PENGONTROLAN KECEPATAN PUTARAN *ENGINE* PADA LENGAN *BICOPTER* DENGAN MENGGUNAKAN KONTROLER PID

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

GIOFANNY WIHAPRATAMA
NIM. 145060301111025

UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2018

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

LEMBAR PENGESAHAN

PENGONTROLAN KECEPATAN PUTARAN *ENGINE* PADA LENGAN *BICOPTER* DENGAN MENGGUNAKAN KONTROLER PID

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

Diajukan untuk memenuhi persyaratan

memperoleh gelar Sarjana Teknik

GIOFANNY WIHAPRATAMA

NIM. 145060301111025

Skripsi ini telah direvisi dan disetujui oleh dosen pembimbing

pada tanggal 2 November 2018

Mengetahui,

Ketua Jurusan Teknik Elektro	**Dosen Pembimbing**
Ir. Hadi Suyono, S.T., M.T., Ph.D. IPM	**Dr. Ir. Bambang Siswojo, M.T.**
NIP. 19730520 200801 1 013	**NIP. 19621211 198802 1 001**

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

JUDUL SKRIPSI :

PENGONTROLAN KECEPATAN PUTARAN *ENGINE* PADA LENGAN *BICOPTER* DENGAN MENGGUNAKAN KONTROLER PID

Nama Mahasiswa : Giofanny Wihapratama

NIM : 145060301111025

Program Studi : Teknik Elektro

Konsentrasi : Teknik Kontrol

KOMISI PEMBIMBING :

Ketua : Dr. Ir. Bambang Siswojo, M.T. .....

TIM DOSEN PENGUJI :

Dosen Penguji 1 : Dr. Ir. Erni Yudaningtyas, M.T. .....

Dosen Penguji 2 : Goegoes Dwi Nusantoro, S.T., M.T. .....

Dosen Penguji 3 : M. Aziz Muslim, S.T., M.T., Ph.D. .....

Tanggal Ujian : 26 Oktober 2018

SK Penguji : No. 2417 Tahun 2018

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. Tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 30 Oktober 2018

Mahasiswa,

Giofanny Wihapratama
NIM. 145060301111025

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

RINGKASAN

Giofanny Wihapratama, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Oktober 2018, *Pengontrolan Kecepatan Putaran Engine Pada Lengan Bicopter Dengan Menggunakan Kontroler PID*, Dosen Pembimbing: Bambang Siswojo.

Perkembangan ilmu pengetahuan dan teknologi setiap tahunnya telah mengalami kemajuan yang sangat pesat khususnya dalam bidang pesawat tanpa awak UAV (*Unmanned Aerial Vehicle*). Salah satu jenis UAV adalah *bicopter*. Teknologi UAV banyak digunakan untuk kebutuhan sehari-hari diantaranya untuk mengambil gambar dari udara, memantau cuaca, sebagai pesawat pengintai dll. *Bicopter* pada umumnya menggunakan tenaga elektrik untuk bergerak yaitu menggunakan baterai. Tugas akhir ini membuat *bicopter* dengan menggunakan *engine* berbahan bakar premium yang memiliki pengontrolan yang berbeda. Dengan menggunakan *engine*, *bicopter* diatur dengan menjaga agar kecepatan putarannya tetap terhadap perubahan daya dorong. Maka dari itu untuk menjaga kestabilan pada *bicopter* dapat dilakukan dengan cara mengatur *speed engine* (kecepatan putaran *engine*) menggunakan kontroller PID. Pada desain *bicopter* ini yaitu menggunakan 1 *engine* dengan 1 lengan untuk menggerakan 1 *propeller* dengan sudut *pitch* diubah-ubah sebagai gangguan. Proses perancangan PID pada penelitian ini menggunakan sensor *hall effect* untuk mengatur kecepatan dan menggunakan metode 1 Ziegler-Nichols, maka parameter PID dapat ditentukan dengan gain Kp = 7.25 , Ki = 9.06, dan Kd = 1.45 yang menunjukan bahwa *error* respon sistem secara keseluruhan tidak melebihi 5% dari *setpoint* dan mampu kembali stabil ketika diberi gangguan perubahan sudut *pitch*. Hal ini menunjukan bahwa kontroler PID dapat mengontrol kecepatan putaran *engine* dengan baik.

Kata kunci : *Bicopter*, kontroler PID, sensor *hall effect*, *engine*, UAV

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

SUMMARY

Giofanny Wihapratama, *Department of Electrical Engineering, Faculty of Engineering University of Brawijaya, October 2018. Controlling The Rotation Speed Engine At Bicopter Arm Using PID Controller, Academic Supervisor*: Bambang Siswojo.

The development of science and technology each year has experienced very rapid progress, especially in the field of unmanned aircraft UAV (Unmanned Aerial Vehicle). One type of UAV is bicopter. UAV technology is widely used for everyday needs such as taking pictures from the air, monitoring the weather, as reconnaissance aircraft etc. Bicopter generally uses electric power to move which is using a battery. This final project makes bicopter using a premium fueled engine that has different controls. By using the engine, bicopter is regulated by keeping the rotation speed constant against changes in thrust. Therefore, to maintain stability in the bicopter can be done by adjusting the engine speed (engine speed) using the PID controller. In this bicopter design, it uses 1 engine with 1 arm to drive 1 propeller with the pitch angle changed as interference. The PID design process in this study uses a sensor hall effect to adjust the speed and use the 1 Ziegler-Nichols method, then the PID parameter can be determined by gain Kp = 7.25 , Ki = 9.06 , and Kd = 1.45 which shows that the overall system response error does not exceed 5% of the setpoint and is able to stabilize when given a change in pitch angle. This shows that the PID controller can control the engine rotation speed well.

Keywords : *Bicopter, PID controller, hall effect sensor, engine, UAV*

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

PENGANTAR

Bismillahirrohmanirrohim. Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT, karena atas segala petunjuk dan nikmat-Nya lah skripsi ini dapat diselesaikan. Skripsi berjudul "*PENGONTROLAN KECEPATAN PUTARAN ENGINE PADA LENGAN BICOPTER DENGAN MENGGUNAKAN KONTROLER PID*" ini disusun untuk memenuhi persyaratan memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Universitas Brawijaya.

Penulis menyadari bahwa dalam penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan ketulusan dan kerendahan hati penulis menyampaikan terima kasih kepada:

1. Kedua orang tua, dan adik saya yang selalu menjadi motivator dalam pengerjaan skripsi ini.
2. Saudara-saudara saya di Tangerang yang selalu memberi motivasi untuk bisa mempercepat pengerjaan skripsi ini.
3. Bapak Ir. Hadi Suyono, S.T., M.T., Ph.D., IPM. selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
4. Ibu Ir. Nurussa'adah, M.T. selaku Sekretaris Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
5. Bapak Ali Mustofa, S.T., M.T. selaku Ketua Program Studi Sarjana Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
6. Bapak Ir. Purwanto, M.T. selaku Ketua Kelompok Dosen Konsentrasi Teknik Kontrol Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
7. Bapak Dr. Ir. Bambang Siswojo, M.T. selaku Dosen Pembimbing atas segala bimbingan, pengarahan, saran, dan kritik yang telah diberikan selama proses pengerjaan skripsi.
8. Segenap dosen pengajar dan staff administrasi Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
9. Sahabat-sahabat (boby, canggih, aldi, Iqbal, shidiq, jihad, agung, pangky, yoppy, faishal) dalam memberikan semangat dan saran dalam pengerjaan skripsi.
10. Teman-teman AufaGama (ariq, aufa, arisyi, rafli, ferry, uzi, riki, budi dkk) terima kasih telah berbagi kesenangan, pelajaran hidup, serta canda dan tawa.
11. Teman-teman sebimbingan Hilmi, dan Kharist atas motivasi yang telah diberikan.
12. Teman-teman Konsentrasi Teknik Kontrol Universitas Brawijaya.

13. Teman-teman DIODA 2014 atas segala bantuan dan kebersamaan yang telah diberikan selama masa studi.
14. Semua pihak yang telah memberikan bantuan dan dukungan baik secara langsung maupun tidak langsung atas penyelesaian skripsi ini.

Dalam penyusunan skripsi ini, penulis menyadar bahwa masih terdapat kekurangan karena kendala dan keterbatasan dalam pengerjaan skripsi ini. Oleh karena itu, penulis berharap saran dan kritik yang membangun untuk penyempurnaan tulisan di masa yang akan datang. Semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, Oktober 2018

Penulis

DAFTAR ISI

Halaman

PENGANTAR ..... **i**

DAFTAR ISI ..... **iii**

DAFTAR TABEL ..... **vii**

DAFTAR GAMBAR ..... **ix**

DAFTAR LAMPIRAN ..... **xi**

BAB I PENDAHULUAN ..... Error! Bookmark not defined.

1.1 Latar Belakang ..... **Error! Bookmark not defined.**

1.2 Rumusan Masalah ..... **Error! Bookmark not defined.**

1.3 Batasan Masalah ..... **Error! Bookmark not defined.**

1.4 Tujuan ..... **Error! Bookmark not defined.**

1.5 Manfaat ..... **Error! Bookmark not defined.**

1.6 Sistematika Penulisan ..... **Error! Bookmark not defined.**

BAB II TINJAUAN PUSTAKA ..... Error! Bookmark not defined.

2.1 *Bicopter* ..... **Error! Bookmark not defined.**

2.2 Motor Bakar 2 tak ..... **Error! Bookmark not defined.**

2.3 *Propeller* (Baling-Baling) ..... **Error! Bookmark not defined.**

2.4 Sensor *Hall Effect* ..... **Error! Bookmark not defined.**

2.5 Arduino Uno ..... **Error! Bookmark not defined.**

2.5.1 Catu Daya ..... **Error! Bookmark not defined.**

2.5.2 Memori ..... **Error! Bookmark not defined.**

2.5.3 Input & Output ..... **Error! Bookmark not defined.**

2.5.4 Komunikasi ..... **Error! Bookmark not defined.**

2.5.5 Perangkat Lunak (Arduino IDE) ..... **Error! Bookmark not defined.**

2.5.6 Otomatis *Software Reset* ..... **Error! Bookmark not defined.**

2.6 Motor Servo ..... **Error! Bookmark not defined.**

2.7 Sistem Pengontrolan ..... **Error! Bookmark not defined.**

2.8 Kontroler ..... **Error! Bookmark not defined.**

2.8.1 Kontroler *Proportional* ..... **Error! Bookmark not defined.**

2.8.2 Kontroler *Integral* ..... **Error! Bookmark not defined.**

2.8.3 Kontroler *Defferensial* ..... **Error! Bookmark not defined.**

2.8.4 Kontroler *Proporsional Integral* (PI) ..... **Error! Bookmark not defined.**

2.8.5 Kontroler *Proporsional Differensial* (PD) ..... **Error! Bookmark not defined.**
2.8.6 Kontroler PID ..... **Error! Bookmark not defined.**
2.8.7 Metode Perancangan Kontroler *Proporsional, Integral, dan defferensial* (PID) Menggunakan Metode Ziegler-Nichols ..... **Error! Bookmark not defined.**
BAB III METODE PENELITIAN ..... Error! Bookmark not defined.
3.1 Spesifikasi Alat ..... **Error! Bookmark not defined.**
3.2 Studi Literatur ..... **Error! Bookmark not defined.**
3.3 Perancangan Dan Perealisasian Alat ..... **Error! Bookmark not defined.**
3.3.1 Perancangan Sistem mekanik ..... **Error! Bookmark not defined.**
3.3.2 Perancangan Sistem Kontroler ..... **Error! Bookmark not defined.**
3.3.3 Perancangan Diagram Blok Sistem Keseluruhan ..... **Error! Bookmark not defined.**
3.4 Perealisasian Alat ..... **Error! Bookmark not defined.**
3.5 Perancangan kontroler PID ..... **Error! Bookmark not defined.**
3.5.1 Penentuan Nilai Penguatan Kontroler ..... **Error! Bookmark not defined.**
3.6 Perancangan Perangkat Lunak ..... **Error! Bookmark not defined.**
BAB IV HASIL DAN PEMBAHASAN ..... Error! Bookmark not defined.
4.1 Pengujian Sensor *Hall Effect* ..... **Error! Bookmark not defined.**
4.1.1 Tujuan Pengujian ..... **Error! Bookmark not defined.**
4.1.2 Peralatan Yang Digunakan ..... **Error! Bookmark not defined.**
4.1.3 Langkah Pengujian ..... **Error! Bookmark not defined.**
4.1.4 Hasil Pengujian ..... **Error! Bookmark not defined.**
4.2 Pengujian Respon Kurva S ..... **Error! Bookmark not defined.**
4.2.1 Tujuan Pengujian ..... **Error! Bookmark not defined.**
4.2.2 Peralatan Yang Digunakan ..... **Error! Bookmark not defined.**
4.2.3 Langkah Pengujian ..... **Error! Bookmark not defined.**
4.2.4 Hasil Pengujian ..... **Error! Bookmark not defined.**
4.3 Pengujian Motor Servo terhadap *Engine* ..... **Error! Bookmark not defined.**
4.3.1 Tujuan Pengujian ..... **Error! Bookmark not defined.**
4.3.2 Peralatan Yang Digunakan ..... **Error! Bookmark not defined.**
4.3.3 Langkah Pengujian ..... **Error! Bookmark not defined.**
4.3.4 Hasil Pengujian ..... **Error! Bookmark not defined.**
4.4 Pengujian Sistem Secara Keseluruhan ..... **Error! Bookmark not defined.**
4.4.1 Tujuan Pengujian ..... **Error! Bookmark not defined.**
4.4.2 Peralatan Yang Digunakan ..... **Error! Bookmark not defined.**

4.4.3 Langkah Pengujian ..... **Error! Bookmark not defined.**
4.4.4 Hasil Pengujian ..... **Error! Bookmark not defined.**
BAB V PENUTUP ..... Error! Bookmark not defined.
5.1 Kesimpulan .....**Error! Bookmark not defined.**
5.2 Saran .....**Error! Bookmark not defined.**
DAFTAR PUSTAKA..... Error! Bookmark not defined.
LAMPIRAN..... Error! Bookmark not defined.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

DAFTAR TABEL

No.	Judul	Halaman
Tabel 2.1	Deskripsi arduino uno secara umum ….…….……	11
Tabel 2.2	Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step Dari *Plant* ..…….……...….	24
Tabel 2.3	Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain Kcr* dan *Critical Period Pcr* ……...……..……...….	25
Tabel 3.1	Aturan Metode 1 Ziegler-Nichols (Ogata K., 1997) ……	32
Tabel 4.1	Aturan Metode 1 Ziegler-Nichols (Ogata K., 1997) ……	38

