuk PID ideal independent (setelah di eskpansi) ditunjukkan pada persamaan 2.6 dibawah ini.

$$m(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \dots\dots\dots 2.10$$

Keterangan :

m = sinyal keluaran kendali

e = sinyal kesalahan penggerak

T_i = waktu integral

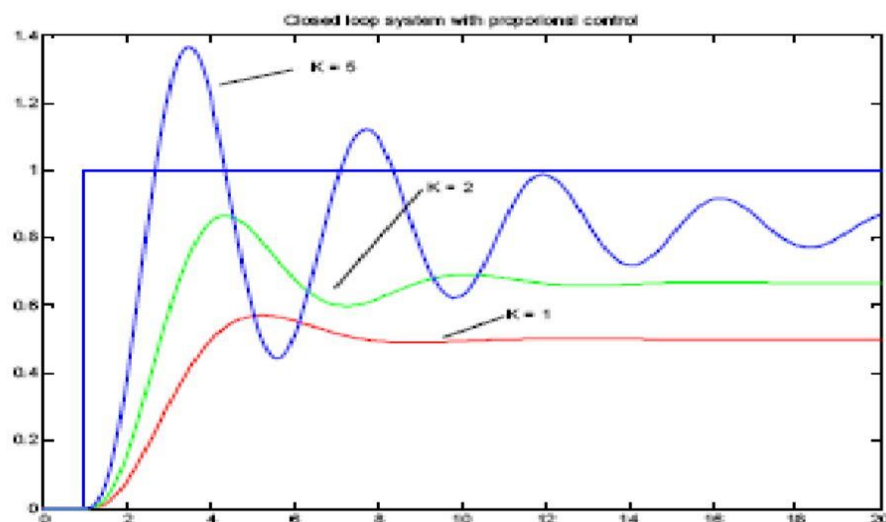
K_p = sensitifitas proporsional / penguatan

T_d = waktu derivatifnya

Kontroler PID mempunyai *transfer function* seperti persamaan 2.7 sebagai berikut:

$$H(s) = \frac{K_D s^2 + K_P s + K_I}{s} \dots\dots\dots 2.11$$

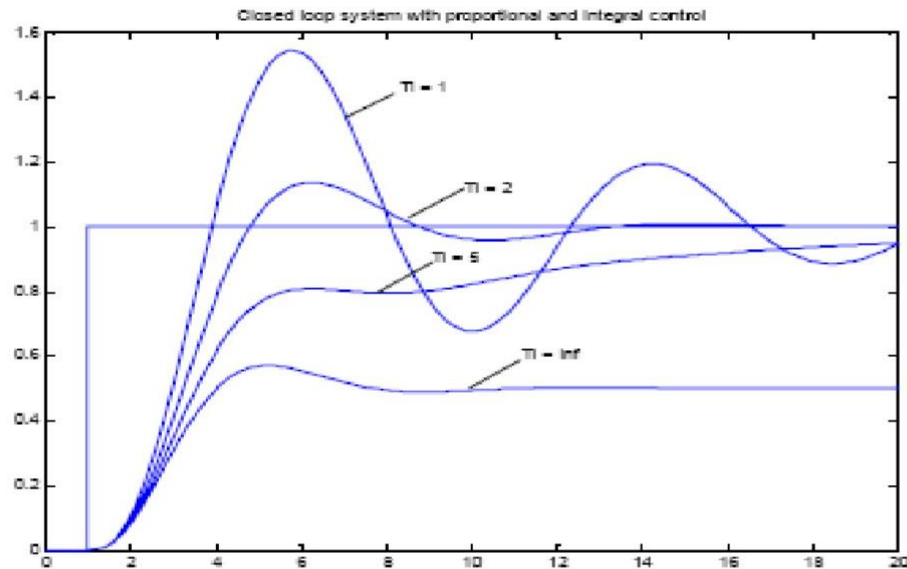
Pada kontroler PID suatu parameter harus diposisikan agar dapat bekerja dengan optimal, itu yang dimaksud sebagai konstanta. Kontrol *Proporsional* memiliki sifat: mesin berputar dengan kecepatan *proporsional* dengan besarnya sinyal error, keluaran tidak dapat mencapai nilai yang diinginkan karena sinyal error bernilai nol, peralatan lebih awet dibandingkan pada kontrol *on off* karena mesin tidak hidup berulang. Pengaruh konstanta *Proporsional* dilihat pada gambar 2.34 dibawah ini.



Gambar 2.34 Grafik aksi Kontrol *Proporsional*

Sumber : (Yulanta, 2006)

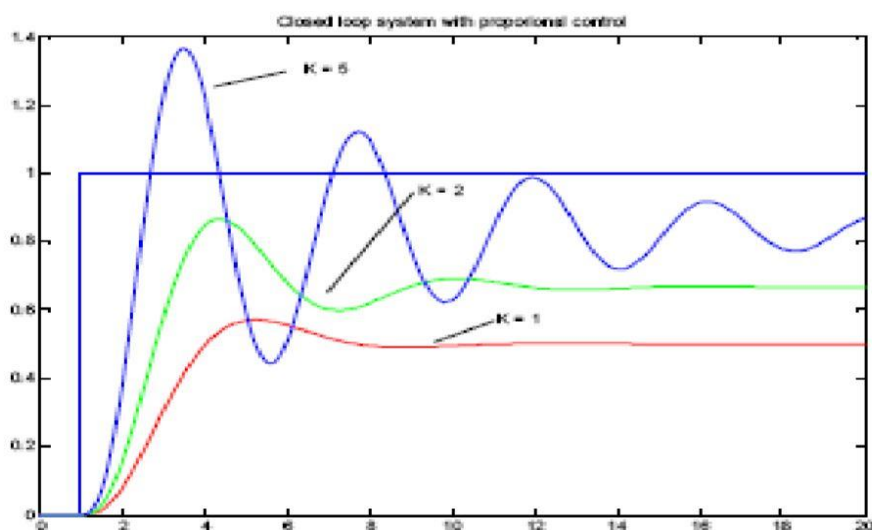
Tujuan dari kontrol *Integral* yaitu untuk menghilangkan error pada keadaan *steady*. Aksi dari kontrol *Integral* biasanya disebut kontrol reset. Pengaruh konstanta *Integral* dilihat pada gambar 2.35 dibawah ini.



Gambar 2.35 Grafik aksi Kontrol *Integral*

Sumber : (Yulanta, 2006)

Kontrol Derivative atau kontrol turunan digunakan pada controller untuk mempercepat respon transient dari sistem kontrol. Adanya aksi derivative pada suatu controller memberi efek penstabil terhadap penguatan loop kontrol.



Gambar 2.36 Grafik aksi Kontrol Derivative

Sumber : (Yulanta, 2006)

Masing-masing jenis tersebut pun memiliki beberapa keunggulan dan kekurangan seperti terlihat pada tabel 2.5 dibawah ini (Rohmad, dkk 2015).

Tabel 2.5 Respon Kontroler PID terhadap perubahan konstanta.

<i>Closed-Loop Response</i>	<i>Rise Time</i>	<i>Overshoot</i>	<i>Settling Time</i>	<i>SS Error</i>
<i>K_p</i>	Turun	Naik	Sedikit perubahan	Turun
<i>K_i</i>	Turun	Naik	Naik	Hilang
<i>K_d</i>	Sedikit perubahan	Turun	Turun	Sedikit perubahan

Tabel tersebut digunakan pada saat perubahan konstanta. Metode yang digunakan pada merancang sistem kontrol PID yang mengharuskan jika ingin mendapatkan hasil yang terbaik yaitu menggunakan metode *trial* dan *error*.