

## BAB II TINJAUAN PUSTAKA

### 2.1. *Mobile Robot*

*Mobile robot* merupakan sebuah robot yang mampu bergerak dari suatu tempat ke tempat lain karena memiliki sebuah alat gerak untuk berpindah posisi. Salah satu contoh dari *mobile robot* adalah *self balancing robot*.

#### 2.1.1. *Self Balancing Robot*

*Self balancing robot* merupakan sebuah robot *mobile* yang hanya memiliki dua roda di sisi kanan dan sisi kirinya. Robot ini bekerja dengan mempertahankan keseimbangan, ketika robot jatuh kedepan maka roda akan berputar kebelakang begitupun sebaliknya sampai robot mendapatkan titik keseimbangan. Robot ini tidak akan dapat berdiri tegak tanpa adanya sensor dan kendali yang pas untuk bisa membuatnya bekerja, sekarang ini konsep robot beroda dua telah digunakan sebagai alat transportasi yang bernama *segway*



**Gambar 2.1** *Self Balancing Robot* (<https://wiki.keyestudio.com>)

Robot ini menerapkan prinsip kerja dari pendulum terbalik, dimana pendulum terbalik yang ada pada kereta beroda, yang bisa bergerak maju dan mundur dan pendulum yang pada ujungnya menyatu dengan kereta beroda sehingga ketika kereta beroda bergerak maka pendulum akan jatuh. Oleh sebab itu dibutuhkan suatu sistem kendali yang baik sehingga pendulum dapat tetap seimbang dan berdiri tegak

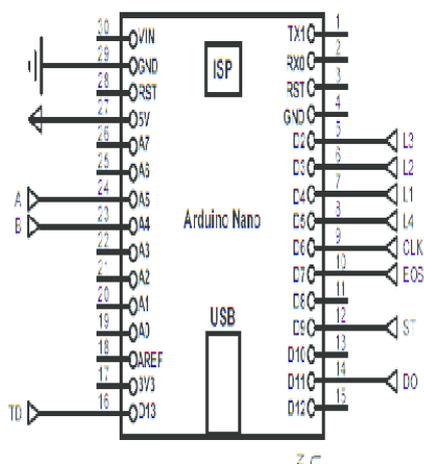
## 2.2. Arduino Nano

Arduino Nano adalah salah satu varian dari produk board mikrokontroler keluaran Arduino. Arduino Nano adalah board Arduino terkecil, menggunakan mikrokontroler Atmega 328 untuk Arduino Nano 3.x dan Atmega168 untuk Arduino Nano 2.x. Varian ini mempunyai rangkaian yang sama dengan jenis Arduino Duemilanove, tetapi dengan ukuran dan desain PCB yang berbeda. Arduino Nano tidak dilengkapi dengan soket catu daya, tetapi terdapat pin untuk catu daya luar atau dapat menggunakan catu daya dari mini USB *port*. Arduino Nano didesain dan diproduksi oleh Gravitech.



**Gambar 2.2** Arduino nano (<https://www.aksesoriskomputerlampung.com>)

Skema rangkaian Arduino Nano dapat dilihat pada gambar berikut ini.



**Gambar 2.3** Skematik arduino nano (<https://www.researchgate.net>)

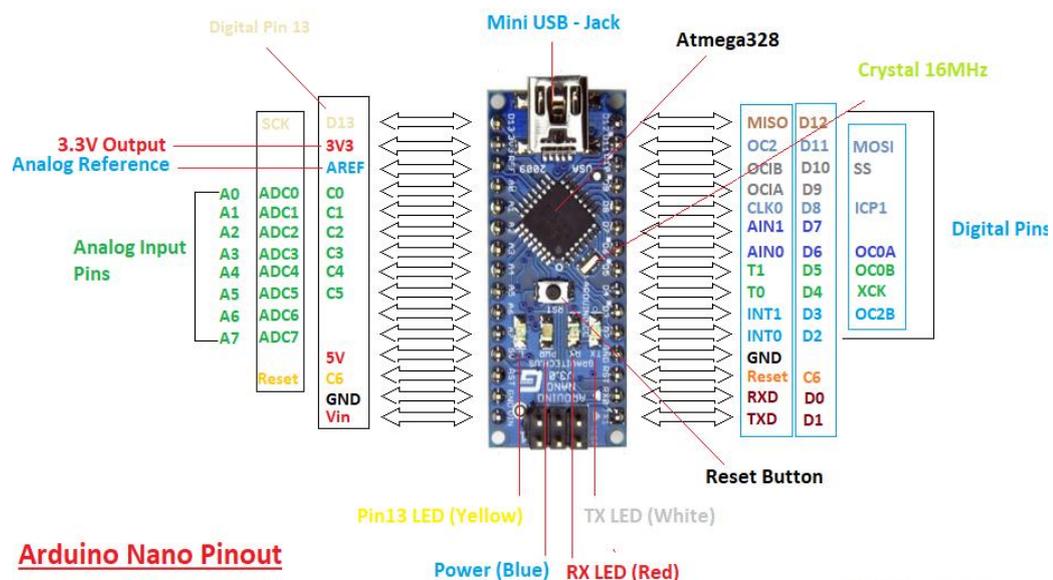
### 2.2.1. Spesifikasi Arduino Nano

Dibawah ini spesifikasi dari Arduino Nano:

- Mikrokontroler : Atmel ATmega168 atau ATmega328
- Tegangan Operasi : 5V
- *Input Voltage* (disarankan) : 7-12V
- *Input Voltage* (limit) : 6-20V
- Pin Digital I/O : 14 (6 pin digunakan sebagai output PWM)
- Pin Input Analog : 8
- Arus DC per pin I/O : 40 mA
- *Flash Memory* : 16KB (ATmega168) atau 32KB (ATmega328) 2KB
- SRAM : 1 KB (ATmega168) atau 2 KB (ATmega328)
- EEPROM : 512 byte (ATmega168) atau 1KB (ATmega328)
- *Clock Speed* : 16 MHz
- Ukuran : 1.85cm x 4.3cm

### 2.2.2. Pin Input/Output Arduino Nano

Berikut ini merupakan pin out dari Arduino nano seperti dibawah ini:



Gambar 2.4 Pin out arduino nano (<https://www.teachmemicro.com>)

## A. Pin Input/Output Digital

Fungsi utama dari pin ini adalah untuk membaca sinyal digital, yaitu berupa nilai 0 dan 1 atau ada juga yang menyebutnya logika *TRUE* dan *FALSE*. Adapun untuk jumlah pin digital pada Arduino Nano yaitu sebanyak 14 pin. Terhitung dari pin RX0, TX1, D2, dan sampai D13. Selain itu, ternyata pin input/output digital masih bisa dikelompokkan lagi berdasarkan fungsi spesifiknya, yaitu:

### 1. Pin Serial

Yaitu Arduino Nano pin yang fungsinya untuk memungkinkan terjadinya komunikasi serial pada arduino. Contohnya yaitu pin RX0 dan TX1. RX berfungsi untuk menerima TTL data serial dan TX berfungsi untuk mengirim TTL data serial.

### 2. Pin *External Interrupt*

Yaitu pin yang dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai rendah, meningkat, menurun, atau perubahan nilai. Pin yang termasuk *Eksternal Interrupt* yaitu pin D2 dan D3.

### 3. Pin PWM Arduino Nano

Yaitu pin yang memungkinkan kita untuk menggunakan fitur PWM (*Pulse Width Modulation*). Pin yang termasuk PWM pada Arduino Nano yaitu pin D3, D5, D6, D9, dan D11. Ini ditandai dengan adanya tanda titik atau strip.

### 4. Pin SPI (*Serial Peripheral Interface*)

Fungsi pin ini adalah memungkinkan terjadinya komunikasi SPI. Contoh yang termasuk pin SPI yaitu pin D10 (SS), D11 (MOSI), D12(MISO), dan pin D13 (SCK).

### 5. Pin LED

Alasan utama mengapa pin 13 disebut pin LED karena fungsi pin ini adalah untuk menyalakan LED yang terpasang secara *built-in* di Arduino.

## **B. Pin Input Analog**

Secara umum, fungsi pin ini adalah untuk membaca sinyal analog untuk diubah ke dalam bentuk sinyal digital. Jumlah pin input analog Arduino Nano berjumlah delapan. Terdiri atas 8 pin diantara delapan pin tersebut ada dua pin yang memiliki fungsi khusus, yaitu memungkinkan terjadinya komunikasi I2C. Pin tersebut antara lain:

### **1. Pin SDA (Serial Data)**

Pin ini berfungsi untuk mentransaksikan data guna mendukung komunikasi I2C atau TWI (*Two Wire Interface*). Pin yang termasuk pin SDA adalah pin analog 4 atau pin A4.

### **2. Pin SCL (Serial Clock)**

Pin ini berfungsi untuk menghantarkan sinyal *clock* guna memungkinkan terjadinya komunikasi I2C atau TWI. Pin yang merupakan pin SCL adalah pin analog 5 atau pin A4.

## **C. Pin Tegangan**

Fungsi dari pin tegangan adalah memungkinkan kita untuk mengatur tegangan yang ada pada Arduino. Beberapa contoh pin tegangan dan fungsinya yaitu:

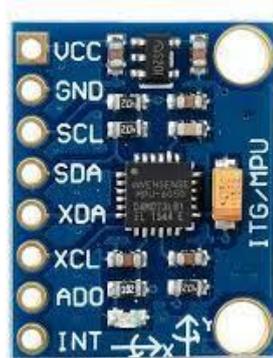
1. VIN, berfungsi sebagai tempat masuknya tegangan eksternal
2. 5V, berfungsi memberikan tegangan yang besarnya 5 volt
3. 3,3V, berfungsi memberikan tegangan yang besarnya 3,3 volt
4. GND (*ground*), berfungsi menghilangkan beda potensial jika sewaktu-waktu terjadi kebocoran tegangan
5. AREF, berfungsi mengatur tegangan referensi eksternal sebagai batas atas pin input analog
6. IOREF, berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler

#### D. Pin *RESET*

Berfungsi untuk merestart ulang program yang sedang berjalan pada Arduino. Caranya dengan menghubungkan pin *RESET* ke salah satu pin digital lalu memasukkan script khusus. Untuk lebih lengkapnya,

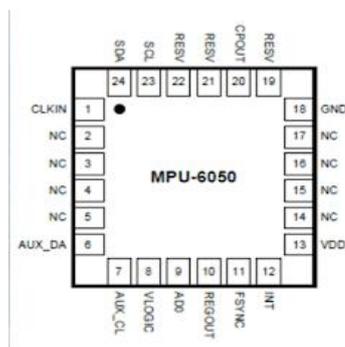
#### 2.3. Sensor MPU6050

Modul sensor MPU6050 merupakan sensor yang berfungsi sebagai *accelerometer* sekaligus *gyroscope* yang disatukan kedalam 1 module yang kompatibel dengan Arduino..



