

**PERANCANGAN SISTEM KONTROL QUADROTOR LINCAH
BERBASIS SIMULASI *HARDWARE IN THE LOOP***

LAPORAN KEMAJUAN TUGAS AKHIR



Oleh

Deedat Fatahillah NIM: 13320072

**PROGRAM STUDI TEKNIK FISIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG
2023**

ABSTRAK

PERANCANGAN SISTEM KONTROL QUADROTOR LINCAH BERBASIS SIMULASI *HARDWARE IN THE LOOP*

Oleh

Deedat Fatahillah NIM: 13320072

(Program Studi Teknik Fisika)

Quadrotor merupakan salah satu tipe wahana nirawak melayang yang memiliki empat buah rotor untuk melakukan manuver yang membuatnya dapat bergerak dengan memiliki enam derajat kebebasan. Hal ini membuat quadrotor menjadi konfigurasi yang populer belakangan ini oleh industri dengan meningkatnya kompleksitas kriteria manuver dari suatu wahana nirawak. Namun sistem kontrol yang sudah ada sebelumnya kurang sesuai dalam melakukan kendali pada quadrotor yang bergerak dengan lincah. Sehingga, kebutuhan sistem kontrol mutakhir dalam melakukan kontrol pengendalian manuver quadrotor dengan lincah sangatlah dibutuhkan untuk memenuhi perkembangan aplikasi quadrotor kedepannya.

Penelitian ini akan membahas terkait desain sistem kontrol drone yang tepat guna dalam melakukan manuver lincah. Desain sistem kontrol akan dirancang secara langsung pada *flight controller* nyata sehingga sistem kontrol yang didesain diharapkan dapat diimplementasikan secara tepat. Selain itu, desain sistem kontrol ini akan dilakukan pengujian dengan lingkungan virtual dengan metode *Hardware in the loop*.

Kata kunci: quadrotor, sistem kontrol, wahana nirawak, *hardware in the loop*

ABSTRACT

CONTROL SYSTEM DESIGN FOR AGILE QUADROTOR BASED ON HARDWARE IN THE LOOP SIMULATION

By

Deedat Fatahillah NIM: 13320072

(Engineering Physics Study Program)

Quadrotor is one type of unmanned aerial vehicle with four rotors for maneuvering, allowing it to move with six degrees of freedom. This has made quadrotor a popular configuration in recent years due to the increasing complexity of maneuver criteria for unmanned vehicles. However, existing control systems are not well-suited for agile quadrotors. Therefore, the need for advanced control systems to handle the agile maneuver control of quadrotors is essential to meet the future applications of quadrotors.

This research will discuss the design of an effective drone control system for agile maneuvering. The control system design will be directly implemented on a real flight controller, with the expectation that the designed control system can be implemented accurately. Additionally, the control system design will be tested in a virtual environment using Hardware in the Loop method.

Keywords: *quadrotor, control systems, unmanned aerial vehicles, hardware in the loop*

**PERANCANGAN SISTEM KONTROL QUADROTOR LINCAH
BERBASIS SIMULASI *HARDWARE IN THE LOOP***

HALAMAN PENGESAHAN

Oleh

Deedat Fatahillah NIM: 13320072

(Program Studi Teknik Fisika)

Institut Teknologi Bandung

Menyetujui

Tim Pembimbing

Tanggal

Pembimbing 1

Pembimbing 2

Dr. Ir. Eko Mursito Budi, M.T.
NIP. 196710061997021001

Faqihza Mukhlis, S.T., M.T., Ph.D.
NIP. 121110001

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah Swt. atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan kemajuan tugas akhir ini dengan judul “Perancangan Sistem Kontrol Quadrotor Lincak Berbasis Simulasi Hardware In The Loop”. Laporan kemajuan tugas akhir ini disusun sebagai salah satu syarat kelulusan mata kuliah TF4091 Metoda Penelitian dan Ujian Komprehensif, serta merupakan salah satu syarat dalam penyelesaian tugas akhir.

Dalam penyusunan laporan kemajuan tugas akhir ini, penulis mendapatkan bimbingan, bantuan, dan dukungan dari berbagai pihak sehingga pada kesempatan ini, penulis mengungkapkan rasa terima kasih kepada:

1. Dr. Ir. Eko Mursito Budi, M.T. dan Faqihza Mukhlis, S.T., M.T., Ph.D. selaku dosen pembimbing yang senantiasa memberikan bimbingan dan arahan kepada penulis dalam penyusunan laporan kemajuan tugas akhir ini.
2. Keluarga penulis atas doa dan dukungan yang diberikan tanpa henti.
3. Seluruh dosen pengajar yang telah membekali penulis ilmu pengetahuan selama menjalankan studi di Program Studi Teknik Fisika ITB.
4. Seluruh staf dan tenaga kependidikan Program Studi Teknik Fisika ITB atas bantuan dan dukungan kepada penulis selama menjadi mahasiswa di Program Studi Teknik Fisika ITB.
5. Seluruh teman penulis yang memberikan dukungan sehingga penulis senantiasa bersemangat dalam penyusunan laporan kemajuan tugas akhir ini.
6. Seluruh pihak yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa laporan kemajuan tugas akhir ini masih memiliki kekurangan dan jauh dari kesempurnaan. Oleh karena itu, penulis sangat terbuka atas saran dan masukan dari berbagai pihak untuk pengembangan dan/atau perbaikan laporan ini agar menjadi lebih baik. Akhir kata, penulis berharap laporan

kemajuan tugas akhir ini dapat menjadi awal yang baik dalam pengerjaan penelitian tugas akhir untuk memberikan kontribusi positif dan bermanfaat bagi pembacanya.

Bandung, 8 Desember 2023

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	viii
DAFTAR SINGKATAN DAN LAMBANG	ix
BAB I PENDAHULUAN	10
1.1 Latar Belakang	10
1.2 Rumusan Masalah	11
1.3 Tujuan Penelitian.....	11
1.4 Batasan dan Asumsi	11
1.5 Sistematika Penulisan.....	11
BAB II STUDI LITERATUR	13
2.1 Posisi Penelitian	13
2.2 Quadrotor	14
2.3 Arsitektur Quadrotor	16
2.4 Perangkat Lunak PX4 Autopilot	17
BAB III METODOLOGI DAN KEMAJUAN	19
3.1 Metodologi	19
3.2 Kemajuan	21
3.2.1 Pemodelan Dinamika Sistem.....	21
3.2.2 Studi Simulasi Gazebo	25
3.2.3 Studi Pemrograman Pixhawk	27
3.2.4 Simulasi <i>Hardware in the Loop</i>	28
BAB III PENUTUP	30
4.1 Rencana Selanjutnya	30
4.2 Penutup.....	30
DAFTAR PUSTAKA	31

DAFTAR GAMBAR

Gambar 2.1 Quadrotor dengan Frame Dji F450 [11]	15
Gambar 2.2 Konfigurasi Quadrotor	15
Gambar 2.3 Enam Derajat Kebebasan Dari Quadrotor [8]	16
Gambar 3.1 Diagram Alir Rencana Penelitian.....	21
Gambar 3.2 Sumbu Stasioner dan Sumbu Quadrotor [8]	22
Gambar 3.3 Cuplikan Kode Simulasi Model Quadrotor.....	26
Gambar 3.4 Simulasi Penerbangan Quadrotor Pada Gazebo.....	27
Gambar 3.5 Hasil Implementasi Program Pada <i>Console</i>	28
Gambar 3.6 Simulasi <i>Hardware in the Loop</i>	29

DAFTAR TABEL

Tabel 2.1 Tabel Perbandingan Penelitian Sebelumnya.....	13
Tabel 2.2 Tabel Perbandingan Penelitian Sebelumnya.....	14

