

# **RANCANG BANGUN PLATFORM DAN SISTEM KONTROL PADA SOLAR CELL UAV ( *UNMANNED AERIAL VEHICLE* )**

## **TUGAS AKHIR**



**TEGUH SEPTI ASWORO  
SERKA KOM NRP 114919**

**PROGRAM STUDI D-3 TEKNIK ELEKTRONIKA  
DIREKTORAT PEMBINAAN DIPLOMA  
SEKOLAH TINGGI TEKNOLOGI ANGKATAN LAUT**

**SURABAYA  
DESEMBER 2020**

**RANCANG BANGUN PLATFORM DAN SISTEM KONTROL  
PADA SOLAR CELL UAV ( *UNMANNED AERIAL VEHICLE* )**

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Program Studi D-3 Teknik Elektronika  
STTAL**

**TEGUH SEPTI ASWORO  
SERKA KOM NRP 114919**

**PROGRAM STUDI D-3 TEKNIK ELEKTRONIKA  
DIREKTORAT PEMBINAAN DIPLOMA  
SEKOLAH TINGGI TEKNOLOGI ANGKATAN LAUT**

**SURABAYA  
DESEMBER 2020**

### LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : Rancang Bangun *Platform* dan Sistem Kontrol  
pada *Solar Cell* UAV (*Unmanned Aerial Vehicle*)  
Nama Mahasiswa : Teguh Septi Asworo  
Pangkat/Korps/NRP : Serka Kom NRP 114919  
Program Studi : D-3 Teknik Elektronika XIII

Tugas Akhir dengan judul dan atas nama mahasiswa tersebut diatas telah disetujui untuk dapat diujikan, sebagai bagian dari persyaratan tugas akhir untuk memperoleh gelar Ahli Madya Teknik pada Program Studi Teknik Elektronika, Direktorat Pembinaan Diploma Sekolah Tinggi Teknologi Angkatan Laut.

### DOSEN PEMBIMBING

Pembimbing I : Dr. Rachmad S., S.T., M.T. (.....)  
Lektor III/d  
NIP 196905291995121001  
  
Pembimbing II : Endin Tri Hartanto, S.T., M.T. (.....)  
Mayor Laut (E) NRP 15625/P  
  
Ditetapkan di : Surabaya  
Tanggal : Desember 2020

Mengetahui  
Direktorat Pembinaan Diploma  
Kaprodik Teknik Elektronika

Dr. Ahmadi, S.Si., M.T.  
Kolonel Laut (KH) NRP 10053/P

**LEMBAR PENGESAHAN UJIAN TUGAS AKHIR**

Judul Tugas Akhir : Rancang Bangun *Platform* dan Sistem Kontrol  
pada *Solar Cell* UAV (*Unmanned Aerial Vehicle*)

Nama Mahasiswa : Teguh Septi Asworo

Pangkat/Korps/NRP : Serka Kom NRP 114919

Program Studi : D-3 Teknik Elektronika XIII

Telah/Tidak\*) berhasil dipertahankan dihadapan Dewan Penguji  
dan Belum/Sudah\*) diterima sebagai bagian persyaratan yang diperlukan  
untuk memperoleh gelar Ahli Madya dalam Program Studi Teknik  
Elektronika, Direktorat Pembinaan Diploma Sekolah Tinggi Teknologi  
Angkatan Laut.

**DEWAN PENGUJI**

Ketua : Dr. Rachmad S., S.T., M.T. (.....)

Lektor III/d NIP 196905291995121001

Anggota : Bagiyo Herwono, S. T., M.T. (.....)

Ass Ahli III/A 9818087

Anggota : Sunarta, S.T., M.T. (.....)

Lekol Laut (E) NRP 12898/P

Anggota : Erpan Sahiri, S.T., M.T. M.Tr Hanla (.....)

Lekol Laut (E) NRP 13481/P

Anggota : Bagus Irawan, S.T., M.M.T. (.....)

Mayor Laut (E) NRP 15618/P

Ditetapkan di : Surabaya

Tanggal : Desember 2020

Mengetahui  
Direktur Pembinaan Diploma

Dr. Ahmadi, S.Si., M.T.  
Kolonel Laut (KH) NRP 10053/P

\*) Coret yang tidak perlu

## PERNYATAAN ORISINALITAS

Yang bertanda tangan dibawah ini :

Judul Tugas Akhir : Rancang Bangun *Platform* dan Sistem Kontrol  
pada *Solar Cell* UAV (*Unmanned Aerial Vehicle*)

Nama Mahasiswa : Teguh Septi Asworo

Pangkat/Korps/NRP : Serka Kom NRP 114919

Program Studi : D-3 Teknik Elektronika XIII

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya atau bagian karya yang pernah diajukan untuk memperoleh gelar Ahli Madya jenjang apapun disuatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat istilah, frasa, kalimat, paragraf, sub bab, atau bab dari karya yang pernah ditulis atau diterbitkan, kecuali yang secara tertulis dirujuk dalam naskah ini dan disebutkan dalam Daftar Referensi.

Apabila di kemudian hari terbukti bahwa terdapat plagiat dalam Tugas Akhir ini, saya bersedia menerima sanksi sesuai ketentuan peraturan dan undang-undang yang berlaku.

Surabaya, Desember 2020

Penulis

## **SURAT PENGALIHAN HAK CIPTA**

Yang bertanda tangan dibawah ini :

Nama : Teguh Septi Asworo

Alamat : Dsn. Kalangan RT 002/RW002 Ds. Wonosari Kec. Ngombol  
Kab. Purworejo Jawa Tengah

Adalah Pihak I selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada :

Nama : Sekolah Tinggi Teknologi Angkatan Laut

Alamat : Jl. Bumimoro Moro Krembangan Surabaya

Adalah Pihak II selaku Pemegang Hak Cipta berupa Tugas Akhir dengan Judul “Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell* UAV (*Unmanned Aerial Vehicle*)” untuk didaftarkan di Direktorat Hak Cipta, Desain Industri, Desain Tata Letak dan Sirkuit Terpadu dan Rahasia Dagang, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum dan Hak Azasi Manusia R.I.

Demikianlah surat pengalihan hak ini kami buat, agar dapat dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta

Surabaya, Desember 2020  
Pencipta

Dr. Ahmadi, S.Si., M.T.  
Kolonel Laut (KH) NRP 10053/P

Teguh Septi Asworo  
Serka Kom NRP 114919

© Hak Cipta Milik STTAL  
Dilindungi Undang-Undang

*“Dilarang mengutip sebagian atau seluruh tesis/skripsi/tugas akhir ini tanpa mencantumkan atau menyebutkan sumbernya. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik, atau tinjauan suatu masalah; dan pengutipan tersebut tidak merugikan kepentingan STTAL”*

*Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apa pun tanpa izin STTAL*

## KATA PENGANTAR

*Assalamu'alaikum Wr. Wb.*

Dengan memanjatkan puji syukur kehadiran Tuhan Yang Maha Kuasa, yang telah mengkaruniakan kekuatan dan kesehatan kepada penulis sehingga dapat menyelesaikan Tugas Akhir ini tepat pada waktunya dengan judul “Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell* UAV (*Unmanned Aerial Vehicle*)”.

Tugas Akhir ini merupakan salah satu persyaratan untuk menyelesaikan pendidikan bagi Bintara Siswa Prodi Diploma 3 Teknik Elektronika Angkatan XIII TA 2020, menyadari dengan sepenuhnya bahwa Tugas Akhir ini masih jauh dari kesempurnaan baik mengenai isi, pembahasan maupun penyusunannya, hal ini tentu disebabkan oleh keterbatasan pada diri penulis, oleh sebab itu penulis sangat mengharapkan adanya koreksi dan saran yang konstruktif dari berbagai pihak guna penyempurnaan.

Penulis menyadari bahwa penulisan Tugas Akhir ini tidak dapat terselesaikan dengan baik dan tepat waktu, tanpa adanya bantuan dari pihak lain, oleh sebab itu penulis mengucapkan terima kasih kepada :

1. Komandan STTAL serta seluruh Staf STTAL atas kesempatan yang telah diberikan kepada penulis untuk dapat mengikuti pendidikan Diploma III di Prodi Teknik Elektronika STTAL.
2. Direktur Pembinaan Diploma, Kaprodi Diploma 3 Teknik Elektronika, Dosen dan seluruh Staf STTAL yang telah membaktikan segenap tenaga dan pemikirannya dalam mendukung proses pembelajaran.
3. Bapak Dr. Rachmad S., S.T., M.T. dan Mayor Laut (E) Endin Tri Hartanto, S.T., M.T. selaku Dosen Pembimbing yang telah memberikan waktu, arahan, bimbingan dan bantuan selama pembuatan Tugas Akhir ini.
4. Seluruh rekan-rekan Diploma 3 Teknik Elektronika XIII STTAL atas kebersamaan dalam suka maupun duka, kerjasama, dukungan dan bantuannya selama ini.



5. Istri tercinta Dian Savitri dan kedua putri kami terkasih Thalita Sakhi Asworo dan Freya Farensa Asworo yang senantiasa mendampingi, menemani, mendukung serta memberikan semangat kepada kami selama penyusunan Tugas Akhir ini.

6. Semua pihak yang tidak mungkin penulis sebutkan satu persatu yang telah membantu dalam penyelesaian Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih belum sempurna dan memerlukan perbaikan, oleh karena itu dengan mohon adanya kritikan, saran, dan masukan serta koreksi dari semua pihak guna penyempurnaan isi proposal ini. Akhirnya dengan segala kerendahan hati penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi pembaca dan menjadi sumbangan pemikiran yang berguna bagi TNI Angkatan Laut di masa yang akan datang.

*Wassalamu'alaikum Wr. Wb.*

Surabaya,      Desember 2020

Teguh Septi Asworo  
Serka Kom NRP 114919

## ABSTRAK

*Unmanned Aerial Vehicle (UAV)*. Penelitian yang dilakukan untuk mendukung pengawasan wilayah udara dalam kaitannya dengan konsep survei area dengan sistem penerbangan mandiri. Pengawasan wilayah udara NKRI merupakan tugas pengawasan yang sangat penting. Kegiatan pengawasan wilayah berbahaya, wilayah yang sulit dijangkau dan kegiatan pemetaan merupakan tugas pesawat tanpa awak atau *Unmanned Aerial Vehicle (UAV)*. UAV bertenaga surya mampu terbang secara terus menerus selama cahaya matahari mencukupi. Untuk mendapatkan energi yang cukup untuk sistem propulsi dan kontrol elektronik diperlukan ukuran sel surya yang besar dan berat. Pada perancangan UAV Solar Cell penulis melakukan perancangan bodi UAV dan sistem kendali. Hasil pengujian yang dilakukan diperoleh kestabilan terbang UAV dan dapat dikendalikan dengan baik. UAV dapat *take off* dengan cara *hand launcher* dan *landing* dengan baik.

Kata Kunci: *UAV, Solar Cell, Hand Launcher*

## **ABSTRACT**

*Unmanned Aerial Vehicle (UAV). Research conducted to support the surveillance of airspace in its environment with a survey concept with an independent flight system. Supervision of the NKRI airspace is a very important supervisory task. Monitoring activities for dangerous areas, areas that are difficult to reach and mapping are the duties of an Unmanned Aerial Vehicle (UAV). Solar powered UAVs are capable of continuous flight as long as there is sufficient sunlight. To obtain sufficient energy for the propulsion system and electronic control requires a large and heavy solar cell. In designing a UAV Solar Cell, the authors design the UAV body and control system. The results obtained were stability, UAV flight and can be controlled well. The UAV can take off by hand launching and landing well.*

*Keywords: UAV, NKRI, Solar powered UAVs*

## DAFTAR ISI

HALAMAN JUDUL.....	ii
LEMBAR PERSETUJUAN DOSEN PEMBIMBING .....	iii
LEMBAR PENGESAHAN UJIAN TUGAS AKHIR .....	iv
PERNYATAAN ORISINALITAS .....	v
SURAT PENGALIHAN HAK CIPTA.....	vi
KATA PENGANTAR .....	viii
ABSTRAK.....	x
ABSTRACT .....	xi
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL.....	xv
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	2
1.3 Tujuan Penelitian .....	2
1.4 Manfaat Penelitian .....	3
1.5 Batasan Masalah .....	3
<b>BAB 2 TINJAUAN PUSTAKA .....</b>	<b>4</b>
2.1 Penelitian Terdahulu .....	4
2.2 Landasan Teori.....	8
2.2.1 <i>Unmanned Aerial Vehicle</i> (UAV) .....	8
2.2.2 <i>Fixed Wing</i> .....	8
2.2.3 <i>Mikrokontroller</i> .....	11
2.2.4 Modul Radio <i>Transmitter</i> dan <i>Reciever</i> .....	12
2.2.5 Motor Brushless DC (BLDC).....	12
2.2.6 ESC ( <i>Electric Speed Control</i> ) .....	14
2.2.7 <i>Propeller</i> ( Baling-baling) .....	14
2.2.8 Motor Servo.....	15
2.2.9 Baterai Lithium Polimer (Lipo) .....	17
<b>BAB 3 METODE PENELITIAN.....</b>	<b>19</b>

3.1	Desain Penelitian.....	19
3.2	Prosedur Penelitian .....	19
3.2.1	Waktu dan Tempat Penelitian.....	19
3.2.2	Alat dan Bahan Penelitian .....	19
3.2.3	Rancangan Penelitian. ....	20
3.2.4	Pengumpulan Data.....	27
3.2.5	Pengolahan Data.....	27
3.2.6	Definisi Operasional .....	27
<b>BAB 4</b>	<b>ANALISA DAN PENGUJIAN.....</b>	<b>29</b>
4.1	Implementasi .....	31
4.1.1	Implementasi Perangkat Remote Control .....	31
4.1.2	Implementasi Perangkat Proses.....	31
4.2	Pengujian.....	32
4.2.1	Pengujian Perangkat Keras .....	32
4.2.2	Pengujian Modul MPU9250 .....	32
4.2.3	Pengujian ESC Dan Motor <i>Brushless</i> .....	34
4.2.4	Perhitungan Aerodinamika Pesawat UAV .....	43
4.2.5	Pengujian Terbang UAV <i>Solar Cell</i> .....	46
<b>BAB 5</b>	<b>KESIMPULAN DAN SARAN .....</b>	<b>48</b>
5.1	Kesimpulan.....	48
5.2	Saran.....	48
	<b>DAFTAR PUSTAKA.....</b>	<b>49</b>
	<b>LAMPIRAN.....</b>	<b>51</b>

## DAFTAR GAMBAR

Gambar 2.1	(a) <i>Constant chord</i> (b) <i>Box wing</i> .....	9
Gambar 2.2	(a) <i>Tapered</i> (b) <i>Reverse tapered</i> (c) <i>Compound tapered</i> ..	10
Gambar 2.3	(a) <i>Tailless delta</i> (b) <i>Tailed delta</i> (c) <i>Cropped delta</i> (d) <i>Compound delta</i> (e) <i>Ogival delta</i> .....	10
Gambar 2.4	<i>Gelombang Transmitter</i> .....	12
Gambar 2.5	<i>Motor Brushless</i> dengan <i>Hall Sensor</i> .....	13
Gambar 2.6	<i>Elektronik Speed Controller</i> .....	14
Gambar 2.7	<i>Propeller</i> (baling - baling) .....	15
Gambar 2.8	<i>Pulsa Kendali Motor Servo</i> .....	16
Gambar 3.1	<i>Flowchart Alur Penelitian</i> .....	21
Gambar 3.2	<i>Model Skyhunter</i> .....	22
Gambar 3.3	<i>Bodi UAV Solar Cell</i> .....	23
Gambar 3.4	<i>Diagram blok Flight Controller</i> .....	24
Gambar 3.5	<i>Diagram blok Remote Control</i> .....	25
Gambar 3.6	<i>Servo Tower Pro</i> .....	26
Gambar 4.1	<i>Diagram Blok Remote Control</i> .....	30
Gambar 4.2	<i>Fungsi Kerja UAV</i> .....	30
Gambar 4.3	<i>Diagram Perkabelan MPU9250 Dengan Arduino</i> <i>Mega 2560</i> .....	33
Gambar 4.4	<i>Pengujian Sensor MPU 9250</i> .....	34
Gambar 4.5	<i>Diagram Blok Proses Konversi</i> .....	35
Gambar 4.6	<i>Pengujian thrust motor</i> .....	36
Gambar 4.7	<i>Pengukuran Daya Dan Thrust Saat Throttle 0%</i> . ....	38
Gambar 4.8	<i>Grafik Perbandingan Daya Dengan Thrust Motor</i> .....	43
Gambar 4.9	<i>Uji Manuver Pesawat UAV Solar Cell</i> .....	47
Gambar 4.10	<i>Pengujian Terbang UAV Solar Cell Searah Arah Angin</i> ....	47
Gambar 4.11	<i>Pengujian Terbang Pesawat UAV Solar Cell</i> .....	47

## DAFTAR TABEL

Tabel 2.1 Penelitian terdahulu .....	6
Tabel 4.1 Tabel Hasil Pengujian <i>Thrust</i> Motor .....	42

## **BAB 1 PENDAHULUAN**

### **1.1 Latar Belakang**

Pengembangan Alutsista dan teknologi militer di TNI-AL giat dilaksanakan dalam rangka memperkuat dan meningkatkan profesionalisme. Hal ini berguna untuk mendukung tugas utama TNI-AL dalam menjaga dan mempertahankan kedaulatan dan kesatuan negara kesatuan Negara Kesatuan Republik Indonesia (NKRI). Beberapa langkah telah dilakukan dalam pengembangan Alutsista dan teknologi militer di kubu TNI-AL dalam bentuk penelitian-penelitian dan pengembangan alusista, salah satunya adalah melakukan penelitian mengenai wahana tanpa awak atau UAV (*Unmanned Aerial Vehicle*). Penelitian yang dilakukan untuk mendukung pengawasan wilayah udara dalam kaitannya dengan konsep survei area dengan sistem penerbangan mandiri.

Pengawasan wilayah perbatasan NKRI yang banyak memiliki area perbatasan perairan merupakan salah satu permasalahan dan kendala bagi TNI-AL dalam menjalankan tugas pokoknya. Pentingnya alusista pendukung dalam menjalankan tugas penting tersebut sangat dibutuhkan personil TNI-AL saat ini. Pengawasan wilayah udara NKRI merupakan tugas pengawasan yang sangat penting. Pesawat tanpa awak atau UAV (*Unmanned Aerial Vehicle*) pada awalnya dilakukan oleh pihak militer untuk melaksanakan kegiatan operasi militer dan intelejen. Kegiatan pengawasan wilayah berbahaya, wilayah yang sulit dijangkau dan kegiatan pemetaan merupakan tugas pesawat tanpa awak atau UAV (*Unmanned Aerial Vehicle*). UAV *solar cell* bertenaga surya yang mampu terbang secara terus menerus masih menjadi impian beberapa tahun lalu, tantangan besar ini menjadi memungkinkan saat ini. Kemajuan secara signifikan diwujudkan dalam teknologi sel surya yang fleksibel dan berdaya tinggi. Konsep UAV *solar cell* dilengkapi dengan sel surya yang menutupi sebagian besar sayap UAV, sel surya akan mengambil energi dari matahari untuk memasok daya ke sistem propulsi, kontrol elektronik, dan pengisian daya baterai. Namun,



upaya disiplin ilmu yang tepat diperlukan untuk mengoptimalkan dan mengintegrasikan konsep agar sistem yang berfungsi penuh. Sebagai pokok permasalahan utama adalah kombinasi ukuran dan bobot sel surya dengan ukuran UAV. Untuk mendapatkan energi yang cukup untuk sistem propulsi dan kontrol elektronik diperlukan ukuran sel surya yang besar dan berat.

Dalam mewujudkan UAV *solar cell* yang mampu terbang secara *glider* dan di dukung sumber daya menggunakan sel surya, penulis mencoba mengangkat tema rancang bangun UAV dengan sebuah tema penelitian yang berjudul Rancang Bangun *Platform* dan Sistem Kontrol pada UAV *Solar Cell* dikarenakan UAV *solar cell* sangat membutuhkan sebuah *platform* dan sistem kontrol yang berbeda dengan UAV pada umumnya.

## **1.2 Perumusan Masalah**

Pada penelitian ini perlu dirumuskan permasalahan-permasalahan dalam membangun UAV *solar cell*. Adapun beberapa permasalahan tersebut adalah :

- a. Menentukan jenis *platform* untuk UAV *solar cell*?
- b. Bagaimana rancang bangun sistem kontrol pada UAV *solar cell*?

## **1.3 Tujuan Penelitian**

Adapun tujuan dari Rancang bangun *Platform* dan sistem kontrol pada UAV *solar cell* ini adalah :

- a. Menentukan jenis *platform* yang tepat untuk UAV *solar cell*.
- b. Merancang dan membangun sistem kontrol pada UAV *solar cell*.

#### **1.4 Manfaat Penelitian**

Adapun manfaat dari pembuatan Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell UAV (Unmanned Aerial Vehicle)* adalah sebagai berikut :

- a. Bagi TNI AL dapat digunakan sebagai wahana udara yang mampu terbang dalam durasi yang lebih lama karena didukung sumber tenaga sel surya sehingga dapat digunakan untuk kegiatan pengintaian atau pun survei area di daerah perbatasan.
- b. Bagi STTAL yaitu sebagai penerapan sistem komunikasi yang berkaitan dengan materi pendidikan pada sistem telekomunikasi, sebagai penerapan materi pemrograman *Arduino Mega 2560* dan diharapkan rancang bangun ini menjadi produk unggulan STTAL pada pameran industri pertahanan dalam negeri.
- c. Bagi diri sendiri dapat Memberikan ilmu pengetahuan tentang UAV dan fungsi komunikasi data untuk kepentingan tugas kedepan.

#### **1.5 Batasan Masalah**

Adapun batasan masalah Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell UAV (Unmanned Aerial Vehicle)* adalah :

- a. Untuk kestabilan *Platform UAV solar cell* terbatas dengan kecepatan angin max 15 knot.
- b. Keterbatasan jarak jangkauan kendali (*range*) max 500 meter.

## BAB 2 TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu referensi bagi penulis untuk melakukan perancangan penelitian Rancang Bangun *Platform* Dan Sistem Kontrol Pada *Solar Cell UAV (Unmanned Aerial Vehicle)* yang penulis kumpulkan merupakan jurnal-jurnal internasional yang telah dipublikasikan secara luas dan penulis tidak menemukan kesamaan sistem yang akan dibuat pada penelitian ini. Penelitian-penelitian terdahulu merupakan referensi-referensi yang telah diakui secara internasional yang sangat bermanfaat dalam memperkaya pengetahuan bagi penulis. Penelitian-penelitian terdahulu yang dikumpulkan merupakan penelitian yang terkait dengan penelitian akan penulis lakukan.

Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan penelitian yang akan dilakukan penulis.

a. Rancang Bangun Wahana Pesawat Tanpa Awak (*Fixed Wing*) Berbasis Ardupilot

Hardy Samuel Saroinsong, Vecky C. Poekoel, Pinrolinvic D.K Manembu (2018), Teknik Elektro, Universitas Sam Ratulangi Manado, melakukan penelitian UAV (*Unmanned Aerial Vehicle*) merupakan istilah yang digunakan untuk merepresentasikan benda terbang dengan suplay daya sendiri yang bisa dikendalikan dari jarak jauh menggunakan *remote control* dari luar pesawat atau dapat bergerak secara otomatis berdasarkan program yang sudah ditanamkan pada sistem komputernya. Sistem ini menggunakan kontrol manual dan *autopilot*, Pada *mode* manual pengguna secara manual mengendalikan pergerakan pesawat melalui *radio kontroller* sedangkan pada *mode autopilot* pesawat dikendalikan oleh *mikrokontroler Arduino Mega 2.8* yang mengolah data-data sensor IMU (*Inertial Measurement Unit*) yang didalamnya terdapat *gyroscope* dan *accelerometer*, GPS dan *barometric altimeter*

sehingga dapat terbang secara otomatis dengan sesuai *waypoint* GPS yang dimasukkan. Perancangan sistem kontrol UAV ini menggunakan kontrol *ardupilot* dengan aplikasi *mission planner*, dengan tujuan mengontrol dan merancang jalur penerbangan pada pesawat sehingga pesawat tersebut dapat terbang dengan *Mode Autopilot (mode otomatis)* diaktifkan secara manual dari *remote control Fsi6*.

b. *Design Analysis of Solar Powered Unmanned Aerial Vehicle*

Karthik Reddy Buchireddy Sri, Poondla Aneesh, Kiran Bhanu, M Natarajan (2016), VIT University – School of Mechanical Engineering Mechanical Department – Vellore/Tamil Nadu – India, mengemukakan penelitian dalam UAV adalah daya tahan atau waktu penerbangan karena pesawat domain umum menggunakan bahan bakar konvensional, yang merupakan polutan, memiliki umur yang terbatas dan mahal. Penelitian ini bertujuan untuk mendorong tentang sumber energi terbarukan untuk penerbangan mengingat tantangan dasar untuk pesawat bertenaga surya yaitu wilayah operasi geografis, pengumpulan dan penyimpanan energi, muatan dan parameter desain. Oleh karena itu, sebuah pesawat dirancang untuk beban 2 kg, termasuk muatan, dan dianalisa dalam berbagai aspek. Selain itu, desain dioptimalkan mulai dari *airfoil* hingga struktur lengkap untuk kinerja yang lebih baik.

c. *Structural Design and Difficulties of Solar UAV*

W W Zhang, L G Zhang, Z W Yan, L Wang (2019), School of Aerospace Engineering, Shenyang Aerospace University, Shenyang, China, mengemukakan penelitian desain pesawat terbang ketinggian tinggi dan tahan lama jelas berbeda dari desain pesawat konvensional. Sistem energi dan kondisi operasinya yang unik membuatnya perlu mempertimbangkan dengan cermat rincian teknis utama dalam desain. Struktur memiliki pengaruh besar pada kinerja daya tahan pesawat surya. Berat struktur dapat diperkirakan

secara kasar dengan proporsi sistem energi udara terhadap total berat pesawat, sehingga dapat mengoptimalkan struktur. Melalui analisis beberapa jenis Pesawat surya, dapat disimpulkan bahwa tata letak aerodinamis dan desain struktur memiliki dampak besar pada waktu penerbangan dan kinerja pesawat surya. Akhirnya, tantangan yang dihadapi oleh pesawat surya disajikan, yang dapat memberikan referensi untuk desain pesawat surya di masa depan dan membantu.

Tabel 2.1 Penelitian terdahulu

No	Nama Peneliti	Judul Penelitian	Hasil Penelitian
1	Hardy Samuel Saroinsong, Vecky C. Poekoel, Pinrolinvic D.K Manembu (2018)	Rancang Bangun Wahana Pesawat Tanpa Awak ( <i>Fixed Wing</i> ) Berbasis <i>Ardupilot</i>	Perancangan sistem kontrol UAV ini menggunakan kontrol <i>ardupilot</i> dengan aplikasi <i>mission planner</i> , dengan tujuan mengontrol dan merancang jalur penerbangan pada pesawat sehingga pesawat tersebut dapat terbang dengan <i>Mode Autopilot (mode otomatis)</i> diaktifkan secara manual dari <i>remote control Fsi6</i> .
Sumber: <a href="https://ejournal.unsrat.ac.id/index.php/elekdankom/article/download/19195/18753">https://ejournal.unsrat.ac.id/index.php/elekdankom/article/download/19195/18753</a>			

Lanjutan Tabel 2.1

No	Nama Peneliti	Judul Penelitian	Hasil Penelitian
2	Karthik Reddy Buchireddy Sri, Poondla Aneesh, Kiran Bhanu, M Natarajan (2016)	<i>Design Analysis of Solar- Powered Unmanned Aerial Vehicle</i>	sebuah pesawat dirancang untuk beban 2 kg, termasuk muatan, dan dianalisa dalam berbagai aspek. Selain itu, desain dioptimalkan mulai dari <i>airfoil</i> hingga struktur lengkap untuk kinerja yang lebih baik
Sumber : <a href="http://www.jatm.com.br/ojs/index.php/jatm/article/view/666">http://www.jatm.com.br/ojs/index.php/jatm/article/view/666</a>			
3	W W Zhang, L G Zhang, Z W Yan, L Wang (2019)	<i>Structural Design and Difficulties of Solar UAV</i>	Struktur memiliki pengaruh besar pada kinerja daya tahan pesawat surya. Berat struktur dapat diperkirakan secara kasar dengan proporsi sistem energi udara terhadap total berat pesawat, sehingga dapat mengoptimalkan struktur.
Sumber: <a href="https://iopscience.iop.org/article/10.1088/1757-899X/608/1/012016/meta">https://iopscience.iop.org/article/10.1088/1757-899X/608/1/012016/meta</a>			

## **2.2 Landasan Teori**

Memberikan informasi mengenai hubungan dari masalah yang melatarbelakangi penelitian ini dengan sumber-sumber kepustakaan sebagai dasar penelitian, mencakup jurnal maupun artikel yang relevan.

### **2.2.1 *Unmanned Aerial Vehicle (UAV)***

*Unmanned Aerial Vehicle (UAV)* atau Pesawat Tanpa Awak adalah sebuah mesin terbang yang berfungsi dengan kendali jarak jauh oleh pilot atau mampu mengendalikan dirinya sendiri, menggunakan hukum aerodinamika untuk mengangkat dirinya, bisa digunakan kembali dan mampu membawa muatan baik senjata maupun muatan lainnya. Penggunaan terbesar dari pesawat tanpa awak ini adalah di bidang militer. Rudal walaupun mempunyai kesamaan tapi tetap dianggap berbeda dengan pesawat tanpa awak karena rudal tidak bisa digunakan kembali dan rudal adalah senjata itu sendiri (Rohman, 2012).

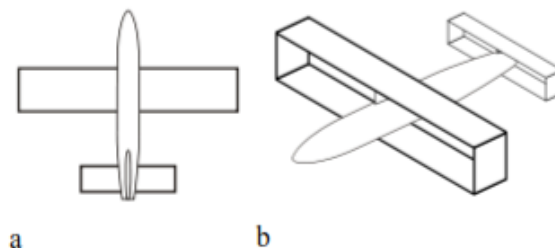
Ada berbagai bentuk UAV, ukuran, konfigurasi dan karakteristik. Secara historis UAV *drone* pada awalnya dibuat dengan sangat sederhana dan akhirnya dikontrol secara otonom (*autonomous*) semakin banyak produksi dan digunakan. Saat ini, UAV telah digunakan untuk melakukan misi intelijen, pemantauan (*surveillance*), pengintaian (*reconnaissance*) serta misi serangan (*attack*). Banyak dilaporkan bahwa UAV telah berhasil dengan tingkat akurasi tinggi dalam melakukan misi intelijen, pemantauan, pengintaian dan serangan dengan menggunakan roket, rudal dan bom. UAV sering lebih disukai untuk misi yang terlalu “membosankan dan berbahaya/beresiko tinggi” bagi pesawat berawak. (Pramadi, 2010). Ada banyak tipe dari UAV, diantara adalah *fixed wing* dan *multi rotor*. Dalam penelitian ini tipe pesawat yang digunakan adalah jenis *fixed wing*.

### **2.2.2 *Fixed Wing***

UAV tipe *fixed wing* atau sayap tetap mempunyai sayap yang kokoh dan memiliki *airfoil* yang ditentukan sehingga mampu mengangkat pesawat maju dengan dorongan dari kecepatan pesawat UAV. Daya dorong ini dihasilkan oleh mesin pembakaran internal atau motor listrik. Kontrol dari

UAV berasal dari papan kontrol yang tertanam dalam pesawat yang biasanya terdiri dari *aileron* sebagai pengangkat pesawat dan kemudi pesawat. Bagian itu menjadikan UAV dapat terbang bebas dan berputar di tiga sumbu yang tegak lurus sama lain dan berpotongan dipusat gravitasi UAV itu sendiri. Kelebihan utama dari *fixed wing* adalah terbuat dari struktur yang lebih sederhana dari *rotary wing*, sehingga perawatan dan perbaikannya tidak begitu sulit serta biaya yang dikeluarkan tidak terlalu besar. Struktur yang sederhana ini memastikan pesawat lebih aerodinamis sehingga pesawat dapat terbang lebih lama dengan kecepatan tinggi dan memetakan *area survey* yang luas dalam sekali penerbangan. (Anonim, 2016) Selain dari pada kedudukan sayap terhadap badan pesawat, maka pesawat terbang dapat juga ditinjau dari bentuk sayap, bentuk sayap sangat erat hubungannya dengan sifat-sifat aerodinamis dari pesawat terbang yang bersangkutan, termasuk kemampuan (*performance*) pesawat terbang tersebut.

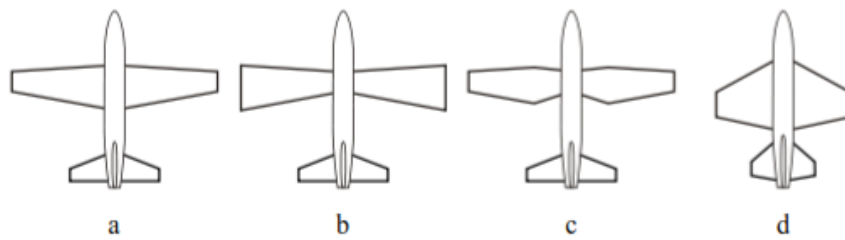
Bentuk sayap yang paling banyak kita kenal adalah: Sayap lurus (*Straight wing/rectangular wing*) yang terdapat pada Gambar 2.1 yang banyak digunakan pada pesawat pada awal industri penerbangan.



Gambar 2.1 (a) *Constant chord* (b) *Box wing*

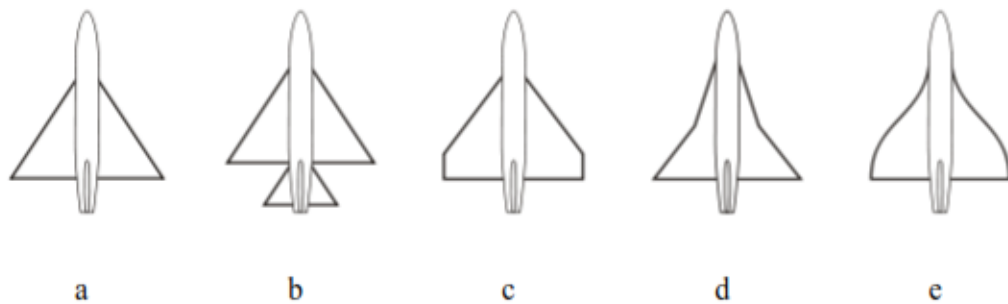
Sayap tirus (*Tapered wing*) pada Gambar 2.2 merupakan sayap yang tirus, didesain demikian untuk mengoptimalkan fungsi sayap pesawat. Variasinya dapat berupa *Tapered*, *Reverse tapered*, *Compound tapered* ataupun *Trapezoidal*.





Gambar 2.2 (a) *Tapered* (b) *Reverse tapered* (c) *Compound tapered*  
(d) *Trapezoidal*

Gambar 2.3 merupakan sayap segi tiga (*delta wing*) menggunakan sayap berbentuk segitiga melebar kebelakang. Beberapa variasi diantaranya:



Gambar 2.3 (a) *Tailless delta* (b) *Tailed delta* (c) *Cropped delta* (d)  
*Compound delta* (e) *Ogival delta*

Keuntungan dari konfigurasi Sayap tetap tersebut adalah pertama, struktur yang lebih simpel dibandingkan pada konfigurasi *multi rotor*. Struktur yang tidak rumit ini tentu saja memudahkan perawatan baik dari segi biaya dan waktu. Kedua, bentuk ini memiliki efisiensi aerodinamik yang efisien sehingga memungkinkan waktu terbang yang jauh lebih lama dibandingkan dengan *multi rotor*. Efisiensi aerodinamis ini juga (biasanya) membuat sebuah UAV Sayap tetap hanya membutuhkan satu atau dua buah motor sebagai daya dukung nya. Sehingga dipilih sayap tetap untuk pembuatan UAV *solar cell* ini dan walaupun hanya memiliki jumlah motor satu atau dua, tetapi sebuah UAV Sayap tetap dapat membawa beban yang lebih berat sehingga memungkinkan operator untuk membawa peralatan maupun sensor tambahan.