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

DAFTAR GAMBAR

No.	Judul	Halaman
Gambar 2.1	*Konfigurasi* pada *Bicopter*	5
Gambar 2.2	*Bicopter*	6
Gambar 2.3	System pada Motor Bakar 2 Tak	8
Gambar 2.4	*Propeller*/Baling-Baling	9
Gambar 2.5	Sensor *Hall Effect*	10
Gambar 2.6	Arduino Uno dan Komponennya	11
Gambar 2.7	Lebar Pulsa	15
Gambar 2.8	Motor Servo	15
Gambar 2.9	Diagram Blok Sistem Kendali *Loop* Terbuka	16
Gambar 2.10	Diagram Blok Sistem Kendali *Loop* Tertutup	16
Gambar 2.11	Diagram Blok Kontroler *Proportional*	17
Gambar 2.12	Diagram Blok Kontroler *Integral*	18
Gambar 2.13	Diagram Blok Kontroler *Differensial*	19
Gambar 2.14	Diagram Blok Sistem Kontrol PID (*Proportional, Integral & Differensial*	21
Gambar 2.15	Hubungan fungsi waktu antara sinyal keluaran dan masukan kontroler PID	21
Gambar 2.16	Kurva Respon Unit Step yang Menunjukkan 25% Maximun *Overshoot*	22
Gambar 2.17	Respon *Plant* Terhadap Masukan Berupa Unit Step	23
Gambar 2.18	Kurva Respon yang Berbentuk S	23
Gambar 2.19	Sistem *Loop* Tertutup dengan *Kontroler Proporsional*	24
Gambar 2.20	Osilasi Berkesinambungan dengan Periode Pcr	25
Gambar 3.1	Desain Alat Lengan *Bicopter*	28
Gambar 3.3	Diagram Blok Kendali Sistem	29
Gambar 3.2	Lengan *Bicopter*	31
Gambar 3.4	Kurva Respon yang Berbentuk S (Ogata, K., 1997)	32
Gambar 3.5	*Flowchart* Perangkat Lunak	33
Gambar 4.1	Diagram Perancangan Peralatan Pengujian Sensor	36
Gambar 4.2	Respon Pengujian Sensor *Hall Effect*	36
Gambar 4.3	Respon Pengujian Kurva S	37
Gambar 4.4	Penentuan parameter tuning PID pada respon Kurva S	38

Gambar 4.5 Grafik hubungan perubahan sudut servo dengan kecepatan putaran *engine* 40
Gambar 4.6 Diagram Peralatan Pengujian *Engine* 41
Gambar 4.7 Respon Pengujian Sistem Secara Keseluruhan 41
Gambar 4.8 Parameter-parameter kerja sistem 42

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

DAFTAR LAMPIRAN

No.	Judul	Halaman
Lampiran 1.	Data Hasil Pengujian ……...……	49
Lampiran 2.	*Listing* Program ……...…...……	52
Lampiran 3.	Dokumentasi Alat ……...…...……	55
Lampiran 4.	*Datasheet* Arduino ATMega 2560 ……...…….....	56

UNIVERSITAS BRAWIJAYA

UNIVERSITAS BRAWIJAYA

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi setiap tahunnya telah mengalami kemajuan yang sangat pesat khususnya pada bidang penerbangan. Penelitian pada bidang penerbangan sudah sangat berkembang, terutama dalam bidang *auto* pilot, dan bahkan sudah memasuki ranah pesawat tanpa awak UAV (*Unmanned Aerial Vehicle*). Salah satu jenis pesawat yang bisa diklasifikasikan sebagai UAV adalah *bicopter*, namun klasifikasi *bicopter* masih dibawah UAV karena komponen pendukungnya tidak selengkap UAV (Carrilo, 2013). Teknologi UAV akhir-akhir ini banyak digunakan untuk kebutuhan sehari-hari diantaranya untuk mengambil gambar dari udara, memantau cuaca, sebagai pesawat pengintai dan untuk mengidentifikasi bencana alam.

Bicopter merupakan salah satu UAV yang memiliki 2 buah lengan, masing-masing lengan memliki motor dan baling-baling (*propeller*) yang menghasilkan daya angkat/daya dorong (*thrust*). Dengan mengubah besaran kecepatan putaran kedua buah motor maka *bicopter* dapat bergerak ke atas, bawah, kiri, kanan, dan rotasi. Pergerakan tersebut lebih dikenal dengan istilah *pitch* (bergerak maju atau mundur), *roll* (bergerak kiri atau kanan), dan *yaw* (rotasi kiri atau rotasi kanan). *Bicopter* digerakan oleh sejumlah baling-baling. Semakin banyak baling-baling yang dimiliki, kecepatan, daya angkat dan keseimbangan *bicopter* akan meningkat, sehingga kapasitas beban yang di angkat akan semakin meningkat (Kristianto, D, 2012).

Bicopter memiliki beberapa gerakan utama yaitu *take off*, *hover*, *cruise* dan *landing*. Saat melakukan gerakan tersebut *bicopter* harus dalam keadaan stabil, mengingat ukuran dari *bicopter* yang kecil sehingga mengakibatkan gerakan *bicopter* mudah terganggu oleh angin. *Bicopter* memiliki kelebihan pada mobilitas dan fleksibelitas sehingga memudahkan untuk melakukan manuver di tempat yang sempit karena ukurannya yang kecil, dan dapat mendarat secara vertikal (VTOL, *vertical take off and landing*). Akan tetapi *bicopter* dengan mobilitas yang tinggi jika kestabilan terganggu maka *bicopter* akan kehilangan keseimbangannya. Maka dari itu *bicopter* membutuhkan sistem kendali untuk meningkatkan keseimbangannya.

Bicopter menggunakan tenaga elektrik untuk bergerak dengan menggunakan baterai. Hal ini membuat *bicopter* mempunyai kelemahan yaitu hanya dapat terbang dalam jangka waktu yang singkat.

Kestabilan dan jangka waktu terbang yang singkat pada *bicopter* merupakan hal yang sangat krusial. Mengingat *bicopter* saat ini masih menggunakan baterai dan sering digunakan untuk kebutuhan sehari-hari yang tentunya membutuhkan waktu berjam-jam. Solusinya sekarang dengan menggunakan tenaga mesin berbahan bakar premium yang memiliki pengontrolan yang berbeda. Jika *bicopter* menggunakan motor listrik yang diatur putarannya untuk mengubah daya dorong maka dengan menggunakan *engine*, *bicopter* diatur dengan menjaga agar kecepatan putarannya tetap terhadap perubahan daya dorong. Maka dari itu untuk menjaga kestabilan pada *bicopter* dapat dilakukan dengan cara mengatur *speed engine* (kecepatan putaran *engine*) menggunakan kontroller PID.

Dalam skripsi ini, desain pada *bicopter* yaitu menggunakan 1 *engine* dengan 1 lengan menggerakan 1 *propeller* dan sensor *hall effect* untuk mengatur kecepatan. *Bicopter* diuji dalam keadaan statis atau diam, dimana yang dikontrol bagian sub sistemnya saja yaitu hanya mengatur atau menstabilkan kecepatan putaran *engine* yang merupakan bagian sistem kontrol *bicopter* secara keseluruhan. Sedangkan sudut *pitch* diubah-ubah sebagai gangguan. Ketika sudut *pitch* diubah-ubah akan menghasilkan pembebanan yang berubah-ubah maka *thrust* akan berubah yang mengakibatkan kecepatan putaran *engine* berubah. Sehingga dibutuhkan sistem kontrol yang dalam kondisi apapun harus dapat mempertahankan kecepatannya agar tetap/konstan yaitu dengan menggunakan kontroler PID. Untuk penelitian lebih lanjut nantinya *bicopter* yang menggunakan *engine* ini dapat dikembangkan dan direalisasikan menjadi *bicopter* secara utuh. Tentunya *bicopter* ini dapat dikembangkan menjadi pesawat UAV lainnya seperti *tricopter, quadcopter* dll. Sebelum merealisasikan itu maka dibutuhkan pengaturan kecepatan pada 1 *engine* dengan 1 lengan untuk menggerakan 1 *propeller*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, dapat disusun rumusan masalah sebagai berikut:

1. Bagaimana mendesain model mekanik 1 lengan *bicopter*?
2. Bagaimana merancang *hardware* dan *software* sistem pengendalian kecepatan putaran *engine* pada lengan *bicopter* dengan menggunakan kontroler PID?

1.3 Batasan Masalah

Hal-hal yang berkaitan dengan penelitian ini akan diberi batas sebagai berikut:

1. Pengontrolan dibatasi hanya untuk membentuk 1 *engine* dengan 1 lengan untuk menggerakan 1 *propeller*.
2. *Bicopter* ini diuji dalam keadaan statis atau diam, jadi hanya untuk mengatur kecepatan putaran pada *engine* ketika sudut *pitch* diubah-ubah sebagai gangguan.
3. Pengontrolan PID berbasis arduino.
4. Penggerak utama menggunakan motor bakar 2 tak yang menggunakan bahan bakar premium.
5. Sensor yang digunakan adalah *hall effect*.

1.4 Tujuan

Tujuan dari penelitian ini adalah untuk mengatur *speed engine* (kecepatan putaran *engine*) pada lengan *bicopter* dengan menggunakan kontroller PID agar kecepatan putaran pada *engine* tetap/konstan ketika daya dorong (*thrust*) berubah-ubah.

1.5 Manfaat

Adapun manfaat yang diharapkan dan diperoleh dari penelitian ini adalah sebagai berikut:

1. *Bicopter* dengan menggunakan bahan bakar premium memungkinkan untuk terbang lebih lama dibandingkan menggunakan baterai yang tentunya hanya dapat bertahan dalam waktu yang singkat..
2. Dengan mengatur kecepatan putaran pada *engine*, maka ketika ada gangguan *thust* kecepatan putaran pada *engine* tetap/konstan, bertujuan untuk menjaga kestabilan saat bermanuver.
3. Hasil pada penelitian ini nantinya *bicopter* dapat dikembangkan dan direalisasikan menjadi *bicopter* secara utuh yaitu dengan menggunakan 2 *engine* dengan 2 lengan untuk menggerakan 2 *propeller*. *Bicopter* ini juga dapat dikembangkan menjadi pesawat UAV lainnya seperti *tricopter, quadcopter* dll.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan penelitian ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisikan latar belakang masalah, rumusan masalah, batasan penulisan, tujuan masalah, manfaat dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi teori-teori yang digunakan untuk mendukung dalam permasalahan yang akan dilakukan dalam penelitian ini.

BAB III : METODE PENELITIAN

Berisi tahapan penyelesaian penelitian ini yang meliputi spesifikasi alat, studi literatur, perancangan dan pembuatan alat, pengujian alat, dan pengambilan kesimpulan.

BAB IV : HASIL DAN PEMBAHASAN

Menjelaskan tentang pengujian alat dan analisa yang meliputi pengujian bagian blok sistem dan pengujian sistem secara keseluruhan.

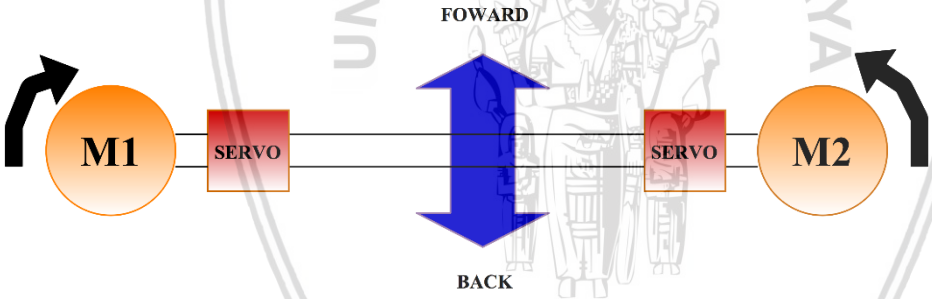
BAB V : PENUTUP

Berisi kesimpulan dari analisis hasil peneltian serta saran untuk penelitian selanjutnya.

BAB II
TINJAUAN PUSTAKA

2.1 *Bicopter*

Bicopter adalah salah satu jenis multirotor yang memiliki 2 rotor sebagai penggerak *propeller* yang menghasilkan gaya angkat. Kedua motor ini dapat dikendalikan kecepatannya secara terpisah, 1 motor berputar searah jarum jam dan 1 motor lainnya berputar berlawanan arah jarum jam. Dengan mengubah besaran kecepatan putaran kedua buah motor maka *bicopter* dapat bergerak ke atas, bawah, kiri, kanan, dan rotasi. Pergerakan tersebut lebih dikenal dengan istilah *pitch* (bergerak maju atau mundur), *roll* (bergerak kiri atau kanan), dan *yaw* (rotasi kiri atau rotasi kanan). *Bicopter* dapat melakukan *take off* dan *landing* secara vertikal sehingga *bicopter* dapat melakukannya di tempat yang sempit. *Konfigurasi* pada *bicopter* dapat diperlihatkan pada gambar 2.1.



Gambar 2.1 Konfigurasi pada *Bicopter*

Ketika *bicopter* sedang dalam keadaan *hovering*/melayang di udara, kecepatan putar pada setiap rotornya adalah sama. Saat *bicopter* melakukan gerakan *foward*, kedua buah *servo* digerakan kearah depan. Saat *bicopter* melakukan gerakan back, kedua buah *servo* digerakan kearah belakang. Ketika *bicopter* melakukan gerakan *turn left* atau *turn right* maka yang diatur kecepatan pada setiap motornya. Jadi gaya angkat yang dihasilkan kedua *propeller* akan mempunyai komponen gaya ke atas dan ke depan sehingga *bicopter* akan maju dengan mempertahankan ketinggiannya. Salah satu contoh model *bicopter* diperlihatkan pada gambar 2.2.

Gambar 2.2 Bicopter
Sumber : dronezone.com

2.2 Motor Bakar 2 tak

Motor Bakar 2 tak adalah mesin pembakaran dalam yang dalam satu siklus pembakaran akan mengalami 2 langkah piston, berbeda dengan putaran 4 tak yang mengalami 4 langkah piston dalam satu siklus pembakaran, meskipun keempat proses seperti intake, kompresi, tenaga dan pembuangan juga terjadi pada motor bakar dua tak.

Untuk prinsip kerja motor bakar 2 tak, mempunyai istilah baku yang berlaku dalam teknik otomotif yaitu :

- TMA (titik mati atas) atau TDC (*top dead centre*): Posisi piston berada pada titik paling atas dalam silinder mesin atau piston berada pada titik paling jauh dari poros engkol (*crankshaft*).
- TMB (titik mati bawah) atau BDC (*bottom dead centre*): Posisi piston berada pada titik paling bawah dalam silinder mesin atau piston berada pada titik paling dekat dengan poros engkol (*crankshaft*).
- Ruang bilas yaitu ruangan di bawah piston dimana terdapat poros engkol (*crankshaft*). Sering disebut sebagai bak engkol (*crankcase*) berfungsi sebagai gas hasil campuran udara, bahan bakar dan pelumas bisa tercampur lebih merata.
- Pembilasan (*scavenging*) yaitu proses pengeluaran gas hasil pembakaran dan proses pemasukan gas untuk pembakaran dalam ruang bakar.

1. Langkah yang pertama, Piston bergerak dari TMA ke TMB.

Pada saat piston bergerak dari TMA ke TMB, maka akan menekan ruang bilas yang berada di bawah piston. Semakin jauh piston meninggalkan TMA menuju TMB maka tekanan di ruang bilas semakin meningkat. Pada titik tertentu, piston (*ring* piston) akan melewati lubang pembuangan gas dan lubang pemasukan gas. Posisi masing-masing lubang tergantung dari desain perancang. Umumnya ring piston akan melewati lubang pembuangan terlebih dahulu. Pada saat *ring* piston melewati lubang pembuangan, gas di dalam ruang bakar keluar melalui lubang pembuangan. Pada saat *ring* piston melewati lubang pemasukan, gas yang tertekan dalam ruang bilas akan terpompa masuk dalam ruang bakar sekaligus mendorong gas yang ada dalam ruang bakar keluar melalui lubang pembuangan. Piston terus menekan ruang bilas sampai titik TMB, sekaligus memompa gas dalam ruang bilas masuk ke dalam ruang bakar.

2. Langkah yang kedua, Piston bergerak dari TMB ke TMA.

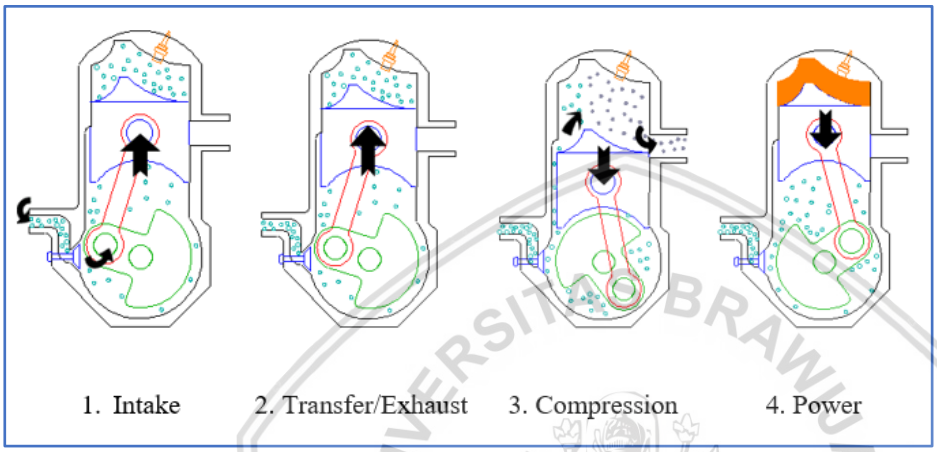
Pada saat piston bergerak TMB ke TMA, maka akan menghisap gas hasil percampuran udara, bahan bakar dan pelumas masuk ke dalam ruang bilas. Percampuran ini dilakukan oleh karburator atau sistem injeksi. Saat melewati lubang pemasukan dan lubang pembuangan, piston akan mengkompresi gas yang terjebak dalam ruang bakar. Piston akan terus mengkompresi gas dalam ruang bakar sampai TMA. Beberapa saat sebelum piston sampai di TMA, busi menyala untuk membakar gas dalam ruang bakar. Waktu nyala busi sebelum piston sampai TMA dengan tujuan agar puncak tekanan dalam ruang bakar akibat pembakaran terjadi saat piston mulai bergerak dari TMA ke TMB karena proses pembakaran sendiri memerlukan waktu dari mulai nyala busi sampai gas terbakar dengan sempurna.

System pada motor bakar 2 dapat dilihat pada gambar dibawah ini :

1) **Langkah Masuk (*Intake*).** Campuran bahan bakar dan udara dihisap masuk ke dalam rumah engkol akibat tekanan vakum yang terjadi pada saat piston bergerak ke atas.
2) **Langkah Penyaluran (*Transfer/Exhaust*)**. Pada saat mendekati posisititik mati bawah, saluran masuk terbuka dan campuran bahan bakar dan udara masuk ke dalam silinder. Pada saat yang sama masuknya campuran bahan bakar dan udara tersebut mendorong sisa hasil pembakaran keluar melalui saluran pengeluaran pada sisi yang berlawanan dari lubang pemasukan.

3) **Langkah Tekan (*Compression*).** Selanjutnya piston bergerak ke atas dan menekan campuran bahan bakar dan udara. (pada saat yang sama terjadi langkah masuk yang berikutnya di bagian bawah piston).

4) **Langkah Tenaga (*Power*).** Pada saat pendekati posisi titik mati atas busi akan menyala dan menyundut campuran bahan bakar dan udara sehingga terjadi ledakan yang mendorong piston ke bawah.



Gambar 2.3 System pada Motor Bakar 2 Tak
Sumber : http://eckonopianto.blogspot.co.id

2.3 *Propeller* (Baling-Baling)

Sebuah *propeller*/baling-baling berperan sebagai sayap berputar. *Propeller* yang digunakan pada *quadcopter* memiliki dua jenis yaitu *Clock Wise* (searah jarum jam) dan *Counter Clock Wise* (berlawanan arah jarum jam). *Propeller* diklasifikasikan berdasarkan Panjang dan *pitch*. Misalkan 9x4,7 baling-baling adalah Panjang 9 inci dan memili *pitch* 4,7.

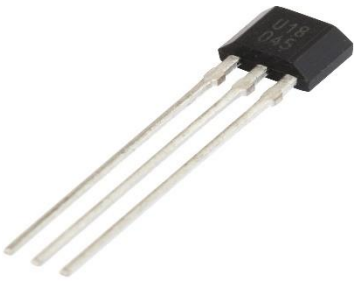
Secara umum *pitch* yang lebih tinggi berarti rotasi lebih lambat, tetapi akan meningkatkan kecepatan kendaraan anda yang juga menggunakan daya lebih. *Propeller* juga harus di sesuaikan dengan motor yang digunakan. Umumnya *propeller* dengan angka nada rendah dapat menghasilkan lebih banyak *torsi* sehingga motor tidak perlu bekerja keras. Untuk *quadcopter* yang lebih besar yang membawa muatan, baling-baling besar dan motor rendah KV cenderung bekerja lebih baik. karena memliki momentum yang lebih rotasi dan akan lebih mudah menjaga stabilitas. Gambar *propeller* dapat diperlihatkan pada gambar 2.4.

Gambar 2.4 Propeller/Baling-Baling
Sumber : http://www.aliexpress.com

2.4 Sensor *Hall Effect*

Sensor Efek Hall atau dalam bahasa Inggris disebut dengan *Hall Effect Sensor* adalah komponen jenis transduser yang dapat mengubah informasi magnetik menjadi sinyal listrik untuk pemrosesan rangkaian elektronik selanjutnya. Sensor Efek Hall ini sering digunakan sebagai sensor untuk mendeteksi kedekatan (*proximity*), mendeteksi posisi (*positioning*), mendeteksi kecepatan (*speed*), mendeteksi pergerakan arah (directional) dan mendeteksi arus listrik (current sensing). Sensor Efek Hall ini merupakan perangkat atau komponen yang diaktifkan oleh medan magnet eksternal. Seperti yang kita ketahui bahwa medan magnet memiliki dua karakteristik penting yaitu densitas flux (*flux density*) dan Kutub (kutub selatan dan kutub utara). Sinyal masukan (Input) dari Sensor Efek Hall ini adalah densitas medan magnet disekitar sensor tersebut, apabila densitas medan magnet melebihi batas ambang yang ditentukan maka sensor akan mendeteksi dan menghasilkan tegangan keluaran (*output*) yang disebut dengan Tegangan Hall (VH).

Prinsip kerja sensor ini adalah dengan memanfaatkan fenomena efek Hall. Efek Hall ini didasarkan pada efek medan magnetik terhadap partikel bermuatan yang bergerak. Ketika ada arus listrik yang mengalir pada alat efek Hall yang ditempatkan dalam medan magnet yang arahnya tegak lurus, pergerakan pembawa muatan akan berbelok ke salah satu sisi dan menghasilkan medan listrik. Medan listrik terus membesar hingga gaya Lorentz yang bekerja pada partikel menjadi nol. Perbedaan potensial antara kedua sisi alat tersebut disebut potensial Hall. Potensial Hall ini sebanding dengan medan magnet dan arus listrik yang melalui alat. Contoh sensor *hall effect* diperlihatkan pada gambar 2.5.

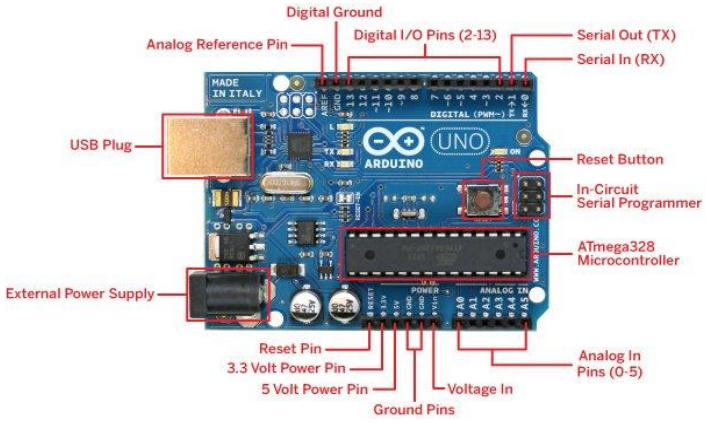


Gambar 2.5 Sensor *Hall Effect*
Sumber : protostack.com

2.5 Arduino Uno

Mikrokontroller Arduino Uno adalah *board* berbasis mikrokontroller pada Atmega328. Sebuah *board* minimun *system* mikrokontroller yang bersifat *open source*. Arduino juga mempunyai bahasa pemrogramannya sendiri yang berupa bahasa C. *Board* ini memiliki 14 digital *input / output* pin (dimana 6 pin dapat digunakan sebagai *output* PWM). 6 *input* analog, 16 MHz *osilator* kristal, koneksi USB, jack listrik, tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya mengubungkan *board* Arduino Uno ke komputer dengan menggunakan kabel USB atau sumber tegangan bisa didapat dari *adaptor* AC ke DC atau menggunakan baterai untuk menjalankannya. Port USB tersebut selain untuk *loader* ketika memprogram, bias juga difungsikan sebagai port komunikasi serial. *Board* Arduino Uno memiliki fitur-fitur baru sebagai berikut:

- Pin *out* 1,0: ditambah SDA dan SCL pin yang dekat dengan pin AREF dan dua pin baru lainnya yang diletakkan dekat dengan pin *RESET*, IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan 5V dan dengan Arduino *Due* yang beroperasi dengan tegangan 3,3V. Yang kedua merupakan sebuah pin yang tak terhubung, yang disediakan untuk tujuan kedepannya.
- Sirkuit *RESET* yang lebih kuat
- ATMega 16U2 menggantikan 8U2



Gambar 2.6 Arduino Uno dan Komponennya
Sumber : https://eeeproject.com/arduino-uno-board/

Tabel 2.1
Deskripsi Arduino Uno secara umum

Microcontroller	ATMega328
Operasi Voltage	5V
Input Voltage	7-12V (Rekomendasi)
Input Voltage	6-20 V (limits)
I/O	14 pin (6 pin untuk PWM)
Arus	50mA
Flash Memory	32KB
Boatloader	SRAM 2KB
EEPROM	1KB
Kecepatan	16MHz

2.5.1 Catu Daya

Arduino Uno dapat diaktifkan memalui koneksi USB atau dengan catu daya *eksternal*. Sumber daya dipilih secara otomatis. *Eksternal* (non-USB) tegangan dapat diperoleh dari sebuah adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2.1mm ke power jack dari *board*. Kabel *lead* dari baterai dapat dimasukkan ke dalam *header* pin *Ground* (Gnd) dan Vin dari konektor POWER.

Board Arduino Uno dapat beroperasi pada sebuah suplai eksternal dari 6 sampai 20 Volt. Jika di suplai kurang dari 7 volt, contohnya ketika pin 5V mendapat suplai kurang dari

5 volt dapat berakibat *board* arduino menjadi tidak stabil. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak *board*. Rentang yang dianjurkan adalah 7-12V. Pin-pin catu daya adalah sebagai berikut:

- VIN. Tegangan *input* ke *board* Arduino ketika menggunakan sumber daya eksternal (sebagai lawan dari 5 volt dari koneksi USB atau sumber daya lainnya diatur). Anda dapat menyediakan tegangan melalui pin ini, atau, jika memasok tegangan melalui colokan listrik, mengaksesnya melelui pin ini.
- 5V. Catu daya diatur digunakan untuk daya mikrokontroller dan komponen lainnya di *board*. Hal ini dapat terjadi baik dari VIN melalui regulator on *board*, atau diberikan oleh USB.
- 3,3V pasokan yang dihasilkan oleh regulator *on-board*. Menarik arus maksimum adalah 50 mA.
- GND.

2.5.2 Memori

ATMega328 memiliki 32KB dengan 0,5 KB digunakan untuk *loading file*. ATMega328 juga memiliki 2 KB dari SRAM dan 1 KB dari EEPROM.

2.5.3 Input & Output

Masing-masing dari 14 pin digital pada Uno dapat digunakan sebagai *input* atau *output*, menggunakan fungsi *pinMode*(), *digitalWire*(), dan *digitalRead*(). Input/output tersebut beroperasi di 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull-up* resistor dari 20-50KOhms. Selain itu, beberapa pin memiliki fungsi sebagai berikut:

- Serial: 0(RX) dan 1(TX). Digunakan untuk menerima (RX) dan menirimkan (TX) data TTL serial. Pin ini terhubung ke pin yang sesuai dari chip ATMega8U2 *USB-to-Serial* TTL.
- Eksternal interupsi: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu interupsi pada nilai yang rendah, tepi naik atau jatuh, atau perubahan nilai. Lihat *attach interupt*() fungsi untuk rincian.
- PWM: 3, 5, 6, 9, 10 dan 11. Menyediakan 8-bit ouput PWM dengan analog *Write* () fungsi.

- SPI: 10 (SS), 11 (mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI.
- LED: 13. Ada built-in LED terhubung ke pin digital13. Ketika pin adlah nilai TINGGI, LED menyala, ketika pin adalah RENDAH, itu *off*.
- Arduino Uno memilih 6 *input* analog, diberi label A0 melalui A5, masing-masing menyediakan 10 bit resolusi yaitu 1024 nilai yang berbeda. Secara *default* sistem mengukur dari tanah sampai 5 volt.
- TWI: A4 atau SDA pin dan A5 atau SCL pin. Mendukung komunikasi TWI.
- Aref. Referensi tegangan untuk *input* analog. Digunakan dengan *analog Reference* ().
- Untuk pemetaan untuk ATMega8, 168, 328 adalah identik.

2.5.4 Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroller lain. ATMega328 ini menyediakan UART TTL (5V) komunikasi serial, yang tersedia pada pin digital 0 (RX) dan 1(TX). Sebuah ATMega16U2 pada saluran *board* ini komunikasi serial melalui USB dan muncul sebagai com port virtual untuk perangkat lunak pada komputer. *Firmware* Arduino menggunakan USB *driver* standar COM, dan tidak ada *driver eksternal* yang dibutuhkan. Namun, pada *windows*, *file* .inf diperlukan. Perangkat lunak Arduino termasuk monitor serial yang memungkinkan data sederhana yang akan dikirim ke *board* Arduino. RX dan TX LED di *board* akan berkedip ketika data sedang dikirim melalui *chip USB-to-Serial* dan koneksi USB ke komputer. ATMega328 ini uga mendukung komunikasi I2C (TWI) dan SPI. Perangkat lunak Arduino untuk penyederhanaan penggunaan dari bus 12C. Fungsi ini digunakan untuk melakukan komunikasi *interface* pada sistem.

2.5.5 Perangkat Lunak (Arduino IDE)

Lingkungan *open-source* Arduino memudahkan untuk menulis kode dan meng-*upload* ke *board* Arduino. Ini berjalan pada Windows, Mac OS X, dan Linux. Berdasarkan pengolahan, avr-gcc, dan perangkat lunak sumber terbuka lainnya. Arduino Uno dapat diprogram dengan perangkat lunak Arduino. Pilih Arduino Uno dari *Tool* lalu sesuaikan dengan mikrokontroller yang digunakan. ATMega328 pada Arduino Uno memiliki *bootloader* yang memungkinkan anda untuk meng-*upload* program baru tanpa

menggunakan *programmer hardware eksternal*. Cara berkomunikasi menggunakan protokol dari bahasa C. Sistem ini dapat menggunakan perangkat lunak FLIP Atmel (Windows) atau *programmer* DFU (Mac OS X dan Linux) untuk memuat *firmware* baru atau anda dapat menggunakan *header* ISP dengan *programmer eksternal*.

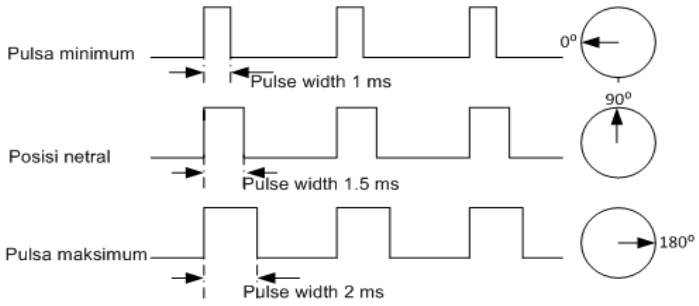
2.5.6 Otomatis *Software Reset*

Tombol *reset* Arduino Uno dirancang untuk menjalankan program yang tersimpan didalam mikrokontroller dari awal. Tombol *reset* terhubung ke ATMega328 melalui kapasitar 100 nf. Setelah tombol *reset* ditekan cukup lama untuk me-*reset chip*, *software* IDE Arduino dapat juga berfungsi untuk meng-*upload* program dengan hanya menekan tombol *upload* di *software* IDE Arduino.

2.6 Motor Servo

Motor servo adalah sebuah perangkat atau aktuator putar (motor) yang dirancang dengan sistem kontrol umpan balik *close loop* (servo), sehingga dapat di *set-up* atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer. Serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas posisi putaran poros motor servo.

Penggunaan sistem kontrol *loop* tertutup pada motor servo berguna untuk mengontrol gerakan dan posisi akhir dari poros motor servo. Posisi poros output akan di sensor untuk mengetahui posisi poros sudah tepat seperti yang di inginkan atau belum, dan jika belum, maka kontrol input akan mengirim sinyal kendali untuk membuat posisi poros tersebut tepat pada posisi yang diinginkan. Motor servo dikendalikan dengan memberikan sinyal modulasi lebar pulsa (*Pulse Wide Modulation* / PWM) melalui kabel kontrol. Lebar pulsa sinyal kontrol yang diberikan akan menentukan posisi sudut putaran dari poros motor servo.



Gambar 2.7 Lebar Pulsa
Sumber : http://trikueni-desain-sistem.blogspot.co.id

Sebagai contoh, lebar pulsa dengan waktu 1,5 ms (mili detik) akan memutar poros motor servo ke posisi sudut 90º. Bila pulsa lebih pendek dari 1,5 ms maka akan berputar ke arah posisi 0º atau ke kiri (berlawanan dengan arah jarum jam), sedangkan bila pulsa yang diberikan lebih lama dari 1,5 ms maka poros motor servo akan berputar ke arah posisi 180º atau ke kanan (searah jarum jam). Ketika lebar pulsa kendali telah diberikan, maka poros motor servo akan bergerak atau berputar ke posisi yang telah diperintahkan, dan berhenti pada posisi tersebut. Salah satu contoh motor servo diperlihatkan pada gambar 2.8.

Gambar 2.8 Motor Servo
Sumber : http://desainsistem.blogspot.co.id

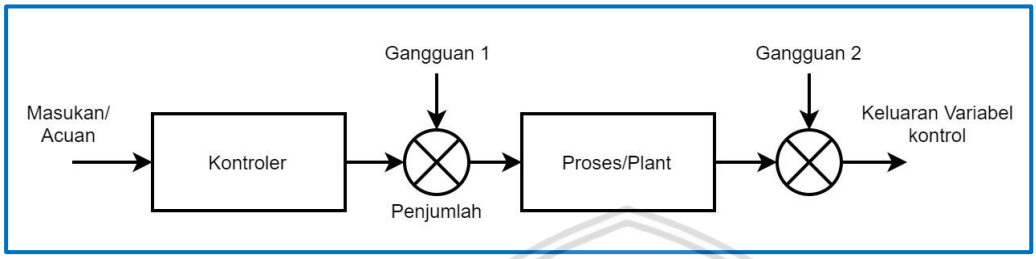
2.7 Sistem Pengontrolan

Sistem Pengendalian terdapat kontroler yang sering juga dikenal sebagai kompensator, kontroler adalah suatu system dinamis yang ditambahkan guna mendapatkan karakteristik keseluruhan yang diinginkan (K.Ogata, 2010). Pengukuran merupakan proses pembandingan beberapa variabel untuk mendapatkan sebuah besaran dengan kesalahan yang sangat kecil.

Menurut bentuk *loop* sistem pengontrolannya, sistem pengontrolan terbagi menjadi 2 macam, yaitu:

1. Sistem Pengendalian *Loop* Terbuka (*Open Loop Control System)*

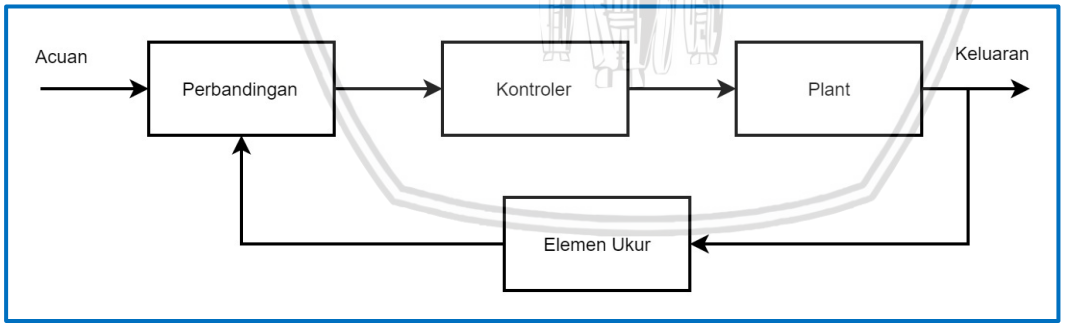
Sistem Pengendalian *loop* terbuka adalah sistem pengendalian yang keluarannya tidak dapat mempengaruhi aksi dari pengendaliannya, jadi pada sistem ini keluaran dari kontroler tidak diukur atau balikkan untuk dibandingkan dengan masukannya (K. Ogata, 1985). Diagram blok *loop* terbuka dapat dilihat pada gambar 2.9.



Gambar 2.9 Diagram Blok Sistem Kendali *Loop* Terbuka

2. Sistem Pengendalian *Loop* Tertutup (*Closed Loop Control System)*

Sistem kontrol *loop* tertutup adalah sistem pengendalian yang sinyal keluarannya memiliki pengaruh langsung terhadap aksi pengendaliannya (K. Ogata, 1985). Kontrol *loop* tertutup termasuk dalam sistem kontrol berumpan balik dimana kesalahan penggerak atau *error* merupakan selisih antara sinyal masukan dan umpan-balik. Diagram blok sistem *loop* tertutup dapat dilihat pada gambar 2.10.



Gambar 2.10 Diagram Blok Sistem Kendali *Loop* Tertutup

2.8 Kontroler

Keberadaan kontroler dalam sebuah sistem kontrol mempunyai kontribusi yang besar terhadap perilaku sistem. Pada prinsipnya hal itu disebabkan oleh tidak dapat diubahnya komponen penyusun sistem tersebut. Artinya, karakteristik plant harus diterima sebagaimana melalui penambahan suatu sub sistem, yaitu kontroler.

Salah satu fungsi komponen kontroler adalah mengurangi sinyal kesalahan, yaitu perbedaan antara nilai referensi/nilai yang diinginkan dan nilai aktual. Hal ini sesuai dengan

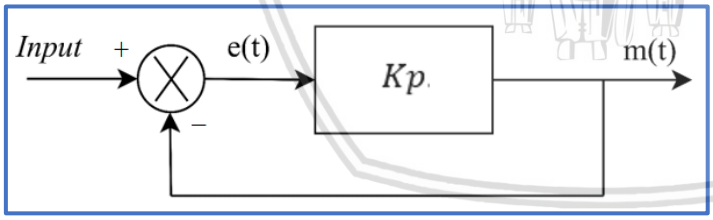
tujuan sistem kontrol mendapat nilai sinyal keluaran sama dengan nilai yang diinginkan referensi. Semakin kecil kesalahan yang terjadi, semakin baik kinerja sistem kontrol yang diterapkan.

Apabila perbedaan antara nilai referensi dengan nilai keluaran relatif besar, maka kontroler yang baik seharusnya mampu mengamati perbedaan ini untuk segera menghasilkan sinyal keluaran untuk mempengaruhi *plant*. Dengan demikian sistem secara cepat mengubah keluaran plant sampai diperoleh selisih dengan nilai referensi sekecil mungkin.

Prinsip kerja kontroler adalah membandingkan nilai aktual keluaran plant dengan nilai referensi, kemudian menentukan nilai kesalahan dan akhirnya menghasilkan sinyal kontrol untuk meminimalkan kesalahan (Ogata, K, 2002).

2.8.1 Kontroler *Proportional*

Kontroler *Proporsional* memiliki keluaran yang sebandiong/*proporsional* dengan besarnya sinyal kesalahan/*error*. Dapat dikatakan bahwa keluaran kontroler proporsional merupakan perkalian antara konstanta *proporsional* dengan masuknya. Perubahan sinyal masukan akan segera menyebabkan sistem secara langsung mengubah keluarannya sebesar konstanta pengalinya. Untuk lebih jelasnya dapat dilihat pada diagram blok pada gambar 2.11.



Gambar 2.11 Diagram Blok Kontroler *Proportional*
Sumber: Ogata, K, 2002

Pada gambar 2.11 menunjukan diagram blok yang menggambarkan hubungan antara input (besaran yang diinginkan), besaran aktual dengan besaran keluaran kontroler *proporsional*, dan besaran kesalahan (*error*). Sinyal kesalahan (*error*) merupakan selisih antara besaran setting dengan besaran aktualnya.

Pada pengendali *proporsional* hubungan antara keluaran kontroler m(t) dan sinyal keluaran e(t) adalah sebagai berikut :

$$M(t) = Kp\ e(t) \qquad (2\text{-}1)$$

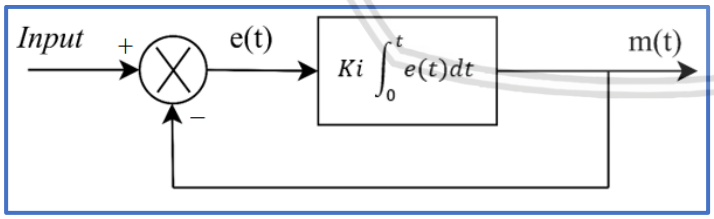
Dengan Kp adalah penguatan *proporsional*, keluaran m(t) hanya bergantung pada Kp dan *error*. Semakin besar *error* maka semakin besar koreksi yang dilakukan. Penambahan Kp akan menaikan penguatan sistem sehingga dapat digunakan untuk memperbesar kecepatan respon dan mengurangi kesalahan keadaan mantap.

Penggunaan kontrol P memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini. Walaupun demikian dalam aplikasi-aplikasi dasar yang sederhana kontrol P ini cukup mampu untuk memperbaiki respon transien khususnya *rise time* dan *settling time*.

2.8.2 Kontroler *Integral*

Kontroler *integral* berfungsi mengurangi kesalahan keadaan mantap yang dihasilkan pada kontroler *proporsioanal* sebelumnya. Kalau sebuah plant tidak memiliki unsur integrator (1/s), kontroler *proporsional* tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan mantapnya nol.

Kontroler *integral* memiliki karakteristik seperti halnya sebuah *integral*. Keluaran kontroler sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan. Keluaran kontroler ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Pada gambar 2.12 menunjukkan diagram blok kontroler *integral*.



Gambar 2.12 Diagram Blok Kontroler *Integral*
Sumber: Ogata, K, 2002

Nilai keluaran kontroler m(t) sebanding dengan *integral* sinyal kesalahan e(t), sehingga:

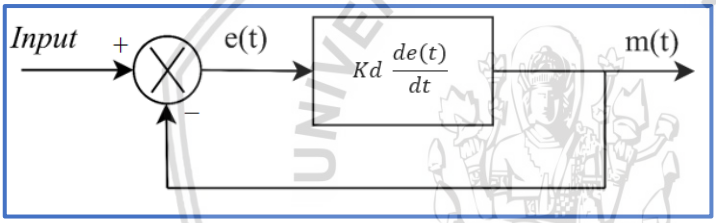
$$\frac{dm(t)}{dt} = Ki.e(t) \qquad (2\text{-}2)$$

$$m(i) = Ki \int_0^t e(t)dt \qquad (2\text{-}3)$$

Dengan Ki adalah konstanta *integral*. Jika sinyal kesalahan e(t)=0, maka laju perubahan sinyal kendali *integral* $\frac{dm(t)}{dt} = 0$ atau sinyal keluaran kendali akan tetap berada pada nilai yang dicapai sebelumnya. Aksi kontrol *integral* digunakan untuk mengilangkan kesalahan posisi dalam keadaan mantap (*error steady state*) tanpa memperhitungkan kecepatan respon.

2.8.3 Kontroler *Defferensial*

Kontroler *differensial* memiliki sifat seperti halnya suatu oprasi *derivative*. Perubahan yang mendadak pada masukan kontroler, akan mengakibatkan perubahan yang sangat besar dan cepat. Gambar 2.13 berikut menunjukkan diagram blok pada kontroler *differensial*.



Gambar 2.13 Diagram Blok Kontroler *Differensial*
Sumber: Ogata, K, 2002

Nilai keluaran kontroler m(t) sebanding laju sinyal kesalahan $\frac{de(t)}{dt}$. Hubungan ini dapat ditulis sebagai:

$$m(t) = Kd \frac{de(t)}{dt} \qquad (2\text{-}4)$$

Kontroler *differensial* akan memberikan sinyal kendali keluaran m(t)=0, untuk sinyal kesalahan e(t) yang konstan sehingga *differensial* tidak mempengaruhi keadaan mantap. Kontroler *differensial* digunakan untuk memperbaiki atau mempercepat respon transien sebuah sistem serta dapat meredam osilasi.

Berdasarakan karakteristik kontroler tersebut, kontroler *differensial* umumnya dipakai untuk mempercepat respon awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan tunaknya. Kerja kontroler *differensial* hanyalah efek dari lingkup yang sempit,

yaitu pada periode peralihan. Oleh sebab itu kontroler *differensial* tidak bias digunakan tanpa ada kontroler lain.

2.8.4 Kontroler *Proporsional Integral* (PI)

Aksi kontrolernya dinyatakan dalam persamaan:

$$m(t) = Kp.e(t) + \frac{Kp}{Ti} \int_0^t e(t)dt \qquad (2\text{-}5)$$

Kontroler ini menghasilkan sinyal kesalahan $\int_0^t e(t)dt$ kemudian ditambahkan dengan sinyal kesalahan e(t).

