

## Aplikasi *Pilot In the Loop* (PIL) untuk uji unjuk kerja dan kekokohan sistem kontrol *proportional-derivative* (PD) dengan *Hardware in The Loop* (HIL)

\*Agus Mukhtar, Joga Dharma Setiawan, Munadi

Program Studi Magister Teknik Mesin, Fakultas Teknik, Universitas Diponegoro

Jl. Prof. Sudharto, SH., Tembalang-Semarang 50275, Telp. +62247460059

\*E-mail: Agus\_mukhtar\_uav@yahoo.com

### ABSTRAK

Pengembangan quadrotor telah banyak dikembangkan dalam berbagai bidang diantaranya adalah sebagai *traffic monitoring*, alat pelacak dan penyelamatan, alat pengintai dan sebagainya. Untuk itu diperlukan sistem kontrol yang kokoh dan teruji serta dapat dikendalikan dengan menggunakan remote kontrol. Dalam penelitian ini dilakukan pembuatan dan pengujian sistem kontrol *proportional-derivative* (PD) untuk quadrotor terhadap ketidakpastian masa dan inersia dengan *Hardware In the Loop System* (HIL) serta dikembangkan *Pilot in The Loop System* (PIL) untuk sistem kendalinya. Identifikasi parameter merupakan hal pertama yang perlu dilakukan untuk pembuatan quadrotor, kemudian dilakukan desain sistem kontrol dengan menggunakan *MATLAB/Simulink* untuk melakukan simulasi desain sistem kontrol dan PIL yang dibuat. Langkah selanjutnya adalah mengalikasikan desain sistem kontrol kedalam quadrotor. Hasil penelitian menunjukkan bahwa dalam simulasi desain kontrol PD terhadap ketidakpastian masa dan inersia dengan *settling time* 0.7s dan PO 3% diperoleh kondisi roll dan pitch respon dengan PO 0% dan *settling time* 0.7s; alt respon PO 3.3% dan *settling time* 0.7s; yaw respon PO 0%, *settling time* 0.9s dan *error steady state* 0.1°. Dalam aplikasinya pengujian dengan HIL pada respon roll diperoleh *error steady state* sekitar 1°. Sedangkan dalam aplikasi *Pilot In the Loop* (PIL) respon roll quadrotor sesuai dengan perintah yang diberikan, namun terdapat *error steady state* dan osilasi pada control PD.

**Kata kunci:** HIL, PD, PIL, sistem kontrol

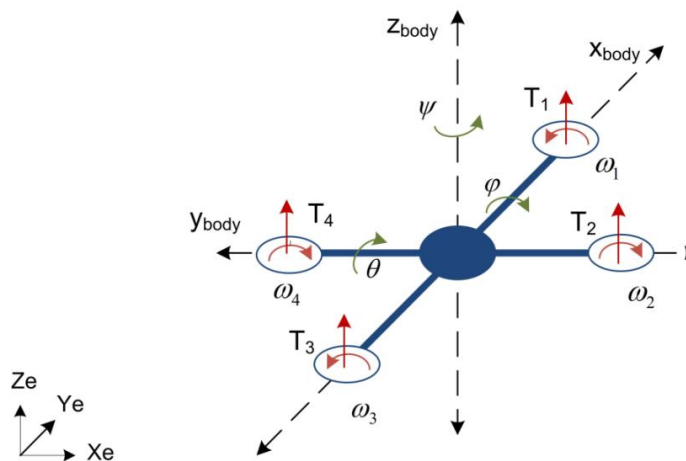
### 1. PENDAHULUAN

Quadrotor telah banyak dikembangkan dalam berbagai bidang diantaranya adalah sebagai alat pelacak dan penyelamatan, alat pengintai, *traffic monitoring* dan sebagainya. Dalam aplikasinya pembuatan quadrotor memerlukan metode simulasi dan penerapannya kedalam hardware (*Hardware In the Loop*) untuk menguji desain yang telah dibuat. Penelitian quadrotor dengan sistem kontrol PID dengan menggunakan HILS telah dikembangkan Utah University [1]. Atılım University juga telah mengembangkan quadrotor dengan sistem kontrol PID dan LQR dengan menggunakan HILS [2]. Dalam penelitian ini dikembangkan *Pilot In the Loop* (PIL) untuk quadrotor dalam pengujian kontrol PD dengan menggunakan HILS.

### 2. MATERIAL DAN METODOLOGI

#### 2.1. KINEMATIKA QUADROTOR

Quadrotor dapat dimodelkan sebagai empat buah rotor yang terhubung oleh lengan yang saling bersilangan. Setiap propeler terhubung dengan sebuah motor elektrik dan terpasang di setiap lengan. Dua frame dapat dianggap sebagai koordinat tetap di bumi (E-frame) dan koordinat badan quadrotor (B-frame) seperti pada Gambar 1.



**Gambar 1.** Koordinat quadrotor

Gaya angkat quadrotor disebabkan oleh putaran dari *propeller* yang selalu menuju ke arah sumbu Z positif terhadap koordinat badan quadrotor (*B-frame*), sedangkan masa quadrotor akan selalu menuju ke arah Z negatif terhadap koordinat tetap di bumi (*E-frame*).

Dalam pemodelan quadrotor diperlukan beberapa asumsi awal untuk membantu mempermudah dalam membuat model quadrotor, diantaranya adalah:

- Struktur quadrotor adalah simetris dan rigid.
- Pusat masa quadrotor berada pada posisi tengah quadrotor.
- Keempat propelernya rigid.
- Besarnya gaya angkat (*thrust*) sebanding dengan besarnya tegangan masukan motor.
- Gaya tarik berlawanan (*drag force*) terhadap badan quadrotor diabaikan karena hanya pada kondisi terbang rendah.

Hubungan kinematik untuk merubah kecepatan translasi dan kecepatan sudut terhadap koordinat badan quadrotor (*B-frame*) kedalam koordinat tetap di bumi (*E-frame*) dapat dihitung menggunakan persamaan 1.

$$\begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{bmatrix} = \begin{bmatrix} {}^G_B R(\eta_2) & 0 \\ 0 & Q(\eta_2) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (1)$$

## 2.2. DINAMIKA QUADROTOR

Persamaan dinamik quadrotor dapat dihitung dari aksioma pertama Euler dan Hukum Newton kedua. Persamaan dinamik 6 DOF dari rigid-body dengan memperhitungkan masa dari badan quadrotor dalam m [kg] dan matrix inersia I [Nms<sup>2</sup>] dapat dihitung dengan persamaan 2.

$$\begin{bmatrix} mI_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I \end{bmatrix} \begin{bmatrix} \dot{V}^B \\ \dot{\omega}^B \end{bmatrix} + \begin{bmatrix} \omega^B \times (mV^B) \\ \omega^B \times (I\omega^B) \end{bmatrix} = \begin{bmatrix} F^B \\ \tau^B \end{bmatrix} \quad (2)$$

Input gaya dan moment pada sumbu x, y dan z terhadap koordinat badan quadrotor (*B-frame*) yang dihasilkan dari empat propeller dapat dituliskan kedalam persamaan 3.

$$U_B = \begin{bmatrix} 0 \\ 0 \\ U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (3)$$

Sedangkan gaya dan moment yang ditimbulkan oleh empat buah motor dan propeller [2] ditunjukkan pada persamaan 4 dan 5.

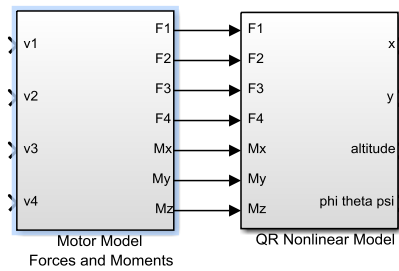
$$F_i = a_i dt_i + b_i \quad (4)$$

$$M_m = c_m dt_m + d_m \quad (5)$$

Secara keseluruhan, persamaan gerak quadrotor terhadap koordinat badan quadrotor (*B-frame*) dapat dituliskan dalam persamaan 6.

$$\begin{aligned} \dot{u} &= (vr - wq) + g \sin \theta \\ \dot{v} &= (wp - ur) - g \cos \theta \sin \phi \\ \dot{w} &= (uq - vp) - g \cos \theta \cos \phi + \frac{U_1}{m} \\ \dot{p} &= \frac{I_{YY} - I_{ZZ}}{I_{XX}} qr + \frac{U_2}{I_{XX}} \\ \dot{q} &= \frac{I_{ZZ} - I_{XX}}{I_{YY}} pr + \frac{U_3}{I_{YY}} \\ \dot{r} &= \frac{I_{XX} - I_{YY}}{I_{ZZ}} pq + \frac{U_4}{I_{ZZ}} \end{aligned} \quad (6)$$

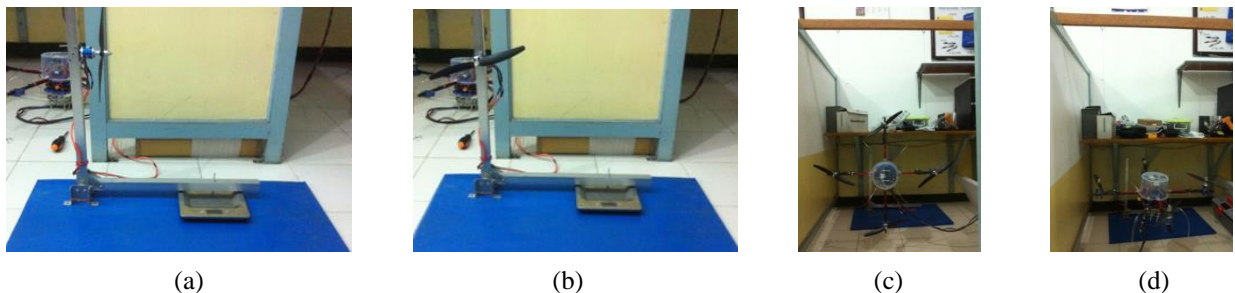
Setelah diketahui model kinematik dan dinamik dari quadrotor, maka langkah selanjutnya adalah melakukan analisis dan perhitungan dari model matematis tersebut menggunakan software MATLAB/Simulink seperti pada Gambar 2.



Gambar 2. Model kinematik dan dinamik quadrotor

### 2.3. IDENTIFIKASI PARAMETER

Identifikasi parameter dari quadrotor dilakukan secara eksperimen. Dalam penelitian ini dilakukan dua tahapan pengukuran untuk identifikasi quadrotor yaitu dengan menghitung korelasi tegangan input dengan output thrust dan moment kemudian mengukur inersia quadrotor menggunakan bifilar pendulum seperti pada Gambar 3.



Gambar 3. (a) Tes stand pengukuran thrust (b) Test stand pengukuran moment (c) Pengukuran roll dan pitch inersia (d) Pengukuran yaw inersia

### 2.4. DESAIN KONTROL PD

Dalam penelitian ini, desain kontrol yang digunakan untuk menstabilkan quadrotor pada kondisi hover adalah *enhanced* kontrol *Proportional-Derivative* (PD). Dengan kontrol PD ini diharapkan dicapai suatu sistem kontrol dengan settling time dan nilai persen overshoot yang kecil. Simulasi desain kontrol PD dilakukan dalam kondisi ideal tanpa gangguan dari luar.

Penggunaan kontrol PD menggunakan persamaan orde kedua. Bentuk standar persamaan orde kedua [4] ditunjukkan pada persamaan 7.

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{7}$$

nilai frekuensi natural yang teredam dapat dihitung dengan menggunakan persamaan 8.

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \tag{8}$$

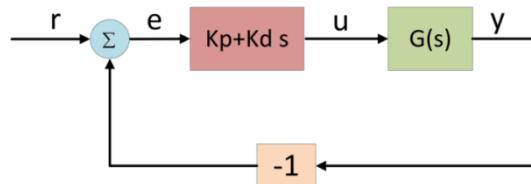
Damping rasio dapat diperoleh dengan menentukan nilai overshoot yang dikehendaki dengan menggunakan persamaan 9.

$$\zeta = \frac{\left| \ln\left(\frac{PO}{100}\right) \right|}{\sqrt{\pi^2 + \left[ \ln\left(\frac{PO}{100}\right) \right]^2}} \tag{9}$$

Nilai frekuensi natural dari frekuensi yang tidak teredam ( $\omega_n$ ) dapat dicari dengan menentukan settling-time ( $t_s$ ) sebesar 2% dari nilai steady state sesuai dengan persamaan 10.

$$\omega_n = \frac{4}{\xi t_s} \tag{10}$$

Untuk memodifikasi rangkaian dinamik sehingga memiliki spesifikasi yang kita kehendaki maka dibuat rangkaian kompensator PD untuk rangkaian SISO dapat dilihat pada Gambar 4.



Gambar 4. PD kompensator

Desain kontrol PD dengan menggunakan fungsi transfer orde kedua dinyatakan dalam persamaan 11.

$$G(s) = \frac{b}{s^2 + a_1 s + a_2} \tag{11}$$

Rangkaian *open loop* dari fungsi transfer dari  $e(s)$  ke  $y(s)$  dapat dilihat dalam persamaan 12.

$$Y(s) = C(s)G(s) = \frac{bk_p + bk_d s}{s^2 + a_1 s + a_2} \tag{12}$$

Bentuk close loop dari fungsi transfer dari  $r(s)$  ke  $y(s)$  ditunjukkan pada persamaan 13.

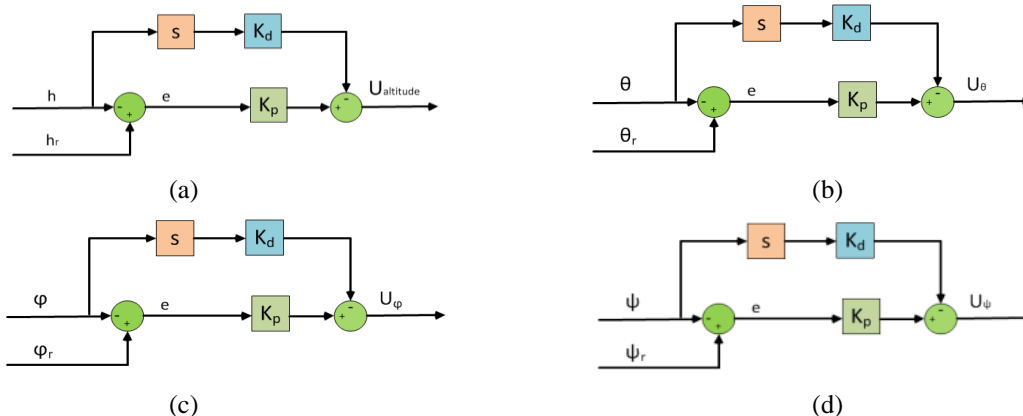
$$\frac{Y(s)}{R(s)} = \frac{b(k_p + k_d s)}{s^2 + (a_1 + bk_d)s + a_2 + bk_p} \tag{13}$$

Sehingga KP dan KD dapat dihitung dengan menggunakan persamaan 14 dan 15.

$$k_p = \frac{\omega_n^2 - a_2}{b} \tag{14}$$

$$k_d = \frac{2\zeta\omega_n - a_1}{b} \tag{15}$$

Dalam quadrotor terdapat empat kondisi gerakan yaitu ketinggian, roll, pitch dan yaw seperti pada Gambar 5.



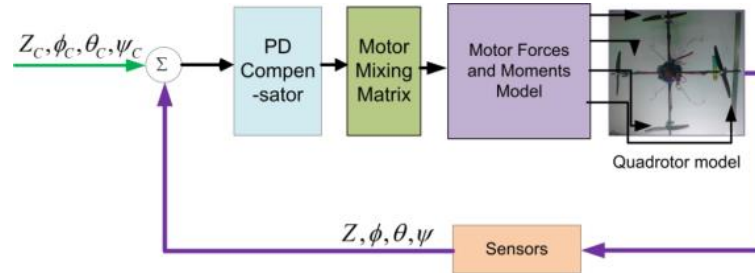
Gambar 5. (a) Diagram blok ketinggian (b) Diagram blok roll (c) Diagram blok pitch (d) Diagram blok yaw

Secara keseluruhan penguatan PD kompensator dengan persen *overshoot* 3% dan *settling time* 2% sebesar 0.7s dapat disimpulkan ke dalam Tabel 1.

**Tabel 1.** Penguatan PD kompensator

Kontrol	$K_P$	$K_D$
Ketinggian	15.3380	2.9780
Roll	2.5010	0.4856
Pitch	2.5875	0.5024
Yaw	28.1519	5.4659

Resultan gain dari sistem quadrotor dinamik nonlinear dengan *multi input multi output* (MIMO) ditunjukkan pada Gambar 6.



**Gambar 6.** Model PD nonlinear

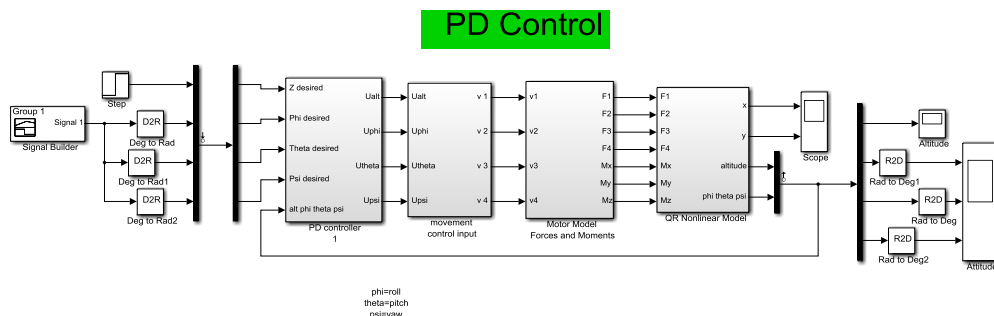
Untuk mengatur respon motor sesuai dengan tegangan masukan yang ada digunakan diagram blok motor mixing matrix [1] yang ditunjukkan pada persamaan 16.

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ 1 & 0 & 1 & 1 \\ 1 & -1 & 0 & -1 \end{bmatrix} \begin{bmatrix} U_{altitude} \\ U_\phi \\ U_\theta \\ U_\psi \end{bmatrix} + \begin{bmatrix} v_{1nom} \\ v_{2nom} \\ v_{3nom} \\ v_{4nom} \end{bmatrix} \quad (16)$$

Besarnya nilai initial voltage untuk setiap motor dapat dihitung dengan menggunakan persamaan 17.

$$v_{i\_nom} = \frac{m_q - b_i}{a_i} \quad (17)$$

Akhirnya model MATLAB/Simulink quadrotor dengan sistem kontrol PD dapat ditunjukkan pada Gambar 7.



**Gambar 7.** PD kontrol quadrotor untuk ketinggian Roll, pitch dan yaw

## 2.5. DESAIN QUADROTOR

Quadrotor terdiri dari beberapa komponen mekanik dan elektronik. Komponen mekanik quadrotor terdiri dari frame, *landing gear* dan propeller. Sedangkan komponen elektronik terdiri dari *brushless DC motor*, *Electronic Speed Control* (ESC), sensor dan *battery*.

Frame quadrotor yang digunakan adalah berbahan fiber karbon yang berbentuk persegi dengan ukuran 1 cm dan terdapat lubang bulat di tengahnya dengan diameter lubang 0.8 cm dan panjang lengan 30 cm dengan konfigurasi terbang plus (+) seperti pada Gambar 8.



**Gambar 8.** Frame quadrotor dengan propeller

*Landing gear* yang digunakan adalah kawat *stainless steel* dengan diameter 3 mm yang dibengkokkan sesuai pada Gambar 9.



**Gambar 9.** *Landing gear* quadrotor

Hal yang perlu diperhatikan dalam memilih propeller adalah gaya thrust yang dihasilkan oleh propeller sebanding dengan kecepatan sudutnya. Dalam penelitian ini menggunakan propeller 10 inch x 6 pitch angle seperti pada Gambar 10.



**Gambar 10.** *Propeller* quadrotor

Dalam penelitian ini menggunakan *Emax BL2220/07 DC motor brushless*. Motor ini membutuhkan tegangan kerja sebesar 11.1 Volt, dimensi 22x20 mm dan berat 85 gram. Motor ini mampu mengangkat beban sebesar 400 – 1200 gram dengan asupan daya yang rendah 1200 Kv yang ditunjukkan pada Gambar 11.



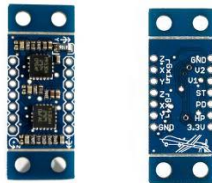
**Gambar 11.** *Emax BL2220/07 DC motor brushless*

*Electronic Speed Control* berfungsi mengatur kecepatan perputaran baling-baling (motor) pesawat, sehingga energi baterai dapat digunakan seefisien mungkin, dan memperlama waktu penerbangan. Untuk ini perlu diperhatikan tegangan yang diperbolehkan dan juga arusnya. Dalam penelitian ini, peneliti menggunakan *ESC VW-H 1250C* yang mampu bertahan pada suhu maksimal 125°C serta mempunyai kapasitas arus maksimum 30A seperti pada Gambar 12.



**Gambar 12.** ESC VW-H 1250C

Sensor *gyroscope* pada Gambar 13 berfungsi untuk memberikan data kecepatan sudut terhadap *body frame*. Sensor ini terdiri dari dua buah IC yaitu LPR530AL untuk sudut pitch dan roll dan LPY530AL untuk sudut yaw; tiga buah *analog output* yang berupa sumbu x, y dan z; 5 buah penguat sinyal. Catu daya yang dibutuhkan adalah 2.7 – 3.6 volt DC.



**Gambar 13.** Gyroscope triple axis 300o/s LPx530AL Board

ADXL335 adalah analog sensor yang berbentuk kecil, tipis, berdaya rendah, lengkap *3-axis accelerometer* dengan sinyal output berupa tegangan AC. Produk ini mampu mengukur percepatan dengan berbagai skala minimal  $\pm 3$  g. Disamping itu dapat juga digunakan untuk mengukur percepatan gravitasi statis di aplikasi penginderaan, seperti akselerasi yang dinamis yang dihasilkan dari gerak, shock, atau getaran seperti pada Gambar 14.



**Gambar 14.** ADXL335 3-axis accelerometer

*Infrared sensor* dalam penelitian ini digunakan untuk mengukur ketinggian dari quadrotor. Sensor ini hanya mampu mengukur pada ketinggian antara 10 – 80 cm. Prinsip kerja sensor ini hanya menghitung waktu pantulan dari bidang pantul yang sebanding dengan jarak. Seperti pada Gambar 15.



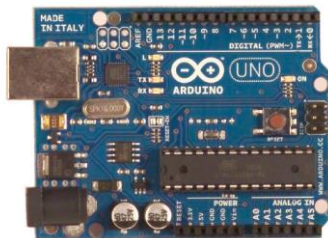
**Gambar 15.** GP2D12 infrared sensor

*Turnigy 2200mAH battery* pada Gambar 16 digunakan sebagai catu daya keempat motor dari quadrotor. Baterai ini memiliki kapasitas arus sebesar 2200mAH, 3 sel battery dan tegangan sebesar 12 volt serta berat baterai sebesar 197 gram. Baterai ini mampu memberikan catu daya ke empat motor dengan masing-masing daya sebesar 1200rpm/volt.



**Gambar 16.** Turnigy 2200mAH battery

Dalam penelitian ini, output dari arduino terdapat *output pin* dengan tegangan 3.3 volt dan 5 volt. *Output pin* 3.3 volt digunakan sebagai catu daya accelerometer dan gyroscope, sedangkan *output pin* 5 volt digunakan untuk catu daya sensor infrared. *Output PWM* digunakan sebagai tegangan masukan yang mengatur kecepatan keempat motor dari quadrotor. Analog output dari arduino digunakan sebagai masukan data dari Humusoft AD622 DAQ card. Seperti pada Gambar 17.



Gambar 17. Arduino uno board

## 2.6. KONFIGURASI HILS

*Humusoft AD622* adalah *data acquisition card* yang digunakan untuk menghubungkan komputer dengan sinyal nyata dari quadrotor. *Humusoft AD622* terdiri dari 8 channel fast 14 bit A/D converter dengan *sampel simultan/hold circuit*, 8 independent 14 bit D/A converter, 8 bit digital input port dan 8 bit digital output port.

Dalam penelitian ini *Humusoft AD 622 DAQ card* digunakan sebagai *input analog* dari sinyal *output sensor accelerometer* (sumbu x, y dan z masing-masing pin 2, 3 dan 4), *gyroscope* (sumbu x, y dan z masing-masing pin 5, 6, dan 7) dan sensor *infrared* (*altitude* pada pin 8). Seperti pada Gambar 18.



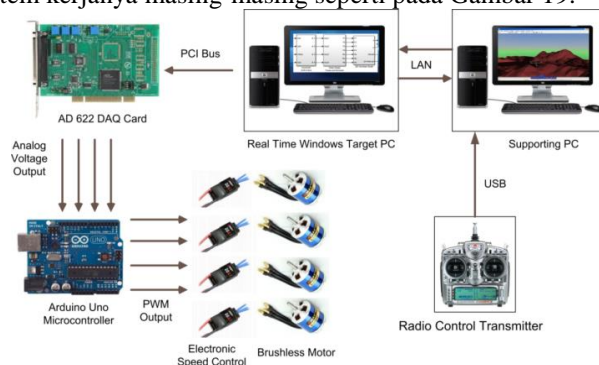
Gambar 18. Humusoft AD622 DAQ card

Dua buah komputer masing-masing berfungsi sebagai *host computer* dan *target computer*. *Host computer* digunakan untuk mengakses file Simulink (*PD control, sensor model, actuator model*). Pada *PC host* dipasang *Humusoft AD622 DAQ Card* sebagai data akuisisi dari plant quadrotor dengan komputer. Sedangkan *target computer* digunakan untuk menampilkan *virtual reality* dari MATLAB/Simulink. Spesifikasi komputer yang digunakan adalah

- PC Host : Intel Pentium core i5, RAM 6GB, HD 500GB, VGA 64MB
- PC Target: Intel Pentium core i3, RAM 4GB, HD 500GB, VGA 1GMB

Koneksi LAN digunakan untuk menghubungkan *PC Host* dengan *PC target*. Saat ini, LAN berbasis pada teknologi *IEEE 802.3 Ethernet*, yang mempunyai kecepatan transfer data 10, 100, atau 1000 Mbit/s. Pemilihan koneksi ini dikarenakan pada MATLAB/Simulink terdapat dua macam koneksi yaitu RS232 dan TCP/IP. Jika menggunakan koneksi RS232 kecepatan transfer data hanya 8 bit/s. Jadi dengan alasan inilah dipilih koneksi TCP/IP untuk transfer data dari *PC Host* ke *PC Target*.

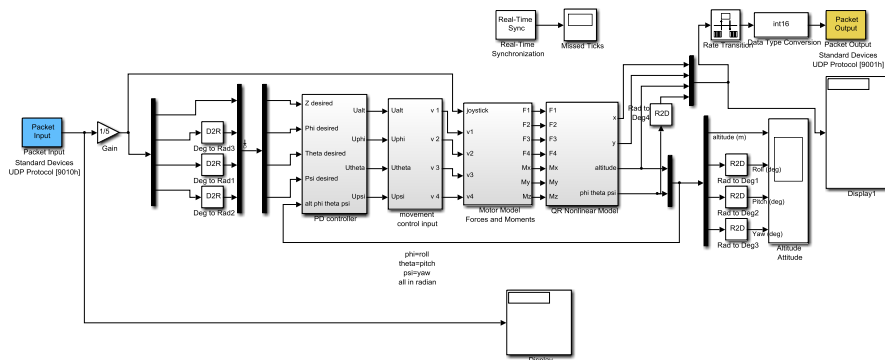
Setelah semua hardware ada maka langkah selanjutnya adalah menghubungkan antara satu hardware dengan yang lainnya sesuai dengan system kerjanya masing-masing seperti pada Gambar 19.



Gambar 19. Konfigurasi hardware HILS

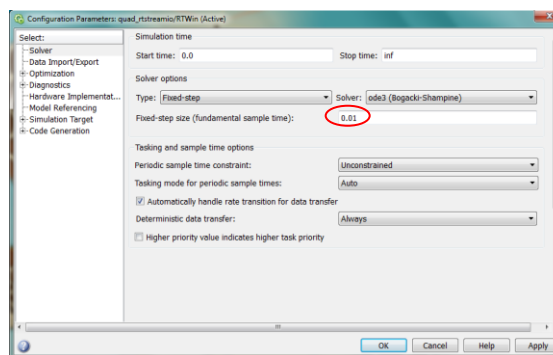


*Realtime Windows Target* merupakan fasilitas yang ada pada MATLAB/Simulink yang digunakan untuk mengeksekusi model pada MATLAB/Simulink pada Microsoft Windows yang terhubung pada I/O board secara real time. *Model plant quadrotor* tersebut dapat dieksekusi oleh *PC Host* dan di tampilan kedalam *PC Target* melalui koneksi LAN. Seperti pada Gambar 20.



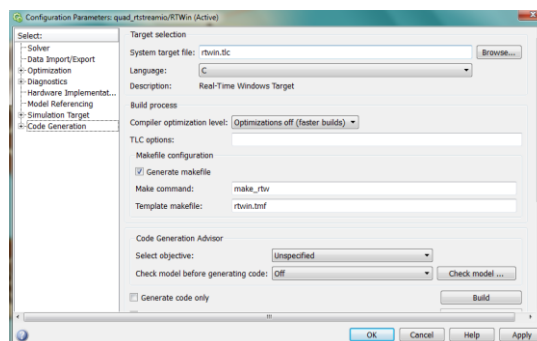
Gambar 20. Plant quadrotor dalam *realtime windows target*

Setelah *plant quadrotor* selesai dibuat maka untuk konfigurasi *Realtime Windows Target* pada *PC Host*, langkah selanjutnya adalah konfigurasi parameter *RTWin* seperti pada Gambar 21 *sampel time* diganti dengan 1/300



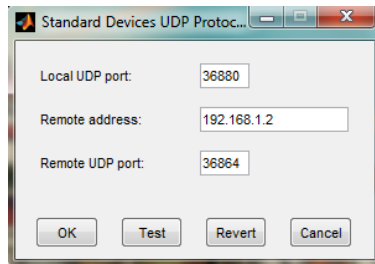
Gambar 21. Konfigurasi parameter *realtime windows target*

Langkah selanjutnya adalah konfigurasi target selection. Pilih *rtwt.tlc* pada *system target file*, kemudian pilih *apply* lalu *build*. Seperti pada Gambar 22.



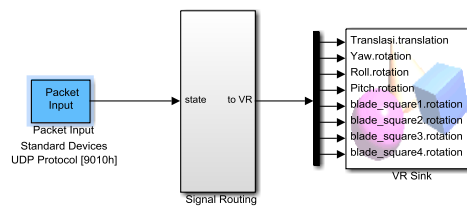
Gambar 22. Konfigurasi target selection *realtime windows target*

Agar *PC Host* dan *PC Target* dapat terhubung, maka perlu adanya konfigurasi *UDP send/receive target* pada *PC Host*. Pada *Local UDP port* diisikan port *UDP PC Host* yaitu 36880. Pada kolom *Remote address* merupakan alamat IP dari *PC target*, sedangkan pada *Remote UDP port* isikan port dari *PC target* seperti pada Gambar 23.



Gambar 23. Konfigurasi *UDP receive target*

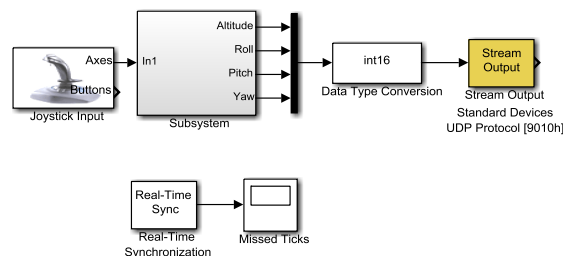
Setelah konfigurasi *PC Host* selesai, maka langkah selanjutnya adalah konfigurasi *PC target*. Dalam *PC target* digunakan model sebagai *virtual reality model*. digunakan untuk menampilkan simulasi dari quadrotor seperti pada Gambar 24.



Gambar 24. Konfigurasi *virtual reality*

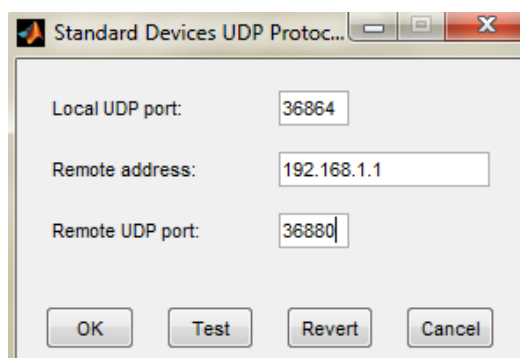
## 2.7. KONFIGURASI PIL

*Input joystick* digunakan sebagai sinyal masukan dari *remote control* dalam aplikasinya sebagai *Pilot In The Loop (PIL)* seperti pada Gambar 25.



Gambar 25. Konfigurasi *input joystick* untuk aplikasi *Pilot In the Loop (PIL)*

Agar *PC Host* dan *PC Target* dapat terhubung, maka perlu adanya konfigurasi *UDP send/receive target* pada *PC Target*. Pada *Local UDP port* diisikan *port UDP PC Target* yaitu 36864. Pada kolom *Remote address* merupakan alamat IP dari *PC Host*, sedangkan pada *Remote UDP port* isikan *port* dari *PC Host* seperti pada Gambar 26.



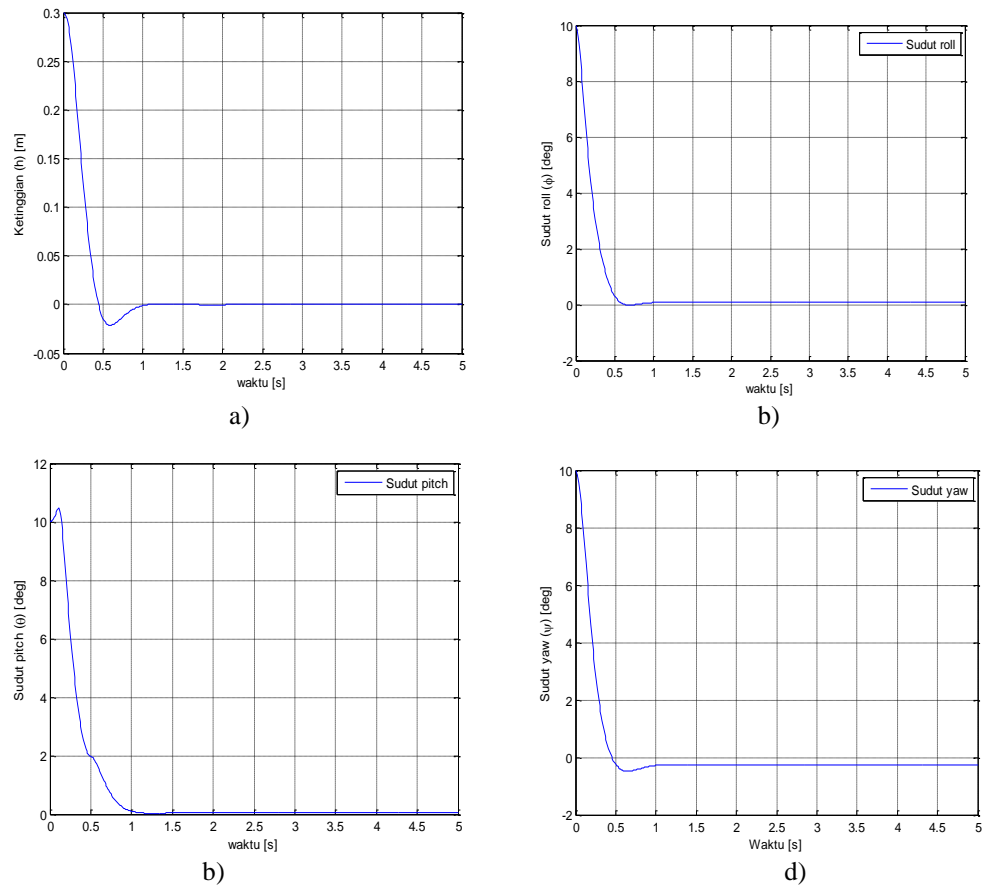
Gambar 26. Konfigurasi *UDP send target*

### 3. HASIL DAN PEMBAHASAN

#### 3.1. HASIL SIMULASI KONTROL PD

Setelah desain kontrol PD dengan menggunakan MATLAB/Simulink selesai dibuat maka diperoleh hasil simulasi untuk respon ketinggian, roll, pitch dan yaw.

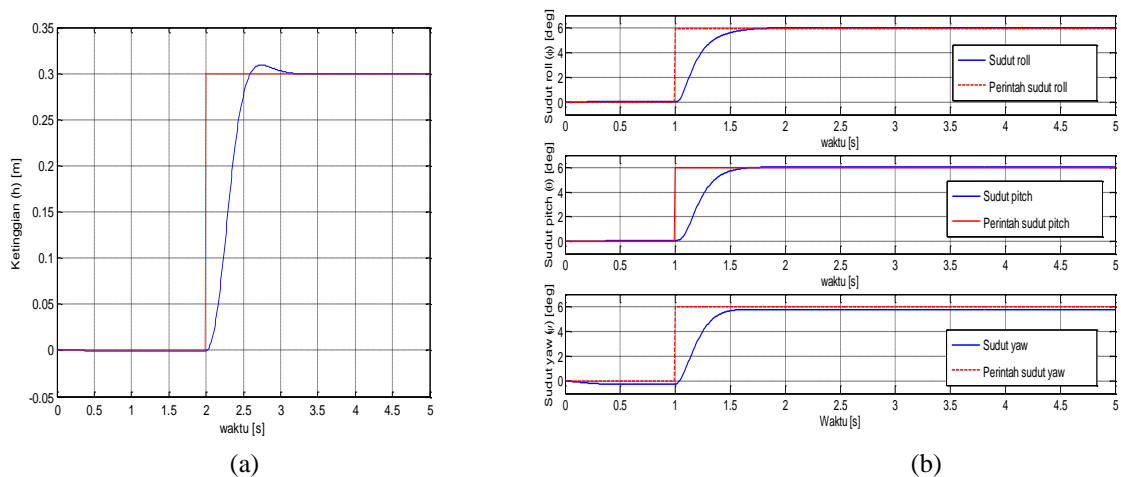
Dalam simulasi yang pertama diberikan inisial kondisi untuk ketinggian sebesar 30 cm dan nilai sudut awal roll, pitch dan yaw sebesar  $10^\circ$  sehingga diperoleh Gambar 27.



**Gambar 27.** Simulasi dengan inisial kondisi (a) Respon ketinggian (b) Respon roll (c) Respon pitch (d) Respon yaw

Dari Gambar 26 diatas dapat dilihat bahwa terdapat *error steady state* sekitar  $-0.1^\circ$  pada respon yaw, sedangkan lainnya tidak terdapat *error steady state*, dengan *settling time* sekitar 1s.

Setelah melakukan simulasi dengan inisial kondisi, maka langkah selanjutnya untuk menguji sistem kontrol yang telah dibuat yaitu dengan memberikan fungsi *step input*. Hasil simulasi dari *input step* yang diberikan kedalam desain kontrol PD dapat dilihat pada Gambar 28.

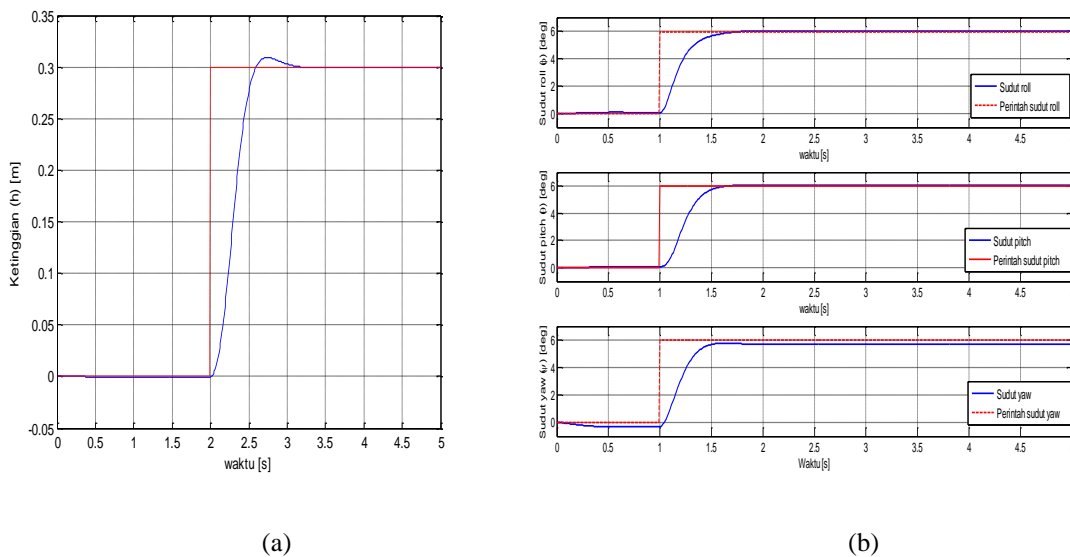


**Gambar 28.** Kontrol PD dengan fungsi step (a) Respon ketinggian (b) Respon roll pitch dan yaw

Dari Gambar 28 diatas dapat dilihat bahwa pada respon ketinggian nilai dari persen *overshoot* sebesar 3.3% dan *settling time* sebesar 0.9s. Sedangkan pada roll, pitch dan yaw, nilai persen *overshoot* adalah 0%, dengan *settling time* sekitar 0.6s. Terdapat *error steady state* sekitar  $-0.1^\circ$  hanya pada respon yaw.

Untuk menguji apakah desain kontrol PD yang telah dibuat kokoh (*robust*), maka dalam penelitian ini dilakukan simulasi ketidakpastian parameter yaitu massa dan inersia masing-masing bertambah sebanyak 10% dari kondisi awal, dan pengurangan masa dan inersia sebanyak 10% dari kondisi awal. Hal ini dilakukan karena pada saat kondisi real, besar masa dan inersia dipastikan akan mengalami ketidakpastian karena faktor eksternal.

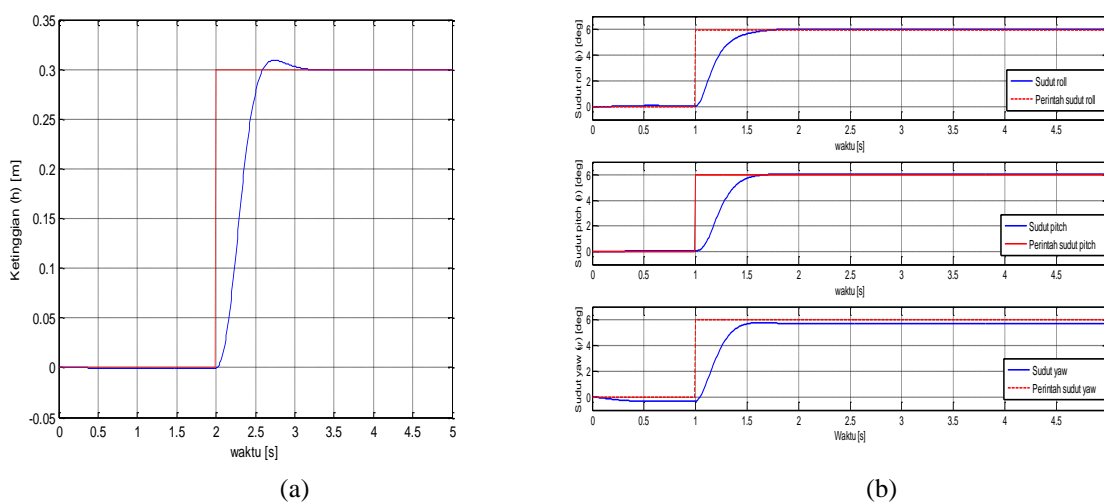
Dengan penambahan masa sebesar 10% dan inersia sebesar 10% dari kondisi awal yaitu masa = 1.68 kg,  $I_{xx}= 0.0264 \text{ kgm}^2$ ,  $I_{yy}= 0.0264 \text{ kgm}^2$  dan  $I_{zz}= 0.0510 \text{ kgm}^2$  dapat dilihat pada Gambar 29.



**Gambar 29.** Penambahan masa 10% dan inersia 10 (a) Respon ketinggian (b) Respon roll pitch dan yaw

Dari Gambar 29 diatas dapat dilihat bahwa pada respon ketinggian nilai dari persen *overshoot* sebesar 3.3% dan *settling time* sebesar 0.9s. Sedangkan pada roll, pitch dan yaw, nilai persen *overshoot* adalah 0%, dengan *settling time* sekitar 0.6s. Terdapat *error steady state* sekitar  $-0.1^\circ$  hanya pada respon yaw.

Dengan pengurangan masa sebesar 10% dan inersia sebesar 10% dari kondisi awal yaitu masa = 1.68 kg,  $I_{xx}= 0.0264 \text{ kgm}^2$ ,  $I_{yy}= 0.0264 \text{ kgm}^2$  dan  $I_{zz}= 0.0510 \text{ kgm}^2$  dapat dilihat pada Gambar 30.

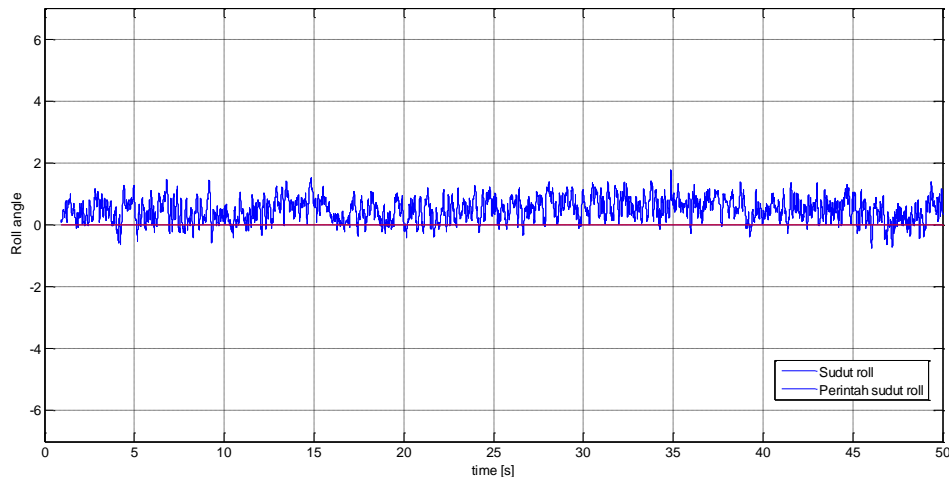


**Gambar 30.** Pengurangan masa 10% dan inersia 10% (a) Respon ketinggian (b) Respon roll pitch dan yaw

Dari Gambar 30 diatas dapat dilihat bahwa pada respon ketinggian nilai dari persen *overshoot* sebesar 5% dan *settling time* sebesar 1.1 s. Sedangkan pada roll, pitch dan yaw, nilai persen *overshoot* adalah 0%, dengan *settling time* sekitar 0.6s. Terdapat *error steady state* sekitar  $-0.1^\circ$  hanya pada respon yaw.

### 3.2. HASIL PENERAPAN KONTROL PD DENGAN HILS

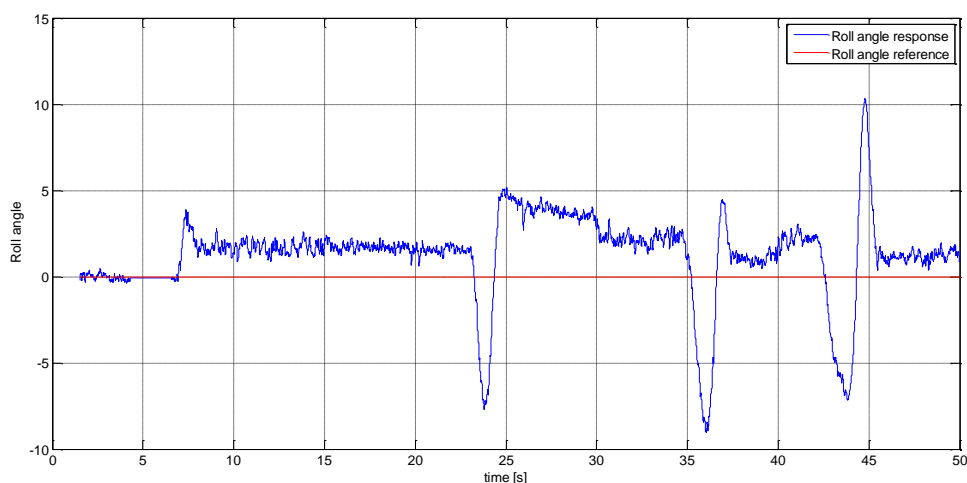
Hasil penelitian dalam penerapan kontrol PD pada *plant quadrotor*, untuk memperoleh nilai *gain* yang paling sesuai dilakukan *tuning gain*. *Tuning gain* ini dilakukan dengan cara melakukan *adjusting* nilai dari konstanta kontrol proporsional (KP) dan konstanta kontrol derivative sehingga diperoleh nilai 0.4 untuk KP dan 0.2 untuk KD. Respon hasil dari perintah roll untuk *zero reference* ditunjukkan pada Gambar 31.



Gambar 31. Aplikasi HILS untuk respon roll dengan *zero reference*

Dari gambar 30 diatas dapat dilihat bahwa terdapat noise pada pembacaan perintah roll sekitar  $1^\circ$  dan terdapat *error steady state* sekitar  $0.5^\circ$ .

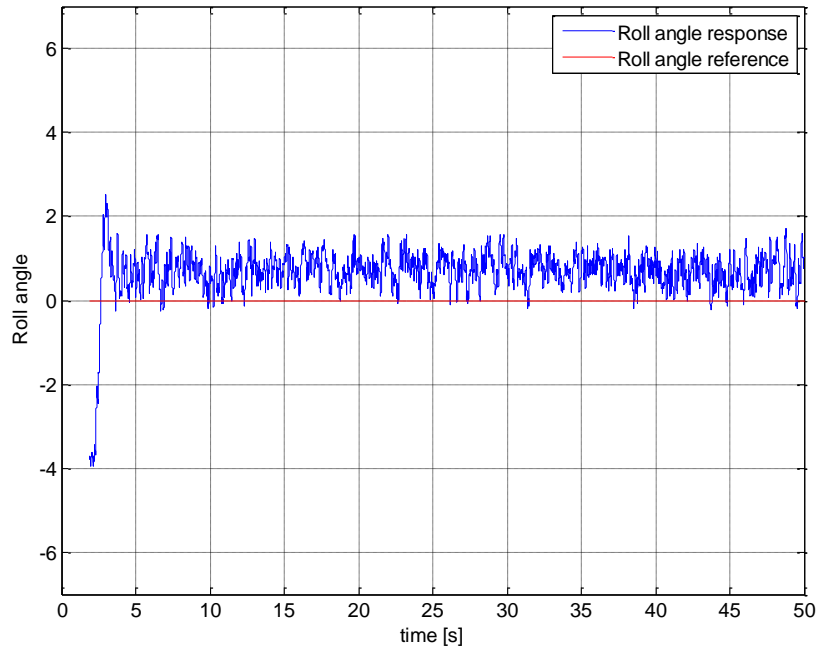
Setelah melakukan pengujian respon quadrotor terhadap *zero reference* maka langkah selanjutnya untuk menguji kekokohan dan unjuk kerja dari sistem kontrol PD yang telah dibuat maka dilakukan pengujian dengan memberikan gangguan yang berupa gaya dari luar yaitu dengan menarik salah satu frame quadrotor ke arah sumbu z negatif (bawah) dan melepaskannya kembali. Apakah quadrotor akan kembali ke posisi semula (titik nol). Seperti pada Gambar 32.



Gambar 32. Aplikasi HILS untuk respon roll dengan *disturbance effect*

Dari Gambar 32 diatas maka dapat dilihat bahwa quadrotor akan kembali menuju ke posisi semula (titik nol) setelah diberikan gangguan dari luar, tetapi terdapat *error steady state* sekitar  $1^\circ$ .

Setelah melakukan pengujian respon quadrotor terhadap *disturbance effect* maka langkah selanjutnya untuk menguji kekokohan dan unjuk kerja dari sistem kontrol PD yang telah dibuat maka dilakukan pengujian kembali dengan memberikan inisial kondisi sebesar  $5^\circ$ . Apakah quadrotor akan kembali ke posisi semula (titik nol). Seperti pada Gambar 33.

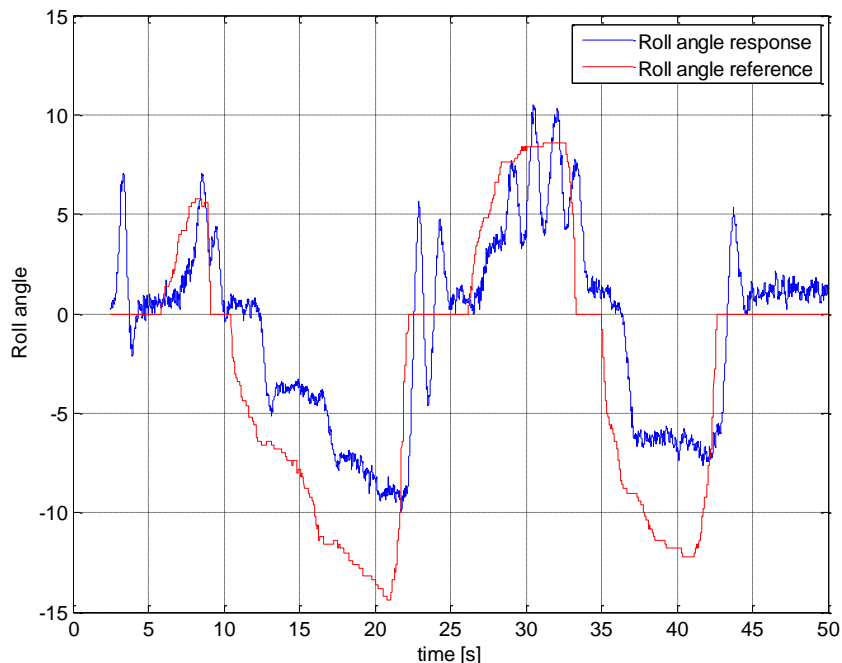


**Gambar 33.** Aplikasi HILS untuk respon roll dengan inisial kondisi  $5^\circ$

Dari Gambar 33 diatas maka dapat dilihat bahwa quadrotor akan kembali menuju ke posisi semula (titik nol) setelah diberikan inisial kondisi sebesar  $5^\circ$ , tetapi terdapat *error steady state* sekitar  $0.5^\circ$ .

### 3.3. HASIL PENERAPAN KONTROL PD DENGAN PIL

Setelah semua pengujian dilakukan maka langkah selanjutnya adalah penerapan aplikasi *Pilot In the Loop* untuk kontrol PD yang telah dibuat. Seperti pada Gambar 34.



**Gambar 34.** Aplikasi PIL untuk respon roll

Dalam aplikasi PIL ini dapat dilihat pada Gambar 34 bahwa respon roll quadrotor sesuai dengan perintah yang diberikan, namun terdapat *error steady state* dan osilasi pada control PD.

#### 4. KESIMPULAN

- 1) Sesuai dengan simulasi, sistem kontrol PD dengan penambahan dan pengurangan masa dan inersia tidak pada respon roll, pitch dan yaw tidak mengalami perubahan nilai persen *overshoot* dan *settling time* dan *error steady state*, sedangkan pada respon ketinggian mengalami perubahan pada persen *overshoot* sebesar 1.7% dari nilai awal.
- 2) Dalam simulasi pada respon yaw terdapat *error steady state* sebesar  $0.1^\circ$
- 3) Dalam aplikasi Pilot In the Loop, terdapat *error steady state* pada respon roll sekitar  $2^\circ$  serta terjadi osilasi saat system akan kembali ke posisi nol.

#### 5. REFERENSI

- [1] Vaibav Gadioc. 2011. "Autonomous Aerial Manipulation Using A Quadrotor" Utah State University, 4:37-66.
- [2] Anil Guclu. 2012. "Attitude and Altitude Control of An Outdoor Quadrotor" Atılım University, 3:10-32.
- [3] Robert L. Williams. 2007. "Linear state-space control system", Ohio University.
- [4] James Diebel. 2006. "Representing Attitude: Euler Angles, Unit Quaternion dan Rotation Vectors", Stanford University.
- [5] L.P. Pook. 2011. "Understanding Pendulums", Laboratory of Robotics and Mecatronics, University of Cassino.
- [6] [http://www.aeroflyhobbies.com/product.php?id\\_product=1049](http://www.aeroflyhobbies.com/product.php?id_product=1049), diakses: 28 Juni 2013.
- [7] <http://www.hobbyking.com/hobbycity/store/US.asp?schUS=propeller+10x6>, diakses: 28 Juni 2013.
- [8] [http://www.aeroflyhobbies.com/product.php?id\\_product=972](http://www.aeroflyhobbies.com/product.php?id_product=972), diakses: 28 Juni 2013.
- [9] [http://www.sgbotic.com/index.php?dispatch=products.view&product\\_id=710](http://www.sgbotic.com/index.php?dispatch=products.view&product_id=710), diakses: 28 Juni 2013.
- [10] [http://www.sgbotic.com/index.php?dispatch=products.view&product\\_id=440](http://www.sgbotic.com/index.php?dispatch=products.view&product_id=440), diakses: 28 Juni 2013.

#### 6. UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Kementerian Riset dan Teknologi, Republik Indonesia, kontrak penelitian Nomor : 38.6/UN7.5/PG/2012 dan Nomor: 39.4/UN7.5/PG/2013 Insentif Riset Sinas