

BAB II

DASAR TEORI

2.1. Tinjauan Pustaka

Junaidi Edy, dkk dalam penelitiannya yang berjudul “Rancang Bangun *Scanner* 3D Menggunakan Sensor Ultrasonik dengan Tampilan *Real Time* Berbasis Mikrokontroler” berhasil dibuat yang diuji scan pada lima buah botol dengan akurasi botol 1 96,50 %, botol 2 87,28 %, botol 3 59,54 %, botol 4 78,88 %, dan botol 5 92,77 %. Berdasarkan model model hasil pemindaian, sistem *scanner* pada penelitian ini hanya untuk memindai benda silinder yang tidak terdapat banyak lekukan seperti botol 1,2, dan 5.

Komarayatna Arrizal dan Adler John dalam penelitiannya yang berjudul “Pemodelan Objek Menggunakan 3D *Scanning*” menggunakan metode triangulasi laser dengan menggunakan perangkat keras *webcam*, *line* laser, dan motor stepper. Dari hasil pengujian yang dilakukan mampu mengenali sesuatu objek dengan cukup baik asalkan objek tersebut mempunyai bahan yang tidak bisa memantulkan sinar dari laser dan memiliki tingkat kerumitan yang cukup tinggi, sedangkan pengujian terhadap objek bangun datar menghasilkan pemodelan yang cukup baik.

Aubreton Olivier, dkk dalam penelitiannya yang berjudul “*Infrared System for 3D Scanning of metallic surfaces*” menyajikan perkembangan yang telah disadari dasar permukaan logam sangat reflektif. Perkembangan ini didasarkan pada pembaharuan sebuah teknik yang disebut “*Scanning From Heating*” yang awalnya didedikasikan untuk bahan kaca. Dibandingkan dengan teknik triangulasi aktif konvensional yang mengukur pantulan radiasi yang terlihat, disini akan diukur emisi *termal* permukaan, yang dipanaskan secara lokal oleh sumber laser. Dalam penelitian ini, mengikuti langkah-langkah dalam teknik “*Scanning From Heating*” ke bahan logam untuk mengevaluasi tampilan dan akhirnya mengembangkan operasional *prototype*.

2.2. Pemindaian Objek 3D

Scanner 3D atau pemindaian tiga dimensi adalah perangkat yang menganalisa objek dunia nyata atau lingkungan untuk mengumpulkan data tentang bentuk dan tampilan (misalnya warna). Data yang terkumpul kemudian dapat digunakan untuk membangun model tiga dimensi digital.

Banyak teknologi yang berbeda dapat digunakan untuk membangun perangkat *Scanner* 3D ini. Setiap teknologi datang dengan keterbatasan sendirinya, keuntungan dan

biaya. Masih adanya banyak keterbatasan dalam jenis objek yang dapat di digitalkan, misalnya teknologi optik menghadapi banyak masalah dengan cahaya dan benda transparan.

Banyak sekali teknologi yang memindai objek tiga dimensi ke dalam bentuk digital. Sebenarnya hanya ada dua jenis *scanner* 3D yang membedakan dari proses kerjanya, yaitu *scanner* 3D atau pemindai objek tiga dimensi secara kontak langsung dan secara kontak tidak langsung.

Metode kontak langsung adalah pemindaian yang dikumpulkan dari hasil sentuhan permukaan objek benda 3D dengan cara melakukan kontak langsung, mirip dengan meraba dalam sebuah ruangan yang gelap. Sedangkan metode kontak tidak langsung menggunakan proses pemindaian dimana data yang dikumpulkan merupakan permukaan objek 3D tanpa melakukan kontak langsung dengan objek. Pemindaian dapat dilakukan dengan menggunakan suatu sinar pemancar dan pantulan sinar tersebut akan ditangkap oleh suatu kamera. Selain menggunakan sinar, teknik ini juga dapat menggunakan suara seperti ultrasonik ataupun sensor lainnya.

2.3. Sensor GP2Y0E03

Sensor Jarak dapat diartikan sebagai sensor yang berfungsi untuk mengukur serta mengetahui letak dari sebuah objek yang berbeda jaraknya. Sensor untuk mengetahui jarak ini pada perkembangannya memiliki dua kelompok, yang pertama adalah sensor ultrasonik dan yang kedua adalah sensor infra merah. Sensor ultrasonik untuk mengukur jarak dihasilkan dari gelombang ultrasonic yang dipancarkan atau dikeluarkan oleh *transmitter* atau alat pemancar gelombang ultrasonic. *Transmitter* mengeluarkan gelombang ultrasonic yang dihasilkan dari frekuensi diatas normal dari gelombang suara.