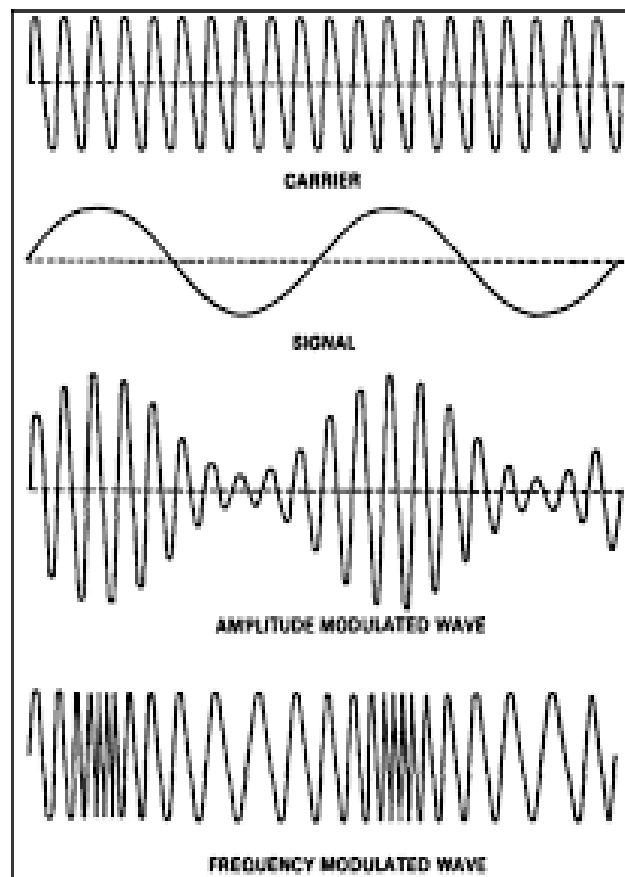
Hingga saat ini, satu-satunya kekurangan dari UAV berjenis Sayap tetap ini adalah kebutuhan untuk *take off* dan *landing* pada permukaan yang cukup luas. Selain itu, karena daya angkat yang didapat berasal dari udara yang bergerak pada permukaan-permukaan sayapnya, maka UAV Sayap tetap tidak dapat berada pada posisi konstan atau yang sering disebut dengan *hovering*. UAV tersebut harus selalu pada posisi maju kedepan untuk tetap mengudara, sehingga kurang cocok untuk aplikasi seperti inspeksi, maupun *survey* yang membutuhkan ketelitian.

### **2.2.3 Mikrokontroller**

Arduino dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*. Pertama-tama perlu dipahami bahwa kata "*platform*" disini adalah sebuah pilihan kata yang tepat. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih.

IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory microcontroller*. Ada banyak projek dan alat-alat yang dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino.

Arduino berevolusi menjadi sebuah *platform* karena ia menjadi pilihan dan acuan bagi banyak praktisi. Salah satu yang membuat arduino memikat hati banyak orang adalah karena sifatnya *open source*, baik untuk *hardware* maupun *softwrenya*. Komponen utama didalam papan Arduino adalah sebuah *microcontroller* 8 bit dengan merk Atmega yang dibuat oleh perusahaan Atmel Corporation. Berbagai papan Arduino menggunakan tipe Atmega yang berbeda-beda tergantung dari spesifikasinya, sebagai contoh Arduino Uno menggunakan Atmega328 sedangkan Arduino Mega 2560 yang lebih canggih menggunakan Atmega2560. (Djuandi : 2011, 2).



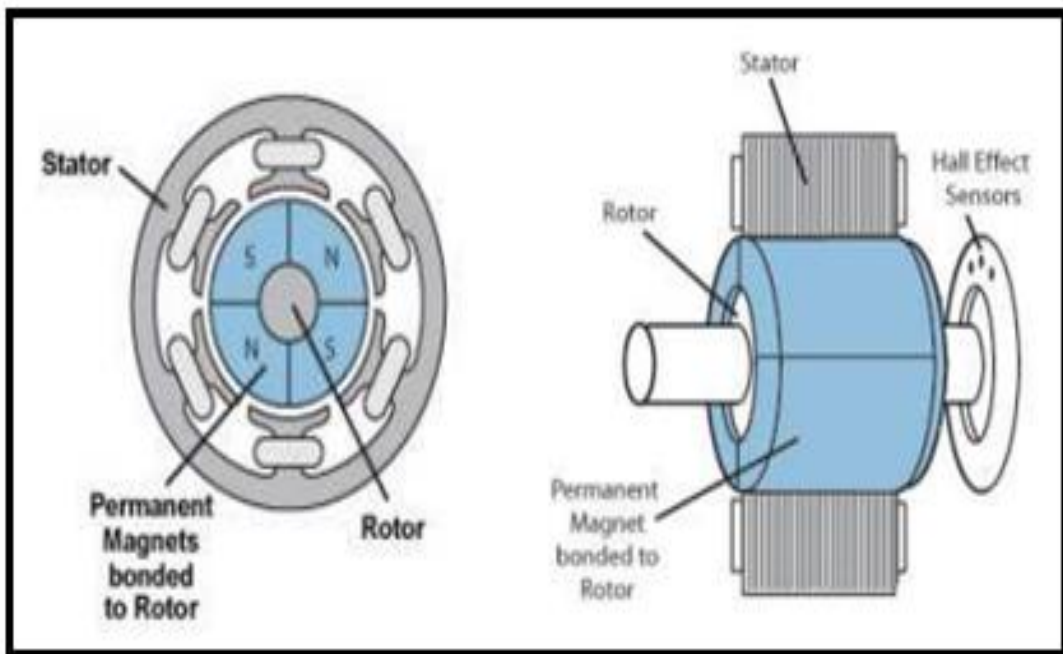
Gambar 2.4 Gelombang *Transmitter*

#### 2.2.4 Modul Radio Transmitter dan Reciever

Transmitter adalah bagian dari sistem komunikasi wireles yang berfungsi untuk mengirimkan data ke tempat lain berupa gelombang radio. Prinsip kerja dari transmitter ini adalah adanya induksi medan magnetik dari sumber potensial yang menyebabkan arus dan menginduksi rangkian lainnya. *Transmitter* yang berupa gelombang sinus ternyata tidak mengandung informasi apapun, maka diperlukan modulasi untuk menyampaikannya. Gambar 2.4 merupakan jenis dari 3 modulasi yaitu *pulse modulation*, *amplitude modulation* dan *frequency modulation*. Sedangkan *receiver* merupakan bagian yang berfungsi untuk menerima sinyal atau data yang dikirimkan *oleh transmitter*.

#### 2.2.5 Motor Brushless DC (BLDC)

Secara umum motor BLDC terdiri dari dua bagian, yakni, *rotor*, bagian yang bergerak, yang terbuat dari permanen magnet dan *stator*,

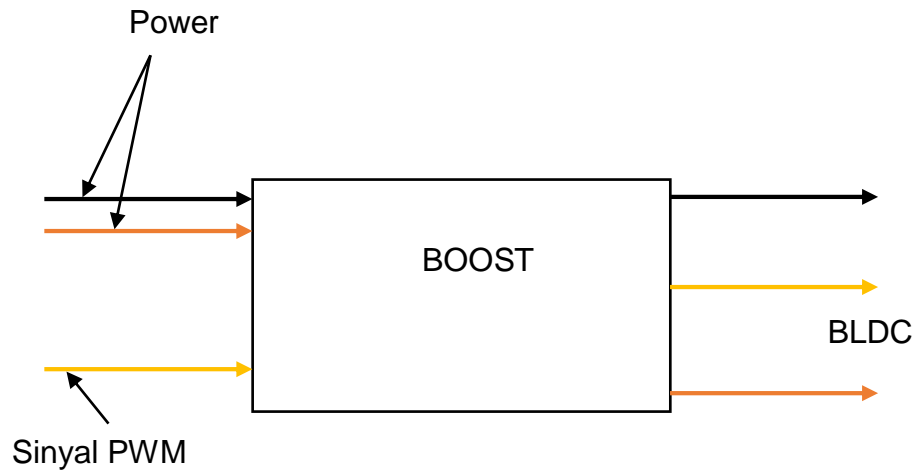


Gambar 2.5 *Motor Brushless dengan Hall Sensor*

bagian yang tidak bergerak, yang terbuat dari kumparan 3 fasa. Walaupun merupakan motor listrik *synchronous* AC 3 fasa, motor ini tetap disebut dengan BLDC karena pada implementasinya BLDC menggunakan sumber DC sebagai sumber energi utama yang kemudian diubah menjadi tegangan AC dengan menggunakan *inverter* 3 fasa. Tujuan dari pemberian tegangan AC 3 fasa pada stator BLDC adalah menciptakan medan magnet putar *stator* untuk menarik magnet *rotor*.

Beberapa keuntungan *brushless* DC motor dibandingkan dengan motor DC biasa, adalah:

- a. Lebih tahan lama, karena tidak memerlukan perawatan terhadap sikatnya.
- b. Memiliki tingkat efisiensi yang tinggi.
- c. *Torsi* awal yang tinggi.
- d. Kecepatan yang tinggi, tergantung pada kekuatan medan magnet yang dihasilkan oleh arus yang dibangkitkan dari kendali penggerakannya.



Gambar 2.6 *Elektronik Speed Controller*

### 2.2.6 ESC (Electric Speed Control)

ESC adalah rangkaian elektronik yang berfungsi sebagai pengatur kecepatan putaran motor pada pesawat atau helikopter dengan cara menterjemahkan sinyal yang diterima *receiver* dari *transmitter*.

Untuk menentukan ESC yang akan digunakan sangatlah penting untuk mengetahui kekuatan (*peak current*) dari motor. Sebaiknya ESC yang digunakan memiliki *Peak Current* yang lebih besar daripada motor. Contoh, kekuatan motor adalah 12A (*ampere*) pada saat *throttle* terbuka penuh. ESC yang akan digunakan adalah ESC yang berkekuatan 18A atau 20A. Jika di paksakan menggunakan ESC 10A kemungkinan pada saat *throttle* dibuka penuh, ESC akan panas bahkan terbakar. (Yulistiyanto, 2013).

### 2.2.7 Propeller ( Baling-baling)

*Propeller* adalah baling baling untuk menjalankan pesawat terbang. Baling-baling ini memindahkan tenaga dengan mengkonversi gerakan rotasi menjadi daya dorong untuk menggerakkan sebuah kendaraan seperti pesawat terbang, untuk melalui suatu massa seperti udara, dengan memutar dua atau lebih bilah kembar dari sebuah poros utama. Sebuah



Gambar 2.7 *Propeller* (baling - baling)

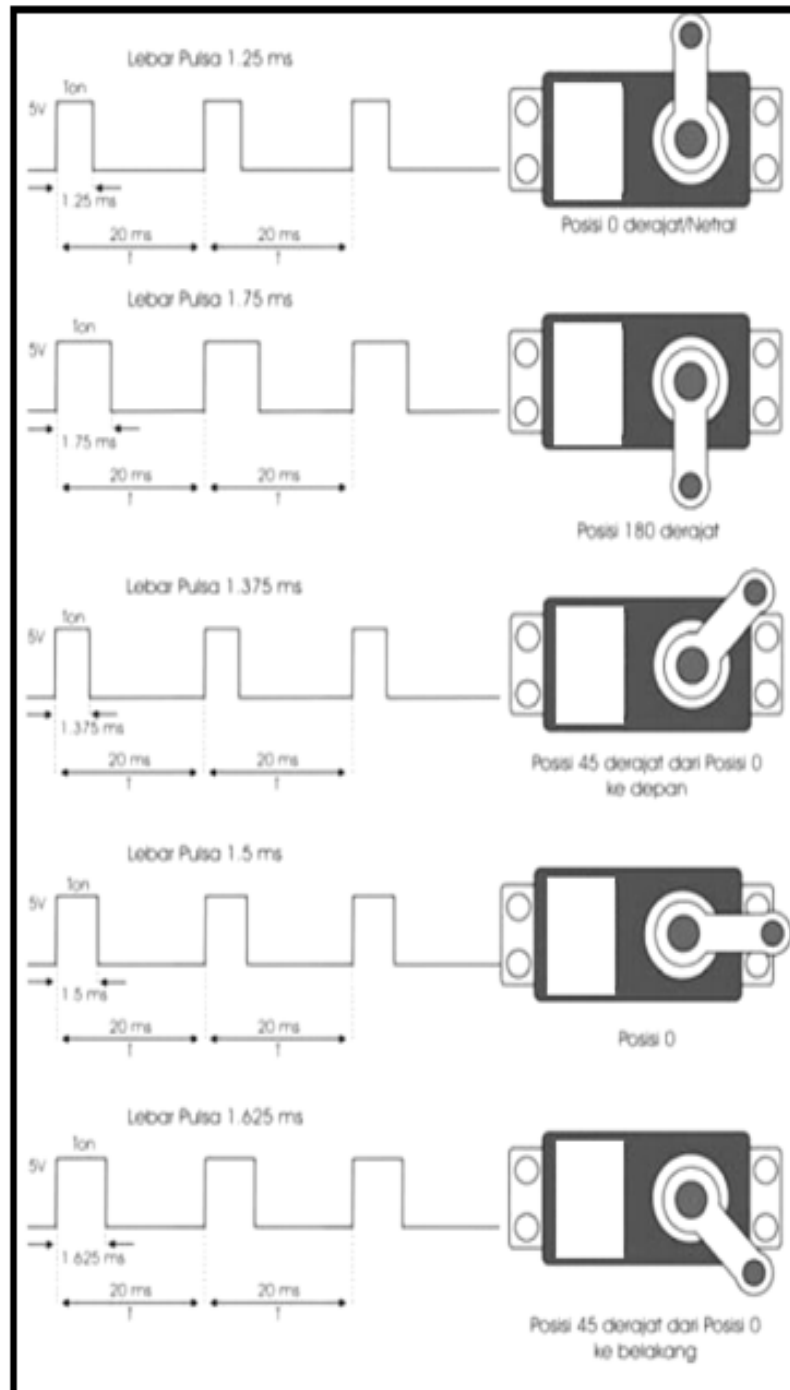
*propeller* berperan sebagai sayap berputar, menghasilkan sebuah perbedaan tekanan antara permukaan depan dan belakang.

#### 2.2.8 Motor Servo

Motor servo adalah motor yang mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dapat dikendalikan dengan memberikan variasi lebar pulsa (*duty cycle*) sinyal PWM pada bagian pin kontrolnya. Jenis Motor Servo Standar 180° hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°. Motor servo *continuous* mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

Pulsa kontrol motor servo operasional dikendalikan oleh sebuah pulsa selebar  $\pm 20$  ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari *range* sudut maksimum. Apabila motor servo diberikan pulsa dengan besar 1.5 ms mencapai gerakan 90°, maka bila kita berikan pulsa kurang dari 1.5 ms maka posisi mendekati 0° dan bila kita berikan pulsa lebih dari 1.5 ms maka posisi mendekati 180°. Pulsa kendali

motor servo akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50 Hz.



Gambar 2.8 Pulsa Kendali *Motor Servo*

### 2.2.9 Baterai *Lithium Polimer (Lipo)*

Baterai LiPo tidak menggunakan cairan sebagai elektrolit melainkan menggunakan elektrolit polimer kering yang berbentuk seperti lapisan plastik film tipis. Lapisan film ini disusun berlapis-lapis diantara anoda dan katoda yang mengakibatkan pertukaran ion. Dengan metode ini baterai LiPo dapat dibuat dalam berbagai bentuk dan ukuran. Diluar dari kelebihan arsitektur baterai LiPo, terdapat juga kekurangan yaitu lemahnya aliran pertukaran ion yang terjadi melalui elektrolit polimer kering. Hal ini menyebabkan penurunan pada *charging* dan *discharging rate*. Masalah ini sebenarnya bisa diatasi dengan memanaskan baterai sehingga menyebabkan pertukaran ion menjadi lebih cepat, namun metode ini dianggap tidak dapat untuk diaplikasikan pada keadaan sehari-hari.

#### a. Tegangan

Baterai Lipo memiliki *rating* tegangan 3,7 volt per sel. Keuntungannya adalah tegangan baterai yang tinggi dapat dicapai dengan menggunakan jumlah sel yang lebih sedikit. Pada setiap paket baterai LiPo selain tegangan ada label yang disimbolkan dengan "S". Disini "S" berarti sel yang dimiliki sebuah paket baterai (*battery pack*). Sementara bilangan yang berada didepan simbol menandakan jumlah sel dan biasanya berkisar antar 2-6S (meskipun kadang ada yang mencapai 10S). Berikut adalah beberapa contoh notasi baterai LiPo.

- 1) 3.7 volt battery = 1 cell x 3.7 volts
- 2) 7.4 volt battery = 2 cells x 3.7 volts (2S)
- 3) 11.1 volt battery = 3 cells x 3.7 volts (3S)
- 4) 14.8 volt battery = 4 cells x 3.7 volts (4S)
- 5) 18.5 volt battery = 5 cells x 3.7 volts (5S)
- 6) 22.2 volt battery = 6 cells x 3.7 volts (6S)

#### b. Kapasitas

Kapasitas baterai menunjukkan seberapa banyak energi yang dapat disimpan oleh sebuah baterai dan diindikasikan dalam *miliampere hours* (mAh). Notasi ini adalah cara lain untuk



mengatakan seberapa banyak beban yang dapat diberikan kepada sebuah baterai selama 1 jam, dimana setelah 1 jam baterai akan benar-benar habis. Sebagai contoh sebuah baterai RC LiPo yang memiliki rating 1000 mAh akan benar-benar habis apabila diberi beban sebesar 1000 *miliampere* selama 1 jam. Apabila baterai yang sama diberi beban 500 *miliampere*, maka baterai akan benar-benar habis setelah selama 2 jam. Begitu pun apabila beban ditingkatkan menjadi 15.000 *miliampere* (15 Amps) maka energi di dalam baterai akan habis terpakai setelah selama 4 menit saja. (15 Amp merupakan jumlah beban yang umum digunakan pada RC kelas 400). Seperti yang telah dijelaskan, dengan beban arus yang begitu besar maka merupakan sebuah keuntungan apabila menggunakan baterai dengan kapasitas yang lebih besar (misal 2000 mAh). Dengan begitu maka waktu *discharge* akan meningkat menjadi 8 menit.

c. *Discharge Rate*

*Discharge rate* biasa disimbolkan dengan "C" merupakan notasi yang menyatakan seberapa cepat sebuah baterai untuk dapat dikosongkan (*discharge*) secara aman. Sesuai dengan penjelasan diatas bahwa energi listrik pada baterai LiPo berasal dari pertukaran ion dari anoda ke katoda. Semakin cepat pertukaran ion yang dapat terjadi maka berarti semakin nilai dari "C". Sebuah baterai dengan *discharge rate* 10C berarti baterai tersebut dapat di *discharge* 10 kali dari kapasitas baterai sebenarnya. begitu juga 15C berarti 15 kali, dan 20C berarti 20 kali.

## **BAB 3 METODE PENELITIAN**

### **3.1 Desain Penelitian**

Desain penelitian rancang bangun *Platform* dan sistem kontrol pada *Unmanned Aerial Vehicle* (UAV) *solar cell* yaitu dengan membuat sebuah *platform* yang sesuai dengan kebutuhan penempatan *solar cell* dengan bentuk wing dan jenis pesawat yang sesuai. Dalam perancangan ini penulis menggunakan modul NRF240L1 sebagai sistem komunikasi antara unit UAV dengan *user* atau pengendali.

### **3.2 Prosedur Penelitian**

Untuk melaksanakan penelitian dengan baik, maka persiapan langkah-langkah untuk menemukan sumber data serta melakukan penelitian perangkat Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell UAV (Unmanned Aerial Vehicle)* yang mencakup instrument penelitian, data primer, data sekunder, serta pengujian keabsahan dan keterandalan data.

#### **3.2.1 Waktu dan Tempat Penelitian**

Lokasi penelitian merupakan suatu tempat atau wilayah dimana penelitian tersebut akan dilakukan. Adapun penelitian yang dilakukan oleh penulis mengambil lokasi di Kampus STTAL Moro Krembangan Surabaya. Waktu yang digunakan dalam penelitian ini mulai bulan Juli 2020 hingga akhir Desember 2020.

#### **3.2.2 Alat dan Bahan Penelitian**

Alat dan bahan yang diperlukan untuk mendukung proses penelitian ini dan penerapan sistem.

##### **a. Alat Penelitian**

Alat yang di pergunakan pada penelitian terdiri dari perangkat keras dan juga perangkat lunak.

- 1) Perangkat Keras
- 2) Unit proses utama Arduino Nano, ESC Aria 40A, *motor brushless* RC timer M5010 360KV, modul NRF24L01 2.4 Ghz 100 mW, motor servo TGS90 tower pro, *power module* dan *power sensor* dan *battery lipo* 3S 2200mAH.
- 3) Perangkat Lunak
- 4) Arduino IDE 1.8.5, merupakan editor program Arduino

b. Bahan Penelitian.

Berikut ini data Arduino IDE yang perlu diketahui sebelum proses pembuatan program mikrokontroller.

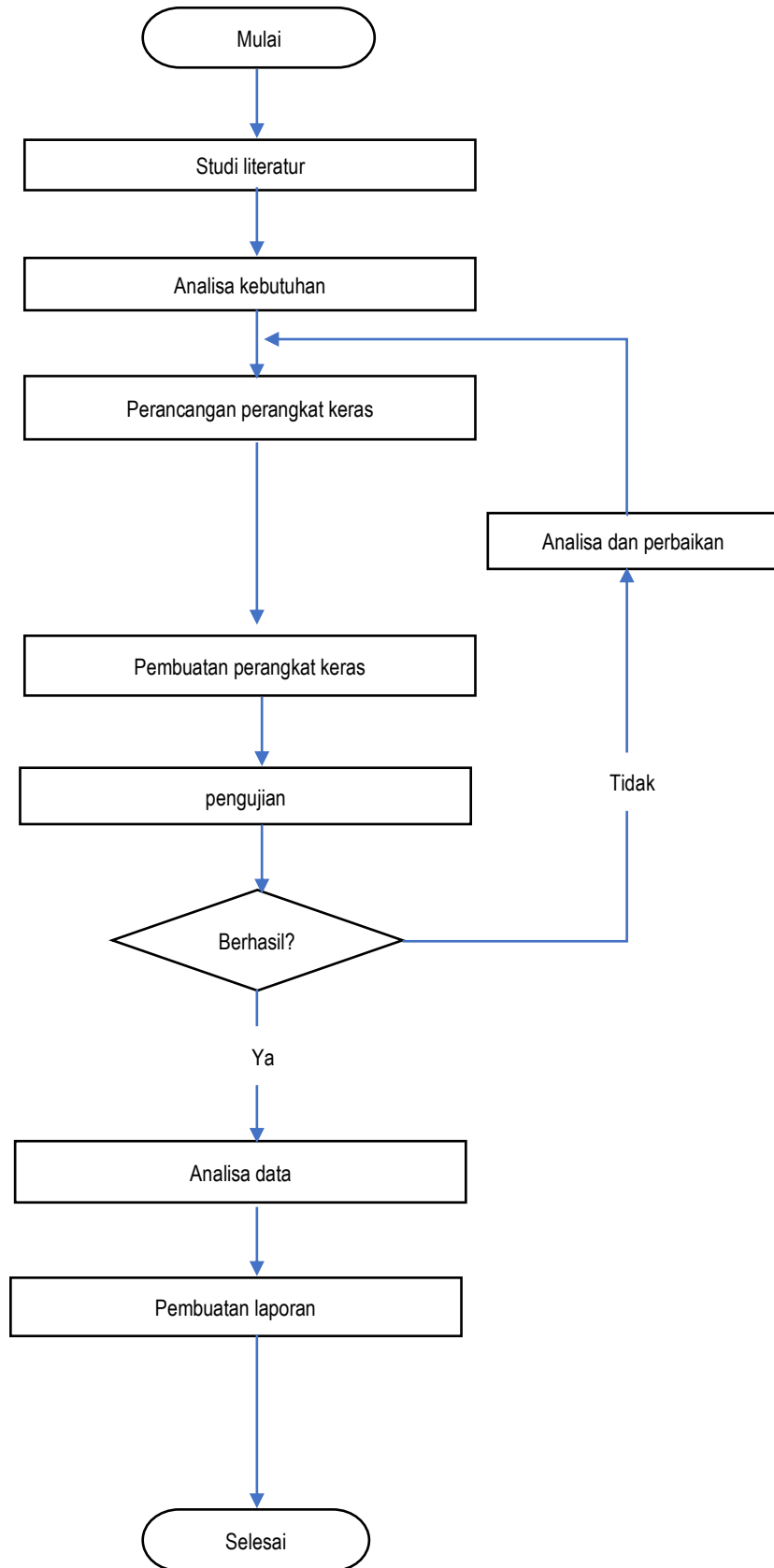
- 1) Kebutuhan minimal sistem untuk menjalankan program Arduino IDE.
- 2) Arduino IDE yang digunakan versi 1.8.5
- 3) Merupakan program sumber terbuka yang dapat di unduh secara gratis.

### 3.2.3 Rancangan Penelitian.

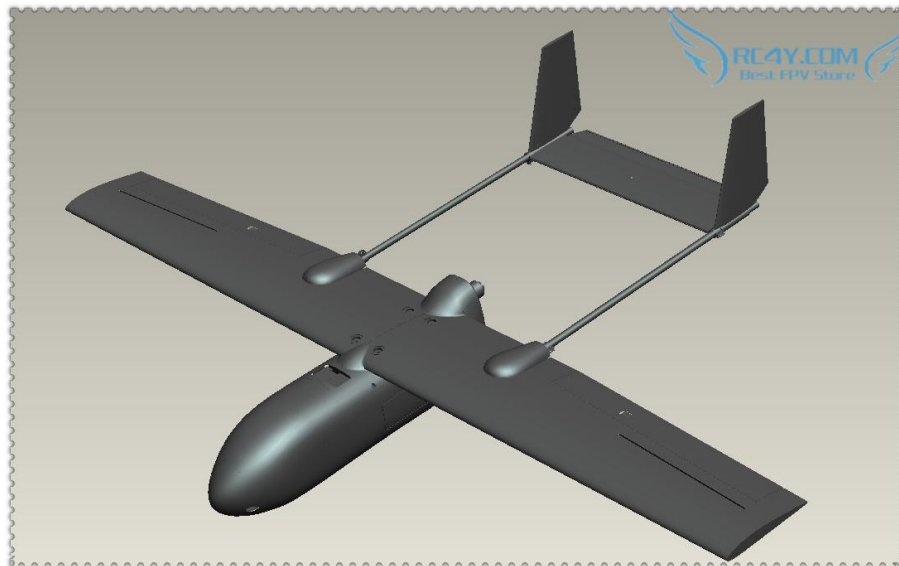
Rancang bangun *Platform* dan sistem kontrol pada *Unmanned Aerial Vehicle (UAV) solar cell* menerapkan tahap-tahap penelitian sebagai berikut:

a. Diagram Alir Penelitian

Untuk menjelaskan langkah-langkah penelitian Rancang Bangun *Platform* dan Sistem Kontrol pada *Solar Cell UAV (Unmanned Aerial Vehicle)* dapat dibuat diagram alir yang terdapat pada Gambar 3.1 agar proses atau langkah penelitian rancang bangun dapat didefinisikan dengan jelas dan tersruktur. Gambar berikut merupakan diagram alir penelitian rancang bangun *Platform* dan sistem kontrol pada *Solar Cell UAV (Unmanned Aerial Vehicle)*.



Gambar 3.1 *Flowchart* Alur Penelitian



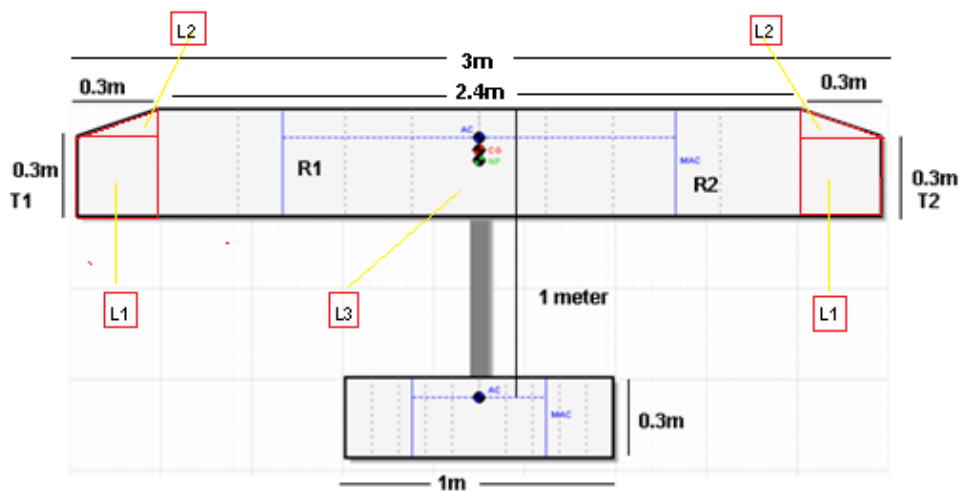
Gambar 3.2 Model Skyhunter

b. Perancangan *Platform* Pada UAV *solar cell*

Badan UAV terdiri dari satu lambung utama yang tepat pada titik tengah atau disebut fuselag dengan panjang 80cm dan lebar 40cm dengan kedua ujung meruncing atau mengerucut. Lambung ini meningkatkan kemampuan UAV melakukan manuver. Dan konstruksi lambung memungkinkan UAV untuk bermanuver dengan keseimbangan yang baik. Badan pesawat UAV dibuat dari material *hard foam* padat yang dirancang mirip dengan model pesawat UAV *skyhunter* Gambar 3.2 dengan wingspan 300 cm dan panjang fuselage dari depan hingga sampai ke *stabilizer* mencapai 160 cm dan dengan menggunakan 2 motor di bagian depan *wing*.

Tujuan utama sayap pada pesawat UAV pada umumnya untuk menghasilkan daya angkat. Daya angkat adalah kekuatan yang membuat pesawat tetap terbang. Mengubah area sayap dapat mengubah jumlah daya angkat yang dihasilkan oleh pesawat, yang dapat mempengaruhi kinerja pesawat. Bagian terpenting dari ukuran sayap adalah luas sayap ketika melihatnya dari bawah. Area ini disebut area *planform*.

Area *planform*, yang termasuk bagian bawah badan pesawat, merupakan indikasi area di mana daya angkat dihasilkan. Jika area



Gambar 3.3 Bodi UAV *Solar Cell*

*planform* yang lebih besar, maka hal tersebut merupakan area yang memiliki daya angkat lebih besar dihasilkan. Jadi bisa diperoleh lebih banyak daya angkat dengan memiliki sayap yang lebih besar, mengapa tidak memiliki sayap sebesar mungkin? Lebih banyak mengangkat bisa menjadi hal yang baik. Hal ini dapat membuat pesawat menjadi lebih berat. Tetapi peningkatan daya angkat datang pada nilai peningkatan berat dan hambatan.

Ketika sebuah pesawat terbang melaju, gaya angkat yang dihasilkannya harus sama dengan beratnya. Jika lebih rendah, pesawat akan jatuh dari langit. Jika lebih tinggi, pesawat akan naik. Ini berarti bahwa jika memiliki lebih banyak daya angkat, bobot akan semakin berat. Dengan asumsi tekanan karena berat pesawat UAV sebesar 4,1 Kg. Pada sub bab ini penulis akan menjabarkan perhitungan mengenai *center of gravity* (CG), *neutral point* (NP) dan *mean aerodynamic chord* (MAC) pesawat UAV yang telah dibuat dan di ilustrasikan pada Gambar 3.3. Berdasarkan ukuran badan UAV *Solar Cell*, dapat dihitung luas area sayap utama dengan formula perhitungan sebagai berikut :

$$Area = (L1 \times 2) + (L2 \times 2) + L3$$

$$L1 = p \times l$$

$$L1 = 0.3 \times 0.3$$

$$L1 = 0.09 \text{ m}^2$$

$$L2 = \frac{1}{2} \times p \times l$$

$$L2 = \frac{1}{2} \times 0.1 \times 0.3$$

$$L2 = 0.015 \text{ m}^2$$

$$L3 = p \times l$$

$$L3 = 2.4 \times 0.4$$

$$L3 = 0.96 \text{ m}^2$$

$$\text{Area} = (0.09 \times 2) + (0.015 \times 2) + 0.96$$

$$\text{Area} = 1.17 \text{ m}^2$$

Dalam satuan cm maka luas area sayap dapat dituliskan

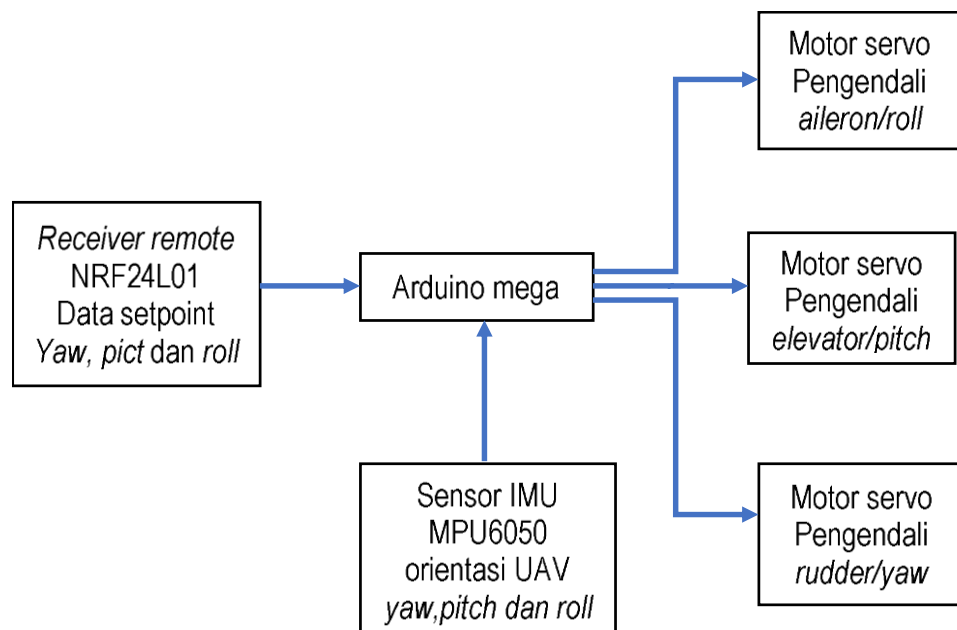
$$\text{Area} = 11700 \text{ cm}^2$$

Pada perhitungan nilai CG (*center of gravity*) penulis menentukan nilai CG adalah 35% dari nilai MAC. Nilai MAC diperoleh dari nilai *root chord* yaitu 40cm. Sehingga nilai CG dapat dihitung dengan formula sebagai berikut :

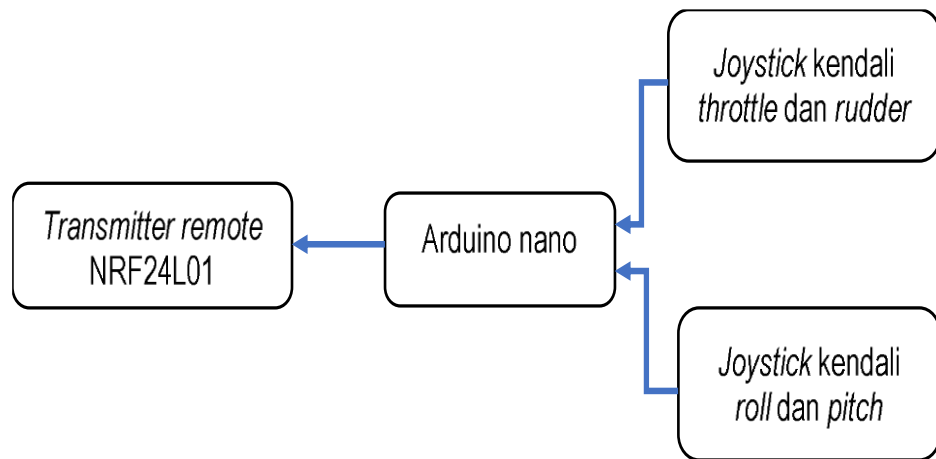
$$CG = 35\% \times MAC$$

$$CG = 35\% \times 40\text{cm}$$

$$CG = 14\text{cm}$$



Gambar 3.4 Diagram blok *Flight Controller*



Gambar 3.5 Diagram blok *Remote Control*

Sistem kontrol dengan loop tertutup digunakan penulis untuk memberikan respon keluaran agar UAV dapat terbang dengan stabil. Nilai error atau kesalahan orientasi UAV diperoleh dengan menghitung selisih dari setpoint dengan nilai pembacaan IMU. Respon keluaran akan menghasilkan variabel PPM untuk mengendalikan motor servo.

Dari Gambar 3.4 dijelaskan alur sistem kendali terbang UAV, perintah gerak diperoleh dari telemetri sementara data orientasi UAV diperoleh dari sensor IMU. Perbedaan nilai perintah dengan orientasi UAV mengakibatkan sistem merespon dengan menghasilkan deretan pula PPM untuk mengendalikan gerak servo bagi *aileron*, *elevator* dan *rudder*. Nilai kecepatan respon merupakan fungsi waktu dan nilai kekuatan untuk merespon kesalahan merupakan proportional. Kombinasi parameter fungsi waktu yang di definisikan sebagai I (Integral) dan komposisi Proporsional dapat tepat dapat diperoleh keseimbangan terbang UAV yang baik.

Dari sistem *flight control* diatas dibutuhkan *Remote Control* yang direncanakan menggunakan mikrokontroller Arduino Nano Gambar 3.5. Unit masukan sebagai kontrol atau kendali arah pesawat menggunakan *joystick arm*. *Joystick* menggunakan sebuah variabel resistor yang berubah dengan perubahan posisi *joystick*. Setiap *joystick arm* terdapat dua potensio, nilai center



potensio merupakan variabel kendali dengan rentang nilai 0-1023. Untuk keseragaman variabel kendali penulis menetapkan nilai 0-1000 yang sesuai dengan nilai kebanyakan sistem remote control. Nilai 0-1000 merupakan nilai PPM dimana, nilai 0 dapat memberikan perintah servo untuk bergerak kearah 0 derajat. Nilai 1000 dapat memberikan perintah servo untuk bergerak kearah 180 derajat.

d. Pemilihan servo

Dalam pembuatan UAV *solar cell* ini menggunakan servo TowerPro SG90S Gambar , yaitu merupakan servo memiliki bentuk yang kecil dan ringan dengan daya output tinggi . Prinsip kerjanya hampir sama dengan motor servo lainnya. Servo dapat memutar sekitar

180 derajat ( 90° ke arah kanan dan 90° ke arah kiri ). Spesifikasi dari TowerPro SG90S adalah sebagai berikut :

- 1) Dimensi sebesar 22.8 mm x 12.2 mm x 28.5 mm
- 2) Sudut maksimal 180°
- 3) Torsi seberat 1.8 kg/cm (4.8V ) 2.2 kg/cm (6 v)
- 4) *Operating speed* sekitar 0.1 sec/60 *degree*(4.8 v)
- 5) *Operating volt* sebesar 4.8-6.0 v



Gambar 3.6 Servo Tower Pro

### 3.2.4 Pengumpulan Data

Pada penelitian Rancang Bangun UAV *solar cell*, penulis menggunakan beberapa 3 metode pengumpulan data. Metode pengumpulan data rancang bangun *Platform* dan sistem kontrol pada *Unmanned Aerial Vehicle (UAV) solar cell* akan dilakukan penulis, yaitu dokumentasi, cara ini dilakukan dengan mengumpulkan data dari buku, catatan, dan hasil-hasil penelitian pada instansi terkait mengenai pesawat UAV *solar cell*.

### 3.2.5 Pengolahan Data

Pengolahan data merupakan proses yang dilakukan penulis setelah memperoleh beberapa informasi yang dikumpulkan pada proses pengumpulan data. Data yang diperoleh akan dilakukan kajian dan menjadi referensi pada proses analisis rancangan dan perancangan sistem.

### 3.2.6 Definisi Operasional

Definisi operasional pada penelitian rancang bangun ini menjelaskan dan mendefinisikan variabel yang di gunakan pada penelitian rancang bangun ini. Berikut definisi operasional pada penelitian rancang bangun, yaitu :

- a. UAV : Merupakan pesawat tanpa pilot dimana sistem kendalinya dapat dikendalikan secara otomatis melalui program komputer yang dirancang, atau melalui kendali jarak jauh.
- b. Arduino : Arduino dikatakan sebagai sebuah platform dari *physical computing* yang bersifat *open source*. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environtment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory microcontroller*.
- c. PID : (dari singkatan bahasa *proportional–integral–derivative*) *controller* merupakan pengendali untuk menentukan

presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Pengendali PID merupakan gabungan dari tiga sistem kendali yang bertujuan untuk mendapatkan keluaran dengan *rise time* yang tinggi dan galat yang kecil.

d. *Telemetry* : merupakan perangkat komunikasi data antara UAV dengan *remote control* melalui jaringan komunikasi radio RF.

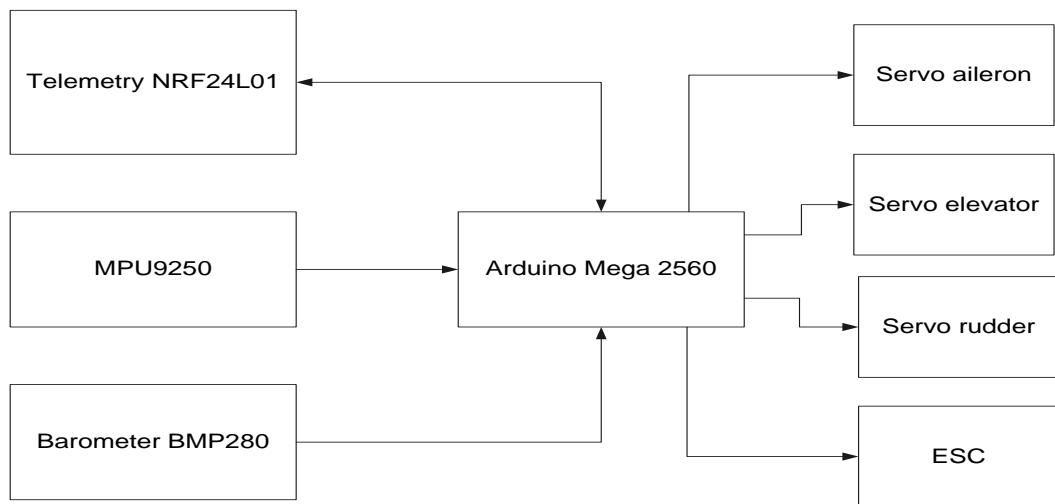
## BAB 4 ANALISA DAN PENGUJIAN

Pada bab ini akan penulis akan menjelaskan mengenai analisa dan pengujian dijelaskan mengenai analisa dan pengujian mengenai rancang bangun *Platform* dan sistem kontrol pada *Unmanned Aerial Vehicle* (UAV) *Solar Cell* yang telah selesai dibuat. Analisa dan pengujian dari perangkat yang dibuat akan dilakukan kajian untuk dilakukan koreksi jika terdapat kegagalan maupaun kesalahan kerja yang tidak sesuai dengan tujuan pembuatan perangkat.

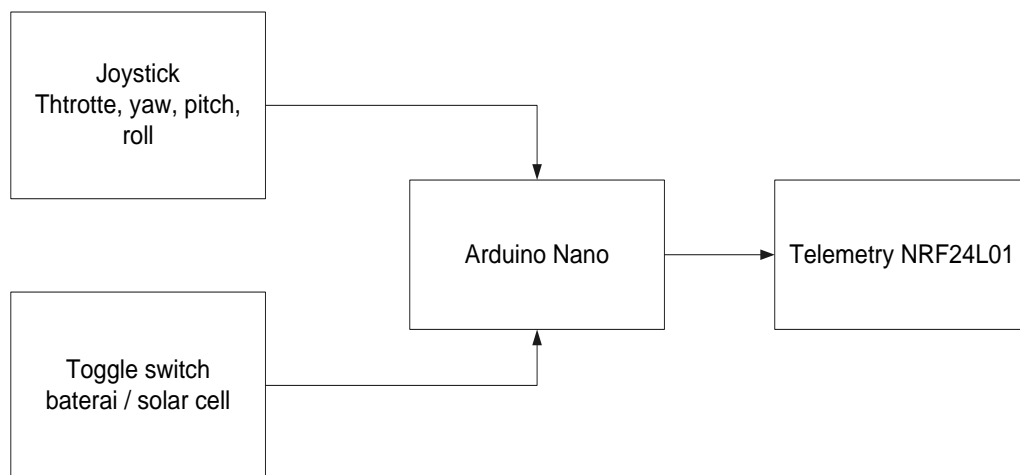
Rancang bangun *Platform* dan sistem kontrol pada *Unmanned Aerial Vehicle* (UAV) *solar cell* terdiri dari beberap fungsi kerja yang dapat disebutkan sebagai berikut :

- a. Kendali terbang menggunakan *remote control* menggunakan Arduino Nano dan telemetry NRF24L01.
- b. *Flight controller* menggunakan Arduino Mega 2560 dan telemetry NRF24L01.
- c. Sensor orientasi UAV menggunakan sensor MPU9250.
- d. Kendali kestabilan terbang menggunakan kontrol PID, dimana masukkan berdasar sensor MPU9250 dan setpoint berdasar perintah *remote control*.

Gambar 4.1 menjelaskan diagram blok perangkat *remote control*. Sebagai komponen utama pada remote control adalah Arduino Nano, joystick sebagai masukan yang dihubungkan melalui saluran analog Arduino. *Telemetry* sebagai perangkat yang mengirimkan data ke perangkat UAV. *Telemetry* dihubungkan melalui saluran SPI Arduino Nano dengan konfigurasi pin *chip enable* dan *chip select* yang diatur melalui kode program. Sementara pengaturan saluran komunikasi SPI diatur melalui hubungan fisik ke pin MOSI, MISO dan SCK.



Gambar 4.2 Fungsi Kerja UAV



Gambar 4.1 Diagram Blok *Remote Control*

Gambar 4.2 menjelaskan fungsi kerja terbang pada UAV *solar cell*. Komponen utama pada blok UAV *solar cell* adalah mikrokontroler Arduino Mega 2560. Arduino memiliki fungsi, yaitu :

- Menerima data dari *remote control* melalui telemetry NRF24L01.
- Mengolah sensor MPU9250, hingga diperoleh data kompas, *yaw*, *pitch* dan *roll*.
- Melakukan perhitungan kontrol kestabilan terbang UAV.
- Mengirimkan data sensor ke *ground station* melalui telemetry NRF24L01.
- Mengatur kerja servo, *aileron*, *elevator* dan *rudder*.

f. Mengatur kerja ESC untuk mengatur kecepatan motor penggerak propeller.

#### **4.1 Implementasi**

Pada tahap implementasi yang dilakukan pada rancang bangun UAV *solar cell* dapat dijabarkan pada langkah-langkah implementasi berikut ini.

##### **4.1.1 Implementasi Perangkat *Remote Control***

Perangkat komunikasi pada *remote control* yang digunakan pada Rancang Bangun UAV *solar cell*, merupakan perangkat komunikasi data yang bertujuan untuk mengirimkan data perintah terbang UAV *solar cell*. Penerapan perangkat komunikasi data menggunakan telemetry NRF24L01 yang dihubungkan ke perangkat Arduino Nano melalui saluran SPI.

Agar perangkat komunikasi dapat bekerja dengan baik sesuai dengan fungsi dan tujuan penggunaan perangkat, maka harus dihubungkan dengan pin-pin mikrokontroler sesuai dengan instalasi perangkat yang telah dijelaskan pada bab sebelumnya. Selain hubungan secara elektrik dengan mikrokontroler, agar dapat bekerja dengan baik maka pada perangkat mikrokontroler harus dilakukan pemrograman menggunakan bahasa C++ dengan perantara editor arduino IDE.

##### **4.1.2 Implementasi Perangkat Proses**

Perangkat atau unit proses untuk memproses data yang digunakan untuk mengatur terbang UAV *solar cell* berbasis Atmega 2560 yang dikemas pada sebuah board dengan nama arduino Mega 2560 Pro.

Kemampuan Arduino Mega 2560 untuk menampung program pembacaan sensor, mengatur kestabilan terbang dan memproses data NRF24L01 diyakini penulis cukup. Arduino Mega 2560 memiliki flash memory yang cukup besar yaitu 256kb dan I/O lebih dari 50 pin. Kebutuhan pin dalam penerapan kendali terbang UAV, yaitu :

- a. Pin I2C berkomunikasi dengan sensor IMU 9250, yaitu 2 pin (SDA dan SCL).
- b. 4 pin PWM sebagai keluaran untuk mengendalikan motor servo.

- c. 1 pin PWM sebagai keluaran untuk mengendalikan ESC.
- d. 4 pin sebagai keluaran dan masukan melalui saluran komunikasi SPI, berguna untuk berkomunikasi dengan modul NRF24L01.

Berdasarkan total kebutuhan-kebutuhan pin sebanyak 11 pin, maka Arduino Mega 2560 dirasa sudah lebih dari cukup.

## **4.2 Pengujian**

Langkah pengujian merupakan langkah lanjutan setelah implementasi dilakukan. Pengujian dilakukan perbagian yang terdiri dari pengujian perangkat *remote control*, perangkat *flight controller*, perangkat mekanik seperti motor servo maupun motor brushless serta pengujian kestabilan terbang. Pengujian dilakukan untuk memastikan semua perangkat telah bekerja dengan baik.

### **4.2.1 Pengujian Perangkat Keras**

Pada langkah pengujian ini, penulis akan melakukan pengujian perangkat elektronik dan perangkat mekanik UAV *solar cell*. Pengujian perangkat elektronik maupun perangkat mekanik dengan menghubungkan perangkat ke Arduino Mega 2560. Pengujian perangkat mekanik seperti motor brushless melalui perangkat ESC. Sementara untuk pengujian motor servo dapat dihubungkan secara langsung ke arduino karena motor servo hanya membutuhkan sumber daya 5v dan pulsa PPM.

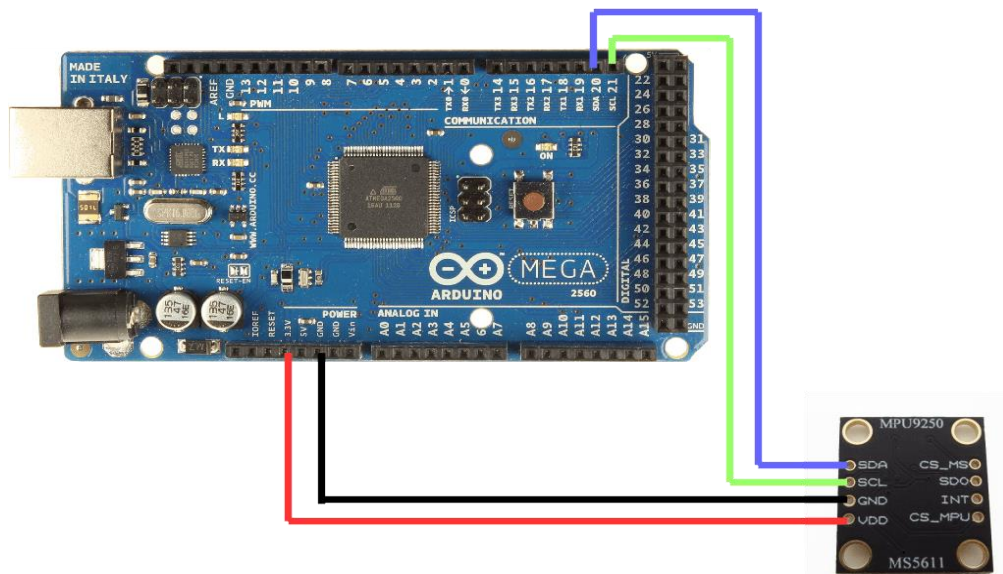
### **4.2.2 Pengujian Modul MPU9250**

Mpu9250 merupakan sensor orientasi kombinasi dari sensor kompas dan sensor IMU. Kompas digunakan untuk mengukur sudut orientasi UAV terhadap sumbu bumi dan sensor IMU digunakan untuk mengukur orientasi UAV terhadap sumbu x,y dan z. MPU9250 adalah salah satu gabungan sensor Akselerometer, Giroskop dan Kompas dengan ukuran kecil. Modul ini memiliki fitur filter *low pass*, deteksi gerakan dan prosesor yang dapat diprogram ulang.

Langkah langkah pengujian MPU9250 dapat dijelaskan sebagai berikut :

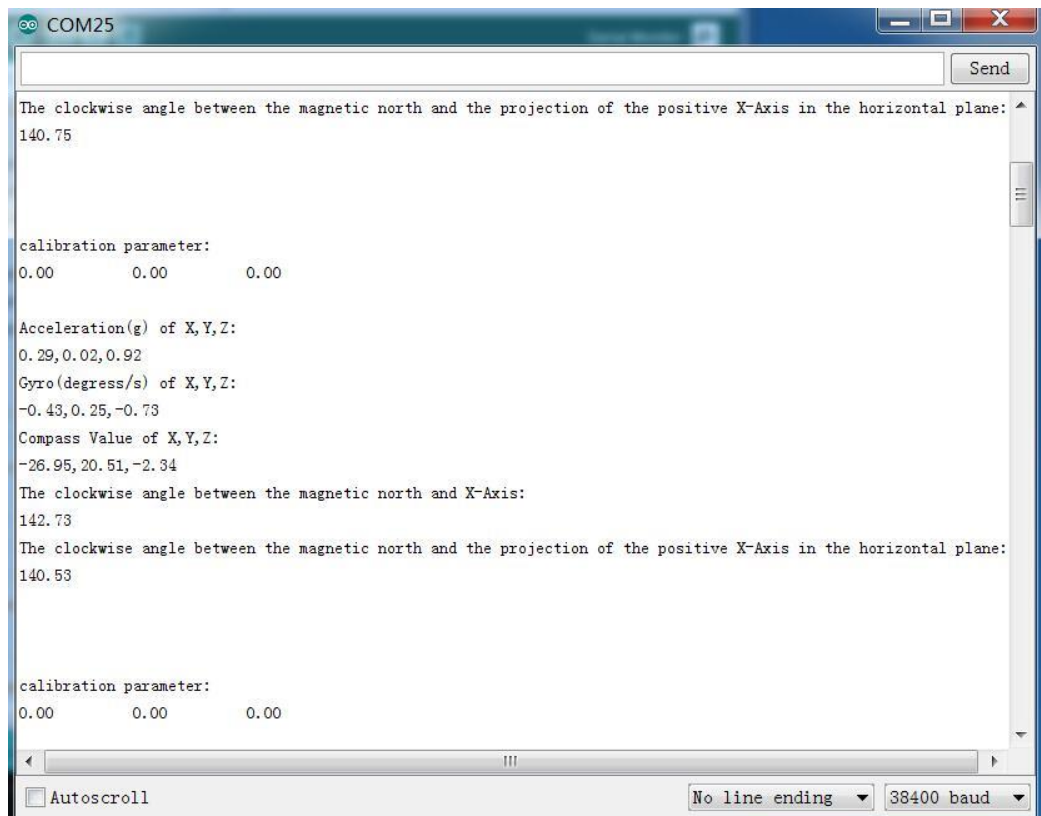
- a. Menghubungkan sumber daya 5V VCC (Kabel Merah), *Ground* (Kabel Hitam), SDA (Kabel Hijau), dan SCL (Kabel Kuning), ke Modul MPU9250
- b. Menghubungkan kabel *Ground* (Kabel hitam) ke pin *Ground* dari modul Arduino
- c. Menghubungkan kabel Power VCC 5V (kabel Merah) ke pin power 5V pada papan Arduino
- d. Menghubungkan kabel SDA (kabel Hijau) ke SDA Arduino Mega 2560
- e. Menghubungkan kabel SCL (Kabel kuning) ke SCL Arduino Mega 2560

Gambar 4.3 menjelaskan hubungan perkabelan mpu9250 dengan Arduino Mega 2560.



Gambar 4.3 Diagram Perkabelan MPU9250 Dengan Arduino Mega 2560





Gambar 4.4 Pengujian Sensor MPU 9250

Setelah pengaturan perkabelan selesai dilakukan, penulis menghubungkan Arduino Mega ke laptop melalui USB *port* dan mengunggah kode program pengujian MPU9250 ke Arduino Mega 2560. Gambar 4.4 merupakan hasil pengujian sensor MPU9250 yang ditampilkan pada serial monitor. Hasil pengujian menunjukkan nilai-nilai sensor *accelerometer*, *gyroscope* dan kompas. Untuk sensor kompas perlu dilakukan kalibrasi dengan memutar sensor searah sumbu z sebesar 360 derajat kekiri dan kekanan. Jika hal ini tidak dilakukan maka sensor kompas akan menghasilkan keluaran yang tidak sesuai dengan sumbu bumi.

#### 4.2.3 Pengujian ESC Dan Motor *Brushless*

Pada pengujian ini penulis membuat sebuah program yang bertujuan untuk mengendalikan kecepatan motor *brushless*. Besar perintah pengaturan kecepatan putar motor *brushless* di dinotasikan dalam bentuk

nilai durasi pulsa. Durasi pulsa dinyatakan dalam besaran waktu *micro second*.

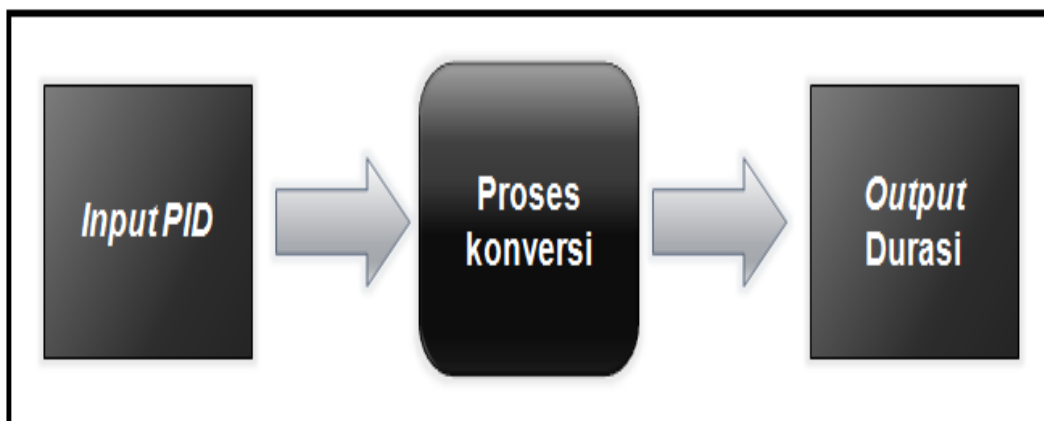
Berikut kode program untuk konversi intruksi gerak motor ke durasi pulsa.

```
pulse_length_esc1 = minMax(pulse_length_esc1, 1100, 2000);  
pulse_length_esc2 = minMax(pulse_length_esc2, 1100, 2000);  
pulse_length_esc3 = minMax(pulse_length_esc3, 1100, 2000);  
pulse_length_esc4 = minMax(pulse_length_esc4, 1100, 2000);
```

Dari kode program diatas dapat dijelaskan nilai durasi pulsa membentang dari 1100us hingga 2000us. Nilai 1100us merupakan kecepatan terendah motor *brushless* dan nilai 2000us merupakan nilai kecepatan maksimal motor *brushless*. Nilai durasi pulsa tersebut sebagai masukan *driver* esc motor *brushless*. Esc merupakan modul *driver* motor yang secara khusus dirancang untuk mengemudikan motor *brushless*.

Gambar 4.5 dan merupakan pengujian durasi pulsa menggunakan mikrokontroler arduino.

Pada pembahasan berikutnya penulis akan menjabarkan proses pengukuran motor *Brushless Thrust* atau kemampuan daya dorong motor brushless, besar arus yang lewat dan daya yang terserap pada saat yang



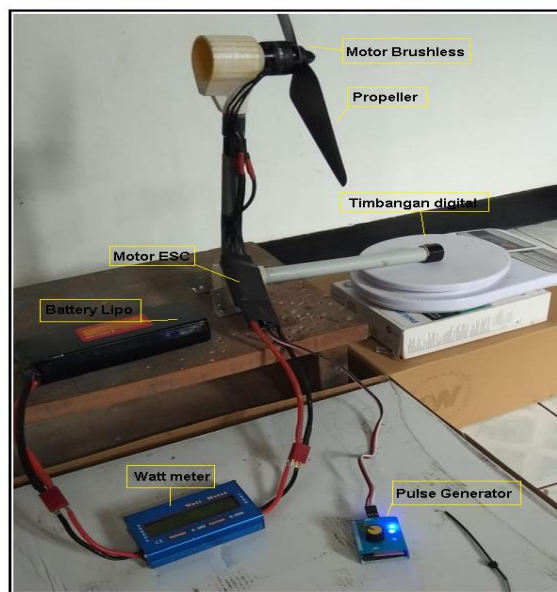
Gambar 4.5 Diagram Blok Proses Konversi

bersamaan. Untuk keperluan ini, penulis menyiapkan sebuah timbangan digital dan alat analisa daya atau *watt* meter. Sangat penting untuk mengetahui daya dorong yang tepat dan arus maksimum yang diserap oleh motor sehingga dapat mengetahui nilai daya yang dibutuhkan untuk membuat UAV *solar cell* dapat *take off*. Pada pengujian ini juga ditujukan untuk mengetahui besar efisiensi motor dengan melihat nilai konversi daya listrik ke daya dorong yang paling optimum saat UAV *solar cell* terbang.

Tujuan dari pengukuran dapat disebutkan sebagai berikut:

- a. Dapat memberikan masukan dalam pengambilan keputusan ukuran *propeller* yang tepat dan efisien untuk motor.
- b. Mengetahui batasan konsumsi arus apakah melebihi kapasitas dari ESC.
- c. Mengetahui daya dorong motor sehingga saat menerapkan ke UAV *solar cell* dapat diketahui apakah kemampuan motor memiliki kapasitas yang cukup untuk menerbangkan drone.
- d. Dapat memperkirakan waktu penerbangan (*flight time*).

Langkah awal yang diperlukan untuk melakukan pengukuran ini adalah memasang motor brushless pada sebuah tiang penopang yang dihubungkan dengan L dan ujung tiang satunya diletakkan tepat pada titik tengah timbangan digital.



Gambar 4.6 Pengujian *thrust* motor

Gambar 4.6 menjelaskan skenario pengukuran daya dorong motor, sumber *battery* dihubungkan ke motor ESC, untuk masukan lebar pulsa mengguna *pulse* generator yang dapat digunakan untuk mengemudikan motor *brushless* dan servo, keluaran motor esc dihubungkan ke motor *brushless* melalui tiga kabel fasa. Motor *brushless* dipasang pada tiang penopang L bagian atas dan ujung tiang satunya diletakkan pada titik tengah timbangan. Pada skenario uji ini dapat dijelaskan bahwa, saat motor berputar maka propeller akan menghasilkan daya tarik terhadap udara yang mengakibatkan tiang penopang yang lain akan menekan timbangan digital, besar nilai tekanan yang dihasilkan dapat dibaca pada layar lcd timbangan yang dinotasikan dalam satuan gram. Sementara besar arus dan tegangan dapat dibaca pada layar lcd power meter.

Pada pengujian penulis memperkirakan pengujian pada area yang aman, karena posisi motor adalah menarik dan titik tumpu terletak pada tengah timbangan digital.

Hal-hal yang perlu diperhatikan sebelum melakukan pengujian adalah :

- a. Motor dalam konfigurasi pull.
- b. Propeller harus diseimbangkan sebelum melakukan uji. Jika tidak seimbang, hal itu akan mengakibatkan munculnya getaran yang cukup keras .
- c. Kalibrasi ESC dengan tepat.
- d. Menghubungkan hubungan kabel power ESC pada di sisi "Load" Power Meter.
- e. Menghubungkan sisi input Power Meter ke baterai Lipo.
- f. Menghubungkan sisi input ESC pada *pulse* generator.
- g. Menaikkan *throttle* hingga mulai dari 0% ke 100% dengan step tiap 5 dan mengamati nilai Thrust pada timbangan dan masing-masing arus, tegangan dan daya pada Power Meter.

Pengujain berikut merupakan hasil pengujian daya dorong (*thrust*) pada masing-masing nilai *throttle*.



Gambar 4.7 Pengukuran Daya Dan Thrust Saat Throttle 0%.

a. Pengujian 1

Pada pengujian 1 (satu) atau pertama penulis mengatur besar *throttle pulse* generator pada posisi 0 %. Seperti diketahui meskipun nilai *throttle* sebesar 0 % akan tetapi motor esc telah menarik arus yang merupakan arus diam dari modul esc driver. Gambar 4.8 merupakan hasil pengujian pada posisi *throttle* 0%, yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital. Nilai daya yang di butuhkan 1,7 watt, arus 0,17A, tegangan 11,93 volt dan *thrust* 0gr.

b. Pengujian 2

Pada pengujian 2 (dua) atau pertama penulis mengatur besar *throttle pulse* generator pada posisi 5 %. Nilai daya yang di butuhkan 4,0 watt, arus 0,35A, tegangan 11,93 volt dan *thrust* 23gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

c. Pengujian 3

Pada pengujian 3 (tiga) penulis mengatur besar *throttle pulse* generator pada posisi 10 %. Nilai daya yang di butuhkan 4,5 watt, arus 0,38A, tegangan 11,93 volt dan *thrust* 34gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

d. Pengujian 4

Pada pengujian 4 (empat) penulis mengatur besar *throttle pulse* generator pada posisi 15 %. Nilai daya yang di butuhkan 6,2 watt, arus 0,51A, tegangan 11,93 volt dan *thrust* 54gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

e. Pengujian 5

Pada pengujian 5 (lima) penulis mengatur besar *throttle pulse* generator pada posisi 20 %. Nilai daya yang di butuhkan 7,9 watt, arus 0,68A, tegangan 11,91 volt dan *thrust* 74gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

f. Pengujian 6

Pada pengujian 6 (enam) penulis mengatur besar *throttle pulse* generator pada posisi 25 %. Nilai daya yang di butuhkan 9 watt, arus 0,74A, tegangan 11,91 volt dan *thrust* 84gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

g. Pengujian 7

Pada pengujian 7 (tujuh) penulis mengatur besar *throttle pulse generator* pada posisi 30 %. Nilai daya yang di butuhkan 10,2 watt, arus 0,87A, tegangan 11,9 volt dan *thrust* 106gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

h. Pengujian 8

Pada pengujian 8 (delapan) penulis mengatur besar *throttle pulse* generator pada posisi 35 %. Nilai daya yang di butuhkan 12,3 watt, arus 1,03A, tegangan 11,9 volt dan *thrust* 123gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

i. Pengujian 9

Pada pengujian 9 (sembilan) penulis mengatur besar *throttle pulse* generator pada posisi 40 %. Nilai daya yang di butuhkan 14,6 watt, arus 1,25A, tegangan 11,89 volt dan *thrust* 148gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

j. Pengujian 10

Pada pengujian 10 (sepuluh) penulis mengatur besar *throttle pulse* generator pada posisi 45 %. Nilai daya yang di butuhkan 17,0 watt, arus 1,43A, tegangan 11,88 volt dan *thrust* 179gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

k. Pengujian 11

Pada pengujian 11 (sebelas) penulis mengatur besar *throttle pulse* generator pada posisi 50 %. Nilai daya yang di butuhkan 22,4 watt, arus 1,96A, tegangan 11,66 volt dan *thrust* 212gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

l. Pengujian 12

Pada pengujian 12 (dua belas) penulis mengatur besar *throttle pulse* generator pada posisi 55 %. Nilai daya yang di butuhkan 26,6 watt, arus 2,23A, tegangan 11,84 volt dan *thrust* 257gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

m. Pengujian 13

Pada pengujian 13 (tiga belas) penulis mengatur besar *throttle pulse* generator pada posisi 60 %. Nilai daya yang di butuhkan 31,3 watt, arus 2,68A, tegangan 11,83 volt dan *thrust* 287gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

n. Pengujian 14

Pada pengujian 14 (empat belas) penulis mengatur besar *throttle pulse* generator pada posisi 65 %. Nilai daya yang di butuhkan 39,5 watt, arus 3,3A, tegangan 11,81 volt dan *thrust* 369gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

o. Pengujian 15

Pada pengujian 15 (lima belas) penulis mengatur besar *throttle pulse* generator pada posisi 70 %. Nilai daya yang di butuhkan 47,5 watt, arus 4,0A, tegangan 11,78 volt dan *thrust* 403gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

p. Pengujian 16

Pada pengujian 16 (enam belas) penulis mengatur besar *throttle pulse* generator pada posisi 75 %. Nilai daya yang di butuhkan 59,5 watt, arus 5,0A, tegangan 11,72 volt dan *thrust* 498gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

q. Pengujian 17

Pada pengujian 17 (tujuh belas) penulis mengatur besar *throttle pulse* generator pada posisi 80 %. Nilai daya yang di butuhkan 5,71 watt, arus 5,71A, tegangan 11,69 volt dan *thrust* 551gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

r. Pengujian 18

Pada pengujian 18 delapan belas) penulis mengatur besar *throttle pulse* generator pada posisi 85 %. Nilai daya yang di butuhkan 76,5 watt, arus 6,52A, tegangan 11,66 volt dan *thrust* 599gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

s. Pengujian 19

Pada pengujian 19 (sembilan belas) penulis mengatur besar *throttle pulse* generator pada posisi 90 %. Nilai daya yang di butuhkan 86,9 watt, arus 7,40A, tegangan 11,63 volt dan *thrust* 648gr. Yang ditunjukkan pengukuran menggunakan *power* meter dan timbangan digital.

t. Pengujian 20

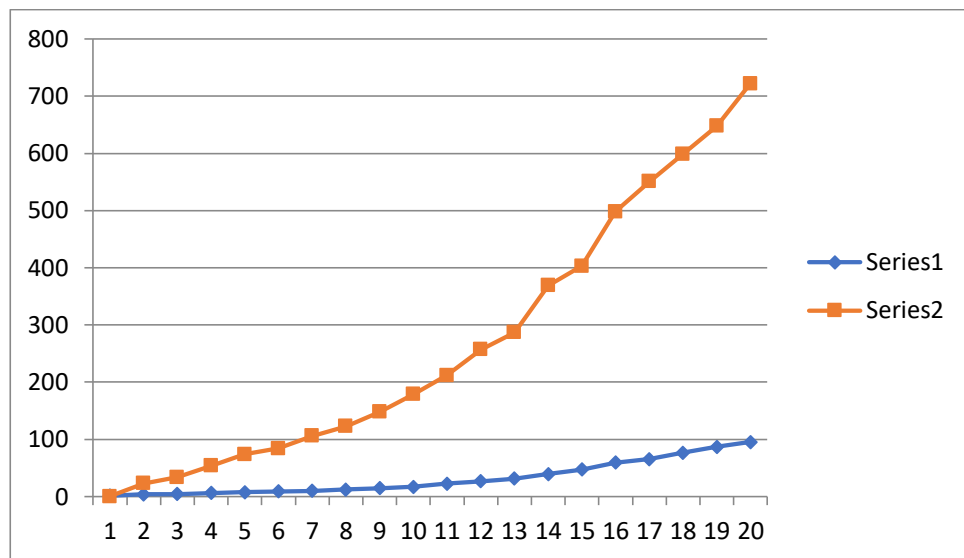


Pada pengujian 20 (dua puluh) penulis mengatur besar *throttle pulse* generator pada posisi 95 %. Nilai daya yang di butuhkan 95,4 watt, arus 8,14A, tegangan 11,6 volt dan *thrust* 722gr. Yang ditunjukkan pengukuran menggunakan *power meter* dan timbangan digital.

Tabel 4.1 Tabel Hasil Pengujian *Thrust* Motor

No	Arus (I)	Tegangan (V)	Power (W)	Thrust (gr)
1	0,17	11,93	1,7	0
2	0,35	11,93	4,0	23
3	0,38	11,93	4,5	34
4	0,51	11,93	6,2	54
5	0,68	11,91	7,9	74
6	0,74	11,91	9	84
7	0,87	11,9	10,2	106
8	1,03	11,9	12,3	123
9	1,25	11,89	14,6	148
10	1,43	11,88	17,0	179
11	1,96	11,66	22,4	212
12	2,23	11,84	26,6	257
13	2,68	11,83	31,3	287
14	3,3	11,81	39,5	369
15	4,0	11,78	47,5	403
16	5,0	11,72	59,5	498
17	5,71	11,69	65,4	551
18	6,52	11,66	76,5	599
19	7,40	11,63	86,9	648
20	8,14	11,6	95,4	722

Berdasarkan tabel pengujian Tabel 4.1 mengenai Pengujian *Thrust* Motor, dapat dibuat sebuah grafik yang menunjukkan kurva perbandingan konsumsi energi listrik menjadi daya dorong atau



Gambar 4.8 Grafik Perbandingan Daya Dengan *Thrust* Motor

*thrust*. Dari grafik yang telah dibuat gambar 4.26 dapat dianalisa bahwa perubahan daya listrik ke daya dorong tidak linier. Dari data pengujian tersebut dapat digunakan untuk menghitung berapa daya yang di butuhkan motor sehingga bisa menentukan kebutuhan dari *solar cell* .

Keterangan

Series 1= daya listrik yang terserap dalam satuan watt (W)

Series 2= *thrust* atau daya dorong dalam satuan gram (gr)

#### 4.2.4 Perhitungan Aerodinamika Pesawat UAV

Tujuan utama sayap pada pada pesawat UAV pada umunya untuk menghasilkan daya angkat. Daya angkat adalah kekuatan yang membuat pesawat tetap terbang. Mengubah area sayap dapat mengubah jumlah daya angkat yang dihasilkan oleh pesawat, yang dapat mempengaruhi kinerja pesawat. Bagian terpenting dari ukuran sayap adalah luas sayap ketika melihatnya dari bawah. Area ini disebut area *planform*. Area *planform*, yang termasuk bagian bawah badan pesawat, merupakan indikasi area di mana daya angkat dihasilkan. Jika area *planform* yang lebih besar, maka hal tersebut merupakan area yang memiliki daya angkat lebih besar dihasilkan. Jadi bisa diperoleh lebih banyak daya angkat dengan

memiliki sayap yang lebih besar, Lebih banyak mengangkat bisa menjadi hal yang baik. Hal ini dapat membuat pesawat menjadi lebih berat. Tetapi peningkatan daya angkat bersamaan pada nilai peningkatan berat dan hambatan. Ketika sebuah pesawat terbang melaju, gaya angkat yang dihasilkannya harus sama dengan beratnya. Jika lebih rendah, pesawat akan jatuh. Jika lebih tinggi, pesawat akan naik. Ini berarti bahwa jika memiliki lebih banyak daya angkat, bobot akan semakin berat.

Pada perencanaan sayap pesawat UAV *Solar Cell* penulis menggunakan persamaan daya angkat atau *lift* sebagai berikut.

$$L = \frac{1}{2} \times \rho \times v^2 \times A \times Cl$$

Dimana

$\rho$  = adalah kepadatan udara di sekitarnya

$v^2$  = adalah kecepatan udara (kecepatan pesawat relatif terhadap udara),

$A$  = adalah area planform

$Cl$  = adalah koefisien *lift*.

Koefisien *lift* menentukan daya angkat yang dihasilkan berdasarkan geometri atau bentuk dari sayap. Jumlah ini berubah dengan berbagai sudut datang, dan juga dengan perubahan kecepatan atau kondisi udara.

Dari persamaan tersebut, dapat dilihat bahwa dengan meningkatkan area bentuk rencana sayap, dapat meningkatkan gaya angkat yang dihasilkan  $L$ , dengan jumlah yang sama. Disebut hubungan proporsional, jadi dapat dikatakan bahwa gaya angkat  $L$  sebanding dengan bidang perencanaan sayap. Dapat dilihat juga bahwa peningkatan kecepatan udara  $v$ , akan meningkatkan gaya angkat yang dihasilkan  $L$ . Variabel  $v$  dikuadratkan, kenaikan yang dilakukan  $v$ , variabel  $a$  lebih besar akan dibuat untuk menghasilkan  $L$ . Misalnya, jika kecepatan udara  $v$  digandakan, maka pesawat atau roket akan menghasilkan empat kali lipat daya angkat. Dapat disebut bahwa *lift* atau daya angkat sebanding dengan kuadrat kecepatan. Secara umum daya angkat juga dapat dipengaruhi oleh kepadatan udara,  $\rho$ , dan koefisien daya angkat,  $CL$ . Kepadatan udara kepadatan ditentukan oleh ketinggian, dan tidak mempengaruhi kendali

pesawat melalui radio komunikasi saat terbang cukup rendah dengan tanah.

Aspek rasio sayap merupakan ukuran rentang sayap berbanding dengan lebar sayap. Aspek rasio dengan mengkuadratkan bentang dan membaginya dengan area sayap.

Aspek rasio sayap dapat diperoleh dengan formula sebagai berikut

$$\text{Aspect Ratio} = \frac{\text{Wing span}}{A}$$

$$\text{Aspect Ratio} = \frac{(3000 \text{ mm})^2}{1170000 \text{ mm}^2}$$

$$\text{Aspect Ratio} = 7.69$$

Dari perencanaan sayap pesawat UAV penulis dapat menghitung daya angkat atau *lift* sebagai berikut.

$$L = \frac{1}{2} \times p \times v^2 \times A \times Cl$$

Dimana

P = adalah kepadatan udara di sekitarnya 1.03 kg/m<sup>3</sup>

$v^2$  = adalah kecepatan udara (kecepatan pesawat relatif terhadap udara), sebagai asumsi 10 km/jam atau 2.7 meter/detik

A = adalah area planform 1170000 mm<sup>2</sup>

Cl adalah koefisien lift.

$$Cl = \frac{2 L}{p \times v^2 \times A}$$

Dengan asumsi tekanan karena berat karena pesawat sebesar 4,1 Kg atau 40,2 Newton, diperoleh dengan mengalikan bobot dengan 9,81 nilai gravitasi bumi. Maka koefisien *lift* diperoleh

$$Cl = \frac{2 \times (4.1 \text{ kg} \times 9,81)}{1,03 \times 2,7^2 \times 1.170 \text{ m}^2}$$

$$Cl = 7.826$$

Pada pengukuran *thrust* motor dengan asumsi daya listrik yang terserap dari hasil pengujian diperoleh nilai daya sebesar 190,8Watt. Pada kedua motor 5010 dengan propeller 17 x 5.5 diperoleh nilai *thrust* sebesar:

$$\text{Power motor kanan} = P/2$$

$$\text{Power motor kanan} = 190,8/2$$

*Power motor kanan = 95.4 watt*

*Power motor kiri =  $P/2$*

*Power motor kiri = 190,8/2*

*Power motor kiri = 95.4 watt*

Dari data pengujian motor 5010, diperoleh nilai *thrust* sebesar 722g pada serapan daya motor 95.4 watt. Sehingga *thrust* total motor dapat dihitung dengan formula sebagai berikut :

*Thrust = thrust per motor x2*

*Thrust = 722g x2*

*Thrust = 1444g*

Nilai *thrust* motor keseluruhan diperoleh nilai sebesar 1,45kg. Dari data tersebut dapat di simpulkan bahwa daya *thrust* tidak mencukupi untuk mengangkat pesawat, namun akan mampu untuk dapat terbang karena di bantu dari daya angkat atau daya *lift* dari wing.

#### **4.2.5 Pengujian Terbang UAV Solar Cell**

Pada pengujian ini penulis melakukan pengujian terbang pesawat UAV, pada area persawahan yang dilakukan didaerah sidoarjo. Lokasi ini dipilih karena area yang terbentang luas lebih dari 10 Ha. Pengujian dilakukan dengabn persyaratan bahwa area cukup jauh dari area perumahan atau perkampungan karena sinyal wifi daerah berpenduduk dapat mengganggu sinyal kendali terbang. Pada pengujian ini pesawat UAV *Solar Cell take off* dengan cara dilempar. Hasil pengujian terbang pesawat UAV setelah take off ditunjukkan pada Gambar 4.11 Setelah *take off* dan terbang sehingga memperoleh terbang stabil, penulis menguji manuver pesawat UAV *Solar Cell* yang ditunjukkan pada Gambar 4.9. Pada pengujian ini penulis mengubah arah terbang pesawat berbelok kekanan. Pada melakukan manuver berbelok 180 derajat. Ada saat dimana pesawat diterpa angin dari sisi samping kiri. Dari hasil pengujian pesawat UAV *Solar Cell* dapat terbang stabil meskipun di terpa angin dari sisi kiri

Pada pengujian terbang UAV *Solar Cell*, penulis melakukan uji terbang pesawat UAV *Solar Cell* searah dengan arah angin. Hasil pengujian ini ditunjukkan pada Gambar 4.10.



Gambar 4.11 Uji Manuver Pesawat UAV *Solar Cell*



Gambar 4.10 Pengujian Terbang UAV *Solar Cell* Searah Arah Angin



Gambar 4.9 Pengujian Terbang Pesawat UAV *Solar Cell*

## **BAB 5**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan pengujian pada penelitian rancang bangun *Platform* dan sistem kontrol pada *solar cell* UAV (*Unmanned Aerial Vehicle*) yang telah dilakukan penulis dapat dijabarkan kesimpulan dan saran sebagai berikut :

- a. Berdasarkan hasil perhitungan yang telah dilakukan penulis dalam merancang bodi pesawat UAV *Solar Cell*. Diperoleh ukuran bentang sayap 3meter dengan berat total sebesar 4.1kg. Maka hasil perhitungan diperoleh nilai rasio sayap sebesar 7.59 yang merupakan rasio sayap untuk pesawat glider. Berdasarkan hasil pengujian sistem propulsi UAV *Solar Cell*, diperoleh nilai *thrust* yang mampu mendorong pesawat UAV *Solar Cell* untuk terbang. Hal ini dibuktikan dengan daya 190,8 watt atau setara dengan *thrust* 1,45kg mampu menerbangkan pesawat UAV dengan bobot 4.1kg.
- b. Sistem kendali *remote control* yang dibuat penulis dapat digunakan untuk mengendalikan pesawat UAV *Solar Cell* dengan baik, dari hasil pengujian terbang pesawat dapat dikendalikan dengan baik. Sistem penerima *remote control* yang dibuat penulis dapat digunakan untuk menerima data dari *remote control* dengan baik. Dari hasil pengujian terbang pesawat penerima dapat menerima data perintah terbang sehingga terbang pesawat UAV *Solar Cell* dapat terbang secara terarah sesuai dengan perintah kendali oleh pilot .

#### **5.2 Saran**

Berdasarkan hasil penelitian Rancang Bangun UAV *Solar Cell*, penulis menyusun saran bagi pengembangan penelitian ini agar diperoleh UAV *Solar Cell* yang lebih baik, yaitu mengembangkan sumber daya elektrik dengan memanfaatkan tiupan angin untuk dikonversi menjadi sumber energi listrik yang dapat digunakan untuk pengisian baterai.

## DAFTAR PUSTAKA

- Abidin, Hasanuddin Z. Geodesi Satelit. Institut Teknologi Bandung. Pradnya Paramita. Jakarta : 2001
- Ariyono Setiawan, Sudrajat, Ridho Rinaldi, Didi Hariyanto, "Perancangan Sistem Kontrol Pesawat Model Tanpa Awak (*Unmanned Aerial Vehicle*) Berbasis Arduino".
- Djuandi, Feri, 2011. "Pengenaln Arduino". Jakarta: Penerbit Elexmedia.  
Sumber : [www.arduino.cc/en/Products/Counterfeit](http://www.arduino.cc/en/Products/Counterfeit), 2014  
<http://aeroengineering.co.id/2016/01/control-surface-pada-pesawat-terbang/>
- Hardy Samuel Saroinsong, V. C. (2018). *Rancang Bangun Wahana Pesawat Tanpa Awak (Fixed Wing)*, Universitas Sam Ratulangi. Diambil kembali dari <https://ejournal.usrat.ac.id/index.php/elekdankom/article/download/19195/18753>.
- Karthik Reddy Buchireddy Sri, P. A. (2016). *Design Analysis of Solar Powered Unmanned Aerial Vehicle* VIT University School of Mechanical Engineering Mechanical Departement Vellore/Tamil Nadu India. Diambil kembali dari <http://www.jatm.com.br/ojs/index.php/jatm/article/view/666>.
- Md. Shaiful Islam, Mohammad Mukit Hasan, Mir Md. Al Kamah Tamal, Md. Jamal Mian, dan Md. Touhidur Rahman Evan, "Detail Solidworks Design and Simulation of an Unmaned Air Vehicle"
- Scott Morton, Ruben D'Sa, dan Nikolaos Papanikolopoulos, "Solar Powered UAV: Design and Experiments"
- W W Zhang, L. G. (2019). *Structural Design and Difficulties of Solar UAV*, School of Aerospace Engineering, Shenyang Aerospace Univercity, Shenyang, China. Diambil kembali dari <https://iopscience.iop.org/article/10.1088/1757-899x/608/1/012016/meta>.
- <https://lastminuteengineers.com/nrf24l01-arduino-wireless-communication/>
- <https://www.insinyoer.com/wp-content/uploads/2015/09/4-pole-bldc-motor021804-550x400.gif>



<https://howtomechatronics.com/wp-content/uploads/2019/02/How-does-an-ESC-Work-Electronic-Speed-Controller.png>

<https://aeroengineering.co.id/2014/01/pesawat-remot-kontrol-aeromodelling, 2014>

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fjournal.untar.ac.id>

[https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/fig7PropellerCartoon\\_web.jpg](https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/fig7PropellerCartoon_web.jpg)

<https://www.insinyoer.com/wp-content/uploads/2019/05/maxresdefault-600x338.jpg>

[https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/fig7PropellerCartoon\\_web.jpg](https://web.mit.edu/16.unified/www/FALL/thermodynamics/notes/fig7PropellerCartoon_web.jpg)

<https://teknologisurya.files.wordpress.com/2011/10/solar-cell-work.jpg?w=584>

## LAMPIRAN

### Listing program Transmitter

```
#include <SPI.h>
#include "RF24.h"
#include "printf.h"
#define CE    9
#define CSN   10
RF24 radio(CE,CSN);
const uint64_t pipes[2] = { 0xABCDABCD71LL, 0x544d52687CLL };
#define YAW_CTRL    A5 //
#define THROTTLE_CTRL  A6 //
#define PITCH_CTRL   A4 //--
#define ROLL_CTRL    A2 //--
#define SW_STAB      4
#define SW_PWR       3
#define LED          2

struct receiveData{
    int t;
    int y;
    int p;
    int r;
    int s;
    int w;
};

void LED_OFF(){
    digitalWrite(LED,HIGH);
}
```

```

void LED_ON(){
    digitalWrite(LED,LOW);
}

```

```

void setup() {
    //analogReference(EXTERNAL);
    Serial.begin(9600);
    printf_begin();
    radio.begin();
    radio.setChannel(123);
    radio.setPALevel(RF24_PA_MAX);
    radio.setDataRate(RF24_250KBPS);
    radio.printDetails();
    radio.setAutoAck(false);
    //radio.setRetries(15,15);
    radio.openWritingPipe(pipes[0]);
    //radio.openReadingPipe(1,pipes[1]);
    radio.stopListening();
    pinMode(LED,OUTPUT);
    pinMode(SW_STAB,INPUT_PULLUP);
    pinMode(SW_PWR,INPUT_PULLUP);
    LED_OFF();

}

```

```

int8_t yaw;
int8_t throttle;
int8_t pitch;
int8_t roll;

```

```

typedef union {

```

```

int a;
byte b[2];
} varconverter;

unsigned long led_t,tm_error;
uint16_t t,y,p,r,cal;
unsigned long trimTime;
int tm = 50;

void readJoystick(){
    unsigned long time = millis();
    receiveData tx;
    tx.t                                     =
constrain(map(analogRead(THROTTLE_CTRL),409,612,2000,1000),1000
,2000);
    tx.y                                     =
constrain(map(analogRead(YAW_CTRL),408,612,1000,2000),1000,2000);
    tx.p                                     =
constrain(map(analogRead(PITCH_CTRL),437,585,2000,1000),1000,200
0);
    tx.r                                     =
constrain(map(analogRead(ROLL_CTRL),408,617,2000,1000),1000,2000
);
    if ( digitalRead(SW_STAB)== HIGH &&  digitalRead(SW_PWR) ==
HIGH ) {
        tx.w = 1000;
    } else if ( digitalRead(SW_STAB)== LOW &&  digitalRead(SW_PWR) ==
HIGH ) {
        tx.w = 1500;
    } else if ( digitalRead(SW_STAB)== HIGH &&  digitalRead(SW_PWR)
== LOW ) {
        tx.w = 2000;
    }
}

```

```

    }
    //tx.s = map(digitalRead(SW_MIX),0,1,1000,2000);
    //tx.w = map(digitalRead(SW_PWR),0,1,1000,2000);
    radio.write(&tx,sizeof(tx));
    LED_SUCCESS();
}

void LED_SUCCESS(){
    if(millis()-led_t >= 50){
        digitalWrite(LED, !digitalRead(LED));
        led_t = millis();
    }
}

int tled;
void LED_ERROR(){
    if(millis()-led_t >= 100){
        tled = !tled;
        digitalWrite(LED, tled);
        led_t = millis();
    }
}

void loop() {
    readJoystick();
    delay(1);
}

```

## Listing program Flight Controller

```
#include "PARAMS.h"
#include <SPI.h>
#include "RF24.h"
#include "printf.h"

const uint64_t pipes[2] = { 0xABCDABCD71LL, 0x544d52687CLL };
const uint64_t pipes_tx[2] = { 0xABCDABCD72LL, 0x544d52687ELL };

#define MOTOR_PIN 3
#define AILERON_R 2
#define AILERON_L 4
#define ELEVATOR 5
#define RUDDER 6
#include <Servo.h>
#define SERVOS 5
uint8_t g_pinsOut[SERVOS] = {AILERON_L, AILERON_R, ELEVATOR,
RUDDER, MOTOR_PIN};
Servo servo[5];

#define CE 44
#define CSN 46
#define CE_ 45
#define CSN_ 47
#define LED 13

struct receiveData{
    int t = 1000;
    int y = 1500;
    int p = 1500;
    int r = 1500;
```

```

    int s = 1000;
    int w = 1000;
};

struct txData{
    int32_t data[8]; //compass, yaw, pitch, roll  lat, lng, speed, sat, hdop, altitude
    bat_voltage, current, solar_voltage
};

void LED_OFF(){
    digitalWrite(LED, HIGH);
}

void LED_ON(){
    digitalWrite(LED, LOW);
}

RF24 radio(CE, CSN);
boolean debug = true;
double Kp = 1.4, Ki = 1.5, Kd = 1.1;
boolean USING_CONTROLLER = false;
#include "Wire.h"

void gps_(){};
void cmps_(){};
int armed = 0;

float error_pid[3], P[3], I[3], D[3], PID_value[3];
float previous_error_pid[3], previous_I[3];

```

```

#include <MPU6050_tockn.h>
MPU6050 mpu6050(Wire);

#define PIN_VSOURCE  A1
#define PIN_ISOURCE  A0
#define PIN_ICHARGE  A3
#define PIN_VBATT    A2

int ANALOG_PIN [] = { PIN_VSOURCE, PIN_ISOURCE, PIN_ICHARGE,
PIN_VBATT};
int ANALOG_DATA [4];

#define NUMS_SENS 4
#define SENSISITIVITY 0.066 // 66mv/Ampere
float CALC_ANALOG [] = { 2.0f, 55.75757576f, 5.7f, 0.7576f };
float offset_voltage = 2.198f;//4.4f/2.0f;
//3.3 = 0.165
//11.50 = 2.734
unsigned long current_timer;

int counter = 0;
double avg_offset = 0;
double avg_current_charging = 0;
double avg_voltage_charging = 0;
int average = 0;

void ANALOG_READ () {
    solar_voltage = ((float)analogRead ( PIN_VSOURCE )/1023.0f)*5.0f *
(11.58f/1.6275f) * 100;

```



```

    current = (((float)analogRead ( PIN_ISOURCE )/1023.0f) * 5.0f ) *
(184.0f/3.3f) * 100;
    int16_t vbat = ((float)analogRead ( PIN_VBATT )/1023.0f) * 5100;
    vbat = map ( vbat, 0, 4300, 0, 1720 );
    if ( vbat < 0 ) vbat = 0;
    if ( counter < 1000 ) {
        avg_offset += analogRead ( PIN_ICHARGE );
        counter++;
    } else {
        double avg_curr_offset = avg_offset /1000;
        float v = (((float)(analogRead ( PIN_ICHARGE )- avg_curr_offset) /
1023.0f) * 5.0f );
        if ( v < 0 ) v = 0;
        if ( average < 10 ) {
            avg_current_charging += ( v / SENSISITIVITY ) * 100;
            avg_voltage_charging += ( vbat /1 );
            average++;
        } else {
            current_charge = avg_current_charging/10;
            bat_voltage = avg_voltage_charging/10;
            avg_current_charging = 0;
            avg_voltage_charging = 0;
            average = 0;
        }
    }

    /*if ( millis() - current_timer >= 100 ) {
        Serial.print(bat_voltage);
        Serial.print("\t");
        Serial.println( current_charge );
        current_timer = millis();
    }*/
}

```

```

//delay (5);

}

//*****

#define PIN_BATT_CHARGE 8//PWM
#define PIN_BATT_USE 9//DIGITAL
#define PIN_CELL_USE 10//DIGITAL
int PWM_CHARGE = 0;
int16_t PEM_CHARGE_MAX = 0;
int16_t PEM_CHARGE_MIN = 255;
#include <PID_v1.h>
double Kpc = 0.5, Kic = 0.75, Kdc = 0.0040;
double Setpoint, Input, Output;
PID PID_CHARGE(&Input, &Output, &Setpoint, Kpc, Kic, Kdc, DIRECT);
float pid_charge_error;
float pid_charge_value;
float CP, CI, CD;
float previous_error_pid_C, previous_I_C;

//*****

boolean SOLAR_PWR = false;
#define CURRENT_CHARGE 200 //2A
void MANAGE_POWER () {
    if ( solar_voltage >= 1700 ) {
        USE_BATTERY_OFF ();
        SOLAR_PWR = true;
        if ( bat_voltage < 1680) {
            pid_charge_error = (CURRENT_CHARGE) - current_charge;
            CP = pid_charge_error;
            CI = CI + previous_I_C;
            CD = pid_charge_error-previous_error_pid_C;
            pid_charge_value = ((Kpc)*CP) + ((Kic)*CI) + ((Kdc)*CD);