DAFTAR SINGKATAN DAN LAMBANG

SINGKATAN	Nama	Pemakaian pertama kali pada halaman
UAV	<i>Unmanned Aerial Vehicle</i>	8
PID	<i>Proportional Integral Derivative</i>	8
LQR	<i>Linear Quardratic Regulator</i>	8
HILS	<i>Hardware in the Loop Simulation</i>	8
MPC	<i>Model Predictive Control</i>	11

LAMBANG

x	Posisi pada koordinat x	10
y	Posisi pada koordinat y	10
z	Posisi pada kootdinat z	10
θ	Sudut <i>roll</i>	10
ϕ	Sudut <i>pitch</i>	10
ψ	Sudut <i>yaw</i>	10

BAB I

PENDAHULUAN

1.1 Latar Belakang

Beberapa tahun terakhir, pengembangan teknologi seputar *unmanned aerial vehicle* (UAV) atau pesawat tanpa awak telah berkembang dengan pesat yang mengakibatkan adanya peningkatan aplikasi UAV yang masif di berbagai macam industri. Macam-macam pengaplikasiannya antara lain di ranah logistik, misi penyelamatan, infrastruktur, agrikultur, maupun seni [1]. Dengan semakin banyak industri yang mengadopsi UAV, kriteria gerak dan manuver dari sebuah UAV juga semakin kompleks. Sehingga, penggunaan konfigurasi kendaraan nirawak dengan empat rotor atau quadrotor menjadi sangatlah populer. Hal ini dikarenakan quadrotor mampu bergerak dengan lincah sehingga mampu melakukan manuver yang kompleks [2], [3]. Agar quadrotor dapat melakukan manuver dengan lincah maka harus terdapat suatu desain sistem kontrol yang dapat menjaga quadrotor tetap stabil.

Pada umumnya, sistem kontrol quadrotor menggunakan suatu bentuk atau variasi dari pengontrol proporsional, integral, dan derivative (PID) [4]. Pengontrol ini merupakan tipe pengontrol yang paling sederhana sehingga seseorang yang tidak memiliki latar belakang di bidang kontrol dapat menggunakannya dengan cukup baik. Namun, berdasarkan beberapa penelitian yang telah dilakukan, pengontrol PID telah terbukti menghasilkan manuver yang lebih buruk dibandingkan dengan sistem kontrol yang berbeda seperti *Fuzzy Logic* maupun *Linear Quadratic Regulator* (LQR) [5], [6]. Sehingga, pada penelitian ini akan dilakukan desain dan implementasi sistem kontrol mutakhir untuk kasus manuver quadrotor secara lincah.

Sistem kontrol yang telah didesain tersebut tentunya perlu diuji untuk melihat kinerja pengontrol dalam melakukan tugasnya. Proses pengujian pengontrol ini dapat dilakukan dengan menggunakan simulasi sehingga resiko kerusakan sistem fisik dapat dihindari. Salah satu strategi untuk melakukan simulasi ini ialah *Hardware in the Loop Simulation* (HILS). Metode ini dapat mengkombinasikan perangkat keras *flight controller* dengan simulasi lingkungan secara virtual.

Sehingga sistem kontrol dapat didesain secara tepat dengan mempertimbangkan limitasi komputasi *flight controller* seperti yang sudah dilakukan pada penelitian sebelumnya [7].

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, permasalahan yang akan dicoba untuk diselesaikan ialah desain dan implementasi sistem kontrol quadrotor mutakhir untuk manuver lincah dengan pendekatan simulasi *Hardware in the loop*.

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini ialah mengembangkan sistem kontrol quadrotor lincah dengan memanfaatkan simulator *hardware in the loop*. Sehingga sasaran dari penelitian ini adalah:

1. Merancang lingkungan simulasi sistem kontrol secara waktu nyata dengan metode *Hardware in the Loop*.
2. Mendesain sistem kontrol mutakhir untuk quadrotor lincah.
3. Memprogram dan mengimplementasi sistem kontrol pada perangkat keras pengontrol penerbangan secara langsung.
4. Menganalisis kinerja sistem kontrol berdasarkan kinerjanya dalam simulator.

1.4 Batasan dan Asumsi

Untuk memenuhi tujuan dan sasaran pada perancangan penelitian ini, beberapa batasan yang digunakan antara lain:

1. Lingkungan simulasi dianggap ideal.
2. Struktur dari quadrotor dianggap tegar dan simetris. Titik pusat massa berada pada titik tengah dari quadrotor.
3. Pengaruh dari gaya horizontal dan vertikal pada baling-baling diabaikan.
4. Simulator quadrotor memiliki sistem pengukuran yang lengkap.

1.5 Sistematika Penulisan

Laporan kemajuan tugas akhir ini dituliskan dengan sistematika sebagai berikut:

1. Bab I Pendahuluan

Pada bab ini dijelaskan latar belakang, rumusan masalah, tujuan, batasan dan asumsi, serta sistematika penulisan laporan kemajuan tugas akhir.

2. Bab II Studi Literatur

Pada bab ini dijelaskan studi literatur yang ditinjau dan digunakan sebagai referensi dalam penelitian ini.

3. Bab III Metodologi dan Kemajuan

Pada bab ini dijelaskan metodologi penelitian dan kemajuan yang sudah dilakukan selama ini.

4. Bab IV Rencana Selanjutnya & Penutup

Pada bab ini memuat rencana riset kedepannya beserta harapan penulis sebagai penutup laporan kemajuan tugas akhir.

BAB II

STUDI LITERATUR

2.1 Posisi Penelitian

Dalam penelitian tugas akhir ini akan dilakukan perancangan sistem kontrol quadrotor lincah berbasis *hardware in the loop*. Penelitian sebelumnya berperan untuk memperkaya referensi penelitian dan membedakannya dengan penelitian yang sedang dilakukan. Adapun poin-poin penting dari studi literatur terkait penelitian ini telah dirangkum pada

Tabel 2.1 Tabel Perbandingan Penelitian Sebelumnya

Jenis	Judul Penelitian	Isi	Alasan tinjauan penelitian
Tesis	“Design and Implementation of Model Predictive Control on Pixhawk Flight Controller”[8]	Pengembangan pengontrol <i>Model Predictive Control</i> (MPC) pada pengontrol pixhawk	Hasil dan metodologi dari tesis ini akan dipakai sebagai acuan hasil dan langkah yang perlu dilakukan pada penelitian ini.
Jurnal	“Development of Hardware-in-the-loop Simulation Based on Gazebo and Pixhawk for Unmanned Aerial Vehicles” [7]	Pengembangan lingkungan simulasi <i>hardware in the loop</i> pada gazebo sebagai pengujian sistem kontrol pixhawk.	Hasil dan metodologi jurnal penelitian ini akan dipakai sebagai acuan hasil dan langkah yang harus dilakukan pada penelitian ini.

Tabel 2.2 Tabel Perbandingan Penelitian Sebelumnya

Jenis	Judul Penelitian	Isi	Alasan tinjauan penelitian
Jurnal	“Aggressive Quadrotor Flight Through Narrow Gaps with Onboard Sensing and Computing”[9]	Pengujian quadrotor lincah dengan rintangan berupa celah sempit yang dapat dilewati oleh quadrotor.	Metode pengujian ini dapat digunakan sebagai referensi pengujian kelincahan sistem kontrol yang pada penelitian ini.
Artikel	“Modeling and Control of X-Shape Quadcopter”[3]	Pemodelan dan rancang sistem kontrol dengan metode proportional-integral-derivative (PID) pada quadrotor tipe x.	Metode penurunan dinamika quadrotor akan digunakan sebagai acuan penurunan dinamika quadrotor pada penelitian ini.