Pancaran gelombang ultrasonic tersebut akan terus dipancarkan menyeluruh dan meluas dalam jangkauannya. Kemudian ketika pancaran gelombang ultrasonic tersebut menabrak sebuah objek tertentu, maka pancaran gelombang ultrasonic tersebut akan terhenti dan dengan kemudian berbalik arah menuju alat penerima sinyal ultrasonic atau lebih dikenal dengan istilah *receiver* yang terdapat pada sensor jarak. Pada saat itu juga *receiver* akan memberikan data dari hasil tangkapan gelombang ultrasonic tadi kepada mikrokontroler yang kemudian oleh mikrokontroler akan diproses menjadi sebuah data mengenai bentuk objek dan jarak dari objek yang tersentuh gelombang ultrasonic tadi.

Jaraknya gelombang yang dipancarkan oleh *transmitter* tergantung pada alat yang digunakan.

Jenis selanjutnya dari sensor jarak adalah sensor inframerah. Perbedaan sensor inframerah dengan sensor ultrasonic sendiri sebenarnya sangat kecil, karena perbedaannya hanya terletak pada cara kerjanya. Apabila pada sensor ultrasonic mempergunakan gelombang ultrasonic untuk mendeteksi sebuah objek pada jarak tertentu. Maka pada sensor inframerah, untuk dapat mendeteksi sebuah objek dan mendapatkan gambaran serta jaraknya adalah dengan menggunakan panas tertentu dari sebuah benda atau objek. Setiap suhu panas dari suatu objek akan tertangkap oleh sensor inframerah karena inframerah menggunakan sumber utamanya yaitu radiasi panas atau juga radiasi termal. Sensor ini biasanya digunakan sebagai indra penglihatan dari robot seperti layaknya sebuah mata pada manusia.

Sensor GP2Y0E03 merupakan sensor pengukur jarak yang terdiri dari kombinasi sensor gambar CMOS dan LED-inframerah yang terintegrasi. Variasi reflektifitas objek, suhu lingkungan dan lama operasi tidak mudah dipengaruhi jarak deteksi karena mengadopsi metode triangulasi. Perangkat ini mengeluarkan tegangan yang sesuai dengan jarak deteksi dan data digital (I²C) untuk berkomunikasi dengan perangkat lainnya.

Adapun bentuk sensor ini dapat diperhatikan pada gambar di bawah.



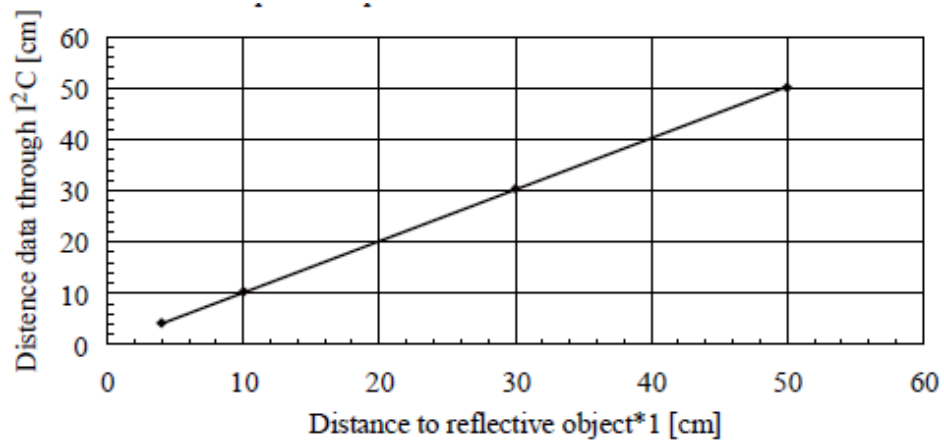
Gambar 2.1 bentuk sensor GP2Y0E03

Dalam sensor GP2Y0E03 ini memiliki 7 diagram konektor yang berfungsi sebagai komunikasi dengan perangkat yang dibutuhkan. Untuk lebih jelas dapat diperhatikan dalam tabel 2.1.

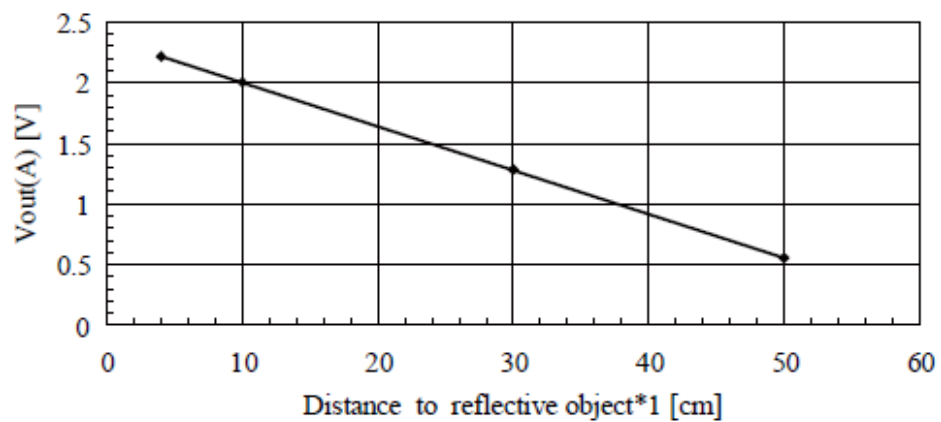
Tabel 2.1 diagram konektor sensor GP2Y0E03

No	Symbol	Nama pin
1	VDD	Tegangan suplai
2	V _{out} (A)	Terminal keluaran
3	GND	Ground
4	V _{IN} (IO)	Tegangan suplai I/O
5	GPIO	Terminal masukan untuk kontrol aktif/stand-by
6	SCL	I ² C clock
7	SDA	I ² C data

Dibawah ini contoh karakteristik jarak keluaran dari sensor GP2Y0E03 dengan karakteristik keluaran I²C terhadap cm dan karakteristik keluaran V terhadap cm seperti gambar 2.2 dan 2.3 dibawah.



Gambar 2.2 contoh karakteristik keluaran I²C terhadap cm



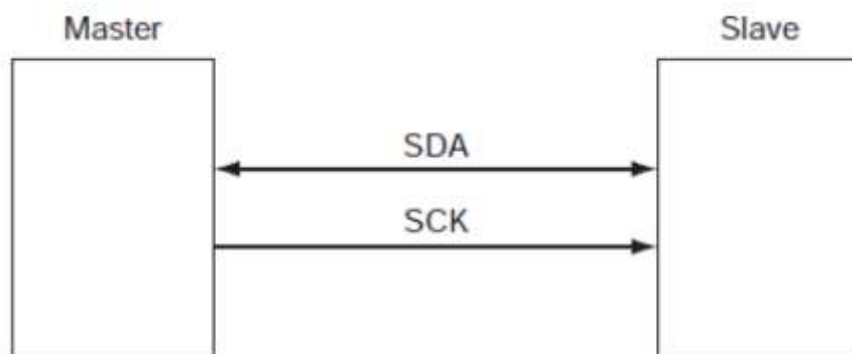
Gambar 2.3 contoh karakteristik keluaran V terhadap cm

Dalam Sensor ini dikenal dengan komunikasi I²C yang digunakan. Komunikasi I²C (*Inter-Integrated Circuit*) merupakan koneksi dibuat untuk menyediakan komunikasi antara perangkat-perangkat terintegrasi, seperti sensor, RTC, dan juga EEPROM. Komunikasi I²C bersifat *synchronous* namun berbeda dengan SPI karena I²C menggunakan *protocol* dan hanya menggunakan dua kabel untuk komunikasi, yaitu *Synchronous clock* (SCL) dan *Synchronous data* (SDA). Secara berurutan data dikirim dari *master* ke *slave* kemudian (setelah komunikasi *master* ke *slave* selesai) dari *slave* ke *master*.

Perangkat I²C menggunakan 2 buah pin *open-drain* dua arah dengan memberikan *pull-up* resistor untuk setiap garis *bus* sehingga berlaku seperti AND menggunakan kabel.

AVR dapat menggunakan 120 jenis perangkat untuk berbagi pada *bus* I²C yang masing-masing disebut sebagai *node*. Setiap *node* beroperasi sebagai *master* atau *slave*. *Master* merupakan perangkat yang menghasilkan *clock* untuk sistem, menginisiasi, dan juga memutuskan sebuah transmisi. *Slave* merupakan *node* yang menerima *clock* dan dialamatkan oleh *master*. Baik *master* dan *slave* dapat menerima dan mentransmisikan data.

I²C merupakan *protocol* komunikasi serial dimana setiap bit data ditransfer pada jalur SDA yang disinkronisasikan dengan pulsa *clock* pada jalur SCL. Jalur data tidak dapat berubah ketika jalur *clock* berada dalam kondisi *high*.



Gambar 2.4 Aliran data I²C

Dalam I²C setiap alamat atau data yang ditransmisikan harus dibentuk dalam sebuah paket dengan panjang 9 bit dimana 8 bit pertama disimpan dalam jalur SDA oleh *transmitter*, dan bit ke-9 merupakan *acknowledge* (atau *not acknowledge*) oleh *receiver*.

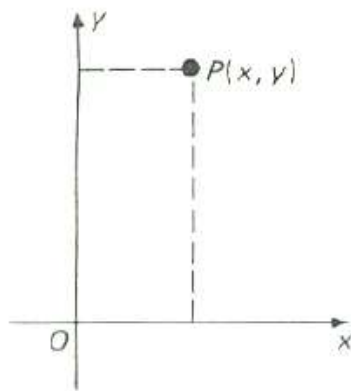
I²C juga diistilahkan sebagai *Two-wire Serial Interface* (TWI), bergantung dari istilah yang digunakan oleh pabrik yang membuat perangkat. Salah satu perangkat yang

digunakan dengan komunikasi I²C adalah *real-time clock* (RTC). Perangkat ini menyediakan komponen jam, menit, dan detik, serta tahun, bulan, dan hari.

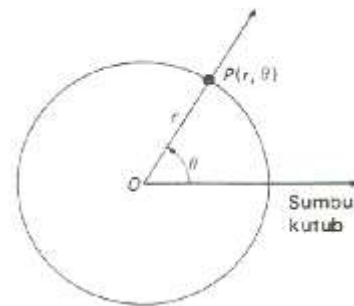
2.4. Koordinat Kutub

Sistem koordinat ini adalah dasar dari geometri analitik, dan sangat membantu pengembangan kalkulus diferensial dan kalkulus integral yang kita capai hingga saat ini. Dasar pemikiran mereka ialah untuk menunjukkan kedudukan titik P pada bidang dengan dua bilangan yang ditulis dengan lambang (x,y) setiap bilangan menggambarkan jarak berarah dari dua sumbu yang tegak lurus sesamanya (Gambar 2.5).

Dimulai dengan menggambar sebuah setengah-garis tetap yang dinamakan sumbu kutub yang berpangkal pada sebuah titik 0. Titik ini disebut kutub atau titik asal. Biasanya sumbu kutub ini kita gambar mendatar dan mengarah ke kanan dan oleh sebab itu sumbu ini dapat disamakan dengan sumbu x positif pada sebuah sistem koordinat siku-siku. Setiap titik P (selain dari kutub) adalah perpotongan antara sebuah lingkaran tunggal yang berpusat di 0 dan sebuah sinar tunggal yang memancar dari 0. Jika r adalah jari-jari lingkaran dan θ adalah salah satu sudut antara sinar dan sumbu kutub, maka (r, θ) dinamakan sepasang koordinat kutub dari titik P (Gambar 2.6).



Gambar 2.5 Koordinat Kartesian



Gambar 2.6 Koordinat kutub

Andaikan sumbu kutub berimpit dengan sumbu x positif sistem koordinat kartesius. Maka koordinat kutub (r, θ) sebuah titik P dan koordinat kartesius (x, y) titik itu dihubungkan oleh persamaan 2.3 dibawah :

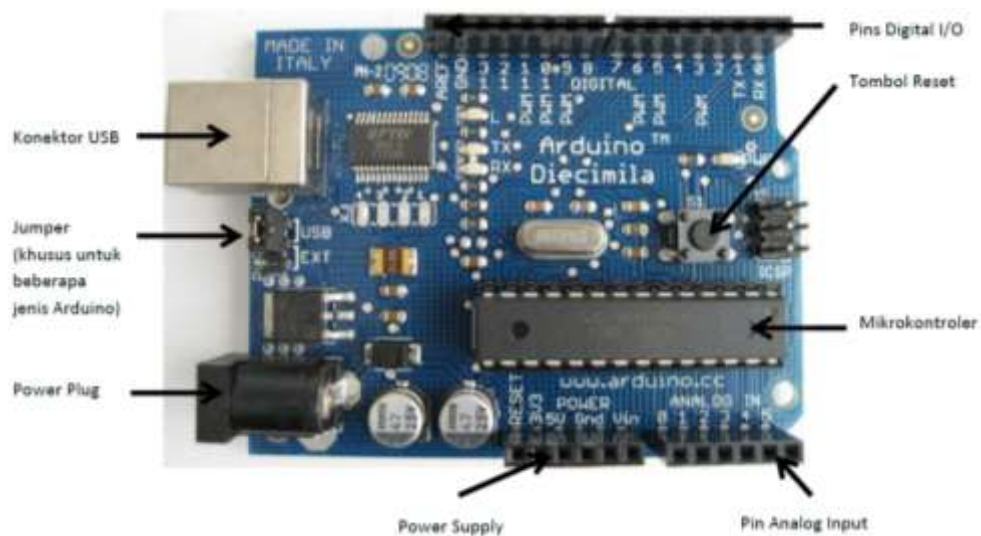
$$x = r \cos \theta \quad (2-1)$$

$$y = r \sin \theta \quad (2-2)$$

$$r^2 = x^2 + y^2 \quad (2-3)$$

$$\tan \theta = \frac{y}{x} \quad (2-4)$$

2.5. Arduino



Gambar 2.7 Bagian-bagian arduino (Diecimila)