2.8.5 Kontroler *Proporsional Differensial* (PD)

Aksi kontrolnya dinyatakan dalam persamaan :

$$m(t) = Kp.e(t) + Kp.Td\ \frac{de(t)}{dt} \qquad (2\text{-}6)$$

Kontroler PD selalu mengukur kemiringan (slope) sinyal kesalahan $\frac{de(t)}{dt}$ dan memperkirakan akan besar *overshoot* yang akan terjadi serta memberikan koreksi sebelum terjadi lewatan sebenarnya sehingga diperoleh maximum *overshoot* yang kecil.

Jika kesalahan keadaan mantap tidak berubah terhadap waktu maka turunan terhadap waktu sama dengan nol, sehingga kontroler PD tidak mempunyai pengaruh terhadap kesalahan keadaan mantap, tetapi jika terdapat perubahan kesalahan, kontroler PD akan mengurangi besar kesalahan keadaan mantap. Jadi kontrol PD digunakan untuk memperbaiki suatu sistem pengendalian yang tanggapan peralihannya mempunyai maximum *overshoot* yang berlebihan tanpa memperhitungkan kecepatan responnya.

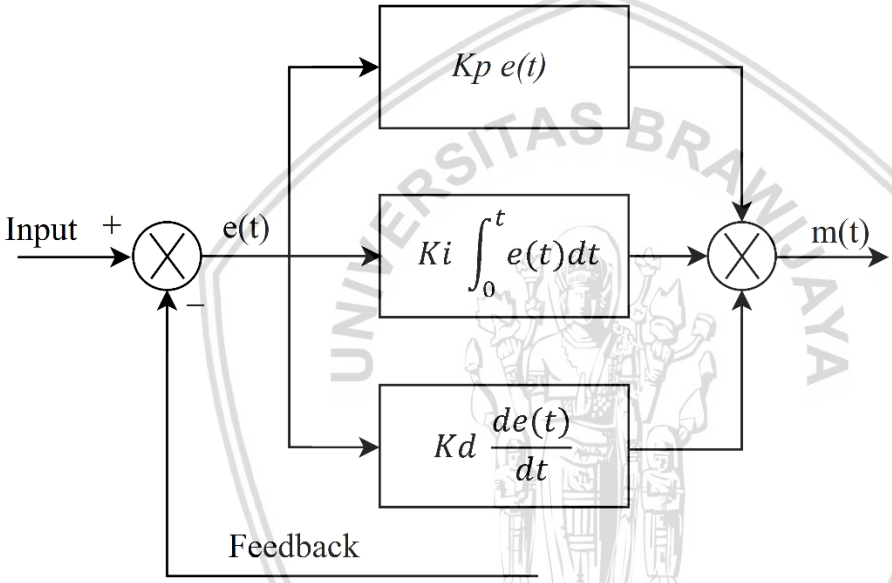
2.8.6 Kontroler PID

Aksi kontrol PID (*Proportional, Integral, Differensial*) banyak ditemukan di dunia industri dan satu–satunya strategi yang paling banyak diadopsi pada pengontrolan proses. Berdasarkan survei, 97% industri yang bergerak dalam bidang proses (seperti kimia, *pulp*, makanan, minyak, dan gas) menggunakan PID sebagai komponen utama dalam pengontrolannya. PID (dari singkatan bahasa Inggris: *Proportional–Integral–Differensial*

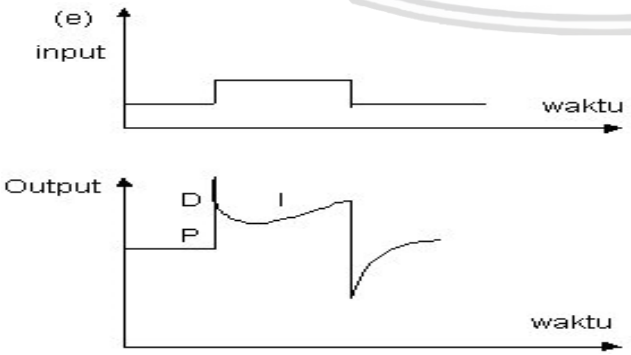
controller) merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut.

Setiap kekurangan dan kelebihan masing-masing kontroler P, I dan D dapat saling menutupi dengan menggabungkan ketiganya secara paralel menjadi kontroler *proporsional integral derivative* (PID). Elemen-elemen P, I dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem,menghilangkan offset dan menghasilkan perubahan awal yang besar (Gunterus, 1994).

Komponen kontrol PID ini terdiri dari tiga jenis yaitu *Proportional*, *Integral* dan *Differensial*.



Gambar 2.14 Diagram Blok Sistem Kontrol PID (*Proportional, Integral & Differensial*)
Sumber : http://id.wikipedia.org/wiki/PID



Gambar 2.15 Hubungan fungsi waktu antara sinyal keluaran dan masukan kontroler PID
Sumber: Gunterus, 1994

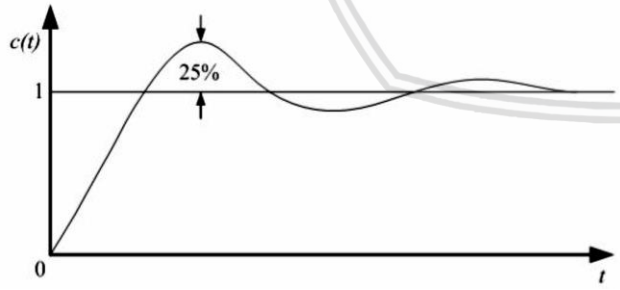
Aksi kontrolnya dapat dinyatakan sebagai berikut:

$$m(t) = K_p e(t) + \frac{Kp}{Ti} \int_0^t e(t)dt + K_p T_d \frac{de(t)}{dt} \quad (2\text{-}7)$$

Jenis kontroler ini digunakan untuk memperbaiki kecepatan respon dan mencegah terjadinya kesalahan keadaan mantap serta mempertahankan kestabilan. Keluaran kontroler PID merupakan penjumlahan dari keluaran kontroler proporsional, integral dan defferensial. Karakteristik kontroler PID sangat dipengaruhi oleh konstribusi besar dari ketiga parameter P,I dan D. Penyetelan konstanta Kp, Ki dan Kd akan mengakibatkan penonjolan sifat dari masing-masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat disetel lebih menonjol dibandingkan yang lain. Konstanta yang menonjol itulah yang akan memberikan kontribusi pengaruh pada respon sistem secara keseluruhan (Gunterus, 1994).

2.8.7 Metode Perancangan Kontroler *Proporsional, Integral, dan defferensial* (PID) Menggunakan Metode Ziegler-Nichols

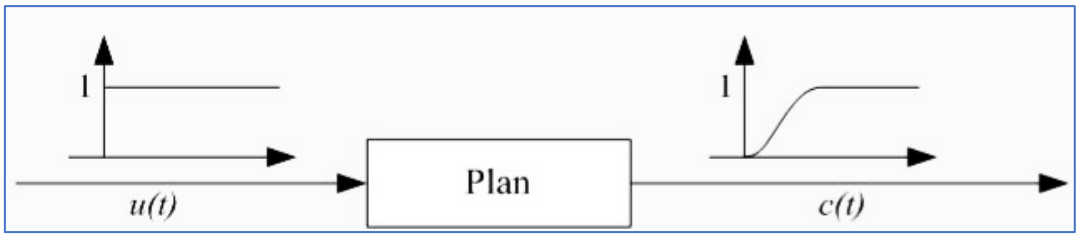
Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional Kp, waktu integral Ti, dan waktu derivatif Td berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan plan (Ogata, K., 1997). Terdapat dua metode yang disebut dengan aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step, seperti ditunjukkan dalam Gambar 2.16.



Gambar 2.16 Kurva Respon Unit Step yang Menunjukkan 25% *Maximun Overshoot*
Sumber: Ogata, K, 1997

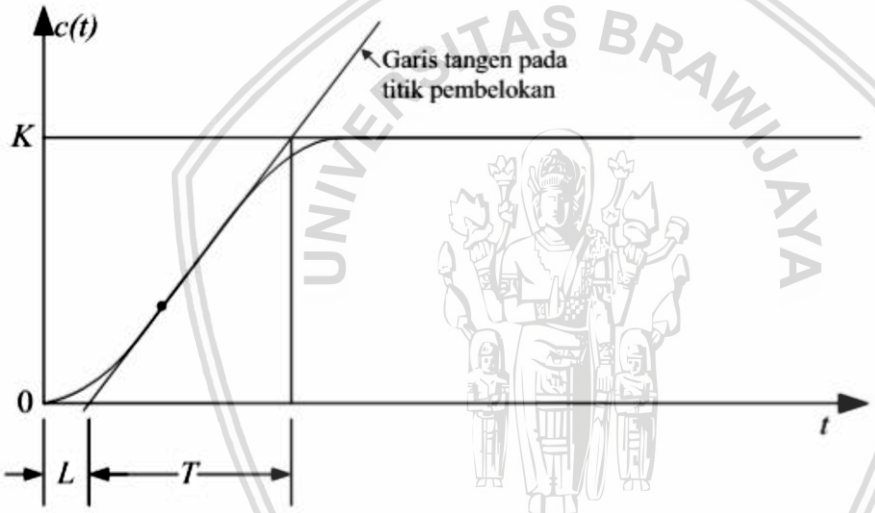
a) Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari plan dapat dapat diperoleh secara eksperimental dengan masukan berupa unit step, seperti yang ditunjukkan dalam Gambar 2.17.



Gambar 2.17 Respon *Plant* Terhadap Masukan Berupa Unit Step
Sumber: Ogata, K. 1997

Jika dalam plan tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit step berbentuk seperti huruf S, jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku.(Ogata, K., 1997). Seperti dalam Gambar 2.18.



Gambar 2.18 Kurva Respon yang Berbentuk S
Sumber: Ogata, K. 1997

Kurva berbentuk S tersebut dapat dikarakteristikkan menjadi dua konstanta yaitu waktu tunda *L* dan konstanta waktu *T*. Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu *t* dan sumbu *c(t)* = *K*, seperti yang telah ditunjukkan dalam Gambar 2.18 Fungsi alih $\frac{C(s)}{U(s)}$ dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts+1}$$

Sumber: Ogata, K. 1997

Ziegler dan Nichols menyarankan untuk menentukan nilai-nilai dari Kp, Ti dan Td berdasarkan pada formula yang ditunjukkan dalam Tabel 2.2 (Ogata, K., 1997).

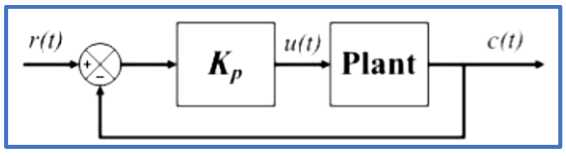
Tabel 2.2
Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step Dari *Plant*

Tipe Kontroler	K_p	T_i	Td
P	$\frac{T}{L}$	∞	0
PI	$0{,}9\,\frac{T}{L}$	$\frac{L}{0{,}3}$	0
PID	$1{,}2\,\frac{T}{L}$	$2L$	$0{,}5\,L$

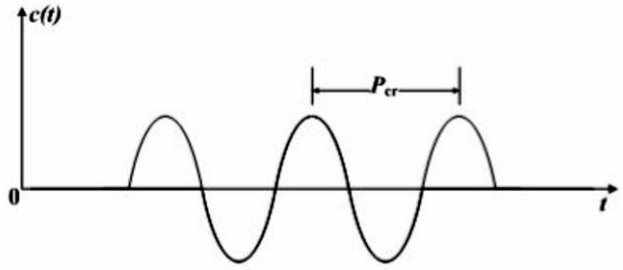
Sumber: Ogata, K. 1997

b) Metode Kedua

Dalam metode kedua ziegler-nichols, mula-mula yang dilakukan adalah membuat Ti = ∞ dan Td = 0. Kemudian hanya dengan menggunakan tindakan kontrol *proporsional*, harga ditingkatkan dari nol ke suatu nilai kritis *K*cr, disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai Kp manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis *K*cr dan periode *P*cr dapat ditentukan. Diagram blok sistem loop tertutup dengan kontroler *proporsional* dapat dilihat dalam Gambar 2.19 dan untuk osilasi berkesinambungan dengan periode Pcr dapat dilihat dalam gambar 2.20 Ziegler dan Nichols menyarankan penyetelan nilai parameter Kp, Ti, Td dan berdasarkan rumus yang diperlihatkan dalam Tabel 2.3 (Ogata, K., 1997).



Gambar 2.19 Sistem *Loop* Tertutup dengan Kontroler *Proporsional*
Sumber: Ogata, K., 1997



Gambar 2.20 Osilasi Berkesinambungan dengan Periode *P*cr
Sumber: Ogata, K., 1997

Tabel 2.3
Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain Kcr* dan *Critical Period Pcr*

Tipe Kontroler	K_p	T_i	Td
P	0.5 Kcr	∞	0
PI	0.45 Kcr	$\frac{1}{1,2}$ Pcr	0
PID	0.60 Kcr	0.5 Pcr	0.125 Pcr

Sumber: Ogata, K., 1997

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

BAB III
METODE PENELITIAN

Penyusunan pada skripsi ini merupakan penelitian yang bersifat aplikatif, yaitu merancang suatu sistem pengendali kecepatan putaran pada *engine* dengan menggunakan Kontroler *Proporsional Integral Differensial* (PID). Perancangan perangkat tersebut meliputi perangkat keras maupun perancangan perangkat lunak.

Untuk menyelesaikan rumusan masalah dan tujuan dari penelitian maka diperlukan langkah-langkah untuk merealisasikan alat yang dirancang yaitu penentuan spesifikasi alat, studi literatur, perancangan dan pembuatan alat, pengujian alat, dan pengambilan kesimpulan.

3.1 Spesifikasi Alat

Sebelum melakukan perencanaan dan pembuatan alat, ditentukan spesifikasi alat yang akan dibangun terlebih dahulu. Hal ini bertujuan sebagai acuan untuk mempermudah dalam membangun alat agar alat yang dibangun sesuai dengan yang diinginkan. Adapun spesifikasi alat yang akan dibangun dan direalisasikan adalah sebagai berikut:

1. *Engine* yang digunakan yaitu Motor Bakar 2 tak berbahan bakar premium yang berfungsi sebagai mesin penggerak.
2. Sensor yang digunakan yaitu *Hall Effect* yang berfungsi untuk mengatur kecepatan pada *engine*.
3. Kontroler PID digunakan untuk mengontrol kecepatan putaran pada *engine*.
4. Servo digunakan untuk mengatur berapa persen bukaan pada *throttle*.
5. Bagian *Propeller* menggunakan *tail helicopter gear*.
6. Arduino ATMega 2560 sebagai tempat pemrograman Kontrol PID.