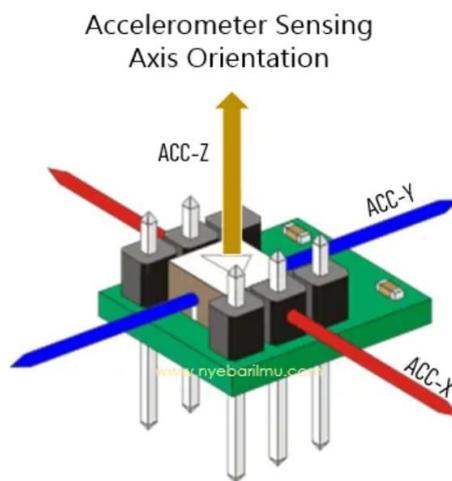
**Gambar 2.5** Sensor MPU6050 (<https://www.nyebarilmu.com>)

Skematik Sensor MPU6050 dapat dilihat pada gambar berikut ini.



**Gambar 2.6** Skematik sensor MPU6050 (<https://www.edukasiElektronika.com>)

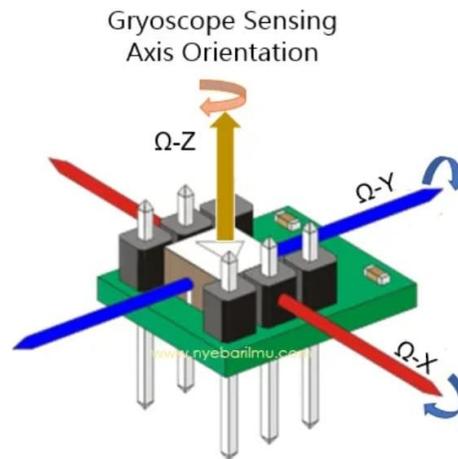
*Accelerometer* merupakan sensor berfungsi untuk mengukur percepatan suatu benda atau objek yang bergerak yang bersifat *dynamic* maupun *static*. Pengukuran *dynamic* merupakan pengukuran percepatan pada objek benda yang bergerak, sedangkan pengukuran *static* merupakan pengukuran percepatan objek terhadap gravitasi bumi. Untuk hal ini *accelerometer* lebih sering digunakan untuk mengukur sudut kemiringan (tilt) dan secara dasarnya prinsip kerja dari *accelerometer* yaitu menggunakan prinsip percepatan (*acceleration*).



**Gambar 2.7** Accelerometer Sensing(<https://www.nyebarilmu.com>)

*Gyroscope* merupakan sensor yang berfungsi untuk menentukan orientasi gerak dengan bertumpu pada roda yang berotasi dengan cepat pada sumbu yang berdasarkan momentum sudut. Sebelum digunakan, *gyro* sensor harus dilakukan kalibrasi terlebih dahulu dengan menggunakan bandul.

Proses kalibrasi tersebut berfungsi untuk memperoleh nilai faktor kalibrasi. Contoh penggunaan *gyroscope* salah satunya ialah pada *smartphone* yang paling umum yaitu pada aplikasi *Google Sky Map*. Anda juga bisa menemukannya saat sedang menikmati konten virtual *reality* berbasis 3D pada *smartphone*. Selain pada ponsel, *gyroscope* juga dipakai pada alat-alat canggih lainnya seperti *drone* dan robot.



**Gambar 2.8** Gyroscope Sensing (<https://www.nyebarilmu.com>)

.Gyroscope memiliki keluaran berupa kecepatan sudut dari arah 3 sumbu yaitu, sumbu x yang nantinya akan menjadi sudut phi (kanan dan kiri) dari sumbu y nantinya menjadi sudut theta (atas dan bawah), dan sumbu z nantinya menjadi sudut psi (depan dan belakang)

#### 2.4. Motor Stepper

Motor Stepper adalah jenis motor yang putarannya berdasarkan langkah (*step*) diskrit. Input pada motor stepper berasal dari pulsa-pulsa digital, berbeda dengan motor DC konvensional yang bekerja berdasarkan komutasi pada komponen *brush* (sikat)-nya.



**Gambar 2.9** Motor Stepper (<https://www.idekubagus.com>, 2017)

Step yang mengendalikan motor berasal dari konstruksi kumparan yang disusun menjadi beberapa kelompok yang disebut fase. Motor dapat berputar apabila diberikan energi pada fase secara berurutan. Motor Stepper mengubah sinyal-sinyal listrik menjadi gerakan mekanis diskrit. Motor Stepper bergerak dalam langkah (*step*) secara teratur. Anda dapat mengendalikan langkah pada motor menggunakan mikrokontroler maupun rangkaian digital.

Torsi dari motor Stepper tidak sebesar motor DC. Namun, motor jenis ini memiliki tingkat presisi yang tinggi dalam putarannya. Kecepatan gerak pada stepper dinyatakan dalam *step per second* atau jumlah step per detik.

#### **2.4.1. Prinsip Kerja Motor Stepper**

Pada dasarnya prinsip kerja stepper motor ini sama dengan DC Motor, yaitu pembangkitan medan magnet untuk memperoleh gaya tarik ataupun gaya lawan dengan menggunakan satu tegangan DC pada lilitan/kumparannya. Perbedaannya terletak pada gaya yang digunakan. Bila DC Motor menggunakan gaya lawan untuk melawan atau mendorong fisik kutub magnet yang dihasilkan maka stepper motor menggunakan gaya tarik untuk menarik fisik kutub magnet yang berlawanan sedekat mungkin ke posisi kutub magnet dihasilkan oleh kumparan. Oleh karena itu, pada DC Motor, putarannya relatif tidak terkendali, jarak tolakannya sangat relatif, tergantung pada besar medan magnet yang dihasilkan. Sebaliknya pada stepper motor, gerakan motor terkendali karena begitu kutub yang berlawanan tadi sudah tarik-menarik dalam posisi yang paling dekat, gerakan akan berhenti dan direm.

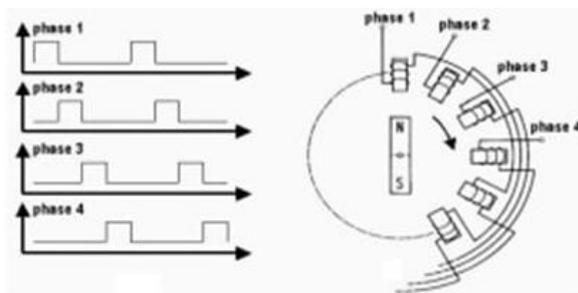
Bila kumparan mendapat tegangan dengan analogi mendapat logika 1, maka akan dibangkitkan kutub magnet yang berlawanan dengan kutub magnet tetap pada rotor. Dengan demikian, posisi kutub magnet rotor akan ditarik mendekati lilitan yang menghasilkan kutub magnet tetap pada rotor itu akan berpindah posisi menuju kutub magnet lilitan yang dihasilkan sekarang. Berarti, telah terjadi gerakan 1 step. Bila langkah ini diulang terus-menerus, dengan memberikan

tegangan secara bergantian ke lilitan-lilitan yang bersebelahan, rotor akan berputar.

Logika perputaran rotor tersebut dapat dianalogikan secara langsung dengan data 0 atau 1 yang diberikan secara serentak terhadap semua lilitan stator motor. Hal ini sangat memudahkan bagi sistem *designer* dalam menciptakan putaran-putaran stepper motor secara bebas dengan hanya mempermainkan bit-bit pada data yang dikirimkan ke rangkaian *interface* stepper motor tersebut. Untuk stepper motor 4 fase, pada prinsipnya ada dua macam cara kerja, yaitu *full step* dan *half step*. Penjabatan formasi logika dalam tabel ini adalah untuk mewakili putaran  $360^\circ$  relatif terhadap fase dari motor.

Motor Stepper yang dijumpai di pasaran sebagian besar melipat gandakan jumlah kutub magnet kumparannya dengan memperbanyak kumparan stator sejenis melingkar berurutan dalam konfigurasi penuh  $360^\circ$  rill terhadap poros rotor (dengan jumlah fase tetap). Hal ini dilakukan untuk memperoleh efek rill putaran 1 step yang lebih presisi, misalnya  $3,6^\circ/\text{step}$  atau  $1,8^\circ/\text{step}$ .

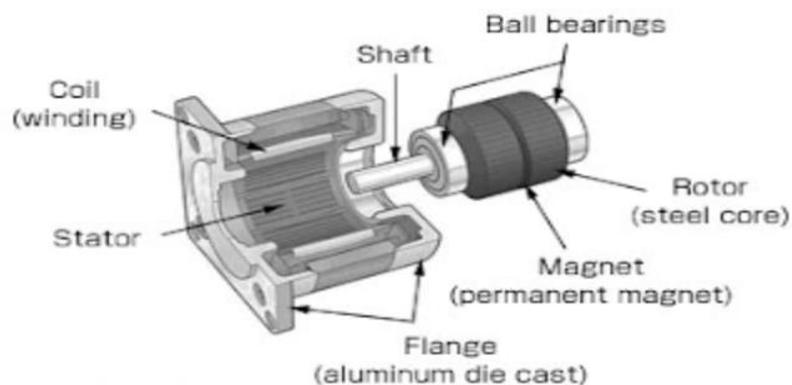
Untuk memperoleh efek cekraman yang lebih kuat, modus data yang diberikan pada mode *full wave* dapat dimanipulasi dengan memberikan *double* aktif bits pada setiap formasi. Dengan cara ini, torsi yang dihasilkan akan lebih besar. Namun demikian, penggunaan arus akan berlipat dua karena pada saat yang bersamaan dua lilitan mendapatkan arus kemudi. Prinsip kerja motor stepper adalah mengubah pulsa-pulsa input menjadi gerakan mekanis diskrit. Oleh karena itu untuk menggerakkan motor stepper diperlukan pengendali motor stepper yang membangkitkan pulsa-pulsa periodik.



**Gambar 2.10** Prinsip kerja motor stepper (<http://blog.unnes.ac.id>)

## 2.4.2. Konstruksi Motor Stepper

Pada dasarnya, motor stepper memiliki konstruksi yang sama dengan motor listrik pada umumnya, yaitu memiliki rotor dan stator. Perbedaannya adalah motor stepper tidak memiliki sikat karena putaran dilakukan dengan memberikan pulsa diskrit pada kumparan motor. Stator pada motor DC terbuat dari logam yang terdapat belitan di sekelilingnya. Lilitan ini tidak dililitkan pada seluruh bagian stator, namun dibelitkan secara individual pada gigi stator, sehingga lebih mirip motor *brushless*. Lilitan inilah yang terhubung pada input dan diberikan pulsa digital. Sedangkan rotor pada stepper terbuat dari besi lunak atau dari batang magnet. Rotor akan merespon medan magnet pada belitan stator dan bergerak sesuai sudut stepnya. Motor Stepper dapat diatur posisinya tanpa mekanisme umpan balik (*feedba.ck*)



**Gambar 2.11** Konstruksi Motor Stepper (<https://www.andalanelektro.id>, 2021)

Bagian-bagian dari motor stepper yaitu tersusun atas rotor, stator, *bearing*, *casing* dan sumbu.

1. Rotor pada motor stepper terdiri dari poros, roda dan sudu gerak.
2. Stator terdiri dari beberapa kutub. Setiap kutub memiliki lilitan yang menghasilkan medan magnet yang akan menggerakkan rotor. Pemberian arus yang berurutan pada kutub-kutubnya menyebabkan medan magnet berputar yang akan menarik rotor ikut berputar. Stator juga memiliki dua bagian plat

yaitu plat inti dan plat lilitan. Plat inti dari motor stepper ini biasanya menyatu dengan casing.

3. *Casing* motor stepper terbuat dari aluminium dan ini berfungsi sebagai dudukan bearing dan stator pemegangnya adalah baur sebanyak empat buah. Di dalam motor stepper memiliki dua buah bearing yaitu bearing bagian atas dan bearing bagian bawah.
4. Sumbu merupakan pegangan dari rotor dimana sumbu merupakan bagian tengah dari rotor, sehingga ketika rotor berputar sumbu ikut berputar.

### 2.4.3. Motor Stepper Nema 17

Motor stepper Nema 17 adalah tipe bipolar, dengan sudut langkah 1,8°, artinya, dapat membagi setiap putaran atau berubah menjadi 200 langkah. Setiap belitan yang ada di dalamnya mendukung intensitas 1.2A pada tegangan 4v, yang dengannya ia mampu mengembangkan gaya yang cukup besar sebesar 3.2 kg / cm.



**Gambar 2.12** Motor Stepper Nema 17 (<http://www.jogjarobotika.com>)

Mesin dari Nema 17 kuat itulah mengapa ini digunakan dalam aplikasi seperti printer 3D rumahan dan robot lain yang harus memiliki konsistensi yang cukup. Contoh printer yang menggunakan mesin ini sebagai dasar pergerakannya adalah pulsa. Ini juga digunakan dalam pemotong laser, mesin CNC, mesin *pick and place*, dan sebagainya.

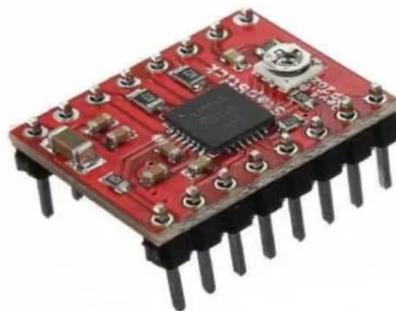
#### 2.4.4. Spesifikasi Nema 17

Motor stepper Nema 17 memiliki Spesifikasi yang ditampilkan seperti dibawah ini:

- Motor stepper.
- Model NEMA 17
- Berat 350 gram
- Ukuran 42.3x48mm tanpa poros
- Diameter poros 5mm D.
- Panjang poros 25mm
- 200 langkah per putaran (1,8° / langkah)
- 1.2A saat ini per belitan
- Tegangan suplai 4v
- Resistensi 3.3 Ohm per koil
- Torsi motor 3.2 kg / cm
- Induktansi 2.8 mH per koil

#### 2.5. Modul Driver A4988

A4988 adalah *driver microstepping* untuk mengendalikan motor stepper bipolar yang mempunyai *translator* bawaan untuk pengoperasian yang lebih mudah. *Driver* motor ini dapat mengontrol kerja motor stepper hanya dengan 2 pin dari kontroler, pin pertama untuk mengontrol arah putaran, pin kedua untuk mengontrol step motor.



**Gambar 2.13** Modul Driver A498 (<https://microcontrollerslab.com>)

Driver ini menyediakan lima step resolutions yang berbeda: *Full-Step*, *Half-Step*, *Quarter-Step*, *Eighth-Step* dan *Sixteenth-Step* dengan konfigurasi Pin sebagai berikut :

**Tabel 2.1** Tabel konfigurasi Pin

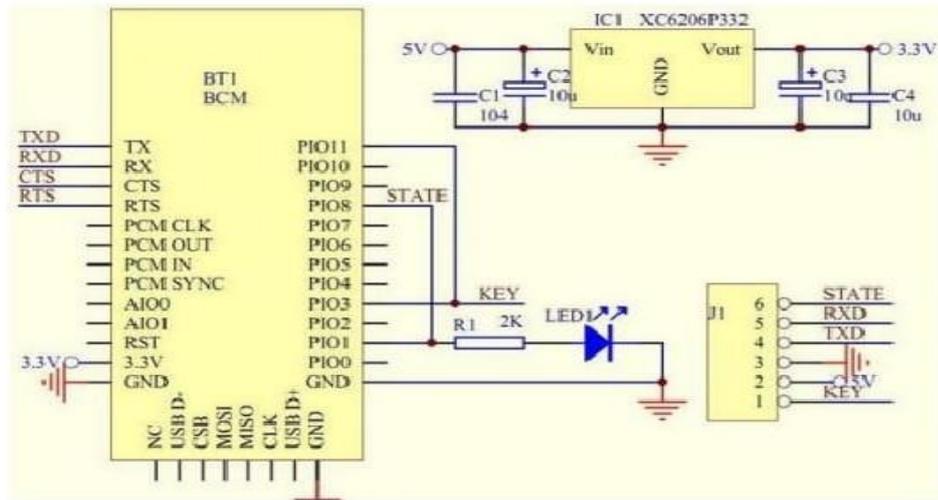
No	MS1	MS2	MS3	Resolusi
1	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Full-Step</i>
2	<i>High</i>	<i>Low</i>	<i>Low</i>	<i>Half-Step</i>
3	<i>Low</i>	<i>High</i>	<i>Low</i>	<i>Quarter-Step</i>
4	<i>High</i>	<i>High</i>	<i>Low</i>	<i>Eighth-Step</i>
5	<i>High</i>	<i>High</i>	<i>High</i>	<i>Sixteenth-Step</i>

## 2.6. HC-05 Bluetooth Modul

Modul Bluetooth HC-05 Adalah sebuah modul Bluetooth SPP (*Serial Port Protocol*) yang mudah digunakan untuk komunikasi serial *wireless* (nirkabel) yang mengkonversi port serial ke Bluetooth. HC-05 menggunakan modulasi bluetooth V2.0 + EDR (*Enhanced Data Rate*) 3 Mbps dengan memanfaatkan gelombang radio berfrekuensi 2,4 GHz.



