

PERANCANGAN SISTEM KENDALI KETINGGIAN ROBOT TERBANG QUADCOPTER MENGGUNAKAN SENSOR ULTRASONIK SRF-04

Pamor Gunoto¹⁾, Samuel Suryanata²⁾

^{1,2)}Program Studi Teknik Elektro, Fakultas Teknik, Universitas Riau Kepulauan Batam
E-mail : pamorgunoto@ft.unrika.ac.id¹⁾, samuel@gmail.com²⁾

ABSTRAK

Quadcopter merupakan robot terbang jenis UAV *rotary-wing* yang dewasa ini terus dikembangkan, baik oleh personal atau individu maupun oleh organisasi dan perusahaan besar. Karena fungsi dan kegunaan dari *quadcopter* sendiri yang belum ditemukan seluruhnya, yang membuat *quadcopter* menjadi primadona di dunia robot saat ini. Sensor ultrasonik dipasang pada bagian bawah *quadcopter* untuk mengetahui jarak antara *quadcopter* dengan tanah. Kontrol PID (Proportional, Integral, Derivative) merupakan pengendali konvensional yang digunakan untuk menentukan presisi suatu sistem dengan karakteristik adanya umpan balik pada sistem tersebut. Dengan menggunakan kontrol PID ini diharapkan *quadcopter* dapat terbang secara otomatis tanpa pilot dan dapat melayang di udara sesuai setpoint yang telah ditentukan yaitu sebesar 50 cm. Untuk menentukan konstanta Kp, Ki dan Kd digunakan metode trial and error. Hasil pengujian terbang yang didapatkan dalam penelitian ini adalah mempunyai nilai Kp = 1, Kd = 0.01 dan Ki = 0.1. *Quadcopter* dapat terbang secara stabil sesuai dengan yang diharapkan.

Kata kunci : *Quadcopter, Sensor, PID, Trial and Error*

ABSTRACT

Quadcopter is a rotary-wing UAV type flying robot that is currently being developed, both by individuals and individuals as well as by large organizations and companies. Because the functions and uses of the quadcopter itself have not been found entirely, which makes the quadcopter to be excellent in today's robot world. An ultrasonic sensor is mounted at the bottom of the quadcopter to determine the distance between the quadcopter and the ground. The PID control (Proportional, Integral, Derivative) is a conventional controller that is used to determine the precision of a system with the characteristics of feedback on the system. By using the PID control, it is expected that the quadcopter can fly automatically without a pilot and can float in the air according to a predetermined setpoint of 50 cm. To determine the constants Kp, Ki and Kd used the trial and error method. The flight test results obtained in this studies have the values of Kp = 1, Kd = 0.01 and Ki = 0.1. The quadcopter can fly stably as expected.

Keyword: *Quadcopter, Sensor, PID, Trial and Error*

1. PENDAHULUAN

Robot merupakan teknologi yang dikembangkan untuk membantu manusia dalam melakukan pekerjaan tertentu, misalnya pekerjaan yang membutuhkan ketelitian tinggi, berisiko tinggi, pekerjaan yang membutuhkan tenaga besar, ataupun pekerjaan yang berulang-ulang dan monoton.

Saat ini penilitan tentang UAV (*Unmanned Aerial Vehicle*) telah berkembang sangat pesat di dunia. Hal ini dikarenakan kegunaan UAV yang sangat penting namun sulit dalam pengendaliannya. Sebagai contoh UAV digunakan sebagai pesawat pengintai, pengendalian dilakukan jarak jauh serta bentuk UAV yang relatif kecil mengakibatkan mudah

terganggu oleh angin. Berbagai bentuk UAV telah dirancang dan salah satunya adalah *quadcopter*.

Quadcopter dapat terbang secara mandiri tanpa operator darat yang mengontrol melalui remote. Sehingga pada aplikasinya *quadcopter* dapat diandalkan sebagai kurir pengantar paket otomatis. Pihak penyedia layanan hanya tinggal memasukan alamat kedalam program dan *quadcopter* akan terbang menghampiri alamat tersebut.

2. TINJAUAN PUSTAKA

2.1 *Quadcopter*

Quadcopter adalah jenis pesawat udara aerodinamis bermotor dan bersayap putar (*rotary wing*). Sebagaimana dengan namanya *Quad* yang berarti empat, sehingga pada *quadcopter* memiliki 4 buah motor *brushless* dan 4 buah baling-baling (*propeller*).

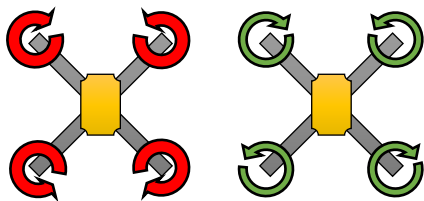
Berdasarkan bentuk rangka (*frame*), *quadcopter* terdiri dari 2 model yaitu model dengan *frame* berbentuk “*plus*” (+) dan model dengan *frame* berbentuk “silang” (x).

2.1.1 Prinsip Kerja *Quadcopter*

Pada ujung setiap *frame* terpasang 4 buah motor *brushless* dan *propeller*, Motor ini yang akan memutar *propeller*. Dengan mengatur kecepatan putaran dan mengatur arah putaran kedalam, sehingga menghilangkan gaya sentrifugal disekitar *quadcopter* mengakibatkan *quadcopter* bisa terangkat/terbang dan bermanuver. Ada 4 pengaturan dasar kecepatan putaran *propeller*. Pengaturan *throttle* *propeller*, pengaturan *pitch* *propeller*, pengaturan *yaw* *propeller*, dan pengaturan *roll* *propeller*.

a. *Throttle*Control

Pengaturan ini bertujuan untuk melakukan *vertical take-off* and *landing* (VTOL) bergerak ke atas dan ke bawah.



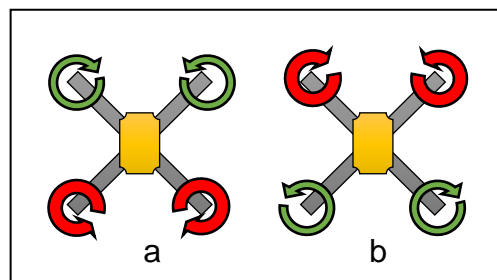
Gambar 1. *Throttle* Control

Quadcopter merupakan robot terbang yang memiliki empat buah motor yang dipasang di setiap sudut sisinya. Biasanya motor yang digunakan adalah motor *brushless*. Karena memiliki lebih dari 1 motor maka dibutuhkan suatu kontrol ketinggian yang dapat membuat ke-4 motor bekerja secara harmonis dan menjaga agar *quadcopter* dapat tetap melayang diudara dengan stabil walaupun ada halangan yang mengganggu.

Putaran yang berwarna merah menandakan *propeller* berputar dengan cepat yang mengakibatkan *quadcopter* akan terangkat (*take-off*), sedangkan putaran yang berwarna hijau menandakan *propeller* berputar dengan lambat yang mengakibatkan *quadcopter* akan turun/mendarat (*landing*).

b. *Pitch* Control

Pengaturan *pitch propeller* adalah pengaturan yang diberikan agar *quadcopter*bergerak maju dan bergerak mundur. Berikut gambar 2 yang menampilkan ilustrasi kecepatan putar *propeller* untuk pengaturan *pitch*.

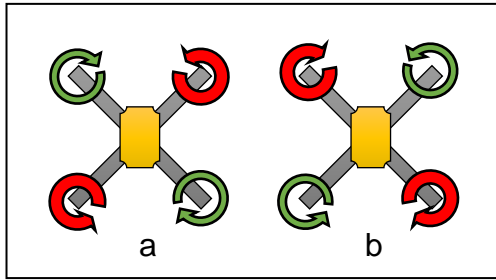


Gambar 2. *Pitch* Control

Pada gambar 2a menampilkan ilustrasi *quadcopter* bergerak maju, sedangkan gambar 2b menampilkan ilustrasi *quadcopter* bergerak mundur.

c. *Yaw* Control

Pengaturan *yaw control* bertujuan agar *quadcopter* bergerak berputar kekiri dan bergerak ke kanan. Ilustrasinya dapat dilihat pada gambar 3 berikut ini.