3.2 Studi Literatur

Pengetahuan yang mendukung perencanaan dan realisasi alat meliputi :

1. *Engine* (Motor Bakar 2 Tak)
2. Sensor *Hall Effect*
3. Motor Servo
4. Kontroler PID

5. *Propeller*
6. Arduino ATMega 2560

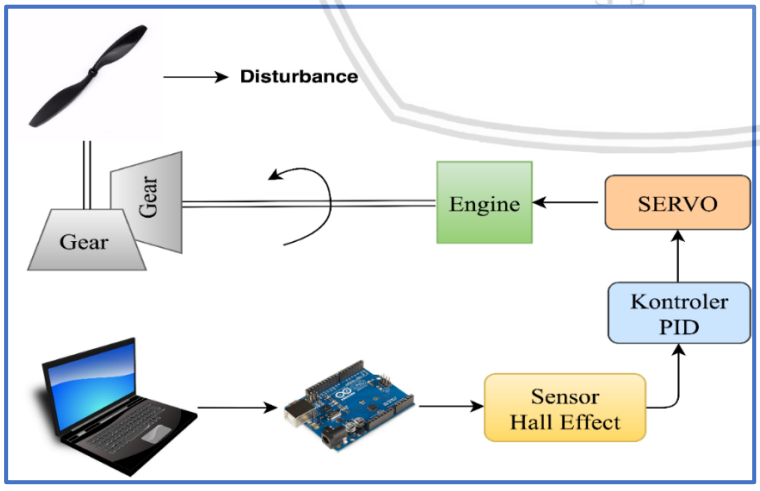
3.3 Perancangan Dan Perealisasian Alat

Perancangan sistem kontrol kecepatan secara keseluruhan mengikuti diagram blok seperti yang diperlihatkan pada gambar 3.2 guna memudahkan analisis pada setiap bloknya maupun secara keseluruhan sistem. Perancangan ini terdiri atas:

1. Perancangan perangkat keras (sensor *hall effect*, *propeller*, *engine* dan motor servo).
2. Perancangan perangkat lunak (perancangan algoritma PID pada software Arduino ATMega).

3.3.1 Perancangan Sistem mekanik

Sebelum mendesain *bicopter* secara keseluruhan diperlukan sistem kontrol kecepatan putaran *engine*. Sistem kontrol kecepatan setiap *engine* ini menggunakan 1 aktuator motor servo, 1 *propeller*, 1 mesin penggerak *engine* (motor bakar 2 tak) dan sensor *hall effect*. Dengan menggunakan kontroler PID, dapat diuji kestabilan pada kecepatan putaran *engine* yang dihasilkan. Sehingga kecepatan putaran *engine* diatur dengan kontroler PID agar kecepatan putaran *engine* tetap ketika pembebanan berubah-ubah. Desain alat diperlihatkan pada gambar 3.1.



Gambar 3.1 Desain Alat

Prinsip kerja alat seperti dalam gambar 3.1 yaitu sebagai berikut:

1. Memberikan nilai *setpoint* berupa tegangan analog 0 – 5V.

2. Tegangan analog 0 – 5V di ubah oleh adc menjadi digital. Dimana 0 – 5V ekivalen dengan nilai rpm 0% (600 rpm) – 80% (8000 rpm).
3. Kecepatan putaran pada *engine* akan berputar pada kecepatan yang diinginkan.
4. Ketika ada *disturbance*/gangguan *thrust* maka kecepatan putaran *engine* akan turun otomatis sensor akan akan membaca rpm.
5. Kemudian keluaran dari sensor berupa periode pulsa yang nantinya akan diubah menjadi rpm oleh arduino. Di arduino terjadi *numerical* yaitu pengurangan nilai *setpoint* dengan hasil sinyal *feedback* dari sensor menggunakan kontroler PID.
6. Keluaran dari arduino berupa PWM yang akan menjadi masukan servo.
7. Servo akan mengatur/menset sudut pada bukaan pada *throttle*.
8. Sehingga kecepatan putaran *engine* kembali kepada kecepatan semula jadi nilai *setpoint* dan *output* sabanding.

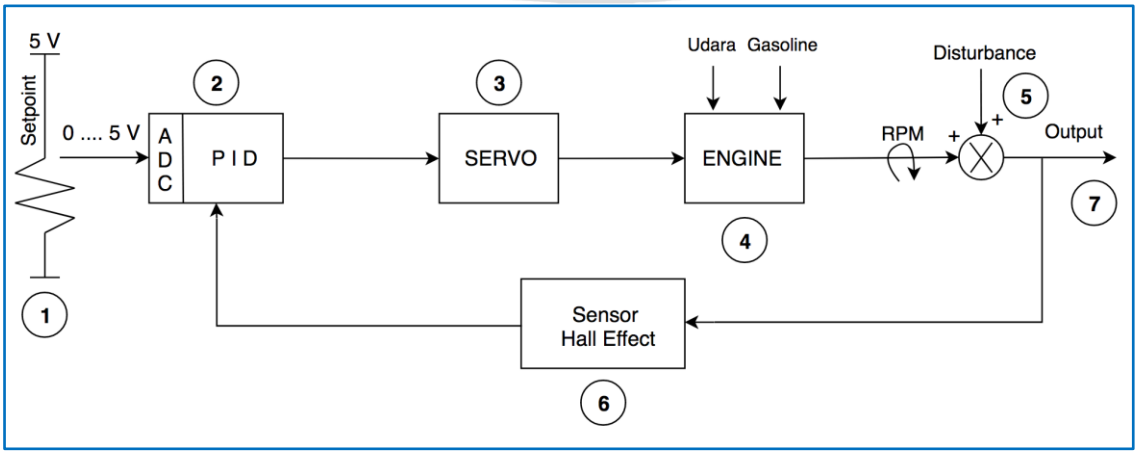
3.3.2 Perancangan Sistem Kontroler

Spesifikasi alat yang di rancang adalah sebagai berikut:

1. Arduino ATMega dan *software* Arduino untuk pengembangan kontroler *Proporsional Integral Differensial* (PID).
2. Sensor yang digunakan berupa sensor *hall effect* berfungsi sebagai *feedback* pengontrolan kecepatan.

3.3.3 Perancangan Diagram Blok Sistem Keseluruhan

Hasil perancangan dalam skripsi ini dibangun secara diagram blok seperti yang diperlihatkan pada gambar 3.2.



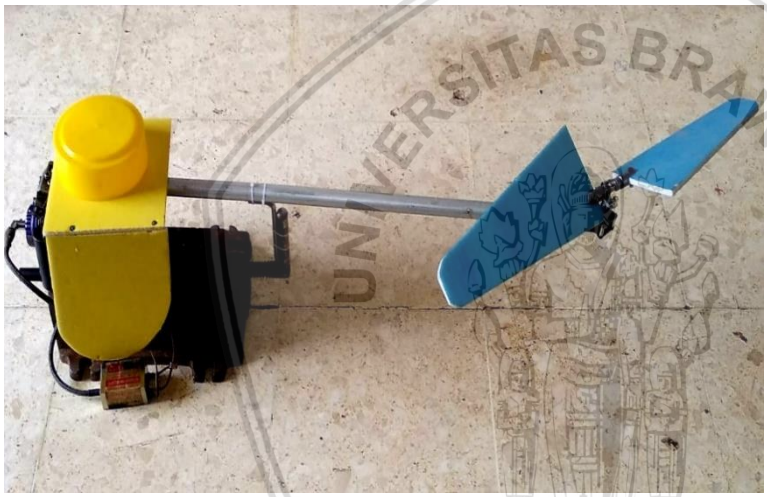
Gambar 3.2 Diagram Blok Kendali Sistem

Keterangan dari diagram blok pada Gambar 3.2:

1) *Setpoint* berfungsi untuk mengeset rpm yang diinginkan dengan menggunakan tegangan analog yang diasup oleh potensiometer dengan tegangan 0-5V ekivalen dengan nilai rpm yang di inginkan. Mulai dari 0% (600 Rpm) – 100% (8000 Rpm).
2) Kontroler PID menggunakan Arduino ATMega 2560 yang berfungsi untuk menjalankan pid sesuai tegangan analog dari *setpoint* untuk diubah oleh adc menjadi nilai berupa rpm. Kemudian pid menerima pulsa dari *hall effect* sensor untuk diterjemahkan menjadi rpm. Selanjutnya *error* diperoleh setelah dikurangi oleh nilai *setpoint*.
3) Motor Servo berfungsi sebagai aktuator untuk mengeset bukaan pada *throttle*. Motor servo dapat di *set-up* atau diatur untuk menentukan dan memastikan posisi sudut dari poros *output*. Posisi poros *output* akan disensor untuk mengetahui posisi poros sudah tepat seperti yang diiginkan atau belum. Jika belum maka kontrol *input* akan mengirim sinyal kendali untuk membuat posisi poros yang diinginkan.
4) *Engine* menggunakan motor bakar 2 tak yang berfungsi sebagai mesin penggerak. *Engine* menghasilkan putaran *propeller* yang mengakibatkan adanya daya dorong. *Throttle* pada *engine* berfungsi untuk menghalangi udara. Pada *gasoline engine* dilengkapi dengan karburator sebagai tempat pencampuran udara dan bahan bakar. Campuran udara dan bahan bakar dihisap ke dalam ruang bakar dan dikompresikan hingga mencapai tekanan dan temperatur tertentu. Pada akhir langkah kompresi, busi memercikkan api sehingga terjadi pembakaran.
5) *Disturbance* berupa sudut *pitch propeller* yang diubah-ubah secara gangguan. Sehingga menyebabkan pembebanan berubah-ubah yang mengakibatkan rpm akan turun.
6) Sensor *hall effect* berfungsi untuk mengatur kecepatan. Keluaran dari sensor berupa pulsa kemudian diubah menjadi rpm oleh arduino. Keluaran dari sensor disebut sinyal *feedback*. Hasil akhir sinyal *feedback* kemudian dikurangkan dengan nilai *input*/*setpoint* sehingga mikrokontroler mampu meminimalisir *error* yang terjadi.
7) *Output* dari mekanik berupa kecepatan putaran pada *engine* yang di kopel untuk menggerakan *propeller*.

3.4 Perealisasian Alat

Perancangan sistem mekanik yaitu dengan membuat model menyerupai *bicopter* tetapi hanya 1 lengan saja. Model ini memiliki 1 lengan, yaitu di satu lengan terdapat *engine*, servo, sensor *hall effect* dan *propeller*. *Propeller* dibuat sebagai gangguan yaitu dengan cara mengubah-ubah sudut *pitch*nya agar pembebanan berubah-ubah. Dengan menggunakan kontroler PID, dapat diuji kestabilan kecepatan putaran *engine* pada *bicopter* ketika diberi gangguan. Sehingga kecepatan putaran *engine* dapat diatur pada kecepatan konstan/tetap ketika sudut *pitch* berubah-ubah. Adapun rancangan mekanik pada keseluruhannya dapat dilihat pada gambar 3.3.

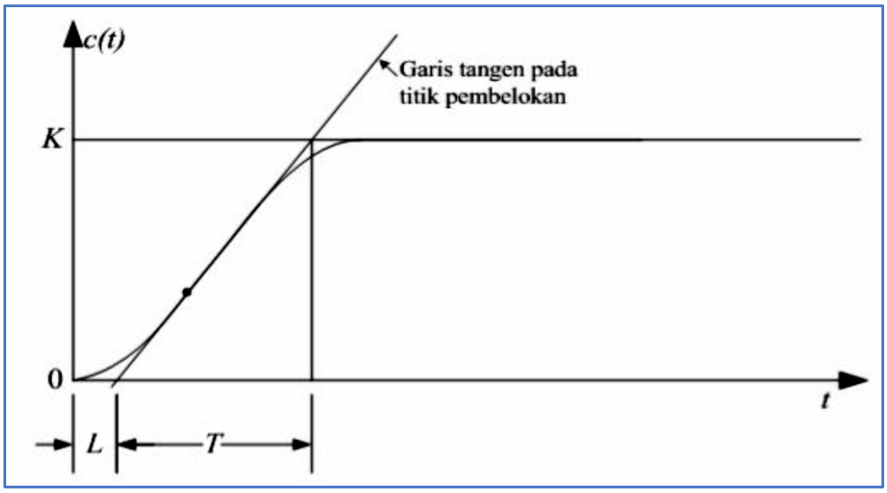


Gambar 3.3 Lengan *Bicopter*

3.5 Perancangan kontroler PID

3.5.1 Penentuan Nilai Penguatan Kontroler

Untuk menentukan penguatan kontroler digunakan metode *Ziegler-Nichols* yang mengemukakan aturan-aturan untuk menentukan nilai dari *gain proporsional* Kp, waktu *integral* Ti, dan waktu *derivative* Td berdasarkan karakteristik respon transien dari plant yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan *plant* (Ogata, K., 1997). Pada pengerjaan skripsi ini digunakanlah metode *Ziegler-Nichols* yang pertama sehingga mendapatkan sinyal kurva S seperti pada Gambar 3.4.



Gambar 3.4 Kurva Respon yang Berbentuk S **(**Ogata, K., 1997**)**

Kurva berbentuk S tersebut dapat dikarakteristikkan menjadi dua konstanta yaitu waktu tunda *L* dan konstanta waktu *T*. Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis *tangent* pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis *tangent* dengan sumbu waktu *t* dan sumbu *c(t)* = K.

Langkah menentukan *tuning* parameter dengan metode 1 *Ziegler-Nichols* adalah sebagai berikut :

1. Menarik garis *tangent* pada titik infleksi grafik karakteristik *open loop*.
2. Menentukan perpotongan garis *tangent* terhadap sumbu waktu *t* untuk mendapatkan *L*.
3. Menentukan perpotongan garis *tangent* terhadap sumbu *steady* untuk mendapatkan nilai *T*.
4. Setelah mendapatkan Nilai *L* dan *T* digunakan untuk menentukan nilai Kp, Ti dan Td sesuai dengan Tabel 3.1.

Tabel 3.1
Aturan Metode 1 Ziegler-Nichols (Ogata., 1997)

Tipe Kontroler	K_p	T_i	Td
P	$\frac{T}{L}$	∞	0
PI	$0{,}9\,\frac{T}{L}$	$\frac{L}{0{,}3}$	0
PID	$1{,}2\,\frac{T}{L}$	$2L$	$0{,}5\,L$

Parameter kontroler diperoleh sebagai berikut:

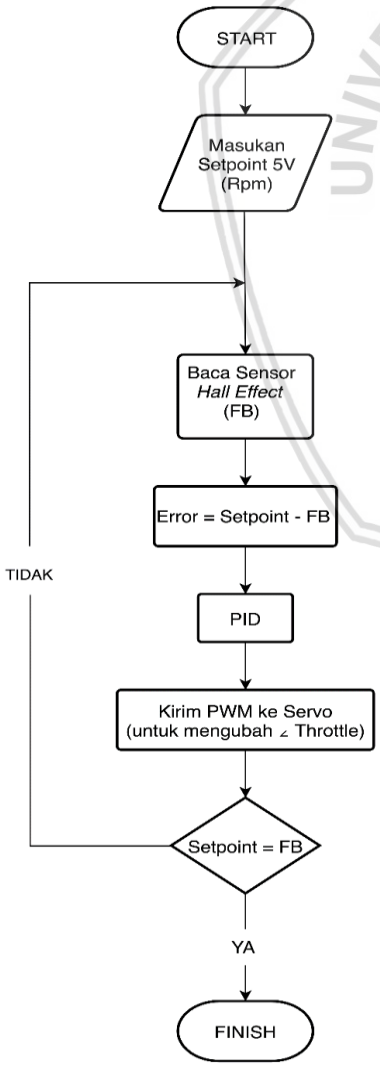
$$K_p = 1{,}2\ \frac{T}{L} \quad (3\text{-}1)$$

$$K_i = \frac{Kp}{T_i} \quad (3\text{-}2)$$

$$K_d = Kp \times T_d \quad (3\text{-}3)$$

3.6 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada pengendalian ini dengan menggunakan bahasa pemrograman C++ dengan menggunakan *software* Arduino. *Tuning* kontroler PID adalah dengan memasukkan nilai Kp, Ki, dan Kd dari hasil perhitungan metode *Ziegler-Nichols*. *Flowchart* perancangan perangkat lunak dapat dilihat pada Gambar 3.5.