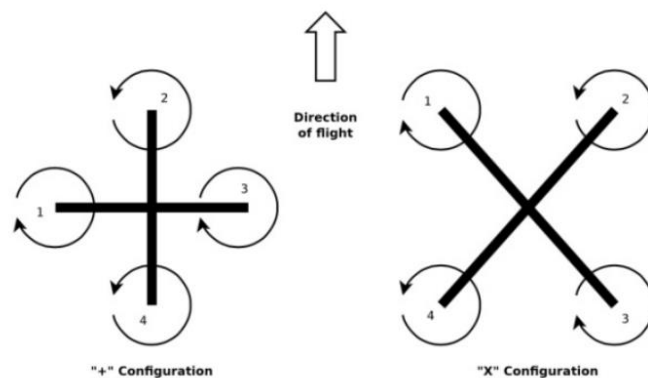
2.2 Quadrotor

Quadrotor adalah sebuah pengembangan dari konsep robot terbang untuk kendaraan terbang nirawak. Kendaraan ini memiliki total empat motor, dengan dua pasang motor bergerak berlawanan arah putar. Motor diletakkan pada sudut-sudut dari quadrotor yang dapat dilihat seperti pada Gambar 2.1 [10]. Konfigurasi ini memastikan wahana dapat terbang dengan stabil di angkasa.



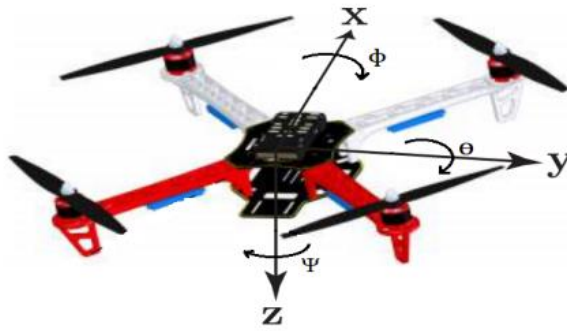
Gambar 2.1 Quadrotor dengan Frame Dji F450 [11]

Quadrotor memiliki dua tipe konfigurasi berdasarkan cara propellernya terkoneksi. Dua tipe ini ialah konfigurasi “X” atau “+” [12]. Baling-baling terhubung berpasangan berlawanan arah untuk membatalkan torsi yang diciptakan oleh baling-baling yang berputar ke satu arah. Pada gambar, baling-baling yang berlabel 1 dan 2 akan berputar secara berlawanan arah jarum, sedangkan baling-baling berlabel 3 dan 4 akan berputar searah dengan arah jarum jam. Hal ini dapat lebih jelas dilihat pada Gambar 2.2.



Gambar 2.2 Konfigurasi Quadrotor

Dengan konfigurasi ini, quadrotor dapat diklasifikasikan sebagai suatu wahana terbang yang memiliki enam derajat kebebasan. Tiga derajatnya ialah translasional yang biasanya dapat direpresentasikan dengan koordinat kartesian dan tiga derajat lain ialah gerak rotasional yang biasanya dapat direpresentasikan dengan menggunakan sudut euler [8]. Ilustrasi derajat kebebasan sebuah quadrotor dapat dilihat pada Gambar 2.3.

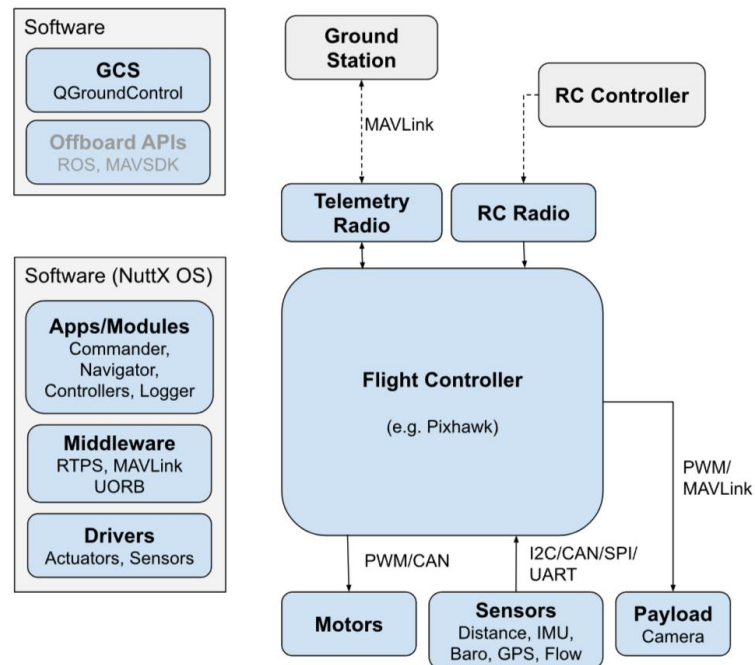


Gambar 2.3 Enam Derajat Kebebasan Dari Quadrotor [8]

Quadrotor juga merupakan sebuah wahana yang dapat diklasifikasikan sebagai sistem yang *underactuated*. Hal ini dikarenakan quadrotor hanya memiliki empat aktuator untuk mengontrol enam derajat kebebasan. Sehingga sebuah sistem kontrol sangat dibutuhkan dalam melakukan kompensasi untuk melakukan stabilisasi dan manuver sebuah quadrotor.

2.3 Arsitektur Quadrotor

Seperti yang sudah dijelaskan sebelumnya, quadrotor merupakan sistem yang *underactuated*. Agar quadrotor dapat melakukan manuver dan pergerakan di angkasa, tentunya sebuah quadrotor perlu memiliki sebuah pengontrol yang dapat memberi perintah pada setiap baling-baling untuk berputar berdasarkan *setpoint* posisi yang telah diberikan oleh pilot. Pengontrol penerbangan (atau yang sering disebut *flight controller*) akan disambungkan kepada segala sensor, aktuator, dan modul komunikasi dari quadrotor untuk mengendalikan penerbangan dari quadrotor tersebut. Arsitektur keseluruhan sistem ini dapat dilihat pada Gambar 2.4



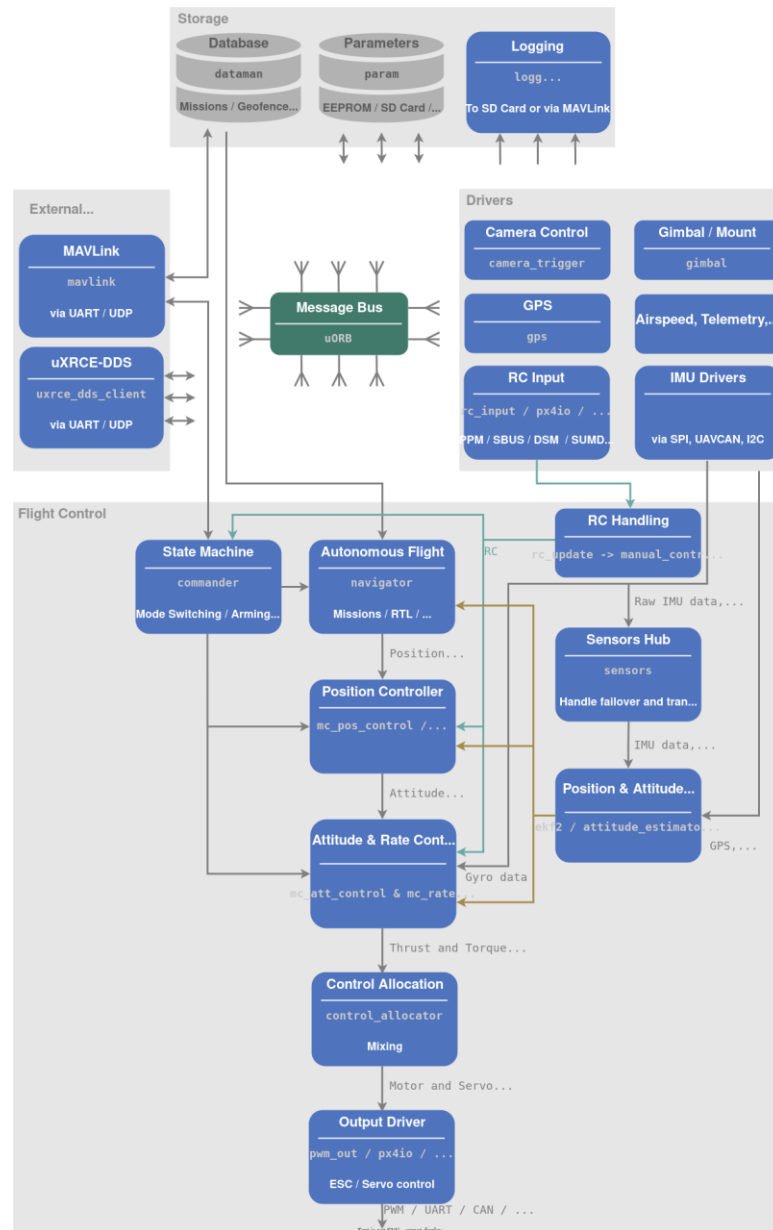
Gambar 2.4 Arsitektur Sistem Quadrotor [13]