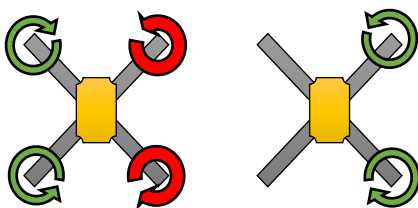


Gambar 3. Yaw Control

Pada gambar 3a menampilkan ilustrasi pergerakan *quadcopter* bergerak memutar kekiri (*left rotate*) dan gambar 3b menampilkan ilustrasi pergerakan *quadcopter* memutar kekanan (*right rotate*).

d. Roll Control

Pengaturan *roll propeller* diberikan agar *quadcopter* bergerak kekiri dan kekanan. Ilustrasi pengaturan roll propeller dapat dilihat pada gambar di bawah ini.

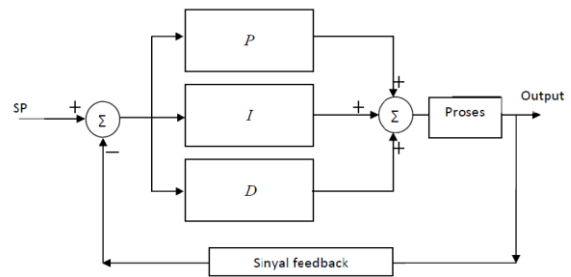


Gambar 4. Roll Control

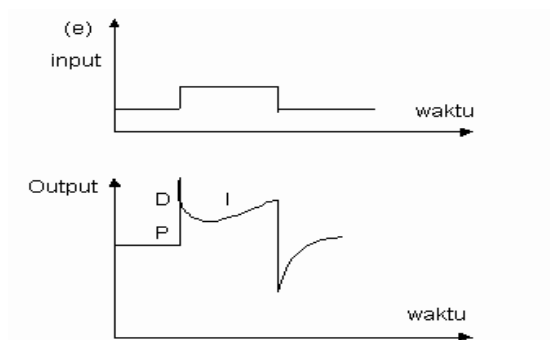
Pada gambar 4a menampilkan ilustrasi gerak kekiri, sedangkan gambar 4b menampilkan ilustrasi gerak kekanan.

2.2 Aksi Kontrol PID

Kontroler PID merupakan kontroler yang cukup kompleks dengan kombinasi kontroler P, kontroler I, dan kontroler D. Kontroler PID dapat menghilangkan *error steady state* dan membuat respon sistem menjadi lebih cepat. Gambar 5 adalah diagram blok PID dan dalam fungsi waktu antara sinyal keluaran dengan masukan untuk kontroler PI D dapat dilihat pada Gambar 6.



Gambar 5. Diagram blok kontrol PID



Gambar 6. Fungsi waktu masukan dan keluaran PID

2.3 Tuning PID

Tuning PID yang digunakan adalah metoda *trial and error* yang dimodifikasi menggunakan metoda *Ziegler-Nichols II*. Berikut adalah masing-masing metode sebelum dimodifikasi yaitu metoda *atrial and error* dan metoda *Ziegler-Nichols II*.

a. Ziegler Nichols II (Metoda Osilasi)

Metode ini didasarkan pada reaksi sistem *closed loop*. Plant disusun serial dengan kontroler PID. Semula parameter integrator dibuat tak berhingga dan parameter diferensial dibuat nol ($T_i = \infty$; $T_d = 0$).

Parameter proporsional kemudian dinaikkan bertahap. Mulai dari nol sampai mencapai harga yang mengakibatkan reaksi sistem tepat berosilasi (*Sustain oscillation*).

Penalaan parameter PID didasarkan terhadap kedua konstanta hasil eksperimen, K_u dan P_u . *Ziegler dan Nichols* menyarankan penyetelan nilai parameter K_p , T_i , dan T_d berdasarkan rumus yang

diperlihatkan pada Tabel dibawah ini.

Tipe Kontroler	K_p	T_i	T_d
P	$0,5 K_u$		
PI	$0,45 K_u$	$1/2 P_u$	
PID	$0,6 K_u$	$0,5 P_u$	$0,125 P_u$

b. Trial and Error

Metoda trial and error disebut juga metoda coba-coba. Metoda ini umumnya cocok digunakan untuk beberapa sistem kontrol, salah satunya adalah sistem control PID. Dalam merancang sistem control PID output yang diharapkan dari metoda trial and error harus memenuhi beberapa kriteria seperti.

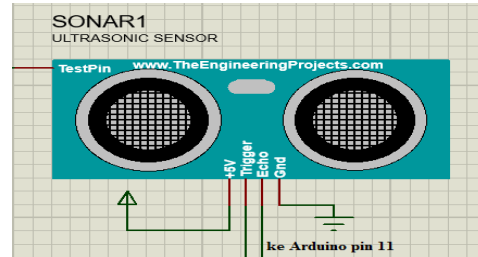
1. Memiliki *risetime* dan *settling time* yang cepat.
2. Tidak memiliki *steadystate error*.
3. *Overshoot* sekecil mungkin.

Langkah-langkah metode *trial and error*:

1. Menentukan nilai K_p yang keluarannya mendekati *setpoint* dengan nilai $T_i = \infty$ dan nilai $T_d = 0$.
2. Setelah diperoleh nilai K_p yang sesuai, tentukan nilai T_i yang diperoleh keluaran tunak yang lebih mendekati *set point*.
3. Menentukan nilai T_d agar sistem lebih peka terhadap *error* (Ogata, K. 1996).

2.4 Sensor Jarak Ultrasonic

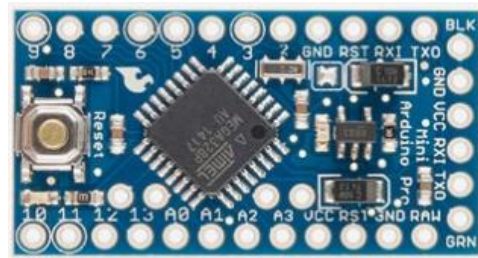
Sensor jarak ultrasonic sangat cocok dipakai untuk aplikasi-aplikasi yang perlu dilakukan pengukuran jarak. Selain itu, sensor ini juga bias bermanfaat dalam sistem *security* dan dapat digunakan sebagai alternatif pengganti sensor *proximity*. Sensor jarak ultrasonic yang dipakai dalam penelitian ini adalah sensor SRF05. Sensor SRF05 memiliki kemampuan membaca jarak dari 3 cm sampai 4 meter. Sensor SRF05 memiliki 2 mode untuk penggunaannya.



Gambar 7. Sensor ultrasonic

2.5 Mikrokontroler Arduino Pro Mini ATmega 328

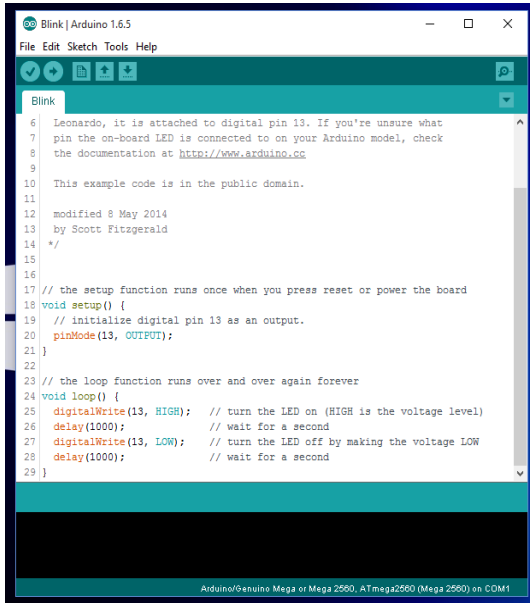
Arduino pro mini merupakan sebuah *converter* mikrokontroler berbasis ATmega 328. Modul ini memiliki 14 *digital input/output* dimana 6 digunakan untuk PWM *output* dan 6 digunakan sebagai *analog input*, 16 MHz osilator Kristal, koneksi PIN, dan tombol *reset*. Memiliki *flash memory* sebesar 32KB sangat cukup untuk menampung program medium. Arduino pro mini tidak memerlukan *flash program external* karena di dalam *chip* mikrokontroler Arduino telah diisi dengan *bootloader* yang membuat proses *upload* program yang kita buat menjadi lebih sederhana dan cepat. Untuk koneksi dengan komputer harus menggunakan perangkat tambahan yaitu USB to TTL Converter PL2303 karena pada *converter* ini Arduino sebagai *developer* menghilangkan *port* USB agar menjaga dimensi dari *converter* ini tetap ringkas.



Gambar 8. Arduino Pro Mini

2.6 Perangkat Lunak (Arduino IDE)

Lingkungan *open-source* Arduino memudahkan untuk menulis kode dan meng-*upload* ke *converter* Arduino. Ini berjalan pada Windows, Mac OS X, dan Linux. Berdasarkan Pengolahan, *avr-gcc*, dan perangkat lunak sumber terbuka lainnya.



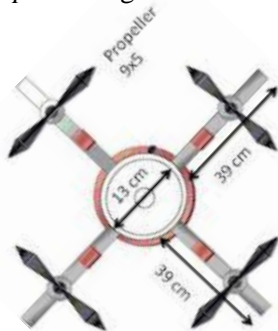
Gambar 9. Arduino IDE

3. METODOLOGI PENELITIAN

Pada penelitian ini metode penelitian terdiri dari metode perancangan dan metode pengujian.

3.1 Perancangan Alat

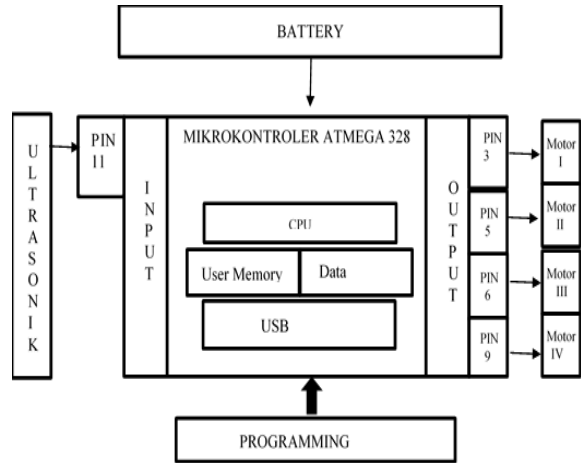
Perancangan *hardware* dimulai dengan pembuatan sketsa alat dengan bantuan *software komputer*, sehingga di dapatkan sketsa *quadcopter* sebagai berikut.



Gambar 10. Perancangan *Quadcopter*

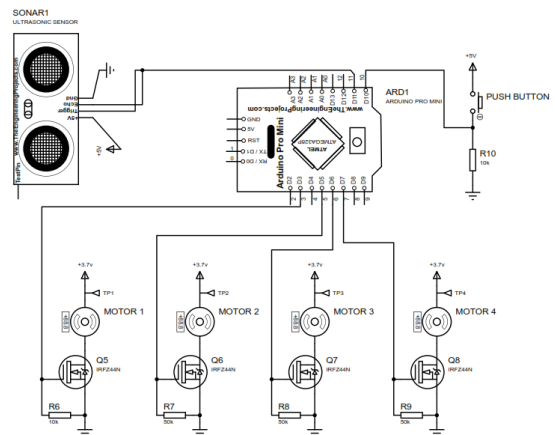
Perancangan sistem yang ada pada *quadcopter* terdiri atas *Input* (sensor ketinggian yang diukur menggunakan sensor ultrasonik),

Process (mikrokontroler *Arduino Pro Mini*) dan *Output*(micromotor DC) dapat digambarkan dalam bentuk blok digram pada gambar 11.



Gambar 11. Diagram blok sistem *quadcopter*

Desain sistem rangkaian elektronik dimulai dari pemilihan modul Arduino dan menghubungkan ke-4 motor yang sebelumnya telah dirangkai dengan ESC yang juga berfungsi sebagai driver motor. Dan yang terakhir menyambungkan pin ultrasonik dan baterai sebagai power supply.