**Gambar 2.14** HC-05 Bluetooth Modul (<https://www.jsumo.com>)



**Gambar 2.15** Skematik modul bluetooth

(<http://www.caturcreativeproject.com>, 2017)

Modul ini dapat digunakan sebagai *slave* maupun *master*. Mode *master* adalah mode dimana Bluetooth dapat berfungsi sebagai pengirim dan penerima data, sedang mode *slave* Bluetooth hanya dapat berfungsi sebagai penerima saja. HC-05 memiliki 2 mode konfigurasi, yaitu AT mode dan *Communication* mode. AT mode berfungsi untuk melakukan pengaturan konfigurasi dari HC-05. Sedangkan *Communication* mode berfungsi untuk melakukan komunikasi bluetooth dengan piranti lain.

**Tabel 2.2** Tabel Konfigurasi pin

NO	Nomor Pin	Nama	Fungsi
1	Pin 1	Key	-
2	Pin 2	vcc	Sumber Tegangan 5V
3	Pin 3	GND	Sumber <i>Ground</i>
4	Pin 4	TxD	Mengirim Data
5	Pin 5	RxD	Menerima Data
6	Pin 6	State	-

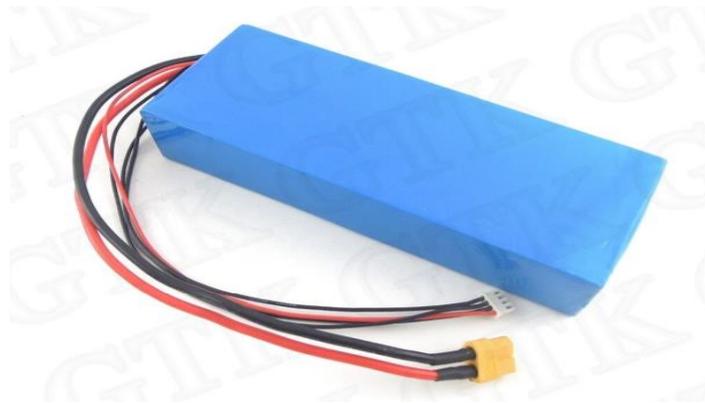
Pada perancangan ini dibutuhkan koneksi antara *android smartphone* dengan mikrokontroler secara nirkabel, maka digunakan media bluetooth sebagai *slave* mikrokontroler untuk sarana komunikasi,. Untuk berkomunikasi antar bluetooth, minimal harus memenuhi dua kondisi, yaitu Komunikasi harus antara *master* atau *slave* dan *Password* harus benar (saat melakukan *pairing*). Jarak sinyal dari HC-05 adalah 30 meter, dengan kondisi tanpa halangan.

#### **2.4.4. Spesifikasi Modul Bluetooth H-05**

- Frekuensi kerja ISM 2.4 GHz
- Bluetooth *protocol* : Bluetooth tipe v2.0+EDR
- Kecepatan dapat mencapai 1Mbps pada mode sinkron
- Kecepatan dapat mencapai 2.1 Mbps / 160 kbps pada mode asinkron maksimum
- Tegangan kerja pada 3,3 – 6 Volt DC
- Konsumsi arus kerja yaitu 50 mA
- Memiliki modulasi *Gaussian Frequency Shift Keying* (GFSK)
- Sensitivitas -84dBm (0.1% BER)
- Daya emisi 4 dBm
- Suhu operasional *range* -20°C — +75°C
- Memiliki keamanan dengan enkripsi data dan enkripsi
- Dimensi modul 15.2×35.7×5.6 mm

#### **2.7. Baterai**

Baterai (*Battery*) adalah sebuah sumber energi yang dapat merubah energi kimia yang disimpannya menjadi energi listrik yang dapat digunakan seperti perangkat elektronik. Dengan adanya baterai, kita tidak perlu menyambungkan kabel listrik ke terminal untuk dapat mengaktifkan perangkat elektronik kita sehingga dapat dengan mudah dibawa kemana-mana.



**Gambar 2.16** Baterai (<http://eprints.polsri.ac.id>)

Setiap baterai terdiri dari terminal positif (Katoda) dan terminal negatif (Anoda) serta elektrolit yang berfungsi sebagai penghantar. Output arus listrik dari baterai adalah arus searah atau disebut juga dengan arus DC (*Direct Current*) Baterai yang dibahas pada proposal ini yang dapat diisi ulang dan biasa digunakan pada kendaraan listrik yaitu baterai *Lithium ion* dan *Lithium Polymer*.

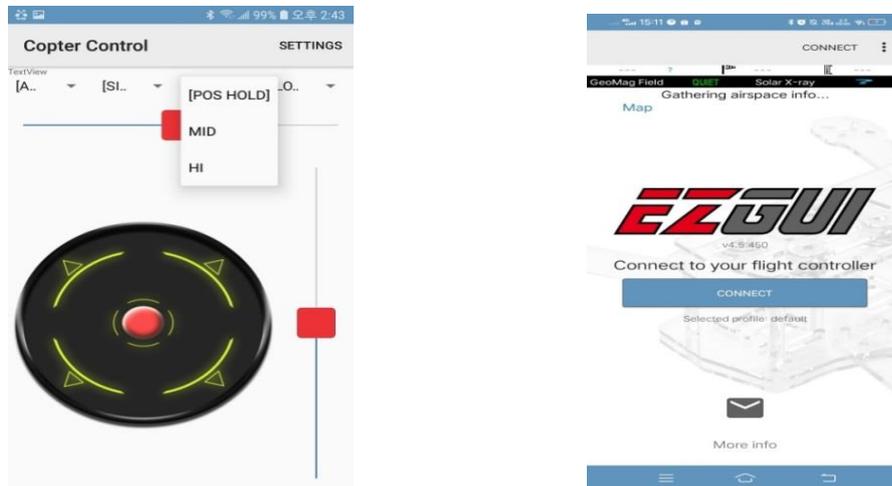
### **2.7.1 Baterai Li-Po**

*Lithium Polymer* adalah baterai yang bersifat cair. Baterai jenis ini menggunakan elektrolit polimer padat dan bisa menghantarkan data lebih cepat. Baterai Li-Po adalah pengembangan dari Li-Ion. Baterai Li-Po lebih ramah lingkungan dibanding Li-Ion. Dibanding Li-Ion, baterai Li-Po lebih ramping. Baterai jenis ini paling banyak dipakai di hp kelas *flagship*. Baterai Li-Po fisiknya lebih *fleksibel*. Karena berbentuk cairan atau gel, baterai jenis ini mudah dibentuk sehingga cocok untuk hp yang tipis. Selain itu, baterai ini juga lebih tahan lama dan siklus isi ulangnya lebih panjang.

### **2.8. EZ-GUI Apk**

EZ-GUI adalah *Ground Control Station (GCS)* suatu aplikasi berbasis Android untuk UAV berbasis MultiWii dan *Cleanflight*. Dapat menampilkan semua data yang tersedia dari pengontrol penerbangan dengan cara yang nyaman. sehingga memungkinkan Anda untuk dengan mudah mengonfigurasi dan

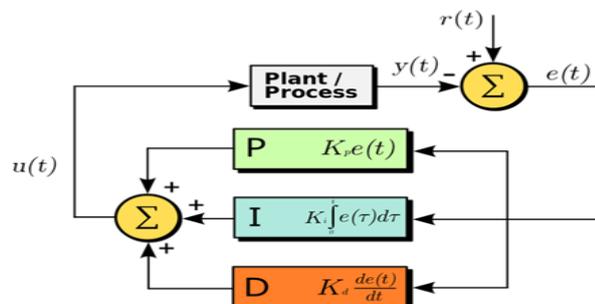
menyetel model Anda dari perangkat Android. Mendukung koneksi USB langsung (Android >3.1) serta Bluetooth, WiFi, dan Radio 3DR.



**Gambar 2.17** Tampilan aplikasi EZ-GUI

## 2.9. Kendali PID

Kendali PID adalah sistem kendal iyang susunan komponen fisik dihubungkan sedmikian rupa untuk mengatur suatu kondisi agar mencapai kondisi yang diharapkan. Sistem kendali ini terdiri idari tiga bagian, yaitu input, proses,dan ouput. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu plant. Blok diagram sederhana dari kontrol PID dapat dilihat pada **Gambar 2.5**.



**Gambar 2.18** Blok Diagram kontrol PID (<https://putraekapermana.com>)

### 2.9.1. Proportional (elemen P)

*Proportional* menangani kesalahan (*error*) dengan segera, yang mana kesalahan tersebut dikalikan dengan suatu nilai konstanta  $K_p$ . Jika nilai kesalahan sama dengan nol, maka keluaran dari kontroler proportional juga sama dengan nol. Akan tetapi, kontroler proportional tidak akan mencapai nilai *setpoint*-nya jika nilai keluarannya yang dibutuhkan bukan nol untuk menjaga *setpoint*-nya.

### 2.9.2. Integral (elemen I)

*Integral* mempelajari nilai yang telah lalu (*lampau*), kesalahan diintegrasikan dan dikalikan dengan konstanta  $K_i$ . Bentuk integrasi memungkinkan kontroler untuk menghilangkan kondisi kesalahan tetap (*steady-state error*) jika proses membutuhkan nilai masukan bukan nol untuk mendapatkan *setpoint* yang diinginkan. Suatu *integral* kontroler akan bereaksi terhadap kesalahan (*error*) oleh peningkatan nilai yang ditambahkan ke nilai keluarannya. Sehingga hal ini akan memaksa kontroler untuk mencapai nilai *setpoint*-nya lebih cepat dibandingkan dengan hanya kontroler proportional dan menekan atau menghilangkan kondisi kesalahan tetap.

### 2.9.3. Derivative (elemen D)

*Derivative* untuk mengantisipasi nilai yang akan datang, *derivative* yang pertama dari kesalahan yang terjadi (*error*) dikalikan dengan suatu konstanta  $K_d$ . Hal ini dapat dipakai untuk mengurangi besarnya *overshoot* yang dihasilkan oleh komponen *proportional* dan *integral*, tetapi kontroler akan sedikit lebih lambat untuk mencapai *setpoint*.