```

```

        previous_I_C= CI;
        previous_error_pid_C = pid_charge_error;
        pid_charge_value = constrain(pid_charge_value,0,255);
        analogWrite ( PIN_BATT_CHARGE, 255-pid_charge_value );
    } else {
        previous_I_C = 0;
        previous_error_pid_C = 0;
        analogWrite ( PIN_BATT_CHARGE, 255 );
        CI = 0;
        CD = 0;
    }
    LED_USE_SOLAR_PANEL();
} else {
    LED_BATERAY_USE_BLINK ();
    USE_BATTERY_ON ();
}
}
}

```

```

void USE_BATTERY_ON () {
    digitalWrite ( PIN_BATT_USE, LOW );
}

```

```

void USE_BATTERY_OFF () {
    digitalWrite ( PIN_BATT_USE, HIGH );
}

```

```

boolean USE_BATERAY = false;
#define BUZZER_PIN 29
#define LED_PIN 31
#define LEDSOLAR_PIN 30

```

```

//#include <TimerOne.h>

double SetpointCmd[3], InputImu[3], OutputPPM[3];
double Kplmu=2, Kilmu=2.1, Kdlmu=0.01;
PID PIDRoll(&InputImu[0], &OutputPPM[0], &SetpointCmd[0], Kplmu,
Kilmu, Kdlmu, DIRECT);
PID PIDPitch(&InputImu[1], &OutputPPM[1], &SetpointCmd[1], Kplmu,
Kilmu, Kdlmu, DIRECT);

void setup() {
    Serial.begin(9600);
    Serial2.begin(9600);
    Setpoint = 1500;

    PIDRoll.SetMode(AUTOMATIC);
    PIDRoll.SetSampleTime(10);
    PIDRoll.SetOutputLimits ( -500, 500 );

    PIDPitch.SetMode(AUTOMATIC);
    PIDPitch.SetSampleTime(10);
    PIDPitch.SetOutputLimits ( -500, 500 );

    PID_CHARGE.SetMode(AUTOMATIC);
    PID_CHARGE.SetSampleTime(100);
    PID_CHARGE.SetOutputLimits ( 125, 255 );
    pinMode ( BUZZER_PIN, OUTPUT );
    pinMode ( LED_PIN, OUTPUT );
    pinMode ( LEDSOLAR_PIN, OUTPUT );
    for ( int i =0; i < 6; i++ ) {
        digitalWrite ( BUZZER_PIN, LOW );
        digitalWrite ( LED_PIN, LOW );
        delay ( 200 );
    }
}

```

```

    digitalWrite ( BUZZER_PIN, HIGH );
    digitalWrite ( LED_PIN, HIGH );
    delay ( 200 );
}

pinMode ( PIN_BATT_CHARGE, OUTPUT );
analogWrite ( PIN_BATT_CHARGE, 255 );
pinMode ( PIN_BATT_USE, OUTPUT );
//digitalWrite ( PIN_BATT_USE, HIGH );
USE_BATTERY_OFF ();
pinMode ( PIN_CELL_USE, OUTPUT );
digitalWrite ( PIN_CELL_USE, LOW);
Wire.begin();
Wire.setClock(400000UL);
mpu6050.begin();
mpu6050.calcGyroOffsets(true,500, 2000);
pinMode ( MOTOR_PIN, OUTPUT );
for (uint8_t i = 0; i < SERVOS; ++i){
    servo[i].attach(g_pinsOut[i],1000,2000);
    servo[i].writeMicroseconds(1000);
}
servo[0].writeMicroseconds(1500);
servo[1].writeMicroseconds(1500);
servo[2].writeMicroseconds(1500);
servo[3].writeMicroseconds(1500);
servo[4].writeMicroseconds(1000);
delay(200);
pinMode(LED,OUTPUT);
LED_ON();
delay(100);

if(debug)printf_begin();
radio.begin();

```

```

radio.setChannel(123);
radio.setPALevel(RF24_PA_HIGH);
radio.setDataRate(RF24_250KBPS);
radio.setAutoAck(false);
radio.openReadingPipe(1, pipes[0]);
radio.openWritingPipe(pipes_tx[0]);
//radio.startListening();
LED_OFF();
if(debug)Serial.println("-----");
if(debug)radio.printDetails();
/*radioTx.begin();
radioTx.setChannel(120);
radioTx.setPALevel(RF24_PA_LOW); //-6db
//RF24_PA_MIN, RF24_PA_LOW, RF24_PA_HIGH and RF24_PA_MAX
//NRF24L01: -18dBm, -12dBm, -6dBm, and 0dBm
radioTx.setDataRate(RF24_250KBPS);
radioTx.setAutoAck(false);
//radio.setRetries(15,15);
radioTx.openWritingPipe(pipes_tx[1]);
radio.openReadingPipe(1, pipes_tx[0]); */
if(debug)Serial.println();
if(debug)Serial.println("-----");
//if(debug)radioTx.printDetails();
if(debug)Serial.println("-----");
//*****
configure_magnetometer();
calibrate_magnetometer();
//config_gyro();
//calibrate_gyro();
//Scheduler.start(gps_, gps_task);
//Scheduler.start(cmps_, getCompass);
//Timer1.initialize(1000);

```

```

    //Timer1.attachInterrupt(gps_task);
}

/*
 * Gps task,
 */
void gps_task(){
    if ( Serial2.available() > 0 ) {
        //delay (10);
        while ( Serial2.available() ) {
            //int dt_gps = Serial2.read();
            //Serial.print((char)dt_gps);
            if ( gps.encode(Serial2.read()) ){
                if ( gps.location.isValid() ){
                    flat = (gps.location.lat() * 1000000);
                    flon = (gps.location.lng() * 1000000);
                }
                if ( gps.satellites.isValid() ) {
                    satellites = gps.satellites.value();
                    //Serial.println (satellites );
                }
                if ( gps.altitude.isValid() ) {
                    altitude_ = (gps.altitude.meters() * 100);
                }
                if ( gps.hdop.isValid() ) {
                    hdop_ = (gps.hdop.hdop() * 100);
                }
                if ( gps.speed.isValid() ) {
                    speed_ = (gps.speed.kmph() * 100);
                }
            }
        }
    }
}

```

```

        //Serial.println();
    }
}

unsigned long led_t, interval_rx;
void led_success(){
    int16_t delta_time = micros()- interval_rx;
    interval_rx = micros ();
    digitalWrite(LED, !digitalRead(LED));
}
void LED_BLINK(){
    if(millis()-led_t >= 500){
        digitalWrite(LED, !digitalRead(LED));
        led_t = millis();
    }
}
//*****
unsigned long sertime;
int16_t input_roll, input_pitch;
int16_t roll_err;
int command[5];
int led_state_batt;
unsigned long led_timer_;

void LED_USE_SOLAR_PANEL (){
    if ( millis() - led_timer_ >= 500 ) {
        digitalWrite ( LED_PIN, led_state_batt );
        led_state_batt = !led_state_batt;
        led_timer_ = millis();
    }
}

```



```

void LED_BATERAY_USE_BLINK () {
    if ( millis() - led_timer_ >= 100 ) {
        digitalWrite ( LED_PIN, led_state_batt );
        led_state_batt = !led_state_batt;
        led_timer_ = millis();
    }
}

```

```

void LED_BATERAY_USE_OFF () {
    digitalWrite ( LED_PIN, HIGH );
}

```

```

void LED_SOLAR_USE_BLINK () {
    if ( millis() - led_timer_ >= 500 ) {
        digitalWrite ( LED_PIN, led_state_batt );
        led_state_batt = !led_state_batt;
        led_timer_ = millis();
    }
}

```

```

void LED_SOLAR_USE_OFF () {
    digitalWrite ( LEDSOLAR_PIN, HIGH );
}

```

```

int DATA_STATE = 0;
#define SERIAL_DEBUG

```

```

int16_t FLAP = 1500;

```

```
float kp_yaw = 5.0f;
float ki_yaw = 5.0f;
float kd_yaw = 2.5f;
```

```
float input_yaw = 0;
int pitch_err = 0;
int yaw_deff = 0;
int roll_deff = 0;
```

```
void loop() {
    ANALOG_READ ();
    MANAGE_POWER ();
    gps_task();
    getCompass();
    //*****
    //*****
    if ( radio.available()){
        led_success ();
        receiveData rxdata;
        radio.read(&rxdata, sizeof(rxdata));
        command[0] = rxdata.y;
        command[1] = map(rxdata.p,1000,2000,1000,2000);
        command[2] = map(rxdata.r,1000,2000,1000,2000);
        command[3] = rxdata.t;
        command[4] = rxdata.w ;
        if ( command [4] == 1000 ) {
            Kp = 1.0; Ki = 1.0; Kd = 0.001;
        } else if ( command [4] == 1500 ) {
            Kp = 2.4; Ki = 2.75; Kd = 0.10;
        } else if ( command [4] == 2000 ) {
            Kp = 5.4; Ki = 7.5; Kd = 0.25;
```

```

    }
    /*for ( int i =0; i <4; i++ ) {
        Serial.print(command[i]); Serial.print("\t");
    }
    Serial.println();*/
    //if( millis()-led_t >= 100){
        /*Serial.print("Yaw : ");
        Serial.print(yaw);
        Serial.print("\t pitch : ");
        Serial.print(pitch);
        Serial.print("\t roll : ");
        Serial.print(roll);
        Serial.print("\t"); */

        // led_t = millis();
    //}

    if ( rxdata.w == 1000 ) {
        USE_BATERAY = false;
    } else if ( rxdata.w == 2000 ) {
        USE_BATERAY = true;
    }
    input_yaw = yaw;
    roll_err = 0;
    pitch_err = 0;
    yaw_deff = 0;
    roll_deff = 0;
    yaw_deff = rxdata.y - 1500;
    error_pid[2] = map((int16_t)(input_yaw),-400,400,-500,500) -
yaw_deff;
    /*if( error_pid[2] <= -0.5 || error_pid[2] >= 0.5 ) {

```

```

P[2] = error_pid[2];
I[2] = I[2] + previous_I[2];
D[2] = error_pid[2]-previous_error_pid[2];
PID_value[2] = (kp_yaw*P[2]) + (ki_yaw*I[2]) + (kd_yaw*D[2]);
previous_I[2]= I[2];
previous_error_pid[2] = error_pid[2];
//PID_value[2]+=1500;
if ( PID_value[2] <= -400 ) PID_value[2] = -400;
if ( PID_value[2] >= 400 ) PID_value[2] = 400;
} else {
    error_pid[2] = 0;
    previous_I[2] = 0;
    previous_error_pid[2] = 0;*/
/*  PID_value[2] = rxdata.y - 1500;
}
/*
*/
if(!debug){
#ifdef SERIAL_DEBUG
    Serial.print("Radio::");
    Serial.print(rxdata.t);
    Serial.print("\t");
    Serial.print(rxdata.y);
    Serial.print("\t");
    Serial.print(rxdata.p);
    Serial.print("\t");
    Serial.print(rxdata.r);
    Serial.print("\t");
    Serial.print(rxdata.s);
    Serial.print("\t");
    Serial.print(rxdata.w);

```

```

Serial.print("\tlmu:");
Serial.print(compass);
Serial.print("\t");
Serial.print(yaw);
Serial.print("\tp: ");
Serial.print(Gyro_pitch_output);
Serial.print("\tr: ");
Serial.print(Gyro_roll_output);
Serial.print("\ter: ");
Serial.print(error_pid[0]);
Serial.print("\tep: ");
Serial.print(error_pid[1]);
Serial.print("\tlat: ");
Serial.print(flat);
Serial.print("\tlon:");
Serial.print(flon);
Serial.print("\tspeed:");
Serial.print(speed_);
Serial.print("\thdop");
Serial.print(hdop_);
Serial.print("\tsat:");
Serial.print(satelites,0);
Serial.print("\talt:");
Serial.println(altitude_);
Serial.print("\tv:");
Serial.print(bat_voltage);
Serial.print("\tc:");
Serial.println(current);
#endif
}
} else {
    if( debug){

```

```

if ( millis()- servertime >= 100 ) {
    txData tx;
    ///compass,yaw,pitch,roll  lat,lng,speed,sat,hdop,altitude
    if ( DATA_STATE == 0 ) {
        tx.data[0] = DATA_STATE;
        tx.data[1] = (int16_t)compass;
        tx.data[2] = (int16_t)yaw;
        tx.data[3] = (int16_t)pitch;
        tx.data[4] = (int16_t)roll;
        tx.data[5] = flat;
        tx.data[6] = flon;
        tx.data[7] = speed_;

        //tx.data[9] = hdop_;
        //tx.data[10] = altitude_;
        DATA_STATE = 1;
    } else if(DATA_STATE == 1){
        if ( solar_voltage < 0 ) solar_voltage = 0;
        if ( current < 0 ) current = 0;
        if ( current_charge < 0 ) current_charge = 0;
        if ( current_charge < 0 ) current_charge = 0;
        tx.data[0] = DATA_STATE;
        tx.data[1] = satellites;
        tx.data[2] = hdop_;
        tx.data[3] = altitude_;
        tx.data[4] = solar_voltage; //v source
        tx.data[5] = current; //i source
        tx.data[6] = current_charge;
        tx.data[7] = bat_voltage;
        DATA_STATE = 0;
    } else if ( DATA_STATE == 2 ) {
        tx.data[0] = DATA_STATE;

```

```

    tx.data[1] = command[0];
    tx.data[2] = command[1];
    tx.data[3] = command[2];
    tx.data[4] = command[3];
    tx.data[5] = command[4];
    tx.data[6] = 0;
    tx.data[7] = 0;
    DATA_STATE = 0;
}
//radio.stopListening();
radio.write(&tx,sizeof(tx));
//delay(1);
//radio.startListening();
#ifdef SERIAL_DEBUG
/*Serial.print("");
Serial.print(command[3]);
Serial.print("\t");
Serial.print(command[0]);
Serial.print("\t");
Serial.print(command[1]);
Serial.print("\t");
Serial.print(command[2]);
Serial.print("\t");
Serial.print(command[4]);
Serial.print("\tcompass : ");
Serial.print("");
Serial.print(compass);
Serial.print("\tyaw : ");
Serial.print(yaw);
Serial.print("\tpitch : ");
Serial.print(pitch);
Serial.print("\troll : : ");

```

```

Serial.print(roll);
Serial.print("\tlat : ");
/*Serial.print(error_pid[0]);
Serial.print("\tpe: ");
Serial.print(error_pid[1]);
Serial.print("\tlat: ");*/
Serial.print(flat);
Serial.print("\tlng : ");
Serial.print(flon);
Serial.print("\t speed : ");
Serial.print(speed_);
Serial.print("\thdop : ");
Serial.print(hdop_);
Serial.print("\tsat : ");
Serial.print(satelites);
Serial.print("\talt : ");
Serial.print(altitude_);
Serial.print("\tsolar v : ");
Serial.print(solar_voltage);
Serial.print("\tbat v : ");
Serial.print(bat_voltage);
Serial.print("\t solar i : ");
Serial.print(current);
Serial.print("\t charge i : ");
Serial.println(current_charge);
#endif
sertime = millis();
}
}
}

input_roll = (double)map ( (int)(roll*100),-3000,3000,1000,2000);
input_pitch = (double)map ( (int)(pitch*100),-3000,3000,1000,2000);

```



```

error_pid[0] = input_roll - command[2];
if( error_pid[0] <= -1 || error_pid[0] >= 1 ) {
    P[0] = error_pid[0];
    I[0] = I[0] + previous_I[0];
    D[0] = error_pid[0]-previous_error_pid[0];
    PID_value[0] = (Kp*P[0]) + (Ki*I[0]) + (Kd*D[0]);
    previous_I[0]= I[0];
    previous_error_pid[0] = error_pid[0];
    PID_value[0]+=1500;
    if ( PID_value[0] <= 1100 ) PID_value[0] = 1100;
    if ( PID_value[0] >= 1900 ) PID_value[0] = 1900;
    servo[0].writeMicroseconds(3000 - PID_value[0]); // + (yaw_deff/3));
    roll_deff = (PID_value[0] - 1500)/2.0;
}else {
    error_pid[0] = 0;
    previous_I[0] = 0;
    previous_error_pid[0] = 0;
    servo[0].writeMicroseconds(3000 - command[2]); // + (yaw_deff/3));
    roll_deff = (command[2] - 1500)/2.0;
}

```

```

error_pid[1] = command[1] - input_pitch;
if( error_pid[1] <= -1 || error_pid[1] >= 1 ) {
    P[1] = error_pid[1];
    I[1] = I[1] + previous_I[1];
    D[1] = error_pid[1]-previous_error_pid[1];
    PID_value[1] = (Kp*P[1]) + (Ki*I[1]) + (Kd*D[1]);
    previous_I[1]= I[1];
    previous_error_pid[1]=error_pid[1];
    PID_value[1]+=1500;
    if ( PID_value[1] < 1100 ) PID_value[1] = 1100;
    if ( PID_value[1] > 1900 ) PID_value[1] = 1900;
}

```

```

        servo[2].writeMicroseconds(3000 - PID_value[1]);
    } else {
        error_pid[1] = 0;
        previous_I[1] = 0;
        previous_error_pid[1] = 0;
        servo[2].writeMicroseconds(3000 - command[1]);
    }
    //servo[0].writeMicroseconds(FLAP);
    int yaw_value = constrain((command[0] - 1500 + roll_deff) +
1500,1200,1800); // using PID
    servo[3].writeMicroseconds((yaw_value));
    servo[4].writeMicroseconds(command[3]);
    //delay(1);
}

```

## BIOGRAFI



Nama : Teguh septi Asworo  
Pangkat/Korps/NRP : Serka KOM NRP 114919  
Alamat : Kalangan, RT/02 RW/02, Ds  
Wonosari, Kec. Ngombol,  
Kab. Purworejo, Jawa  
Tengah

Merupakan anak tunggal dari pasangan Bapak Sodi Asworo dengan Ibu Legiyem. Menikah dengan Dian Savitri pada tanggal 04 September 2015 dan dikaruniai dua anak bernama Thalita Sakhi Asworo dan Freya Farensa Asworo. Masa pendidikan dari SD sampai SMA di Purworejo. Pada tahun 2010 menyelesaikan Pendidikan Calon Bintara (Dikmaba) PK Angkatan XXIX di Surabaya. Dinas pertama di LANTAMAL II Padang pada tahun 2010 sampai tahun 2017. Pada bulan Desember tahun 2016 melaksanakan tugas belajar di Sekolah Tinggi Teknologi Angkatan Laut (STTAL) Surabaya, Program Studi Teknik Elektronika Angkatan ke XIII.