

Rancang Bangun Simulasi dan Pengendali *Unmanned Aerial Vehicle* (UAV)

Rizki Wahyu Pratama ¹⁾, Ferry Hadary ²⁾, Redi Ratiandi Yacoub³⁾
 Jurusan Teknik Elektro Fakultas Teknik Universitas Tanjungpura
 e-mail: rizki633469@gmail.com ¹⁾, ferry.hadary@ee.untan.ac.id ²⁾, rediyacoub@yahoo.com ³⁾

Abstrak—Pada penelitian ini merancang simulasi dan pengendali *Unmanned aerial vehicle* (UAV) untuk menganalisis keoptimalan sistem. Software matlab digunakan untuk merancang simulasi dan pengendali UAV.

Kata kunci: Matlab, quadcopter

I. LATAR BELAKANG

Perancangan simulasi dan pengendali UAV dalam proses perancangan merupakan hal yang penting dalam pembuatan suatu alat. Sebelum membangun suatu alat disarankan untuk membuat perancangan simulasi dan pengendalinya terlebih dahulu agar sistem yang dibangun optimal.

II. KONSEP DASAR PENELITIAN

A. Penelitian Terkait

Metode yang digunakan dalam penelitian ini adalah membuat simulasi dan pengendali UAV.

B. *Unmanned aerial vehicle* (UAV)

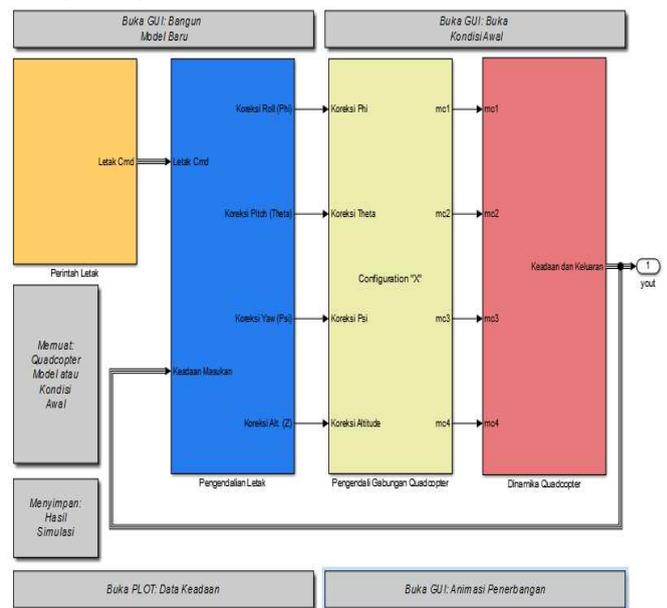
Unmanned aerial vehicle (UAV) merupakan sebuah mesin terbang yang dapat dikendalikan dengan pengendali jarak jauh atau dengan bantuan perangkat yang telah diprogram sesuai dengan perintah yang diberikan kepada UAV, sehingga UAV dapat menjalankan misinya secara otomatis tanpa pilot (*autopilot*). Penggunaan UAV sekarang ini sangat luas digunakan misalnya untuk pencarian korban bencana alam, penginderaan jauh, mengawasi kawasan perhutanan, daerah perbatasan, dan sebagai alat pemetaan serta pengawasan pada suatu wilayah.

III. PERANCANGAN

A. Perancangan Simulasi dan Pengendali

Simulink berbasis pada simulasi *quadcopter* yang dimaksudkan adalah untuk membangun dan menjalankan *quadcopter* dengan jenis simulasi dari kinerja dinamik *quadcopter*. Simulasi ini berfungsi sebagai data awal untuk membantu menganalisa dan mengembangkan teknik bentuk pengendali *quadcopter*. Simulasi menggambarkan perpaduan dari beberapa pendekatan untuk memodelkan tingkah laku dari *quadcopter* dengan sedikit model tambahan yang ditambahkan. Banyak pengaruh penting (pengaruh aerodinamis, putaran baling-baling) diabaikan atau disederhanakan. Pembuatan simulink ini menggunakan *guide* pada Matlab untuk membuat

simulasi dan pengendali *quadcopter*. Simulink *quadcopter* ditampilkan pada Gambar 1.



Gambar 1. Tampilan GUI Simulink *Quadcopter*

IV. PENGUJIAN DAN ANALISIS SISTEM

Dalam pembahasan dibahas hubungan antara *thrust*, *torque*, dan *TC* (waktu konstan) *step response*. Pembahasan menggunakan variabel *seconds*, *throttle*, RPM, dan *gramsMeas* yang diperoleh dari data pengujian. Setelah semua variabel utama dan pendukung didapatkan, data dianalisa dengan memplotkan hubungan *throttle*, *thrust*, RPM dan *torque* berdasarkan masukan data pada skript program Matlab.

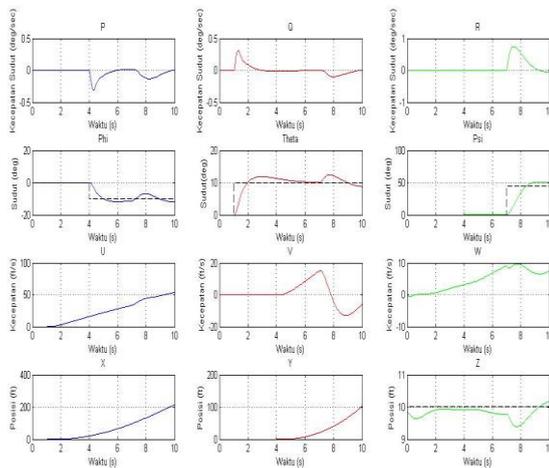
A. Pengujian GUI *Quadcopter*

Pada pengujian ini digunakan simulink berbasis pada simulasi *quadcopter* yang bertugas untuk membangun dan menjalankan simulasi dari kinerja dinamis *quadcopter*. Masukan yang digunakan untuk mendapatkan data plot dan simulasi GUI dalam simulink adalah data kondisi awal dan data model *quadcopter* yang digunakan. Data yang dimasukkan ke dalam GUI berdasarkan pada data pengujian yang dilakukan. Tujuan dari pemodelan *quadcopter* adalah untuk menginisialisasi model *quadcopter* yang digunakan dalam simulasi untuk diproses dengan simulink. Data kondisi awal merupakan data awal yang diperlukan untuk mensimulasikan pergerakan *quadcopter* berdasarkan kecepatan sudut, sudut euler, kecepatan motor, kecepatan translasi, dan posisi dalam kerangka tanah. Tujuan dari memasukkan data kondisi awal

adalah sebagai data awal yang digunakan dalam simulink untuk mensimulasikan pergerakan *quadcopter* yang dirancang. Data kondisi awal digunakan sebagai referensi pergerakan *quadcopter* berdasarkan parameter model *quadcopter* dan kondisi awal sehingga *quadcopter* dapat disimulasikan sesuai dengan masukan yang diberikan.

Pada pengujian ini, data yang dimasukkan untuk pemodelan *quadcopter* dan kondisi awal GUI adalah data momen inersia, data tes motor, dan data kondisi awal *quadcopter*.

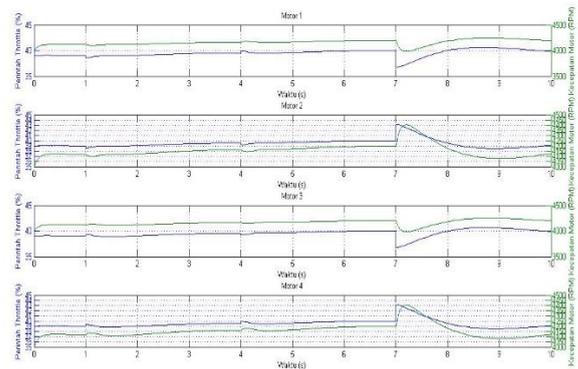
Setelah data model *quadcopter* dan kondisi awal dimasukkan ke dalam simulink, kemudian tampilkan plot data keadaan dan GUI animasi penerbangan. Pada data plot terdapat dua plot yang ditampilkan. Plot pertama adalah hubungan antara P, Q, R, Phi, Theta, Psi, U, V, W, dan X, Y, Z. Pada plot kedua menggambarkan *response* yang terjadi pada motor 1, 2, 3 dan 4. Data pada plot pertama ditampilkan pada Gambar 2 dan data plot kedua ditampilkan pada Gambar 3.



Gambar 2. Data Plot Simulink Pergerakan *Quadcopter*.

Pada Gambar 2, hubungan kecepatan sudut (deg/sec), sudut (deg), kecepatan, (ft/s), dan posisi (ft) terhadap waktu (s) juga menampilkan *response*. *Response* yang ditampilkan didapat berdasarkan hubungan antara RPM motor yang diberikan terhadap posisi yang ditentukan dan pengaruh luar yang mempengaruhi kinerja *quadcopter* sehingga *response* sistem ditampilkan pada Gambar 2. Dalam analisa ini, yang dianalisa hanya hubungan antara posisi dan waktu pada bidang Z. Untuk hubungan kecepatan sudut (deg/sec), sudut (deg), dan kecepatan, (ft/s), terhadap waktu (s) tidak dianalisa, karena masukan data kondisi awal yang diberikan hanya pada masukan data posisi dan RPM sehingga hanya menganalisa hubungan antara posisi terhadap waktu.

Setelah mendapatkan data pada plot pertama, kemudian diperoleh data pada plot kedua. Plot kedua ini merupakan *response* yang terjadi pada motor 1, 2, 3, dan 4. Data pada plot kedua ditampilkan pada Gambar 3.

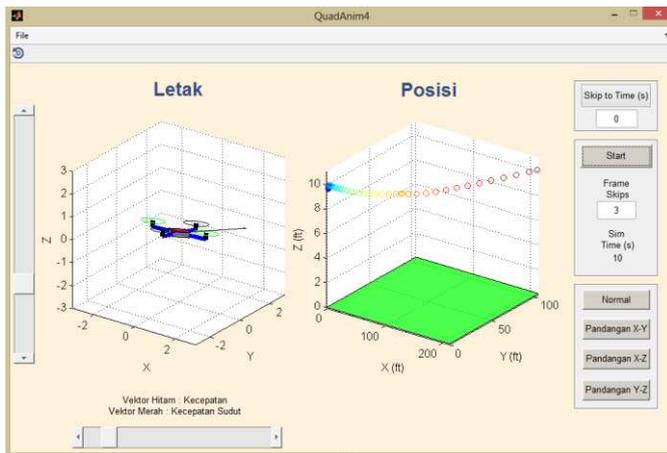


Gambar 3. Data Plot Simulink Motor yang Bekerja pada *Quadcopter*

Pada Gambar 3 motor mengalami *response transient underdamped response*, hal ini mengakibatkan *response* pada motor mengalami osilasi pada waktu tertentu. Pada *response* motor 1, 2, 3, dan 4 menggambarkan *output* melesat naik untuk mencapai *input* kemudian turun dan berhenti pada kisaran nilai *input* dalam keadaan *steady state*. Pada motor 1 dan 3, terjadi penurunan *response* pada waktu ke 7,022 seconds dengan RPM sebesar 3686 dan perintah *throttle* pada 4,076%. Penurunan *response* terjadi disebabkan oleh beberapa faktor luar dan dalam. Berdasarkan data ini, *overshoot* pada motor 1 dan 3 untuk data perintah *throttle* pada 0,11 % dan data kecepatan motor sebesar 5223 RPM. *Overshoot* merupakan nilai relatif yang menyatakan perbandingan harga maksimum *response* yang melampaui nilai *steady state* dibandingkan dengan nilai *steady state*. *Overshoot* ini terjadi disebabkan karena *response* berusaha untuk mencapai nilai *steady state* sehingga terjadi lonjakan yang mengakibatkan *response* mengalami osilasi hingga mencapai keadaan *steady state*. Berdasarkan Gambar 3, *response* perintah *throttle* (%) dan kecepatan motor (RPM) mencapai *steady state* adalah pada waktu ke 7,755 seconds, dengan kecepatan motor pada saat mencapai 4963 RPM dan perintah *throttle* sebesar 7,297 (%). Keadaan *steady state* dari *response* ini menyatakan bahwa setelah *response* mengalami *response transient underdamped response*, sistem berusaha menyesuaikan *output* berdasarkan *input* yang diberikan hingga mencapai keadaan *steady state*. Pada motor 2 dan 4, terjadi kenaikan *response* pada waktu ke 7,022 seconds dengan RPM sebesar 5074 dan perintah *throttle* pada 10,36 %. Kenaikan *response* terjadi disebabkan oleh beberapa faktor luar dan dalam. Berdasarkan *response* perintah *throttle* (%) dan kecepatan motor (RPM) mencapai *steady state* adalah pada waktu ke 7,695 seconds, dengan kecepatan motor pada saat mencapai 4944 RPM dan perintah *throttle* sebesar 7,323 (%). Keadaan *steady state* dari *response* ini menyatakan bahwa setelah *response* mengalami *response transient underdamped response*, sistem berusaha menyesuaikan *output* berdasarkan *input* yang diberikan hingga mencapai keadaan *steady state*.

B. Pengujian Alat Simulasi GUI

Pengujian ini dilakukan berdasarkan data masukan kondisi awal dan pemodelan *quadcopter* yang digunakan. Pengujian ini berfungsi untuk mengetahui pergerakan *quadcopter* yang disimulasikan sesuai dengan masukan kondisi awal dan pemodelan *quadcopter* yang digunakan. Simulasi *quadcopter* yang digunakan ditampilkan pada Gambar 4.



Gambar 4. Simulasi GUI

Gambar 4 menggambarkan pergerakan *quadcopter* berdasarkan masukan data yang diberikan. Hasil simulasi dari pengujian ini menggambarkan bahwa *quadcopter* melayang pada ketinggian 3 meter dengan kecepatan motor sebesar 4000 RPM, kemudian *quadcopter* ini bergerak dan menyimpang. Berdasarkan simulasi pada Gambar 4, pada waktu ke 7 seconds, *quadcopter* pada simulasi melakukan manuver menyimpang. Menyimpangnya *quadcopter* pada waktu ke 7 seconds disebabkan oleh faktor gangguan yang dirancang pada simulink. Gangguan yang dirancang pada simulink ini berdasarkan pada data *torque* dan gaya pada dimensi *quadcopter* yang disesuaikan dengan kondisi sebenarnya di lapangan.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari keseluruhan pembahasan dalam penelitian ini dapat disimpulkan:

1. Pengujian hubungan RPM, *Throttle*, *Trust*, *Torque*, dan *TC Step Respon*s menggambarkan hubungan masing-masing pengujian yang diplotkan. Dalam pengujian ini dilihat hubungan masing-masing data pengujian yang digunakan untuk mengetahui kemampuan motor sehingga dapat diketahui

spesifikasi dan kebutuhan beban yang diperlukan untuk *quadcopter* bekerja dengan baik.

2. Pengujian GUI *quadcopter* pada pengujian ini digambarkan dengan simulasi dan pergerakan *quadcopter* yang selanjutnya dibaca pergerakan dan simulasi dari GUI, sehingga dalam implementasi nyatanya dapat diperkirakan pergerakan *quadcopter* berdasarkan masukan data yang diberikan pada GUI.

B. Saran

Saran untuk pengembangan Rancang Bangun Simulasi dan Pengendali Unmanned Aerial Vehicle (UAV) ini adalah:

1. Membangun simulasi sistem terlebih dahulu sebelum membangun alat.
2. Menguji simulasi dan pengendali hingga mencapai kondisi optimal agar hasil yang diinginkan sesuai.

DAFTAR PUSTAKA

- [1] Atmel Corporation. 2012, September 04. *8-bit AVR microcontroller with 8/16/32K Bytes of ISP Flash and USB Controller*. May 05, 2015. <http://www.atmel.com>
- [2] Atmel Corporation. 2012, October 04. *8-bit AVR microcontroller with 64K/128K/256K Bytes In-System Programmable Flash*. May 05, 2015. <http://www.atmel.com>
- [3] DIYDrones Development Team, 2015. Arduino-based autopilot for multicopter craft, from quadcopters to multicopter, (Online), (<http://copter.ardupilot.com/>), diakses May 2015.

Biografi