

# DESAIN DAN SIMULASI SISTEM KONTROL PID PADA PENGENDALIAN KECEPATAN MOTOR PENGGERAK QUADCOPTER

Adi Mulyadi<sup>\*)</sup>, Wijono, Bambang Siswojo

Jurusan Teknik Elektro, Universitas Brawijaya Malang, Indonesia  
Jalan Jl. Veteran, Malang 65145 INDONESIA

<sup>\*)</sup> E-mail: [adimulyadi.mt@gmail.com](mailto:adimulyadi.mt@gmail.com)

## Abstrak

Pergerakan *quadcopter* dipengaruhi oleh kecepatan empat baling-baling yang digerakkan oleh motor BLDC. Perputaran baling-baling tersebut akan menghasilkan gaya dorong yang arahnya *vertical*. Pada pengoperasiannya, dua motor dikendalikan searah jarum jam, dan dua motor lainnya berlawanan jarum jam. Permasalahan yang terjadi pada pembangkitan gaya angkat *quadcopter* adalah bahwa kecepatan empat motor harus sama, sehingga gerakan *quadcopter* dapat mencapai ketinggian yang diinginkan. Makalah ini mendiskusikan mengenai desain sistem kontrol kecepatan motor untuk mendapatkan kestabilan dari gerakan *altitude vertical take-off quadcopter*. Sistem kontrol melibatkan beban motor dan kecepatan gaya angkat. Perhitungan *vertical take-off* dilakukan dengan menggunakan metode Euler-Newton. Metoda *Proportional-Integral-Derivative* (PID) diusulkan untuk mendapatkan kestabilan gerakan. Selain itu, simulasi menggunakan *MATLAB Simulink* digunakan untuk memvalidasi hasil perancangan. Hasil simulasi menunjukkan bahwa dengan kontrol PID pada ketinggian 10 m sampai 90 m didapatkan respon waktu naik lebih cepat 0.01 detik, *overshoot* 0%, dan waktu *steady state* 0.06 detik. Sedangkan pengaturan PI pada ketinggian 10 m sampai 90 m menghasilkan respon waktu naik lebih lama 0.013 detik, *overshoot* 0% dan waktu *steady state* 0.1 detik.

*Kata kunci: motor brushless dc, quadcopter, proportional-integral-derivative (PID) controller, Euler-Newton.*

## Abstract

*Quadcopter movement is a function the speed of four propellers driven by a BLDC motor. The rotation of the propeller will produce a vertical thrust. In the quadcopter operation, two motors are controlled clockwise, and two other motors are counterclockwise. The problem that occurs in the generation of a quadcopter lift force is that the speed of the four motors must be the same, so that the quadcopter's movement can reach the targeted height. This paper discusses on the design of a motor speed control system, especially on getting its stability of the altitude vertical take-off of the quadcopter. The control system involves the motor loading and the lift speed. The vertical take-off calculations are performed using the Euler-Newton method. Proportional Integral Derivative (PID) method is proposed to get the movement stability. In addition, simulations using MATLAB Simulink are used to validate the design results. The simulation results show that using PID control method for an altitude of 10 m to 90 m, it yields the rise time response of 0.01 seconds, the overshoot of 0%, and the steady state time of 0.06 seconds. Whereas using PI method, the control system produces a longer rise time response of 0.013 seconds, the overshoot of 0%, and a longer steady state time of 0.1 seconds.*

*Keywords: brushless dc motor, quadcopter, proportional-integral-derivative (PID) controller, Euler-Newton.*

## 1. Pendahuluan

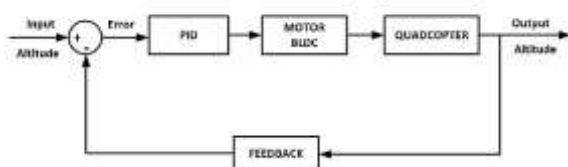
Pergerakan *quadcopter* dipengaruhi oleh kecepatan empat baling-baling yang digerakkan oleh motor BLDC. Perputaran baling-baling tersebut akan menghasilkan gaya dorong yang arahnya *vertical*. *Quadcopter* digerakkan dengan empat buah motor dan *propeller*, dimana setiap putaran *propeller* diatur dengan *clockwise* (CW) dan *counter clockwise* (CCW) untuk menghasilkan torsi dan *thrust force* [1]. Permasalahan yang terjadi pada gaya angkat *quadcopter* adalah kecepatan empat motor harus diatur dengan kecepatan yang sama, sehingga

gerakan *quadcopter* dapat mencapai ketinggian yang diinginkan [2]. Desain sistem kontrol kecepatan motor digunakan untuk mengendalikan kestabilan gerakan *altitude vertical take-off quadcopter*. Desain kontrol ditentukan berdasarkan beban dan gaya angkat [3]. *Vertical Take-off* merupakan persamaan untuk pengaturan gaya angkat dengan memperhatikan massa *body frame*. dan pengaturan dapat diperoleh dengan persamaan Euler-Newton. Model sistem berdasarkan hubungan antara *thrust* empat motor dan kecepatan sudut. Kontrol kecepatan motor menentukan stabilitas *altitude vertical take-off landing quadcopter* [4]. Euler-Newton digunakan untuk

memodelkan *body frame*. Pemodelan berdasarkan *E-frame* dan *B-Frame*, sehingga *E-frame* dan *B-Frame* bisa diatur dengan mudah pada setiap sudut ( $x,y,z$ ). *Controller* menghasilkan gerakan *lift force* secara *vertical*, yang mana *body frame* dapat bergerak dengan posisi (*roll*, *pitch* dan *yaw*) [5]. *Controller* dibagi menjadi tiga jenis yaitu *take-off*, navigasi dan *landing*. Pengaturan *take-off* digunakan untuk mengatur *thrust force* dari tanah sampai melayang di udara, dan *take-off* harus menjamin ketinggian dalam kondisi yang konsisten. Pengaturan *landing* membutuhkan keamanan pada saat turun ke tanah, karena *landing* dipengaruhi oleh kecepatan motor, sehingga *quadcopter* tidak mengalami penurunan secara drastis. Pengaturan navigasi membutuhkan *trajectory tracking* agar *quadcopter* dapat terbang sesuai arah yang sudah ditentukan [6]. Dalam prakteknya, *trajectory tracking* membutuhkan kestabilan untuk gaya angkat *vertical take-off landing*. oleh sebab itu pengaturan *altitude* dibutuhkan untuk mempertahankan stabilitas gerakan yang diinginkan [7]. Selain gerakan *vertikal*, *maneuver* juga diperhatikan dalam rotasi setiap sudut. *Maneuver* digerakkan oleh *independent* empat motor dengan kecepatan yang berbeda. Pergerakan *roll* memerlukan penurunan kecepatan dua motor, sehingga *quadcopter* bergerak rotasi ke kanan atau ke kiri. Pergerakan *pitch* mengurangi dua kecepatan motor (belakang) untuk maju dan mengurangi dua kecepatan motor (depan) untuk mundur. Pergerakan *yaw* menambahkan kecepatan motor 1, 3 untuk rotasi ke kanan, dan menambah kecepatan motor 2, 4 untuk rotasi ke kiri [8]. Penelitian ini mendiskusikan tentang desain sistem kontrol kecepatan motor penggerak *quadcopter*. Kontrol dirancang untuk mengatur kestabilan gerakan *altitude vertical take-off* dengan memperhatikan beban dan gaya angkat. *Vertical take-off* diperoleh dengan metode Euler Newton. Pengaturan PID diusulkan untuk mengontrol gerakan secara vertikal. Kemudian desain kontrol disimulasi dan divalidasi dengan MATLAB Simulink.

## 2. Metode

Metode PID yang diusulkan untuk mengatur kecepatan motor dan kestabilan gerakan *altitude vertical take-off quadcopter* dijelaskan pada gambar 1 [9].

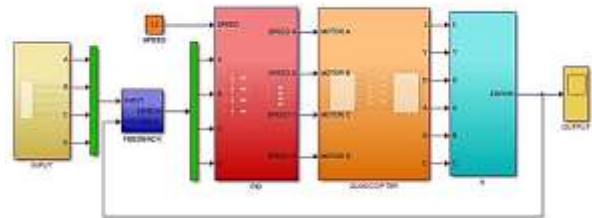


Gambar 1. *Proposed* Kontrol PID [9].

Gambar 1 menjelaskan kontrol PID pada kecepatan motor *quadcopter*. Kontrol PID mengoreksi nilai kesalahan antara *input* kecepatan dengan *output* kecepatan yang

diukur untuk mendapatkan respon kecepatan yang diinginkan. Jika nilai *input* kecepatan dan *output* kecepatan tidak sama, kemudian diumpun balikkan dan kontrol PID mengatur agar *input* dan *output* sama [9].

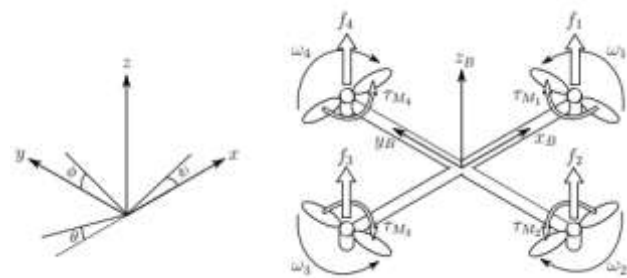
Gambar 2 merupakan model sistem *Simulink quadcopter* dimana *input* terdiri dari gerakan translasi dan rotasi pada sumbu *roll*, *pitch*, *yaw*. Gerakan rotasi *roll* digunakan pada gerakan maju mundur dengan mengurangi kecepatan dua motor bagian belakang atau depan. Gerakan *pitch* digunakan sebagai gerakan rotasi pada posisi ke kiri atau ke kanan dengan mengurangi kecepatan motor bagian kanan atau kiri. Gerakan *yaw* digunakan pada gerakan putaran ke kanan atau ke kiri, dimana putaran ke kanan dengan mengurangi kecepatan motor 1 dan 3, sedangkan putaran ke kiri mengurangi kecepatan motor 2 dan 4.



Gambar 2. Model Simulink *Quadcopter*.

### 2.1. Model Matematis *Quadcopter*

*Quadcopter* digerakkan dengan empat motor dan *propeller*. Pergerakan rotasi dan translasi diperoleh dengan menaikkan atau menurunkan *thrust force* dengan mempertahankan kecepatan motor. Pergerakan maju, mundur dan kiri, kanan diperoleh dengan perbedaan pengaturan *thrust* setiap motor. Pengaturan putaran kiri dan kanan setiap motor diatur dengan *clockwise* dan *counter clockwise*. Pegerakan maju dan mundur diatur dengan menaikkan dua bagian depan dan dua bagian belakang *thrust rotor* [10].

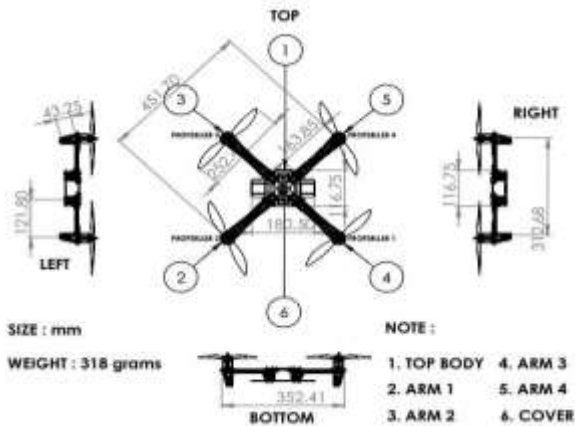


Gambar 3. Kerangka Inersia *Quadcopter* [10].

Gambar 3 menjelaskan kerangka inersia *quadcopter* yang diberikan gaya sama dengan nol atau tidak bergerak, dimana  $f_1$  adalah motor 1,  $f_2$  adalah motor 2,  $f_3$  motor 3, dan  $f_4$  adalah motor 4. Kerangka inersia dijelaskan sebagai

gerakan translasi dan rotasi *quadcopter* pada sumbu x,y,z [10].

*Quadcopter* terdiri dari beberapa bagian seperti *top body*, *cover*, *arms*, *motors* dan *propeller*. Desain dirancang untuk mengetahui massa *quadcopter* dan menentukan gaya angkat secara *altitude vertical take-off*. Total massa diperoleh 318 grams. desain ditunjukkan pada gambar 4.



Gambar 4. Desain *Quadcopter*.

Gambar 4 merupakan desain *quadcopter* yang diwakili sebagai kinematika *body frame* dan *inertial frame*. Kinematika *body frame* dan *inertial frame* menjelaskan sebagai gerakan posisi dan kecepatan *quadcopter*. *Inertial frame*  $\dot{x} = (\dot{x}_a, \dot{y}_a, \dot{z}_a)^T$  digunakan sebagai turunan dari gerakan translasi *quadcopter* pada sumbu (x,y,z). Gerakan *quadcopter* dibagi dua yaitu translasi pada sumbu x,y,z dan gerakan rotasi sudut roll, pitch dan yaw dari *body frame* adalah  $\theta = (\theta_a, \phi_a, \psi_a)$ . Kecepatan sudut yang sesuai dengan  $\dot{\theta} = (\dot{\theta}_a, \dot{\phi}_a, \dot{\psi}_a)$  disebut vektor kecepatan sudut  $\omega \neq \dot{\theta}$  dan kecepatan sudut adalah vektor menunjuk pada sumbu rotasi. Sedangkan  $\dot{\theta}$  merupakan turunan roll, pitch, dan yaw. Untuk mengubah kecepatan sudut menjadi vektor kecepatan sudut menggunakan persamaan berikut [11].

$$\omega = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix} \dot{\theta} \quad (1)$$

Dimana  $\omega$  vektor kecepatan sudut dalam *body frame*. Hubungan *body frame* dan *inertial frame* dengan matrik rotasi R. Matrik diturunkan dengan menggunakan konversi sudut ZYZ Euler dan s adalah sinus dan c adalah cosinus [11].

$$R = \begin{bmatrix} c_\phi c_\psi - c_\phi s_\phi s_\psi & -c_\psi s_\phi - c_\phi c_\theta s_\psi & s_\theta s_\psi \\ s_\theta c_\psi s_\phi + c_\phi s_\phi s_\psi & c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\psi s_\theta \\ s_\phi s_\theta & c_\phi s_\theta & c_\theta \end{bmatrix} \quad (2)$$

## 2.2. Fisik *Quadcopter*

Pemodelan sistem dinamis membutuhkan pemahaman tentang sifat fisik *quadcopter*. Penggunaan motor dan energi *quadcopter* dipertimbangkan untuk memperoleh force dan thrust. Semua motor pada *quadcopter* merupakan identik dan *propeller* beorientasi satu sama lain. Jika *propeller* berputar searah jarum jam, kemudian dua *propeller* yang berdekatan akan berputar berlawanan arah jarum jam. Sehingga torsi seimbang dan *propeller* berputar dengan kecepatan yang sama [11].

## 2.3. Persamaan Gerakan *Quadcopter*

Gerakan *quadcopter* disebabkan oleh thrust force, gravitasi dan gesekan linear. Vektor thrust pada *inertial frame* diperoleh dengan matrik rotasi R untuk memetakan vektor thrust dari *body frame* terhadap *inertial frame*. Oleh sebab itu gerak linear dapat dilihat dalam persamaan (3) [11].

$$m\ddot{x} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + RT_B + F_D \quad (3)$$

Dimana  $\vec{x}$  merupakan posisi *quadcopter*, g kecepatan diakibatkan gravitasi,  $F_D$  drag force,  $T_B$  merupakan vektor thrust dalam *body frame*. Persamaan gerak linear dalam *inertial frame*, dan persamaan gerak rotasi digunakan untuk gerakan *quadcopter*. Persamaan gerak rotasi rigid *body frame* diturunkan dengan persamaan Euler [11].

$$I\dot{\omega} + \omega \times (I\omega) = \tau \quad (4)$$

Dimana  $\omega$  vektor mewakili kecepatan sudut, I merupakan matrix inersia, dan  $\tau$  merupakan torsi eksternal [11].

$$\dot{\omega} = \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = I^{-1}(\tau - \omega \times (I\omega)) \quad (5)$$

*Quadcopter* dapat dimodelkan dengan dua batang tipis yang melintasi massa motor. Pemodelan ini akan menghasilkan matrik inersia berbentuk diagonal [11].

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (6)$$

Kemudian pemodelan didapatkan untuk gerak rotasi *body frame* dengan persamaan (7) [11].

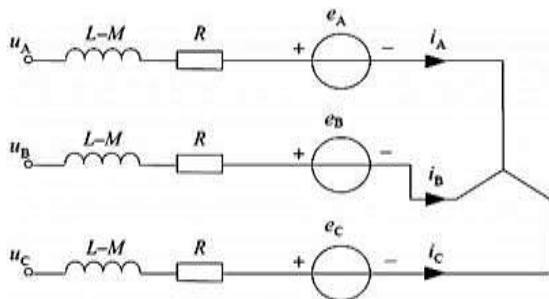
$$\dot{\omega} = \begin{bmatrix} \tau_\phi I_{xx}^{-1} \\ \tau_\theta I_{yy}^{-1} \\ \tau_\psi I_{zz}^{-1} \end{bmatrix} - \begin{bmatrix} \frac{I_{yy}-I_{zz}}{I_{xx}} \omega_y \omega_z \\ \frac{I_{zz}-I_{xx}}{I_{yy}} \omega_x \omega_z \\ \frac{I_{xx}-I_{yy}}{I_{zz}} \omega_x \omega_y \end{bmatrix} \quad (7)$$

2.4. Motor BLDC

Motor *brushless dc* (BLDC) terdiri dari empat motor untuk mengatur 6 *degree of freedom* (DOF). 6 DOF diperoleh dengan menghitung persamaan *geometric*, *dynamic* dan evaluasi *aerodynamic* dari desain mekanik motor [12].

$$\omega = A\omega + BV + C \tag{8}$$

Motor BLDC dapat dimodelkan dengan rangkaian ekuivalen sebagai berikut. Tegangan input merupakan  $U_A, U_B, U_C$ , *phase inductance of winding* adalah  $L$ , *phase resistance of winding* adalah  $R$ , back-emf adalah  $e_A, e_B, e_C$  dan  $i_A, i_B, i_C$  adalah arus fasa [12]. Dimana A merupakan koefisien kecepatan *propeller*, B koefisien tegangan input, C koefisien konstan,  $\omega$  kecepatan *propeller*, dan V vektor tegangan input. [13].



Gambar 5. Rangkaian Ekuivalen Motor BLDC [12]

Gambar 5 rangkaian ekuivalen motor BLDC dengan mengatur variasi arus dan tegangan pada motor. Pengaturan kecepatan motor menggunakan sinyal *pulse width modulation* (PWM) dengan *electronic speed control* (ESC). Kecepatan putaran diukur dengan sensor. ESC digunakan untuk mengatur torsi motor dengan frekuensi dan jumlah arus pulsa pada belitan motor [13]. Aplikasi motor digunakan dalam *quadcopter* dan torsi dihasilkan oleh persamaan (9).

$$\tau = K_t(I - I_0) \tag{9}$$

Dimana  $\tau$  merupakan torsi motor,  $I$  mewakili input arus,  $I_0$  adalah arus ketika tanpa beban, dan  $K_t$  merupakan konstanta torsi. Tegangan yang melewati motor merupakan jumlah back-EMF dan rugi-rugi resistif [13].

$$V = IR_m + K_v\omega \tag{10}$$

Dimana  $V$  merupakan tegangan *drop* yang melintasi motor,  $R_m$  mewakili resistansi motor,  $\omega$  merupakan kecepatan sudut motor, dan  $K_v$  adalah konstanta yang mengindikasikan back-EMF untuk menghasilkan RPM. Perhitungan daya pada motor menggunakan persamaan (11) [13].

$$P = IV = \frac{(\tau + K_t I_0)(K_t I_0 R_m + \tau R_m + K_t K_v \omega)}{K_t^2} \tag{11}$$

Untuk menyederhanakan model, persamaan model diasumsikan dengan mengabaikan resistansi motor. Kemudian daya menjadi kecepatan sudut [13].

$$P \approx \frac{(\tau + K_t I_0) K_v \omega}{K_t} \tag{12}$$

$K_t I_0 \ll \tau$ .  $I_0$  diasumsikan sebagai arus tanpa beban. Kemudian persamaan daya dapat disederhanakan sebagai berikut [13].

$$P \approx \frac{K_v}{K_t} \tau \omega \tag{13}$$

2.5. Gaya

Daya digunakan untuk menjaga *quadcopter* tetap melayang di udara. Dengan menjaga daya motor dalam waktu periode sama dengan gaya yang dihasilkan pada *propeller* dikalikan dengan jarak udara ( $P \cdot dt = F \cdot dx$ ). sehingga daya sama dengan *thrust* dikalikan kecepatan udara ( $P = F \frac{dx}{dt}$ ) [14].

$$P = T v_h \tag{14}$$

Kecepatan *quadcopter* pada udara diasumsikan rendah, sehingga  $v_h$  merupakan kecepatan udara ketika melayang dan  $v_\infty$  adalah 0. Teori momentum memberikan persamaan kecepatan pada saat melayang sebagai fungsi *thrust* [14].

$$v_h = \sqrt{\frac{T}{2\rho A}} \tag{15}$$

Dimana  $\rho$  merupakan kerapatan udara dan A adalah luas rotor. persamaan daya dapat ditulis sebagai berikut [14].

$$P = \frac{K_v}{K_t} \tau \omega = \frac{K_v K_t}{K_t} T \omega = \frac{T^{\frac{3}{2}}}{\sqrt{2\rho A}} \tag{16}$$

Pada kasus umum  $\tau = \vec{r} \times \vec{F}$ , sedangkan pada kasus ini, torsi harus sesuai dengan *thrust* T dan rasio konstanta  $K_t$  ditentukan oleh parameter. *Magnitude thrust* T sama dengan kecepatan sudut motor [14].

$$P = \left( \frac{K_v K_t \sqrt{2\rho A}}{K_t} \omega \right)^2 = k\omega^2 \tag{17}$$

Dimana  $k$  mewakili konstanta. Penjumlahan total *thrust* motor pada *quadcopter* dalam *body frame* dijelaskan dengan persamaan (18) [14].

$$T_B = \sum_{i=1}^4 T_i = k \begin{bmatrix} 0 \\ 0 \\ \sum \omega_i^2 \end{bmatrix} \tag{18}$$

Gaya *thrust* dimodelkan sebagai gaya *drag* yang sesuai dengan kecepatan linear pada setiap arah. Pemodelan gaya *thrust* memudahkan untuk melihat gesekan fluida. gaya *drag* secara umum dapat dilihat pada persamaan (19) [14].

$$F_D = \begin{bmatrix} -k_d \dot{x} \\ -k_d \dot{y} \\ -k_d \dot{z} \end{bmatrix} \quad (19)$$

## 2.6. Torsi

Torsi *quadcopter* harus dihitung. Setiap motor menghasilkan torsi pada sumbu z. Torsi dibutuhkan untuk menjaga *propeller* berputar dan menghasilkan *thrust*. Torsi dibuat secara terus menerus pada kecepatan sudut dan mengatasi gesekan gaya *drag*. Persamaan *drag force* dari fluida dinamis memberikan gaya gesek [14].

$$F_D = \frac{1}{2} \rho C_D A V^2 \quad (20)$$

Dimana  $\rho$  merupakan kerapatan fluida, A mewakili luas refrensi (*propeller*) dan  $C_D$  adalah konstanta tanpa dimensi. Torsi diakibatkan gaya *drag* pada persamaan (21) [14].

$$\tau_D = \frac{1}{2} R_\rho C_D A v^2 = \frac{1}{2} R_\rho C_D A (\omega R)^2 = b \omega^2 \quad (21)$$

Dimana  $\omega$  kecepatan sudut *propeller*. R merupakan jari-jari *propeller*.  $b$  mewakili konstanta dimensi. Torsi *drag* sebanding dengan kuadrat kecepatan sudut. Torsi *drag* secara keseluruhan ditulis sebagai berikut sumbu z ith [14].

$$\tau_z = b \omega^2 + I_M \dot{\omega} \quad (22)$$

Dimana  $I_M$  adalah momen inersia motor sumbu z,  $\dot{\omega}$  mewakili kecepatan sudut *propeller*. dan  $b$  adalah koefisien *drag*. Dalam keadaan terbang *steady state* (tidak *take off* dan *landing*) sebagian besar kecepatan *propeller* dipertahankan konstan, sehingga persamaan (23) disederhanakan sebagai berikut [14].

$$\tau_z = (-1)^{i+1} b \omega_i^2 \quad (23)$$

Dimana  $(-1)^{i+1}$  merupakan pola positif untuk *propeller* ith, *propeller* berputar searah jarum jam (negatif) dan berputar berlawanan jarum jam (positif). Total torsi sumbu z diberikan dengan menjumlahkan torsi dari setiap baling-baling [14].

$$\tau_\psi = b(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \quad (24)$$

Torsi *roll* dan *pitch* merupakan standart gerakan secara mekanik, sehingga arus motor  $i = 1$  dan  $i = 3$ , dan torsi *roll*, *pitch* menjadi persamaan (25) [14].

$$\tau_\phi = \sum r \times T = L(k\omega_1^2 - k\omega_3^2) = Lk(\omega_1^2 - \omega_3^2) \quad (25)$$

Torsi *pitch* diberikan dengan persamaan yang sama sebagai berikut [14].

$$\tau_\theta = Lk(\omega_2^2 - \omega_4^2) \quad (26)$$

Dimana L jarak dari pusat *quadcopter* terhadap *propeller*. torsi dalam *body frame* adalah [14].

$$\tau_B = \begin{bmatrix} Lk(\omega_1^2 - \omega_3^2) \\ Lk(\omega_2^2 - \omega_4^2) \\ b(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \quad (27)$$

Model disederhanakan dengan mengabaikan dinamika *quadcopter nonlinear*, *drag force*, sudu *propeller* yang mengakibatkan kecepatan tinggi, dan kecepatan fluida. Model ini memiliki bagian yang dibutuhkan pada dinamika *quadcopter* [14].

## 2.7. Euler-Newton Method

Pergerakan *quadcopter* berdasarkan translasi dan rotasi pada sudut  $(x, y, z, \phi, \theta, \psi)$  dengan menggunakan persamaan Euler-Newton sebagai berikut [15].

$$\dot{x} = \frac{dx}{dt} \quad (28)$$

$$\dot{y} = \frac{dy}{dt} \quad (29)$$

$$\dot{z} = \frac{dz}{dt} \quad (30)$$

$$\dot{x} = u_1(\cos \phi \cos \psi + \sin \phi \sin \psi) - K_1 \frac{\dot{x}}{m} \quad (31)$$

$$\dot{y} = u_1(\sin \phi \sin \theta \cos \psi + \cos \phi \sin \psi) - K_1 \frac{\dot{y}}{m} \quad (32)$$

$$\dot{z} = u_1(\cos \phi \cos \psi) - g - K_3 \frac{\dot{z}}{m} \quad (33)$$

$$\dot{\theta} = \omega_x \quad (34)$$

$$\dot{\psi} = \omega_y \quad (35)$$

$$\dot{\phi} = \omega_z \quad (36)$$

$$\ddot{\theta} = u_2 - 1K_4 \frac{\dot{\theta}}{I_1} \quad (37)$$

$$\ddot{\psi} = u_3 - 1K_5 \frac{\dot{\psi}}{I_2} \quad (38)$$

$$\ddot{\phi} = u_4 - 1K_6 \frac{\dot{\phi}}{I_3} \quad (39)$$

Dimana  $(x,y,z)$  mewakili gerakan translasi. Euler-Newton diwakili dengan gerakan rotasi  $(\phi, \theta, \psi)$ . Variabel titik di

atas  $(\dot{\phi}, \dot{\gamma}, \dot{z})$  mewakili *velocity* dan  $(\ddot{x}, \ddot{y}, \ddot{z})$  mewakili *acceleration* dalam arah masing-masing [15].

### 2.9. PD Controller

*Quadcopter* dikendalikan dengan menggunakan kontrol *Proportional Integral* (PD). PD menyesuaikan *error* antara *input* yang diinginkan dengan *output* yang diharapkan. *Quadcopter* memiliki *gyro*, sehingga gerakan dapat dikendalikan dengan sudut  $\phi, \theta, \psi$ . Nilai ini akan memberikan *error* pada *output*, dan integral akan memberikan *error* aktual dengan kestabilan pada posisi horisontal. Kecepatan dan sudut yang diinginkan akan menjadi nol. Torsi berhubungan dengan kecepatan sudut  $\tau = I\dot{\theta}$ . Jadi pengaturan torsi yang sesuai output dengan kontrol adalah  $\tau = Iu(t)$  [15].

$$\begin{bmatrix} \tau_{\phi} \\ \tau_{\theta} \\ \tau_{\psi} \end{bmatrix} = \begin{bmatrix} -I_{xx} \left( K_d \dot{\phi} + K_p \int_0^T \dot{\phi} dt \right) \\ -I_{yy} \left( K_d \dot{\theta} + K_p \int_0^T \dot{\theta} dt \right) \\ -I_{zz} \left( K_d \dot{\psi} + K_p \int_0^T \dot{\psi} dt \right) \end{bmatrix} \quad (40)$$

Sebelumnya persamaan diturunkan dengan hubungan antara torsi dan *input* [15].

$$\tau_B = \begin{bmatrix} Lk(\gamma_1 - \gamma_3) \\ Lk(\gamma_2 - \gamma_4) \\ b(\gamma_1 - \gamma_2 + \gamma_3 - \gamma_4) \end{bmatrix}$$

$$\tau_B = \begin{bmatrix} -I_{xx} \left( K_d \dot{\phi} + K_p \int_0^T \dot{\phi} dt \right) \\ -I_{yy} \left( K_d \dot{\theta} + K_p \int_0^T \dot{\theta} dt \right) \\ -I_{zz} \left( K_d \dot{\psi} + K_p \int_0^T \dot{\psi} dt \right) \end{bmatrix} \quad (41)$$

Sehingga persamaan dibatasi dan input harus menjaga kestabilan *quadcopter* dalam posisi melayang [15].

$$T = mg \quad (42)$$

Persamaan mengabaikan bahwa *thrust* tidak akan diarahkan langsung ke atas. *Thrust* akan membatasi kontroler. Jika kestabilan ditentukan dengan sudut arus secara akurat, kestabilan akan mengkompensasi. Jika *gyro* cukup presisi, *gyro* dapat menghasilkan nilai yang diperoleh dari sudut  $\theta, \phi$ . Perhitungan *thrust* dibutuhkan untuk menjaga *quadcopter* pada posisi melayang dengan *thrust mg* ke dalam sumbu z [15].

$$T_{proj} = mg \cos \theta \cos \phi \quad (43)$$

Pengukuran sudut yang presisi, sebaliknya dengan menggunakan syarat *thrust* sama dengan T [15].

$$T = \frac{mg}{\cos \theta \cos \phi} \quad (44)$$

Dimana *thrust* menunjuk pada sumbu z positif akan sama dengan *mg*. *thrust* sesuai dengan jumlah berat *input* [15].

$$T = \frac{mg}{\cos \theta \cos \phi} = k \sum \gamma_i \quad (45)$$

$$\sum \gamma_i = \frac{mg}{K \cos \theta \cos \phi} \quad (46)$$

Dengan batasan, *thrust* dikendalikan dengan empat persamaan yang tidak diketahui  $\gamma_i$ . Nilai input sebagai berikut [15].

$$\gamma_1 = \frac{mg}{4K \cos \theta \cos \phi} - \frac{2be_{\phi} I_{xx} + e_{\psi} I_{zz} kL}{4bkl} \quad (47)$$

$$\gamma_2 = \frac{mg}{4K \cos \theta \cos \phi} - \frac{e_{\psi} I_{zz}}{4b} - \frac{e_{\theta} I_{yy}}{2kl} \quad (48)$$

$$\gamma_3 = \frac{mg}{4K \cos \theta \cos \phi} - \frac{-2be_{\phi} I_{xx} + e_{\psi} I_{zz} kL}{4bkl} \quad (49)$$

$$\gamma_4 = \frac{mg}{4K \cos \theta \cos \phi} - \frac{e_{\psi} I_{zz}}{4b} - \frac{e_{\theta} I_{yy}}{2kl} \quad (50)$$

### 2.10. PID Controller

PID *controller* digunakan sebagai pengaturan stabilitas gerakan *altitude quadcopter* dengan persamaan (51-53). Pengaturan PI merupakan pegendali yang mudah dalam pengaplikasiannya, tetapi PI tidak layak untuk mengendalikan sistem mekanik, karena masih ada *nois*, gangguan dan *error steady state*. Sehingga pengaturan PID harus mampu menjaga stabilitas *quadcopter* dengan nilai *error* yang lebih kecil. Persamaan pengaturan PID identik dengan penambahan ketentuan dalam *error* [15].

$$e_{\phi} = K_d \dot{\phi} + K_p \int_0^T \dot{\phi} dt + K_i \int_0^T \int_0^T \dot{\phi} dt dt \quad (51)$$

$$e_{\theta} = K_d \dot{\theta} + K_p \int_0^T \dot{\theta} dt + K_i \int_0^T \int_0^T \dot{\theta} dt dt \quad (52)$$

$$e_{\psi} = K_d \dot{\psi} + K_p \int_0^T \dot{\psi} dt + K_i \int_0^T \int_0^T \dot{\psi} dt dt \quad (53)$$

### 2.11. Altitude Controller

*Altitude controller quadcopter* melibatkan *input* kecepatan motor, Pemodelan input kecepatan motor dijelaskan pada persamaan (54-57). Persamaan pertama  $U_1$  mewakili *altitude* motor. *Step input* akan merubah gaya angkat dari *quadcopter*. *Step input*  $U_2$  mewakili gaya putaran *roll*. *Step input*  $U_3$  mewakili gaya putaran *pitch* dan *step input*  $U_4$  mewakili gaya putaran *yaw*. Sedangkan *thrust* dihasilkan oleh  $f_1$ . Momen inersia  $I_i$ ,  $m$  merupakan massa dari *quadcopter*,  $l$  merupakan panjang satu sisi *quadcopter*, dan  $C$  mewakili *moment scaling factor* [15].

$$U_2 = \frac{(f_1 + f_2 + f_3 + f_4)}{m} \quad (54)$$



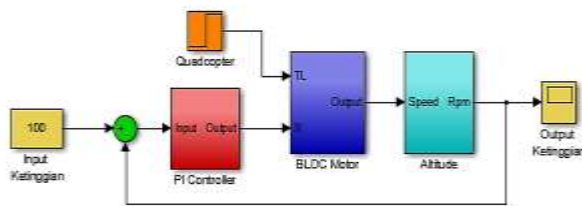
$$U_2 = \frac{l(-f_1 - f_2 + f_3 + f_4)}{I_1} \quad (55)$$

$$U_3 = \frac{l(-f_1 + f_2 + f_3 - f_4)}{I_2} \quad (56)$$

$$U_4 = \frac{c(f_1 - f_2 + f_3 - f_4)}{I_3} \quad (57)$$

### 3. Hasil dan Pembahasan

Hasil dan pembahasan pada desain sistem kontrol motor pada penggerak *quadcopter* dijelaskan pada gambar 6.



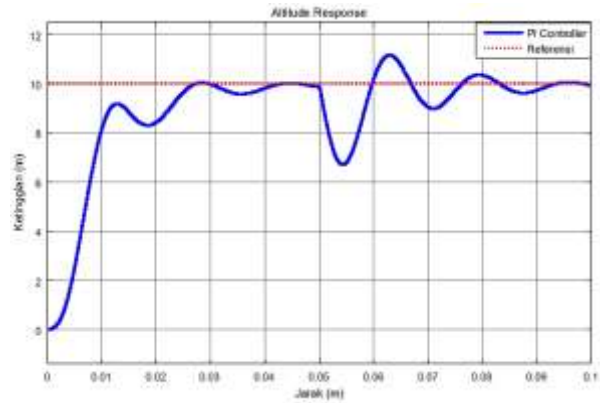
Gambar 6. Desain Sistem Kontrol Kecepatan Motor pada Gerakan *Quadcopter*

Kecepatan motor dikendalikan dengan *electronic speed control* (ESC) dan sensor posisi yang dipasang pada sisi rotor. Sensor posisi dapat mendeteksi magnet pada rotor, dimana sistem kontrol terdiri dari *inverter* tiga fasa IGBT. *Inverter* tiga fasa digunakan sebagai pengendali untuk menghasilkan sinyal *pulse width modulation* (PWM). *Brushless dc motor* (BLDC) dikendalikan dengan sinyal PWM berdasarkan variasi pensaklaran, dan pendeteksian posisi rotor menggunakan *hall sensor*. Kecepatan motor BLDC diatur dengan sinyal PWM dan *duty cycle* yang dikontrol dengan pengaturan bervariasi sehingga menghasilkan *output* kecepatan [16]. *Output* kecepatan motor diaplikasikan pada *quadcopter* untuk menghasilkan gaya dorong secara *vertical* dan sebagai pengaturan objek gerakan ketinggian dari 10 m sampai 90 m. Kemudian *output* kecepatan diumpun balikkan jika terjadi selisih nilai kesalahan antara *output* kecepatan dan *input* kecepatan. Selanjutnya nilai kesalahan dikendalikan oleh kontrol PID agar keluaran kecepatan dan masukan kecepatan sama.

#### 3.1. Respon Motor pada Gerakan Ketinggian *Quadcopter*

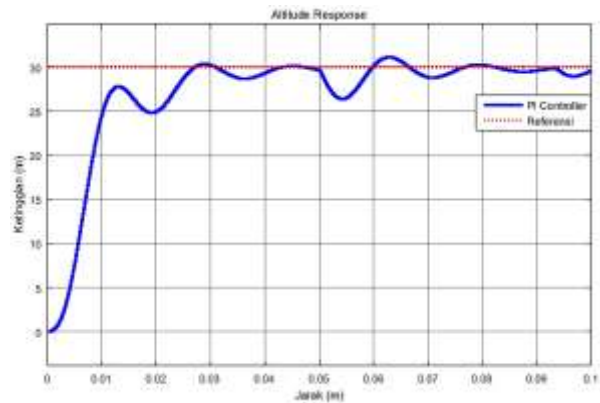
Respon kecepatan motor pada gerakan ketinggian *quadcopter* menggunakan kontrol PID dengan parameter  $K_p$  10,  $K_i$  0.002 menghasilkan osilasi dengan waktu naik 10 m. Pemilihan parameter dilakukan dengan percobaan sehingga mendapatkan respon kecepatan motor yang sesuai dengan gerakan ketinggian *quadcopter*. Kemudian torsi beban motor *brushless dc* diberikan sebesar 0.0318 kg. Torsi beban (*quadcopter*) mempengaruhi kecepatan motor untuk mencapai ketinggian yang *steady state*. Respon waktu naik 0,013 detik, *overshoot* 0%, dan waktu

*steady state* memerlukan 0.1 detik dan respon memberikan osilasi pada beban dengan *amplitude* 8.5-9.5 dapat dilihat pada gambar 7.



Gambar 7. Respon Motor Pada Gerakan Ketinggian 10 m Menggunakan Kontrol PI

Gambar 8 respon kecepatan motor *brushless dc* pada gerakan ketinggian *quadcopter* menghasilkan ketinggian 30 m dengan torsi beban konstan. Respon waktu naik adalah 0.013 detik, *overshoot* 0%, sedangkan respon beban beresilasi pada waktu 0.01 detik dengan *amplitude* 26-28 yang dipengaruhi oleh *quadcopter* dan waktu *steady state* 0.13 detik

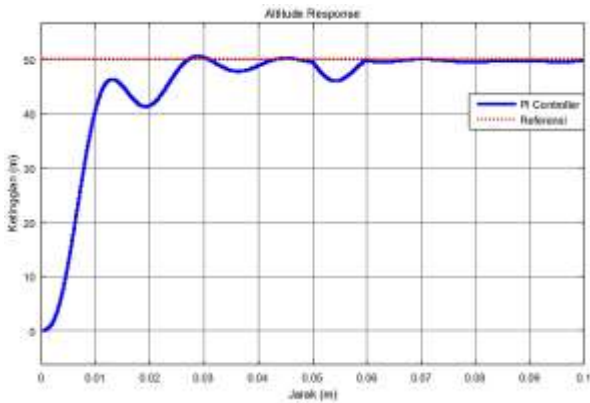


Gambar 8. Respon Motor pada Gerakan Ketinggian 30 m Menggunakan Kontrol PI

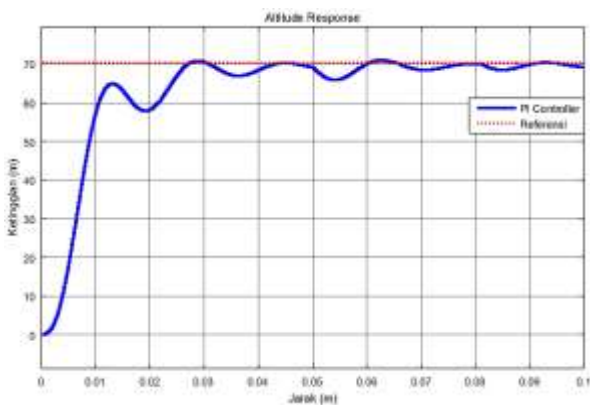
Gambar 9 hasil simulasi respon kecepatan motor pada gerakan ketinggian *quadcopter* mencapai ketinggian 50 m dan torsi beban (*quadcopter*) 0.0138 kg. gerakan ketinggian *quadcopter* menghasilkan osilasi 110 m, sedangkan waktu *steady state* 0.05 detik. Respon perubahan beban terjadi pada waktu 0.1 detik dengan *amplitude* 45-46. respon waktu naik diperoleh 0.01 detik, *overshoot* 0%, dan waktu *steady state* mencapai waktu 0.12 detik.

Gambar 10 menunjukkan respon kecepatan motor *brushless dc* pada gerakan ketinggian *quadcopter* 70 m.

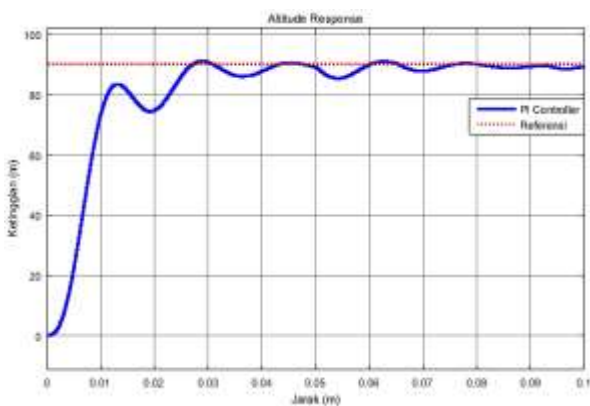
Waktu mencapai *steady state* 0.06 detik. Respon perubahan beban terjadi pada waktu 0.05 detik dengan *amplitude* sebesar 63-65 yang dipengaruhi beban torsi (*quadcopter*). Kemudian respon waktu naik dicapai 0.01 detik, *overshoot* 0%.



Gambar 9. Respon Motor pada Gerakan Ketinggian 50 m Menggunakan Kontrol PI



Gambar 10. Respon Motor pada Gerakan Ketinggian 70 m Menggunakan Kontrol PI



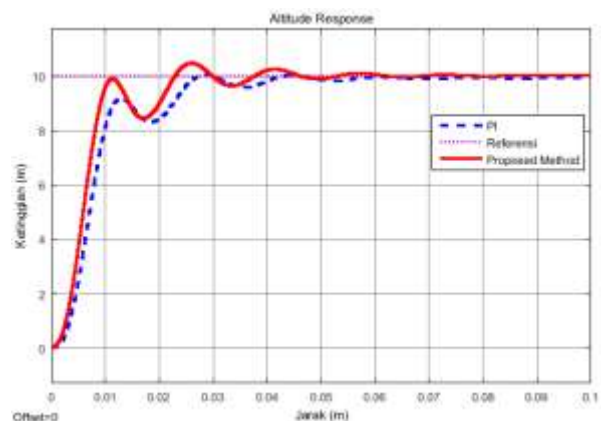
Gambar 11. Respon Motor pada Gerakan Ketinggian 90 m Menggunakan Kontrol PI

Gambar 11 menjelaskan bahwa respon dengan kecepatan motor *brushless* dc pada gerakan ketinggian *quadcopter* 90

m. Respon yang dihasilkan mengalami osilasi 190 m dan waktu mencapai *steady state* 0.6 detik. Respon perubahan beban terjadi pada waktu 0.1 detik dengan *amplitude* sebesar 82-83 yang dipengaruhi beban torsi (*quadcopter*). Kemudian respon kembali dalam keadaan *steady state* 0.1 detik, respon waktu naik 0.13 detik, dan *overshoot* 0%.

### 3.2. Respon Motor pada Gerakan Ketinggian *Quadcopter* Menggunakan PID

Gambar 12 Pemilihan parameter PID dan PID dilakukan dengan percobaan sehingga menghasilkan respon kecepatan motor pada penggerak *quadcopter* Respon kecepatan motor pada gerakan ketinggian *quadcopter* menggunakan kontrol (PI) dengan nilai parameter  $K_p$  0.002,  $K_i$  10 dan kontrol (PID) dengan nilai  $K_p$  0.0009,  $K_i$  13,  $K_d$  0.00009 menghasilkan osilasi dengan waktu naik 10 m. Sedangkan torsi beban motor *brushless* dc diberikan sebesar 0.0318 kg. Torsi beban (*quadcopter*) mempengaruhi kecepatan motor untuk mencapai ketinggian yang *steady state*. respon waktu naik untuk mencapai *steady state* memerlukan 0.07 detik. Perbandingan pengaturan PI dengan *proposed* PID bahwa PID memiliki *overshoot* 0% detik dan waktu *steady state* 0.06 detik.



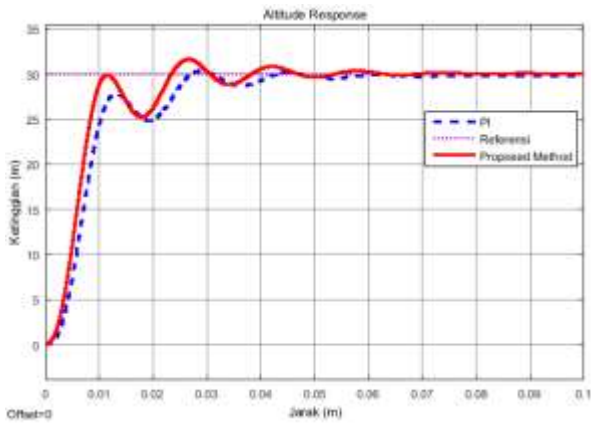
Gambar 12. Respon Motor pada Gerakan Ketinggian 10 m Menggunakan Kontrol PID

Hasil simulasi kecepatan motor *brushless* dc pada gerakan ketinggian *quadcopter* menggunakan pengaturan PI dan PID ditunjukkan pada gambar 13. Respon mencapai ketinggian 30 m dengan beban konstan dan waktu *steady state* 0.08 detik. Perubahan beban terjadi pada waktu 0.1 detik dengan osilasi *amplitude* 27-31. Respon PID memiliki respon waktu naik 0.01 detik, *overshoot* 0%, dan waktu *steady state* 0.06 detik lebih cepat dibandingkan dengan respon PI.

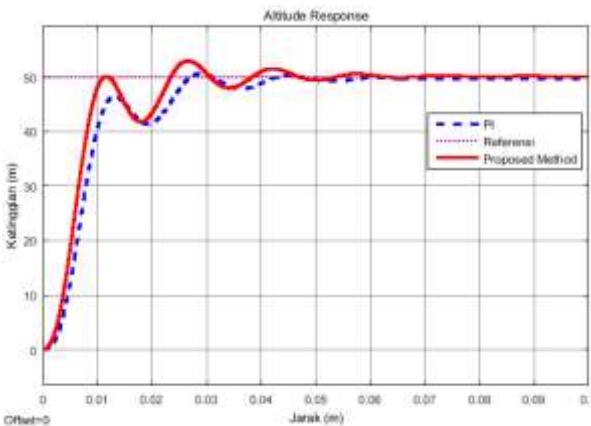
Gambar 14 respon kecepatan motor *brushless* dc pada gerakan ketinggian *quadcopter* menggunakan pengaturan PI dan PID dengan parameter masing-masing adalah  $K_p$  0.002,  $K_i$  10 dan  $K_p$  0.0009,  $K_i$  13,  $K_d$  0.00009. Respon mencapai ketinggian 50 m dengan beban konstan dan waktu



steady state 0.8 detik. Perubahan beban terjadi pada waktu 0.1 detik dengan osilasi *amplitude* 46-49. Respon waktu naik PI mencapai 0.013 detik, *overshoot* 0%, sedangkan respon waktu naik dengan kontrol PID diperoleh 0.01 detik, *overshoot* 0%, dan waktu *steady state* 0.06 detik.

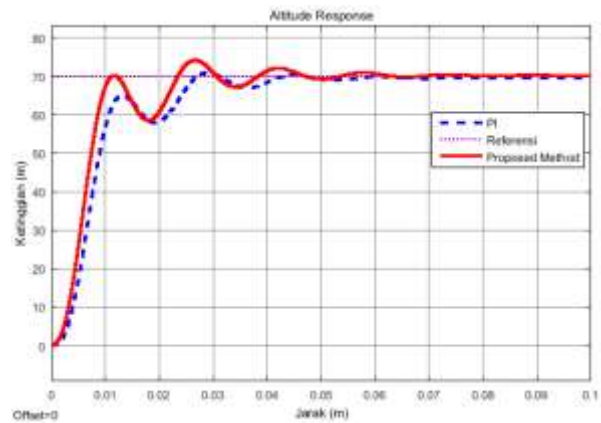


Gambar 13. Respon Motor pada Gerakan Ketinggian 30 m Menggunakan Kontrol PID



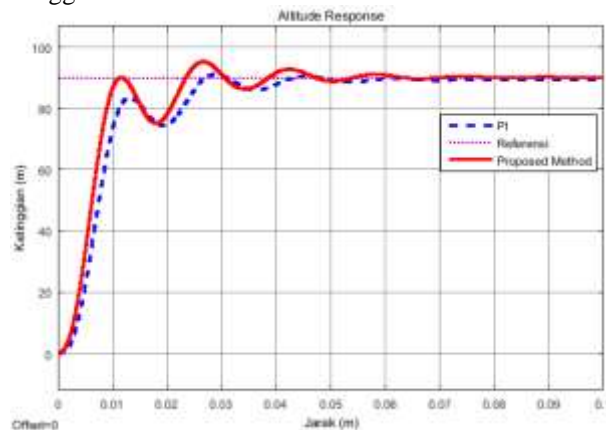
Gambar 14. Respon Motor pada Gerakan Ketinggian 50 m Menggunakan Kontrol PID

Gambar 15 menunjukkan respon kecepatan motor *brushless* dc pada gerakan ketinggian *quadcopter* memberikan respon ketinggian 70 m. Ketinggian diatur menggunakan kontrol PID dengan nilai parameter  $K_p$  0.002,  $K_i$  10. Kontrol PID dengan nilai parameter  $K_p$  0.0009,  $K_i$  13,  $K_d$  0.00009 memperoleh respon osilasi 50 dengan waktu naik 0.01 detik. Respon berosilasi dipengaruhi oleh beban konstan. Perubahan beban terjadi pada waktu 0.1 detik dengan *amplitude* sebesar 66-69. Kemudian waktu *steady state* PI mencapai 0.09 detik, *overshoot* 0%, dan respon waktu naik 0.013 detik. Sedangkan PID mencapai respon waktu naik 0.01 detik, *overshoot* 0% dan waktu mencapai *steady state* 0.1 detik. Perbandingan respon waktu naik respon kontrol PI 0.013 detik lebih lambat dibandingkan PID 0.01 detik.



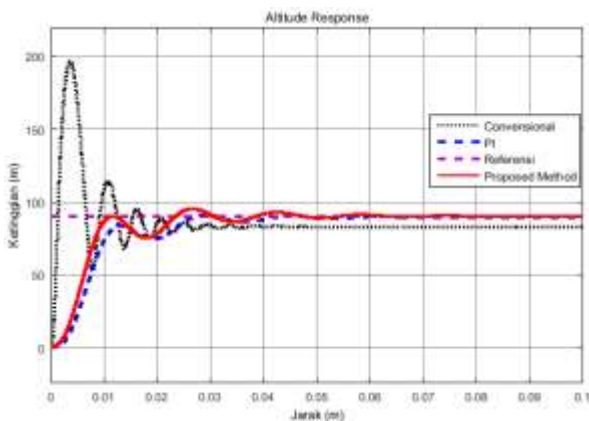
Gambar 15. Respon Motor pada Gerakan Ketinggian 70 m Menggunakan Kontrol PID

Gambar 16 menunjukkan hasil simulasi *altitude* kecepatan motor *brushless* dc menggunakan pengaturan  $K_p$  0.002,  $K_i$  10 dan  $K_d$  0.0009,  $K_i$  13,  $K_d$  0.00009. Respon *altitude* mencapai ketinggian 90 m dengan beban konstan dan waktu *steady state* 0.8 detik. Kontrol PI mencapai respon waktu naik 0.013 detik, *overshoot* 0%, waktu *steady state* 0.1 detik dan kontrol PID mencapai respon waktu naik 0.01 detik, *overshoot* 0%, dan waktu *steady state* 0.07 detik dengan osilasi *amplitude* 86-89. Respon mencapai keadaan *steady state* PID lebih cepat dibandingkan respon menggunakan PI.



Gambar 16. Respon Motor pada Gerakan Ketinggian 90 m Menggunakan Kontrol PID

Gambar 17 menunjukkan hasil simulasi respon kecepatan motor *brushless* dc pada gerakan ketinggian *quadcopter* menggunakan pengaturan PI dan PID. Kontrol PID memperoleh respon waktu naik lebih cepat dibandingkan dengan kontrol PI dan konvensional. Waktu mencapai *steady state* pada kontrol PID 0.06 detik, *overshoot* 0%, respon waktu naik 0.01 detik, Sedangkan kontrol konvensional PI menghasilkan respon waktu naik 0.013 detik, *overshoot* 0%, dan waktu mencapai *steady state* 0.1 detik. Osilasi ketiga kontrol mencapai ketinggian 190, 90 dan 80.



Gambar 17. Respon Motor pada Gerakan Ketinggian 90 m Menggunakan Kontrol PID

#### 4. Kesimpulan

Desain sistem kontrol kecepatan motor *brushless* dc digunakan untuk mengendalikan kestabilan gerakan *altitude vertical take-off*. Desain kontrol berdasarkan beban dan *thrust force*. Kecepatan motor mempengaruhi ketinggian gaya angkat *quadcopter*. Pergerakan *quadcopter* diperoleh dengan metoda Euler-Newton. *Proporsional Integral Derivatif* (PID) diusulkan untuk mengatur kestabilan gerakan *altitude*, kemudian desain kontrol disimulasi dan divalidasi menggunakan Matlab Simulink. Hasil simulasi menunjukkan bahwa pengaturan PID pada ketinggian 10 m sampai 90 m mendapatkan respon waktu naik lebih cepat 0.01 detik, *overshoot* 0%, waktu *steady state* 0.06 detik. Sedangkan pengaturan PI pada ketinggian 10 m sampai 90 m menghasilkan respon waktu naik lebih lama 0.013 detik, *overshoot* 0%, waktu *steady state* 0.1 detik.

#### Referensi

[1]. J. A. Prakosa, D. V. Samokhvalov, G. R. V. Ponce, and F. S. Al-Mahturi, "Speed control of brushless DC motor for quad copter drone ground test," *Proc. 2019 IEEE Conf. Russ. Young Res. Electr. Electron. Eng. (ElConRus)* 28-31 January, 2019, pp. 644–648, 2019.

[2]. M. Latif and B. Hairil, "Perancangan Sistem Autonomous Quadcopter," *Prociding. Seminar Nasional Sains dan Teknologi*, vol. 54, no. 4, pp. 1–5, 2014.

[3]. F. Ahmed, P. Kumar, and P. P. Patil, "Modeling and simulation of a quadcopter Using PID Controller" *International Journal of Control Theory and Application (IJCTA)*, vol. 9, no 15, pp. 7151-7158, 2016.

[4]. Y. Benmoussa, A. Mansouri, and A. Ahaitouf, *Quadrotor flight simulator modeling*, vol. 915. Springer International Publishing, 2019.

[5]. M. H. Tanveer and S. F. Ahmed, "Stabilized controller design for attitude and altitude controlling of quad-rotor under disturbance and noisy conditions" *American Journal of Applied Sciences*. vol. 8, no. 10, pp. 891-831, 2013.

[6]. M. Z. Mustapa, "Altitude controller design for quadcopter UAV," *Journal Teknologi*, vol. 74, no. 1, pp. 187–194, 2015.

[7]. A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadrotor aircraft," *IEEE Trans. Control System Technology*, vol. 14, no. 3, pp. 562–571, 2006.

[8]. T. Oktay and O. Köse, "Dynamic Modeling and Simulation of Quadrotor for Different Flight Conditions" *European Journal of Science Technology*, no. 15, pp. 132–142, 2019.

[9]. E. Yudaningtyas, *Belajar Sistem Kontrol Soal dan Pembahasan*. Malang: UB Press Malang, 2017.

[10]. T. Luukkonen, "Modelling and control of quadcopter" *Aalto University School of Science*, 2011.

[11]. A. Gibiansky, "Quadcopter Dynamics , Simulation , and Control Introduction Quadcopter Dynamics" *Andrew.gibiansky.com*, pp. 1–18, 2012.

[12]. A. A. El-samahy and M. A. Shamseldin, "Brushless DC motor tracking control using self-tuning fuzzy PID control and model reference adaptive control," *Ain Shams Engineering Journal.*, vol. 9, no. 3, pp. 341–352, 2018.

[13]. M. Becker, R. C. B. Sampaio, S. Bouabdallah, V. Perrot, and R. Siegwart, "In flight collision avoidance for a Mini-UAV robot based on onboard sensors" *Journal. Brazilian Soc. Mech. Sci. Eng.*, vol. 2, no. 1 February, p. 12, 2012.

[14]. E. Gopalakrishnan, "Quadcopter Flight Mechanics Model and Control Algorithms" *Master Thesis*, Prague, Czech Technical University, 2017.

[15]. U. A. Fiaz and A. Mukarram, "Altitude Control of a Quadcopter" *Master Thesis*, Pakistan Institute of Engineering and Applied Science, Nilore, Islamabad. June 2015.

[16]. D. Akbar and S. Riyadi, "Pengaturan Kecepatan Motor Brushless DC (BLDC) menggunakan PWM (Pulse Width Modulation)," *Seminar Nasional Instrumentasi Kontrol dan Otomasi 2018*, pp. 2–2, 2018.