**Tabel 2.3** Tabel Karakteristik Pengendali PID

<b>Respon</b>	<b><i>Overshoot</i></b>	<b><i>Rise Time</i></b>	<b><i>Settling Time</i></b>	<b><i>Steady-state error</i></b>
$K_p$	<i>Increase</i>	<i>Decrease</i>	<i>Small Change</i>	<i>Decrease</i>
$K_i$	<i>Increase</i>	<i>Decrease</i>	<i>Increase</i>	<i>Eliminate</i>
$K_d$	<i>Decrease</i>	<i>Small Change</i>	<i>Decrease</i>	<i>Small Change</i>

## 2.10. Komunikasi Bluetooth

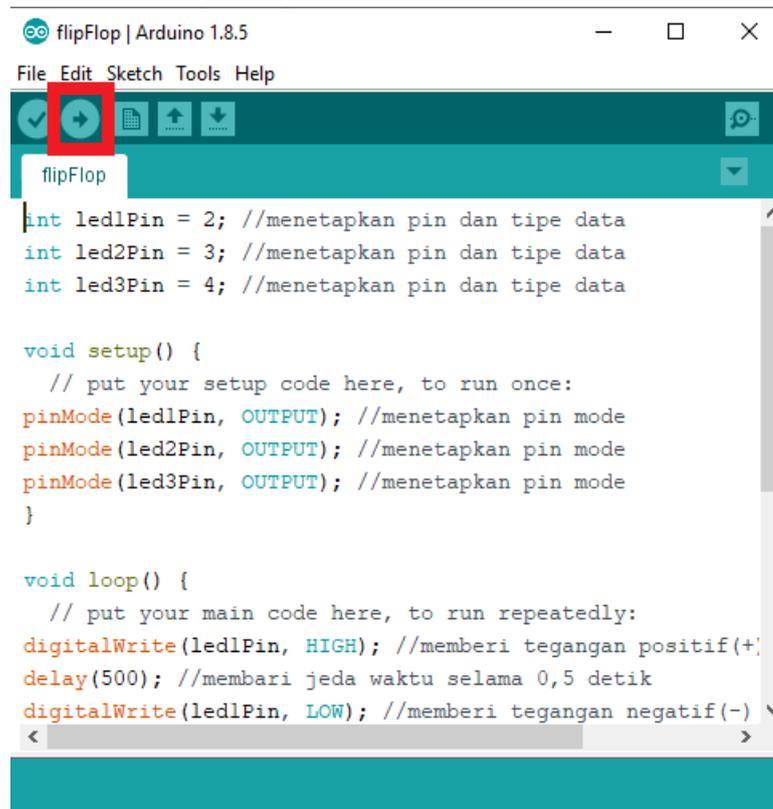
Protokol bluetooth menggunakan sebuah kombinasi antara *circuit switching* dan *packet switching*. Bluetooth dapat mendukung sebuah kanal data asinkron, tiga kanal suara sinkron simultan atau sebuah kanal dimana secara bersamaan mendukung layanan data asinkron dan suara sinkron. Setiap kanal suara mendukung sebuah kanal suara sinkron 64 kb/s. Kanal asinkron dapat mendukung kecepatan maksimal 723,2 kb/s asimetris, dimana untuk arah sebaliknya dapat mendukung sampai dengan kecepatan 57,6 kb/s. Sedangkan untuk mode simetris dapat mendukung sampai dengan kecepatan 433,9 kb/s. Sebuah perangkat yang memiliki teknologi *wireless* bluetooth akan mempunyai kemampuan untuk melakukan pertukaran informasi dengan jarak jangkauan sampai dengan 10 meter (~30 feet). Sistem bluetooth menyediakan layanan komunikasi point to point maupun komunikasi *point to multipoint*.

Produk bluetooth dapat berupa PC *card* atau USB *adapter* yang dimasukkan ke dalam perangkat. Perangkat-perangkat yang dapat diintegrasikan dengan teknologi bluetooth antara lain : *mobile PC*, *mobile phone*, *PDA (Personal Digital Assistant)*, *headset*, *kamera*, *printer*, *router* dan sebagainya. Aplikasi-aplikasi yang dapat disediakan oleh layanan bluetooth ini antara lain : *PC to PC file transfer*, *PC to PC file synch ( notebook to desktop)*, *PC to mobile phone*, *PC to PDA*, *wireless headset*, *LAN connection via ethernet access point* dan sebagainya.

## 2.12. Arduino IDE

Arduino Nano dapat diprogram dengan perangkat lunak Arduino . Pada ATmega328 di Arduino terdapat *bootloader* yang memungkinkan Anda untuk meng-upload kode baru untuk itu tanpa menggunakan *programmer hardware* eksternal. *Integrated Development Environment (IDE)* Arduino terdiri dari editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol- tombol untuk fungsi yang umum dan beberapa menu. *Integrated Development Environment (IDE)* Arduino terhubung ke *arduino board* untuk

meng-*upload* program dan juga untuk berkomunikasi dengan arduino *board*. Perangkat lunak (*software*) yang ditulis menggunakan *Integrated Development Environment* (IDE) Arduino disebut *sketch*.



**Gambar 2.19** Tampilan Software Arduino IDE

*Sketch* ditulis pada editor teks. *Sketch* disimpan dengan *file* berekstensi. Area pesan memberikan informasi dan pesan *error* ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan output teks *dari Integrated Development Environment* (IDE) Arduino dan juga jendela *Integrated Development Environment* (IDE) Arduino menunjukkan jenis *board* dan *port* serial yang sedang digunakan. Tombol toolbar digunakan untuk memeriksa dan meng-*uploadsketch*, membuat, membuka, atau menyimpan *sketch* dan menampilkan serial monitor. IDE Arduino terdiri dari:

1. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.

2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan Arduino.