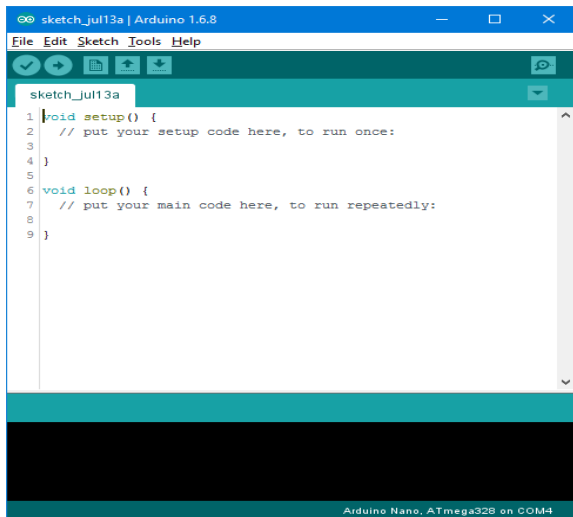
Gambar 12. Desain rangkaian utama

3.2 Perancangan Sistem

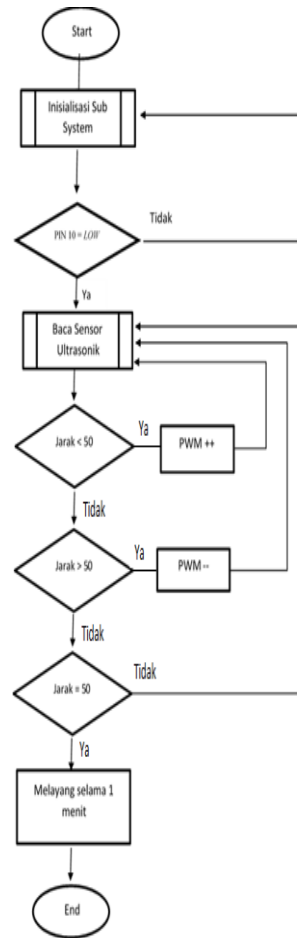
Sistem yang dirancang merupakan simulasi dengan menggunakan kontrol PID dalam penerapan sebagai output dari software kepada motor dc brushless. *Tuning* PID pada *quadcopter* dimaksudkan agar

quadcopter memiliki *performance* yang maksimal sesuai yang diharapkan. Hasil tuning PID akan digunakan agar jarak *quadcopter* dengan tanah tetap dijaga berkisar 50 cm.

Karena menggunakan *Arduino Pro Mini* maka bahasa pemrograman yang digunakan adalah bahasa C# dan menggunakan software *Arduino IDE*. *Arduino IDE* merupakan software *Open Source* yang bebas diunduh dari situs resminya di www.Arduino.cc. Tampilan *software* ditunjukkan seperti gambar 12. *Flowchart* program ditunjukkan pada gambar 13.



Gambar 12. Tampilan program Arduino IDE



Gambar 13. Flowchart sistem

4. HASIL DAN PEMBAHASAN

Pengumpulan dan analisa data berdasarkan hasil pengujian terhadap alat yang telah dilakukan. Pengujian dilakukan untuk memastikan bahwa alat yang bekerja sesuai yang diharapkan pada bab perancangan sebelumnya.

4.1 Pengujian Sensor Ultrasonik

Data yang didapatkan dari pengujian alat dikelompokkan kedalam tabel-tabel agar mudah dilakukan analisa sehingga dapat dijadikan acuan untuk menarik kesimpulan yang sesuai. yang bervariasi di tentukan dengan bantuan busur derajat sedangkan hasil bacaan sensor dilihat dari *display* terminal *virtual* dari *Arduino IDE* Tabel pengujian yang dilakukan dapat dilihat pada tabel 1.

Tabel 1. Tabel pengujian sensor ultrasonik

Jarak (cm)	Sudut Pandang Sensor Terhadap Tanah (Derajat)	Nilai Bacaan Sensor Ultrasonik (cm)
50	130	59
50	125	55
50	90	50
50	60	53
50	50	55

Pada Tabel 1 hasil bacaan sensor ultrasonik kurang akurat pada sudut lebih dari 120° dan kurang dari 70° , yaitu untuk kemiringan mencapai $+30^{\circ}$ tegak lurus terhadap tanah (tegak lurus kita asumsikan 90°) dan -40° tegak lurus terhadap tanah, hasil pembacaan sensor mulai kurang akurat.

