

# Sistem Navigasi Untuk Rover Tanpa Awak

## (Unmanned Rover Navigation System)

1<sup>st</sup> Gilang Purwana  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia  
gilangpurwana@student.telkomuniversity.ac.id

2<sup>nd</sup> Irwan Purwana  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia  
irwanp@telkomuniversity.ac.id

3<sup>rd</sup> Angga Rusdinar  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia  
anggarusdinar@telkomuniversity.ac.id

### Abstrak

*Unmanned Ground Vehicle (UGV)* merupakan perangkat mekanik yang dapat dioperasikan secara manual maupun otomatis memiliki kegunaan diantaranya untuk keperluan transportasi, keperluan logistic industry, aktivitas Search and Rescue (SAR) dan kepentingan militer untuk pengawasan, pengintaian bahkan pertempuran. Pada tulisan ini dijelaskan bagaimana merancang UGV yang dapat bergerak secara otomatis menggunakan metode navigasi *waypoint* dengan memanfaatkan beberapa sensor yang terdapat pada *flight control* seperti *Accelerometer*, GPS dan *compass* dapat menyelesaikan misi yang telah ditentukan. Pada penelitian ini, sistem kendali PID digunakan untuk steering dan *throttle* agar mendapatkan respon yang sesuai pada saat mode auto. Dengan menggunakan metoda Ziegler-Nichols didaatkan bahwa nilai konstanta  $P = 0.382$ ,  $I = 0.174$  dan  $D = 0.210$  untuk steering dan nilai konstanta  $P = 1.000$ ,  $I = 0.416$  dan  $D = 0.600$  untuk *throttle*. Dengan menggunakan nilai PID tersebut, penyimpangan rata-rata yang dihasilkan adalah 0.000010035% derajat latitude dan 0.000000074% derajat longitude atau jika dikonversikan ke dalam satuan meter pergerseran yang terjadi adalah sebanyak 0.385 meter atau 38.5 cm pada saat pengujian navigasi metode *waypoint*.

**Kata Kunci:** *Unmanned Ground Vehicle, PID, Mission Planner, Autonomous.*

### Abstract

*An unmanned Ground Vehicle (UGV) is a mechanical device that can be operated*

*manually or automatically for purposes of transportation, logistics industry, Search and Rescue activities, and military purposes for surveillance, even combat. In this final project, an Unmanned Ground Vehicle (UGV) is designed that can move automatically using the waypoint navigation method by utilizing several sensors contained in flight control such as Accelerometer, GPS, and compass to complete the specified mission. This study uses a PID control system for steering and throttle in order to get the appropriate response during auto mode, getting  $P = 0.382$ ,  $I = 0.174$  and  $D = 0.210$  for steering and getting  $P = 1000$ ,  $I = 0.416$  and  $D = 0.600$  for steering. throttles. Using the PID value, produces an average deviation of 0.000010035% degrees of latitude and 0.000000074% degrees of longitude when converted into meters, the shift that occurs is 0.385 meters or 38.5 cm when testing the waypoint navigation method.*

**Keywords:** *Unmanned Ground Vehicle, PID, Mission Planner, Autonomous.*

### I. PENDAHULUAN

Dalam era sakarang ini, teknologi semakin lama semakin pesat perkembanganya. Banyak sekali temuan-temuan dari para peneliti dan ilmuwan dunia yang semakin memudahkan dan bermanfaat bagi pekerjaan manusia. Salah satu teknologi yang menjadi fokus peneliti saat ini adalah teknologi autopilot telah banyak digunakan pada berbagai kendaraan otomatis seperti *Unmanned Areal Vehicle (UAV)*, *Unmanned Surfaces Vehicle (USV)*, *Unmanned Ground Vehicle (UGV)*[1]. Salah satu kendaraan otomatis yang banyak dipergunakan yaitu

UGV karena memiliki kegunaan diantaranya untuk keperluan transportasi, keperluan logistic industry, aktivitas *Search and Rescue* (SAR) dan kepentingan militer untuk pengawasan, pengintaian bahkan pertempuran. UGV adalah perangkat mekanik yang dapat dioperasikan secara manual maupun otomatis diatas permukaan tanah tanpa adanya kontak secara langsung oleh manusia[2]. Pada bidang militer UGV dapat diaplikasikan pada panser yang biasa digunakan militer untuk digunakan dimedan pertempuran. Panser merupakan kendaraan tempur yang sering digunakan militer dalam pertempuran darat yang membutuhkan kendaraan yang dapat menerjang berbagai medan, anti peluru serta memiliki sistem persenjataan yang cukup memadai. Panser varian anoa yang di-design secara spesifik untuk menjadi kendaraan logistik dalam pertempuran, kendaraan ini memiliki kemampuan untuk mengangkut bahan bakar, amunisi serta perlengkapan untuk perbaikan kendaraan lainnya[3]. Saat ini panser masih dioperasikan oleh manusia untuk menerjang medan tempur yang memiliki resiko sangat besar bagi personel militer baik cedera, cacat hingga resiko kematian.

Teknologi UGV menjadi peran utama dalam aplikasi di bidang militer, dalam pengaplikasiannya UGV dituntut agar bisa bergerak secara mandiri tanpa ada operator untuk pengoperasiannya agar mengurangi jatuhnya korban personel militer, sistem navigasi pada UGV menggunakan metode *waypoint* diimplementasikan dengan *compass* dengan melihat arah gerak UGV dapat mengikuti rute perjalanan dengan tingkat akurasi pada modul *compass*[4]. Navigasi *waypoint* adalah suatu metode untuk mengatur gerak dari satu posisi ke posisi yang dituju, dengan mengasumsikan setiap posisi dalam proses pergerakannya menjadi suatu titik dalam sistem koordinat tertentu (Latitude dan Longitude berdasarkan sistem koordinat bumi). Pada tugas akhir ini penulis akan membuat prototype UGV yang bisa bergerak secara otomatis sesuai dengan titik yang ditentukan menggunakan *Global Positioning System* (GPS) dan *Compass* dengan otomatis menggunakan *waypoint*.

## II. KAJIAN TEORI

### 2.1 Unmanned Ground Vehicle

*Unmanned Ground Vehicle* (UGV) adalah sebuah teknologi kendaraan yang bergerak tanpa awak pada permukaan tanah

dan dapat dikendalikan dari jarak jauh, UGV dapat bergerak secara otomatis menggunakan *Global Positioning System* (GPS) untuk berpindah dari satu titik ke titik lainnya menggunakan metode *waypoint* dan UGV ini juga dilengkapi dengan beberapa sensor lainnya sebagai komponen pendukung untuk UGV ini bisa bekerja sesuai dengan fungsinya.

### 2.2 Flight Controller

*Controller* merupakan komponen utama untuk mengatur jalannya semua komponen sehingga dapat bergerak sesuai dengan yang ditentukan, pada Tugas Akhir ini *controller* yang digunakan yaitu Pixhawk 2.4.8. Pada Pixhawk terdapat gabungan beberapa komponen dan bisa berfungsi untuk mengendalikan *Unmanned Ground Vehicle* (UGV). Pada Pixhawk ini juga sudah dilengkapi dengan sensor *gyroscope*, *accelerometer*, *magnetometer* yang mana sensor ini juga berguna untuk memantau keseimbangan, kecepatan dan arah pergerakan dari UGV ini, dan Pixhawk ini juga mendukung penggunaan *Compass*, *Global Positioning System* (GPS) dan *telemetry* yang bisa difungsikan sebagai sistem komunikasi untuk pembuatan *waypoint* UGV untuk bergerak secara otomatis dengan titik-titik yang telah ditentukan. Selain itu *controller* juga menyimpan informasi tugas yang akan dijalaninya yang diunggah dari mission planner oleh pengguna[5].

### 2.3 Global Positioning System (GPS)

*Global Positioning System* (GPS) merupakan sistem navigasi dengan memakai teknologi satelit yang bisa menerima sinyal dari satelit. Sistem ini memakai 24 satelit yang mengirimkan sinyal gelombang mikro ke bumi. Sinyal ini diterima oleh perlengkapan penerima (*receiver*) di permukaan, dimana GPS *receiver* ini hendak mengumpulkan data dari satelit GPS. Suatu GPS *receiver* wajib mengunci sinyal minimum 3 satelit buat menghitung posisi 2D (*latitude* serta *longitude*) serta *track* pergerakan. Bila GPS *receiver* bisa menerima 4 ataupun lebih satelit, hingga bisa menghitung posisi 3D (*latitude*, *longitude* serta *altitude*).

### 2.4 Ground Control Station (GCS)

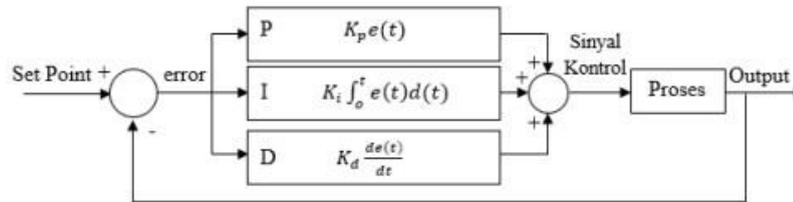
*Ground Control Station* (GCS) merupakan alat pengendali dari darat, yang mana GCS akan mengendalikan jalan dari

UGV, GSC memiliki komponen yang lengkap untuk mengendalikan USV dengan jarak jauh, semua parameter pada USV kan di kendalikan melalui GCS ini, baik itu dari arus listrik, kecepatan, kemiringan dan lain-lainnya, dan GCS berkomunikasi melalui USV menggunakan telemetry sebagai komunikasi untuk menampilkan keadaan dan posisi USV terkini, GCS juga mengatur jalannya UGV sesuai dengan waypoint yang ditentukan sehingga UGV bisa berjalan otomatis ke titik-titik yang telah ditentukan

dan pada Tugas akhir ini saya menggunakan *Mission Planner*.

### 2.5 Kendali PID

Kendali PID merupakan salah satu jenis metode kendali yang banyak digunakan. Selain itu sistem kendali ini mudah digabungkan dengan metode pengaturan yang lain seperti *Fuzzy* dan *Robust*. