BAB II

LANDASAN TEORI

Pada bab ini berisi tentang teori mengenai permasalahan yang dibahas dalam tugas akhir ini, dimulai dari definisi Sensor MPU6050, *remote control* (RX dan TX), Arduino Leonardo, *Pulse With Modulation* (PWM), PID, motor servo, *Liquid Crystal Display* (LCD), dan motor DC.

2.1 Sensor MPU 6050

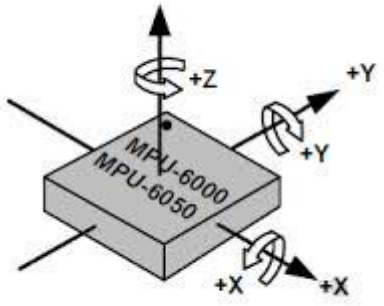
Sensor MPU 6050 merupakan *combo* sensor antara *accelerometer* dan *gyroscope* yang terintegrasi dalam satu chip. Sensor ini merupakan 6 *axis motion processing unit* dengan penambahan regulator tegangan dan 3,3V sehingga bisa langsung dihubungkan ke tegangan 5V. Sensor MPU 6050 memiliki dua buah *output* yaitu SCL dan SDA.



Gambar 2.1 Sensor MPU 6050 – 3 *Axis Accelerometer* + 3 *Axis Gyroscope*

Fitur yang ditawarkan sensor ini antara lain :

- Sensifitas *accelerometer* yang dapat dipilih mulai 2/4/8 sampai 16 g.
- Sensitifitas *gyroscope* yang dapat dipilih mulai 250/500/1000 sampai 2000 derajat/s .
- Range 16 bit untuk kedua sensor.
- Sensitivitas percepatan linier dari *gyroscope* 0,1 derajat/s
- Data *rate output* hingga 1000Hz, dilengkapi digital *low pass filter* dan memiliki frekuensi sudut maksimum 256 Hz.



Gambar 2.2 Cara Kerja Sensor terhadap *Axis* x, y, z

2.2 *Remote Control* (TX – RX)

Remote Control gelombang radio terdiri dari 2 bagian, yaitu pesawat pemancar (TX) dan pesawat penerima (RX). Pada *remote control* ini digunakan supaya mobil bergerak maju-mundur, belok ke kiri, belok ke kanan, dan sebagai *on/off* sistem.

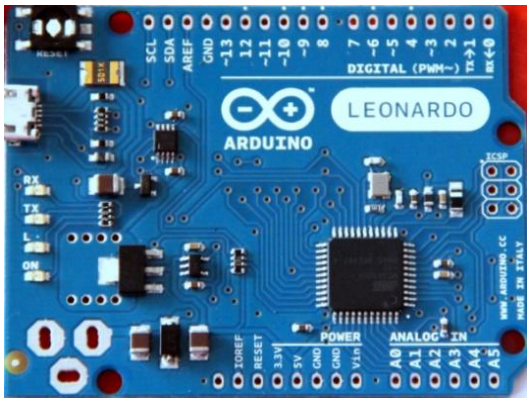


Gambar 2.3 *Remote Control Track Star 46*

Remote control ini memiliki 3-*channel* dengan jangkauan 2,4GHz digital radio dengan telemetri. *Remote* ini aktif pada tegangan input 4,4 V – 8,6 V.

2.3 Arduino Leonardo

Arduino Leonardo adalah papan mikrokontroler berbasis ATmega32u4. Arduino Leonardo memiliki 20 digital pin *input / output* (yang mana 7 pin dapat digunakan sebagai *output* PWM dan 12 pin sebagai *input* analog), 16 MHz kristal osilator, koneksi mikro USB, jack *power suplay* tegangan, *header* ICSP, dan tombol *reset*.



Gambar 2.4 Arduino Leonardo

Leonardo berbeda dari semua papan Arduino yang lainnya karena ATmega32u4 secara terintegrasi (*built-in*) telah memiliki komunikasi USB.

Tabel 2.1 Spesifikasi Arduino Leonardo :

Mikrokontroler	ATmega32u4
Tegangan Operasi	5V
Input Voltage (disarankan)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pin	20 pin
Channel PWM	7 pin
Input Analog	12 pin
Arus DC per pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Flash Memory	32 KB (ATmega32u4) 4 KB digunakan bootloader
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Clock Speed	16 MHz

Tabel 2.2 dibawah menunjukkan perbandingan Arduino Leonardo, Arduino Uno, dan Arduino Mega.

Tabel 2.2 Uraian Perbandingan Jenis Arduino

Spesifikasi	**Jenis Arduino**		
	Arduino Uno	**Arduino Leonardo**	**Arduino Mega**
Flash memory	32 kb	32 kb	256 kb
SRAM	2 kb	2,5 kb	8 kb
I/O	14	32	54
Harga	Rp. 185.000,00	Rp. 200.000,00	Rp. 585.000,00

2.3.1 Sumber Daya Arduino Leonardo

Arduino leonardo dapat diaktifkan melalui koneksi USB mikro atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya *eksternal* (non-USB) dapat berasal baik dari adaptor AC-DC atau baterai. Papan arduino leonardo dapat beroperasi dengan pasokan daya *eksternal* 6 Volt sampai

20 volt. Jika diberi tegangan kurang dari 7 Volt, maka, pin 5 Volt mungkin akan menghasilkan tegangan kurang dari 5 Volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 Volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 Volt sampai 12 Volt.

Pin tegangan yang tersedia pada papan arduino leonardo adalah sebagai berikut:

- **Vin**: Adalah *input* tegangan untuk papan arduino ketika menggunakan sumber daya *eksternal* (sebagai 'saingan' tegangan 5 Volt dari koneksi USB atau sumber daya ter-regulator lainnya). Anda dapat memberikan tegangan melalui pin ini, atau jika memasok tegangan untuk papan melalui jack *power*, kita bisa mengakses/mengambil tegangan melalui pin ini.
- **5V**: Tegangan listrik ter-regulator yang digunakan untuk daya mikrokontroler dan komponen lainnya pada papan arduino. Tegangan dapat menggunakan pin Vin melalui regulator *on-board*, atau dipasok oleh USB atau *power suplay* lain dengan besar tegangan 5V ter-regulator.
- **3V3**: Sebuah pin yang menghasilkan tegangan 3,3 Volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (*on-board*). Arus maksimum yang dihasilkan adalah 50 mA.
- **GND**: Pin *Ground* atau Massa.
- **IOREF**: Pin ini pada papan arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler (atau VCC untuk papan). Pin ini bertegangan 5V pada arduino leonardo.

2.3.2 Memori Arduino Leonardo

ATmega32u4 memiliki memori sebesar 32 KB (4 KB digunakan untuk *bootloader*). Juga memiliki 2,5 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM).

2.3.3 *Input* dan *Output* Arduino Leonardo

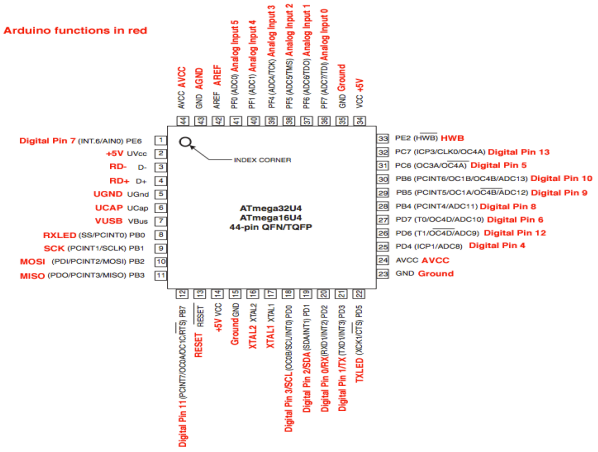
20 pin digital I/O pada Leonardo dapat digunakan sebagai *input* atau *output*, menggunakan fungsi pin *mode*, digital *write*, dan digital *read*. Mereka beroperasi pada tegangan 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki resistor *pull-up* internal sebesar 20-50 kOhm yang terputus secara *default*. Selain itu, beberapa pin memiliki fungsi khusus, yaitu:

- **Serial**: Pin 0 (RX) dan pin 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data serial TTL menggunakan *hardware* ATmega32U4 yang memiliki kemampuan serial didalamnya.
- **TWI**: Pin 2 (SDA) dan pin 3 (SCL). Dukungan komunikasi TWI menggunakan perpustakaan *wire*.
- **Eksternal Interupsi**: Pin 3 (*interrupt* 0), pin 2 (*interrupt* 1), pin 0 (*interrupt* 2), pin 1 (*interrupt* 3) dan pin 7 (*interrupt* 4). Pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau merubah nilai.
- **PWM**: Pin 3, 5, 6, 9, 10, 11, dan 13. Menyediakan 8-bit output PWM dengan fungsi analog *write*.

- **SPI**: Pin pada header ICSP ini mendukung komunikasi SPI menggunakan perpustakaan SPI. Perhatikan bahwa pin SPI tidak terhubung ke salah satu pun pin digital I/O karena yang terhubung langsung hanya pada arduino uno. Mereka hanya menyediakan konektor ICSP. Ini berarti bahwa jika Anda memiliki *shield* yang menggunakan SPI, tetapi tidak terdapat 6 pin konektor ICSP yang terhubung ke 6 pin ICSP *header* arduino leonardo, maka *shield* tidak akan bekerja.
- **LED**: Pin 13. Tersedia secara *built-in* pada papan Arduino ATmega2560. LED terhubung ke pin digital 13. Ketika pin diset bernilai *high*, maka LED menyala (*on*), dan ketika pin diset bernilai *low*, maka LED padam (*off*).
- ***Input Analog***: Pin A0-A5, Pin A6 - A11 (pada pin digital 4, 6, 8, 9, 10, dan 12). arduino leonardo memiliki 12 *input* analog, berlabel A0 sampai A11, yang semuanya juga dapat digunakan sebagai digital I/O. Pin A0-A5 terdapat di lokasi yang sama seperti pada arduino uno. Pin *input* A6-A11 masing-masing ada pada digital I/O pin 4, 6, 8, 9, 10, dan 12. Masing-masing pin menyediakan resolusi 10 bit (yaitu 1024 nilai yang berbeda). Secara *default* pin ini dapat diukur/diatur dari mulai *ground* sampai dengan 5 Volt, juga memungkinkan untuk mengubah titik jangkauan tertinggi atau terendah mereka menggunakan pin AREF dan fungsi *analog reference*.

Masih ada beberapa pin lainnya pada arduino leonardo, yaitu:

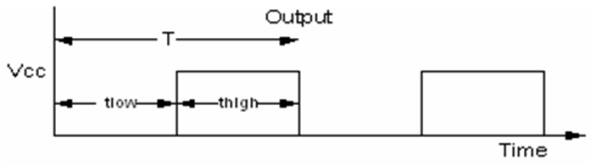
- **AREF**: Referensi tegangan untuk *input* analog. Digunakan dengan fungsi *analog reference.*
- **RESET**: Jalur *Low* ini digunakan untuk me-*reset* (menghidupkan ulang) mikrokontroler. Jalur ini biasanya digunakan untuk menambahkan tombol *reset* pada *shield* yang menghalangi papan utama arduino.



Gambar 2.5 Pemetaan Pin Arduino Leonardo

2.4 *Pulse Width Modulation (PWM)*

PWM merupakan sebuah metoda untuk membangkitkan sinyal keluaran berupa pulsa yang periodenya berulang antara *high* atau *low* dimana kita dapat mengontrol durasi sinyal *high* atau *low* sesua dengan yang kita inginkan. PWM bisa dibangkitkan secara *software* maupun *hardware*, sehingga dapat dibentuk gelombang dengan *duty cycle* yang dapat diatur sesuai dengan program.



Gambar 2.6 Sinyal PWM

Duty cycle merupakan perbandingan periode lamanya suatu sistem bernilai logika *high* dan *low*. Variasi *duty cycle* ini memberikan harga tegangan rata–rata yang berbeda–beda. Sinyal PWM dengan *duty cycle* yang besar memiliki nilai rata-rata tegangan yang lebih besar dibandingkan dengan *duty cycle* kecil. Nilai tegangan yang diberikan sebanding dengan nilai *duty cycle* yang diberikan. Adapun rumus *duty cycle*, yaitu:

$$Duty\ Cycle = \frac{t_{high}}{(t_{high}+t_{low})} \times 100\% \text{.....(2.1)}$$

Sedangkan untuk menghitung nilai tegangan rata-rata *output* dapat dihitung dengan rumus sebagai berikut:

$$V_{out} = Duty\ Cycle \times V_{total} \text{ .....(2.2)}$$

Dimana Vtotal merupakan tegangan yang diberikan untuk mengaktifkan motor.

2.5 Kendali *Proportional - Integral - Derivative (PID)*

PID merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu ***proportional, integratif*** dan ***derivatif***. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu *plant*, sehingga kelemahan-kelemahan pada salah satu komponen dapat ditutupi oleh komponen yang lain. Komponen I dan D tidak dapat berdiri sendiri dan selalu dikombinasikan dengan komponen P, menjadi pengontrol PI atau PID. Pengontrol PID akan mengeluarkan aksi kontrol dengan membandingkan kesalahan atau *error* yang merupakan selisih dari *process variable* dan *setpoint*, yang akan digunakan sebagai masukan pengontrol untuk mengeluarkan sinyal kontrol (*u(t)*).

Secara umum bentuk persamaan pengontrol PID dapat dinyatakan oleh persamaan 2.3 berikut:

$$mv(t) = K_p\left(e(t) + \frac{1}{T_i}\int e(t)\partial t + T_d\frac{\partial e(t)}{\partial t}\right) \quad \text{.....(2.3)}$$

mv(t)= *manipulated variable* keluaran dari pengontrol PID

K_p = penguatan proporsional

T_i = waktu integral

T_d = waktu derivatif

e(t) = *error* = *setpoint* – keluaran

2.5.1 Kendali *Proportional*

Kontrol P jika G(s) = kp, dengan k adalah konstanta. Jika u = G(s) • e maka u = Kp • e dengan Kp adalah konstanta proporsional. Kp berlaku sebagai *Gain* (penguat) saja tanpa memberikan efek dinamik kepada kinerja kontroler. Penggunaan kontrol P memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini. Walaupun demikian dalam aplikasi-aplikasi dasar yang sederhana kontrol P ini cukup mampu untuk memperbaiki respon transien khususnya *rise time* dan *settling time*.

Pengaruh pada sistem :

1. Menambah atau mengurangi kestabilan.
2. Dapat memperbaiki respon transien khususnya : *rise time, settling time*
3. Mengurangi (bukan menghilangkan) *error steady state*

2.5.2 Kendali *Integrative*

Jika G(s) adalah kontrol I maka u dapat dinyatakan sebagai $u(t) = [\int t \, . dt]Ki$ dengan Ki adalah konstanta integral, dan dari persamaan di atas, G(s) dapat dinyatakan sebagai $u = Kd.\left[\emptyset \frac{e}{\emptyset t}\right]$. Jika e(T) mendekati konstan (bukan nol) maka u(t) akan menjadi sangat besar sehingga diharapkan dapat memperbaiki *error*. Jika e(T) mendekati nol maka efek kontrol I ini semakin kecil. Kontrol I dapat memperbaiki sekaligus menghilangkan respon *steady-state*, namun pemilihan Ki yang tidak tepat dapat menyebabkan respon transien yang tinggi sehingga dapat menyebabkan ketidakstabilan sistem. Pemilihan Ki yang sangat tinggi justru dapat menyebabkan output berosilasi karena menambah orde sistem.

Pengaruh pada sistem :

1. Menghilangkan *Error Steady State*
2. Respon lebih lambat (dibandingkan dengan P)
3. Dapat menambah ketidakstabilan (karena menambah orde pada sistem)

2.5.3 Kendali *Derivative*

Sinyal kontrol u yang dihasilkan oleh kontrol D dapat dinyatakan sebagai $G(s) = s.Kd$. Dari persamaan di atas, tampak bahwa sifat dari kontrol D ini dalam konteks "kecepatan" atau rate dari *error*. Dengan sifat ini, dapat digunakan untuk memperbaiki respon transien dengan memprediksi *error* yang akan terjadi. Kontrol *derivative* hanya berubah saat ada perubahan *error* sehingga saat *error* statis kontrol ini tidak akan bereaksi, hal ini pula yang menyebabkan kontroler *derivative* tidak dapat dipakai sendiri.

Pengaruh pada sistem :

1. Memberikan efek redaman pada sistem yang berosilasi sehingga bisa memperbesar pemberian nilai Kp.
2. Memperbaiki respon transien, karena memberikan aksi saat ada perubahan *error*.
3. D hanya berubah saat ada perubahan *error*, sehingga saat ada *error* statis D tidak beraksi. Sehingga D tidak boleh digunakan sendiri.

2.6 Motor Servo

Motor servo adalah sebuah motor DC dengan sistem umpan balik tertutup dimana posisi rotornya 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. Pada motor servo biasanya terdapat tiga buah kabel, yang pertama untuk tegangan masukkan, yang kedua untuk menerima sinyal PWM, dan yang ketiga untuk *ground*.



Gambar 2.7 Motor Servo

Motor servo ini terbagi menjadi 2 jenis, yaitu :

a. Motor servo standard

   Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW)

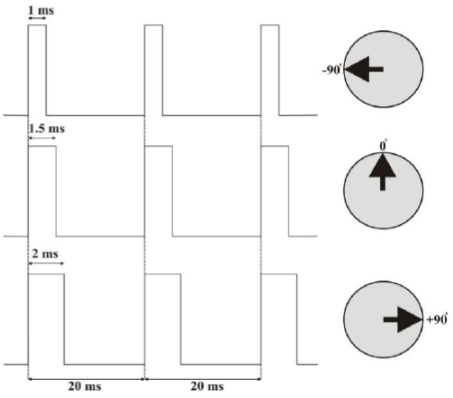
   dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°.

b. Motor servo *continous*

   Motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

Dari kedua jenis motor servo tersebut, metoda PWM dapat digunakan untuk menentukan posisi sudut motor, hanya saja terdapat perbedaan pada besarnya arah puataran motor.

Pada motor servo standard hanya dapat bergerak sebesar 180° yaitu dari 0°-180° atau (-90°)-90°. Secara umum untuk mengakses motor servo tipe standard adalah dengan cara memberikan pulsa *high* selama 1,5 ms dan mengulangnya setiap 20 ms, maka posisi servo akan berada ditengah atau netral (0°). Untuk pulsa 1 ms maka akan bergerak berkebalikan arah jarum jam dengan sudut -90°. Dan pulsa *high* selama 2 ms akan bergerak searah jarum jam sebesar 90° seperti terlihat pada gambar 2.8



Gambar 2.8 Sinyal Kontrol Motor Servo

Pengaturan pergerakan motor servo *continous* tidak jauh berbeda dengan pengaturan pengendalian motor servo standard, secara umum untuk berputar (rotasi) searah jarum jam harus diberi pulsa *high* selama 1,3 ms. Sedangkan untuk berputar berlawanan arah jarum jam harus diberi logika *high* selama 1,7 ms. Jika motor servo *continous* diberi pulsa high selama 1,5 ms maka akan berhenti. Pin signal pada motor servo dapat dikoneksi ke pin mikrokontroler sebagai pengendali dari pergerakan motor servo ini.

2.7 *Liquid Crystal Display* (LCD)

Liquid Crystal Display (LCD) adalah satu layar bagian dari modul peraga yang menampilkan karakter yang diinginkan. Layar LCD menggunakan 2 buah lembaran bahan yang dapat mempolarisasikan dan kristal cair diantara kedua lembaran tersebut. Arus listrik yang melewati cairan menyebabkan kristal merata sehingga cahaya tidak dapat melalui setiap kristal. Sehingga dapat mengubah bentuk kristal cairnya membentuk tampilan angka atau huruf pada layar.

Dalam tugas akhir ini LCD *character* digunakan untuk menampilkan informasi data sensor, data nilai PID, dan indikator nilai PWM,. Bentuk LCD *character* 16x2 ini dapat dilihat pada gambar 2.9.



Gambar 2.9 16x2 *Character* LCD *Module*

2.8 Motor DC

Motor DC adalah mesin yang mengubah energi listrik arus searah menjadi energi mekanis. Bentuk fisik motor DC dapat dilihat pada gambar 2.10.



Gambar 2.10 Motor DC

Cepat lambatnya motor DC dapat diatur berdasarkan *duty cycle* yang diberikan, jika *duty cycle* pada sinyal PWM besar, maka akan menghasilkan rata-rata tegangan yang besar pula sehingga motor DC akan berputar cepat, begitu juga sebaliknya jika *duty cycle* yang diberikan kecil akan mengahasilkan tegangan rata-rata yang kecil dan motor DC akan berputar lebih lambat.

2.9 Catu Daya (Baterai)

Baterai *Lithium Polimer* atau biasa disebut dngan LiPo merupakan salah satu jenis baterai yang sering digunakan dalam dunia RC. Baterai ini bersifat cair (*Liquid*), menggunakan elektrolit polimer yang padat, dan mampu menghantarkan

daya lebih cepat dan jenis baterai ini adalah hasil pengembangan dari *Lithium Ion*. Baterai Li-Po ini disebut sebagai baterai ramah lingkungan.



Gambar 2.11 Baterai Li-Po

Kelebihan Baterai Li-Po

1. Ramah lingkungan.
2. Memiliki kapasitas penyimpanan energi listrik yang besar.
3. Memiliki bobot yang ringan.

Kekurangan Baterai Li-Po

1. Baterai Li-Po menggunakan bahan elektrolit yang mudah terbakar.
2. Butuh perawatan khusus untuk isi ulang, seperti jangan sampai baterai habis baru di isi ulang.
3. Usia baterai lebih pendek.

2.10 Ubec

UBEC 3A adalah mode switch DC/DC regulator, mengeluarkan voltase yang konstan dan aman untuk *receiver*, *gyro* dan servo.



Gambar 2.12 Ubec

Spesifikasi :

- *Output voltage* : 5V@3A atau 6V@3A (dipilih dengan menukarkan posisi jumper)
- *Input* : 5.5V 26V (2-6S Lipo atau 5-18 *cells* NiMH / NiCd)

Fitur :

- Menggunakan IC yang canggih untuk *switch mode* DC-DC regulator
- Voltase sangat konstan bahkan dapat bekerja dengan baterai LiPo 4S-6S
- Dilindungi metal dan juga penyaring untuk memperkecil pengaruh gelombang elektromagnetik
- Status alat ditunjukkan dengan lampu LED