Gambar 3.5 Flowchart Perangkat Lunak

Bila dijelaskan lebih detail, dengan menggunakan simbol dan keterangan *flowchart* dapat dijabarkan langkah-langkah yang bisa dilakukan oleh user sebagai berikut:

1. *User* pilih "START" yang diwakili oleh simbol *terminator* yang menggambarkan kegiatan awal atau akhir dari suatu proses. Pada langkah ini simbol *terminator* menjabarkan kegiatan awal program memberi masukan sebesar 0-5 volt dan nilai Rpm yang diinginkan.
2. Kemudian membaca nilai sensor *hall effect* (*feedback*) dimana diwakili oleh simbol proses yang berfungsi menggambarkan suatu proses.
3. Setelah mendapatkan nilai sensor *hall effect*, kemudian melakukan proses berupa nilai *setpoint* Rpm awal dikurangkan dengan nilai *hall effect* yang terbaca (*feedback*) menghasilkan nilai *error*.
4. nilai *error* sebagai masukan kontroler PID untuk menentukan gain Kp, Ki, dan Kd.
5. Kontroler PID melakukan proses mengirim sinyal pulsa PWM ke servo untuk menset bukaan pada *throttle*.

Setelah mengirim sinyal pulsa PWM, *user* disediakan dua kondisi yang harus dia pilih. Kondisi yang ada diwakili oleh simbol *decision*. *Decision* adalah simbol untuk menunjukkan sebuah langkah pengambilan keputusan. Umumnya, menggunakan bentuk pertanyaan, dan biasanya jawabannya terdiri dari '*yes*' dan '*no*' atau 'ya' dan 'tidak' yang menentukan bagaimana alur dalam *flowchart* berjalan selanjutnya berdasarkan kriteria atau pertanyaan tersebut. Jika kondisi sesuai dengan kriteria maka *user* memilih "*FINISH*" yang menggambarkan akhir dari proses ini. Jika kondisi tidak sesuai, maka *user* kembali ke proses membaca sensor *hall effect* (*feedback*). Proses terus berulang sampai kondisi sesuai dengan yang diinginkan.

BAB IV
HASIL DAN PEMBAHASAN

Tujuan pengujian sistem ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian pada sistem ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok ini dilakukan untuk menemukan letak kesalahan dan mempermudah analisis pada sistem apabila alat tidak bekerja sesuai dengan perancangan.

Adapun langkah – langkah pengujian yang dilakukan adalah:

1. Pengujian Sensor *Hall Effect*.
2. Pengujian Respon Kurva S pada *Engine*.
3. Pengujian Motor Servo terhadap *Engine*.
4. Pengujian Sistem Secara Keseluruhan.

4.1 Pengujian Sensor *Hall Effect*

4.1.1 Tujuan Pengujian

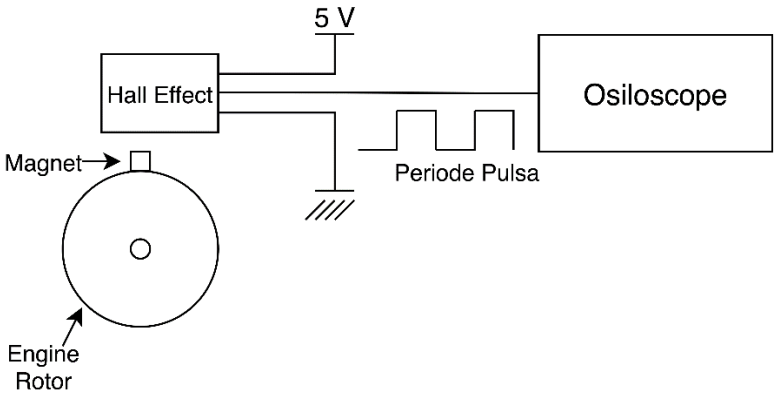
Menguji respon sensor *hall effect* berupa pulsa digital. Hasil berupa hubungan antara kecepatan dengan periode pulsa dalam ms.

1.1.2 Peralatan Yang Digunakan

1. Sensor *hall effect*.
2. *Osiloscope* digital.
3. *Engine*.

4.1.3 Langkah Pengujian

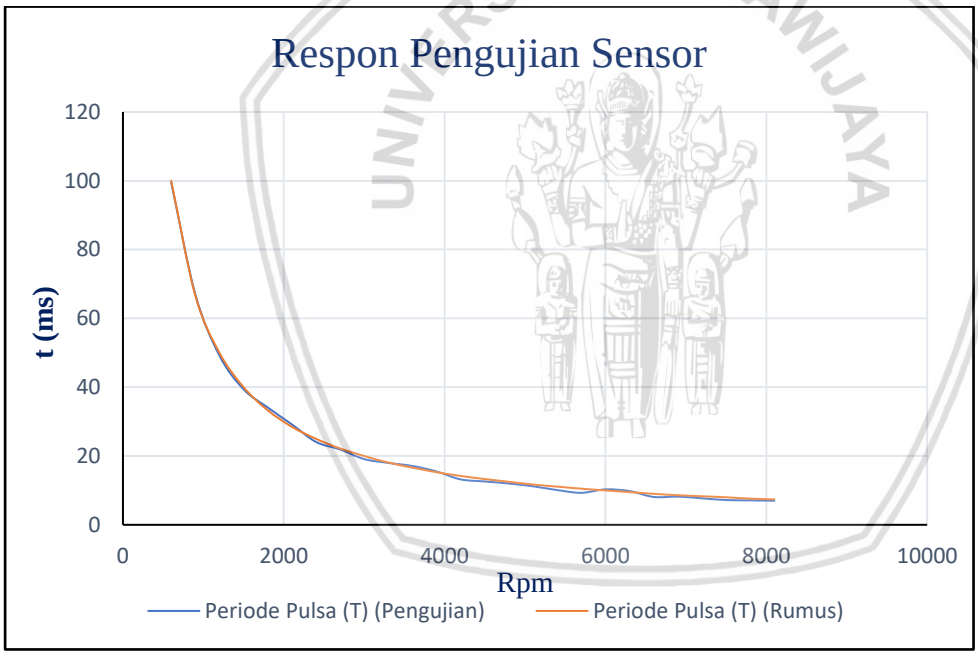
1. Merangkai peralatan pengujian seperti pada Gambar 4.1.
2. Sambungkan catu daya.
3. Mengatur kecepatan putaran *engine* mulai 600 rpm sampai 8000 rpm melalui servo untuk menset bukaan pada *throttle*.
4. Mengukur periode pulsa keluaran *hall effect* menggunakan *osiloscope* digital.



Gambar 4.1 Diagram Perancangan Peralatan Pengujian Sensor

4.1.4 Hasil Pengujian

Hasil pengujian didapatkan dengan melihat tegangan keluaran yang terbaca oleh sensor *hall effect* diperlihatkan pada gambar 4.2.



Gambar 4.2 Respon Pengujian Sensor *Hall Effect*

Dari hasil data di tabel 1 yang terdapat pada lampiran 1 secara teoritis sebenarnya dengan menggunakan rumus hasil perhitungannya tidak jauh dengan nilai pada hasil pengujian. Dalam pengujian terdapat *error* dengan rata-rata *error* sebesar 0,0832 ms.

$$0{,}0832 \text{ x } 100\% = 8{,}32\ \%$$

Presentase *error* yang didapat yaitu 8,32%.

4.2 Pengujian Respon Kurva S

4.2.1 Tujuan Pengujian

Menguji respon *engine* secara *open loop* untuk mendapatkan Kurva S.

4.2.2 Peralatan Yang Digunakan

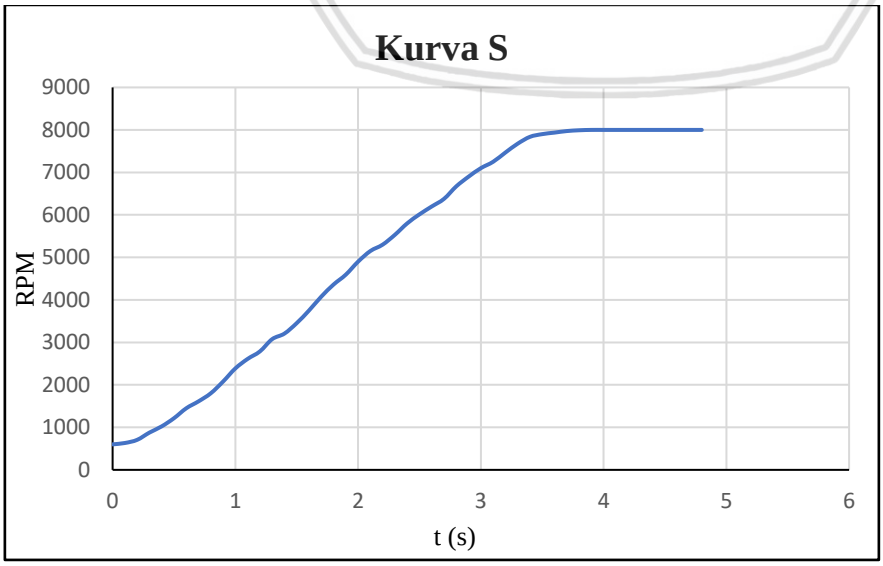
1. *Engine*.
2. *Osiloscope* digital.
3. Sensor.
4. Servo.

4.2.3 Langkah Pengujian

1. Menghubungkan *Osiloscope* digital dengan sensor.
2. Sambungkan catu daya.
3. *Engine start* dengan ideal 600 rpm.
4. Mengaktifkan motor servo pada bukaan *throttle* 80% sebagai fungsi step.
5. Kurva S dari respon putaran yang diukur oleh *Osiloscope* digambarkan dalam bentuk grafik.

4.2.4 Hasil Pengujian

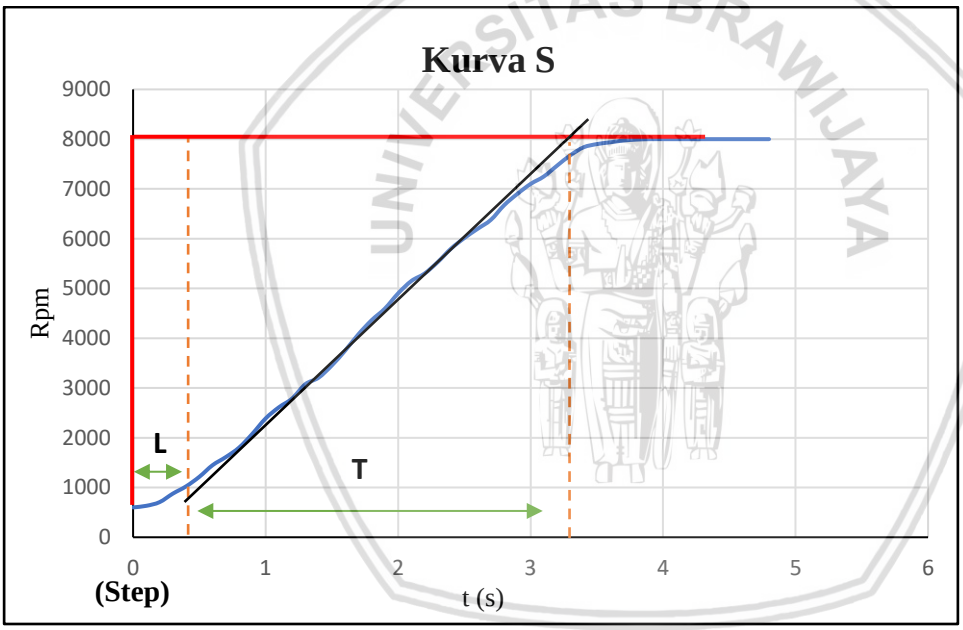
Setelah melakukan pengujian sesuai dengan data pada tabel 2 yang diperlihatkan di lampiran 1, didapatkan respon kurva S pada gambar 4.3.



Gambar 4.3 Respon Pengujian Kurva S

Dari hasil pengujian yang diperlihatkan dalam Gambar 4.3 berupa kurva S akan digunakan menentukan parameter *tuning* PID dengan metode 1 Ziegler-Nichols seperti yang diperlihatkan dalam Gambar 4.4 Adapun langkah-langkah yang dilakukan sebagai berikut :

1. Menarik garis *tangent* pada titik infleksi grafik karakteristik *open loop* seperti yang diperlihatkan dalam Gambar 4.4.
2. Menentukan perpotongan garis *tangent* terhadap sumbu waktu t untuk mendapatkan L.
3. Menentukan perpotongan garis *tangent* terhadap sumbu *steady* untuk mendapatkan nilai.
4. Nilai L dan T digunakan untuk menentukan nilai Kp Ti dan Td sesuai dengan Tabel 4.1.



Gambar 4.4 Penentuan parameter *tuning* PID pada respon Kurva S

Tabel 4.1
Aturan Metode 1 Ziegler-Nichols (Ogata K, 1997)

Tipe Kontrol	**Kp**	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0{,}9\ \frac{T}{L}$	$\frac{L}{0{,}3}$	0
PID	$1{,}2\ \frac{T}{L}$	2L	0,5L

Dari Gambar 4.4 diperoleh besarnya L = 0,4 s dan T = 3,3 – 0,4 = 2,9 s. Dengan demikian parameter kontroler diperoleh sebagai berikut :

$$\boldsymbol{K_p} = 1{,}2\ \frac{T}{L} = 1{,}2 \times \frac{2{,}9\ s}{0{,}4\ s} = 7{,}25$$

$$T_i = 2L = 2 \times 0{,}4\ s = 0{,}8$$

$$T_d = 0{,}5L = 0{,}5 \times 0{,}4\ s = 0{,}2$$

Selanjutnya akan diperoleh Ki dan Kd sebagai berikut :

$$\boldsymbol{K_i} = \frac{Kp}{Ti} = \frac{7{,}25}{0{,}8} = 9{,}06 \quad \boldsymbol{K_d} = Kp \times T_d = 7{,}25 \times 0{,}2 = 1{,}45$$

4.3 Pengujian Motor Servo terhadap *Engine*

4.3.1 Tujuan Pengujian

Pengujian ini dilakukan untuk mengetahui pengaruh perubahan sudut pada motor servo terhadap kecepatan putaran *engine*.