4.2 Tuning PID pada set point 50 cm

Pada pengujian ini akan diambil beberapa data untuk *set point*(50 cm) Pengujian mencari nilai parameter K_p yang sesuai dengan membuat *quadcopter* beresilasi dengan cara menentukan mengubah parameter gain *Proportional* (K_p), mengabaikan parameter *Derivative* (K_d) dan *Derivative* (K_i) terlebih dahulu. Gain *Proportional* (K_p) menguatkan *output* agar *respon* mendekati input(*set point*) dan memelihara *system* untuk selalu dalam keadaan stabil.

Setelah didapat K_p , K_i , K_d pengujian selanjutnya adalah pengujian data-data hasil *tuning* terbaik pada kontroler P,PI dan PID akan dibandingkan setelah didapatkan nilai parameter yang terbaik. Yang dimaksudterbaik disini adalah yang memiliki *settling time* tercepat dan *error steady state* terkecil. Tabel 2 diperoleh nilai tuning pada kontrler PID.

Tabel 2. Nilai tuning kontroler PID

K_p	K_d	K_i	<i>Respon</i> Sudut ketinggian
1	0	0	<i>Respon</i> lambat dan terjadi osilasi
1	0	1	<i>Respon</i> terlihat meningkat namun masih terjadi osilasi
1	1	1	<i>Respon</i> cukup cepat tapi belum sesuai yang diinginkan dan osilasi juga terjadi
1	0.1	1	<i>Respon</i> sesuai dengan kriteria yang diinginkan namun tetap terjadi osilasi
1	0.5	0.5	<i>Respon</i> cepat tapi masih terjadi osilasi
1	0.1	0.1	<i>Respon</i> cepat dan terjadi <i>overshoot</i> .
1	0.01	0.1	<i>Respon</i> cepat sesuai yang diinginkan dan tanpa <i>overshoot</i>

Kondisi awal *quadcopter* pada posisi nol (0°) diletakan di atas tanah, sehingga diharapkan *quadcopter* dapat menyesuaikan *set point* yang diberikan dalam rentan waktu tertentu yang dirasa sesuai. Hasil *tuning* yang didapatkan dapat dilihat dari tabel 2 yaitu $K_p = 1$, $K_d = 0.01$ dan $K_i = 0.1$. Dari *respon* grafik ketinggian yang ditampilkan pada gambar 5.1 terlihat *quadcopter* dapat menjaga posisi pada sekitar *set point* yang diatur sekitar ketinggian 50cm.

4.3 Pengujian Sistem *Quadcopter*

Pengujian ini dilakukan untuk membuktikan bahwa *systemhardware* dan *software* (program) telah berjalan dengan sesuai. Pengujian dilakukan sebanyak 3 kali dengan jarak yang berbeda. Jarak ultrasonik didapatkan dari tampilan *virtual terminal Arduino IDE*, PWM di atur melalui program sebelum program di *upload*, dan tegangan pada ke-4 *motordan* hasilnya dapat dilihat pada tabel 3.

Tabel 3. Pengujian sistem *quadcopter*

Pengujian	Jarak Ultrasonik (cm)	PWM	Tegangan pada (Volt)			
			Motor 1 (Volt)	Motor 2 (Volt)	Motor 3 (Volt)	Motor 4 (Volt)
1	30	254	3.5	3.5	3.5	3.5
2	50	200	3.2	3.2	3.2	3.2
3	70	179	2.9	2.9	2.9	2.9

Setelah dilakukan serangkaian tes *point* dan pengujian alat maka akan dilakukan analisa dari data yang telah di dapatkan. Sudut pandang dari bacaan ultrasonik sangat berpengaruh pada kestabilan sitem pada saat menuju *set point*. Jika sudut pandang dapat dijaga pada sudut 90^0 maka waktu yang dibutuhkan untuk mencapai *set point* akan semakin cepat.

Nilai PID hasil *tuning* yang didapatkan sangat mempengaruhi sistem dari *quadcopter*. Nilai Kp, Ki dan Kd membuat respon sistem menjadi cepat dalam mencapai *set point* dan sistemnya stabil tidak ada *overshot* maupun *osilasi*.

Tegangan yang terukur pada motor yang dipengaruhi oleh besar kecilnya PWM sangat berpengaruh pada baris program pengontrolkecepatan motor. Dengan data yang didapat penulis dapat mengetahui batasan output dari system kontrol PID yaitu minimum motor dapat berputar pada nilai 60 dan motor berputar secara maksimum pada nilai 254. Tetapi karena ketidaksesuaian berat total *quadcopter* maka *quadcopter* yang penulis buat tidak dapat melayang diudara. Berat total qudcopter lebih berat dari beban maksimum yang bisa diangkat oleh ke 4 *motor*. Dan juga karena penulis hanya membahas tentang 1 sudut saja yaitu *YAW* maka *quadcopter* harus diletakan pada lintasan tiang vertikal agar menghasilkan gerakan *VOTL (Vertikal take-off and Landing)*

Sesuai dengan tabel 3 maka program yang dibuat sebagai kontrol sistem PID telah sesai dan membuat sistem berjalan dengan baik sesuai seperti tujuan penelitian ini yaitu menjaga ketinggian *quadcopter* sesuai dengan *set point*. Jadi ketika *quadcopter* belum mencapai posisi *set point* atau masih dibawah

set point maka sistem (program) akan memberikan nilai PWM maksimal kepada ke empat motor yang seharusnya akan membuat *quadcopter* take-off dan bergerak vertikal keatas, dan apabila *quadcopter* ternyata melewati posisi *set point* atau berada diatas *set point* maka sistem (program) akan mengurangi nilai PWM ke empat motor sehingga putaran motor akan melambat dan membuat posisi *quadcopter* menjadi turun sehingga seharusnya akan kembali ke posisi *set point*. Pada saat posisi *quadcopter* tepat berada di *set point* maka sistem sistem akan memberikan nilai PWM referensi yang telah ditentukan di awal sebesar 200 yang diharapkan pada nilai PWM tersebut *quadcopter* melayang-layang.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian yang dilakukan telah dibuktikan sistem bekerja dengan baik dan dapat diambil beberapa kesimpulan sebagai berikut :

1. Besarnya nilai parameter yang didapatkan dari hasil tuning terstruktur adalah $K_p = 1$, $K_d = 0.01$ dan $K_i = 0.1$.
2. Besarnya nilai PWM yang dibutuhkan agar pesawat *quadcopter* mampu melayang (*hovering*) pada posisi dan ketinggian (*altitude*) terbang sesuai *set-point* yang diberikan, tanpa bantuan operator darat adalah 200, dengan maksimum 254 dan minimum 170.

5.2 Saran

Saran-saran yang dapat diberikan untuk perbaikan dan pengembangan penelitian ini adalah sebagai berikut :

1. Sensor ultrasonik yang digunakan hendaknya memiliki kecepatan membaca jarak dengan cepat untuk menanggulangi perubahan error yang extreme.
2. Sistem kontrol yang digunakan hendaknya lebih memperdalam tuning PID sehingga akan didapatkan konstanta yang lebih tepat.
3. Penelitian selanjutnya harus memperhatikan berat total yang bisa diangkat oleh ke empat motor quacopter. Sehingga tidak akan membuat *quadcopter* mengalami overload. Dan harus memperhatikan ke 3 sudut *quadcopter*.

DAFTAR PUSTAKA

1. Akmad Hendiawan, Gilang Prasetyo, Hary Oktavianto (2012), *Sistem Kontrol Altitude Pada UAV Model Quadcopter Dengan Metode PID*, The 14th Industrial Electronic Seminar (IES), Electronic Engineering Polytechnic Institute of Surabaya Indonesia, October 2012.
2. Bimo Jati Utomo (2017), *Rancang Bangun UAV (Unmanned Aerial Vehicle) Model Quadcopter dengan Menggunakan Algoritma Proportional Integral Derivative*, Program Studi Teknik Komputer, Fakultas Ilmu Terapan, Universitas Telkom Bandung, Juli 2017.
3. Ogata, K. 1996. *buku Modern Control Engineering, Third Edition*, Jakarta : Gramedia.
4. Kadir, Abdul. 2014. *Buku Pintar Pemrograman Arduino*, Yogyakarta: MediaKom
5. Widodo, Budiharto, 2005, *Perancangan Sistem dan Aplikasi Mikrokontroler*, Jakarta:PT.Elex Media Komputindo
6. Norris, Donald. 2014. *Build Your Own Quadcopter*, Kanada: McGraw-Hill Education