Dari gambar 2.11 diatas dapat diamati bagian-bagian dari Arduino yaitu jenis Diecimila. Namun secara umum Arduino Diecimila dengan arduino lainnya memiliki kesamaan baik secara bagian-bagian seperti gambar 2.11 diatas. Untuk menghubungkan Arduino ke komputer diperlukan kabel USB dari Konektor USB di Arduino. Adapun berbagai tujuan untuk menghubungkan dengan komputer yaitu :

- Mengupload software baru pada papan
- Berkomunikasi dengan papan Arduino dari komputer
- Menyalurkan daya ke dalam papan Arduino

Sebagai perangkat elektronik, Arduino membutuhkan daya (power). Salah satu cara untuk menghubungkan daya dengan papan Arduino adalah dengan menghubungkannya ke port USB komputer, akan tetapi itu bukan solusi yang baik dalam beberapa kasus. Beberapa proyek tidak memerlukan komputer untuk menjalankannya, dan akan terasa berlebihan untuk menyalakan sebuah komputer dan menggunakan dayanya hanya untuk menyalakan Arduino. Port USB juga hanya memberikan 5 volt, dan kadang-kadang dibutuhkan lebih banyak daya.

Dalam situasi ini, solusi terbaik biasanya adalah adaptor AC yang memberikan tegangan 9V dengan adaptor ujung laras 2,1 mm dan pusat positif. Tancapkan ke power plug Arduino, dan Arduino dapat dinyalakan, bahkan ketika tidak terhubung ke komputer. Dengan cara itu, jika dihubungkan Arduino ke port USB, Arduino akan menggunakan daya eksternal jika tersedia.

Harap diperhatikan bahwa beberapa versi papan Arduino seperti Arduino NG dan Diecimila tidak beralih secara otomatis antara daya eksternal dan pasokan USB. Arduino tersebut dirancang dengan *jumper* seleksi listrik (perhatikan gambar 2.11) berlabel PWR_SEL dan harus mengaturnya secara manual untuk masing-masing EXT atau USB.

Sekarang diketahui dua cara untuk menyalurkan daya ke Arduino. Ketika Arduino dihubungkan dengan komputer, Arduino tidak mengambil semua daya yang ada, namun Arduino juga membagi daya tersebut dengan perangkat lain. Pada bagian gambar 2.7, dapat diperhatikan beberapa soket terkait dengan power supply :

- a. Menggunakan pin berlabel 3V3 dan 5V, mendapatkan daya perangkat eksternal terhubung ke Arduino dengan 3,3 volt atau 5 volt.
- b. Dua pin berlabel ground memungkinkan yang perangkat eksternal untuk berbagi kesamaan dengan Arduino.
- c. Beberapa proyek bersifat portabel, sehingga akan digunakan pasokan listrik portabel seperti baterai. Akan dihubungkan sumber daya eksternal seperti baterai ke Vin dan Gnd soket. Jika menghubungkan adaptor AC ke colokan (soket) listrik Arduino, dapat disalurkan tegangan dari adaptor melalui pin ini.

Pada bagian kanan bawah papan, ada enam pin input analog A0-A5. Ini dapat digunakan untuk menghubungkan sensor analog ke Arduino. Data sensor akan diambil dan merubahnya menjadi angka antara 0 dan 1023. Dibagian atas papan ada 14 pin digital IO (input-output) bernama D0-D13. Pin ini dapat difungsikan tergantung kebutuhan perancangan.

Pin-pin ini dapat digunakan untuk input atau output digital, sehingga nantinya dapat untuk menghidupkan dan mematikan LED. Pin-pin tersebut (D3, D5, D6, D9, D10, D11) juga dapat bertindak sebagai input/ output analog. Dalam mode ini, dapat dikonversi dari 0 sampai 255 angka digital menjadi tegangan analog. Semua pin ini dihubungkan ke mikrokontroler. Mikrokontroler menggabungkan CPU dengan beberapa fungsi periferal seperti saluran I/O. Ada banyak jenis mikrokontroler yang tersedia, tetapi biasanya

Arduino dilengkapi dengan Atmega328 atau Atmega168. Keduanya adalah mikrokontroler 8-bit yang diproduksi oleh sebuah perusahaan bernama Atmel.

Meskipun komputer modern memuat program dari hard drive, mikrokontroler biasanya harus diprogram terlebih dahulu. Itu berarti harus memuat perangkat lunak tersebut ke mikrokontroler melalui kabel. Hanya perlu mengunggahnya ke dalam mikrokontroler sekali saja. Program tersebut akan tetap berada di dalam mikrokontroler sampai mendapat program baru.

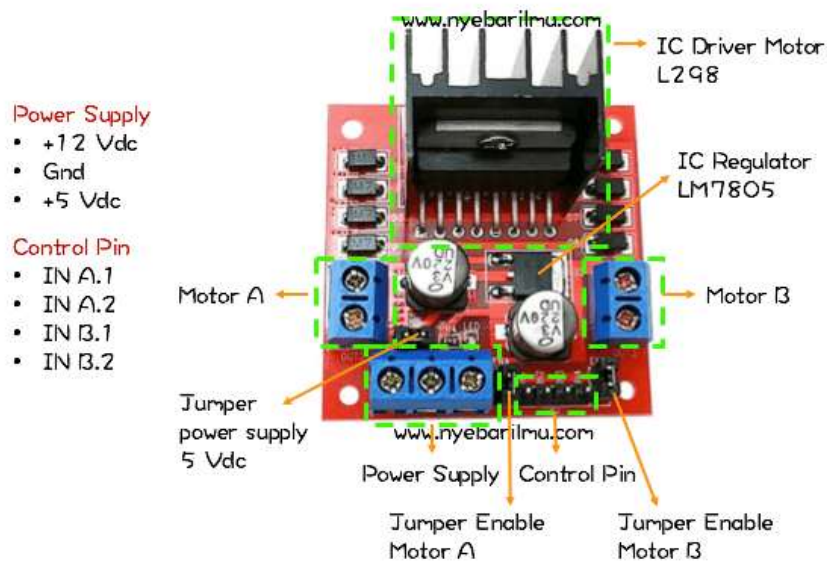
Setiap kali diberikan suplai listrik ke Arduino, program yang ada dalam mikrokontroler akan disimpan dalam mikrokontroler dan dijalankan secara otomatis. Jika menekan tombol reset pada Arduino, semuanya akan *reinitialized*. Program yang telah disimpan tidak akan hilang, namun berjalan dari awal.

2.6. Driver Motor L298N

Driver motor L298N merupakan module driver motor DC yang paling banyak digunakan atau dipakai di dunia elektronika yang difungsikan untuk mengontrol kecepatan serta arah perputaran motor DC.

IC L298 merupakan sebuah IC tipe H-bridge yang mampu mengendalikan beban-beban induktif seperti relay, solenoid, motor DC dan motor stepper. Pada IC L298 terdiri dari transistor-transistor logik (TTL) dengan gerbang NAND yang berfungsi untuk memudahkan dalam menentukan arah putaran suatu motor dc maupun motor stepper.

Untuk dipasaran sudah terdapat modul driver motor menggunakan IC L298 ini, sehingga lebih praktis dalam penggunaannya karena pin I/O nya sudah tersusun dengan rapi dan mudah digunakan. Kelebihan akan modul driver motor L298N ini yaitu dalam hal kepresisian dalam mengontrol motor sehingga motor lebih mudah untuk dikontrol. Adapun gambar pinout beserta keterangannya dapat diperhatikan pada gambar 2.8 di bawah.



Gambar 2.8 Pinout driver motor L298

Keterangan :

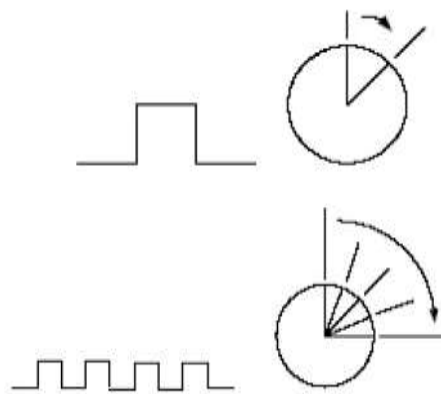
- Enable A : berfungsi untuk mengaktifkan bagian output motor A
- Enable B : berfungsi untuk mengaktifkan bagian output motor B
- Jumper 5 Vdc : sebagai mode pemilihan sumber tegangan 5Vdc, jika tidak di jumper maka akan ke mode sumber tegangan 12 Vdc
- Control Pin : Sebagai kendali perputaran dan kecepatan motor yang dihubungkan ke Mikrokontroler

Adapun untuk spesifikasi dari driver motor L298N dapat dijabarkan seperti berikut:

- Menggunakan IC L298N (Double H bridge Drive Chip)
- Tegangan minimal untuk masukan power antara 5V-35V
- Tegangan operasional : 5V
- Arus untuk masukan antara 0-36mA
- Arus maksimal untuk keluaran per Output A maupun B yaitu 2A
- Daya maksimal yaitu 25W
- Dimensi modul yaitu 43 x 43 x 26mm
- Berat : 26g

2.7. Motor Stepper

Motor stepper adalah motor listrik yang dikendalikan dengan pulsa-pulsa digital, bukan dengan memberikan tegangan yang terus-menerus. Deretan pulsa diterjemahkan menjadi putaran shaft, dimana setiap putaran membutuhkan jumlah pulsa yang ditentukan. Satu pulsa menghasilkan satu kenaikan putaran atau *step*, yang merupakan bagian dari satu putaran penuh. Oleh karena itu, perhitungan jumlah pulsa dapat diterapkan untuk mendapatkan jumlah putaran yang diinginkan. Perhitungan pulsa secara otomatis menunjukkan besarnya putaran yang telah dilakukan, tanpa memerlukan informasi balik (*feedback*).



Gambar 2.9 Jumlah pulsa mewakili jumlah putaran

Ketepatan kontrol gerak motor stepper terutama dipengaruhi oleh jumlah *step* tiap putaran yang semakin banyak jumlah *step*, semakin tepat gerak yang dihasilkan. Untuk ketepatan yang lebih tinggi, beberapa *driver* motor stepper membagi *step* normal menjadi setengah *step* (*half step*) atau *mikro step*.

Bagian-bagian dari motor stepper yaitu tersusun atas rotor, stator, bearing, casing dan sumbu. Sumbu merupakan pegangan dari rotor dimana sumbu merupakan bagian tengah dari rotor, sehingga ketika rotor berputar sumbu ikut berputar. Stator memiliki dua bagian yaitu pelat inti dan lilitan. Plat inti dari motor stepper ini biasanya menyatu dengan casing. Casing motor stepper terbuat dari aluminium dan ini berfungsi sebagaiudukan bearing dan stator pemegangnya adalah boud sebanyak empat buah. Di dalam motor stapper memiliki dua buah bearing yaitu bearing bagian atas dan bearing bagian bawah.

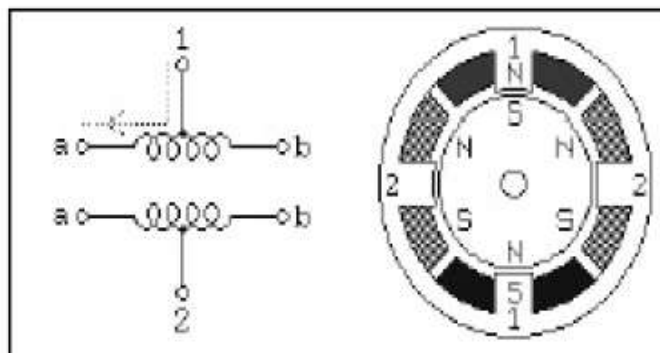


Gambar 2.10 Bearing dalam motor stepper

Dilihat dari lilitannya, motor stepper terbagi menjadi 2 jenis yaitu motor stepper unipolar dan motor stepper bipolar.

a. Motor stepper unipolar

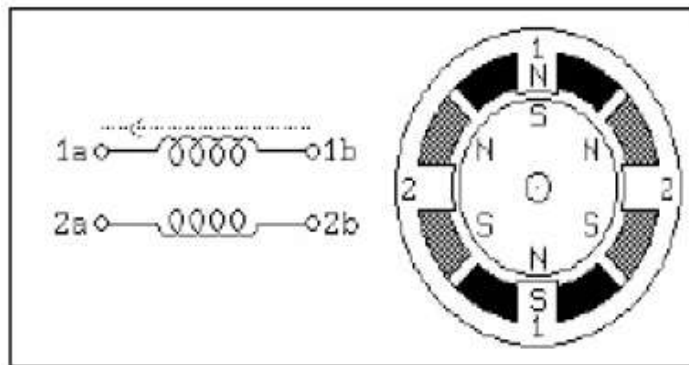
Motor stepper unipolar terdiri dari dua lilitan yang memiliki center tap. Center tap dari masing masing lilitan ada yang berupa kabel terpisah ada juga yang sudah terhubung didalamnya sehingga center tap yang keluar hanya satu kabel. Untuk motor stepper yang center tapnya ada pada masing – masing lilitan kabel inputnya ada 6 kabel. Namun jika center tapnya sudah terhubung di dalam kabel inputannya hanya 5 kabel. Center tap dari motor stepper dapat dihubungkan ke pentanahan atau ada juga yang menghubungkannya ke +VCC hal ini sangat dipengaruhi oleh driver yang digunakan. Sebagai gambaran dapat dilihat konstruksi motor stepper unipolar pada gambar 2.11 berikut.



Gambar 2.11 Motor stepper unipolar

b. Motor stepper bipolar

Motor stepper bipolar memiliki dua lilitan perbedaaan dari tipe unipolar adalah bahwa pada tipe bipolar lilitannya tidak memiliki center tap. Keunggulan tipe bipolar yaitu memiliki torsi yang lebih besar jika dibandingkan dengan tipe unipolar untuk ukuran yang sama. Pada motor stepper tipe ini hanya memiliki empat kabel masukan. Namun ntuk menggerakkan motor stepper tipe ini lebih rumit jika dibandingkan dengan menggerakkan motor stepper tipe unipolar. Sebagai gambaran dapat dilihat konstruksi motor stepper bipolar pada gambar 2.12 berikut:



Gambar 2.12 Motor stepper bipolar

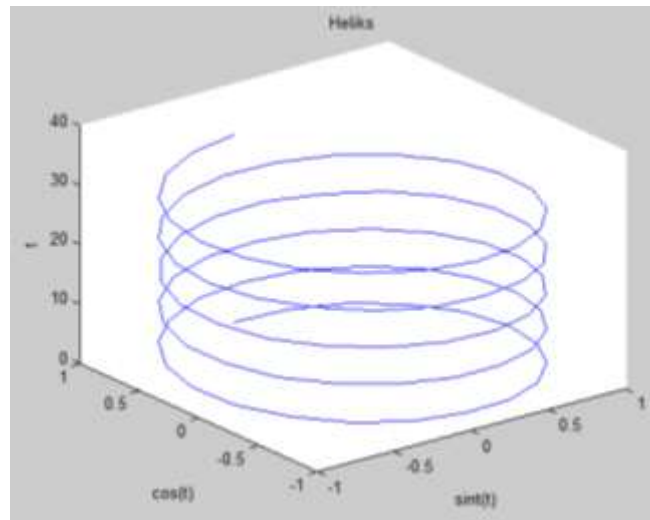
2.8. Grafik Tiga Dimensi pada MATLAB

Perintah plot dari dunia dua dimensi disempurnakan oleh perintah plot3 untuk bekerja dalam tiga dimensi. Format yang digunakan sama dengan perintah plot untuk dua dimensi, kecuali data yang digunakan adalah tiga satuan, bukan sepasang. Format umum dari plot3 adalah `plot3(x1,y1,z1,S1,x2,y2,z2,S2,...)` dengan x_n dan y_n adalah vektor atau matriks, sedangkan S_n adalah karakter string opsional yang mengatur warna, simbol penanda, atau style garis. Di bawah ini adalah contoh sebuah heliks tiga dimensi. Berikut *script* kodenya seperti gambar 2.13 dibawah.

```
t=linspace(0,10*pi);  
plot3(sin(t),cos(t),t);  
title('Heliks'),xlabel('sint(t)'),y  
label('cos(t)'),zlabel('t')
```

Gambar 2.13 *script* kode heliks tiga dimensi

Adapun hasil menjalankan *script* kode diatas yaitu seperti pada gambar 2.14 dibawah.



Gambar 2.14 hasil dari *script* kode heliks tiga dimensi

2.9. Moving Average

Moving Average merupakan *filter* yang paling umum digunakan dalam Sinyal Pengolahan Digital, terutama karena ini merupakan *filter* digital termudah untuk dipahami dan digunakan. *Filter Moving Average* ini optimalnya untuk mengurangi gangguan acak dengan mempertahankan respon langkah yang tajam. Ini membuat *filter* ini menjadi filter utama untuk domain waktu yang disandikan sinyal. Namun, *Moving Average* merupakan *filter* yang buruk untuk sinyal domain frekuensi, dengan sedikit kemampuan untuk memisahkan satu band frekuensi dari frekuensi lainnya.

Seperti namanya, *filter* ini beroperasi dengan rata-rata angka sinyal masukan untuk menghasilkan setiap titik dalam sinyal keluaran. Persamaannya :

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i+j] \quad (2-5)$$

Diatas merupakan persamaan *filter Moving Average*. Pada $x[]$ merupakan sinyal masukan, $y[]$ merupakan sinyal keluaran, dan M merupakan periode yang digunakan dalam *moving average*. Sebagai contoh, dibawah ini merupakan *filter Moving Average* dengan 5 periode.

$$y[80] = \frac{x[80] + x[81] + x[82] + x[83] + x[84]}{5}$$