BAB II

DASAR TEORI

2.1 Sistem Kendali

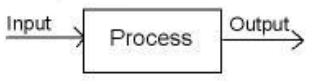
Sistem Kendali atau *control system* terdiri dari dua kata yaitu *system* dan *control*. *System* berasal dari Bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk mencapai suatu tujuan tertentu. *Control* itu memiliki arti mengatur, mengarah dan mengendalikan. Jadi *system control* adalah hubungan antara komponen-komponen fisik yang membentuk suatu konfigurasi sistem sehingga memberikan hasil yang diharapkan yang dapat dipraktekkan secara otomatis. Pengontrolan atau pengendalian perangkat menggunakan *Inoduino Board* dimana *Inoduino Board* ini mengontrol atau mengendalikan seluruh perangkat, misal motor Servo, motor DC, *Liquid Crystal Display* 16x2 (LCD) serta sensor *Load Cell*.

2.1.1 Jenis-Jenis Sistem Kendali

Jenis-jenis sistem kendali terdiri dari 2 macam yaitu sistem *open loop* dan *close loop.*

2.1.1.1 Sistem *Open Loop*

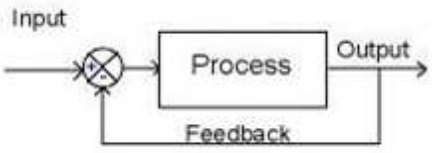
Sistem kendali *open loop* adalah sebuah sistem yang tidak memiliki umpan balik (*feedback*), sehingga bila terdapat gangguan dari dalam maupun dari luar maka sistem tidak dapat melaksanakan tugas seperti yang diharapkan. Suatu sinyal masukan diberikan ke dalam sistem kendali dimana keluarannya bertindak sebagai sinyal penggerak dimana sinyal penggerak ini yang kemudian menghasilkan proses yang akan dikendalikan untuk menghasilkan *output* yang diinginkan. Proses *open loop* dapat dilihat pada gambar 2.1.



Gambar 2.1 Sistem *Open Loop*

2.1.1.2 Sistem *Close Loop*

Sistem kendali *close loop* adalah sistem kontrol yang memiliki *feedback*, berbeda dengan sistem *open loop*. Pada bagian *output* dari sistem kontrol ini akan dikirim kembali untuk dibandingkan dengan *input* yang diberikan. Bila masih terdapat selisih antara *output* dan *input*, maka sistem masih memiliki *error* . *Error* merupakan selisih antara *input* dan *output* atau sistem belum mencapai hasil yang diinginkan. Proses kerja dari sistem *close loop* dapat dilihat pada gambar 2.2.

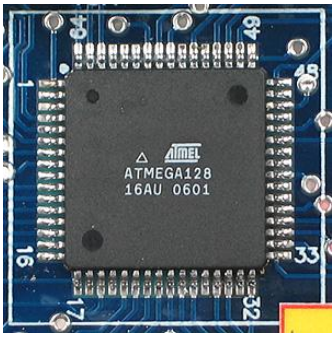


Gambar 2.2 Sistem *Close Loop*

2.2 Perangkat Keras

2.2.1 Mikrokontroler

Mikrokontroler merupakan rangkaian terintegrasi yang berisi komponen-komponen yang diperlukan oleh sebuah komputer seperti CPU, I/O, jalur komunikasi, memori, *timer* dan lain sebagainya. Mikrokontroler dapat diberi sebuah program yang bekerja sesuai dengan keinginan pengguna dan dapat dihapus dengan cara khusus. [1]



Gambar 2.3 ATMEGA128

Beberapa fitur yang terdapat pada mikrokontroller :

1. CPU (*Central Procesing Unit*)

   CPU adalah suatu unit pengolahan pusat yang terdiri dari 2 bagian, yakni unit pengendali (*Control Unit*) dan logika (*Arithmetic Logic Unit*). Disamping itu, CPU mempunyai beberapa simpanan yang berukuran kecil yang disebut dengan *register*.

2. Bus Alamat

   Bus alamat berfungsi sebagai jumlah lintasan saluran pengalamatan. Pengalamatan ini harus ditentukan terlebih dahulu untuk menghindari terjadinya kesalahan pengiriman sebuah intruksi dan terjadinya tabrakan antara dua buah alat yang bekerja secara bersamaan.

3. Bus Data

   Bus data merupakan lintasan saluran keluaran masuknya data dalam suatu mikrokontroler. Umumnya saluran data yang masuk sama dengan saluran data yang keluar.

4. Bus *control*

   Bus *control* atau bus kendali berfungsi untuk mengintegrasi operasi mikrokontroler dengan operasi rangkaian luar.

5. Memori

   Memori berfungsi untuk menyimpan data atau program. Ada beberapa jenis memori, diantaranya adalah ROM (*Read Only Memory*) dan RAM (*Random Access Memory*) serta berdasarkan tingkatannya, memori terbagi menjadi register internal, memori utama dan memori masal. Register internal adalah memori yang terdapat di dalam ALU (*Arithmatic Logic Unit*). Memori utama adalah memori yang ada pada suatu sistem, waktu aksesnya lebih lambat dibandingkan dengan *register* internal. Sedangkan memori masal yang dipakai untuk menyimpan daya yang berkapasitas tinggi, yang biasanya berbentuk disket, pita magnetik atau kaset.

6. RAM

   RAM adalah memori yang dapat dibaca atau ditulis. Data dalam RAM bersifat *volatile*, dimana isinya akan hilang begitu IC kehilangan

catu daya. Karena bersifat yang demikian, RAM hanya digunakan untuk menyimpan data pada saat program bekerja.

7. ROM

   ROM merupakan memori yang hanya dapat dibaca, dimana isinya tidak dapat dirubah apabila IC telah kehilangan catu daya. ROM dipakai untuk menyimpan program, pada saat direset maka mikrokontroler akan langsung bekerja dengan program yang terdapat di dalam ROM tersebut. Ada berbagai jenis ROM antara lain ROM murni, PROM (*Programmable Read Only Memory*), yang paling banyak digunakan diantara jenis-jenis diatas adalah EPROM yang dapat diprogram ulang dan dapat juga dihapus dengan sinar *ultraviolet*.

8. Perangkat *Input/Output*

   Perangkat *input* dan *output* mikrokontroler adalah suatu peranti yang menghubungkan proses didalam mikrokontroler dengan dunia luar (rangkaian lain), peranti ini dibutuhkan sebagai media komunikasi dengan perangkat lain atau peubah tipe sinyal. Contoh perangkat I/O antara lain:

   1. USART (*Universal Synchronous Asynchronous Receiver Transmitter*) merupakan adapter yang digunakan sebagai komunikasi *serial sinkron* dan *asinkron*.
   2. SPI (*Serial Peripheral Interface*) merupakan *port* komunikasi *serial sinkron*.
   3. I2C *bus* (*Inter-Itergrated Circuit Bus*) merupakan antarmuka *serial bus* yang dikembangkan oleh *philihps*.
   4. *Analog to Digital Conversion (ADC)* adalah rangkaian yang digunakan untuk mengubah data *analog* ke data *digital*.
   5. *Digital to Analog*(DAC) adalah rangkaian untuk mengubah data *digital* ke data *analog*.
   6. *Analog Comparator* adalah rangkaian komparator tegangan, saat ini peranti ini telah terintegrasi dalam *port* mikrokontroller.
   7. USB *Converter* merupakan peranti yang memungkinkan komunikasi *serial* dengan karakteristik format komunikasi USB.

9. Clock

Clock atau pewaktu berfungsi memberikan referensi waktu dan sinkronisasi antar elemen.

Inoduino Board merupakan sebuah modul mikrokontroler *singlechip* berbasis mikrokontroler AT90USB1286 yang kompatibel dengan Arduino. berikut spesifikasi *Inoduino Board* :

1. Berbasis mikrokontroller AT90USB1296 dengan *flash memory* sebesar 124 KB (total 128 KB, 4 KB telah digunakan untuk *bootloader*) dan EEPROM 4 KB.
2. Dilengkapi dengan program *bootloader* yang dapat diprogram menggunakan *software* Arduino.
3. Memiliki 39 jalur *input*/ *output* + 8 *channel* ADC 10 bit.
4. Tersedia jalur komunikasi serial USB, sekaligus sebagai jalur untuk pemrograman mikrokontroler.
5. Frekuensi osilator sebesar 16 MHz.
6. Tersedia rangkaian *reset* manual.
7. Tersedia tombol HWB untuk masuk ke dalam mode *bootloader*.
8. Dilengkapi dengan regulator 3.3 Volt dan 5 Volt dengan arus maksimum 800 mA.
9. Membutuhkan catu daya *input* : 6.5 s/d 12 VDC.

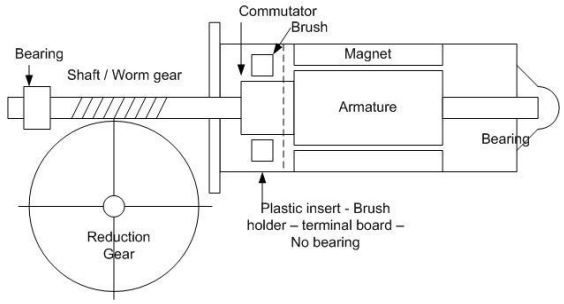


Gambar 2.4 *Inoduino Board*

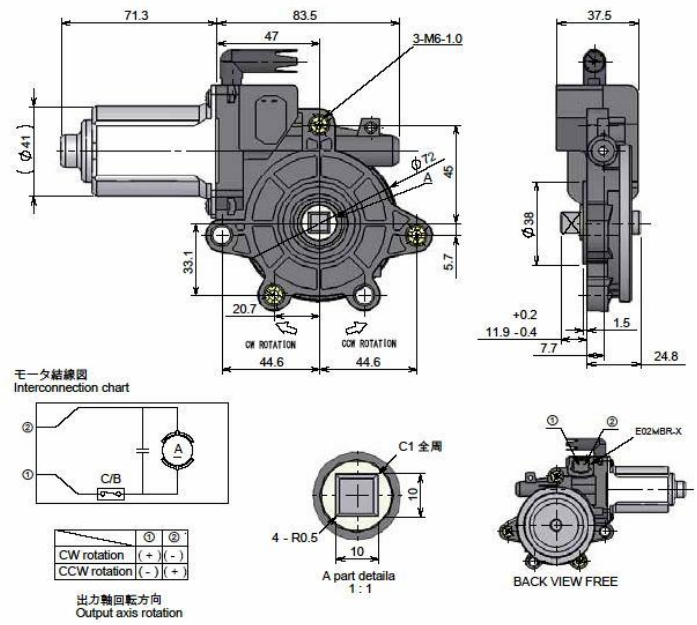
2.3 Jenis-Jenis Motor

2.3.1 Motor DC (*Power Window*)

Motor DC (*Power Window*) adalah suatu motor yang mengubah energi listrik searah menjadi mekanis yang berupa tenaga penggerak torsi. Motor DC digunakan dimana kontrol kecepatan dan kecepatan torsi diperlukan untuk memenuhi kebutuhan. Bagian DC yang paling penting adalah *rotor* dan *strator*. Bagian *stator* adalah badan motor, sikat-sikat dan inti kutub magnet. Bagian *rotor* adalah bagian yang berputar dari suatu motor DC. Yang termasuk *rotor* ialah lilitan jangkar, komutator, tali, *isolator*, poros, bantalan dan kipas. Jenis motor dc yang dipergunakan dapat dilihat pada gambar dibawah ini. [5]



Gambar 2.5 Dasar *Power Window*



Gambar 2.6 Kontruksi Motor *Power Windows*

Motor *power window* banyak dipergunakan karena torsi tinggi dengan rating tegangan input yang rendah yaitu 12 VDC dan dimensi motor yang relatif sederhana dilengkapi dengan interval *gearbox* sehingga memudahkan untuk instalasi mekanik. Prinsip kerja motor DC *power window* mempunyai bagian *stator* yang berupa magnet permanen dan bagian yang bergerak *rotor* yang berupa koil atau gulungan kawat tembaga. Dimana setiap ujungnya tersambung dengan komutator ini dihubungkan dengan kutub *positive* (+) dan kutub *negative* (-) dari catu daya.

Prinsip kerja dari arus searah adalah membalik phasa tegangan dari gelombang yang mempunyai nilai positif dengan menggunakan komutator, dengan demikian arus yang berbalik arah dengan kumparan jangkar yang berputar dalam medan magnet. Bentuk motor dc paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen. Catu tegangan dc dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan pada gambar 2.3 disebut *Angker dynamo*. *Angker dynamo* adalah sebutan untuk komponen yang berputar di antara medan magnet. Motor DC memiliki 2 bagian dasar :

1. Bagian yang tetap atau stasioner yang disebut stator. stator ini menghasilkan medan magnet baik yang dibangkitkan dari sebuah koil (elektro magnet) ataupun magnet permanen.
2. Bagian yang berputar disebut rotor. Rotor ini berupa sebuah koil dimana arus listrik mengalir.

Gaya elektromagnetik pada motor DC timbul saat ada arus yang mengalir pada penghantar yang berada dalam medan magnet. Medan magnet itu sendiri ditimbulkan oleh magnet permanen. Garis-garis gaya magnet mengalir diantara dua kutub magnet dari kutub utara ke kutub selatan. Menurut hukum gaya *Lourentz*, arus yang mengalir pada penghantar yang terletak dalam medan magnet akan menimbulkan gaya . Gaya (F), timbul tergantung pada arah arus (I), dan arah medan magnet (B).

Karena putaran *rotor*, arus listrik di dalam kawat akan berjalan bolak-balik karena jalannya sesuai dengan medan magnet, maka *rotor* akan selalu berputar terus menerus selama arus listrik tetap mengalir di dalam kawat. Untuk karakteristik pada motor dc antara lain:

1. Kutub Medan Secara langsung sederhana digambarkan bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor dc. Motor DC memiliki kutub medan yang stasioner dan *dynamo* yang menggerakkan *bearing* pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub medan yaitu utara dan selatan. Garis *magneticenergy* membesar melintasi bukan di antara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih komplek terdapat satu atau lebih *electromagnet*. *Electromagnet* menerima listrik dari sumber daya dari luar sebagai penyedia struktur medan.
2. *Dynamo* Bila arus masuk menuju *dynamo*, maka arus ini akan menjadi *electromagnet*. *Dynamo* yang terbentuk silinder, dihubungkan ke as penggerak untuk menggerakan beban. Untuk kasus motor DC yang kecil, *dynamo* berputar dalam medan magnet yang terbentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi. Jika hal ini terjadi, arusnya berbalik untuk merubah kutub-kutub utara dan selatan *dynamo*.
3. *Commutator* Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikan arah arus listrik dalam *dynamo*. *Commutato*r juga membantu dalam transmisi arus antara *dynamo* dan sumber daya. Keuntungan utama motor DC adalah sebagai pengendali kecepatan yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur tegangan *dynamo* (meningkatkan tegangan *dynamo* yang dimana akan meningkatkan kecepatan), arus medan (menurunkan arus medan akan meningkatkan kecepatan).

2.3.2 Motor Stepper

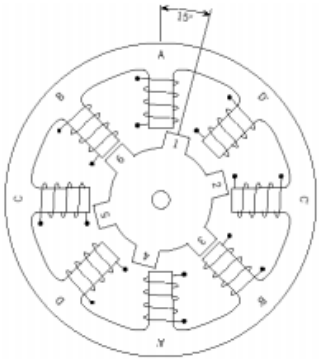
Motor *stepper* adalah perangkat elektromekanis yang bekerja dengan mengubah pulsa elektronis menjadi gerakan mekanis diskrit. Motor *stepper* bergerak berdasarkan urutan pulsa yang diberikan kepada motor. Karena itu, untuk menggerakkan motor *stepper* diperlukan pengendali motor *stepper* yang membangkitkan pulsa-pulsa periodik. Penggunaan motor *stepper* memiliki beberapa keunggulan dibandingkan dengan penggunaan motor DC biasa. Keunggulannya antara lain adalah :

1. Sudut rotasi motor proposional dengan pulsa masukan sehingga lebih mudah diatur.
2. Motor dapat langsung memberikan torsi penuh pada saat mulai bergerak.
3. Posisi dan pergerakan repetisinya dapat ditentukan secara presisi.
4. Memiliki respon yang sangat baik terhadap mulai, berhenti dan berbalik (perputaran).
5. Sangat relibel karena tidak adanya sikat yang bersentuhan dengan rotor seperti pada motor DC.
6. Dapat menghasilkan perputaran yang lambat sehingga beban dapat dikopel langsung ke porosnya.
7. Frekuensi perputaran dapat ditentukan secara bebas dan mudah pada range luas.

Pada dasarnya terdapat 3 tipe motor *stepper* yaitu :

1. Motor *stepper* tipe *variable reluctance* (VR)

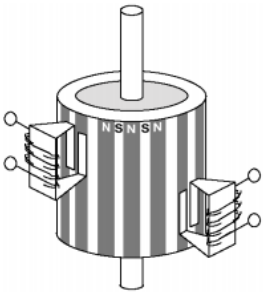
   Motor *stepper* jenis ini telah lama ada dan merupakan jenis motor yang secara struktural paling mudah untuk dipahami. Motor ini terdiri atas sebuah *roto*r besi lunak dengan beberapa gerigi dan sebuah lilitan stator. Ketika lilitan *stato*r diberi energi dengan arus DC, kutub-kutubnya menjadi termagnetasi. Perputaran terjadi ketika gigi-gigi rotor tertarik oleh kutub-kutub *stator*. Berikut ini adalah penampang melintang dari motor *stepper* tipe *variable reluctance* (VR):



Gambar 2.7 Penampang Melintang Dari Motor *Stepper* tipe VR

2. Motor *stepper* tipe *Permanent Magnet* (PM)

Motor *stepper* jenis ini memiliki *rotor* yang berbentuk seperti kaleng bundar (*tin can*) yang terdiri atas lapisan magnet permanen yang diselang-seling dengan kutub yang berlawanan (perhatikan gambar 2.8). Dengan adanya magnet permanen, maka intensitas *fluks* magnet dalam motor ini akan meningkat sehingga dapat menghasilkan torsi yang lebih besar. Motor jenis ini biasanya memiliki resolusi langkah (*step*) yang rendah yaitu antara 7,50 hingga 150 per langkah atau 48 hingga 24 langkah setiap putarannya. Berikut ini adalah ilustrasi sederhana dari motor *stepper* tipe *permanent magnet:*

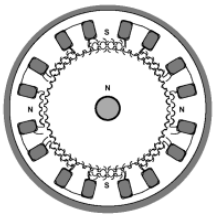


Gambar 2.8 Ilustrasi Motor *Stepper* tipe *P*M

3. Motor s*tepper* tipe *Hybrid* (HB)

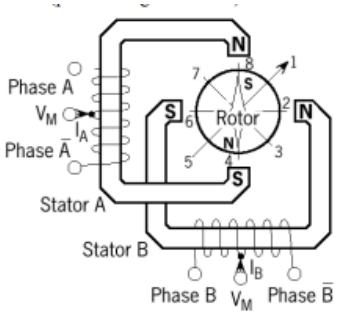
Motor *stepper* tipe hibrid memiliki struktur yang merupakan kombinasi dari kedua tipe motor *stepper* sebelumnya. Motor *stepper* tipe hibrid memiliki gigi-gigi seperti pada motor tipe VR dan juga memiliki

magnet permanen yang tersusun secara aksial pada batang porosnya seperti motor tipe PM. Motor tipe ini paling banyak digunakan dalam berbagai aplikasi karena kinerja lebih baik. Motor tipe hibrid dapat menghasilkan resolusi langkah yang tinggi yaitu antara 3,60 hingga 0,90 per langkah atau 100-400 langkah setiap putarannya. Berikut ini adalah penampang melintang dari motor *stepper* tipe hibrid:



Gambar 2.9 Penampang Melintang Motor *Stepper* tipe *hybrid*

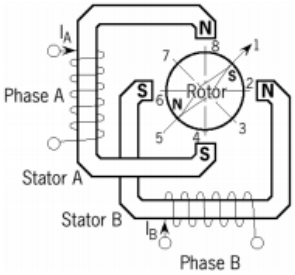
Berdasarkan metode perancangan rangkain pengendalinya, motor *stepper* dapat dibagi menjadi jenis unipolar dan bipolar. Rangkaian pengendali motor *stepper* unipolar lebih mudah dirancang karena hanya memerlukan satu *switch* atau transistor setiap lilitannya. Untuk menjalankan dan menghentikan motor ini cukup dengan menerapkan pulsa digital yang hanya terdiri atas tegangan positif dan nol (*ground*) pada salah satu terminal lilitan (*wound*) motor sementara terminal lainnya dicatu dengan tegangan positif konstan (VM) pada bagian tengah (*center tap*) dari lilitan.



Gambar 2.10 Motor *Stepper* Dengan Lilitan Unipolar

Untuk motor *stepper* dengan lilitan bipolar, diperlukan sinyal pulsa yang berubah-ubah dari positif ke negatif dan sebaliknya. Jadi pada setiap terminal lilitan (A & B) harus dihubungkan dengan sinyal yang mengayun dari positif ke

negatif dan sebaliknya (perhatikan gambar 2.11). Karena itu dibutuhkan rangkaian pengendali yang agak lebih kompleks daripada rangkaian pengendali untuk motor unipolar. Motor *stepper* bipolar memiliki keunggulan dibandingkan dengan motor *stepper* unipolar dalam hal torsi yang lebih besar untuk ukuran yang sama.

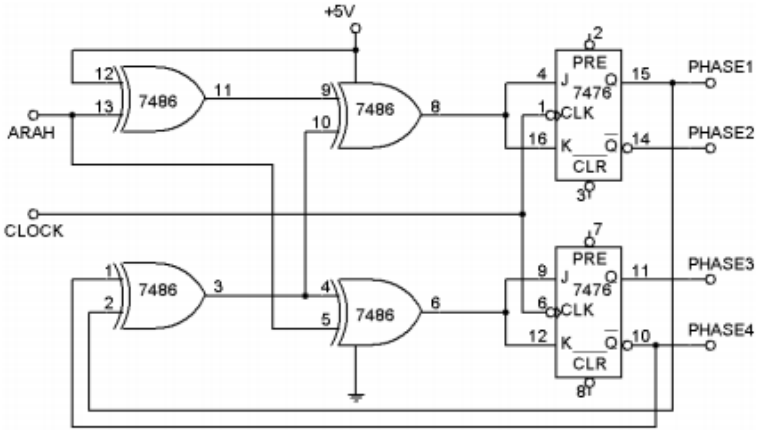


Gambar 2.11 Motor *Stepper* Dengan Lilitan Bipolar

2.3.2.1 Pengendali Motor *Stepper*

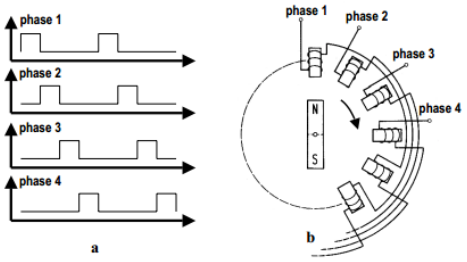
Berikut ini akan diberikan contoh perancangan dan perhitungan rangkaian pengendali motor *stepper* sederhana. Motor *stepper* yang digunakan pada contoh ini dengan tipe hibrid unipolar, memiliki empat fasa dan panjang langkah sebesar 1,80 per langkah. Motor diharapkan dapat berputar dalam dua arah dan memiliki dua kecepatan. Karena itu diperlukan pengendali motor *stepper* yang memiliki empat keluaran pulsa dengan kemampuan dua arah perputaran dan dua macam frekuensi pulsa guna mengatur kecepatan motor.

Rangkaian pengendali motor *stepper* (*stepper motor driver*) menggunakan komponen utama berupa sebuah IC logika XOR (74LS86) dan sebuah IC JK *flip-flop* (74LS76). Rangkain dengan kedua IC tersebut berfungsi untuk menghasilkan empat pulsa keluaran berurutan yang dapat berbalik urutannya dengan menerapkan logika tertentu pada rangkaian. Rangkaian tersebut memerlukan pulsa *clock* untuk dapat beroperasi. Sebagai sumber *clock* digunakan rangkaian berbasis IC *timer* 555. Rangkain pembangkit *clock* ini dapat menghasilkan dua macam frekuensi pulsa keluaran guna mendukung dua kecepatan motor *stepper*. Kemudian untuk mendukung pulsa-pulsa dengan arus besar (sekitar 1 - 3 A) digunakan transistor daya NPN tipe TIP31 sebagai *solid state switch*. Untuk lebih jelasnya perhatikan rangkaian utama dari pengendali motor *stepper* lihat pada gambar 2.12



Gambar 2.12 Skema Rangkaian Pengendali Motor *Stepper*

Pada Gambar 2.12 adalah skema rangkaian pengendali motor *stepper* yang dapat bergerak ke dua arah. Keluaran pengendali motor *stepper* ini ada empat (pin15, 14, 11, 10 dari IC 74LS76). Pin-pin tersebut akan menghasilkan pulsa yang dapat menggerakkan motor *stepper*. Berikut ini adalah ilustrasi struktur motor *stepper* sederhana dan pulsa yang dibutuhkan untuk menggerakkannya.[6]



Gambar 2.13 Bentuk Pulsa Keluaran Dan Penerapan Pulsa Pengendali Motor

Arah putaran motor dapat diatur dengan mengatur kondisi logika masukan pada pin EN (Enable, EN 1.2, EN 3.4) dari IC L293D. Jika diterapkan logika 0, maka motor akan berputar berlawanan dengan arah jarum jam (*counter clock wise*) sedangkan jika diterapkan logika 1, maka motor akan berputar dengan arah sesuai dengan arah jarum jam (*clockwise*). Gambar 2.13 di atas adalah contoh bentuk pulsa keluaran yang menggerakkan motor *stepper* pada arah sesuai dengan jarum jam (*clockwise*) (Gambar 2.13).

2.3.3 Motor Servo

Motor Servo adalah sebuah motor dengan sistem umpan balik tertutup di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor DC, serangkaian *gear*, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Karena motor servo merupakan alat untuk mengubah energi listrik menjadi energi mekanik, maka magnet permanen motor servo yang mengubah energi listrik kedalam energi mekanik melalui interaksi dari dua medan magnet. Salah satu medan dihasilkan oleh magnet permanen dan yang satunya dihasilkan oleh arus yang mengalir dalam kumparan motor. Resultan dari dua medan magnet tersebut menghasilkan torsi yang membangkitkan putaran motor tersebut. Saat motor berputar, arus pada kumparan motor menghasilkan torsi yang nilainya konstan.

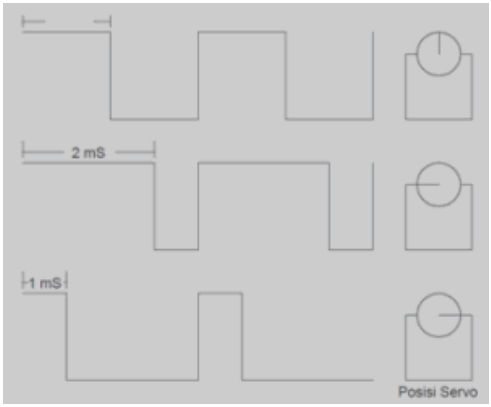
Secara umum terdapat 2 jenis motor servo. yaitu motor servo *standard* motor servo *Continous*. Servo motor tipe standar hanya mampu berputar 90 derajat dan 180 derajat, motor servo standar sering dipakai pada sistem robotika misalnya untuk membuat *RobotArm* (Robot Lengan). Sedangkan jenis servo motor *Continous* dapat berputar sebesar 360 derajat. Motor servo *Continous* sering dipakai untuk *MobileRobot*.

Motor servo merupakan sebuah motor dc kecil yang diberi sistem *gear* dan potensiometer sehingga dia dapat menempatkan posisi yang dikehendaki. Karena motor ini menggunakan *system Close Loop* sehingga posisi yang dikehendaki bisa dipertahankan.



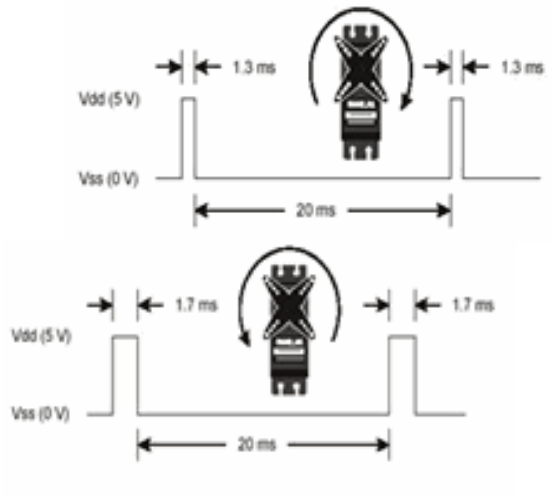
Gambar 2.14 Motor Servo

Pengendalian gerakan batang motor servo dapat dilakukan dengan menggunakan metode PWM (*Pulse Width Modulation*). Teknik ini menggunakan sistem lebar pulsa untuk mengemudikan putaran motor. Sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Tampak pada gambar 2.15 dengan pulsa 1.5 ms pada periode selebar 2 ms maka sudut dari sumbu motor akan berada pada posisi tengah. Semakin lebar pulsa OFF maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa OFF maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan arah jarum jam.



Gambar 2.15 Posisi Servo

Untuk menggerakkan motor servo ke kanan atau ke kiri, tergantung dari nilai *delay* yang kita berikan. Untuk membuat servo pada posisi *center*, berikan pulsa 1.5ms. Untuk memutar servo ke kanan, berikan pulsa ≤ 1.3ms, dan pulsa ≥ 1.7ms untuk berputar ke kiri dengan *delay* 20ms, seperti ilustrasi pada gambar 2.16



Gambar 2.16 Prinsip Kerja Servo

2.4 *Driver Relay* Motor DC

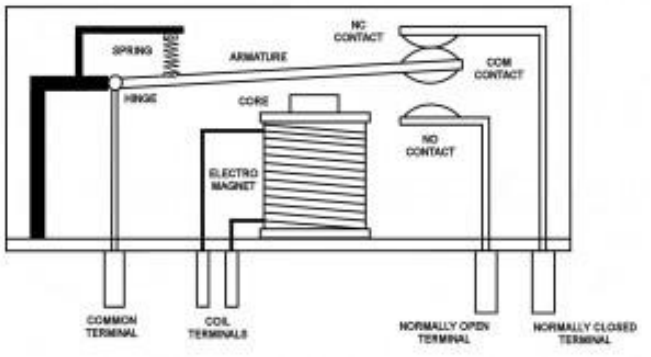
2.4.1 *Relay*

Relay adalah saklar mekanik yang dikendalikan atau dikontrol secara elektronik (elektro magnetik). Saklar pada *relay* akan terjadi perubahan posisi OFF ke ON pada saat diberikan energi elektro magnetik pada *armature* r*elay* tersebut. *Relay* pada dasarnya terdiri dari 2 bagian utama yaitu saklar mekanik dan sistem pembangkit elektromagnetik (induktor inti besi). Saklar atau kontaktor *relay* dikendalikan menggunakan tegangan listrik yang diberikan ke induktor pembangkit magnet untuk menarik tuas saklar atau kontaktor *relay*. *Relay* yang ada di pasaran terdapat berbagai bentuk dan ukuran dengan tegangan kerja dan jumlah saklar yang bervariasi.[1]

Relay dibutuhkan dalam rangkaian elektronika sebagai eksekutor sekaligus *interface* antara beban dan sistem kendali eletronik yang berbeda sistem *power supply*. Secara fisik antara saklar atau kontaktor dengan elektromagnet *relay* terpisah sehingga antara beban dan sistem kontrol terpisah. Bagian utama *relay* elektro mekanik adalah kumparan elektromagnet, saklar atau kontaktor, *swing armature dan spring* (pegas).

Relay elektromagnet dibagi menjadi 2 jenis yaitu :

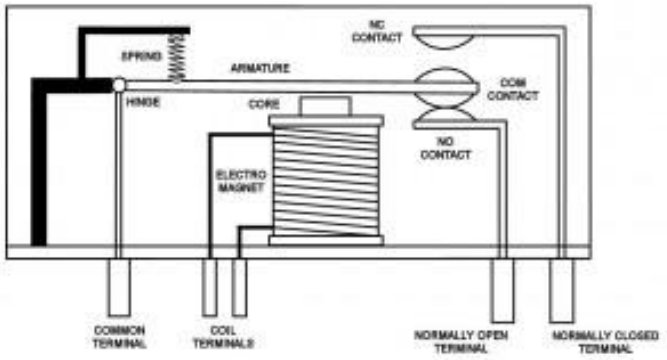
1. *Relay* Elektro Mekanik Posisi NC (*Normally Close*)



Gambar 2.17 *Relay* Elektro Mekanik Posisi NC (*Normally Close*)

Kontruksi *relay* elektro mekanik diatas dapat diuraikan sistem kerja atau proses *relay* bekerja. Pada saat elektromagnet tidak diberikan sumber tegangan maka tidak ada medan magnet yang menarik *armature*, sehingga saklar *relay* tetap terhubung ke terminal NC (*Normally Close*) seperti terlihat pada gambar 2.17. Kemudian pada saat elektromagnet diberikan sumber tegangan maka terdapa medan magnet yang menarik *armature*, sehingga saklar terhubung ke terminal NO (*Normally Open*).

2. *Relay* Elektro Mekanik Posisi NO (*Normally Open*)



Gambar 2.18 *Relay* Elektro Mekanik Posisi NO (*Normally Open*)

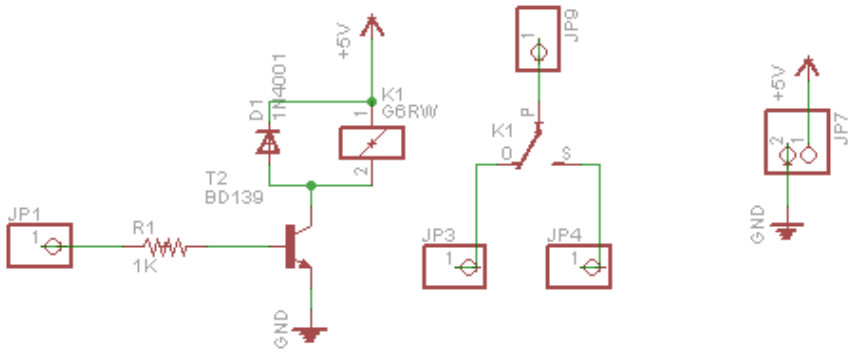
Relay elektro mekanik memiliki kondisi saklar atau kontaktor dalam 3 posisi. Ketiga posisi saklar atau kontaktor relay ini akan berubah pada saat *relay* mendapat tegangan sumber pada elektromagnetnya. ketiga posisi saklar tersebut adalah :

1. Posisi *Normally Open* (NO) yaitu posisi saklar *relay* yang terhubung ke terminal NO (*Normally Open*). Kondisi ini akan terjadi pada saat *relay* mendapat tegangan sumber pada elektromagnet.
2. Posisi *Normally Close* (NC) yaitu posisi saklar yang terhubung ke terminali NC (*Normally Close*). Kondisi ini akan terjadi pada saat *relay* tidak mendapat tegangan sumber pada elektromagnet.
3. Posisi *Change Over* (CO) yaitu kondisi perubahan armature saklar *relay* yang berubah dari posisi NC ke NO atau sebaliknya dari NO ke NC. Kondisi ini akan terjadi saat sumber tegangan diberikan ke elektromagnet atau saat sumber tegangan diputus dari elektromagnet *relay*.

Jenis *relay* dan desain saklar *relay* yang dapat digunakan antra lain:

1. *Single Pole Single Throw* (SPST), relay ini memiliki 4 terminal yaitu 2 terminal untuk input kumparan elektromagnet dan 2 terminal saklar, *relay* ini hanya memiliki posisi NO (*Normally Open*) saja.
2. *Single Pole Double Throw* (SPDT), relay ini memiliki 5 terminal yaitu terdiri dari 2 terminal untuk *input* kumparan elektromagnetik dan 3 terminal saklar. *Relay* jenis ini memiliki 2 kondisi NO dan NC.
3. *Double Pole Single Throw* (DPST), *relay* jenis ini memiliki 6 terminal yaitu terdiri dari 2 terminal untuk input kumparan elektromagnetik dan 4 terminal saklar untuk 2 saklar yang masing-masing saklar hanya meiliki kondisi NO saja.
4. *Double Pole Double Throw* (DPDT), *relay* jenis ini memiliki 8 terminal yang terdiri dari 2 jenis terminal untuk kumparan elektromagnet dari 6 terminal untuk 2 sakkar dengan kondisi NC dan NO untuk masing-masig saklarnya.

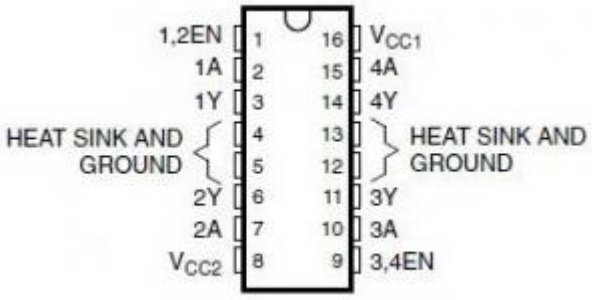
Disini penulis menggunkan *relay Double Pole Double Throw* dengan karakter *Normally Close*, berikut rangkaian Driver *Relay* pada gambar 2.19



Gambar 2.19 Rangkaian *Driver Relay*

2.4.2 L293D

IC L293D adalah IC yang didesain khusus sebagai *driver* motor DC dan dapat dikendalikan dengan rangkaian TTL maupun mikrokontroler. Motor DC yang dikontrol dengan *driver* IC L293D dapat dihubungkan ke *ground* maupun ke sumber tegangan positif karena di dalam *driver* L293D sistem driver yang digunakan adalah *totem pool*. Dalam 1 unit chip IC L293D terdiri dari 4 buah *driver* motor DC yang berdiri sendiri dengan kemampuan mengalirkan arus 1 *Ampere* tiap drivernya. Sehingga dapat digunakan untuk membuat driver *H-bridge* untuk 2 buah motor DC. Konstruksi pin *driver* motor DC IC L293D dapat dilihat pada gambar 2.20 [6]



Gambar 2.20 Kontruksi IC L293D

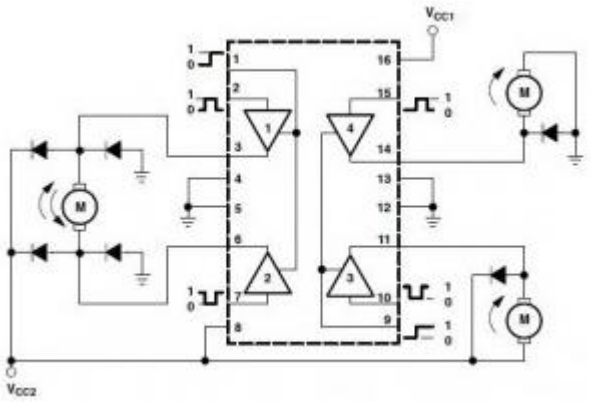
Fungsi Pin *Driver* motor DC IC L293D, yaitu :

1. Pin EN (*Enable*, EN1.2, EN3.4) berfungsi untuk mengijinkan *driver* menerima perintah untuk menggerakan motor DC.
2. Pin *In* (*Input*, 1A, 2A, 3A, 4A) adalah pin *input* sinyal kendali motor DC.
3. Pin *Out* (*Output*, 1Y, 2Y, 3Y, 4Y) adalah jalur *output* masing-masing *driver* yang dihubungkan ke motor DC.
4. Pin VCC (VCC1, VCC2) adalah jalur *input* tegangan sumber *driver* motor DC, dimana VCC1 adalah jalur *input* sumber tegangan rangkaian kontrol *dirver* dan VCC2 adalah jalur *input* sumber tegangan untuk motor DC yang dikendalikan.
5. Pin GND (*Ground*) adalah jalur yang harus dihubungkan ke *ground*, pin GND ini ada 4 buah yang berdekatan dan dapat dihubungkan ke sebuah pendingin kecil.

Driver motor DC IC L293D memiliki fitur yang lengkap untuk sebuah d*river* motor DC sehingga dapat diaplikasikan dalam beberapa teknik *driver* motor DC dan dapat digunakan untuk mengendalikan beberapa jenis motor DC. Fitur yang dimiliki *driver* motor DC IC L293D sesuai dengan *datasheet* adalah sebagai berikut :

1. *Wide supply*- Voltage Range :4.5 V s/d 36 V.
2. *Separate Input- Logic Supply*.
3. *Internal ESD Protection*.
4. *Thermal Shutdown*.
5. *High-Noise-Immunity Inputs*.
6. *Functionally Similar* to SGS L293 and SGS L293D.
7. *Ouput Current* 1 A per Channel (600 mA for L293D).
8. *Peak Output Current* 2 A per Channel (1.2 A for L293D).
9. *Output Clamp Diodes for Inductive Transient Suppression* (l293D).

Berikut rangkaian aplikasi *driver* motor DC IC L293D pada gambar 2.21



Gambar 2.21 Rangkaian *Driver* Motor DC IC L293D

2.5 Modul HX711

HX711 merupakan *module* timbangan yang memiliki prinsip kerja mengkonversi perubahan yang terukur dalam perubahan resistensi dan mengkonversinya ke dalam besaran tegangan melalui rangkaian yang ada. HX711 ini terintegrasi dari "*Avia Semiconductor*", HX711 presisi 24-bit *analog to digital* (ADC) yang didesain untuk sensor timbangan digital dalam industrial kontrol aplikasi yang terkoneksi sensor *load cell*. Berikut spesifikasi teknis modul HX711 dan Gambar kontruksi HX711 [8] :

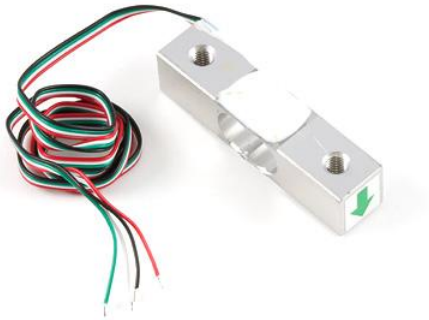
1. Dua kanal ADC (dapat digunakan untuk 2 *load cell*) dengan keluaran TTL (serial tersinkronisasi, *Dout* dan S*CK*).
2. Tegangan operasi 5 VDC
3. Tegangan masukan differensial ± 40 mV pada skala penuh.
4. Akurasi data 24 bit (24-bit ADC).
5. Frekuensi pembacaan (*refresh rate*) 80 Hz.
6. Konsumsi arus kurang dari 10 mV



Gambar 2.22 Modul HX711

2.6 Sensor *Load Cell*

Load Cell adalah komponen utama pada sistem timbangan d*igital*. Tingkat keakurasian timbangan tergantung dari jenis *load cell* yang dipakai. Sensor *load cell* apabila diberi beban pada inti besi maka nilai resistansi di *strain gauge* akan berubah yang dikeluarkan melalui empat buah kabel. Dua kabel sebagai eksitansi dan dua kabel lainnya sebagai sinyal keluaran ke kontrolnya. Gambar 2.23 adalah bentuk fisik dari sensor *load cell*.[8]



Gambar 2.23 Sensor *Load Cell*

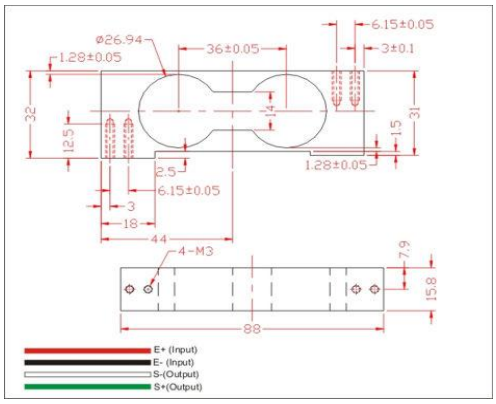
Sebuah *Load Cell* terdiri dari konduktor, *strain gaguge*, dan *wheatsone bridge*. Berikut ini beberapa penjelasan mengenai definisi *load cell*. Tegangan keluaran dari sensor *Load Cell* sangat kecil, sehingga untuk mengetahui perubahan tegangan keluaran secara *linier* dibutuhkan rangkaian penguat instrumen. Dalam hal ini digunakan Module HX711 yang memang dibuat khusus untuk menguatkan tegangan keluaran yang sangat kecil hingga kurang dari satuan *milivolt*, salah satunya sensor *Load Cell*, hingga ukuran tegangan dalam satuan *milivolt*. Konversi terjadi secara tidak langsung dalam dua tahap.[9]

Lewat pengaturan mekanis, gaya tekan dideteksi berdasarkan deformasi dari matriks pengukur regangan (*strain gauges*) dalam bentuk *resistor planar*. Regangan ini mengubah hambatan efektif (*effective resistance*) empat pengukur regangan yang disusun dalam konfigurasi jembatan *Wheatstone* (*Wheatstone bridge*) yang kemudian dibaca berupa perbedaan potensial (tegangan). Karena perbedaan yang terukur sangat kecil dalam orde µV (mikro Volt, seperjuta Volt), dibutuhkan rangkaian pengubah sinyal *analog* menjadi *digital* yang sangat presisi,

untuk itulah pada timbangan ini harus menggunakan modul HX711 yang beresolusi 24 bit.[9]

Dengan tingkat presisi setinggi ini, penulis dapat mengukur berat beban dalam resolusi 5 Kg/2^{24} atau setara dengan ketepatan 298 µg (0,298 mg atau 0,000298 gr). Ketepatan ini tiga kali lipat lebih tinggi dibandingkan tingkat ketepatan yang ada pada timbangan emas atau permata (*jewelry weight scale*) komersial kelas premium yang presisinya hanya mencapai 0,001 gr (1 mg), sehingga tantangan pembuatan timbangan elektronis yang presisi bukan lagi terletak pada sisi elektronisnya namun lebih pada akurasi rancang bangun mekanis dari timbangan tersebut. Berikut spesifikasi teknis dan gambar kontruksi *load cell* :

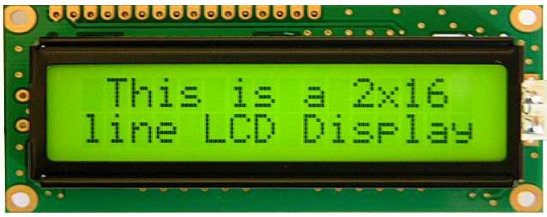
1. Beban maksimum : 5000 gram (5 Kg).
2. Rentang tegangan keluaran : 0,1 mV ~ 1,0 mV/ V (skala 1:1000 terhadap tegangan masukan, *error margin* ≤ 1,5%).
3. Implementasi masukan (*input impedance*) : 1066 Ω ±20%.
4. Impedansi keluaran (*output impedance*) : 1000 Ω ±10%.
5. Tegangan masukan maksimum : 10 Volt DC.
6. Rentang suhu operasional : -20 ~ +65˚C.
7. Material : *Alumunium Alloy*.



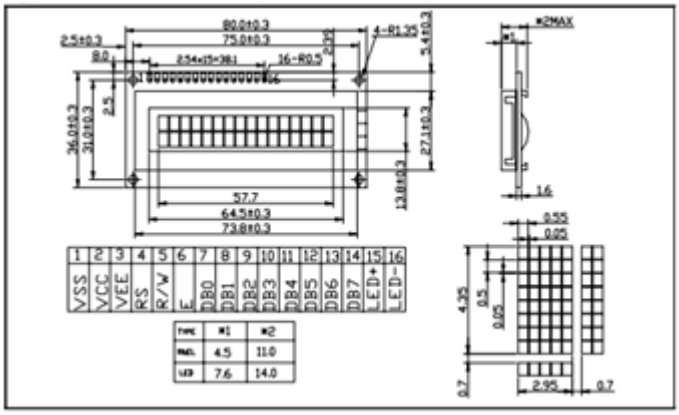
Gambar 2.24 Kontruksi Sensor *Load Cell*

2.7 *Liquid Crystal Display* (LCD)

LCD (*Liquid Crystal Display*) merupakan perangkat elektronik yang terbuat dari kristal cair yang digunakan untuk menampilkan karakter atau bilangan sebagai informasi dari kerja suatu *system* berbasis mikrokontroler. Setiap LCD sudah dilengkapi IC *driver* yang berfungsi sebagai kontrolnya. Salah satu tipe LCD yang banyak digunakan adalah LCD tipe JHD162A yaitu LCD yang mempunyai 2 baris tampilan dan setiap baris dapat menampilkan hingga 16 kolom karakter. Setiap baris dan kolom mempunyai alamat sendiri-sendiri.



Gambar 2.25 LCD 2x16



Gambar 2.26 Rangkaian *Interface* LCD

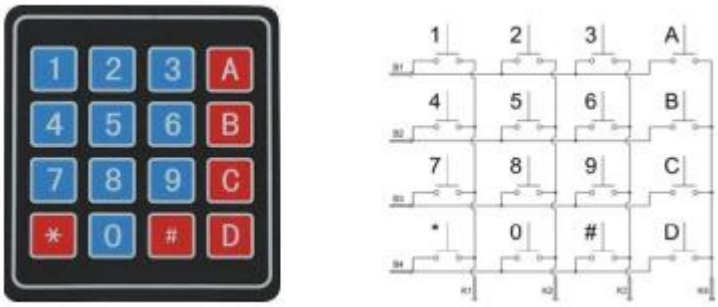
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VCC	VEE	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	LED+	LED-

Gambar 2.27 Pin Konfigurasi

2.8 Keypad 4x4

Keypad Matriks 4x4 adalah tombol-tombol yang disusun secara matriks (baris x kolom) sehingga dapat mengurangi penggunaan pin *input*. Sebagai contoh *Keypad* Matriks 4x4 cukup menggunakan 8 pin untuk 16 tombol. Hal tersebut dimungkinkan karena rangkaian tombol disusun secara *horizontal* membentuk baris dan secara vertikal membentuk kolom.[1]

Proses pengecekan dari tombol yang dirangkai secara matriks adalah teknik *scanning*, yaitu proses pengecekan yang dilakukan dengan cara memberi umpan data pada satu bagian dengan mengecek *feedback* pada bagian yang lain. Dalam hal ini, pemberian umpan data dilakukan pada bagian baris dan pengecekan *feedback* pada bagian kolom.



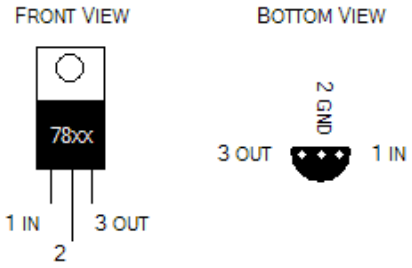
Gambar 2.28 *Keypad* 4x4

Namun demikian, sebagai konsekuensi dari penggunaan bersama satu jalur (missal: baris satu (B1)), maka tidak dimungkinkan pengecekan dua tombol sekaligus dalam satu slot waktu. Proses pengecekan dari tombol yang dirangkai secara matriks adalah dengan teknik *scanning*, yaitu proses pengecekan yang dilakukan dengan cara memberikan umpan data pada satu bagian dan mengecek *feedback* pada bagian yang lain. Dalam hal ini, pemberian umpan data dilakukan pada bagian baris dan pengecekan *feedback* pada bagian kolom. Pada saat pemberian umpan data pada satu baris, maka baris yang lain harus dalam kondisi invensi. Tombol yang ditekan dapat diketahui dengan melihat asal data dan di kolom mana data tersebut terdeteksi.

Cara kerja pada gambar 2.28 , tombol yang ditekan adalah tombol “5” maka B2 bernilai nol, sedangkan B1, B3, dan B4 adalah satu. Kemudian dengan mengetahui asal data dari B2, dan *feedback* terdeteksi pada K2, maka dapat disimpulkan bahwa tombol yang ditekan adalah tombol “5”.

2.9 Regulator

Regulator tegangan menyediakan *output* tegangan DC yang konstan dan secara terus menerus menahan tegangan *output* pada nilai yang diinginkan. Regulator hanya dapat bekerja jika tegangan *input* (*Vin*) lebih besar daripada tegangan *output* (*Vout*). Biasanya perbedaan tegangan *input* dengan *output* yang direkomendasikan tertera pada *datasheet* komponen tersebut



Gambar 2.29 IC Regulator

2.10 Perangkat Lunak

Arduino IDE merupakan *free software* yang yang dikembangkan khusus untuk mengakomodasi *board* Ardunio, seperti melakukan *compile* program, pengisian kode program, pengisian *bootloader*, dan sejenisnya. Program ini memiliki *library* internal yang berfungsi untuk mempermudah dalam pengaksesan fitur-fitur yang dimiliki oleh *board* Arduino. Oleh sebab itu, apabila menggunakan *board* Arduino, maka *software* yang digunakan untuk membuat program disarankan menggunakan Arduino IDE. Apabila menggunakan *software compiler* lain, seperti *Code Vision AVR*, *Bascom AVR*, ataupun *AVRGCC*, maka fitur dan kemudahan yang ditawarkan oleh Arduino tidak dapat dijumpai.

Arduino IDE telah menyediakan fitur untuk melakukan proses pengisian kode program (*upload sketch*) dan *burnbootloader*. Proses pengisian kode

program (*uploadsketch*) digunakan untuk mengisikan kode program yang telah dibuat agar masuk ke dalam mikrokontroler dengan bantuan *bootloader*. Pada proses ini hanya perlu memberikan catu daya pada DT-AVR Inoduino sebesar 9 s/d 12 Volt dan menghubungkan DT-AVR Inoduino ke PC melalui koneksi USB (*virtualCOM*).

Proses *burnbootloader* merupakan proses pengisian *bootloader* ke dalam DT-AVR Inoduino dengan bantuan eksternal programmer. adapun fungsi dari *bootloader* ini adalah untuk membantu proses pengisian kode program (*uploadsketch*) melalui *port* USB (*virtual* COM). Perihal eksternal program yang digunakan pada proses ini adalah eksternal programmer yang didukung oleh Arduino IDE. Adapun programmer yang kompatibel adalah programmer *AVRISP*, *AVRISPmkll*, *USBtinyISP* dan sejenisnya.

2.11 Pengukuran Akurasi Dan Eror

Pengukuran adalah proses perbandingan antara suatu besaran yang tidak diketahui dengan suatu besaran standar yang diperoleh, meliputi hubungan suatu alat ukur di dalam sistem dengan pertimbangan dan pengamatan dari hasil respon pada *instrument*. Pengukuran yang diperoleh adalah adalah pengukuran besaran yang disebut dengan "haga sebenarnya (*true value*)" akan tetapi sangat sulit untuk memberi definisi harga yang sebenarnya. Untuk *instrument* ini secara cermat, diperlukan pemahaman terhadap prinsip kerjanya dan kemampuan dalam memperkirakan apakah instrument tersebut sesuai untuk pemakaian yang direncanakan. Dalam pengukuran, digunakan sejumlah istilah yang akan didefinisikan sebagai berikut :

1. Ketelitian (*Accuracy*) adalah harga terdekat dengan mana suatu pembacaan *instrument* mendekati harga sebenarnya dari variabel yang diukur ; tingkat ketepatan pengukuran dibandingkan dengan harga (nilai) yang diharapkan.
2. Kesalahan (*Error*) adalah penyimpangan variabel yang diukur dari harga (nilai) yang sebenarnya.

Tingkatan dimana suatu pengukuran sesuai dengan harga (nilai) yang diharapkan ditunjukan dalam syarat-syarat kesalahan dari pengukuran. Kesalahan mungkin ditunjukan lain yaitu dengan kesalahan *absolute* atau presentase kesalahan. Kesalahan *absolute* dapat didefinisikan sebagai perbedaan antara variabel harga (nilai) yang diharapkan dengan variabel harga (nilai) pengukuran.

$$e = Y_n - X_n \quad \text{..... (2.1)}$$

dimana :

e : Kesalahan *absolute*

Y_n : Harga (nilai) yang diharapkan

X_n : Harga (nilai) pengukuran

Jika kita mengharapkan untuk menunjukan kesalahan sebagai presentase, maka dapat kita rumuskan sebagai hasil bagi antara kesalahan *absolute* dengan harga yang diharapkan dikali dengan 100%

$$\% \textit{ Kesalahan} = \frac{Yn - Xn}{Yn} \text{ x } 100\% \quad \text{..... (2.2)}$$

Persamaan di atas lebih sering digunakan untuk menyatakan pengukuran dengan syarat akurasi relative lebih besar dari pada kesalahan, atau

$$A = 1 - \left|\frac{Yn - Xn}{Yn}\right| \text{ x } 100\% \quad \text{..... (2.3)}$$

Dimana A adalah akurasi *relative* yang dapat dinyatakan sebagai presentase akurasi, yaitu :

$$a = 100\% - \% \textit{ Kesalahan} \quad \text{..... (2.4)}$$