Jenis *flight controller* yang akan digunakan pada penelitian ini ialah pixhawk. Belakangan ini, pixhawk sering digunakan dalam berbagai penelitian maupun aplikasi wahana terbang lainnya. Pixhawk menjadi opsi yang cukup populer dikarenakan aspek *open-source*-nya yang membuat pengontrol ini menjadi opsi yang terbaik di kalangan peneliti. Pixhawk juga memiliki fitur simulasi *hardware in the loop* dengan menggunakan *firmware* PX4. Aspek *open source* dan kapabilitas melakukan simulasi *hardware in the loop* inilah yang membuat pixhawk merupakan jenis pengontrol yang sesuai dalam melakukan penelitian ini.

2.4 Perangkat Lunak PX4 Autopilot

Pixhawk nantinya diunggah *firmware* PX4 yang berfungsi sebagai otak dari quadrotor yang akan diuji. Secara garis besar, pengontrol PX4 memiliki dua lapisan utama yaitu *flight stack* dan *middleware*. *Flight stack* merupakan bagian yang berfungsi untuk melakukan estimasi dan menjalankan algoritma sistem kontrol pada penerbangan, sedangkan *middleware* merupakan lapisan yang berfungsi untuk melakukan komunikasi antara komponen dari quadrotor seperti sensor dan modul telemetry radio [14]. Secara keseluruhan arsitektur perangkat lunak dari PX4 ini dapat dilihat pada Gambar 2.5. Semua fungsi perangkat lunak ini akan dialokasikan

dan diorganisir oleh satu sistem operasi sekaligus protokol komunikasi yang bernama uORB. Sehingga pembuatan perangkat lunak dapat dilakukan dengan memberi perintah pada *compiler* untuk mengompilasi semua fungsi yang ada dalam satu sistem operasi yang dapat dijalankan pada pixhawk secara *non-blocking* dan efisien.



Gambar 2.5 Arsitektur Perangkat Lunak PX4

BAB III

METODOLOGI DAN KEMAJUAN

3.1 Metodologi

Rencana penelitian yang akan dilakukan dapat dilihat pada gambar di bawah. Secara garis besar, tahapan di bawah dapat dibagi menjadi dua bagian utama yaitu perancangan sistem kontrol dan pengujian pengontrol. Bagian perancangan sistem kontrol terdiri dari tiga bagian utama yaitu, pemodelan dan analisis dinamika sistem, membangun lingkungan simulasi pada gazebo, desain pengontrol agile drone, serta studi pemrograman pixhawk. Sedangkan bagian pengujian terdiri dari dua bagian utama yaitu, pengujian performa kontroler secara simulasi dan pengujian performa kontroler pada drone fisik.

Tahap desain sistem kontrol ini akan dilakukan tentunya dengan pertama-tama melakukan analisis dinamika sistem dan memodelkan sistem kontrol secara keseluruhan. Tahap ini dilakukan untuk mengetahui variabel apa saja yang perlu dilakukan aksi kontrol sehingga pengontrol dapat didesain dengan tepat. Seperti motor apa yang harus digerakan untuk mengubah posisi maupun sudut dari UAV $(x, y, z, \theta, \phi, \psi)$.

Setelah itu, akan dilakukan pembangunan lingkungan simulasi pada gazebo. Gazebo sendiri merupakan sebuah perangkat lunak yang akan digunakan dalam membuat lingkungan simulasi sekaligus melakukan pengujian sistem kontrol secara simulasi. Pembuatan lingkungan simulasi ini akan dilakukan dengan sebaik mungkin sehingga aproksimasi yang telah dibuat pada simulasi dapat merepresentasikan kondisi nyata dari UAV maupun lingkungan sekitarnya dengan ketelitian yang tinggi. Tahap ini tidak hanya penting dalam proses pengujian sistem kontrol nanti, namun dalam melakukan desain juga. Tahap ini akan krusial dalam mengetahui karakteristik gangguan yang akan diterima oleh UAV.

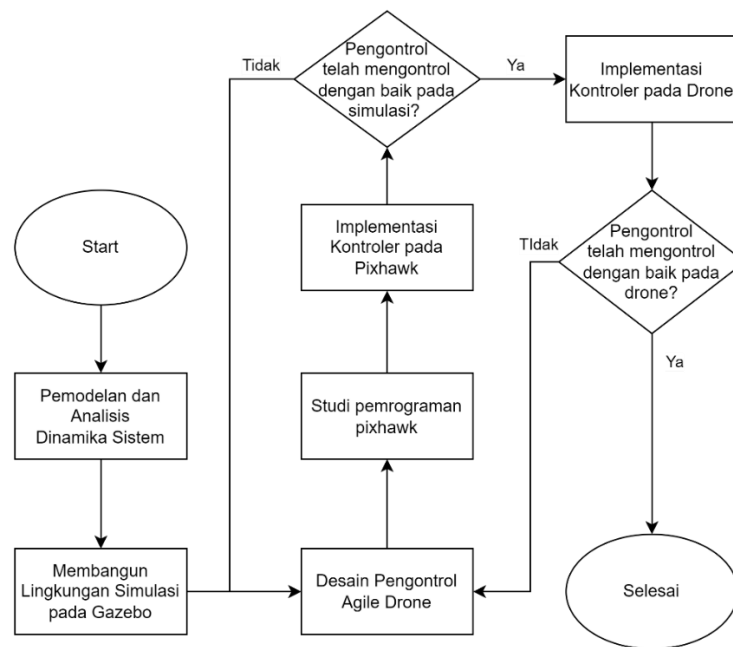
Dengan mengetahui karakteristik dinamika sistem UAV dan karakteristik gangguan yang ada, maka langkah selanjutnya ialah melakukan desain pengontrol yang tepat guna untuk mencapai objektif yang dituju. Pada tahap inilah, akan dilakukan pembelajaran lebih lanjut mengenai berbagai macam sistem kontrol yang sekiranya

sesuai, seperti *model predictive control* (MPC), *linear quadratic control* (LQR), atau bentuk kontroler lainnya yang akan diuji. Selain aspek-aspek gangguan dari luar, aspek seperti limitasi komputasi dan waktu cacah dari *flight controller* juga akan diperhitungkan. Hal ini perlu dipertimbangkan karena sistem kontrol ini akan diimplementasi pada perangkat *flight controller* nyata sehingga aspek-aspek ini tentunya merupakan aspek yang cukup penting.

Desain sistem kontrol yang dilakukan akan menghasilkan diagram blok dan persamaan matematis dari sistem. Namun hal ini tidak bisa semerta-merta diimplementasi secara langsung kepada *flight controller* yang ada. Sehingga untuk mengimplementasi sistem kontrol yang sudah dibuat, maka perlu dilakukan studi mengenai bagaimana cara mengimplementasi sistem kontrol yang sudah dirancang pada *flight controller* pixhawk secara *software*. Dengan memahami ini, maka tahap selanjutnya dapat dilakukan secara langsung.

Tahap selanjutnya ialah, tahap pengujian sistem kontrol yang telah didesain. Tahap ini diawali dengan menggunakan lingkungan simulasi yang sudah dibuat sebelumnya dengan menggunakan metode *hardware in the loop* (HILS). Metode ini akan dilakukan dengan pertama-tama mengunggah kode sumber sistem kontrol kepada *flight controller* secara langsung dan menghubungkan *flight controller* tersebut kepada lingkungan simulasi yang telah dirancang. *Flight controller* akan menganggap dirinya hidup di dalam lingkungan tersebut sehingga pengujian sistem kontrol yang telah dibuat. Jika keluaran dari sistem kontrol ini kurang sesuai dengan yang diinginkan, maka desain sistem kontrol akan dilakukan kembali sampai sistem kontrol yang diimplementasi dan diuji telah sesuai dengan kriteria yang telah ditentukan.

Setelah sistem kontrol yang cocok telah ditemukan pada tahap sebelumnya, sistem kontrol ini akan diimplementasi secara langsung kepada wahan terbang quadcopter nyata. Sama seperti tahap sebelumnya, hal ini dilakukan untuk mengetahui kinerja sistem kontrol secara nyata dan bila perlu dilakukan modifikasi kepada sistem kontrol yang ada. Namun, tahap ini merupakan tahap yang bersifat opsional dikarenakan terdapat penelitian lain yang akan mengerjakan aspek ini.

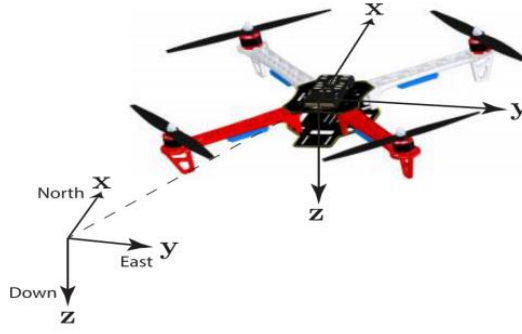


Gambar 3.1 Diagram Alir Rencana Penelitian

3.2 Kemajuan

3.2.1 Pemodelan Dinamika Sistem

Dalam melakukan pemodelan sistem secara keseluruhan, hal pertama yang harus dilakukan terlebih dahulu ialah menentukan kerangka acuan dari quadrotor dan bumi. Sumbu stasioner akan dinamakan X_E, Y_E , dan Z_E sedangkan sumbu drone akan dinamakan X_b, Y_b , dan Z_b . Sumbu stasioner akan diletakkan pada bumi yang akan menunjukkan kepada arah utara, timur, dan ke arah bumi. Sedangkan sumbu x dari drone akan menghadap di antara propeller 1 dan propeller 3, sumbu y akan menghadap di antara propeller 1 dan propeller 4 dan sumbu z akan menghadap kepada arah bumi. Ilustrasi dari sumbu quadcopter dapat dilihat pada gambar.



Gambar 3.2 Sumbu Stasioner dan Sumbu Quadrotor [8]

Sehingga vektor koordinat X_b, Y_b , dan Z_b dapat dituliskan sebagai berikut.

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = R \begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix}$$

Dengan matriks rotasi R dapat ditulis sebagai berikut.

$$R = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$

Dimana ϕ , θ , dan ψ merupakan sudut euler dari quadrotor yang dapat dilihat lebih jelas pada Gambar 2.3. Sedangkan S_ϕ dan C_ϕ merupakan $\sin(\phi)$ dan $\cos(\phi)$ masing-masingnya. Matriks rotasi R ini akan digunakan dalam menurunkan dinamika translasi dari quadrotor.

Dengan asumsi permukaan bumi adalah rata, sehingga pengaruh permukaan tanah dapat diabaikan, persamaan gerak quadrotor yang memiliki enam derajat kebebasan dapat diturunkan dengan menggunakan persamaan Newton-Euler.

$$\begin{bmatrix} mI & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega}^b \end{bmatrix} + \begin{bmatrix} \omega^b \times mV \\ \omega^b \times J\omega^b \end{bmatrix} = \begin{bmatrix} f \\ \tau^b \end{bmatrix}$$

Dengan I matriks identitas, m massa total dari quadrotor, J momen inerti, ω kecepatan angular, V kecepatan linear, f gaya total, dan τ momen total.

Pada saat melakukan pergerakan di udara, quadrotor akan dipengaruhi oleh 3 gaya, gaya-gaya tersebut adalah gaya gravitasi, gaya gesekan dengan udara dan gaya angkat baling-baling yang dijabarkan oleh persamaan berikut.

$$F_{net} = F_{ar} + F_g + F_s$$

$$F_{net} = f = \omega^b \times mV + mI\dot{V}$$

Dengan F_{ar} adalah gaya angkat total yang dihasilkan oleh empat motor dengan konfigurasi sebagai berikut.

$$F_{ar} = [F_{ax} \ F_{ay} \ F_{az}]^T$$

Namun, dikarenakan hanya terdapat empat motor ini akan memberikan resultan gaya hanya pada sumbu z yang merupakan arah absolut dari badang quadrotor sehingga.

$$F_{ax} = 0$$

$$F_{ay} = 0$$

$$F_{az} = (F_{a1} + F_{a2} + F_{a3} + F_{a4})$$

Dengan pengaruh gaya gravitasi.

$$F_g = mg$$

Pengaruh gesekan badan quadrotor terhadap udara

$$F_{sx} = \frac{1}{2} C_x A_c \rho \dot{x} |\dot{x}|$$

$$F_{sy} = \frac{1}{2} C_y A_c \rho \dot{y} |\dot{y}|$$

$$F_{sz} = \frac{1}{2} C_z A_c \rho \dot{z} |\dot{z}|$$

$$F_s = [F_{sx} \ F_{sy} \ F_{sz}]^T$$

Namun, F_{ar} dengan F_g tidak bekerja pada kerangka koordinat yang sama. Sehingga perlu dilakukannya transformasi dari satu acuan koordinat menjadi bentuk acuan koordinat yang lain. Dengan demikian gaya translasi dalam kerangka koordinat bumi dapat ditulis sebagai berikut.

$$F_{net} = -F_a + F_g - F_s$$

$$F_{net} = -R F_a + F_g - F_s$$

Sehingga didapatkan persamaan sebagai berikut.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \frac{-(F_{a1} + F_{a2} + F_{a3} + F_{a4})}{m} \begin{bmatrix} c\phi s\theta c\psi + s\phi s\psi \\ c\phi s\theta s\psi - s\phi c\psi \\ c\phi c\theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} - \frac{1}{2m} \begin{bmatrix} C_x A_c \rho \dot{x} |\dot{x}| \\ C_y A_c \rho \dot{y} |\dot{y}| \\ C_z A_c \rho \dot{z} |\dot{z}| \end{bmatrix}$$

Setelah mendapatkan persamaan translasi quadrotor, dinamika rotasi dari quadrotor dapat dicari dengan menggunakan persamaan Newton-Euler juga yang dituliskan sebagai berikut.

$$J\dot{\omega}^b + \omega^b \times J\omega^b = \tau^b$$

Dengan mengasumsikan benda tegar dengan massa konstan dan sumbu-sumbu inersia sesuai dengan sumbu utama, maka persamaan dinamika dapat dituliskan sebagai berikut.

$$\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \tau^b$$

Di mana ω merupakan sudut-sudut euler berupa roll (ϕ), pitch (θ), dan yaw (ψ). Momen gaya ditimbulkan oleh kombinasi empat rotor adalah tiga pergerakan sesuai dengan arah sudut euler, yaitu

$$\begin{aligned} M_x &= (F_{a1} + F_{a4} - F_{a2} - F_{a3})l \sin(\alpha) \\ M_y &= (F_{a1} + F_{a2} - F_{a3} - F_{a4})l \sin(\beta) \\ M_z &= (-T_1 - T_3 + T_2 + T_4) \end{aligned}$$

Dengan α merupakan sudut yang dibentuk oleh lengan rotor ke sumbu x_b dan β merupakan sudut yang dibentuk oleh lengan rotor ke sumbu y_b . Sehingga momen gaya quadrotor dapat dituliskan sebagai berikut.

$$\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} (F_{a1} + F_{a4} - F_{a2} - F_{a3})l \sin(\alpha) \\ (F_{a1} + F_{a2} - F_{a3} - F_{a4})l \sin(\beta) \\ (-T_1 - T_3 + T_2 + T_4) \end{bmatrix}$$

$$\begin{bmatrix} I_{xx}\ddot{\phi} \\ I_{yy}\ddot{\theta} \\ I_{zz}\ddot{\psi} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})\dot{\theta}\dot{\psi} \\ (I_{zz} - I_{xx})\dot{\phi}\dot{\psi} \\ (I_{xx} - I_{yy})\dot{\theta}\dot{\phi} \end{bmatrix} + \begin{bmatrix} (F_{a1} + F_{a4} - F_{a2} - F_{a3})l \sin(\alpha) \\ (F_{a1} + F_{a2} - F_{a3} - F_{a4})l \sin(\beta) \\ (-T_1 - T_3 + T_2 + T_4) \end{bmatrix}$$

Sehingga didapatkan representasi dinamika quadrotor dalam persamaan differensial adalah sebagai berikut.

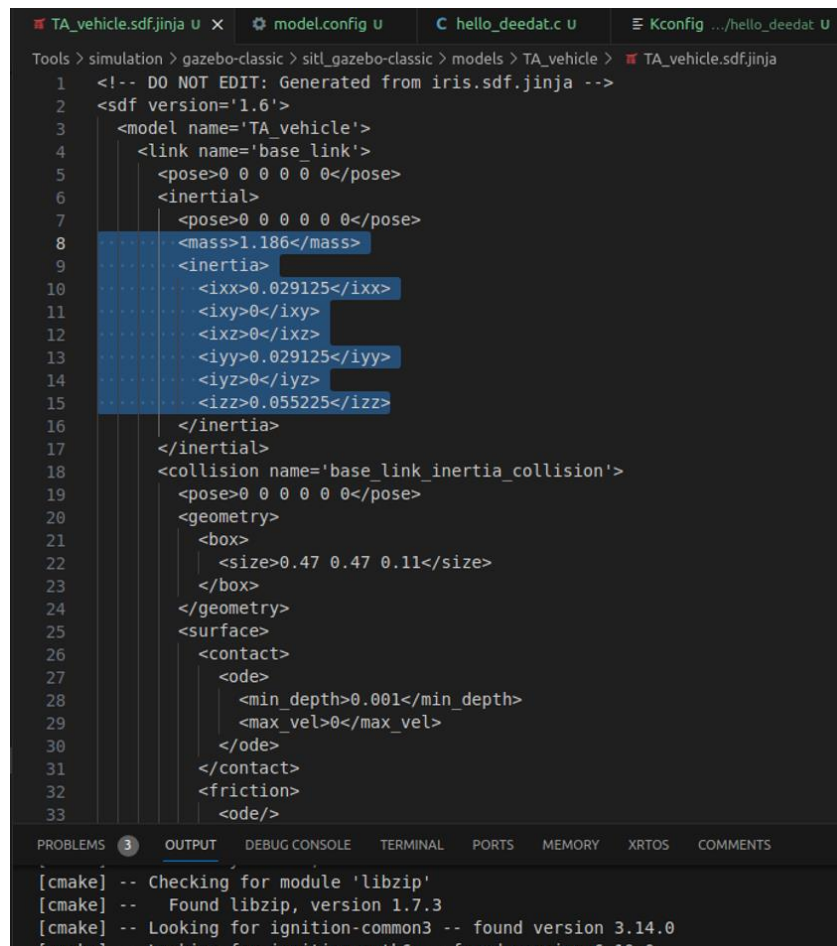
$$\begin{aligned} \dot{\Omega}_{mi} &= -\left(\frac{K_e K_{tm}}{R_m J_m} + \frac{2d\Omega_{m0}}{\eta r_g^3 J_m}\right) \Omega_{mi} + \frac{K_{tm}}{R_m J_m} U_{ai} + \left(\frac{d_m \Omega_{m0}^2}{\eta r_g^3 J_m}\right), i = 1, 2, 3, 4 \\ I_{xx}\ddot{\phi} &= (I_{yy} - I_{zz})\dot{\theta}\dot{\psi} + (\Omega_{m1}^2 + \Omega_{m4}^2 - \Omega_{m2}^2 - \Omega_{m3}^2)l \sin(\alpha) C_a \rho A r^2 \\ I_{yy}\ddot{\theta} &= (I_{zz} - I_{xx})\dot{\phi}\dot{\psi} + (\Omega_{m1}^2 + \Omega_{m2}^2 - \Omega_{m3}^2 - \Omega_{m4}^2)l \sin(\beta) C_a \rho A r^2 \\ I_{zz}\ddot{\psi} &= (I_{xx} - I_{yy})\dot{\theta}\dot{\phi} + C_T \rho A r^3 \sum_{i=1}^4 (-1)^i \Omega_{mi}^2 \\ m\ddot{x} &= (-\cos \phi \sin \theta \cos \psi - \sin \phi \sin \psi) \cdot C_a \rho A r^2 \cdot \sum_{i=1}^4 \Omega_{mi}^2 - \frac{1}{2} C_x A_c \rho \dot{x} |\dot{x}| \\ m\ddot{y} &= (-\cos \phi \sin \theta \sin \psi + \sin \phi \cos \psi) \cdot C_a \rho A r^2 \cdot \sum_{i=1}^4 \Omega_{mi}^2 - \frac{1}{2} C_y A_c \rho \dot{y} |\dot{y}| \\ m\ddot{z} &= -(\cos \phi \cos \theta) \cdot C_a \rho A r^2 \cdot \sum_{i=1}^4 \Omega_{mi}^2 + mg - \frac{1}{2} C_z A_c \rho \dot{z} |\dot{z}| \end{aligned}$$

3.2.2 Studi Simulasi Gazebo

Seperti yang sebelumnya telah dijelaskan, sistem kontrol akan diuji pada lingkungan simulasi berbasis *hardware in the loop* dengan menggunakan aplikasi gazebo. Agar simulasi menghasilkan keluaran yang sesuai dengan model aslinya, maka parameter fisis quadrotor seperti massa, momen inersia, dan konstanta daya motor perlu diukur dan diimplementasi ke dalam simulasi.

Pada saat ini, studi mengenai penerapan parameter dinamika fisis pada gazebo telah dilakukan. Selain itu, telah dilakukan modifikasi program simulasi gazebo yang dapat mengubah parameter model simulasi dengan parameter model drone. Ditemukan bahwa bagian kode yang berperan dalam mengubah massa dan momen

inersia dapat diubah sesuai dengan parameter yang didapatkan melalui pengujian fisis yang akan dilakukan selanjutnya. Parameter fisis quadrotor tersebut dapat diubah pada bagian *TA_vehicle.sdf.jinja* yang dapat dilihat pada gambar. Parameter ini telah berhasil diterapkan pada simulasi secara *software in the loop*, yang dapat dilihat pada Gambar 3.4. Pada Gambar 3.4 juga terlihat bahwa terdapat model bernama *TA_vehicle* yang merupakan model quadrotor yang dimodifikasi untuk penelitian ini.



```

1  <!-- DO NOT EDIT: Generated from iris.sdf.jinja -->
2  <sdf version='1.6'>
3    <model name='TA_vehicle'>
4      <link name='base_link'>
5        <pose>0 0 0 0 0 0</pose>
6        <inertial>
7          <pose>0 0 0 0 0 0</pose>
8          <mass>1.186</mass>
9          <inertia>
10           <ixx>0.029125</ixx>
11           <ixy>0</ixy>
12           <ixz>0</ixz>
13           <iyy>0.029125</iyy>
14           <iyz>0</iyz>
15           <izz>0.055225</izz>
16         </inertia>
17       </link>
18       <collision name='base_link_inertia_collision'>
19         <pose>0 0 0 0 0 0</pose>
20         <geometry>
21           <box>
22             <size>0.47 0.47 0.11</size>
23           </box>
24         </geometry>
25         <surface>
26           <contact>
27             <ode>
28               <min_depth>0.001</min_depth>
29               <max_vel>0</max_vel>
30             </ode>
31           </contact>
32           <friction>
33             <ode/>

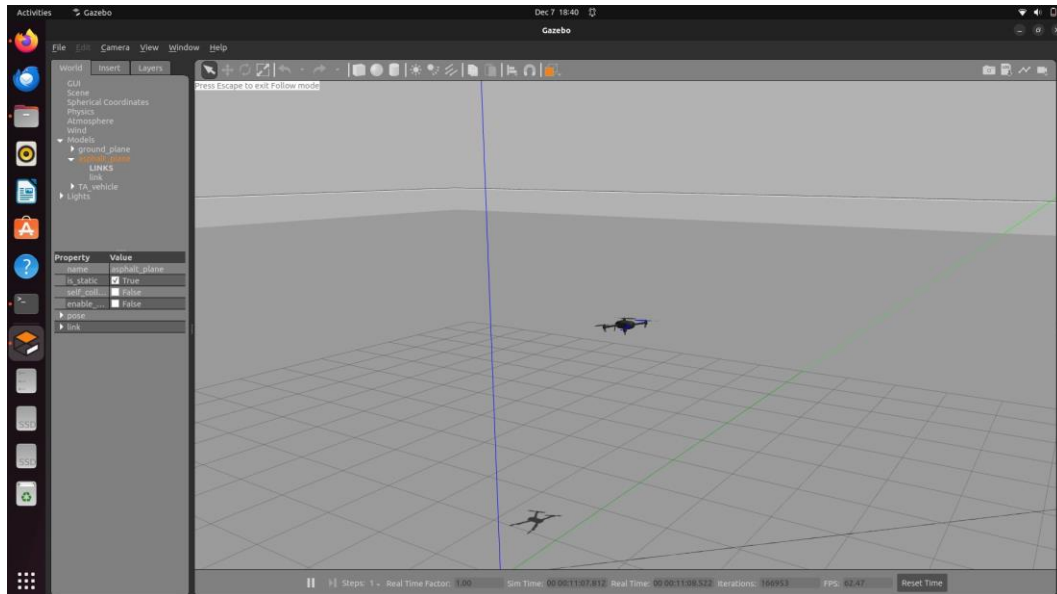
```

```

[cmake] -- Checking for module 'libzip'
[cmake] -- Found libzip, version 1.7.3
[cmake] -- Looking for ignition-common3 -- found version 3.14.0
[cmake] -- Looking for ignition-math6 -- found version 6.10.0

```

Gambar 3.3 Cuplikan Kode Simulasi Model Quadrotor



Gambar 3.4 Simulasi Penerbangan Quadrotor Pada Gazebo

3.2.3 Studi Pemrograman Pixhawk

Bagian ini akan menjelaskan mengenai apa yang telah dilakukan dalam mempelajari cara implementasi pemrograman pada pengontrol pixhawk. Pada penelitian ini, *microcontroller* pixhawk akan diunggah dengan *firmware* PX4 yang nantinya akan dimodifikasi untuk mengimplementasi bentuk pengontrol dalam mengontrol quadrotor.

Pada saat ini, telah dibuat program *hello_deedat* yang bertujuan untuk mempelajari lebih dalam mengenai kerangka pengembangan PX4. Proses pengembangan kode ini akan dilakukan sesuai dengan dokumentasi *firmware* yang sudah disediakan oleh PX4 [15]. Hal pertama yang harus dilakukan ialah membuat folder pada direktori src pada kode sumber *firmware* PX4. Kode yang akan diimplementasi merupakan kode sederhana yang dapat mengeluarkan sebuah kalimat yang akan ditampilkan pada terminal melalui protokol mavlink.

Agar fungsi yang telah dibuat dapat di-*compile* dan dieksekusi, file CMakeLists dan Kconfig perlu dibuat agar *compiler* dapat mengompilasi kode sumber tersebut ke pada PX4 yang nantinya dapat digunakan melalui *console* mavlink. Pada gambar terlihat bahwa fungsi dapat diakses dan digunakan pada terminal dengan mengetik nama fungsi *hello_deedat* pada *console*.

```
fdeedat@fdeedat-IdeaPad-Gaming-3-15ACH6: ~/TA-PX4/TA-PX4
ls.sdf
SITL COMMAND: "/home/fdeedat/TA-PX4/TA-PX4/build/px4_sitl_default/bin/px4" "/home/fdeedat/TA-PX4/TA-PX4/build/px4_sitl_default/etc

PX4

px4 starting.

INFO [px4] startup script: /bin/sh etc/init.d-posix/rcs 0
INFO [init] found model autostart file as SYS_AUTOSTART=10015
INFO [param] selected parameter default file parameters.bson
INFO [param] importing from 'parameters.bson'
INFO [parameters] BSON document size 404 bytes, decoded 404 bytes (INT32:14, FLOAT:6)
INFO [param] selected parameter backup file parameters_backup.bson
INFO [dataman] data manager file './dataman' size is 7868376 bytes
INFO [init] PX4_SITL_HOSTNAME: localhost
INFO [simulator_mavlink] Waiting for simulator to accept connection on TCP port 4560
INFO [simulator_mavlink] Simulator connected on TCP port 4560
INFO [lockstep_scheduler] setting initial absolute time to 3140000 us
INFO [commander] LED: open /dev/led0 failed (22)
WARN [health_and_arwing_checks] Preflight Fail: ekf2 missing data
Gazebo multi-robot simulator, version 11.10.2
Copyright (c) 2012 Open Source Robotics Foundation.
Released under the Apache 2 License.
http://gazebo.in.org

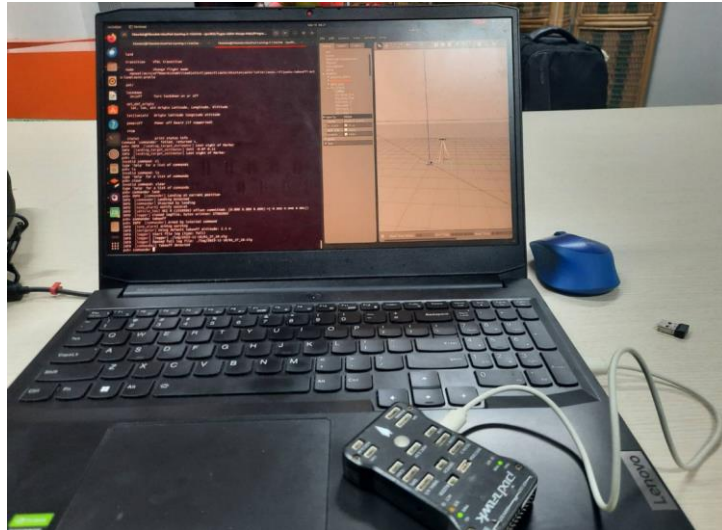
[Msg] Waiting for master.
[Msg] Connected to gazebo master @ http://127.0.0.1:11345
[Msg] Publicized address: 192.168.100.18
INFO [mavlink] mode: Normal, data rate: 4000000 B/s on udp port 18570 remote port 14550
INFO [mavlink] mode: Onboard, data rate: 4000000 B/s on udp port 14580 remote port 14540
INFO [mavlink] mode: Onboard, data rate: 4000 B/s on udp port 14280 remote port 14030
INFO [mavlink] mode: GInbal, data rate: 400000 B/s on udp port 13030 remote port 13280
INFO [logger] logger started (mode=all)
INFO [logger] Start file log (type: full)
INFO [logger] [logger] ./log/2023-12-07/12_23_52.ulg
INFO [logger] Opened full log file: ./log/2023-12-07/12_23_52.ulg
INFO [mavlink] MAVLink only on localhost (set param MAV_{t}_BROADCAST = 1 to enable network)
INFO [mavlink] MAVLink only on localhost (set param MAV_{t}_BROADCAST = 1 to enable network)
INFO [px4] Startup script returned successfully
pxh> [Msg] Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland to run on
Wayland anyway.
INFO [tone_alarm] home set
INFO [tone_alarm] notify negative
[Warn] [Event.cc:61] Warning: Deleting a connection right after creation. Make sure to save the Connec
tionPer from a Connect call
INFO [commander] Ready for takeoff!
pxh> hello_deedat
INFO [hello_deedat] Selamat Terbang Deedat!!!
pxh>
```

Gambar 3.5 Hasil Implementasi Program Pada Console

3.2.4 Simulasi *Hardware in the Loop*

Seperti dengan judul penelitian ini, simulasi *hardware in the loop* merupakan salah satu aspek yang penting dalam proses pengujian sistem kontrol yang akan didesain. Pada saat ini, simulasi dasar *hardware in the loop* telah dilakukan dengan mengunggah *firmware* PX4 kepada mikrokontroler pixhawk secara langsung yang akan dipakai sebagai perangkat keras simulasi *hardware in the loop*.

Hal pertama yang dilakukan dalam melaksanakan simulasi *hardware in the loop* ini, ialah mengunggah *firmware* PX4 ke pada pixhawk dengan menggunakan aplikasi QGroundControl. Pada saat mengunggah *firmware* PX4, perhatikan untuk memilih pilihan HIL Quadcopter sebagai *airframe* dari pixhawk. Setelah *firmware* PX4 telah diunggah, simulasi *hardware in the loop* dapat dilakukan dengan mengikuti instruksi yang sudah disediakan pada dokumentasi pixhawk [16]. Ilustrasi dari simulasi ini dapat dilihat pada Gambar 3.6.



Gambar 3.6 Simulasi *Hardware in the Loop*

Simulasi telah dilakukan dengan baik, namun model quadrotor dengan parameter quadrotor nyata belum dapat diimplementasikan. Sehingga, studi lebih lanjut mengenai bagaimana mengimplementasi model simulasi quadrotor dengan parameter yang dapat diubah perlu untuk ditelusuri lebih lanjut.

BAB III

PENUTUP

4.1 Rencana Selanjutnya

Dari kemajuan yang telah dipaparkan sebelumnya, rencana selanjutnya dari riset ini dapat dirangkum seperti berikut:

1. Mengambil data parameter dinamika sistem quadrotor nyata.
2. Menerapkan parameter dinamika sistem quadrotor yang diperoleh kepada simulasi gazebo.
3. Studi lebih lanjut mengenai kemungkinan sistem kontrol yang akan diterapkan.
4. Studi lebih lanjut mengenai cara mengimplemntasi sistem kontrol pada pixhawk.

4.2 Penutup

Demikian laporan kemajuan tugas akhir dengan judul “Perancangan Sistem Kontrol Quadrotor Lincah Berbasis Simulasi Hardware In The Loop “ telah dibuat dengan sebaik-baiknya. Semoga laporan kemajuan tugas akhir ini dapat diterima dengan baik dan penelitian tugas akhir ini dapat berjalan dengan lancar.

DAFTAR PUSTAKA

- [1] M. Deja, M. S. Siemitkowski, G. C. Vosniakos, dan G. Maltezos, “Opportunities and challenges for exploiting drones in agile manufacturing systems,” dalam *Procedia Manufacturing*, Elsevier B.V., 2020, hlm. 527–534. doi: 10.1016/j.promfg.2020.10.074.
- [2] J. Verbeke dan J. De Schutter, “Experimental maneuverability and agility quantification for rotary unmanned aerial vehicle,” *International Journal of Micro Air Vehicles*, vol. 10, no. 1. SAGE Publications Inc., hlm. 3–11, 1 Maret 2018. doi: 10.1177/1756829317736204.
- [3] E. T. Efaz, M. M. Mowlee, J. Jabin, I. Khan, dan M. R. Islam, “Modeling of a high-speed and cost-effective FPV quadcopter for surveillance,” dalam *ICCIT 2020 - 23rd International Conference on Computer and Information Technology, Proceedings*, Institute of Electrical and Electronics Engineers Inc., Des 2020. doi: 10.1109/ICCIT51783.2020.9392696.
- [4] “https://docs.px4.io/main/en/flight_stack/controller_diagrams.html,” Controller Diagram PX4 .
- [5] M. W. Mueller dan R. D’Andrea, *A Model Predictive Controller For Quadcopter State Interception*. 2013.
- [6] A. Surriani, un Budiyanto, dan M. Arrofiq, “Altitude Control of Quadrotor using Fuzzy Self Tuning PID Controller,” 2015.
- [7] K. D. Nguyen dan C. Ha, “Development of Hardware-in-the-Loop Simulation Based on Gazebo and Pixhawk for Unmanned Aerial Vehicles,” *International Journal of Aeronautical and Space Sciences*, vol. 19, no. 1, hlm. 238–249, Mar 2018, doi: 10.1007/s42405-018-0012-8.
- [8] W. Smit, “Design and Implementation of Model Predictive Control on Pixhawk Flight Controller,” 2018. [Daring]. Tersedia pada: <https://scholar.sun.ac.za>
- [9] D. Falanga, E. Mueggler, M. Faessler, dan D. Scaramuzza, “Aggressive Quadrotor Flight through Narrow Gaps with Onboard Sensing and Computing.” [Daring]. Tersedia pada: <https://www.youtube.com/watch?v=meSItatXQ7M>
- [10] Y. Fan, “Flight Control System Simulation for Quadcopter Unmanned Aerial Vehicle (UAV) based on Matlab Simulink,” dalam *Journal of Physics: Conference Series*, Institute of Physics, 2022. doi: 10.1088/1742-6596/2283/1/012011.
- [11] P. Burggräf, A. R. Pérez Martínez, H. Roth, dan J. Wagner, “Quadrotors in factory applications: design and implementation of the quadrotor’s P-PID

- cascade control system: Modeling and implementation,” *SN Appl Sci*, vol. 1, no. 7, Jul 2019, doi: 10.1007/s42452-019-0698-7.
- [12] K. M. Thu dan A. I. Gavrilov, “Designing and Modeling of Quadcopter Control System Using L1 Adaptive Control,” dalam *Procedia Computer Science*, Elsevier B.V., 2017, hlm. 528–535. doi: 10.1016/j.procs.2017.01.046.
 - [13] “PX4 System Architecture | PX4 User Guide.” Diakses: 8 Desember 2023. [Daring]. Tersedia pada: https://docs.px4.io/main/en/concept/px4_systems_architecture.html
 - [14] “PX4 Architectural Overview .” Diakses: 8 Desember 2023. [Daring]. Tersedia pada: <https://docs.px4.io/main/en/concept/architecture.html>
 - [15] “Writing your First Application using PX4 .” Diakses: 7 Desember 2023. [Daring]. Tersedia pada: https://docs.px4.io/main/en/modules/hello_sky.html
 - [16] “Hardware in the Loop Simulation (HITL) | PX4 User Guide.” Diakses: 8 Desember 2023. [Daring]. Tersedia pada: <https://docs.px4.io/main/en/simulation/hitl.html>