4.3.2 Peralatan Yang Digunakan

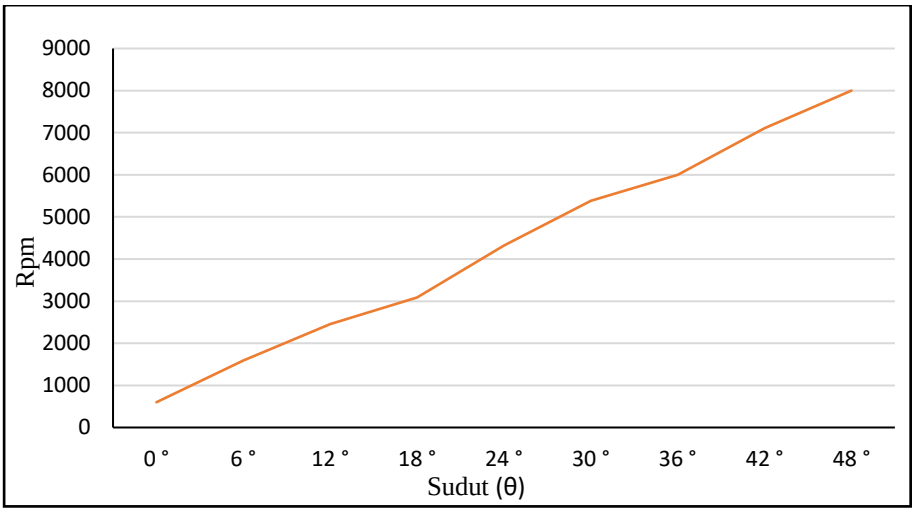
1. *Engine*.
2. Sensor *Hall Effect*.
3. Servo.
4. Arduino ATMega 2560.

4.3.3 Langkah Pengujian

1. Merangkai peralatan pengujian.
2. Load program.
3. Run program.
4. Hasil ditampilkan melalui serial com.
5. Hasil data di plot.

4.3.4 Hasil Pengujian

Pengujian ini adalah hubungan antara kecepatan *engine* terhadap sudut *throttle*. Sudut *throttle* digerakan oleh servo dimana sudut servo sama dengan sudut *throttle*. Sudut servo ditentukan oleh PWM yang diberikan dimana 0 – 255 sama dengan 0° – 60°. Setelah melakukan pengujian didapatkan respon hubungan sudut dengan rpm pada gambar 4.5.



Gambar 4.5 Grafik hubungan perubahan sudut servo dengan kecepatan putaran *engine*

Dari hasil pengujian yang dilakukan, dalam grafik masih terdapat beberapa gangguan yang membuat kelinieran kecepatan *engine* tidak sempurna. Tetapi secara keseluruhan dapat dilihat bahwa semakin besar perubahan sudut maka kecepatan putarannya semakin besar.

4.4 Pengujian Sistem Secara Keseluruhan

4.4.1 Tujuan Pengujian

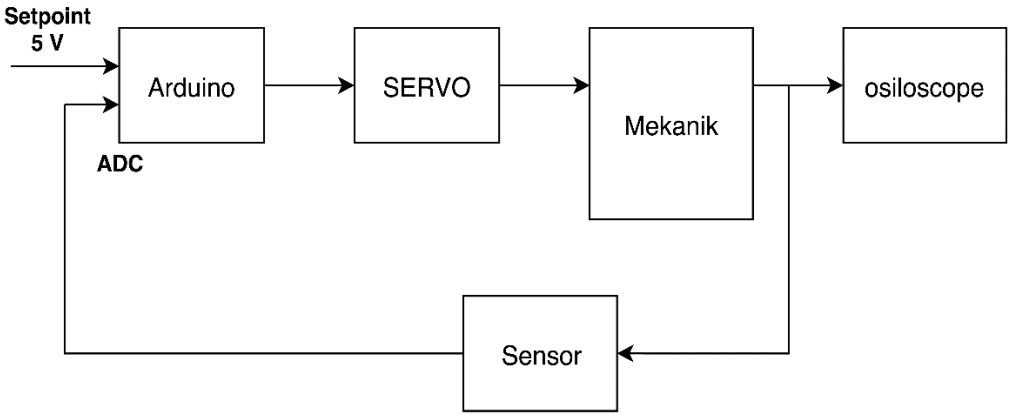
Pengujian ini bertujuan untuk mengetahui respon sistem secara *closed loop* dengan memberikan gangguan.

4.4.2 Peralatan Yang Digunakan

1. Catu Daya 5 V DC.
2. Mikrokontroler Arduino ATMega 2560.
3. Komputer.
4. *Digital Oscilloscope* dan software PC Lab 2000SE.
5. Program dan *software* Arduino.

4.4.3 Langkah Pengujian

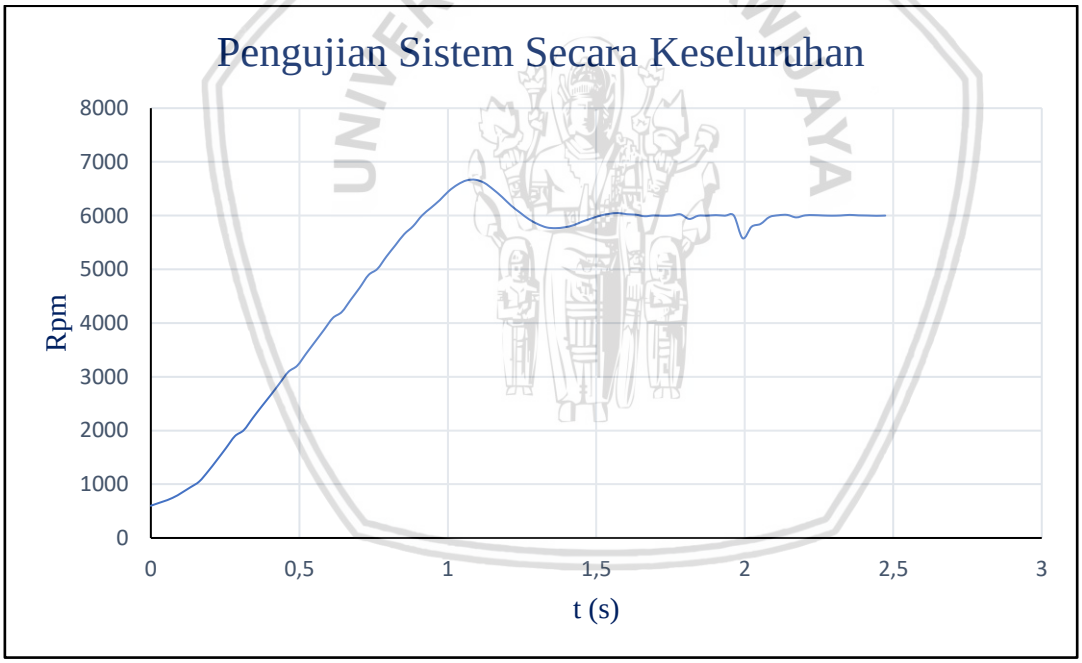
1. Merangkai peralatan seperti Gambar 4.6.
2. *Setpoint* diset pada 6000 rpm pada tegangan 3,64 V.
3. Mengunduh program pengatur kecepatan putaran *engine* pada *software* Arduino.
4. Menjalankan *software* PC Lab 2000SE.
5. Memilih mode *osiloscope* lalu pilih *Run* untuk menjalankan *osiloscope*.
6. Mengamati sinyal kontrol dan parameter motor pada *osiloscope*.



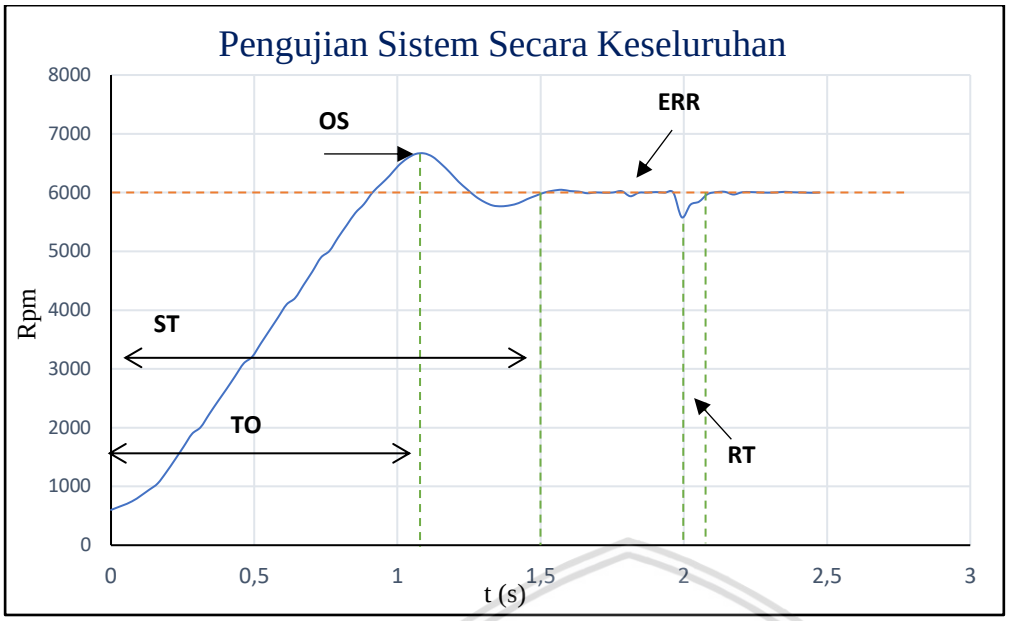
Gambar 4.6 Diagram Peralatan Pengujian *Engine*

4.4.4 Hasil Pengujian

Setelah melakukan pengujian sesuai dengan data pada tabel 3 yang diperlihatkan di lampiran 1, secara grafik hasil pengujian di perlihatkan pada gambar 4.7.



Gambar 4.7 Respon pengujian sistem secara keseluruhan



Gambar 4.8 Parameter-parameter kerja sistem

Dari hasil pengujian secara *closed loop* setelah diberi beban sistem mampu mencapai *steady state* pada waktu 1,5 s yang diperlihatkan pada Gambar 4.8, kemudian diberi gangguan dengan cara mengubah-ubah sudut *pitch*nya secara *random* (acak). Sehingga menghasilkan *error* sebesar 3,81% dan respon mampu mengembalikan kembali pada keadaan *steady* (setimbang) dengan *recovery time* sebesar 0,13 s. Untuk lebih jelasnya parameter pada kerja sistem yang diperoleh sebagai berikut:

1. *Time overshoot* (TO) adalah waktu yang diperlukan untuk respon mencapai puncak pertama *overshoot*. *Time overshoot* (TO) terjadi pada sekitar 1,2 s.
2. *Steady State* terjadi pada sekitar 1,5 s.
3. Besarnya *Maximum Overshoot* (OS) sebesar:

$$\frac{6667 - 6000}{6000} \times 100\% = 11{,}1\%$$

4. *Error* (ERR) *steady state* adalah nilai kesalahan saat respon telah mencapai pada keadaan tunak/*steady*. Presentase kesalahan dapat dicari dengan menggunakan pesamaan:

$$\%E = \frac{1}{N}\sum_{i=1}^{N} \frac{|\mathrm{PV} - \mathrm{SP}|}{\mathrm{SP}}\ x\ 100 \qquad (4\text{-}1)$$

Dimana :

E = *Error*

N = Jumlah data saat mencapai *setpoint*

PV = *Present value*

SP = *Setpoint*

Jadi *error steady state* yang didapat sebesar:

$$\%E = \sum_{i=1}^{30} \frac{|6000 - 5771|}{6000} \times 100\% = 3{,}81\ \%$$

5. *Recovery Time* (RT) adalah waktu yang dibutuhkan sistem untuk mengembalikan ke kondisi *steady state* setelah terjadinya *error*. Berdasarkan pengujian *recovery time* dengan gangguan secara acak adalah 0,13 s.

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

BAB V
PENUTUP

5.1 Kesimpulan

Kesimpulan yang dapat diambil dalam penelitian ini adalah sebagai berikut:

1. Mekanik yang digunakan adalah sebuah model yang menyerupai *bicopter* tetapi hanya 1 lengan saja. Model ini memiliki 1 lengan, yaitu di satu lengan terdapat *engine*, servo, sensor *hall effect* dan *propeller*. *Propeller* dibuat secara gangguan yaitu dengan cara mengubah-ubah sudut *pitch*nya agar pembebanan berubah-ubah.
2. Berdasarkan data respons sistem yang diperoleh dari pengujian dengan menggunakan metode 1 Ziegler-Nichols, maka parameter kontroler PID dapat ditentukan dengan *gain* Kp = 7,25 : Ki = 9,06 : Kd = 1,45. Pada pengujian sistem secara keseluruhan dengan memberikan gangguan berupa perubahan sudut *picth* mendapatkan nilai time *overshoot* sebesar 1,2 s, waktu saat *steady state* yaitu 1,5 s, *overshoot* sebesar 11,1% dan *error* 3,81 %. Secara keseluruhan sistem kecepatan pada *engine* dapat kembali pada keadaan *steady* dan mampu memberikan respon sistem yang baik ketika terjadi gangguan dengan *recovery time* sebesar 0,13 s.

5.2 Saran

Dalam perancangan dan pembuatan alat ini masih terdapat kelemahan. Untuk memperbaiki kinerja alat dan pengembangan lebih lanjut disarankan:

1. *Bicopter* dengan menggunakan *engine* ini dapat di kembangkan lagi dengan menggunakan 2 *engine* dengan 2 lengan menggerakan 2 *propeller* yang masing-masing 1 *engine* dengan 1 lengan menggerakan 1 *propeller*.
2. *Bicopter* ini dapat dikembangkan lagi karena ini masih tahap awal dalam pembuatan *bicopter* menjadi *bicopter* seutuhnya dan bahkan bisa dikembangkan menjadi pesawat UAV lainnya seperti *tricopter*, *quadcopter* dll.

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

DAFTAR PUSTAKA

Anugerah, Risha. 2016. *Flight Controller* Pada Sistem *Quadcopter* Menggunakan Sensor IMU (*Inertial Measurement Unit*) Berbasis Mikrokontroller ATMega 2560. Jurusan Teknik Elektro. Universitas Sanata Dharma. Yogyakarta.

Carillo, L. R. G., Lopez,A. E. D., Lozano, R. & Pegard, C. 2013. *Quad rotorcraft control vision-based hovering and navigation*, London: Springer.

D. Schmidt, Michael. 2011 *Simulation and control of a quadcopter Unmanned Aerial Vechile*, Tesis, University of Kentucky.

Fadillah, Ariski. 2016. Perancangan Kesetimbangan Gerak *Pitch* Pada *Bicopter* Secara Statis Dengan Metode Mengubah-Ubah *Pitch Propeller* Menggunakan Kontroler PID. *Sripsi*. Tidak dipublikasikan. Malang: Universitas Brawijaya.

Gunterus, Frans. 1994. *Falsafah Dasar : Sistem Pengendalian Proses*. Jakarta: Elex Media Komputindo.

Kho, Dickson. Pengertian Sensor Efek Hall (Hall Effect Sensor) dan Prinsip Kerjanya. https://teknikelektronika.com/pengertian-sensor-efek-hall-hall-effect-sensor-prinsip-kerja-efek-hall/. Diakses 5 Maret 2018.

Modul Praktikum Sistem Kontrol. *Praktikum Sistem Kontrol Dasar*. Malang: Laboratorium Sistem Kontrol Teknik Kontrol Teknik Elektro Universitas Brawijaya.

Ogata, K. 1997. Teknik kontrol Automatik. terjemahan: Edi Laksono Ir. Jakarta : Penerbit Erlangga.

Ogata, K. 2002. *Modern Control Engineering*. *New Jersey : Prentice-Hall, Inc.*

Projects, EEE 2017. Arduino Uno board – *Full Description*. https://eeeproject.com/arduino-uno-board/. Diakses 3 Maret 2018.

Sirajudin 2013. Rancang bangun Robot terbang *Quadcopter* berbasis Mikrokontroler ATMega16. Jurusan Teknik Elektro. Universitas Tanjung Putra. Kalimantan Barat.

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA