

PENALAAAN PID BERBASIS LQR UNTUK MODE TERBANG HOVER PADA QUADROTOR

Tesis

untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-2

Program Studi S2 Teknik Elektro
Konsentrasi Sistem Isyarat Elektronis
Departemen Teknik Elektro dan Teknologi Informasi



diajukan oleh
Ardy Seto Priambodo
14/370768/PTK/09697

Kepada
PROGRAM PASCASARJANA
FAKULTAS TEKNIK
UNIVERSITAS GADJAH MADA
YOGYAKARTA
2018

TESIS


**PENALAAAN PID BERBASIS LQR UNTUK MODE TERBANG HOVER PADA
QUADROTOR**

Dipersiapkan dan disusun oleh

Ardy Seto Priambodo
14/370768/PTK/09697

Telah dipertahankan di depan dewan penguji
Pada tanggal : **16 Januari 2018**
Susunan Dewan Penguji

Pembimbing Utama

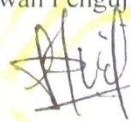


Dr. Eng. Adha Imam Cahyadi, S.T., M.Eng.
Pembimbing Pendamping



Dr. Ir. Samiadji Herdjunto, M. Sc.

Anggota Dewan Penguji Lain



Dr. Eng. Igi Ardiyanto, S.T., M.Eng.

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister

Tanggal: **22 JAN 2018**

Pengelola Program Studi: S2 Teknik Elektro



Dr. Ir. Risanuri Hidayat, M.Sc.
NIP. 196708021993031002



Mengetahui,
Ketua Departemen/Wakil Penanggung Jawab Program Studi
Teknik Elektro dan Teknologi Informasi



Sarjiya, S.T., M.T., Ph.D.
NIP. 197307061999031005

PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Tesis ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi tesis yang terkait hak milik, hak intelektual dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing dan Universitas Gadjah Mada. Dalam hal penggunaan informasi dan materi tesis terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.

Yogyakarta, 18 Januari 2018



Ardy Seto Priambodo

PRAKATA

Alhamdulillah, puji dan syukur dihaturkan kepada Allah ta'ala yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tesis dengan judul **“PENALAAAN PID BERBASIS LQR UNTUK MODE TERBANG HOVER PADA QUADROTOR”**. Sholawat dan salam semoga Allah ta'ala curahan kepada junjungan kita, penutup para nabi dan rasul, Nabi Muhammad ﷺ, keluarga, para sahabat, dan pengikutnya hingga hari kiamat kelak.

Dalam menyelesaikan laporan tesis ini tentu banyak hal yang telah membantu, selain doa, banyak pihak turut memberikan bantuan dan dukungan. Pada kesempatan ini, penulis mengucapkan terima kasih yang tak terhingga kepada :

1. Dr. Eng. Adha Imam Cahyadi, S.T., M.Eng., selaku dosen pembimbing utama, dan Dr. Ir. Samiadji Herdjunto, M.Sc., selaku dosen pembimbing pendamping yang telah dengan penuh kesabaran dan ketulusan memberikan ilmu dan bimbingan terbaik kepada penulis dan yang memberikan izin kepada penulis untuk belajar. Dan apabila saya memiliki banyak kesalahan, saya meminta maaf sebesar-besarnya.
2. Sarjiya, S.T., M.T., Ph.D, selaku Ketua Departemen Teknik Elektro dan Teknologi Informasi Universitas Gadjah Mada yang memberikan izin kepada penulis untuk belajar.
3. Dr. Ir. Risanuri Hidayat, M.Sc., selaku Ketua Program Studi Magister Teknik Elektro Universitas Gadjah Mada yang telah memberikan peluang kepada penulis untuk berkontribusi pada dunia pendidikan.
4. Orang tua penulis, Drs. Gonggong Haryono, M.M. dan Dra. Tatik Hidayati Purwaningtyas yang selalu memberikan dukungan dan mendoakan yang terbaik bagi putra putrinya.
5. Istri tercinta, Cindera Permataningsih, yang selalu menemani walaupun dalam keadaan susah, dan buah hati kami, Aidah Fatimah yang selalu punya banyak cerita baik suka dan duka, semoga engkau sehat selalu nak.

6. Para Dosen Program Studi S2 Teknik Elektro Fakultas Teknik Universitas Gadjah Mada yang telah memberikan bekal ilmu kepada penulis.
7. Para Karyawan/wati Program Studi S2 Teknik Elektro Fakultas Teknik Universitas Gadjah Mada yang telah membantu penulis dalam proses belajar.
8. Rekan-rekan S1 di Lab. IK, khususnya dalam *research group* ‘UAV’ atas kerjasama dan telah membagi pengetahuannya secara langsung maupun tidak langsung.
9. Semua rekan SIE angkatan 2014 atas kerjasamanya selama menjalani studi.
10. Semua pihak yang tidak sempat disebutkan disini yang telah membantu penulis secara langsung maupun tidak langsung.

Penulis menyadari sepenuhnya bahwa laporan tesis ini masih jauh dari sempurna, untuk itu semua jenis saran, kritik dan masukan yang bersifat membangun sangat penulis harapkan. Akhir kata, semoga tulisan ini dapat memberikan manfaat dan memberikan wawasan tambahan bagi para pembaca dan khususnya bagi penulis sendiri.

Yogyakarta, 18 Januari 2018

Ardy Seto Priambodo

ARTI LAMBANG DAN SINGKATAN

ϕ	Posisi angular sudut roll
$\dot{\phi}$	Kecepatan angular roll
θ	Posisi angular sudut pitch
$\dot{\theta}$	Kecepatan angular pitch
ψ	Posisi angular sudut yaw
$\dot{\psi}$	Kecepatan angular yaw
x	Posisi linier pada sumbu x
\dot{x}	Kecepatan translasi pada sumbu x
y	Posisi linier pada sumbu y
\dot{y}	Kecepatan translasi pada sumbu z
z	Posisi linier pada sumbu z / ketinggian
\dot{z}	Kecepatan translasi pada sumbu z
R	Matriks rotasi
$\dot{\eta}$	Vektor kecepatan angular roll, pitch, dan yaw
ω	Vektor kecepatan liner pada sumbu x, y, dan z
J	Matriks inersia diagonal pada quadrotor
ω	<i>Angular body rates</i>
M_G	Momentum giroskopik
M_B	Momentum gaya
J_r	Inersia motor
Ω_r	Kecepatan 4 motor total $\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$
ρ	Kerapatan udara
A	Luas penampang baling-baling
C_T, C_D	Koefisien aerodinamika
r	Jari-jari baling-baling
Ω_i	Kecepatan sudut dari motor ke- i
K_f	Konstanta thrust

K_M	Konstanta drag
l	Panjang lengan quadrotor
r	Jarak quadrotor diukur dari koordinat bumi
m	Massa dari quadrotor
g	Akselerasi gravitasi $g = 9.81m/s^2$
F_B	Gaya non-gravitasional yang ada pada koordinat quadrotor
I_{xx}	Inersia pada sumbu x
I_{yy}	Inersia pada sumbu y
I_{zz}	Inersia pada sumbu z
U_1	Unit masukan untuk gaya angkat (thrust)
U_2	Unit masukan untuk sudut roll
U_3	Unit masukan untuk sudut pitch
U_4	Unit masukan untuk sudut yaw
K_p	Koefisien parameter kendali proporsional
K_i	Koefisien parameter kendali integral
K_d	Koefisien parameter kendali derivatif
$e(t)$	Selisih variabel terukur dengan set poin

ABSTRACT

Currently Unmanned Aerial Vehicle (UAV) is undergoing rapid development. Quadrotor is one of the popular types of UAVs. Quadrotor has many advantages compared to other types of UAVs. One of the advantage is that the quadrotor can fly hover. Therefore quadrotor can be used in applications that require precision. Quadrotor consists of 4 motor used to rotate the propeller. Thrust force is produced by rotation of the propeller. There are 6 outputs that can be measured from the quadrotor which is translation position and rotation position (or attitude). Because input is less than output then quadrotor is under-actuated system. Therefore, quadrotor system is susceptible non-linear, instability, and control difficult.

In this research is presented a mathematical model that represents the dynamics of a quadrotor. Beside discuss about the math model of quadrotor, this research proposes a PID tuning method that will be used to stabilize hover mode of quadrotor. Value of parameter PID control obtained from LQR method. Quadrotor model and control system proposed will be run on matlab simulink. To find out the performance level of the proposed method, the results will be compared with conventional PID control systems and LQR control systems. Respond obtained from the simulation shows that the proposed control system has the fastest settling time and without but classic LQR control is the best in robustness.

Keywords : Quadrotor, PID Control, LQR Control, LQR-PID, PID Tuning

INTISARI

Sekarang ini *Unmanned Aerial Vehicle* (UAV) sedang mengalami perkembangan yang pesat. Salah satu jenis UAV yang populer adalah quadrotor. Quadrotor memiliki banyak keunggulan dibandingkan dengan jenis UAV lainnya. Salah satu keunggulannya adalah quadrotor dapat melakukan terbang hover. Oleh karena itu maka quadrotor dapat dimanfaatkan pada aplikasi yang membutuhkan tingkat presisi yang baik. Quadrotor terdiri atas 4 buah motor yang digunakan untuk memutar baling-baling. Gaya angkat terbentuk dari putaran yang dihasilkan baling-baling. Ada 6 keluaran yang dapat di ukur dari quadrotor yaitu posisi translasi dan posisi rotasi. karena jumlah masukan kurang dari jumlah keluaran maka quadrotor termasuk dalam sistem yang *under-actuated*. Sistem yang *under-actuated* merupakan sistem yang tidak linier dan rentan tidak stabil sehingga sulit untuk dikendalikan.

Pada penelitian ini dipaparkan model matematika yang merepresentasikan dinamika dari sebuah quadrotor. Selain itu pada penelitian ini mengusulkan sebuah metode tuning PID yang akan digunakan untuk menstabilkan mode hover dari quadrotor. Metode tuning yang digunakan didapatkan dari sistem kendali lain yaitu LQR. Hasil dari perancangan pada penelitian ini akan dijalankan pada matlab simulink. Untuk mengetahui tingkat performa dari metode yang diusulkan maka hasilnya akan dibandingkan dengan sistem kendali PID konvensional dan sistem kendali klasik LQR. Dari hasil simulasi didapatkan bahwa metode yang diusulkan menghasilkan respon yang paling cepat dan tanpa adanya *overshoot* namun dari sisi *robustness* sistem kendali klasik LQR merupakan yang terbaik.

Kata kunci – Quadrotor, Kendali PID, Kendali LQR, LQR-PID, penalaan PID

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
PERNYATAAN.....	iii
PRAKATA.....	iv
ARTI LAMBANG DAN SINGKATAN	vi
ABSTRACT	viii
INTISARI.....	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat dan Kontribusi Penelitian	4
1.6 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI.....	6
2.1 Tinjauan Pustaka	6
2.2 Landasan Teori	9
2.2.1 Pengenalan Quadrotor.....	9
2.2.2 Model Quadrotor.....	13
2.2.2.1 Kinematika Quadrotor	15
2.2.2.2 Dinamika Quadrotor	16
2.2.3 Sistem Kendali PID.....	24
2.2.3.1 Kendali Proporsional	25
2.2.3.2 Kendali Integral	25
2.2.3.3 Kendali Derivatif.....	26
2.2.3.4 Kendali PID (Proporsional-Integral-Derivatif).....	26
2.2.4 Sistem Kendali LQR	27
BAB III METODOLOGI.....	31
3.1 Perlengkapan Penelitian.....	31
3.2 Alur Penelitian.....	31
3.2.1 Studi Kepustakaan.....	31
3.2.2 Identifikasi Masalah	31
3.2.3 Persiapan Penelitian	32
3.2.4 Desain Model dan Sistem Kendali	33
3.2.5 Pengujian.....	33

3.2.6	Analisis Hasil dan Pembahasan	33
3.2.7	Penulisan Laporan	33
3.3	Perancangan Penelitian	34
3.3.1	Model Quadrotor	34
3.3.2	Kendali PID.....	35
3.3.2.1	Sistem Kendali PID Ketinggian.....	36
3.3.2.2	Sistem Kendali PID RPY	37
3.3.3	Kendali LQR	39
3.3.4	Kendali LQR-PD.....	41
BAB IV	HASIL DAN PEMBAHASAN.....	45
4.1	Pengujian Model (Kalang terbuka)	45
4.1.1	Pengujian Ketinggian	45
4.1.1.1	Pengujian keadaan naik	46
4.1.1.2	Pengujian keadaan turun.....	47
4.1.1.3	Pengujian keadaan melayang.....	48
4.1.2	Pengujian Roll.....	49
4.1.3	Pengujian Pitch.....	50
4.1.4	Pengujian Yaw.....	51
4.2	Pengujian Sistem Kendali	52
4.2.1	Pengujian Sistem Kendali PID	52
4.2.2	Pengujian Sistem Kendali LQR	55
4.2.3	Pengujian Sistem Kendali LQR-PID	59
4.3	Perbandingan Sistem Kendali PID, LQR, dan LQR-PID.....	61
4.4	Pengujian dengan Gangguan.....	63
4.5	Pengujian <i>Robustness</i>	65
4.5.1	Parameter -50%.....	65
4.5.2	Parameter +50%.....	66
BAB V	KESIMPULAN DAN SARAN.....	68
5.1	Kesimpulan	68
5.2	Saran	68
DAFTAR	PUSTAKA	70
LAMPIRAN	73

DAFTAR GAMBAR

Gambar 2.1 Quadrotor Konfigurasi + (Plus).....	10
Gambar 2.2 Gerakan Vertikal.....	11
Gambar 2.3 Gerakan Roll	12
Gambar 2.4 Gerakan Pitch	12
Gambar 2.5 Gerakan Yaw	13
Gambar 2.6 Koordinat Referensi pada Quadrotor (sumbu inersia x_o, y_o, z_o dan sumbu badan quadrotor x_b, y_b, z_b).....	14
Gambar 2.7 Gaya dan Momentum pada Quadrotor	19
Gambar 2.8 Diagram Blok Sistem Kendali PID	27
Gambar 2.9 Diagram Blok Sistem Kendali LQR.....	29
 Gambar 3.1 Diagram Alur Penelitian.....	 32
Gambar 3.2 Diagram Blok Sistem Kendali PID Ketinggian	36
Gambar 3.3 Diagram Blok Sistem Kendali PID RPY	38
 Gambar 4.1 Blok Model Quadrotor	 46
Gambar 4.2 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan naik.....	47
Gambar 4.3 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan turun	48
Gambar 4.4 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan melayang	49
Gambar 4.5 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada gerakan roll.....	50
Gambar 4.6 Respon pengujian gerakan pitch	51
Gambar 4.7 Respon pengujian gerakan yaw	52
Gambar 4.8 Model Simulink Sistem Kendali PID pada Ketinggian	53
Gambar 4.9 Respon pengujian sudut roll pada berbagai nilai K_p, K_i , dan K_d ...	54
Gambar 4.10 Respon pengujian kendali PID	54
Gambar 4.11 Diagram Blok Pengujian sistem kendali LQR	58
Gambar 4.12 Respon pengujian kendali LQR	59
Gambar 4.12 Respon pengujian kendali LQR-PID.....	60
Gambar 4.14 Perbandingan Respon Kendali PID, LQR, dan LQR-PID	62
Gambar 4.15 Sinyal kendali PID, LQR dan LQR-PID pada masing-masing motor	62
Gambar 4.16 Blok Simulink Gangguan	63
Gambar 4.17 Bentuk Unit Step dari Gangguan	64
Gambar 4.18 Respon Ketinggian dan Sudut RPY dengan Gangguan	64
Gambar 4.18 Respon (a) Ketinggian dan (b) yaw pada parameter -50%	66
Gambar 4.19 Respon (a) roll dan (b) pitch pada parameter -50%	66
Gambar 4.20 Respon (a) Ketinggian dan (b) yaw pada parameter +50%	66

Gambar 4.21 Respon (a) roll dan (b) pitch pada parameter +50%	67
---	----

DAFTAR TABEL

Tabel 2.1 Literature Review Penelitian sebelumnya	8
Tabel 2.2 Karakteristik Respon Perubahan Parameter Sistem Kendali PID	27
Tabel 4.1 Parameter Model Quadrotor	45
Tabel 4.2 Kondisi Awal Quadrotor dan Set poin dari Sistem kendali PID	53
Tabel 4.3 Data respon kendali PID	55
Tabel 4.4 Nilai maksimal sudut RPY dan ketinggian	55
Tabel 4.5 Nilai masukan (U)	56
Tabel 4.6 Data respon kendali LQR	59
Tabel 4.7 Parameter PID dengan penalaan berbasis LQR	59
Tabel 4.8 Data respon kendali LQR-PID	61
Tabel 4.9 Perbandingan Respon Sistem Kendali PID, LQR, LQR-PID	61
Tabel 4.10 Nilai Parameter I_{xx} , I_{yy} , dan I_{zz} pada parameter -50%	65
Tabel 4.11 Nilai Parameter I_{xx} , I_{yy} , dan I_{zz} pada parameter +50%.....	66

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebutuhan akan robot pada aplikasi atau tugas yang berbahaya dan sulit dijangkau oleh manusia secara langsung saat ini menjadi topik yang banyak diminati oleh para peneliti dalam berbagai bidang, salah satu bidang yang menarik adalah wahana udara nirawak atau *Unmanned Aerial Vehicles (UAV)*. UAV merupakan wahana atau armada terbang yang dalam pengoperasiannya tidak membutuhkan manusia didalam armada tersebut [1]. Dalam pengoperasiannya, armada atau wahana terbang ini menggunakan kendali jarak jauh yang sifatnya nirkabel [2]. Terdapat 2 konfigurasi yang ada pada UAV yaitu sayap tetap (*fixed-wing*) dan sayap berputar (*rotary-wing*).

UAV jenis sayap tetap dapat terbang dengan jarak yang jauh dan mampu terbang dengan ketinggian yang cukup tinggi namun memiliki kekurangan dalam kemampuan bermanuver. Sedangkan konfigurasi yang lain, jenis sayap berputar atau biasa disebut *rotorcraft* memiliki keunggulan dalam hal kemampuan bermanuver, dapat terbang melayang (*hover*) pada suatu lokasi tertentu, lepas landas dan mendarat secara vertikal (*vertical take-off landing*) [3] yang tidak dapat dilakukan oleh UAV sayap tetap dimana membutuhkan sebuah landasan pacu. Karena hal tersebut, pemanfaatan UAV jenis sayap berputar atau *rotorcraft* sangat luas pada berbagai bidang aplikasi baik dari sisi sipil maupun militer.

Banyak aplikasi yang dapat dikerjakan dalam pemanfaatan *rotorcraft*, diantaranya adalah pemetaan suatu daerah, fotografi dan video udara, misi pencarian dan penyelamatan dalam kondisi bencana [5] bahkan dalam bidang hiburan *rotorcraft* dapat digunakan dalam seni pertunjukkan seni tari [4]. Selain dalam bidang sipil, *rotorcraft* juga digunakan dalam bidang militer yaitu sebagai alat mata-mata pada daerah musuh dan pengawasan daerah rawan perang [6]. Salah satu jenis *rotorcraft* yang banyak menjadi obyek penelitian adalah quadrotor atau

quadcopter. Jenis ini menjadi populer karena didukung oleh perkembangan teknologi pada bidang sensor dan aktuator, bidang mikroprosesor atau sistem tertanam, dan bidang penyimpanan sumber daya listrik seperti baterai. Quadrotor adalah sebuah armada terbang yang memiliki 4 buah motor yang digunakan untuk memutar baling-baling sehingga menghasilkan gaya angkat. Baling-baling yang digunakan adalah jenis tetap (*fixed-pitch blade*) dimana berbeda dengan jenis helikopter.

Helikopter konvensional juga merupakan salah satu jenis *rotorcraft* namun karena mekanisme pegerakannya yang cukup kompleks membuat quadrotor lebih banyak disukai dan berkembang dalam bidang penelitian. Dengan mengatur kecepatan setiap motor yang ada pada quadrotor akan berimbas berbedanya gaya angkat pada masing-masing motor sehingga kita dapat mengatur gerakan terbang quadrotor. Luasnya bidang aplikasi yang dapat menjadi peluang penggunaan quadrotor maka kestabilan terbang yang stabil dan tahan terhadap gangguan merupakan suatu hal yang krusial. Berkaitan dengan kestabilan terbang tentunya yang dimaksud adalah tentang sistem kendali yang ada pada quadrotor tersebut.

Secara umum quadrotor terdiri atas beberapa komponen penyusun seperti *motor brushless*, *electronics speed controller* (ESC), *flight controller*, baling-baling, kerangka utama, baterai, dan *radio control* [7]. Sebuah sistem kendali yang telah dirancang kemudian ditanamkan pada *flight controller* tersebut agar quadrotor dapat terbang sesuai dengan parameter-parameter yang diberikan. Pada penelitian Zaki Mustapa [8], sistem kendali PID diimplementasikan guna menjaga kestabilan ketinggian pada quadrotor. Besar sudut roll, pitch, dan yaw dianggap konstan pada nilai 0 sesuai dengan kondisi quadrotor pada saat melakukan *hover* sehingga hanya pada ketinggian saja sistem kendali PID diaplikasikan.

Selanjutnya pada penelitian Hugo Meric [9], persamaan *Newton-Euler* dan *Lagrange-Euler* digunakan untuk merumuskan model gerak translasi dan rotasi dari quadrotor. Ketinggian dan sudut roll, pitch, yaw dikendalikan oleh masing-masing blok sistem kendali PD. Meskipun terdapat keunggulan mudah dalam implementasi, sistem kendali PD kestabilannya tidak lebih baik dibandingkan dengan dengan sebelumnya. Nurul Amirah Ismail [13] mensimulasikan sistem

kendali LQR guna mengatur ketinggian dan sudut roll, pitch, yaw (RPY). *Gain* matriks K pada kendali LQR menjadi umpan balik pada sistem yang kemudian hasil responnya dibandingkan dengan set poin. Sistem kendali LQR yang dijalankan hasilnya lebih baik dibandingkan sistem kendali PD karena memiliki respon yang lebih cepat.

Pada penelitian yang dilakukan oleh Faiz Ahmed [14] *attitude* atau sudut roll, pitch, dan yaw (RPY) model dari quadrotor distabilkan menggunakan metode kontrol Fuzzy-PID dimana nilai parameter dari PID didapatkan dari logika fuzzy. Penelitian selanjutnya dilakukan oleh Tnunay Hilton [15] yang membahas tentang kendali PD dengan *auto-tuning* berbasis *loop shaping* dimana menggunakan 2 metode yaitu *recursive least square* dan *gradient method*. Metode tersebut digunakan untuk mengestimasi parameter K_p dan K_d dari kendali PD.

Pada penelitian ini akan menitikberatkan pada bagaimana mendapatkan parameter K_p , K_i , dan K_d karena parameter tersebut sangat mempengaruhi hasil respon yang terjadi. Sistem kendali PID dipilih dalam penelitian ini karena struktur penyusunnya sederhana dan mudah dalam implementasinya.

1.2 Perumusan Masalah

Perumusan masalah dalam penelitian ini adalah sebagai berikut.

1. Sistem quadrotor merupakan sistem yang nonlinier dan dinamis serta sistem yang rentan terjadi *coupling*, pada setiap perubahan dari salah satu keluarannya saling berpengaruh terhadap karakteristik quadrotor. Sehingga rumit dalam mengatur koefisien pengendali yang digunakan.
2. Quadrotor adalah sistem dengan beberapa masukan dan beberapa keluaran atau *multiple input multiple output* (MIMO) dan dalam aktualnya susah dalam metode kendalinya sehingga diperlukan penyederhanaan model dari quadrotor.

1.3 Batasan Masalah

Batasan masalah yang ada pada penelitian ini adalah sebagai berikut :

1. Sistem kendali yang dirancang diaplikasikan untuk menstabilkan posisi ketinggian dan sudut roll, pitch, yaw (RPY) dari sebuah model quadrotor pada kondisi *hover* atau melayang.
2. Bentuk pemodelan dari quadrotor berdasarkan dari persamaan *euler-lagrange* yang persamaannya keluarannya menghasilkan 6 keadaan keluaran.
3. Fokus penelitian adalah pada kondisi melayang atau *hover* sehingga tidak ada sistem kendali yang diaplikasikan pada gerak translasi X dan Y sehingga perubahan pada gerak translasi X dan Y dapat diabaikan.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah merancang sistem kendali PID dengan penalaan berdasarkan LQR (*Linear Quadratic Regulator*) pada wahana terbang quadrotor. Hasilnya kemudian dibandingkan dengan sistem kendali PID dan LQR.

1.5 Manfaat dan Kontribusi Penelitian

Penelitian ini berorientasi pada pengembangan sistem kendali yang digunakan pada quadrotor, dan diharapkan dapat memberikan manfaat sebagai berikut :

1. Dapat digunakan untuk mengendalikan quadrotor pada sudut RPY dan posisi ketinggian sesuai dengan nilai set poin yang diberikan.
2. Memberikan kontribusi pengembangan sistem kendali terutama pada bidang sistem kendali pada quadrotor sehingga dapat dikendalikan dengan baik serta dapat tahan terhadap gangguan.
3. Menyajikan model quadrotor pada simulasi menggunakan matlab yang dapat digunakan untuk penelitian selanjutnya.
4. Dengan hasil sistem kendali yang stabil sehingga resiko hilang kendali pada quadrotor berkurang dan poin selanjutnya adalah pemanfaatan quadrotor dalam aplikasi-aplikasi tertentu. Misal untuk pengamatan daerah bencana atau pemeriksaan jaringan listrik.

1.6 Sistematika Penulisan

Terdapat 5 bab dan 1 lampiran pada laporan tesis ini :

1. Bab 1 (Pendahuluan)

Bab ini berisi tentang latar belakang yang menjadi dasar penelitian ini dilakukan, perumusan masalah, kontribusi penelitian yang menjelaskan penelitian yang sudah dilakukan sebelumnya dan rencana penelitian ini, tujuan dan manfaat penelitian dan sistematika penulisan laporan penelitian.

2. Bab 2 (Tinjauan Pustaka dan Landasan Teori)

Berisi tinjauan pustaka dan landasan teori yang menjadi dasar dalam penelitian ini. Termasuk didalamnya adalah dasar tentang quadrotor termasuk didalamnya terdapat model matematika dari obyek penelitian dan materi tentang sistem kendali yang digunakan pada penelitian ini PID, LQR, dan LQR-PID.

3. Bab 3 (Metodologi Penelitian)

Bab 3 berisi tentang material pada penelitian ini, skema atau sistematik penelitian yang dilakukan, perancangan model quadrotor dan sistem kendali PID, LQR dan LQR-PID.

4. Bab 4 (Hasil dan Pembahasan)

Menjelaskan tentang hasil pengujian yang dilakukan pada penelitian ini dan pembahasannya. Ada beberapa pengujian yaitu pengujian kalang terbuka, pengujian menggunakan kendali yang telah dirancang sebelumnya dan terakhir adalah pengujian yang berkaitan dengan *robustness*.

5. Bab 5 (Kesimpulan dan Saran)

Terakhir adalah berisi kesimpulan dari penelitian yang telah dilakukan dan beberapa saran yang dapat dipertimbangkan untuk penelitian selanjutnya agar lebih baik.

6. Lampiran

Berisi kode sumber yang dituliskan pada program matlab dan simulink yang digunakan untuk proses simulasi.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Sistem kendali PID (Proporsional-Integral-Derivatif) merupakan salah satu sistem kendali yang banyak digunakan karena struktur dari bagian-bagiannya cukup sederhana dan mudah diimplementasikan bahkan pada sebuah mikrokontroler 8-bit [17]. Sistem kendali PID merupakan sistem kendali yang membutuhkan umpan balik dari hasil variable yang terjadi, dalam aktualnya umpan balik ini dihasilkan oleh pembacaan sensor. Berkembangnya bidang sensor juga mempengaruhi perkembangan obyek penelitian yang lain terutama dalam bidang robotika. Semakin kecil bentuknya dengan pembacaan hasil yang lebih baik membuat salah satu jenis robot terbang, quadrotor, menjadi sangat berkembang baik sekedar hobi dan profesional bahkan di bidang riset.

Quadrotor merupakan sistem yang kompleks dengan 4 masukan namun mempunyai 6 buah keluaran sehingga sistem ini termasuk sistem yang *under-actuated*. Sistem kendali yang handal sangat diperlukan agar quadrotor dapat melakukan tugas atau terbang dengan baik dan stabil. Disebutkan sebelumnya bahwa sistem kendali PID adalah sistem kendali yang sederhana dan mudah diimplementasikan. Hal ini menyebabkan ada beberapa penelitian yang menggunakan sistem kendali PID terutama dalam mengendalikan obyek berupa quadrotor. Sistem kendali PID terdiri dari 3 buah parameter yaitu K_p , K_i , dan K_d . Nilai masing-masing parameter sangat mempengaruhi hasil dari respon yang terjadi. Secara umum semakin besar nilai K_p akan mempercepat respon yang terjadi, namun apabila nilainya besar maka overshoot akan terjadi bahkan apabila nilainya terlalu besar dapat menyebabkan sistem tidak stabil dan berosilasi. Kendali Integral secara umum berfungsi untuk menghilangkan kesalahan keadaan tunak yang ini tidak dapat dihilangkan oleh pengendali proporsional dan kendali Derivatif secara umum berfungsi untuk meredam respon agar tidak terjadi overshoot [17].

Dalam obyek penelitian quadrotor sistem kendali PID juga digunakan untuk

mengendalikan ketinggiannya. Sudut roll, pitch, dan yaw dianggap nol atau sangat kecil karena quadrotor diasumsikan keadaan melayang atau *hover*. Penelitian yang dilakukan oleh Z. Mustafa [8] terkait pengendalian ketinggian menggunakan sistem kendali PID pada quadrotor menggunakan metode eksperimen guna mendapatkan parameter K_p , K_i , dan K_d . Penelitian lain yang menggunakan sistem kendali PID adalah penelitian yang dilakukan oleh Atheer L. Salih [19] yang menggunakan 4 blok sistem kendali PID pada masing-masing ketinggian, sudut roll, pitch, dan yaw. Model quadrotor yang digunakan adalah hasil dari pendekatan euler-lagrange.

Dalam penelitian [18] sistem kendali PID juga digunakan untuk mengendalikan sudut roll, pitch, dan yaw dari quadrotor. Namun parameter yang digunakan dalam kendali PID didapatkan dari sebuah metode kontrol yang lain yaitu logika fuzzy sehingga disebut Fuzzy-PID. Dalam metode ini didapatkan dahulu sebuah tabel logika dimana untuk mendapatkannya menggunakan metode fuzzy yang dari tabel tersebut digunakan untuk menentukan nilai K_p , K_i , dan K_d dari kontroler PID.

Penelitian terkait kendali PID yang dibandingkan dengan kendali LQR dilakukan oleh S. Khatoon [20]. Model dari quadrotor yang digunakan adalah model persamaan keadaan yang diturunkan dari dinamika quadrotor. Dari hasil respon penelitian tersebut didapatkan hasil bahwa kendali PID lebih baik dalam hal stabilitas sedangkan LQR lebih baik dalam hal kesalahan keadaan tunak dan kecepatan respon. Selain itu ada penelitian [25] yang menggunakan metode penalaan *zieger-nichols* untuk mendapatkan parameter kendali PD. Dalam penelitiannya, Zefang He, merancang sebuah sistem kendali nonlinier yang terdiri atas 2 bagian yaitu bagian linier dan bagian nonlinier. Bagian linier disini adalah kendali PD yang bertindak pada model quadrotor yang dilinierisasi dan bagian nonlinier adalah linierisasi dari umpan balik yang mengubah sistem nonlinier menjadi sistem linier.

Tabel 2.1 Literature Review Penelitian sebelumnya

Peneliti	Sistem Kendali	Metode Tuning	Keterangan
Z. Mustafa	PID	Experimental	- Diaplikasikan pada obyek ketinggian
AL. Salih	PID	Experimental	- Model Berdasarkan Euler-Lagrange yang di sederhanakan - Diaplikasikan pada Sudut Roll, Pitch, Yaw dan Ketinggian
S. Ahmed	PID	<i>Fuzzy Logic</i>	- Masing-masing parameter K_p , K_i , dan K_d didapatkan dari sebuah tabel logika fuzzy
S. Khatoon	PID & LQR		- PID lebih baik dalam hal stabilitas karena pole-nya bernilai lebih negatif dibandingkan LQR - LQR lebih <i>robust</i> dan memiliki <i>error steady state</i> yang kecil
ZF. Hee	PID	<i>Zieger-Nichols</i>	- Diaplikasikan pada sudut RPY - Mengadopsi kendali non-linier yang terdiri dari kendali linier dan non-linier
LM. Argentim	PID	LQR	- Diaplikasikan hanya pada gerakan vertikal - Tidak ada analisis terkait dengan <i>robustness</i> dari sistem kendali yang digunakan

Pada penelitian sebelumnya yang dilakukan oleh LM. Argentim [28] metode LQR diaplikasikan guna mendapatkan parameter dari sebuah sistem kendali PID pada obyek quadrotor. Sistem kendali PID yang diaplikasikan hanya pada ketinggian quadrotor dan belum adanya analisis terkait dengan *robustness* yang ada pada masing-masing sistem kendali. Resume dari penelitian sebelumnya dapat dilihat pada tabel 2.1.

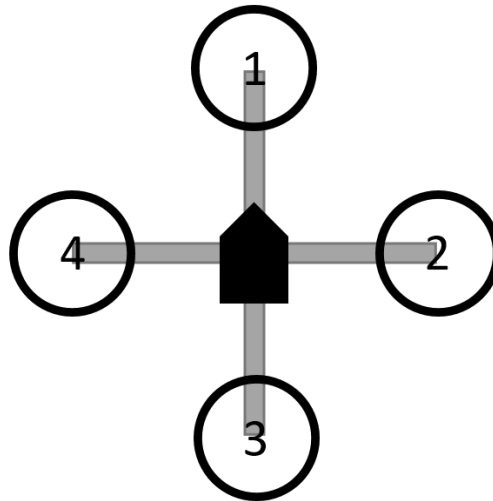
Kontribusi dari penelitian ini adalah memberikan sebuah metode tuning pada sistem kendali PID dengan menggunakan metode LQR. Hasil dari parameter yang didapatkan akan disimulasikan pada obyek penelitian dan hasilnya akan dibandingkan dengan sistem kendali PID dengan metode tuning eksperimental dan sistem kendali LQR. Tidak hanya pada obyek ketinggian quadrotor saja namun akan diaplikasikan pada mode terbang hover yang melibatkan ketinggian, dan *attitude* atau sudut RPY dari sebuah quadrotor. Selain itu pada penelitian ini akan dibahas pula tentang *robustness* dari sistem yang dirancang.

2.2 Landasan Teori

Pada sub-bab ini akan memaparkan tentang dasar-dasar teori yang digunakan dalam penelitian ini yaitu berkaitan dengan model matematika dari quadrotor, sistem kendali PID, LQR, dan PID dengan penalaan berbasis LQR.

2.2.1 Pengenalan Quadrotor

Quadrotor atau juga disebut quadcopter adalah salah satu jenis armada terbang yang memiliki banyak keunggulan dibandingkan dengan armada terbang lainnya. Apabila dibandingkan dengan helikopter konvensional yang sama-sama dapat melakukan lepas landas dan mendarat secara vertikal, quadrotor lebih sederhana dalam struktur mekanisnya. Keunggulan lainnya dibandingkan dengan *fixed wing* adalah quadrotor memiliki manuver yang lebih baik dan bebas sehingga dapat digunakan pada daerah-daerah sempit.



Gambar 2.1 Quadrotor Konfigurasi + (Plus)

Quadrotor terdiri atas 2 bagian besar yaitu bagian mekanik dan bagian elektronik. Bagian mekanik dari quadrotor meliputi bagian dan struktur kerangka badan dari quadrotor sedangkan elektronik terdiri atas perangkat elektronis seperti motor, baterai, esc dan *flight controller*. Terdapat 2 konfigurasi dari bentuk badan quadrotor, yaitu konfigurasi + dan X. Rangka dari quadrotor digunakan untuk tempat peletakan komponen-komponen elektronis. Secara umum letak dari motor atau motor adalah di ujung lengan-lengan dari quadrotor dan perangkat elektronis lainnya berada di tengah badan dari quadrotor. Konfigurasi yang digunakan pada penelitian adalah konfigurasi + karena dari sisi perancangan model lebih sederhana dibandingkan dengan X. Quadrotor konfigurasi + (plus) dapat dilihat pada Gambar 2.1.

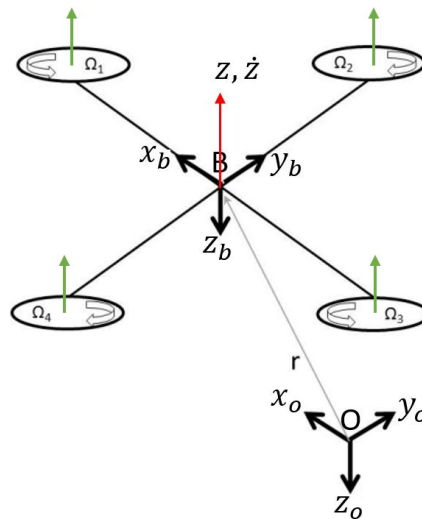
Terdapat 4 gerakan dasar yang ada pada quadrotor. Konfigurasi motor yang digunakan adalah motor depan (1), motor kanan (2), motor belakang (3), dan motor kiri (4). Untuk menghasilkan gerakan-gerakan dasar tersebut adalah dengan cara mengatur kecepatan masing-masing motor yang ada pada quadrotor. Gerakan-gerakan dasar [21] tersebut adalah :

1. Gerakan Vertikal

Gerakan vertikal adalah gerakan naik dan turun. Dalam gerakan ini yang berubah pada kondisi quadrotor adalah ketinggiannya. Pada kasus ini kecepatan ke-4 motor adalah sama besarnya sehingga quadrotor bergerak naik atau turun

sesuai dengan kecepatan ke-4 motor dibandingkan dengan kecepatan minimum.

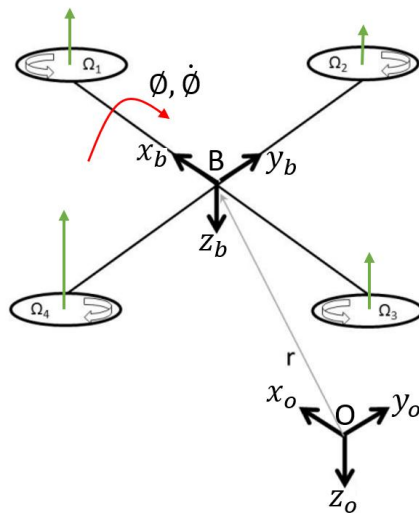
Kecepatan minimum disini adalah kecepatan dimana gaya angkat yang dihasilkan dari ke-4 buah motor tersebut sama dengan beban yang ada pada quadrotor. Apabila kecepatan motor melebihi kecepatan minimum maka quadrotor akan bergerak naik dan apabila kecepatan motor dibawah kecepatan minimum maka quadrotor akan berkurang ketinggiannya. Gerakan ini digambarkan pada gambar 2.2.



Gambar 2.2 Gerakan Vertikal

2. Gerakan Roll

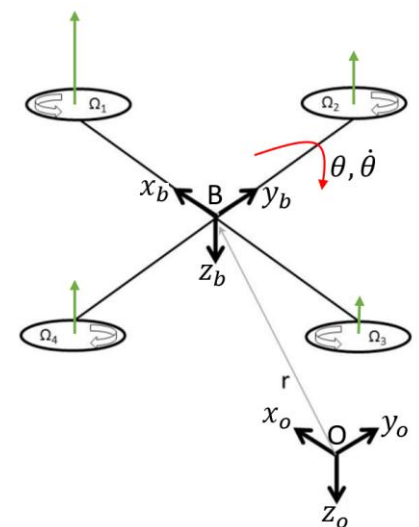
Gerakan ini dipengaruhi oleh kecepatan motor kanan dan kiri. Gerakan ini mengubah sudut roll yang ada pada quadrotor. Nilainya dapat bernilai positif atau negatif sesuai dengan arah pergerakannya. Apabila kecepatan motor kanan (2) lebih besar dibandingkan dengan kecepatan motor kiri (4) maka quadrotor akan bergerak roll kiri dan juga sebaliknya apabila kecepatan motor kiri (4) lebih besar dibanding motor kanan (2) maka quadrotor akan bergerak roll kanan. Penambahan dan pengurangan kecepatan pada masing-masing motor nilainya adalah sama sehingga efeknya bisa seimbang. Gerakan ini digambarkan pada gambar 2.3.



Gambar 2.3 Gerakan Roll

3. Gerakan Pitch

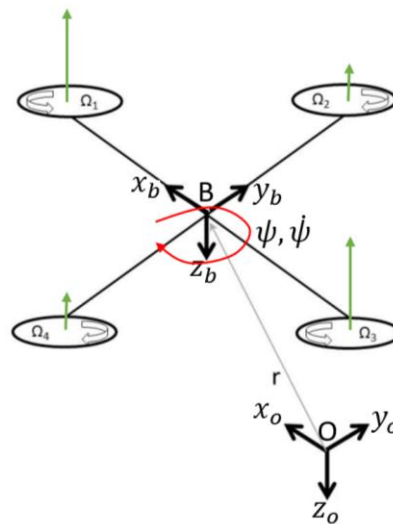
Gerakan ketiga adalah gerakan pitch. Motor depan (1) dan motor belakang (2) adalah motor yang mempengaruhi gerakan ini. Gerakan ini mengubah sudut pitch quadrotor. Sama halnya dengan roll, untuk melakukan gerakan ini salah satu motor (depan (1) / belakang (3)) kecepatannya harus lebih besar dibandingkan motor lainnya. Misal kecepatan motor depan (1) lebih besar dibanding kecepatan motor belakang (3) maka quadrotor akan bergerak pitch mundur dan begitupula sebaliknya. Gerakan ini digambarkan pada gambar 2.4.



Gambar 2.4 Gerakan Pitch

4. Gerakan Yaw

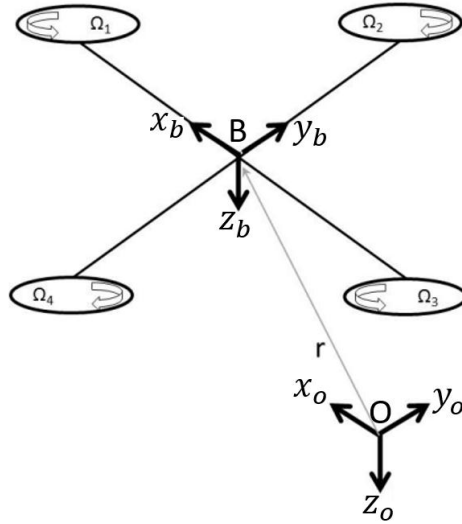
Gerakan terakhir yang ada pada quadrotor adalah gerakan yaw. Perubahan sudut yaw terjadi apabila gerakan ini terjadi. Konfigurasi motor depan (1) dan motor belakang (3) serta motor kanan (2) dan motor kiri (4) mempengaruhi gerakan ini. Apabila motor depan (1) dan belakang (3) kecepatannya lebih besar dibanding motor kanan (2) dan kiri (4) maka quadrotor akan bergerak yaw kiri dan untuk bergerak yaw kanan kecepatan motor kanan (2) dan kiri (4) harus lebih besar dibandingkan dengan kecepatan motor depan (1) dan belakang (3). Gerakan ini digambarkan pada gambar 2.5.



Gambar 2.5 Gerakan Yaw

2.2.2 Model Quadrotor

Untuk menentukan model dari sebuah obyek diperlukan 2 sistem koordinat. Pertama, sistem koordinat bumi yang kondisinya tidak pernah berubah dan sistem koordinat quadrotor yang berubah-ubah mengikuti dari sikap quadrotor [22]. Gambar 2.3 memperlihatkan kedua sistem koordinat yang ada pada quadrotor. Sesuai dengan sistem koordinat tersebut, sudut pitch adalah sudut putar pada sumbu x_b , sudut roll adalah sudut putar pada sumbu y_b , dan sudut yaw adalah sudut putar pada sumbu z_b .



Gambar 2.6 Koordinat Referensi pada Quadrotor (sumbu inersia x_o, y_o, z_o dan sumbu badan quadrotor x_b, y_b, z_b)

Pemodelan dari sistem quadrotor terdiri dari model kinematik dan model dinamika yang diturunkan berdasarkan persamaan Newton-Euler. Sebelum menentukan persamaan model dari quadrotor kita harus mendefinisikan terlebih dahulu vektor keadaan dari quadrotor yang ditunjukkan pada persamaan 2.1.

$$X = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12}]^T \quad (2.1)$$

X merepresentasikan derajat kebebasan quadrotor sesuai dengan persamaan 2.2.

$$X = [\phi \ \dot{\phi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi} \ x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]^T \quad (2.2)$$

dimana ϕ dan $\dot{\phi}$ adalah sudut roll dan kecepatan sudut roll yang ditunjukkan pada gambar 2.3. Notasi θ dan $\dot{\theta}$ adalah sudut pitch dan kecepatan sudut pitch yang pergerakannya ditunjukkan pada gambar 2.4. Notasi ψ dan $\dot{\psi}$ adalah sudut yaw dan kecepatan sudut yaw yang ditunjukkan pada gambar 2.5, sedangkan x dan \dot{x} adalah posisi dan kecepatan translasi pada sumbu x_b yang ditunjukkan pada gambar 2.2. y dan \dot{y} adalah posisi pada sumbu y_b dan kecepatan translasi pada sumbu y_b yang ditunjukkan pada gambar 2.4. Dan terakhir z dan \dot{z} adalah posisi pada sumbu z_b dan kecepatan translasi pada sumbu z_b yang ditunjukkan pada gambar 2.3.

Vektor input U juga harus ditentukan yang isi komponennya adalah

$$U = [U_1 \ U_2 \ U_3 \ U_4]^T \quad (2.3)$$

dimana,

U_1 = Masukan pada gerakan vertikal

U_2 = Masukan pada gerakan roll

U_3 = Masukan pada gerakan pitch

U_4 = Masukan pada gerakan yaw

2.2.2.1 Kinematika Quadrotor

Pada Gambar 2.3 menjelaskan tentang koordinat yang digunakan pada pemodelan sebuah quadrotor. Pada sumbu bumi atau biasa juga disebut sumbu inersia atau sumbu referensi terdiri dari 3 sumbu dan dituliskan dalam notasi x_o, y_o, z_o atau juga dapat dinotasikan dengan N, E , dan D dimana N singkatan dari *North* merupakan sumbu x_o , E dari *East* merupakan sumbu y_o dan D adalah *Downwards* yang merupakan sumbu z_o . Sedangkan untuk koordinat quadrotor berada pada pusat dari badan quadrotor dengan sumbu x_b kearah motor depan (1), sumbu y_b kearah motor kanan (2) dan sumbu z_b kearah tanah atau bawah.

Jarak antara koordinat bumi dan koordinat quadrotor adalah jarak absolut dari pusat quadrotor $r = [x \ y \ z]^T$. Matrik rotasi R merupakan orientasi dari koordinat bumi menuju koordinat quadrotor. Orientasi tersebut adalah roll (ϕ), pitch (θ), dan yaw (ψ) yang merepresentasikan besar putaran sudut pada sumbu- x_b, y_b dan z_b secara berurutan. Matriks rotasi R sesuai dengan kaidah dasar matriks rotasi didapatkan seperti pada persamaan 2.4.

$$R = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\phi s\psi & s\phi s\theta s\psi + c\theta c\psi & c\phi s\theta s\psi - s\theta c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (2.4)$$

Dimana $s\phi = \sin \phi$, $c\phi = \cos \phi$, $s\theta = \sin \theta$, $c\theta = \cos \theta$, $s\psi = \sin \psi$, $c\psi = \cos \psi$

Matriks rotasi R diperlukan dalam mendapatkan persamaan model dinamika dari quadrotor. Matriks ini memiliki peranan penting karena beberapa persamaan dihitung pada koordinat bumi sedangkan lainnya dihitung pada koordinat badan quadrotor sehingga untuk menghubungkan kedua persamaan ini, transformasi dari satu koordinat menuju koordinat lain (dalam hal ini koordinat bumi dan badan quadrotor) sangat diperlukan.

Untuk mendapatkan data terkait dengan kecepatan sudut dari quadrotor, maka digunakan sensor IMU (*Inertial Measurement Unit*). Untuk menghubungkan antara *Euler rates* $\dot{\eta} = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ yang diukur pada koordinat bumi dan *angular body rates* $\omega = [p \ q \ r]^T$ maka diperlukan sebuah transformasi yang ditunjukkan pada persamaan 2.5.

$$\omega = R_r \dot{\eta} \quad (2.5)$$

dimana,

$$R_r = \begin{bmatrix} 1 & 1 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\sin\theta \end{bmatrix}$$

Pada sekitar posisi melayang atau *hover*, diasumsikan sudut yang terjadi sangat kecil (mendekati 0 rad) sehingga nilai dari $\cos\phi = 1$, $\cos\theta = 1$, dan $\sin\phi = \sin\theta = 0$. Jadi R_r dapat disederhanakan menjadi sebuah matriks $I[3 \times 3]$

2.2.2.2 Dinamika Quadrotor

Pergerakan dari quadrotor dapat dikelompokkan menjadi 2 subsistem yaitu subsistem rotasi / putar (sudut roll, pitch, dan yaw) dan subsistem translasi (posisi x, y, dan z atau ketinggian). Persamaan pada pergerakan rotasi diturunkan dari koordinat bumi berdasarkan metode *Newton-Euler* dengan bentuk persamaan umumnya adalah

$$J_q \dot{\omega} + \omega \times J_q \omega + M_G = M_B \quad (2.6)$$

dimana,

J_q = Matriks inersia diagonal pada quadrotor

ω = *Angular body rates*

M_G = Momentum giroskopik

M_B = Momentum gaya

$J_q \dot{\omega}$ dan $\omega \times J_q \omega$ merepresentasikan perubahan momentum kecepatan putar pada koordinat quadrotor. Sedangkan M_G merupakan momentum giroskopik yang terjadi karena adanya inersia motor J_r . Momentum giroskopik didefinisikan menjadi

$\omega \times [0 \ 0 \ J_r \Omega_r]^T$, sehingga dapat dituliskan kembali persamaan rotasi dari pergerakan quadrotor menjadi persamaan 2.7.

$$J_q \dot{\omega} + \omega \times J_q \omega + \omega \times [0 \ 0 \ J_r \Omega_r]^T = M_B \quad (2.7)$$

dimana,

J_r = inersia motor

Ω_r = kecepatan motor $\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$

Matriks inersia dari quadrotor adalah sebuah matriks diagonal yang mana elemen dari selain diagonalnya adalah 0. Matriks inersia dari quadrotor ditunjukkan pada persamaan 2.8. I_{xx} , I_{yy} , dan I_{zz} merupakan momentum inersia pada sumbu x_b , y_b dan z_b pada koordinat badan quadrotor.

$$J_q = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (2.8)$$

Variabel M_B merupakan komponen terakhir pada persamaan 2.6. Untuk membahas variabel ini dibutuhkan pendefinisian 2 buah efek fisik yang ada pada quadrotor, yaitu gaya dan momentum aerodinamika dan momentum yang dihasilkan oleh motor. Sebagai efek dari putaran yang terjadi, terdapat gaya yang dihasilkan dan disebut gaya aerodinamika atau gaya angkat dan terdapat momentum yang dinamakan momentum aerodinamika. Persamaan 2.9 dan 2.10 menunjukkan gaya aerodinamika F_i dan momentum aerodinamika M_i yang dihasilkan motor ke- i .

$$F_i = \frac{1}{2} \rho A C_T r^2 \Omega_i^2 \quad (2.9)$$

$$M_i = \frac{1}{2} \rho A C_D r^2 \Omega_i^2 \quad (2.10)$$

dimana,

ρ = kerapatan udara

A = luas penampang baling-baling

C_T, C_D = koefisien aerodinamika

r = jari-jari baling-baling

Ω_i = kecepatan sudut dari motor ke- i

Gaya dan momentum aerodinamika tergantung dari bentuk baling-baling dan kerapatan udara. Obyek penelitian adalah sebuah quadrotor, dan umumnya ketinggian terbang dari quadrotor itu terbatas maka kerapatan udara dapat diasumsikan tetap sehingga persamaan 2.9 dan 2.10 dapat disederhanakan menjadi persamaan 2.11 dan 2.12.

$$F_i = K_f \Omega_i^2 \quad (2.11)$$

$$M_i = K_M \Omega_i^2 \quad (2.12)$$

dimana,

K_f = konstanta gaya aerodinamika

K_M = konstanta momentum aerodinamika

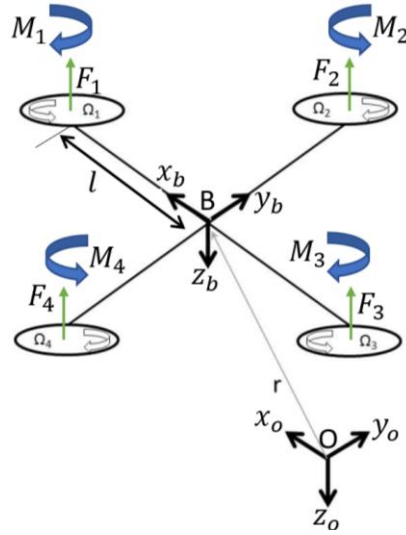
Untuk mendapatkan nilai kedua konstanta diatas dapat diperoleh berdasarkan percobaan untuk masing-masing baling-baling yang digunakan. Dengan mengidentifikasi gaya dan momentum yang dihasilkan dari baling-baling kita dapat mempelajari momentum M_B . Gambar 2.7 memperlihatkan gaya dan momentum yang ada pada sebuah quadrotor. Setiap motor menyebabkan gaya keatas F_i dan menghasilkan pula momentum M_i yang arahnya berlawanan dengan arah putaran motor ke- i .

Dengan menggunakan aturan tangan kanan pada koordinat quadrotor, F_2 dikalikan dengan panjang lengan l menghasilkan momentum negatif pada sumbu x_b dan dengan cara yang sama, F_4 menghasilkan momentum positif. Sehingga total momentum yang terjadi pada sumbu x_b dapat dituliskan

$$M_x = -F_2 l + F_4 l$$

$$M_x = -(K_f \Omega_2^2) l + (K_f \Omega_4^2) l$$

$$M_x = l K_f (-\Omega_2^2 + \Omega_4^2) \quad (2.13)$$



Gambar 2.7 Gaya dan Momentum pada Quadrotor

Sama halnya cara sebelumnya digunakan untuk mencari total momentum yang terjadi pada sumbu y_b sehingga dapat dituliskan

$$\begin{aligned}
 M_y &= F_1 l - F_3 l \\
 M_y &= (K_f \Omega_1^2) l - (K_f \Omega_3^2) l \\
 M_y &= l K_f (\Omega_1^2 - \Omega_3^2)
 \end{aligned} \tag{2.14}$$

Dan terakhir total momentum yang terjadi pada sumbu z_b adalah

$$\begin{aligned}
 M_z &= M_1 - M_2 + M_3 - M_4 \\
 M_z &= (K_M \Omega_1^2) - (K_M \Omega_2^2) + (K_M \Omega_3^2) - (K_M \Omega_4^2) \\
 M_z &= K_M (\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2)
 \end{aligned} \tag{2.15}$$

Dengan mengkombinasi persamaan 2.13, 2.14, dan 2.15 kita dapat sebuah vektor dari total momentum yang terjadi pada sebuah obyek quadrotor yang ditunjukkan pada persamaan 2.16.

$$M_B = \begin{bmatrix} l K_f (-\Omega_2^2 + \Omega_4^2) \\ l K_f (\Omega_1^2 - \Omega_3^2) \\ K_M (\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \end{bmatrix} \tag{2.16}$$

Vektor masukan kendali U terdiri dari 4 masukan yaitu direpresentasikan dengan notasi U_1-U_4 yang dituliskan pada persamaan 2.3. Komponen dari masing-

masing variable tersebut dapat kita jabarkan sebagai berikut

$$U_1 = K_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \quad (2.17)$$

$$U_2 = K_f(-\Omega_2^2 + \Omega_4^2) \quad (2.18)$$

$$U_3 = K_f(\Omega_1^2 - \Omega_3^2) \quad (2.19)$$

$$U_4 = K_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \quad (2.20)$$

Sehingga persamaan 2.16 dapat ditulis ulang menjadi

$$M_B = \begin{bmatrix} l U_2 \\ l U_3 \\ U_4 \end{bmatrix} \quad (2.21)$$

Dengan mensubstitusi persamaan 2.21 kedalam persamaan 2.6 dan dijabarkan tiap masing-masing komponennya maka didapatkan

$$\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ J_r \Omega_r \end{bmatrix} = \begin{bmatrix} l U_2 \\ l U_3 \\ U_4 \end{bmatrix}$$

Kemudian dapat disederhanakan menjadi

$$\begin{bmatrix} I_{xx} \ddot{\phi} \\ I_{yy} \ddot{\theta} \\ I_{zz} \ddot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\theta} I_{zz} \dot{\psi} - \dot{\psi} I_{yy} \dot{\theta} \\ \dot{\psi} I_{xx} \dot{\phi} - \dot{\phi} I_{zz} \dot{\psi} \\ \dot{\phi} I_{yy} \dot{\theta} - \dot{\theta} I_{xx} \dot{\phi} \end{bmatrix} + \begin{bmatrix} \dot{\theta} J_r \Omega_r \\ -\dot{\phi} J_r \Omega_r \\ 0 \end{bmatrix} = \begin{bmatrix} l U_2 \\ l U_3 \\ U_4 \end{bmatrix} \quad (2.22)$$

Sehingga persamaan gerak rotasi RPY dari model quadrotor adalah

$$\ddot{\phi} = \frac{1}{I_{xx}} U_2 - \frac{J_r}{I_{xx}} \dot{\theta} \Omega_r + \frac{I_{yy}}{I_{xx}} \dot{\psi} \dot{\theta} - \frac{I_{zz}}{I_{xx}} \dot{\theta} \dot{\psi} \quad (2.23)$$

$$\ddot{\theta} = \frac{1}{I_{yy}} U_3 - \frac{J_r}{I_{yy}} \dot{\phi} \Omega_r + \frac{I_{zz}}{I_{yy}} \dot{\phi} \dot{\psi} - \frac{I_{xx}}{I_{yy}} \dot{\psi} \dot{\phi} \quad (2.24)$$

$$\ddot{\psi} = \frac{1}{I_{zz}} U_4 + \frac{I_{xx}}{I_{zz}} \dot{\theta} \dot{\phi} - \frac{I_{yy}}{I_{zz}} \dot{\phi} \dot{\theta} \quad (2.25)$$

dapat disederhanakan dengan mendefinisikan

$$a_1 = \frac{I_{yy} - I_{zz}}{I_{xx}}$$

$$\begin{aligned}
a_2 &= \frac{J_r}{I_{xx}} & b_1 &= \frac{l}{I_{xx}} \\
a_3 &= \frac{I_{zz} - I_{xx}}{I_{yy}} & b_2 &= \frac{l}{I_{yy}} \\
a_4 &= \frac{J_r}{I_{yy}} & b_3 &= \frac{l}{I_{zz}} \\
a_5 &= \frac{I_{xx} - I_{yy}}{I_{zz}}
\end{aligned}$$

Dengan memasukkan variabel $a_1 - a_5$ dan $b_1 - b_3$ kedalam persamaan 2.23 sampai 2.25 kita mendapatkan persamaan model yang lebih sederhana seperti ditunjukkan pada persamaan 2.26 sampai 2.28.

$$\ddot{\phi} = b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6 \quad (2.26)$$

$$\ddot{\theta} = b_2 U_3 + a_4 x_2 \Omega_r + a_3 x_2 x_6 \quad (2.27)$$

$$\ddot{\psi} = b_3 U_4 + a_5 x_2 x_4 \quad (2.28)$$

Persamaan gerak translasi yang ada pada quadrotor didasarkan pada hukum Newton kedua dan persamaan ini menggunakan referensi koordinat bumi.

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + RF_B \quad (2.29)$$

dimana,

$r = [x \ y \ z]^T$ = jarak quadrotor diukur dari koordinat bumi

m = massa dari quadrotor

g = akselerasi gravitasi $g = 9.81m/s^2$

F_B = gaya non-gravitasiional yang ada pada koordinat quadrotor

Ketika quadrotor pada posisi level (tidak terjadi gerakan roll dan pitch) hanya gaya non-gravitasiional yang terjadi dan gaya angkat yang dihasilkan dari putaran baling-baling adalah sebanding dengan kuadrat dari kecepatan putar dari baling-baling sesuai dengan persamaan 2.11. Sehingga, gaya non-gravitasiional yang

terjadi pada quadrotor, F_B , dapat dituliskan menjadi

$$F_B = \begin{bmatrix} 0 \\ 0 \\ -K_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix} \quad (2.30)$$

Dua baris teratas nilainya 0 karena tidak ada gaya yang terjadi pada arah x_o dan y_o dan baris terakhir merupakan gaya angkat yang dihasilkan oleh ke-4 baling-baling. Tanda negative pada baris terakhir menandakan arah gaya berlawanan dengan sumbu z_o dimana gaya yang bekerja kearah atas sedangkan arah sumbu z_o kearah tanah / bawah. Dengan mensubstitusi persamaan 2.17 pada persamaan 2.30 maka didapatkan

$$F_B = \begin{bmatrix} 0 \\ 0 \\ -U_1 \end{bmatrix} \quad (2.31)$$

Dengan menguraikan persamaan 2.29 didapatkan

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\phi s\psi & s\phi s\theta s\psi + c\theta c\psi & c\phi s\theta s\psi - s\theta c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -U_1 \end{bmatrix}$$

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \begin{bmatrix} (s\phi s\psi + c\phi c\psi s\theta)(-U_1) \\ (c\phi s\psi s\theta - c\psi s\phi)(-U_1) \\ (c\phi c\theta)(-U_1) \end{bmatrix} \quad (2.32)$$

Persamaan 2.32 kemudian diurai tiap komponen sehingga didapatkan

$$\ddot{x} = \frac{U_1}{m} (\sin\phi \sin\psi + \cos\phi \cos\psi \sin\theta) \quad (2.33)$$

$$\ddot{y} = \frac{U_1}{m} (\cos\phi \sin\psi \sin\theta - \cos\psi \sin\phi) \quad (2.34)$$

$$\ddot{z} = g - \frac{U_1}{m} (\cos\phi \cos\theta) \quad (2.35)$$

Dituliskan ulang sesuai dengan variabel keadaan X sehingga menjadi

$$\ddot{x} = \frac{-U_1}{m} (\sin x_1 \sin x_5 + \cos x_1 \cos x_5 \sin x_3) \quad (2.36)$$

$$\ddot{y} = \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1) \quad (2.37)$$

$$\ddot{z} = g - \frac{-U_1}{m}(\cos x_1 \cos x_3) \quad (2.38)$$

Kita sudah mendapatkan persamaan dinamika dari quadrotor, pergerakan rotasi ditunjukkan pada persamaan 2.26 sampai 2.28 dan persamaan gerak translasi ditunjukkan pada persamaan 2.36 sampai 2.38 sehingga persamaan lengkap model matematika dari quadrotor dapat dituliskan dalam representasi persamaan keadaan adalah sebagai berikut

$$\dot{x}_1 = \dot{\phi} = x_2$$

$$\dot{x}_2 = \ddot{\phi} = b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6$$

$$\dot{x}_3 = \dot{\theta} = x_4$$

$$\dot{x}_4 = \ddot{\theta} = b_2 U_3 + a_4 x_2 \Omega_r + a_3 x_2 x_6$$

$$\dot{x}_5 = \dot{\psi} = x_6$$

$$\dot{x}_6 = \ddot{\psi} = b_3 U_4 + a_5 x_2 x_4$$

$$\dot{x}_7 = \dot{z} = x_8$$

$$\dot{x}_8 = \ddot{z} = g - \frac{-U_1}{m}(\cos x_1 \cos x_3)$$

$$\dot{x}_9 = \dot{x} = x_{10}$$

$$\dot{x}_{10} = \ddot{x} = \frac{-U_1}{m}(\sin x_1 \sin x_5 + \cos x_1 \cos x_5 \sin x_3)$$

$$\dot{x}_{11} = \dot{y} = x_{12}$$

$$\dot{x}_{12} = \ddot{y} = \frac{-U_1}{m}(\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1)$$

$$f(X, U) = \begin{bmatrix} x_2 \\ b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6 \\ x_4 \\ b_2 U_3 + a_4 x_2 \Omega_r + a_3 x_2 x_6 \\ x_6 \\ b_3 U_4 + a_5 x_2 x_4 \\ x_8 \\ g - \frac{-U_1}{m} (\cos x_1 \cos x_3) \\ x_{10} \\ \frac{-U_1}{m} (\sin x_1 \sin x_5 + \cos x_1 \cos x_5 \sin x_3) \\ x_{12} \\ \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1) \end{bmatrix} \quad (2.39)$$

2.2.3 Sistem Kendali PID

Sistem kendali PID merupakan sistem kendali yang paling banyak digunakan pada industri [23]. Beberapa poin penting kesuksesan sistem kendali konvensional ini adalah :

- Struktur penyusun yang sederhana
- Hasil respon yang baik pada berbagai aplikasi
- Mudah dalam implementasi kedalam sistem tertanam (mikrokontroler)
- Banyak contoh kasus sehingga terdapat banyak referensi

Sistem kendali PID merupakan sebuah sistem pengendali yang terdiri dari sistem kendali proporsional, sistem kendali integral, dan sistem kendali derivatif. Kombinasi ketiga jenis sistem kendali ini bertujuan untuk saling melengkapi kekurangan yang ada pada masing-masing sistem kendali. Persamaan umum dari sistem kendali PID ditunjukkan pada persamaan 2.40.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (2.40)$$

dimana,

K_p = Koefisien parameter kendali proporsional

K_i = Koefisien parameter kendali integral

K_d = Koefisien parameter kendali derivatif

$e(t)$ = Selisih variabel terukur dengan set poin

2.2.3.1 Kendali Proporsional

Kendali proporsional adalah kendali yang menghasilkan keluaran yang nilainya proporsional dengan kesalahan yang terjadi. Secara sederhana dapat dikatakan bahwa keluaran kendali proporsional adalah hasil kali konstanta K_p dan besarnya kesalahan yang terjadi. Respon keluaran kendali proporsional dapat disesuaikan dengan mengatur nilai parameter K_p . Nilai K_p yang tinggi akan menghasilkan respon yang cepat menuju set poin yang diberikan, namun jika nilai K_p yang terlalu besar membuat sistem menjadi tidak stabil. Sebaliknya ketika nilai K_p terlalu kecil akan membuat respon lambat dan tidak responsif. Dalam prakteknya, K_p memegang peranan penting dalam sistem kendali karena kontribusinya paling besar dalam sebuah sistem kendali. Persamaan kendali proporsional ditunjukkan pada persamaan 2-41.

$$P_{out} = K_p e(t) \quad (2.41)$$

dimana,

P_{out} = Sinyal keluaran kendali proporsional

2.2.3.2 Kendali Integral

Sistem kendali integral memanfaatkan dari karakteristik dari sebuah persamaan integral. Keluaran kendali ini sangat dipengaruhi oleh jumlah dari total kesalahan yang terjadi dalam waktu tertentu. Hasil keluaran dari kendali ini dipengaruhi oleh parameter K_i yang merupakan pengali dari total jumlah kesalahan yang terjadi. Persamaan dari kendali ini ditunjukkan pada persamaan 2.42. Kendali ini membutuhkan waktu tertentu sehingga cenderung memperlambat respon.

$$I_{out} = K_i \int_0^t e(t) dt \quad (2.42)$$

dimana,

I_{out} = Sinyal keluaran kendali integral

Secara umum kendali integral berfungsi untuk menghilangkan kesalahan keadaan tunak yang tidak dapat dihilangkan oleh kendali proporsional. Dalam prakteknya kendali integral tidak bisa bekerja sendiri, memerlukan kendali lain yaitu kendali proporsional dan menjadi kendali PI (proporsional integral).

2.3.2.3 Kendali Derivatif

Kendali derivatif memiliki sifat seperti halnya suatu operasi derivatif. Selisih kesalahan saat itu dan kesalahan pada saat sebelumnya digunakan pada kendali derivative. Sehingga besarnya keluaran kendali ini ditentukan oleh selisih tersebut dan sebuah parameter K_d . Kendali derivatif digunakan untuk meningkatkan settling time dan kestabilan dari respon yang dihasilkan.

$$D_{out} = K_d \frac{de(t)}{dt} \quad (2.43)$$

dimana,

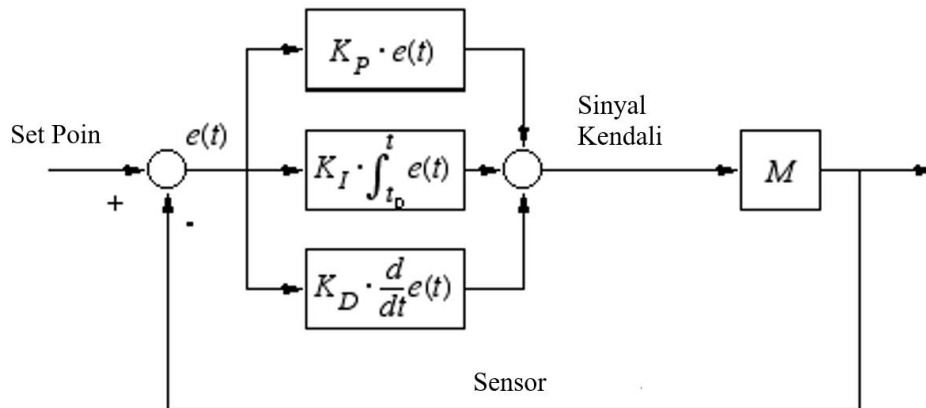
D_{out} = Sinyal keluaran kendali derivatif

2.3.2.4 Kendali PID (Proporsional-Integral-Derivatif)

Sistem kendali PID merupakan gabungan dari masing-masing kendali P, I dan D. Diagram blok PID ditunjukkan pada gambar 2.8. Ketiga sifat dari masing-masing kendali akan memberikan respon yang lebih baik dimana responnya lebih cepat, tanpa overshoot dan kesalahan keadaan tunak mendekati 0. Dalam blok sistem kendali PID terdapat 3 bagian penting, yaitu nilai set poin yang diinginkan, sinyal umpan balik yang digunakan untuk menghitung kesalahan atau *error*, dan sinyal kendali yang dihasilkan oleh kendali PID.

Hasil dari masing-masing sistem kendali P, I, dan D kemudian dijumlahkan sehingga hasil tersebut merupakan sinyal kendali yang menjadi acuan sebuah aktuator dalam bekerja. Keluaran dari obyek yang dikendalikan kemudian dibaca oleh sensor yang akan memberikan umpan balik menuju kontroler, namun

sebelumnya nilainya dibandingkan terlebih dahulu dengan sebuah nilai set poin untuk menghitung besar kesalahan yang terjadi. Kesalahan yang terjadi kemudian digunakan kembali sebagai acuan dari sistem kendali dalam menghitung sinyal kendali yang digunakan oleh aktuator sehingga hasil keluarannya sudah sesuai dengan set poin yang diberikan.



Gambar 2.8 Diagram Blok Sistem Kendali PID

Kecepatan respon yang terjadi, besarnya overshoot dan apakah terjadi kesalahan tunak ini dipengaruhi oleh parameter-parameter yang ada pada masing-masing sistem kendali. Karakteristik dari masing-masing perubahan parameter ditunjukkan pada tabel 2.2.

Tabel 2.2 Karakteristik Respon Perubahan Parameter Sistem Kendali PID

Kenaikan Parameter	Rise Time	Overshoot	Settling Time	Steady State Error
K_p	↓	↑	Perubahan Kecil	↓
K_i	↓	↑	↑	Menghilangkan
K_d	Perubahan Kecil	↓	↓	Perubahan Kecil

2.2.4 Sistem Kendali LQR

Dalam sistem kendali optimal salah satu tantangannya adalah merancang sistem kendali yang dapat memberikan performa terbaik sesuai dengan beberapa ukuran kinerja tertentu. Dalam hal ini, ukuran kinerja tertentu (disebut juga *optimal criterion*) akan menjadi energi dari sinyal kendali. Sistem Kendali *Linier Quadratic*

Regulator atau disingkat LQR merupakan sebuah teknik kendali modern yang menggunakan pendekatan persamaan keadaan atau *state space*. Dalam kendali LQR energi sinyal kendali diukur dengan *cost function* dimana mengandung faktor pembobotan yang didefinisikan oleh perancang sistem kendali. Diagram blok dari kendali LQR ditunjukkan pada gambar 2.6. Persamaan keadaan dari sebuah sistem adalah :

$$\dot{x} = Ax + Bu \quad (2.44)$$

dimana,

x = *state vector*

u = masukan

A = matriks sistem

B = matriks masukan

Untuk *cost function* didefinisikan sebagai :

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (2.45)$$

dimana matriks Q berukuran $n \times n$, dimana n sesuai dengan jumlah *state* yang dikendalikan dan matriks R berukuran $m \times m$, dimana m adalah jumlah dari masukan. Matriks Q harus berupa matriks definit positif atau matriks simetris semi definit positif. Matriks R harus berupa matriks simetris dan definit positif. Salah satu metode praktisnya adalah matriks Q dan R adalah matriks diagonal. Nilai dari elemen matriks Q dan R terkait dengan pengaruhnya pada *cost function* J . Aturan kendali umpan balik yang meminimalkan nilai dari *cost function* adalah

$$u = -Kx \quad (2.46)$$

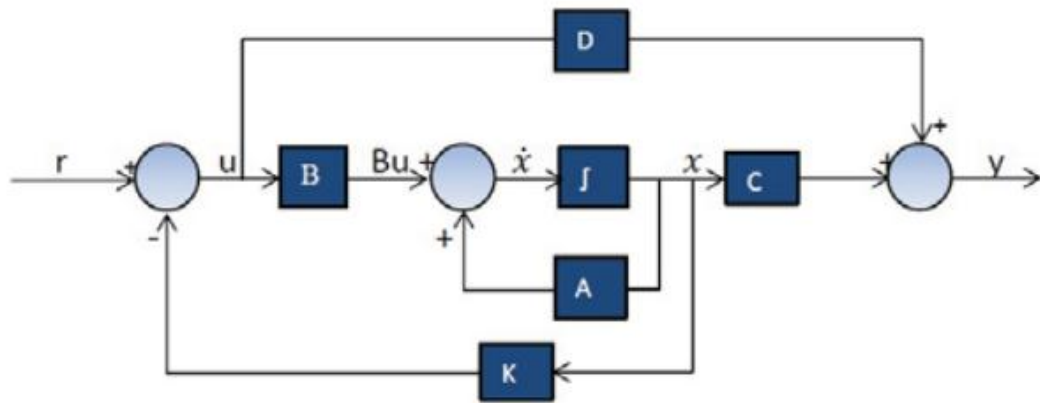
dimana u merupakan vektor dari aksi kendali dan matriks *gain* K dari LQR didapatkan dari

$$K = R^{-1} B^T P \quad (2.47)$$

Untuk mendapatkan P dengan menyelesaikan persamaan aljabar riccati dan dalam matlab kita diberikan sebuah fungsi *toolbox* agar memudahkan dalam mencari nilai

matriks P dengan perintah *care*. Persamaan umum dalam mencari nilai P adalah ditunjukkan oleh persamaan 2.48.

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (2.48)$$



Gambar 2.9 Diagram Blok Sistem Kendali LQR

Dalam persamaan 2.48 terdapat variable Q dan R yang nilainya kita tentukan sendiri. Penentuan nilai dari matriks Q dan R ini adalah krusial dan sangat berpengaruh kepada respon dari sebuah sistem tersebut. Sistem akan menjadi lebih tahan terhadap gangguan dan settling time akan menjadi lebih cepat jika Q bernilai besar. Pemilihan nilai matriks Q dan R akan mempengaruhi :

- Semakin besar nilai matriks Q akan memperbesar nilai gain K dimana meningkatkan kecepatan respon dari sistem
- Semakin besar nilai matriks R akan menurunkan nilai dari gain K dimana akan memperlambat kondisi *steady state* dan berkaitan dengan penggunaan energi

Tidak ada solusi unik bagaimana menentukan nilai pada matriks Q dan R , biasanya nilainya didapatkan dari sejumlah iterasi dari proses simulasi. Dalam prakteknya ada metode yang dapat digunakan untuk menentukan matriks Q dan R yaitu adalah dengan menggunakan metode *Bryson's rule* [26]. Namun metode ini juga tidak menjamin akan memberikan hasil yang diinginkan. *Bryson's Rule* memberikan aturan dalam menentukan matriks Q dan R dapat dimulai dengan

$$\begin{aligned}
Q_{ii} &= \frac{1}{(x_i)_{max}^2}, & i \in \{1, 2, \dots, n\} \\
R_{jj} &= \frac{1}{(u_j)_{max}^2}, & j \in \{1, 2, \dots, m\}
\end{aligned}
\tag{2. 49}$$

dimana $(x_i)_{max}$ adalah nilai maksimal dari x_1 yang diperbolehkan dan $(u_i)_{max}$ adalah nilai u_j maksimal yang diperbolehkan. Perlu menjadi catatan bahwa nilai yang ditentukan menggunakan *bryson's rule* menghasilkan respon yang baik namun sebenarnya ini adalah nilai awal dalam memulai iterasi sehingga mendapatkan respon sesuai yang diinginkan.

BAB III

METODOLOGI

3.1 Perlengkapan Penelitian

Pada penelitian ini perancangan sistem kendali dan sistem model dari obyek yang telah didapatkan persamaan akan disimulasikan pada perangkat lunak matlab terutama pada Simulink. Berikut ini adalah pembahasan terkait alat-alat dan bahan yang digunakan pada penelitian. Untuk menjalankan simulasi dibutuhkan perangkat pendukung yaitu sebuah laptop dan program yang digunakan adalah MATLAB R2013b. Selain kedua hal tersebut *paper*, jurnal, laporan tesis, dan buku pustaka yang terkait dengan topik penelitian didapatkan dari internet guna mendukung penelitian ini. Hal ini perlu dikarenakan agar penulis dapat memahami terkait dengan masalah yang ada pada penelitian ini.

3.2 Alur Penelitian

Secara umum, alur penelitian yang digunakan pada penelitian ini adalah seperti ditunjukkan pada Gambar 3.1.

3.2.1 Studi Kepustakaan

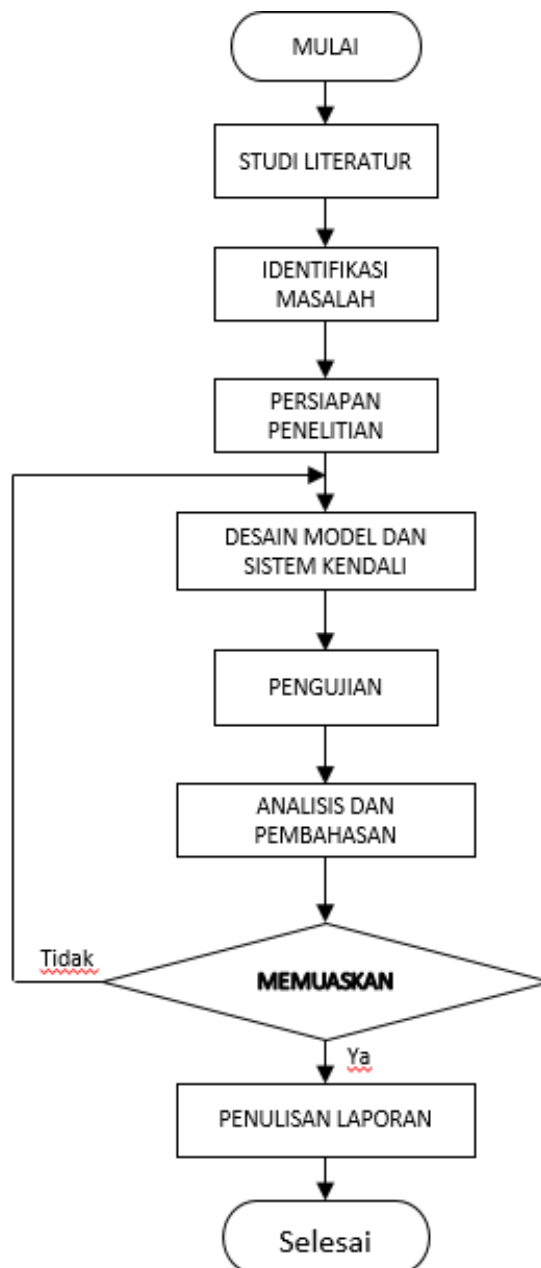
Pada bagian studi literatur, yang dilakukan adalah mencari referensi-referensi baik itu jurnal, paper, dan laporan tesis maupun buku-buku yang terkait dengan topik penelitian. Hal ini dilakukan agar penulis tahu sampai sejauh mana perkembangan penelitian di bidang ini serta untuk mempelajari hal yang menjadi permasalahan dalam topik penelitian.

3.2.2 Identifikasi Masalah

Identifikasi masalah dilakukan setelah melakukan studi kepustakaan. Hal ini dilakukan untuk mengetahui pokok-pokok permasalahan yang menjadi topik penelitian sehingga dapat menentukan tujuan dan manfaat penelitian.

3.2.3 Persiapan Penelitian

Sebelum memulai penelitian itu sendiri, persiapan alat dan bahan diperlukan agar penelitian dapat berjalan dengan lancar. Alat dan bahan yang digunakan pada penelitian dapat dilihat pada sub-bab 3.1.



Gambar 3.1 Diagram Alur Penelitian

3.2.4 Desain Model dan Sistem Kendali

Pada proses ini, perancangan model dari quadrotor yang didasarkan dari kinematika dan dinamika dari quadrotor dan sistem kendali PID dengan penalaan berbasis LQR dilakukan. Sistem kendali PID dengan penalaan LQR yang dirancang bertujuan untuk menstabilkan sudut roll, pitch, dan yaw dan ketinggian yang ada pada quadrotor agar dapat terbang pada mode hover dengan baik dan sesuai nilai referensi yang diberikan. Selain itu sistem kendali lain, yaitu PID dan LQR juga dirancang yang digunakan untuk membandingkan dengan sistem kendali yang diusulkan.

3.2.5 Pengujian

Setelah melakukan perancangan model dan sistem kendali, kemudian langkah selanjutnya adalah simulasi. Hasil perancangan pada langkah sebelumnya dituliskan pada perangkat lunak MATLAB. Pengujian dilakukan terhadap 4 sistem kendali yang dirancang yaitu kendali sudut roll, pitch, yaw dan ketinggian. Pengujian dilakukan bertahap mulai dari simulasi model untuk memvalidasi dari respon model kemudian dilanjutkan dengan simulasi menggunakan kendali yang telah dirancang. Selain itu juga dilakukan pengujian dengan gangguan dari luar sistem model.

3.2.6 Analisis Hasil dan Pembahasan

Hasil dari simulasi adalah grafik-grafik yang merupakan representatif dari respon sudut roll, pitch, yaw dan ketinggian. Dari grafik-grafik ini kemudian dilakukan analisis sesuai dengan langkah analisis sehingga didapatkan kesimpulan yang tepat.

3.2.7 Penulisan Laporan

Penulisan laporan merupakan langkah terakhir yang ada pada penelitian ini. Selain menjadi syarat dalam kelulusan, tujuan penulisan laporan ini adalah agar adanya publikasi terkait penelitian yang dilakukan agar orang lain dapat meneruskan atau mengembangkan penelitian dalam bidang quadrotor terutama

pada bagian sistem kendali.

3.3 Perancangan Penelitian

Pada penelitian ini akan diimplementasikan sebuah algoritma sistem kendali PID yang parameternya didapatkan menggunakan algoritma LQR pada sebuah model quadrotor. Untuk mendapatkan hasil yang baik tentunya dalam perancangan perlu memperhatikan hal-hal berikut, perancangan model quadrotor, sistem kendali PID dengan metode penalaan secara iterasi, dan algoritma LQR yang digunakan untuk mendapatkan parameter PID secara otomatis. Berikut ini adalah penjelasan dari masing-masing hal tersebut. Hasil dari sistem kendali yang diusulkan kemudian dibandingkan dengan sistem kendali PID dan LQR agar dapat diketahui kinerjanya.

3.3.1 Model Quadrotor

Untuk menerapkan sistem kendali LQR pada sebuah sistem quadrotor, sistem yang tidak linier kemudian dilinierisasi pada keadaan *equilibrium* atau sekitar keadaan melayang sehingga kecepatan sudut $[p \ q \ r]^T$ sesuai dengan turunan dari kemiringan sudut Euler ($[p \ q \ r]^T = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$). Keadaan melayang disini adalah dianggap sudut dari RPY adalah 0 sehingga persamaan model (2.39) dapat disederhanakan. Secara singkat, sistem dinamis yang ada pada model quadrotor yang telah di liniersasi pada keadaan melayang adalah

$$\begin{aligned}\ddot{z} &= g - U_1/m \\ \ddot{\phi} &= U_2/I_{xx} \\ \ddot{\theta} &= U_3/I_{yy} \\ \ddot{\psi} &= U_4/I_{zz}\end{aligned}\tag{3. 1}$$

Kita dapat mengabaikan posisi x dan y karena dalam penelitian ini yang menjadi perhatian kita adalah pada ke-4 variabel diatas. Persamaan dinamika 3.1 pada keadaan linier ini yang nantinya akan digunakan pada langkah penghitungan nilai matriks gain K pada bagian sistem kendali LQR. Persamaan 3.1 dituliskan

dalam persamaan keadaan yang baru menjadi

$$\begin{aligned}\dot{\hat{x}}_1 &= \dot{\phi} = \hat{x}_2 \\ \dot{\hat{x}}_2 &= \ddot{\phi} = U_2/I_{xx} \\ \dot{\hat{x}}_3 &= \dot{\theta} = \hat{x}_4 \\ \dot{\hat{x}}_4 &= \ddot{\theta} = U_2/I_{xx} \\ \dot{\hat{x}}_5 &= \dot{\psi} = \hat{x}_6 \\ \dot{\hat{x}}_6 &= \ddot{\psi} = U_4/I_{zz} \\ \dot{\hat{x}}_7 &= \dot{z} = \hat{x}_8 \\ \dot{\hat{x}}_8 &= \ddot{z} = g - U_1/m\end{aligned}$$

$$f(\hat{X}, U) = \begin{bmatrix} \hat{x}_2 \\ U_2/I_{xx} \\ \hat{x}_4 \\ U_2/I_{xx} \\ \hat{x}_6 \\ U_4/I_{zz} \\ \hat{x}_8 \\ g - U_1/m \end{bmatrix} \quad (3.2)$$

3.3.2 Kendali PID

Meskipun kendali PID termasuk sistem kendali yang sudah terbilang lama sistem kendali PID dikenal cukup handal untuk berbagai sistem. Kendali PID adalah jenis kendali SISO (*Single Input Single Output*) sehingga aplikasinya langsung dilakukan pada masing-masing obyek kendali yaitu roll, pitch, yaw dan ketinggian. Penelitian yang sudah dilakukan menggunakan metode ini adalah [] namun hanya bekerja pada ketinggiannya saja. Telah dijelaskan pada bab sebelumnya kendali PID terdiri atas 3 komponen utama yaitu proporsional, integral dan derivative.

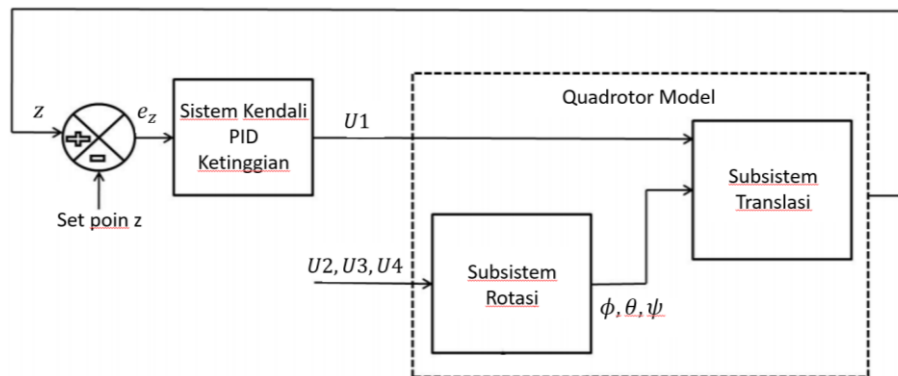
Pada kasus ini nilai dari parameter PID yang digunakan adalah dengan metode struktural atau iterasi yaitu nilai parameter PID yang didapatkan

berdasarkan aturan :

1. Memberikan nilai $K_p = 1$ (atau bernilai kecil) dan $K_i = K_d = 0$
2. Menaikkan nilai K_p sampai terjadi *overshot* sedikit diatas set poin yang diberikan
3. Memberikan nilai K_p sekitar 80-90% dari nilai K_p yang didapatkan, kemudian naikan nilai K_d agar redaman terjadi pada respon
4. Apabila masih terjadi kesalahan keadaan tunak atur nilai K_i
5. Langkah terakhir adalah sedikit perubahan pada semua nilai K_p , K_i , dan K_d agar respon sesuai dengan keinginan

3.3.2.1 Sistem Kendali PID Ketinggian

Sistem kendali PID yang pertama dirancang adalah sistem kendali untuk *altitude* atau ketinggian. Diagram blok yang digunakan pada sistem ketinggian ini adalah ditunjukkan pada gambar 3.2.



Gambar 3.2 Diagram Blok Sistem Kendali PID Ketinggian

Pada gambar 3.2 z merupakan ketinggian, e_z adalah selisih antara ketinggian aktual dengan set poin ketinggian dan sistem kendali PID ketinggian menghasilkan sinyal kendali U_1 yang berpengaruh pada subsistem translasi pada model quadrotor. Berikut adalah kendali PID yang di implementasikan untuk mengendalikan ketinggian quadrotor

$$U_1 = K_{pz}(Z_{sp} - Z_{act}) + K_{iz} \int_0^t (Z_{sp} - Z_{act}) dt + K_{dz}(\dot{Z}_{sp} - \dot{Z}_{act}) \quad (3.3)$$

dimana,

U_1 = Sinyal kendali untuk Kendali Ketinggian

K_{pz} = Parameter Proporsional Kendali Ketinggian

K_{iz} = Parameter Integral Kendali Ketinggian

K_{dz} = Parameter Derivatif Kendali Ketinggian

Z_{sp} = Ketinggian yang diinginkan (set poin)

Z_{act} = Ketinggian aktual

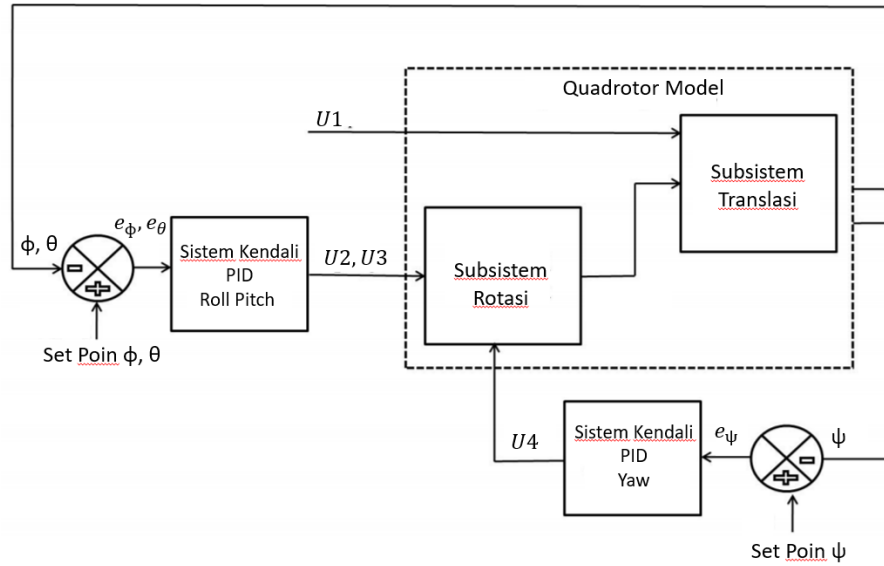
\dot{Z}_{sp} = Kecepatan vertikal yang diinginkan (set poin)

\dot{Z}_{act} = Kecepatan vertikal actual

3.3.2.2 Sistem Kendali PID RPY

Sistem kendali PID pada sudut roll, pitch, dan yaw terdiri atas 3 sistem kendali independen yang keluarannya direpresentasikan dengan U_2, U_3, U_4 . Karena sifatnya independen sehingga nilai parameter pada masing-masing sistem kendali juga bersifat independen dan dapat diatur masing-masing. Sinyal kendali yang dihasilkan pada sistem kendali RPY berpengaruh pada subsistem rotasi model quadrotor. Terdapat 3 set poin pada masing-masing blok sistem kendali seperti terlihat pada gambar 3.3

Terlihat pula pada gambar 3.3 terdapat e_ϕ, e_θ, e_ψ yang merupakan notasi dari kesalahan pada masing-masing obyek pengukuran. Kesalahan disini maksudnya adalah perbedaan nilai aktual yang terukur dengan set poin yang diberikan. Dari kesalahan yang terukur tersebut kemudian sistem kendali PID akan mengkalkulasi sinyal kendali agar sistem dapat bergerak menuju set poin yang diberikan.



Gambar 3.3 Diagram Blok Sistem Kendali PID RPY

Berikut adalah persamaan kendali PID yang digunakan untuk mengendalikan sudut RPY pada quadrotor

$$\begin{aligned}
 U_2 &= K_{p\phi}(\phi_{sp} - \phi_{act}) + K_{i\phi} \int_0^t (\phi_{sp} - \phi_{act}) dt + K_{d\phi}(\dot{\phi}_{sp} - \dot{\phi}_{act}) \\
 U_3 &= K_{p\theta}(\theta_{sp} - \theta_{act}) + K_{i\theta} \int_0^t (\theta_{sp} - \theta_{act}) dt + K_{d\theta}(\dot{\theta}_{sp} - \dot{\theta}_{act}) \\
 U_4 &= K_{p\psi}(\psi_{sp} - \psi_{act}) + K_{i\psi} \int_0^t (\psi_{sp} - \psi_{act}) dt + K_{d\psi}(\dot{\psi}_{sp} - \dot{\psi}_{act})
 \end{aligned} \tag{3.4}$$

dimana,

U_2, U_3, U_4 = Sinyal kendali untuk Kendali sudut RPY

$K_{p\phi}, K_{p\theta}, K_{p\psi}$ = Parameter Proporsional Kendali sudut RPY

$K_{i\phi}, K_{i\theta}, K_{i\psi}$ = Parameter Integral Kendali sudut RPY

$K_{d\phi}, K_{d\theta}, K_{d\psi}$ = Parameter Derivatif Kendali sudut RPY

$\phi_{sp}, \theta_{sp}, \psi_{sp}$ = sudut RPY yang diinginkan (set poin)

$\phi_{act}, \theta_{act}, \psi_{act}$ = sudut RPY aktual

$\dot{\phi}_{sp}, \dot{\theta}_{sp}, \dot{\psi}_{sp}$ = Kecepatan sudut RPY yang diinginkan (set poin)

$\dot{\phi}_{act}, \dot{\theta}_{act}, \dot{\psi}_{act}$ = Kecepatan sudut RPY aktual

3.3.3 Kendali LQR

Telah dijelaskan bahwa LQR merupakan salah satu metode kendali optimal yang masih berkembang dan aplikasinya mulai banyak digunakan. Desain dari kendali LQR berdasarkan pemilihan *gain* balik K yang menyebabkan fungsi *cost* atau performa index J menjadi minimal. Keuntungan menggunakan sistem kendali LQR ini adalah sistem menjadi stabil dan *robust* kecuali pada kondisi dimana sistem *uncontrollable*. Quadrotor adalah sistem dimana dengan jumlah input dan output lebih dari satu sehingga dinamakan *multiple input multiple output* (MIMO) [24]. Pada perancangan sistem kendali LQR kita harus menentukan persamaan keadaan

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned}\tag{3.5}$$

dengan performa indeks J adalah

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt\tag{3.6}$$

Matriks Q dan R disebut dengan matriks bobot. Dimana Q adalah matriks faktor pembobot *state* (semi-definit positif) dan R adalah matriks faktor pembobot variabel kontrol (definit positif). Pada kasus ini persamaan dari model quadrotor terdapat 8 *state* yaitu 6 *state* untuk rotasi sudut roll (ϕ) dan kecepatan angular roll ($\dot{\phi}$), sudut pitch (θ) dan kecepatan angularnya ($\dot{\theta}$), sudut yaw (ψ) dan kecepatan angular yaw ($\dot{\psi}$), kemudian 2 *state* untuk translasi yaitu posisi ketinggian (z) dan kecepatannya (\dot{z}) yang semuanya tersebut dinotasikan dalam \hat{X} .

$$\hat{X} = [z \quad \phi \quad \theta \quad \psi \quad \dot{z} \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi}]^T\tag{3.7}$$

Masukan dari persamaan keadaan tersebut adalah U dimana terdiri dari 4 *state* yaitu U_1, U_2, U_3 , dan U_4 sesuai dengan persamaan (3-8)

$$U = [U_1 \quad U_2 \quad U_3 \quad U_4]^T \quad (3.8)$$

Sehingga apabila kita tuliskan lengkap persamaan keadaannya menjadi

$$\hat{\dot{X}} = A\hat{X} + BU \quad (3.9)$$

$$\begin{bmatrix} \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = A \begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + B \begin{bmatrix} U1 \\ U2 \\ U3 \\ U4 \end{bmatrix} \quad (3.10)$$

dimana nilai masing-masing dari nilai matriks A dan B adalah

$$A = \begin{bmatrix} O_{(4x4)} & I_{(4x4)} \\ O_{(4x4)} & O_{(4x4)} \end{bmatrix}$$

$$B = \begin{bmatrix} O_{(4x4)} \\ -1/m & 0 & 0 & 0 \\ 0 & 1/I_{xx} & 0 & 0 \\ 0 & 0 & 1/I_{yy} & 0 \\ 0 & 0 & 0 & 1/I_{zz} \end{bmatrix} \quad (3.11)$$

sehingga apabila dituliskan menjadi persamaan keadaan lengkap adalah

$$\begin{bmatrix} \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} O_{(4x4)} & O_{(4x4)} \\ O_{(4x4)} & I_{(4x4)} \end{bmatrix} \begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} O_{(4x4)} \\ -1/m & 0 & 0 & 0 \\ 0 & 1/I_{xx} & 0 & 0 \\ 0 & 0 & 1/I_{yy} & 0 \\ 0 & 0 & 0 & 1/I_{zz} \end{bmatrix} \begin{bmatrix} U1 \\ U2 \\ U3 \\ U4 \end{bmatrix} \quad (3.12)$$

Sedangkan untuk *state* keluaran y dari model quadrotor yang kita miliki adalah

$$y = Cx \quad (3.13)$$

$$\begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = C \begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + D \begin{bmatrix} U1 \\ U2 \\ U3 \\ U4 \end{bmatrix} \quad (3.14)$$

dan nilai dari matriks C adalah sebuah matriks identitas dengan jumlah baris dan kolom 8

$$C = [I_{(8 \times 8)}] \quad (3.15)$$

Nilai dari matriks D adalah nol semua jadi tidak perlu dituliskan lagi. Sehingga apabila dituliskan pada persamaan keluaran lengkap menjadi

$$\begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = [I_{(8 \times 8)}] \begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.16)$$

Pemilihan nilai Q dan R pada sebuah kendali LQR adalah hal yang krusial karena sangat mempengaruhi respon yang terjadi. Sedikit perubahan nilai Q dan R akan memberikan hasil yang berbeda. Pada penelitian ini untuk menentukan awal dari matrik bobot Q dan R digunakan aturan *Bryson's* yang aturannya sudah dituliskan pada bab sebelumnya. Penentuan nilai Q dan R akan dibahas pada bab selanjutnya.

3.3.4 Kendali LQR-PD

Sub-bab ini akan membahas terkait metode yang diusulkan pada penelitian ini yaitu sistem kendali PID dengan tuning berdasarkan metode LQR. Ada 3 parameter dari sistem kendali PID yang didapatkan dari metode LQR. Parameter tersebut adalah K_p , K_i , dan K_d . Telah kita ketahui bersama bahwa aturan kendali

umpan balik adalah seperti pada persamaan 3.17

$$u = -Kx \quad (3.17)$$

dimana K merupakan matriks *gain*. Dari persamaan 3.17 tersebut dapat kita turunkan agar sesuai dengan keluaran pada sistem kendali PID sehingga

$$u(t) = -Kx(t)$$

$$u(t) = -[-K_i \quad -K_p \quad -K_d] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

$$u(t) = K_i \int_0^t e(t)dt + K_p e(t) + K_d \frac{de(t)}{dt} \quad (3.18)$$

Pada bagian state x terdapat 3 komponen yaitu $e(t)$, $\int_0^t e(t)$, dan $\frac{de(t)}{dt}$. Dari dasar tersebut kita dapat membuat sebuah matriks state baru ϑ dimana nilainya adalah

$$\vartheta = [\vartheta_1 \quad \vartheta_2 \quad \vartheta_3]^T \quad (3.19)$$

dimana nilai dari masing-masing komponen ϑ_1 , ϑ_2 , dan ϑ_3 adalah komponen integral *error*, *error*, dan derivative *error* secara berurutan seperti ditunjukkan pada persamaan 3.20.

$$\begin{aligned} \vartheta_1 &= \left[\int_0^t z(s)ds \quad \int_0^t \phi(s)ds \quad \int_0^t \theta(s)ds \quad \int_0^t \psi(s)ds \right]^T \\ \vartheta_2 &= [z \quad \phi \quad \theta \quad \psi]^T \\ \vartheta_3 &= [\dot{z} \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi}]^T \end{aligned} \quad (3.20)$$

Dituliskan dalam keadaan lengkap menjadi Vektor dari *state* ϑ menjadi

$$\vartheta = \left[\int_0^t z(s)ds \quad \int_0^t \phi(s)ds \quad \int_0^t \theta(s)ds \quad \int_0^t \psi(s)ds \quad z \quad \phi \quad \theta \quad \psi \quad \dot{z} \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi} \right]^T \quad (3.21)$$

Kita telah mendapatkan sebuah matriks *state* baru ϑ sehingga persamaan keadaan

yang baru adalah

$$\dot{\vartheta} = \hat{A}\vartheta + \hat{B}U \quad (3.22)$$

Untuk mendapatkan gain matriks K yang ada pada state baru ini kita perlu mencari nilai nilai matriks \hat{A} dan \hat{B} terlebih dahulu maka perlu kita jabarkan terlebih dahulu persamaan 3.22 menjadi

$$\begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \hat{A} \begin{bmatrix} \int_0^t z(s)ds \\ \int_0^t \phi(s)ds \\ \int_0^t \theta(s)ds \\ \int_0^t \psi(s)ds \\ z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \hat{B} \begin{bmatrix} U1 \\ U2 \\ U3 \\ U4 \end{bmatrix} \quad (3.23)$$

dimana nilai matriks \hat{A} dan \hat{B} adalah

$$\hat{A} = \left[\begin{array}{c|c} I_{(4 \times 4)} & O_{(4 \times 4)} \\ O_{(12 \times 4)} & \begin{array}{c} O_{(4 \times 4)} \quad I_{(4 \times 4)} \\ O_{(8 \times 4)} \end{array} \end{array} \right]$$

$$\hat{B} = \begin{bmatrix} \begin{array}{c} O_{(8 \times 4)} \\ -1/m \quad 0 \quad 0 \quad 0 \\ 0 \quad 1/I_{xx} \quad 0 \quad 0 \\ 0 \quad 0 \quad 1/I_{yy} \quad 0 \\ 0 \quad 0 \quad 0 \quad 1/I_{zz} \end{array} \end{bmatrix} \quad (3.24)$$

Pada *state* keluaran nilai matriks C adalah matriks identitas dan matriks D adalah matriks nol. Sama pada langkah sebelumnya yaitu menentukan nilai Q dan R yang merupakan matriks bobot yang mempengaruhi nilai dari *gain* matriks \hat{K} . Setelah

mendapatkan matriks gain \hat{K} baru kita dapat menentukan parameter-parameter dari \hat{K}_p , \hat{K}_i , dan \hat{K}_d sesuai dengan persamaan 3.25.

$$-\hat{K} = -[\hat{K}_i \quad \hat{K}_p \quad \hat{K}_d] \quad (3.25)$$

Setelah mendapatkan parameter tersebut langkah selanjutnya adalah memasukkan nilai-nilai tersebut pada sistem kendali PID yang telah kita rancang sebelumnya.

BAB IV

HASIL DAN PEMBAHASAN

Dalam penelitian ini, dilakukan pengujian meliputi pengujian model pada keadaan kalang terbuka, dan menggunakan sistem kendali dengan nilai set poin tertentu dan diakhir dilakukan pengujian terkait *robustness* dengan cara mengubah parameter yang ada pada quadrotor.

4.1 Pengujian Model (Kalang terbuka)

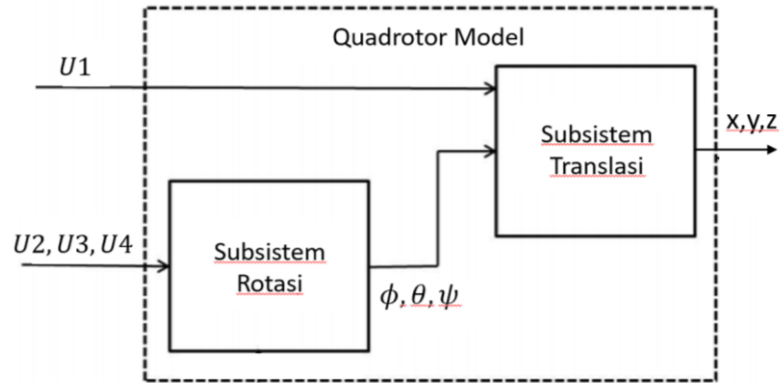
Pada pembahasan ini akan dilakukan pengujian terhadap model simulink yang telah dibuat. Ada 4 pengujian yang dilakukan yaitu terkait dengan ketinggian, sudut roll, sudut pitch, dan sudut yaw daripada sebuah model quadrotor. Parameter-parameter yang digunakan pada model quadrotor pada simulasi adalah sesuai dengan tabel 4.1 [27].

Tabel 4.1 Parameter Model Quadrotor

g	9.81 m/s ²
b	3.13 x 10 ⁻⁵
d	7.5 x 10 ⁻⁷
m	0.65 kg
I_{xx}	7.5 x 10 ⁻³
I_{yy}	7.5 x 10 ⁻³
I_{zz}	1.3 x 10 ⁻²
Jr	6 x 10 ⁻⁵
L	0.23 m

4.1.1 Pengujian Ketinggian

Dalam pengujian ketinggian ada 3 tahapan pengujian yang dilakukan yaitu keadaan naik, turun dan keadaan melayang.



Gambar 4.1 Blok Model Quadrotor

Pada bab 3 telah dijelaskan mengenai model matematika dari sebuah obyek quadrotor. Terdapat 4 buah masukan yang berupa kecepatan putar dari masing-masing motor (ω_i). Telah kita ketahui total gaya angkat yang harus dihasilkan dari ke-4 motor tersebut agar quadrotor dapat terbang melayang adalah sama dengan total beratnya. Persamaan 4.1 merupakan total berat yang ada pada quadrotor

$$T = mg \quad (4.1)$$

Dimana m adalah massa dari quadrotor dan g adalah gravitasi yang merupakan konstanta. Sehingga untuk mendapatkan kecepatan putar motor pada kondisi melayang adalah

$$T = mg = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \quad (4.2)$$

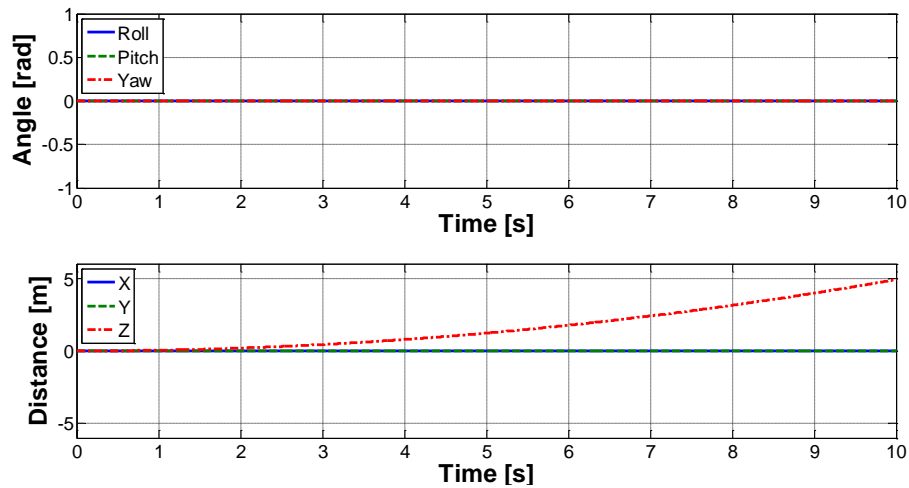
Pada kondisi melayang $\omega_1 = \omega_2 = \omega_3 = \omega_4$ sehingga dapat diperoleh kecepatan putar pada kondisi melayang ω_H adalah

$$\omega_H = \sqrt{\frac{mg}{4b}} \quad (4.3)$$

4.1.1.1 Pengujian keadaan naik

Pada pengujian ini, pengaturan dilakukan pada kecepatan masing-masing motor dengan keadaan motor berputar melebihi kecepatan putar pada kondisi melayang ($\omega_i > \omega_H$). Untuk memudahkan dalam pengujian maka diberikan sebuah faktor pengali (dalam simulator diberikan notasi rasio i , $i = 1, 2, 3, 4$). Apabila rasio bernilai 1 maka kondisinya kecepatan motor sama dengan kecepatan putar pada

kondisi melayang ($\omega_i = \omega_H$). Sehingga pada pengujian ini karena yang diinginkan adalah $\omega_i > \omega_H$ maka rasio yang diberikan nilainya lebih dari 1. Pada kasus ini respon yang terjadi pada sudut RPY dan posisi translasi adalah seperti terlihat pada gambar 4.2.



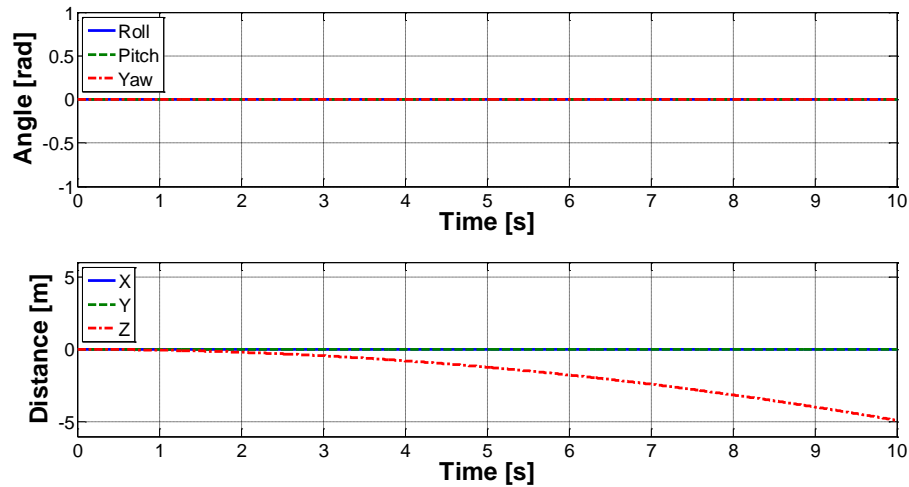
Gambar 4.2 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan naik

Dalam gambar 4.2 terlihat tidak ada perubahan kecuali pada ketinggian (z). Hal ini menunjukkan bahwa quadrotor mengalami kenaikan ketinggian dengan posisi sudut RPY adalah nol dan tidak terjadi pegeseran pada posisi x dan y . Quadrotor dapat terbang naik karena gaya angkat yang dihasilkan oleh putaran baling-baling melebihi berat dari quadrotor itu sendiri.

4.1.1.2 Pengujian keadaan turun

Pengujian ini dilakukan dengan memberikan nilai rasio kurang dari 1 sehingga menyebabkan nilai kecepatan putar tiap motor nilainya kurang dari kecepatan putar saat melayang ($\omega_i < \omega_H$). Pada kasus respon yang terjadi adalah seperti pada gambar 4.3. Berlawanan dengan pengujian sebelumnya, quadrotor mengalami kenaikan, pada pengujian ini quadrotor mengalami penurunan ketinggian. Tidak ada perubahan pada posisi x dan y karena sudut RPY yang terjadi adalah nol. Penurunan dari quadrotor terjadi karena nilai putaran motor kurang dari putaran motor saat melayang yang menyebabkan gaya angkat yang dihasilkan dari putaran baling-baling kurang dari berat quadrotor. Dalam aktualnya tidak ada nilai

ketinggian negatif sehingga seharusnya nilai paling rendah untuk ketinggian adalah 0 meter.

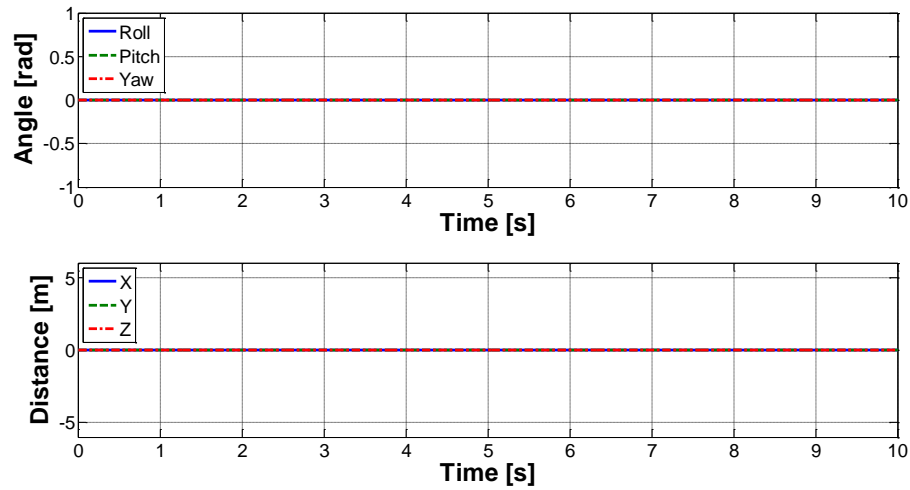


Gambar 4.3 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan turun

4.1.1.3 Pengujian keadaan melayang

Pengujian terakhir pada bagian ini adalah pengujian keadaan melayang. Kita telah mengetahui nilai dari putaran minimum agar quadrotor dapat terbang melayang adalah sama dengan beratnya sesuai dengan persamaan (4.1) sehingga rasio yang diberikan adalah 1 sehingga ($\omega_i = \omega_H$).

Sama dengan 2 pengujian sebelumnya tidak ada perubahan pada sudut RPY, tidak terjadi pula perubahan pada posisi translasi baik posisi x , y dan z . Yang menjadi perhatian kita pada pengujian ini adalah pada ketinggian quadrotor yang tetap sesuai dengan keadaan awal yaitu 0. Hal ini dapat terjadi karena berat gaya angkat yang terjadi besarnya sama dengan berat quadrotor sehingga saling menghilangkan dan tidak terjadi perubahan.

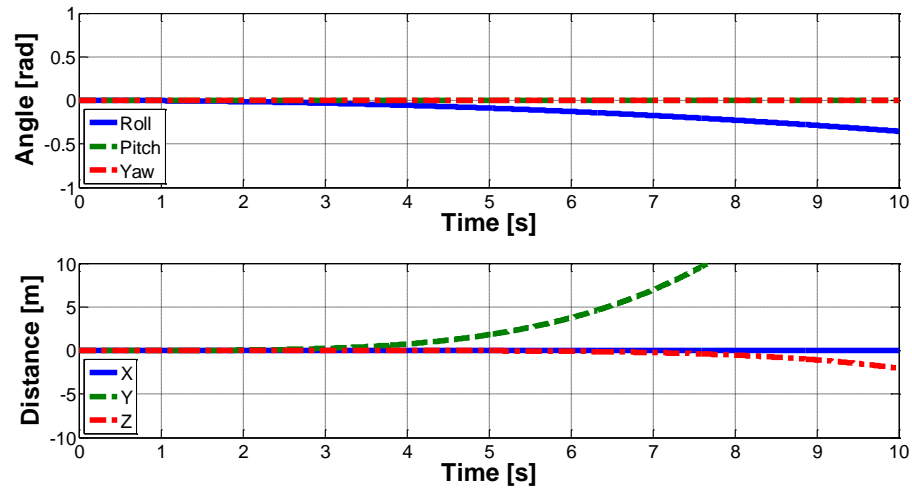


Gambar 4.4 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada keadaan melayang

4.1.2 Pengujian Roll

Pada bab 2 telah dijelaskan tentang dasar-dasar gerakan yang ada pada quadrotor dan salah satunya adalah gerakan roll. Untuk menghasilkan gerakan roll, motor kanan (ω_2) dan kiri (ω_4) pada quadrotor diatur kecepatannya (naik / turun). Pengujian dilakukan dengan kombinasi nilai $\omega_2 > \omega_H$ & $\omega_4 < \omega_H$ untuk melakukan gerakan roll ke kiri atau $\omega_2 < \omega_H$ & $\omega_4 > \omega_H$ untuk gerakan roll ke kanan. Nilai selisih pada masing-masing motor biasanya sama agar gerakan yang dihasilkan tetap pada sumbu x . Gambar 4.5 adalah respon yang terjadi pada pengujian ini.

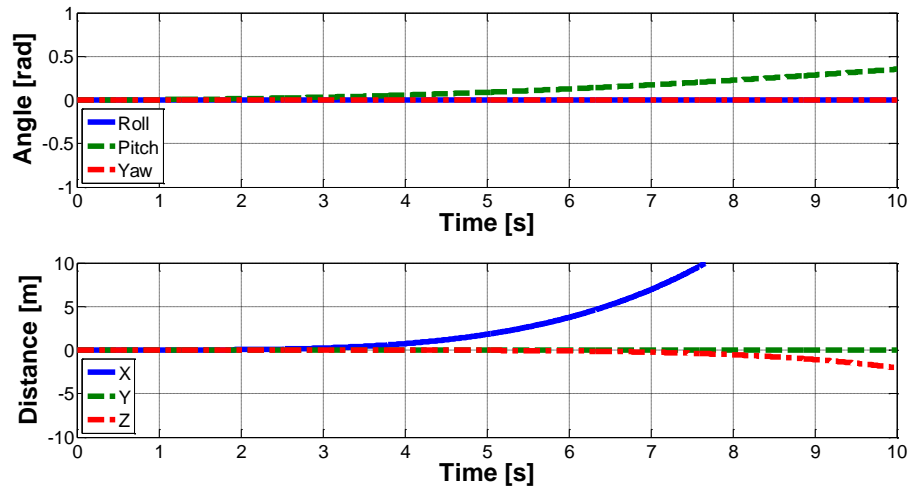
Dapat dilihat pada respon sudut roll, terjadi perubahan yang awalnya 0 rad nilainya menuju minus, ini menunjukkan quadrotor sedang melakukan gerakan roll ke kiri. Efek gerakan roll ini menyebabkan quadrotor bergerak pada sumbu y , hal ini sudah sesuai dengan kaidah dasar dari gerakan-gerakan dasar quadrotor. Selain itu ketinggian juga terjadi penurunan, hal inilah yang disebut *coupling*.



Gambar 4.5 Respon pengujian sudut RPY (atas) dan posisi translasi (bawah) pada gerakan roll

4.1.3 Pengujian Pitch

Pada pengujian ini serupa dengan pengujian sebelumnya, hanya saja motor yang berpengaruh pada gerakan pitch sesuai yang sudah dijelaskan sebelumnya adalah motor depan (ω_1) dan motor belakang (ω_3). Apabila salah satu kecepatan putarnya lebih besar dibanding dengan satunya maka akan terjadi gerakan pitch. Kombinasi yang digunakan adalah $\omega_1 > \omega_H$ & $\omega_3 < \omega_H$ untuk melakukan gerakan pitch ke atas atau $\omega_1 < \omega_H$ & $\omega_3 > \omega_H$ untuk gerakan pitch ke bawah. Gerakan ini digunakan untuk bergerak maju karena quadrotor dapat bergerak maju ketika ada sudut pitch yang terjadi. Hal ini sesuai dengan respon yang terjadi sesuai dengan gambar 4.6. Sama halnya pada pengujian roll, pada pengujian pitch juga terjadi *coupling* yang terjadi pada ketinggian quadrotor.

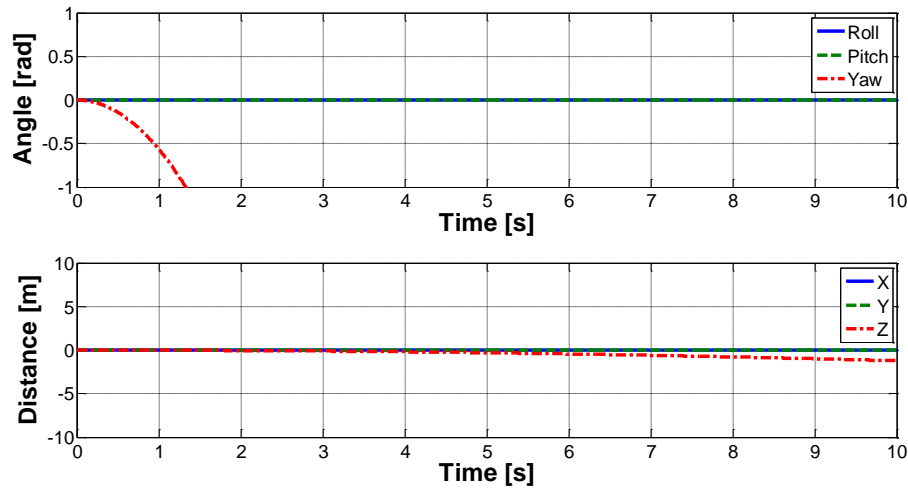


Gambar 4.6 Respon pengujian gerakan pitch

4.1.4 Pengujian Yaw

Pengujian terakhir untuk model dalam kondisi kalang terbuka adalah pengujian gerakan yaw. Gerakan ini terjadi karena perbedaan kecepatan putar pasangan motor 1 & 3 serta 2 & 4. Apabila motor 1 & 3 kecepatan putarnya melebihi kecepatan minimum pada kondisi melayang dan motor 2 & 4 kecepatan putarnya dibawah kecepatan minimum pada kondisi melayang maka gerakan yaw berputar sesuai dengan arah jarum jam. Begitu pula sebaliknya apabila motor 1 & 3 kecepatan putarnya dibawah kecepatan minimum pada kondisi melayang dan motor 2 & 4 kecepatan putarnya melebihi kecepatan minimum pada kondisi melayang maka gerakan yaw berputar berlawanan dengan arah jarum jam. Dapat dinotasikan pada kondisi sesuai arah jarum jam adalah $\omega_1 > \omega_H$ & $\omega_3 > \omega_H$ dan $\omega_2 < \omega_H$ & $\omega_4 < \omega_H$.

Pada pengujian ini responnya dapat dilihat pada gambar 4.7, pada respon sudut yaw terjadi perubahan. Perubahan pada sudut yaw ini tidak mempengaruhi posisi pada sumbu x , y dan ketinggian karena quadrotor hanya berputar pada sumbu z . Selain hal tersebut juga terjadi penurunan ketinggian dari quadrotor.



Gambar 4.7 Respon pengujian gerakan yaw

4.2 Pengujian Sistem Kendali

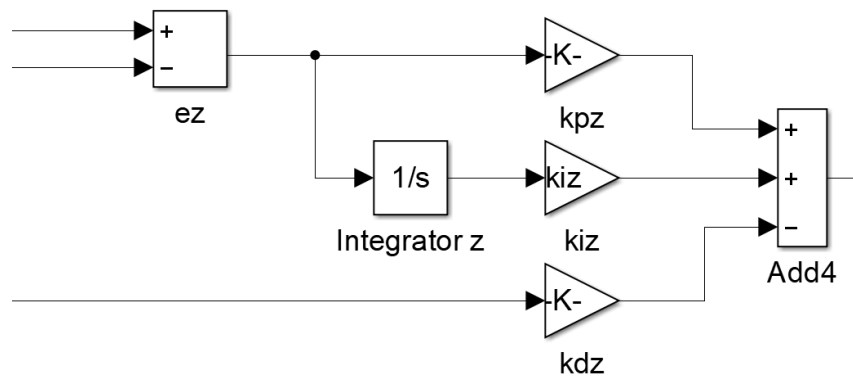
Bagian ini akan menjelaskan pengujian sistem kendali yang diimplementasikan pada simulasi model quadrotor. Ada 3 sistem kendali yang dilakukan pengujian kemudian dibandingkan, yaitu sistem kendali PID (Proporsional-Integral-Derivatif), sistem kendali LQR (*Linear Quadratic Regulator*) dan sistem kendali kombinasi PID dengan LQR dimana LQR berfungsi untuk menentukan parameter-parameter yang ada pada sistem kendali PID yaitu K_p , K_i , dan K_d .

4.2.1 Pengujian Sistem Kendali PID

Sistem kendali yang di akan disimulasikan pertama adalah sistem kendali PID (Proporsional-Integral-Derivatif). Sistem kendali PID merupakan jenis SISO (*Single Input Single Output*) sehingga pada masing-masing komponen yang ingin dikendalikan terdapat sebuah sistem kendali PID. Sesuai dengan pembahasan pada perancangan kendali PID, pada kasus quadrotor terbang melayang, ada 4 komponen yang dikendalikan yaitu sudut roll (ϕ), pitch (θ), dan yaw (ψ) serta ketinggiannya (z) sehingga ada 4 buah blok kendali PID pada masing-masing komponen.

Model Simulink dari sistem yang sudah dilengkapi dengan kendali PID ditunjukkan oleh gambar 4.8. Terdapat masukan pada sistem kendali PID yang merupakan data kondisi aktual dari sudut RPY dan ketinggian saat itu, kemudian

terdapat *Matlab Function* dimana kode sistem kendali PID dituliskan sebagaimana algoritma telah dijelaskan di atas, dan keluaran dari masing-masing sistem kendali dikombinasikan sesuai dengan persamanaa 2.17 – 2.20 untuk mendapatkan kecepatan putar dari masing-masing motor.



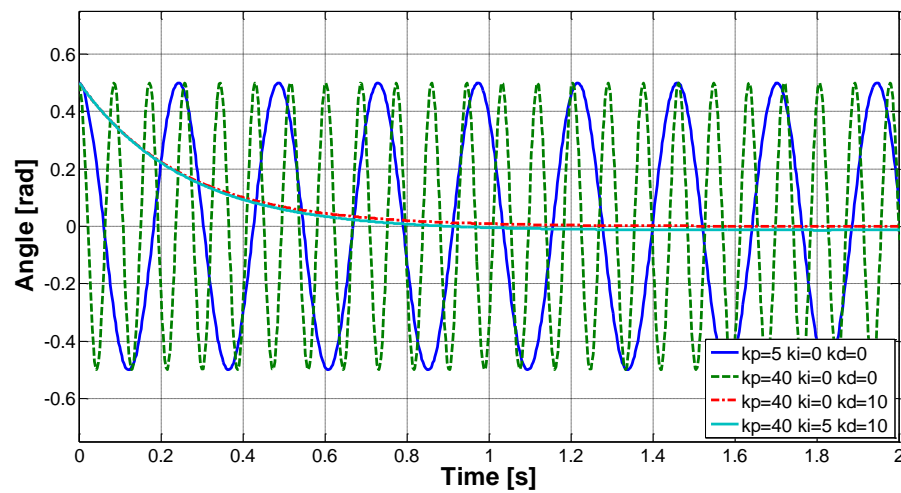
Gambar 4.8 Model Simulink Sistem Kendali PID pada Ketinggian

Ada beberapa poin penting yaitu adanya saklar untuk memudahkan pergantian dari kalang terbuka atau mode dengan sistem kendali, terdapat kolom untuk mengatur set poin pada masing-masing komponen sudut roll, pitch, yaw, dan ketinggian pada sisi kanan dari blok sistem kendali PID. Perlu menjadi catatan karena yang dikendalikan pada simulasi ini hanya pada 4 komponen tersebut sehingga posisi pada sumbu X dan Y dapat diabaikan.

Tabel 4.2 Kondisi Awal Quadrotor dan Set poin dari Sistem kendali PID

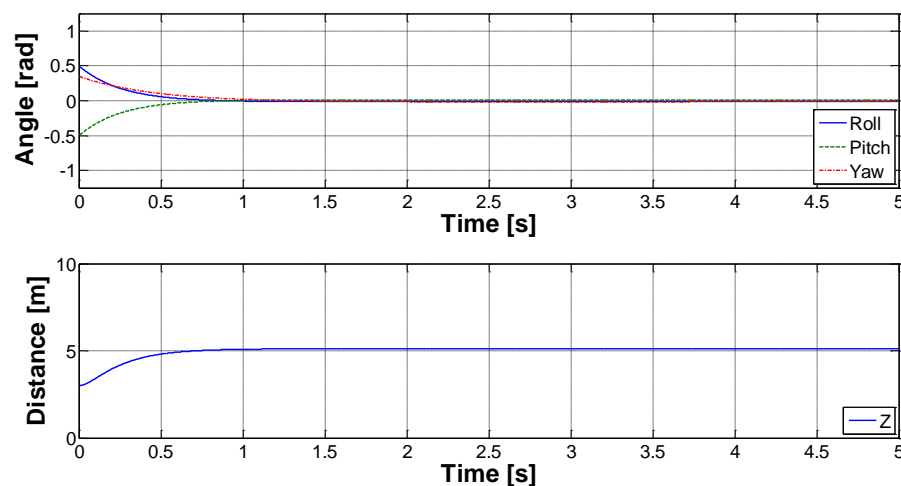
Variabel	Nilai Awal	Set Poin
Sudut Roll	0.5 rad	0 rad
Sudut Pitch	-0.5 rad	0 rad
Sudut Yaw	0.35 rad	0 rad
Ketinggian	3 meter	5 meter

Kondisi awal dari quadrotor dan set poin yang diinginkan ditunjukkan pada tabel 4.2. Pada kasus ini parameter K_p , K_i , dan K_d pada masing-masing kendali didapat dengan cara struktural seperti penjelasan pada bab sebelumnya. Pada gambar 4.9 merupakan grafik sudut roll dari beberapa nilai K_p , K_i , dan K_d yang di ujikan pada pengujian ini.



Gambar 4.9 Respon pengujian sudut roll pada berbagai nilai K_p , K_i , dan K_d

Terlihat pada gambar 4.9 pada pengujian nilai $K_p=5$ respon berosilasi, begitupula pada saat $K_p=40$. Semakin besar nilai K_p dapat dilihat respon semakin cepat. Saat ada penambahan kendali Derivatif dengan nilai $K_d=10$ respon yang didapatkan sudah baik dan stabil namun masih terjadi offset dengan nilai set poin yang diberikan. Terakhir adalah pemberian nilai $K_i=5$ sehingga offset sudah tidak terjadilagi.



Gambar 4.10 Respon pengujian kendali PID

Terlihat pada gambar 4.10 yang merupakan respon dari 4 komponen tersebut, pada ketinggian yang awalnya 3 meter dapat mencapai set poin pada 5 meter dalam waktu kurang dari 2 detik, dan untuk sudut roll, pitch, dan yaw dapat menuju titik

0 rad juga kurang dari 2 detik. Disini terlihat kendali PID dapat bekerja dengan baik sesuai dengan set poin yang diberikan.

Data dari respon terkait dengan settling time dan *overshot* yang terjadi pada pengujian ini ditunjukkan pada tabel 4.3.

Tabel 4.3 Data respon kendali PID

	Settling Time	Overshot
Ketinggian	1.275 s	0 %
Roll	1.108 s	0 %
Pitch	1.106 s	0 %
Yaw	1.3 s	0 %

4.2.2 Pengujian Sistem Kendali LQR

Sistem kendali kedua yang dilakukan untuk mengendalikan quadrotor pada kondisi melayang adalah sistem kendali LQR. Pada bab sebelumnya sudah didapatkan nilai matriks A , B dan C yang nantinya digunakan dalam menentukan *gain* K dari sebuah sistem kendali LQR. Nilai *gain* K pada kasus ini dicari menggunakan penyelesaian aljabar persamaan ricati yang dalam matlab telah disediakan fungsi *CARE*.

Sebelum mencari *gain* K kita perlu menentukan matriks bobot Q dan R terlebih dahulu. Tidak ada aturan baku dalam penentuan nilai matriks Q dan R . Sesuai dengan aturan yang dikemukakan oleh *Bryson* kita dapat menentukan awal dari nilai matriks Q dan R sesuai dengan persamaan 2.50. Untuk awal dari nilai matriks Q adalah dengan menentukan nilai maksimal dari tiap-tiap *state* x . Tabel 4.4 menunjukkan asumsi awal nilai maksimal untuk x .

Tabel 4.4 Nilai maksimal sudut RPY dan ketinggian

Parameter	Nilai
$ z $	$\leq 0.1 \text{ m}$
$ \phi $	$\leq 0.1 \text{ rad}$
$ \theta $	$\leq 0.1 \text{ rad}$

$ \psi $	$\leq 0.1 \text{ rad}$
$ \dot{z} $	$\leq 0.1 \text{ m/s}$
$ \dot{\phi} $	$\leq 1 \text{ rad/s}$
$ \dot{\theta} $	$\leq 1 \text{ rad/s}$
$ \dot{\psi} $	$\leq 1 \text{ rad/s}$

Dalam penentuan nilai matriks R kita perlu menentukan nilai maksimal dari masukan u . Asumsi nilai maksimal pada masukan u ditunjukkan pada tabel 4.4

Tabel 4.5 Nilai masukan (U)

Masukan	Nilai
U_1	≤ 1
U_2	≤ 1
U_3	≤ 1
U_4	≤ 1

Sesuai aturan dari *Bryson* kita dapat menentukan matriks dari Q dan R adalah seperti pada persamaan 4.4.

$$Q = \begin{bmatrix} \frac{1}{(z)_{max}^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{(\phi)_{max}^2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{(\theta)_{max}^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{(\psi)_{max}^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{(\dot{z})_{max}^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{(\dot{\phi})_{max}^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{(\dot{\theta})_{max}^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{(\dot{\psi})_{max}^2} \end{bmatrix}$$

$$R = \begin{bmatrix} \frac{1}{(U_1)_{max}^2} & 0 & 0 & 0 \\ 0 & \frac{1}{(U_2)_{max}^2} & 0 & 0 \\ 0 & 0 & \frac{1}{(U_3)_{max}^2} & 0 \\ 0 & 0 & 0 & \frac{1}{(U_4)_{max}^2} \end{bmatrix} \quad (4.4)$$

Setelah dilakukan beberapa iterasi dari nilai awal yang berdasarkan dengan tabel 4.3 dan 4.4 sehingga didapatkan nilai matriks Q dan R seperti yang ditunjukkan persamaan 4.5.

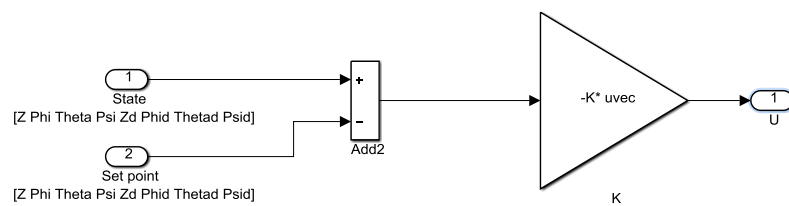
$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 750 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 750 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 750 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 20 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

Dengan berdasarkan matriks Q dan R yang telah kita dapatkan, kita bisa menentukan matriks K yang merupakan *gain* kendali dari sistem LQR yang kita rancang. Didapatkan nilai K adalah

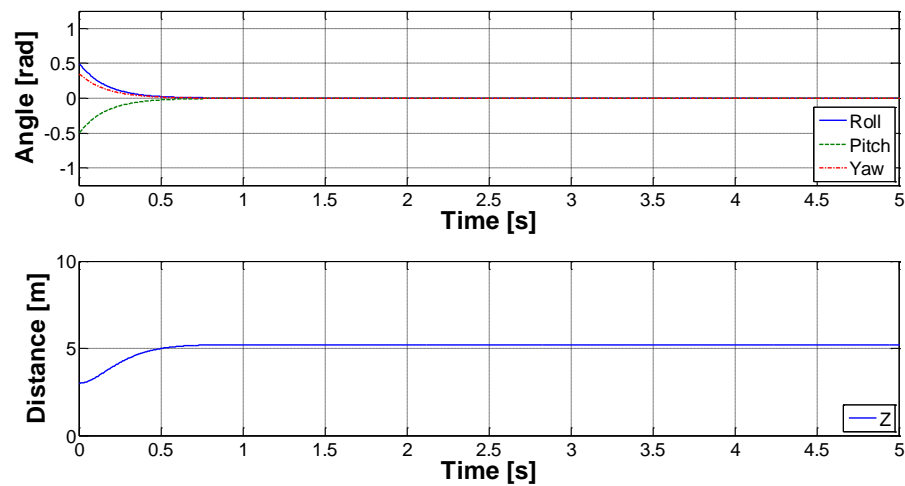
$$K = \begin{bmatrix} -31.62 & 0 & 0 & 0 & -8.13 & 0 & 0 & 0 \\ 0 & 27.38 & 0 & 0 & 0 & 4.51 & 0 & 0 \\ 0 & 0 & 27.38 & 0 & 0 & 0 & 4.51 & 0 \\ 0 & 0 & 0 & 27.38 & 0 & 0 & 0 & 4.55 \end{bmatrix} \quad (4.6)$$

Setelah mendapatkan nilai matriks K yang merupakan *gain* kendali LQR kemudian disimulasikan pada model quadrotor yang akan dijelaskan pada bab sebelumnya. Terlihat pada gambar 4.11 adalah blok diagram yang digunakan untuk melakukan simulasi pengujian kendali LQR. Ada 2 blok tambahan yang berbeda dibandingkan dengan pengujian sebelumnya, pengujian sistem kendali PID, yaitu *gain* matriks K dan suatu blok fungsi yang digunakan untuk mengubah nilai U menjadi ω masing-masing motor karena hasil perkalian antara kesalahan dan nilai *gain* matriks K adalah U (U_1 , U_2 , U_3 , dan U_4) sedangkan masukan dari model dinamika dari quadrotor adalah ω sehingga diperlukan proses mengubah dari bentuk U menjadi ω .



Gambar 4.11 Diagram Blok Pengujian sistem kendali LQR

Hasil simulasi pada pengujian ini ditunjukkan pada gambar 4.12. Terlihat pada gambar 4.12 pada ke-4 komponen yang dikendalikan dapat mengikuti set poin dengan baik. Nilai referensi yang diberikan pada ketinggian adalah 5 meter, sedangkan pada sudut roll, pitch, dan yaw adalah 0 rad. Nilai awal dari masing-masing komponen adalah 3 meter, 0.5 rad, 0.35 rad, dan -0.5 rad. Pada respon ketinggian pada waktu kurang dari 1 detik ketinggian sudah mencapai set poin yang diberikan begitupula pada komponen sudut roll, pitch, dan yaw terlihat kurang dari 1 detik nilai referensi sudah terpenuhi. Namun terlihat pada respon ketinggian terdapat sedikit overshoot sesaat sebelum mencapai set poin yang diinginkan.



Gambar 4.12 Respon pengujian kendali LQR

Untuk data respon baik settling time dan overshoot yang terjadi pada pengujian kendali LQR ditampilkan pada tabel 4.5.

Tabel 4.6 Data respon kendali LQR

	Settling Time	Overshoot
Ketinggian	0.85 s	1.5 %
Roll	0.82 s	0 %
Pitch	0.80 s	0 %
Yaw	0.75 s	0 %

4.2.3 Pengujian Sistem Kendali LQR-PID

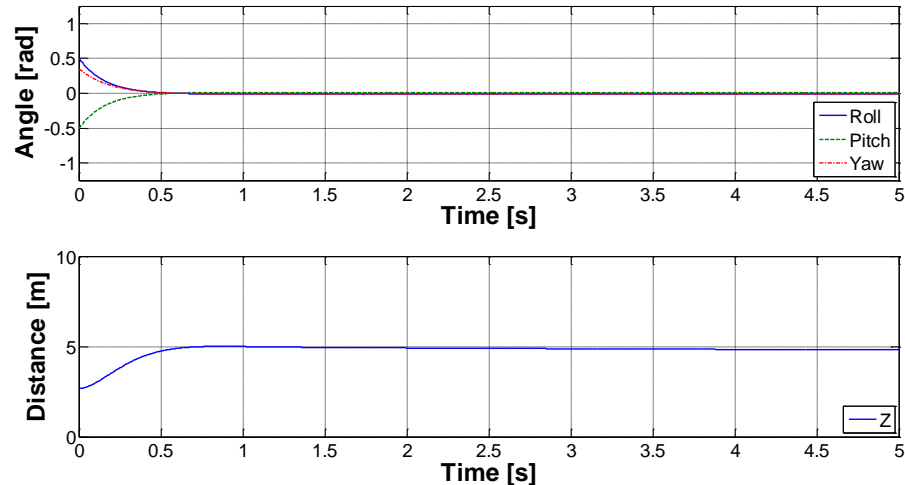
Pengujian sistem kendali LQR-PID adalah pengujian sistem kendali PID yang nilai parameternya didapatkan dari *gain* matriks K dari solusi LQR. Pada tabel 4.6 merupakan nilai K_p , K_i , dan K_d pada masing-masing blok sistem kendali PID.

Tabel 4.7 Parameter PID dengan penalaan berbasis LQR

Parameter	Nilai
K_{p_z}	33.73
K_{i_z}	7.07

Kd_z	9.68
$Kp_\phi = Kp_\theta$	32.48
$Ki_\phi = Ki_\theta$	5.47
$Kd_\phi = Kd_\theta$	5.04
Kp_ψ	28.46
Ki_ψ	5.91
Kd_ψ	5.07

Nilai parameter tersebut kemudian dimasukkan pada nilai parameter yang ada pada Simulink dan hasil respon yang terjadi terlihat seperti pada gambar 4.13. Terlihat pada grafik respon yang terjadi sangat baik dan terlihat cepat dan tidak ada overshoot yang terjadi pada semua respon.



Gambar 4.13 Respon pengujian kendali LQR-PID

Hasil dari data respon pengujian kendali PID dengan penalaan LQR baik settling time dan overshoot nya dapat dilihat pada tabel 4.8.

Tabel 4.8 Data respon kendali LQR-PID

	Settling Time	Overshot
Ketinggian	0.8 s	0 %
Roll	0.61 s	0 %
Pitch	0.62 s	0 %
Yaw	0.6 s	0 %

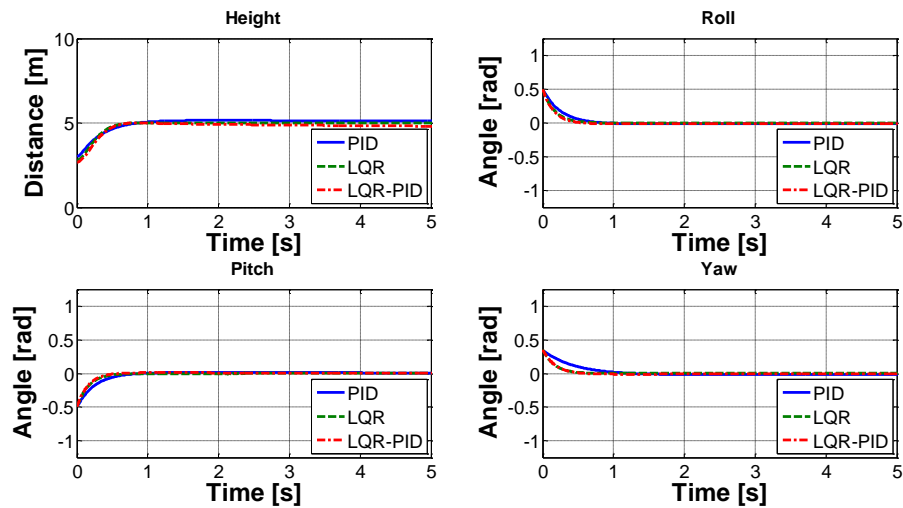
4.3 Perbandingan Sistem Kendali PID, LQR, dan LQR-PID

Setelah kita melakukan pengujian dari masing-masing sistem kendali pada sub-bab ini akan membahas mengenai perbandingan ketiga sistem kendali tersebut. Respon dari masing-masing sistem kendali kemudian diplot dalam 1 gambar agar memudahkan analisis. Dari respon yang terlihat kita bisa mengambil informasi sesuai dengan tabel 4.9.

Tabel 4.9 Perbandingan Respon Sistem Kendali PID, LQR, LQR-PID

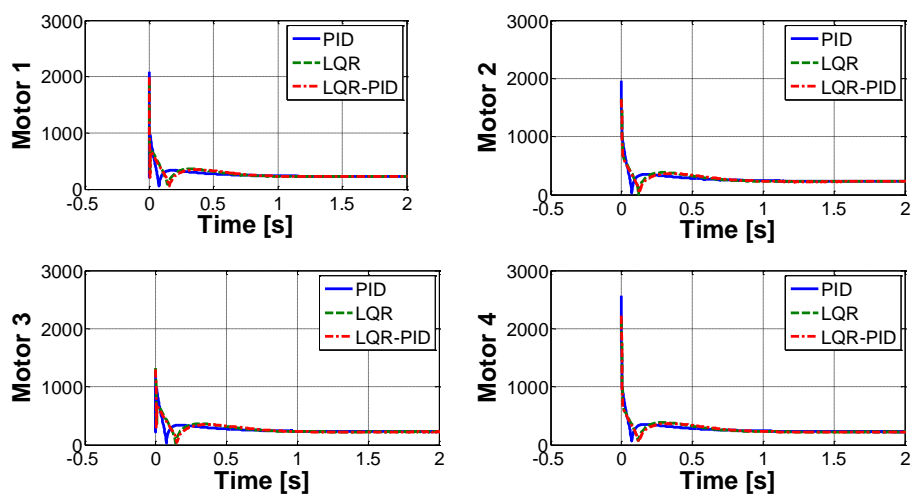
		PID	LQR	LQR-PID
Settling Time	Ketinggian	1.275 s	0.85 s	0.8 s
	Roll	1.108 s	0.82 s	0.61 s
	Pitch	1.106 s	0.80 s	0.62 s
	Yaw	1.3 s	0.75 s	0.6 s
Overshot(%)	Ketinggian	0 %	0 %	0 %
	Roll	0 %	0 %	0 %
	Pitch	0 %	0 %	0 %
	Yaw	0 %	0 %	0 %

Dari hasil respon yang terjadi semua sistem kendali yang dirancang dapat bekerja dengan baik terlihat pada saat tertentu semua respon sudah mengikuti dengan set poin yang diberikan. Namun apabila dibandingkan pada ketiga sistem kendali tersebut terlihat PID dengan penalaan berbasis LQR terlihat yang tercepat dalam settling time-nya pada semua respon kendali Grafik perbandingan diantara 3 sistem kendali ini digambarkan pada gambar 4.14..



Gambar 4.14 Perbandingan Respon Kendali PID, LQR, dan LQR-PID

Selain melihat respon keluaran pada masing-masing sistem kendali PID, LQR dan LQR-PID perlu diperhatikan juga terkait sinyal kendali yang dikeluarkan. Dalam keadaan nyata sinyal kendali nilainya terbatas, misal dalam kasus ini adalah quadrotor, motor memiliki batas maksimal putaran yang merupakan hasil konversi dari sebuah nilai sinyal kendali sehingga sangat tidak mungkin apabila dalam simulasi hasil responnya baik namun sinyal kendali yang dihasilkan terlalu besar sehingga tidak dapat diimplementasikan. Pada gambar 4.15 terlihat bentuk sinyal kendali pada masing-masing sistem kendali.



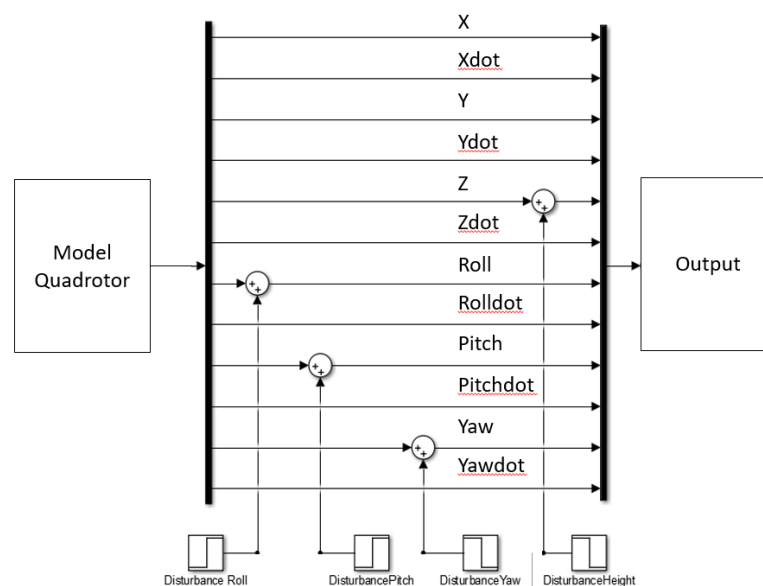
Gambar 4.15 Sinyal kendali PID, LQR dan LQR-PID pada masing-masing motor

Dari gambar 4.15 dapat dilihat bahwa sinyal kendali dari sistem kendali PID di awal waktu sangatlah besar sehingga mungkin saja hasil implementasi sistem kendali yang dirancang tidak sesuai dengan hasil simulasi. Selain dari keterbatasan tersebut, sinyal kendali berkaitan dengan energi sehingga apabila aktuator tersebut dapat menerima sinyal kendali yang sebegitu besar namun mungkin tidak menjadi efektif karena sumber daya yang dibutuhkan akan sangat besar.

4.4 Pengujian dengan Gangguan

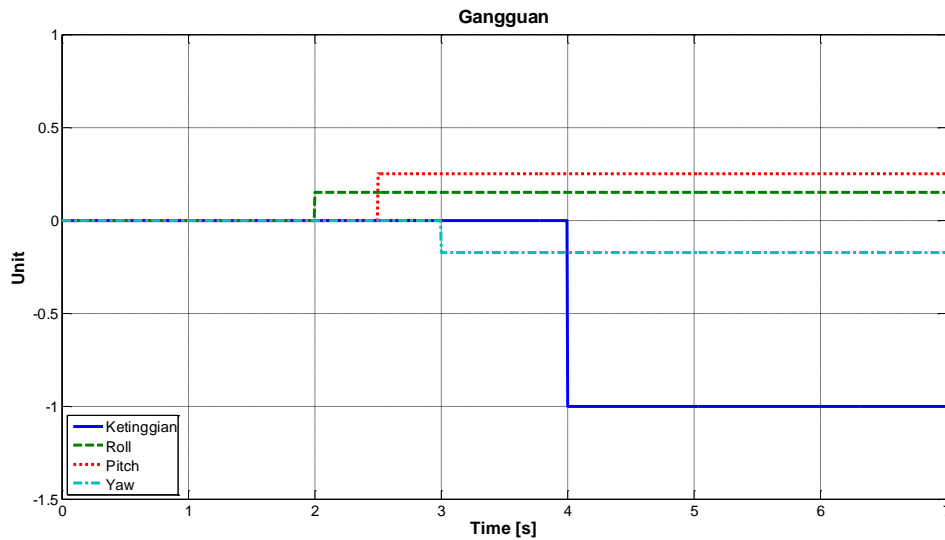
Pengujian selanjutnya adalah pengujian sistem kendali yang modelnya diberikan gangguan. Gangguan yang ditampilkan disini berpengaruh pada ketinggian, sudut roll, pitch, dan yaw.

Asumsi gangguan disini pada dunia nyata adalah angin dari luar sistem dengan besaran tertentu dan waktu tertentu pula sehingga pada simulasi gangguan dapat dianggap berupa unit step.



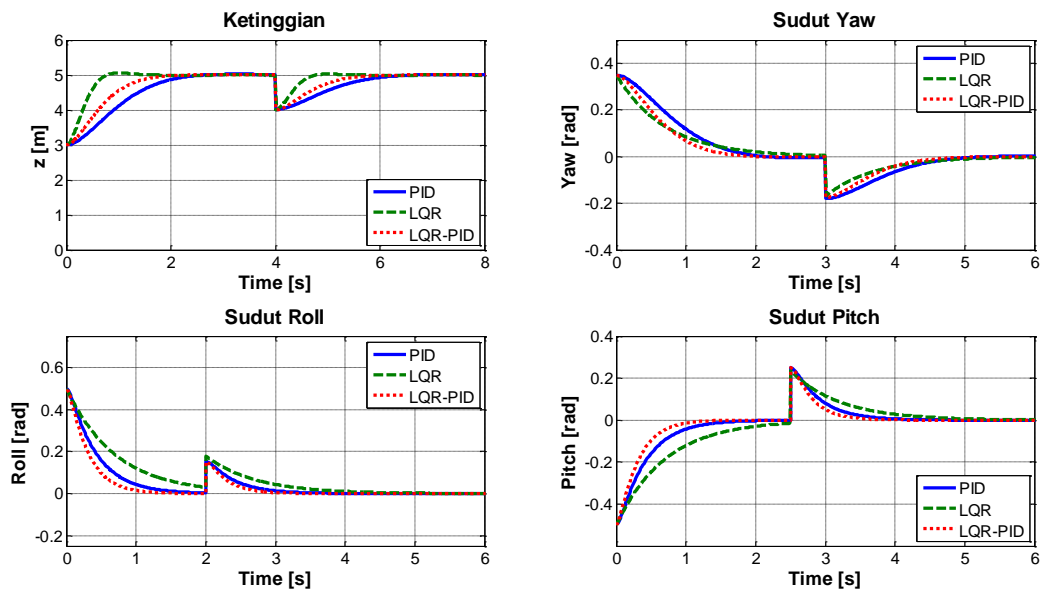
Gambar 4.16 Blok Simulink Gangguan

Pada gambar 4.16 terlihat gambar blok Simulink dari gangguan yang digunakan. Besarnya gangguan yang diberikan pada sudut roll adalah 0.15 rad pada detik ke-2, sudut pitch 0.25 rad pada detik ke-2.5, sudut yaw sebesar -0.175 pada detik ke-3, dan terakhir pada ketinggian dengan besar -1 m pada detik ke-4. Bentuk dari gangguan dapat dilihat seperti pada gambar 4.17.



Gambar 4.17 Bentuk Unit Step dari Gangguan

Set poin yang diberikan masih sama seperti pada pengujian sebelumnya yaitu untuk sudut roll, pitch, dan yaw adalah 0 rad, dan ketinggiannya 5 meter. Dengan kondisi awal juga sama seperti pada pengujian sebelumnya kemudian simulasi dijalankan dengan gangguan. Respon dari ketiga sistem kendali yaitu PID, LQR, dan LQR-PID terlihat pada gambar 4.18.



Gambar 4.18 Respon Ketinggian dan Sudut RPY dengan Gangguan

Terlihat pada grafik ketinggian sistem kendali LQR paling cepat dalam menanggapi respon sedangkan untuk respon sudut roll, pitch, dan yaw sistem kendali LQR-PID adalah yang terbaik dibanding sistem kendali lainnya.

4.5 Pengujian *Robustness*

Pengujian pada sub-bab ini adalah pengujian yang berkenaan dengan *robustness*. Langkah yang digunakan pada pengujian ini secara bertahap mengubah parameter dari model quadrotor. Parameter yang akan diubah adalah I_{xx} , I_{yy} , dan I_{zz} . Besarnya perubahan yang dilakukan adalah berkisar antara -50% sampai dengan +50% pada masing-masing parameter. Besarnya tiap perubahan kemudian dilakukan pengamatan terhadap respon yang terjadi pada masing-masing obyek kendali.

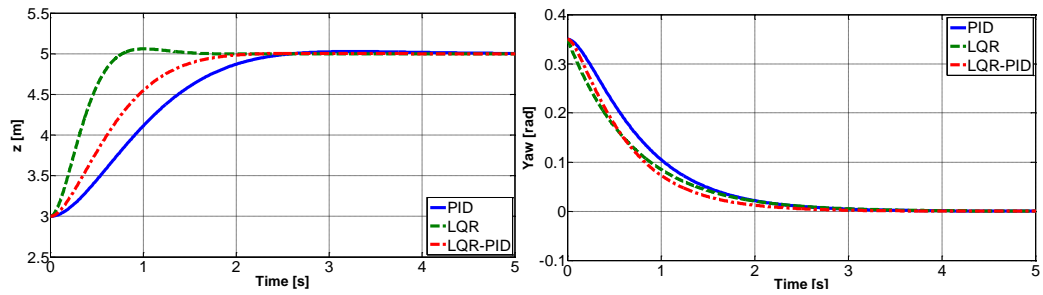
4.5.1 Parameter -50%

Nilai I_{xx} , I_{yy} , dan I_{zz} pada pengujian ini diubah nilainya menjadi sesuai dengan tabel 4.9. Pengurangan sebesar 50% dari nilai awal diberikan pada masing-masing nilai I_{xx} , I_{yy} , dan I_{zz} .

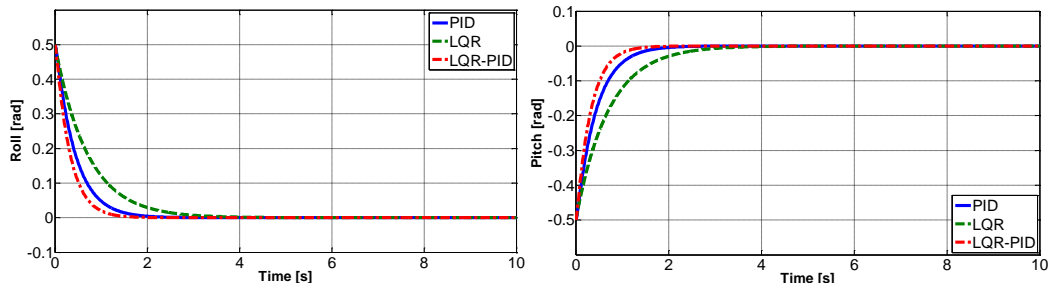
Tabel 4.10 Nilai Parameter I_{xx} , I_{yy} , dan I_{zz} pada parameter -50%

I_{xx}	0.037
I_{yy}	0.037
I_{zz}	0.065

Kemudian masing-masing sistem kendali PID, LQR, dan LQR-PID dijalankan dengan parameter yang baru, respon hasil dari perubahan parameter ini dapat dilihat pada gambar 4.18 – 4.19. Pada pengujian parameter -50% tidak terlihat perubahan yang berarti di semua sistem kendali baik pada ketinggian, sudut roll, pitch dan yaw. Sehingga pengujian selanjutnya perlu dilakukan agar kita dapat melihat apakah sistem kendali bekerja dengan konstan ketika terdapat perubahan parameter.



Gambar 4.19 Respon (a) Ketinggian dan (b) yaw pada parameter -50%



Gambar 4.20 Respon (a) roll dan (b) pitch pada parameter -50%

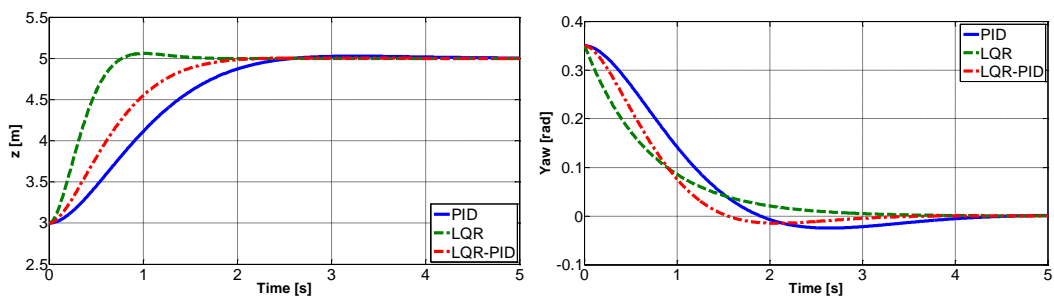
4.5.2 Parameter +50%

Nilai I_{xx} , I_{yy} , dan I_{zz} pada pengujian ini diubah nilainya menjadi sesuai dengan tabel 4.10.

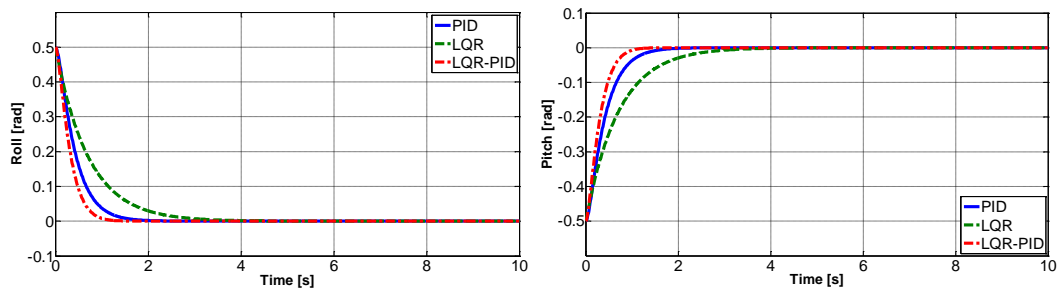
Tabel 4.11 Nilai Parameter I_{xx} , I_{yy} , dan I_{zz} pada parameter +50%

I_{xx}	0.0113
I_{yy}	0.0113
I_{zz}	0.0195

Hasil dari perubahan parameter ini dapat dilihat pada gambar 4.20 – 4.21.



Gambar 4.21 Respon (a) Ketinggian dan (b) yaw pada parameter +50%



Gambar 4.22 Respon (a) roll dan (b) pitch pada parameter +50%

Untuk respon ketinggian, sudut roll, dan pitch tidak terlihat adanya perubahan yang berarti. Semua sistem kendali dapat menghasilkan respon yang sesuai dengan nilai referensi yang diberikan, hanya saja perbedaan terletak pada kecepatan respon sudut yaw. Pada sub-bab 4.3 telah dijelaskan bahwa sistem kendali PID dengan tuning LQR adalah yang memiliki kecepatan respon paling baik diantara yang lain namun terlihat pada respon sudut yaw bahwa sistem kendali LQR-PID mengalami *overshoot* begitupula dengan sistem kendali PID sedangkan sistem kendali LQR tidak terjadi *overshoot*. Besarnya *overshoot* yang terjadi pada kendali PID adalah sekitar 4% sedangkan LQR-PID 3%. Ini membuktikan bahwa sistem kendali LQR lebih *robust* dibandingkan dengan sistem kendali lainnya yang diujikan karena dengan perubahan parameter, sistem kendali LQR tetap dapat bekerja dengan baik tanpa terjadi *overshoot*.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam penelitian ini menggunakan 3 sistem kendali yaitu PID, LQR, dan LQR-PID dalam menstabilkan terbang melayang quadrotor. Dari hasil perancangan kemudian disimulasikan dalam matlab Simulink dan hasil dari masing-masing sistem kendali sudah dipaparkan. Dari hasil tersebut dapat ditarik beberapa kesimpulan :

1. Secara umum semua sistem kendali yang dirancang sudah baik dan hasilnya sesuai yang diharapkan. Namun kecepatan respon dari sistem kendali LQR-PID lebih baik dari sistem kendali lainnya.
2. Dalam menentukan parameter Q dan R dari sebuah kendali LQR tidaklah unik, sehingga diawal proses iterasi bisa didapatkan respon yang tidak stabil sehingga penentuan nilai Q dan R membutuhkan waktu iterasi yang cukup agar hasilnya sesuai yang diharapkan
3. Hasil dari pengujian *robustness* menunjukkan bahwa sistem kendali LQR adalah yang terbaik dengan nilai 100% pada tingkat perubahan parameter +50% sedangkan kendali PID hanya bernilai 96% dan LQR-PID bernilai 97%.

5.2 Saran

Sebuah penelitian selalu ada pembaharuan, sehingga penelitian ini tentunya masih memerlukan penelitian lanjutan. Adapun saran-saran untuk penelitian selanjutnya dapat diberikan sebagai berikut :

1. Belum adanya kontroler untuk posisi X dan Y membuat quadrotor bergerak pada posisi translasi yang tidak terarah sehingga *double controller (inner & outer loop)* diperlukan untuk mengendalikan posisi translasi dari quadrotor

2. Sistem kendali yang telah dirancang dapat diimplementasikan pada obyek quadrotor yang nyata untuk lebih membuktikan bahwa sistem kendali yang dirancang sesuai kriteria yang diinginkan

DAFTAR PUSTAKA

- [1] P. Malone, H. Apgar, S. Stukes and S. Sterk, "Unmanned Aerial Vehicles unique cost estimating requirements", 2013 IEEE Aerospace Conference, 2013.
- [2] Y. Zeng, R. Zhang and T. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges", IEEE Communications Magazine, vol. 54, no. 5, pp. 36-42, 2016.
- [3] S. Gupte, Paul Infant Teenu Mohandas and J. Conrad, "A survey of quadrotor Unmanned Aerial Vehicles", 2012 Proceedings of IEEE Southeastcon, 2012.
- [4] A. Schollig, F. Augugliaro, R. D'Andrea, "A platform for dance performances with multiple quadrocopters", Proc. IEEE IRSJ Int. Conf Intelligent Robots and Systems-Workshop on Robots and Musical Expressions, pp. 1-8 2010.
- [5] S. Norouzi Ghazbi, Y. Aghli, M. Alimohammadi, A. A. Akbari, "Quadrotors Unmanned Aerial Vehicles: A Review", International Journal On Smart Sensing and Intelligent Systems, vol. 9 no. 1 2016.
- [6] M. Ma'sum, M. Arrofi, G. Jati, F. Arifin, M. Kurniawan, P. Mursanto and W. Jatmiko, "Simulation of intelligent Unmanned Aerial Vehicle (UAV) For military surveillance", 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2013.
- [7] H. Lim, J. Park, D. Lee and H. Kim, "Build Your Own Quadrotor: Open-Source Projects on Unmanned Aerial Vehicles", IEEE Robotics & Automation Magazine, vol. 19, no. 3, pp. 33-45, 2012.
- [8] Z. Mustapa, S. Saat, S. Husin and N. Abas, "Altitude controller design for multi-copter UAV", 2014 International Conference on Computer, Communications, and Control Technology (I4CT), 2014.
- [9] H. Meric, "Stabilization of Four rotors UAV", Australia : Institut Supérieur de l'Aéronautique et de l'Espace and National ICT Australia, 2010.
- [10] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo. Robotics. McGraw-Hill.
- [11] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and control for aerial grasping and manipulation," in Intelligent

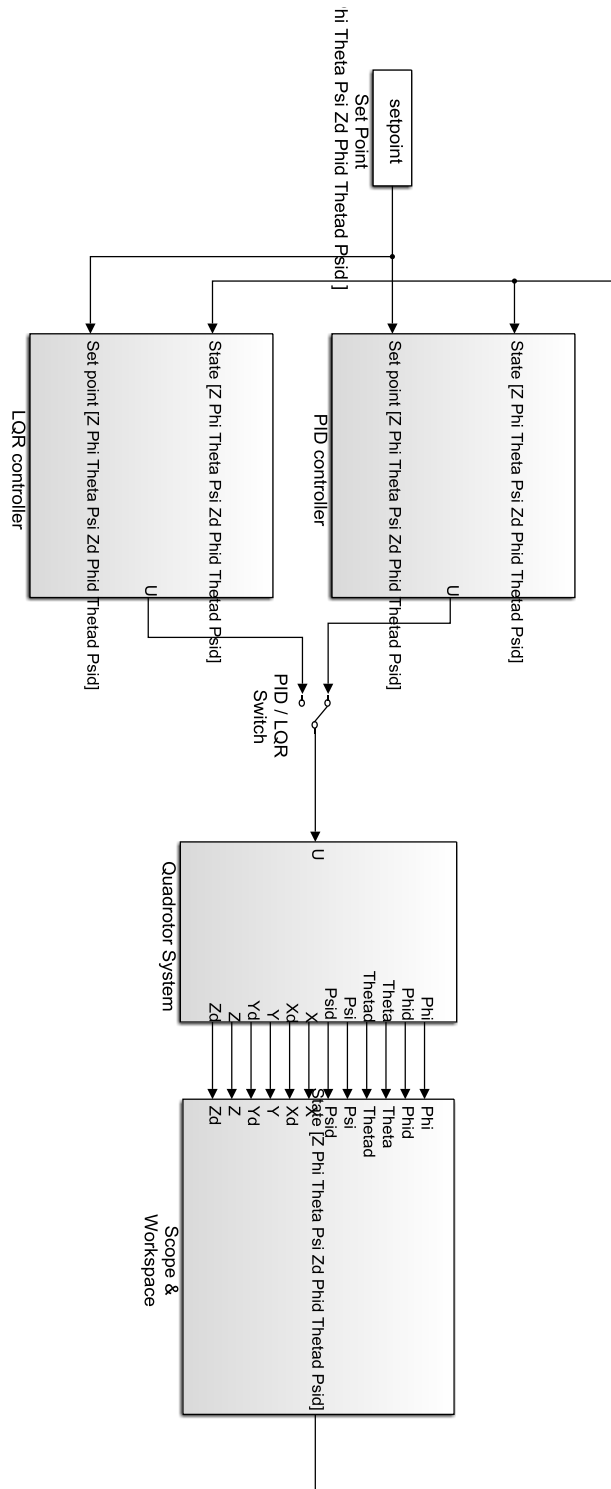
Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE, 2011, pp. 2668–2673.

- [12] D. Lee, T. Burg, D. Dawson, D. Shu, B. Xian, and E. Tatlicioglu, “Robust tracking control of an underactuated quadrotor aerial-robot based on a parametric uncertain model”, in Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on, 2009, pp. 3187–3192.
- [13] N. A. Ismail N. L. Othman Z. M. Zain D. Pebrianti L. Bayuaji "Attitude control of quadrotor" ARPN J. Eng. Appl. Sci. vol. 10 no. 22 pp. 17206-17211 Dec. 2015.
- [14] S. Ahmed, K. Kushsairy, M. Bakar, D. Hazry and M. Joyo, "Attitude stabilization of Quad-rotor (UAV) system using Fuzzy PID controller (an experimental test)", 2015 Second International Conference on Computing Technology and Information Management (ICCTIM), 2015.
- [15] H. Tnunay, M. Abdurrohman, Y. Nugroho, R. Inovan, A. Cahyadi and Y. Yamamoto, "Auto-tuning quadcopter using Loop Shaping", 2013 International Conference on Computer, Control, Informatics and Its Applications (IC3INA), 2013.
- [16] Kiam Heong Ang, G. Chong and Yun Li, "PID control system analysis, design, and technology", IEEE Transactions on Control Systems Technology, vol. 13, no. 4, pp. 559-576, 2005.
- [17] N. Mohamed, A. Abdalaziz, A. Ahmed and A. Ahmed, "Implementation of a PID control system on microcontroller (DC motor case study)", 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), 2017.
- [18] S. Ahmed, K. Kushsairy, M. Bakar, D. Hazry and M. Joyo, "Attitude stabilization of Quad-rotor (UAV) system using Fuzzy PID controller (an experimental test)", 2015 Second International Conference on Computing Technology and Information Management (ICCTIM), 2015.
- [19] A. Salih, M. Moghavvemi, H. Mohamed and K. Gaeid, "Modelling and PID controller design for a quadrotor unmanned air vehicle", 2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), 2010.
- [20] S. Khatoon, D. Gupta and L. Das, "PID & LQR control for a quadrotor: Modeling and simulation", 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2014.

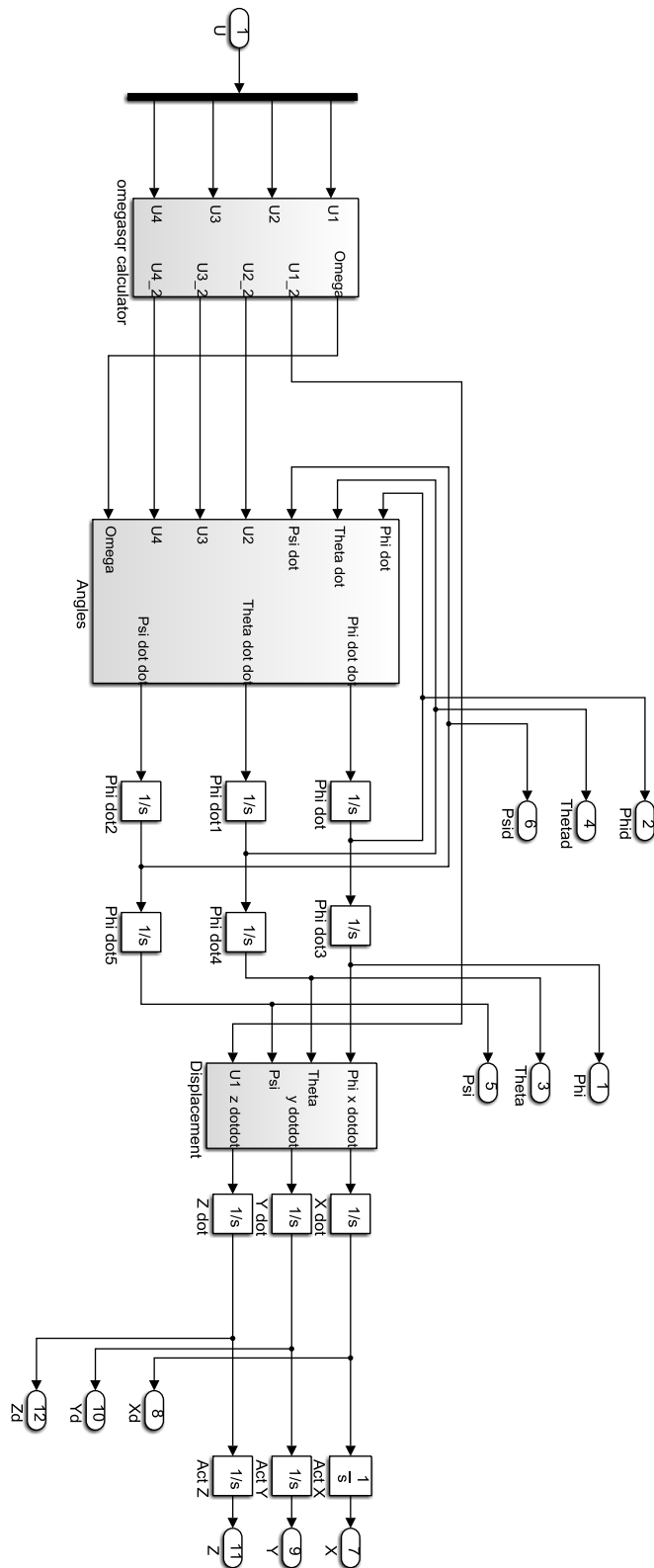
- [21] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor", 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007.
- [22] R. Mahony, V. Kumar and P. Corke, "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor", IEEE Robotics & Automation Magazine, vol. 19, no. 3, pp. 20-32, 2012.
- [23] D. Sagar Sahu and S. Sharma, "A Survey Paper on PID Control System", International Journal of Engineering Trends and Technology, vol. 21, no. 7, pp. 366-368, 2015.
- [24] K. Ogata, "Modern Control Engineering", 5th ed. Boston: Prentice-Hall electrical engineering series Instrumentation and control series, 2010.
- [25] Z. F. He and L. Zhao, "A Simple Attitude Control of Quadrotor Helicopter Based on Ziegler-Nichols Rules for Tuning PD Parameters," The Scientific World Journal, vol. 2014, Article ID 280180, 13 pages, 2014.
- [26] A.E. Bryson and Y.C. Ho, "Applied Optimal Control", Wiley New York, 1975.
- [27] S. Bouabdallah, "Design and Control of Quadrotors with Application to Autonomous Flying", Ecole Polytechnique Federale de Lausanne, 2007.
- [28] Argentim, L., Rezende, W., Santos, P. and Aguiar, R. (2013), "PID, LQR and LQR-PID on a quadcopter platform" International Conference on Informatics, Electronics and Vision (ICIEV), 2013.

LAMPIRAN

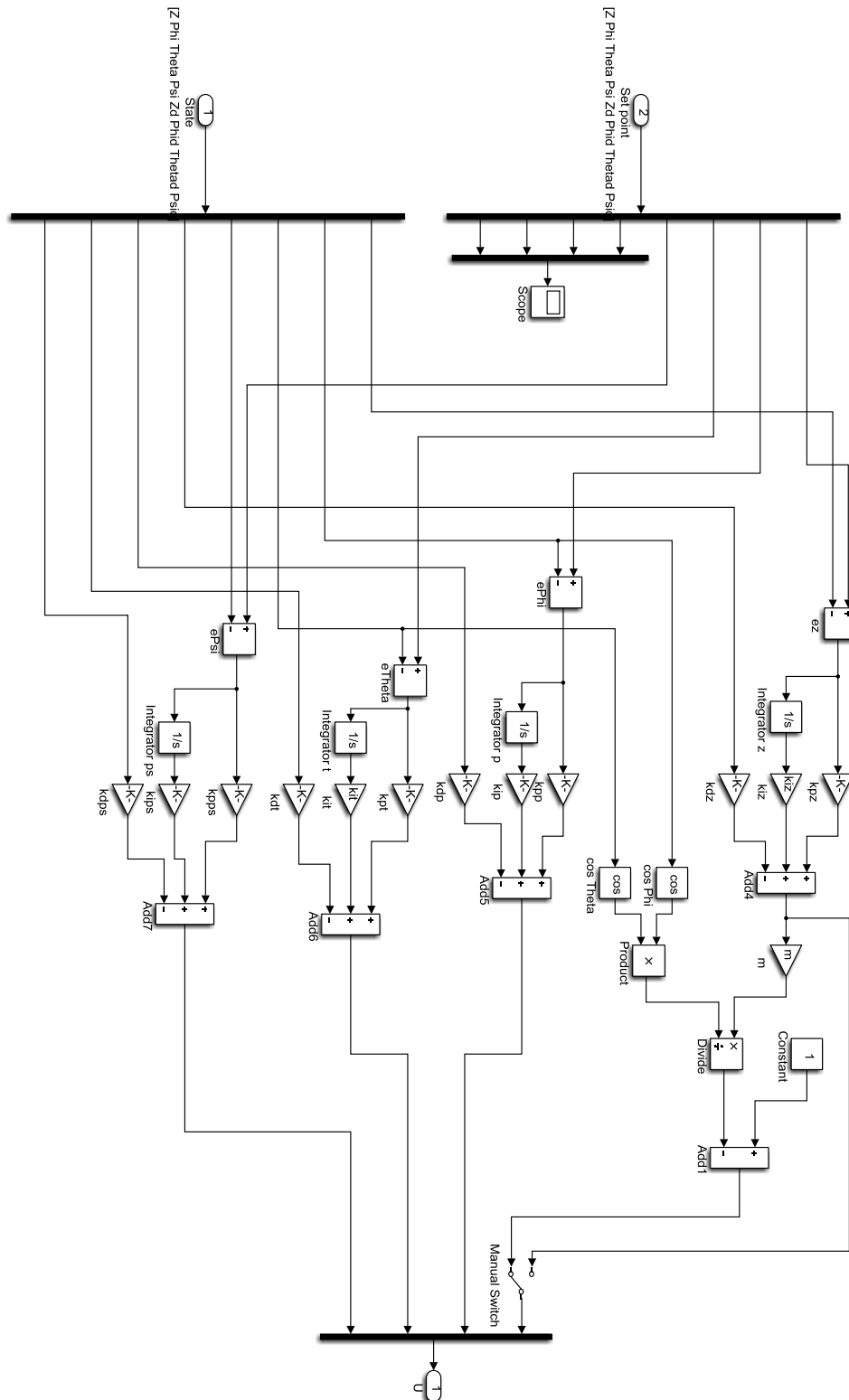
A . Blok Simulink



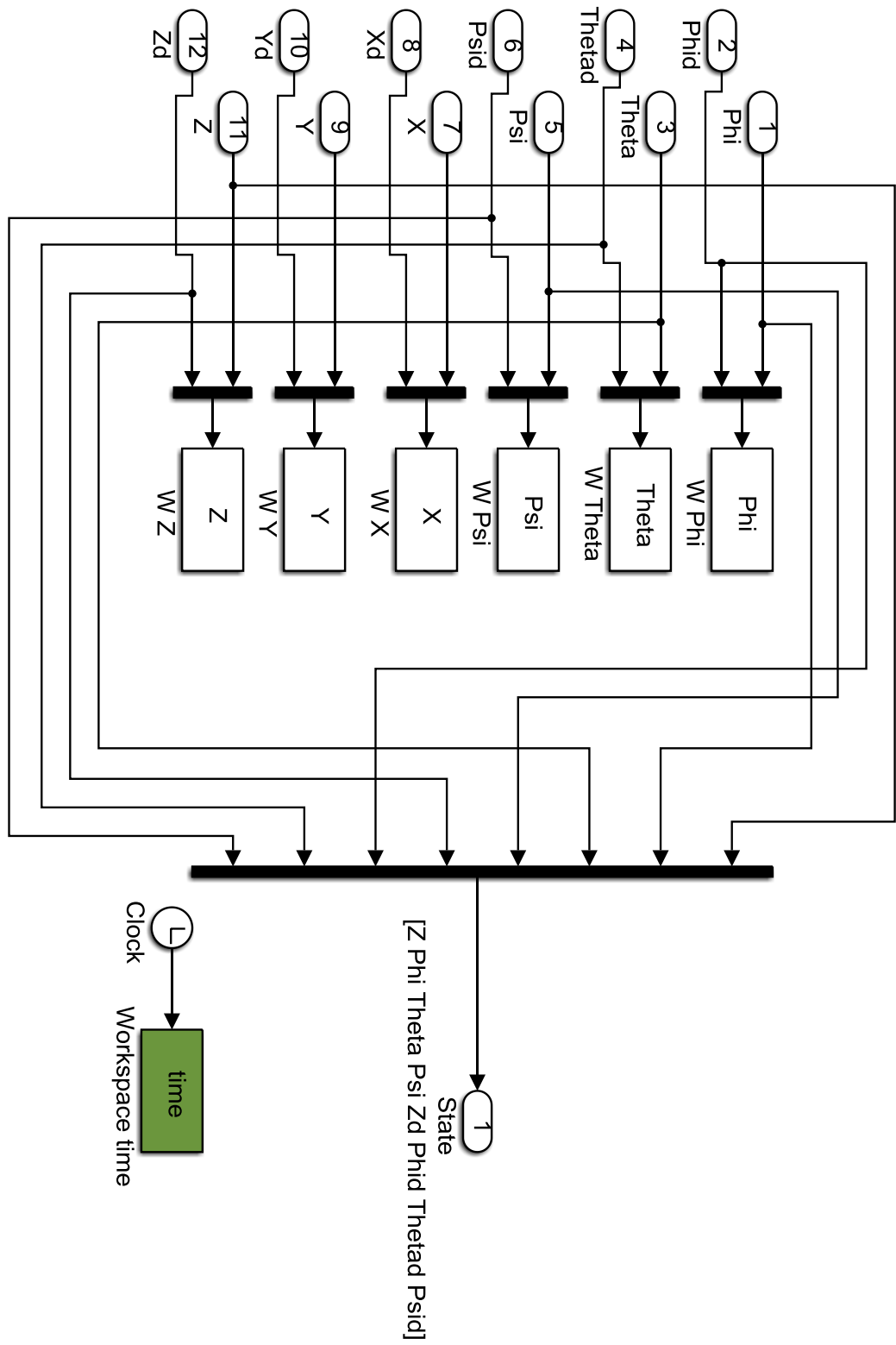
B. Blok Simulink Model Quadrotor



C. Blok Simulink Sistem Kendali PID



D. Blok Simulink Scope



E. Quadparameters.m

```
%Quadrotor parameter full
%by Ardy seto priambodo
%DTETI UGM

global lx ly lz Jr b d l m g
global kpp kip kdp kpt kit kdt kpps kips kdps kpz kiz kdz
global init setpoint
global integral_roll integral_pitch integral_yaw integral_z

% initial condition of Quadrotor
init = [0 0 0 0 3 0 0.5 0 -0.5 0 0.35 0]; %X Xd Y Yd Z Zd Phi Phid Theta Thetad
Psi Psid

% setpoint for Quadrotor condition
setpoint=[5 0 0 0 0 0 0 0]; %Z Phi Theta Psi Zd Phid Thetad Psid

integral_roll = 0;
integral_pitch = 0;
integral_yaw = 0;
integral_z = 0;

% PID controller

kpp = 40;
kip = 5;
kdp = 10;

kpt = 40;
kit = 5;
kdt = 10;

kpps = 35;
kips = 5;
kdps = 15;

kpz = 85;
kiz = 10;
kdz = 30;

% Quadrotor constants
lx = 7.5*10-3; % Quadrotor moment of inertia around X axis
ly = 7.5*10-3; % Quadrotor moment of inertia around Y axis
lz = 1.3*10-2; % Quadrotor moment of inertia around Z axis
Jr = 6.5*10-5; % Total rotational moment of inertia around the propeller axis
b = 3.13*10-5; % Thrust factor
d = 7.5*10-7; % Drag factor
l = 0.23; % Distance to the center of the Quadrotor
m = 0.65; % Mass of the Quadrotor in Kg
```

```
g = 9.81; % Gravitational acceleration
```

```
% gain feedback K  
matrix;
```

F. matrix.m

```
%load quadparameters  
%Quadparameters;
```

```
%matrix state A 8x8
```

```
A=[0 0 0 0 1 0 0 0;  
    0 0 0 0 0 1 0 0;  
    0 0 0 0 0 0 1 0;  
    0 0 0 0 0 0 0 1;  
    0 0 0 0 0 0 0 0;  
    0 0 0 0 0 0 0 0;  
    0 0 0 0 0 0 0 0;  
    0 0 0 0 0 0 0 0;  
    ];
```

```
%matrix state B 8x4
```

```
B=[0 0 0 0;  
    0 0 0 0;  
    0 0 0 0;  
    0 0 0 0;  
    -1/m 0 0 0;  
    0 1/lx 0 0;  
    0 0 1/ly 0;  
    0 0 0 1/lz;  
    ];
```

```
%matrix C 8x8
```

```
%matrix D 8x4
```

```
C=eye(8);  
D=zeros(8,4);
```

```
% matrix weight Q 8x8
```

```
Q=eye(8);
```

```
Q(1,1)=1000;  
Q(2,2)=750;  
Q(3,3)=750;  
Q(4,4)=750;  
Q(5,5)=25;  
Q(6,6)=20;  
Q(7,7)=20;  
Q(8,8)=20;
```

```

% matrix weight R 4x4
R=eye(4);

R(1,1)=1;
R(2,2)=1;
R(3,3)=1;
R(4,4)=1;

% algorithm for find gain matrix K 4x8 using algebra ricatti equation
[P,s,K]=care(A,B,Q,R);

```

G. Plot_all.m

```

nfig=0;
nfig=nfig+1;
figure(nfig);

subplot(2,1,1)
plot(time, Phi(:,1), '-', time, Theta(:,1), '--', time, Psi(:,1), '-.', 'LineWidth',2)
ylim([-1.25 1.25])
xlim([0 5])
set(gca,'fontsize',20)
ylabel('Angle [rad]', 'FontSize',27,'FontName','arial','FontWeight','bold')
xlabel('Time [s]', 'FontSize',27,'FontName','arial','FontWeight','bold')
legend({'Roll','Pitch','Yaw'}, 'FontSize',20,'Location','southeast')
grid on;

subplot(2,1,2)
plot(time, Z(:,1), '-', 'LineWidth',2)
ylim([0 10])
xlim([0 5])
set(gca,'fontsize',20)
ylabel('Distance [m]', 'FontSize',27,'FontName','arial','FontWeight','bold')
xlabel('Time [s]', 'FontSize',27,'FontName','arial','FontWeight','bold')
legend({'Z'}, 'FontSize',20,'Location','southeast')
grid on;

```

H. Matrix2.m

```

%load quadparameters
Quadparameters;

%matrix state A 12x12
A=[0 0 0 0 1 0 0 0 0 0 0 0;

```

```

0 0 0 0 0 1 0 0 0 0 0 0;
0 0 0 0 0 0 1 0 0 0 0 0;
0 0 0 0 0 0 0 1 0 0 0 0;
0 0 0 0 0 0 0 0 1 0 0 0;
0 0 0 0 0 0 0 0 0 1 0 0;
0 0 0 0 0 0 0 0 0 0 1 0;
0 0 0 0 0 0 0 0 0 0 0 1;
0 0 0 0 0 0 0 0 0 0 0 1;
0 0 0 0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0 0 0 0;
];

```

```

%matrix state B 12x4

```

```

B=[0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
0 0 0 0;
-1/m 0 0 0;
0 1/lx 0 0;
0 0 1/ly 0;
0 0 0 1/lz;
];

```

```

%matrix C 12x12

```

```

%matrix D 12x4

```

```

C=eye(12);
D=zeros(12,4);

```

```

% matrix weight Q 12x12

```

```

Q=eye(12);

```

```

Q(1,1)=50;
Q(2,2)=30;
Q(3,3)=30;
Q(4,4)=35;
Q(5,5)=1250;
Q(6,6)=1000;
Q(7,7)=1000;
Q(8,8)=750;
Q(9,9)=50;
Q(10,10)=25;
Q(11,11)=25;
Q(12,12)=25;

```

```

% matrix weight R 4x4

```

```

R=eye(4);

R(1,1)=1;
R(2,2)=1;
R(3,3)=1;
R(4,4)=1;

% algorithm for find gain matrix K 4x8 using algebra ricatti equation
[P,s,K]=care(A,B,Q,R);

% K -> PID Parameter
kpp = abs(K(2,6));
kip = abs(K(2,2));
kdp = abs(K(2,10));

kpt = abs(K(3,7));
kit = abs(K(3,3));
kdt = abs(K(3,11));

kpps = abs(K(4,8));
kips = abs(K(4,4));
kdps = abs(K(4,12));

kpz = abs(K(1,5));
kiz = abs(K(1,1));
kdz = abs(K(1,9));

matrix;

```