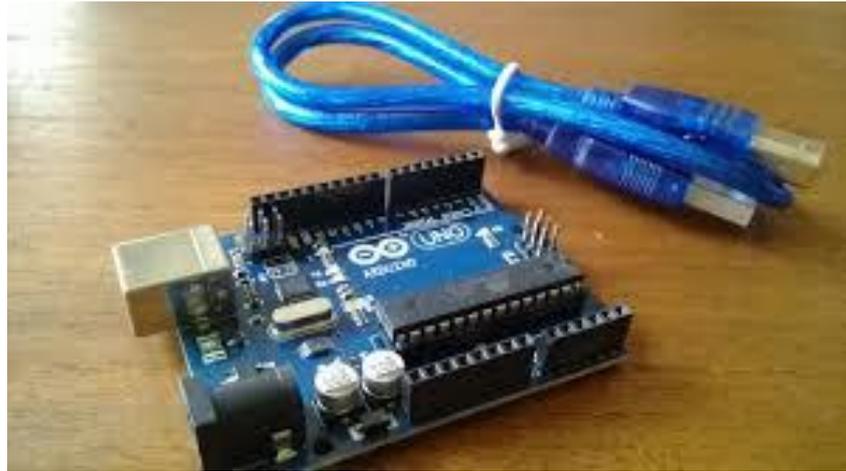


BAB II TINJAUAN PUSTAKA

2.1. Pengertian Arduino UNO



Gambar 2.1. Board Arduino

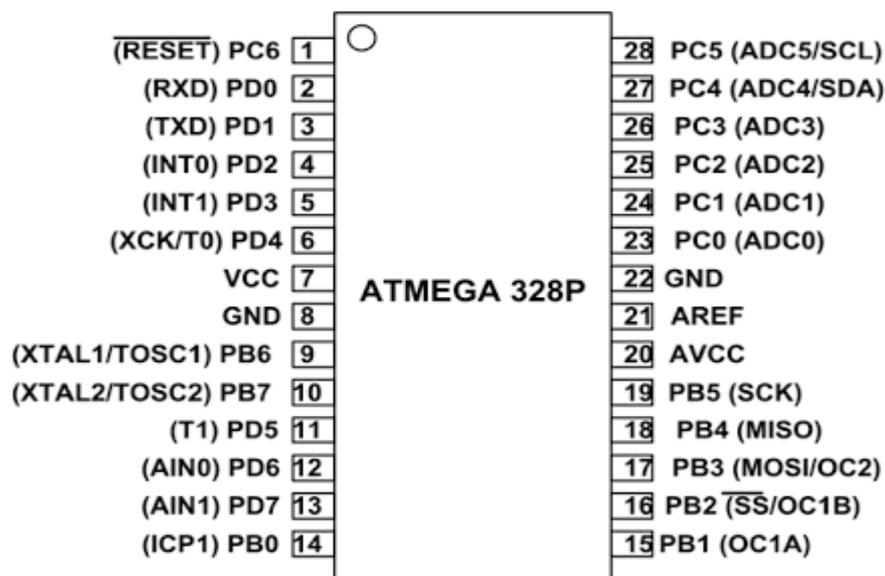
Menurut Abdul Kadir (2016:1), Arduino merupakan perangkat keras sekaligus perangkat lunak yang memungkinkan melakukan pembuatan *prototype* suatu rangkaian elektronika yang berbasis mikrokontroler dengan mudah dan cepat. Arduino berbasis mikrokontroler yang dikeluarkan oleh perusahaan Atmel contohnya Arduino Uno yang menggunakan mikrokontroler Atmega328P. Perlu diketahui, pada saat ini istilah Arduino Uno digunakan untuk produk yang dikeluarkan di Amerika Serikat, sedangkan Genuino Uno digunakan untuk produk yang dipasarkan di luar Amerika Serikat. Namun, untuk penyederhanaan disebut Arduino atau Arduino Uno. Dari sisi perangkat lunak, Arduino IDE adalah *tool* yang bermanfaat untuk menuliskan program (yang secara khusus dinamakan sketsa di Arduino), mengompilasinya, dan sekaligus mengunggahnya ke papan Arduino.

Papan Arduino Uno bekerja dengan tegangan masukan 7-12 Volt. Adapun tegangan kerja yang digunakan adalah 5 Volt. Papan ini mengandung 14 *pin digital* dan 6 di antara *pin* tersebut dapat bertindak sebagai *pin PWM (Pulse Width*

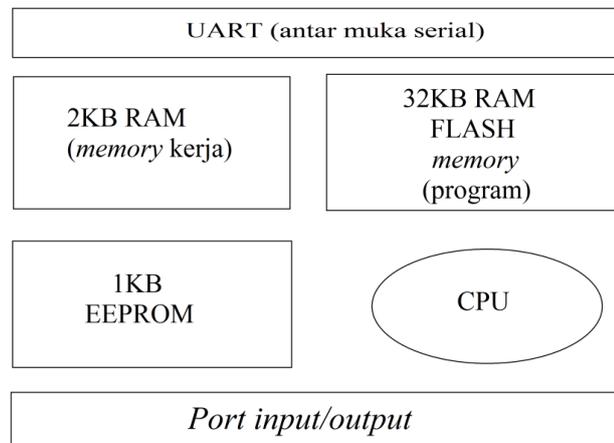
Modulation), yang memungkinkan untuk mendapatkan isyarat *analog* di *pin digital*. PWM berguna misalnya untuk meredupkan LED atau mengatur kecepatan putar motor. Papan ini juga menyediakan 6 *pin analog*.

2.2. Mikrokontroler Atmega 328P

ATMega328 merupakan mikrokontroler keluarga AVR 8 bit. Beberapa *type* mikrokontroler yang sama dengan ATMega8 ini antara lain ATMega8535, ATMega16, ATMega32, ATMega328, yang membedakan antara mikrokontroler antara lain adalah, ukuran *memory*, banyaknya GPIO (*pin input/output*), *peripheral* (USART, *timer*, *counter*, dll). Dari segi ukuran fisik, ATMega328 memiliki ukuran fisik lebih kecil dibandingkan dengan beberapa mikrokontroler diatas. Namun untuk segi *memory* dan *peripheral* lainnya ATMega328 tidak kalah dengan yang lainnya karena ukuran memori dan *peripheral relative* sama dengan ATMega8535, ATMega32, hanya saja jumlah GPIO lebih sedikit dibandingkan mikrokontroler diatas. (S. Wicaksono, 2017)



Gambar 2.2. Pin *Chip* Atmega328

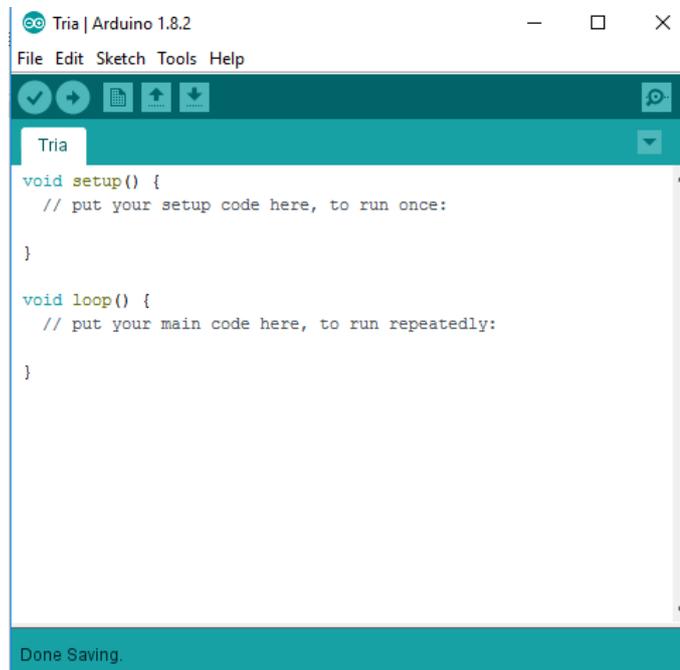


Gambar 2.3. Arsitektur Atmega 328P

Menurut Yuwono Martta Dinata (2016:8), Arsitektur Atmega328P memiliki beberapa bagian seperti gambar 2.3 diatas dan adapun keterangannya sebagai berikut.

- *Universal Asynchronous Receiver/Transmitter* (UART) adalah antarmuka yang digunakan untuk komunikasi serial, seperti pada RS-232, RS-422, dan RS-485.
- 2 KB RAM pada *memory* kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variabel-variabel di dalam program.
- 32 KB ROM *flash memory* bersifat *non-volatile*, digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, *flash memory* juga menyimpan *bootloader*.
- *Bootloader* adalah program inisiasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah *bootloader* selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi.
- 1 KB EEPROM bersifat *non-volatile*, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada papan Arduino.
- *Central Processing Unit* (CPU), bagian dari mikrokontroller untuk menjalankan setiap intruksi dari program.
- *Port input/output* (I/O), *pin* untuk menerima data (*input*) *digital* atau *analog*, dan mengeluarkan data (*output*) *digital* atau *analog*.

2.3. *Integrated Development Environment (IDE) Arduino*



Gambar 2.4. Ide Arduino

Menurut Heri dan Aan (2015:31-38), *software* IDE Arduino adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari *platform Wiring*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang, *hardware* menggunakan *processor* Atmel AVR dan *software* memiliki bahasa pemrograman C++ yang sederhana dan fungsi-fungsinya yang lengkap.

1) *File*

Tabel 2.1. Pilihan pada Menu *File*

<i>New</i>	Membuat <i>sketch</i> * baru
<i>Open</i>	Membuka <i>file sketch</i> yang sudah disimpan
<i>Sketchbook</i>	Membuka <i>file sketch</i> yang pernah dibuat
<i>Example</i>	Membuka contoh-contoh <i>file sketch</i> yang berisi berbagai macam aplikasi yang disediakan oleh arduino
<i>Close</i>	Menutup <i>sketch</i>
<i>Save</i>	Menyimpan <i>sketch</i>

<i>Save as</i>	Menyimpan <i>sketch</i> dengan nama lain
<i>Upload to I/O Board</i>	Mengunggah program ke <i>board</i>
<i>Page Setup</i>	Mengatur ukuran halaman pada pencetak
<i>Print</i>	Mencetak <i>sketch</i>
<i>Preference</i>	Mengatur <i>setting</i> IDE Arduino
<i>Quit</i>	Keluar dari IDE Arduino

2) Menu *Edit*

Cut, Copy, Copy for Forum, Copy as HTML, Paste, Select All, Comment, Increase Indent, Decrease Indent, Find, Find Next.

3) Menu *Sketch*

Tabel 2.2. Pilihan pada Menu *Sketch*

<i>Verify/Compile</i>	Mengompilasi program
<i>Stop</i>	Menghentikan kompilasi (apabila ' <i>Hang</i> ')
<i>Show sketch folder</i>	Menampilkan <i>folder</i> dari <i>sketch</i> yang sedang dibuka
<i>Import Library</i>	Mengambil <i>hader library</i> dari fungsi-fungsi tambahan
<i>Add File</i>	Menambah buka <i>file sketch</i> pada jendela yang sama

4) Menu *Tools*

Tabel 2.3. Pilihan pada Menu *Tools*

<i>Auto Format</i>	Mengatur <i>format sketch</i> secara otomatis
<i>Archive Sketch</i>	Menyimpan <i>sketch</i> dalam bentuk <i>Zip file</i> (kompresi)
<i>Fix Encoding & Reload</i>	Membatalkan perubahan <i>sketch</i> dan mengambil ulang <i>sketch</i> sebelumnya yang telah disimpan
<i>Serial Monitor</i>	Mengaktifkan jendela tampilan komunikasi serial pada komputer
<i>Board</i>	Menentukan jenis <i>board</i> arduino yang digunakan
<i>Serial Port</i>	Menentukan <i>port serial</i> yang digunakan untuk mengunggah program dan tersambung pada board arduino
<i>Burn Bootloader</i>	Memasukkan <i>bootloader</i> pada mikrokontroler yang ada pada <i>board</i> arduino melalui ICSP

5) Menu *Help*

Tabel 2.4. Pilihan pada Menu *Help*

<i>Verify</i>	Untuk mengkompilasi program artinya mengkonversi program pada arduino menjadi informasi/data yang dapat dieksekusi/dibaca oleh mikrokontroler
<i>Upload</i>	Untuk mengunggah program ke dalam <i>board</i> arduino
<i>New</i>	Untuk membuat <i>file sketch</i> baru
<i>Open</i>	Untuk membuka <i>file sketch</i> yang sudah pernah dibuat
<i>Save</i>	Untuk menyimpan <i>sketch (list program)</i> yang sedang dibuat
<i>Serial Monitor</i>	Untuk mengaktifkan jendela komunikasi <i>serial</i> , dan <i>transfer</i> data (kirim/terima) antara <i>board</i> arduino dan komputer

2.4. *Bluetooth HC-05*

Menurut Heri dan Aan (2015:137), *Bluetooth* adalah teknologi komunikasi nirkabel yang beroperasi pada pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific, and medical)*. *Bluetooth* dapat dipakai untuk melakukan komunikasi data di antara peralatan dengan jarak jangkauan yang cukup jauh. Besarnya jarak jangkauan tergantung pada kelas *Bluetooth* dalam *transceiver Bluetooth* ada tiga kelas pembagian daya yaitu:

- Daya kelas 1 beroperasi antara 100mW (20dBm) dan 1mW (0dBm), dirancang untuk perangkat dengan jangkauan yang jauh hingga mencapai 100 m.
- Daya kelas 2 beroperasi antara 2,5 mW (4dBm) dan 0,25mW (-6dBm), dirancang untuk perangkat dengan jangkauan hingga mencapai 10 m.
- Daya kelas 3 beroperasi pada 1 mW (0dBm), dirancang untuk perangkat dengan jangkauan pendek atau sekitar 1 m.



Gambar 2.5. Modul *bluetooth* HC-05

2.5. Motor DC

Motor DC adalah jenis motor yang menggunakan tegangan searah (DC) yang secara otomatis akan berputar terus-menerus selama mendapatkan pasokan tegangan. Tegangan yang digunakan untuk menggerakkan motor DC bervariasi. Sebagai contoh, terdapat motor DC yang dapat digerakkan dengan baterai 3 Volt, tetapi ada yang memerlukan tegangan yang lebih tinggi. (Abdul Kadir, 2016)



Gambar 2.6. Motor DC

Mariza Wijayanti (2015), Motor arus searah (DC) digunakan pada penggunaan khusus dimana diperlukan penyalaan *torque* yang tinggi atau percepatan yang tetap untuk kisaran kecepatan yang luas. Motor DC memiliki tiga komponen utama yaitu:

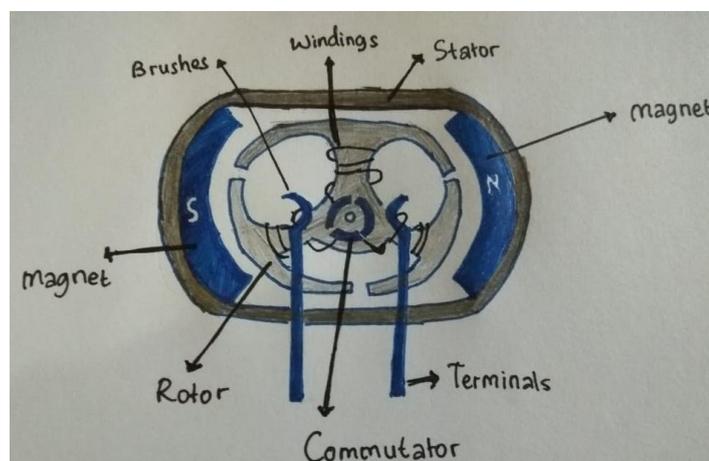
- **Kutub medan.** Secara sederhana digambarkan bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor DC. Motor DC memiliki kutub medan yang stasioner dan dinamo yang menggerakkan *bearing* pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub

medan: kutub utara dan kutub selatan. Garis magnet energi membesar melintasi bukaan diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet. Elektromagnet menerima listrik dari sumber daya dari luar sebagai penyedia struktur medan.

- **Dinamo.** Bila arus masuk menuju dinamo, maka arus ini akan menjadi elektromagnet. Dinamo yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, dinamo berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi. Jika hal ini terjadi, arusnya berbalik untuk merubah kutub-kutub utara dan selatan dinamo.
- **Commutator.** Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikkan arah arus listrik dalam dinamo. *Commutator* juga membantu dalam transmisi arus antara dinamo dan sumber daya.

Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur:

- Tegangan dinamo – meningkatkan tegangan dinamo akan meningkatkan kecepatan.
- Arus medan – menurunkan arus medan akan meningkatkan kecepatan.



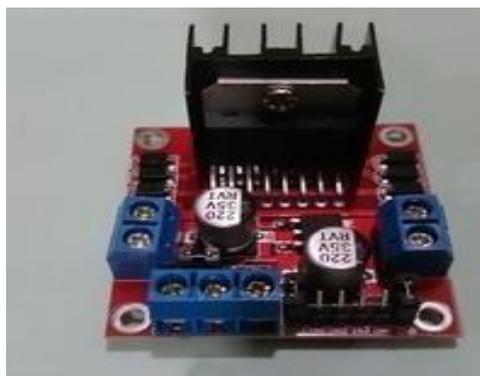
Gambar 2.7. Bagian-bagian motor DC

▪ Prinsip Arah Putaran Motor

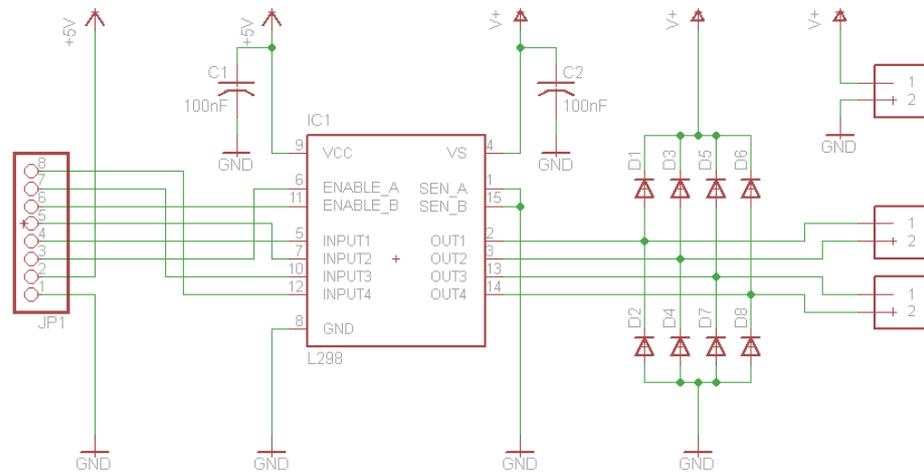
Untuk menentukan arah putaran motor digunakan kaedah Flamming tangan kiri. Kutub-kutub magnet akan menghasilkan medan magnet dengan arah dari kutub utara ke kutub selatan. Jika medan magnet memotong sebuah kawat penghantar yang dialiri arus searah dengan empat jari, maka akan timbul gerak searah ibu jari. Gaya ini disebut gaya Lorentz, yang besarnya sama dengan F . Prinsip motor yaitu aliran arus di dalam penghantar yang berada di dalam pengaruh medan magnet akan menghasilkan gerakan. Besarnya gaya pada penghantar akan bertambah besar jika arus yang melalui penghantar bertambah besar (Chairul Huda, 2010).

2.6. Driver Motor L298N

Menurut Dyah Lestari (2015), *driver* motor L298N adalah *Driver* motor yang didalamnya terdapat 2 rangkaian *Driver H-bridge*, terdapat 15 kaki diantaranya terdapat 2 kaki VCC yaitu vcc untuk motor dan vcc untuk *Driver* L298, terdapat 2 input PWM dan 4 *input* logika untuk putaran motor, terdapat 4 *output* untuk motor, dan 1 kaki adalah *ground*. Gambar 2.8 berikut rangkaian *driver* motor.



Gambar 2.8. *Driver* motor L298N



Gambar 2.9. Skematik Driver Motor L298N

2.7. Motor Servo

Menurut Abdul Kadir (2016), motor servo adalah jenis motor bertegangan searah yang sudut putarannya dapat diatur. Normalnya, putaran motor dapat diatur sebesar 0 derajat hingga 180 derajat. Akan tetapi, terdapat pula jenis motor servo yang dapat berputar 360 derajat. Motor servo biasa digunakan pada lengan robot. Motor servo memiliki tiga kabel. Kegunaan masing-masing adalah seperti berikut.

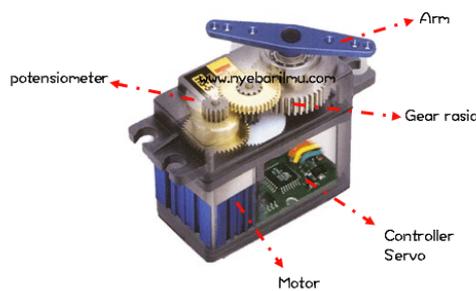
- Kabel merah dihubungkan ke pemasok tegangan hubungan positif
- Kabel coklat dihubungkan ke ground.
- Kabel kuning dihubungkan ke *pin* PWM yang digunakan untuk mengatur sudut putaran motor.



Gambar 2.10. Motor Servo

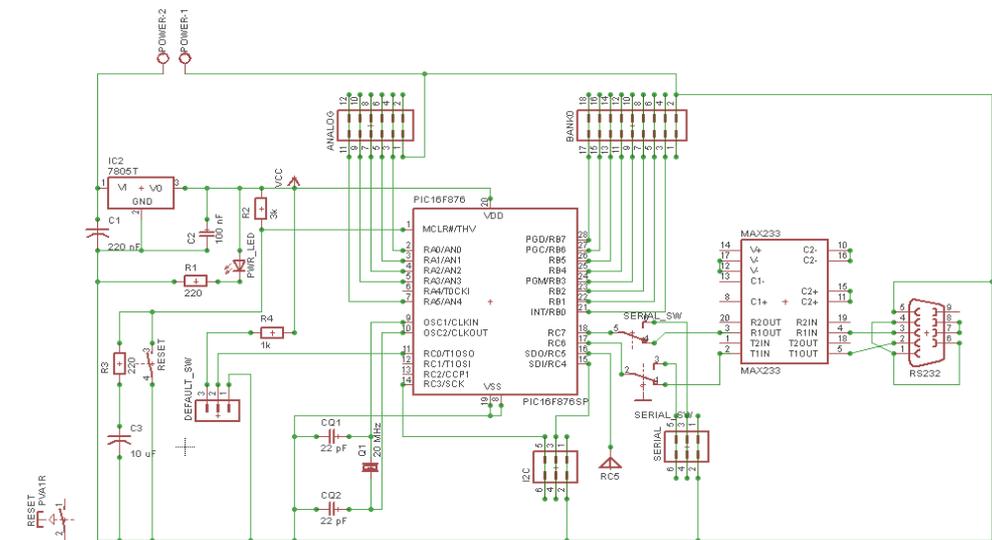
Motor servo disusun dari sebuah motor DC, gearbox, variabel resistor (VR) atau potensiometer dan rangkaian kontrol. Potensiometer berfungsi menentukan batas maksimum putaran sumbu (axis) motor servo. Sudut sumbu motor servo diatur berdasarkan lebar pulsa pada pin kontrol motor servo. Motor servo adalah motor yang mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dikendalikan dengan memberikan variasi lebar pulsa (duty cycle) sinyal PWM pada bagian pin kontrolnya. Berikut jenis motor servo:

1. Motor servo standar 180° Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°.
2. Motor servo continuous Motor servo mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).



Gambar 2.11. Bagian-bagian motor servo

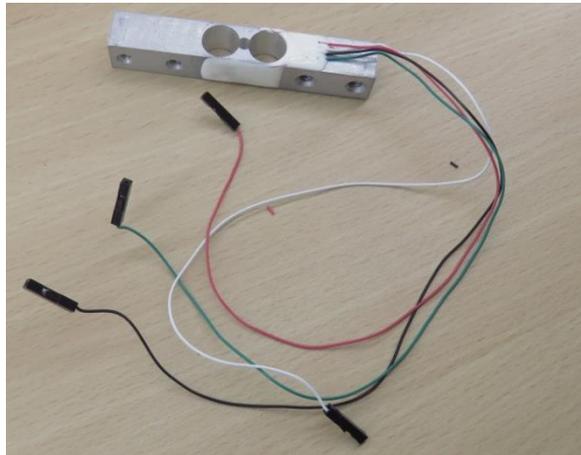
(www.nyebarilmu.com/cara-mengakses-motor-servo-menggunakan-arduino)



Gambar 2.12. Skematik Motor Servo

2.8. Loadcell HX711

Menurut jurnal mekanikal Ramang Magga (2011:53), *Loadcell* HX711 adalah modul timbangan, memiliki prinsip kerja mengkonversi perubahan terukur dalam perubahan resistansi dan mengkonversinya ke dalam besaran tegangan melalui rangkaian yang ada. *Loadcell* memiliki bagian yang sangat penting yaitu *Strain Gage* dengan fungsi untuk mendeteksi besarnya perubahan dimensi jarak yang disebabkan oleh suatu elemen gaya. *Strain gages* secara umum digunakan dalam pengukuran presisi gaya, berat, tekanan, torsi, perpindahan dan kuantitas mekanis lainnya dan dikonversi menjadi ketegangan dalam anggota mekanis.



Gambar 2.13. Loadcell HX711

2.9. Bahasa Pemrograman Arduino Berbasis C

Menurut Yuwono Martta Dinata (2016:43-88), dasar struktur dari bahasa pemrograman Arduino agar program dapat berjalan dengan baik memiliki dua bagian atau fungsi utama yaitu *setup()* untuk inialisasi program (*setting input* atau *setting serial*, dan lain-lain) dan *loop()* untuk membaca *input*, atau *trigger output*. Berikut adalah fungsi-fungsi dasar lainnya pada bahasa pemrograman arduino:

Tabel 2.5. Fungsi-fungsi dasar pemrograman arduino

No.	Fungsi-fungsi Dasar	Keterangan
1.	<i>Setup()</i>	Inisialisasi <i>pin</i> dan <i>serial</i>
2.	<i>Loop()</i>	Mengeksekusi fungsi mengulang terus menerus
3.	<i>Function()</i>	Sekumpulan program yang diberi nama khusus
4.	{ } kurung kurawal	Mengawali dan mengakhiri fungsi
5.	; (Titik koma)	Tanda akhir dari instruksi
6.	<i>/*.....*/</i> Blok komentar	Memberi komentar atau catatan tentang program > 1 baris
7.	<i>//</i> Komentar	Memberi komentar tunggal program
8.	Variabel	<i>Byte</i> : menyimpan data numerik 8 bit <i>Int</i> : menyimpan data angka 16 bit <i>Long</i> : menyimpan data angka 32 bit <i>Float</i> : data numerik memiliki nilai desimal
9.	<i>Array</i>	Kumpulan nilai-nilai yang diakses dengan nomor <i>index</i>
10.	Aritmatika	Penambahan, pengurangan, perkalian, dan pembagian
11.	Operasi Gabungan	Operasi aritmatika gabungan
12.	Operator Perbandingan	Membandingkan 2 konstanta atau variabel
13.	Operator Logika	Membandingkan 2 ekspresi <i>TRUE</i> atau <i>FALSE</i>
14.	Konstanta	Nilai-nilai yang telah ditetapkan
15.	<i>TRUE/FALSE</i>	Konstanta Boolean yang mendefinisikan nilai logika
16.	<i>HIGH/LOW</i>	Menentukan nilai <i>pin HIGH</i> atau <i>LOW</i>
17.	<i>Input/Output</i>	Menentukan <i>mode pin digital</i> sebagai <i>input</i> atau <i>output</i>
18.	<i>If</i>	Instruksi untuk menguji suatu kondisi telah tercapai
19.	<i>If.....else</i>	Mengeksekusi instruksi lain jika suatu kondisi

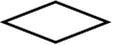
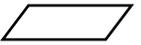
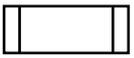
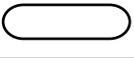
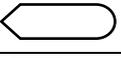
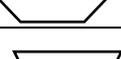
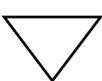
		tidak terpenuhi
20.	<i>For</i>	Untuk mengulang suatu blok instruksi di dalam kurung kurawal
21.	<i>While</i>	Menjalan program secara terus menerus sehingga suatu kondisi salah atau <i>false</i>
22.	<i>Do....write</i>	Perintah melakukan terus menerus hingga mencapai suatu kondisi yang tidak memenuhi kondisi yang diinginkan
23.	<i>pinMode(pin,Mode)</i>	Inisialisasi <i>pin</i> sebagai <i>input</i> atau <i>output</i>
24.	<i>digitalRead(pin)</i>	Membaca <i>input</i> dari suatu <i>pin</i>
25.	<i>digitalWrite(pin,value)</i>	Memberi nilai <i>output HIGH</i> (1) atau <i>LOW</i> (0) pada <i>pin digital</i>
26.	<i>analogRead(pin)</i>	Membaca nilai <i>input analog</i> dengan resolusi 10 <i>bit</i> .
27.	<i>analogWrite(pin,value)</i>	Memberi nilai PWM (<i>pulse width modulation</i>) pada <i>output</i>
28.	<i>delay(ms)</i>	Memberi jeda sebelum lanjut ke program selanjutnya
29.	<i>millis()</i>	Mengambil nilai waktu dari program berjalan sampai berhenti
30.	<i>Tone</i> (<i>pin,frekuensi,durasi</i>)	Menghasilkan nada frekuensi
31.	<i>noTone(pin)</i>	Menghentikan nada frekuensi
32.	<i>randomSeed(seed)</i>	Mengambil nilai acak dengan <i>seed</i>
33.	<i>Random(min,max)</i>	Mengambil nilai acak <i>min</i> dan <i>max</i>
34.	<i>Serial.begin(rate)</i>	Membuka <i>port data serial</i>
35.	<i>Serial.print()</i>	Untuk mengirimkan data ke <i>serial port</i>
36.	<i>Serial.read()</i>	Menerima data dari <i>serial port</i>
37.	<i>Serial.available()</i>	Instruksi untuk mendeteksi apakah menerima data dari serial port

2.10. Flowchart

Menurut I Gusti Nguah Suryantara (2009), badan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Tabel 2.6. Simbol *Flowchart*

(Sumber: Buku Analisis dan Perancangan Sisfo; Algoritma dan Pemrograman)

No.	Simbol	Nama Simbol	Keterangan
1.		<i>Alternate Process</i>	Menyatakan segala jenis operasi yang diproses dengan menggunakan mesin yang memiliki <i>keyboard</i>
2.		<i>Decision</i>	suatu penyelesaian kondisi dalam program
3.		<i>Data</i>	Mewakilik data <i>input</i> atau <i>output</i>
4.		<i>Predefined Process</i>	Suatu operasi yang rinciannya di tunjukkan di tempat lain
5.		<i>Document</i>	<i>Document input</i> dan <i>output</i> baik untuk proses manual, mekanik atau komputer
6.		<i>Terminator</i>	Untuk menunjukkan awal dan akhir dari suatu proses
7.		<i>Process</i>	Kegiatan proses dari operasi program komputer
8.		<i>Manual Input</i>	<i>Input</i> yang menggunakan <i>online keyboard</i>
9.		<i>Conector</i>	Penghubung ke halaman yang masih sama
10.		<i>Off-Page Connector</i>	Penghubung ke halaman lain
11.		<i>Display</i>	<i>Output</i> yang ditampilkan di monitor
12.		<i>Delay</i>	Menunjukkan penundaan
13.		<i>Preparation</i>	Memberi nilai awal suatu besaran
14.		<i>Manual Operation</i>	Pekerjaan manual
15.		<i>Card</i>	<i>Input</i> atau <i>output</i> yang menggunakan kartu
16.		<i>Punch Tape</i>	<i>Input</i> atau <i>output</i> menggunakan pita kertas berlubang
17.		<i>Merge</i>	Penggabungan atau penyimpanan beberapa proses atau informasi sebagai salah satu

18.		<i>Dirrect Access Storage</i>	<i>Input</i> atau <i>output</i> menggunakan drum magnetik
19.		<i>Magnetic Disk</i>	<i>Input</i> atau <i>output</i> menggunakan <i>hard disk</i>
20.		<i>Sequential Access Storage</i>	<i>Input</i> atau <i>output</i> menggunakan pita magnetik
21.		<i>Sort</i>	Proses pengurutan data di luar komputer
22.		<i>Stored Data</i>	<i>Input</i> atau <i>output</i> menggunakan <i>diskette</i>
23.		<i>Extract</i>	Proses dalam jalur paralel
24.		<i>Arrow</i>	Menyatakan jalan atau arus suatu proses
25.		<i>Summing Junction</i>	Untuk berkumpul beberapa cabang sebagai proses tunggal
26.		<i>Or</i>	Proses menyimpang dalam dua proses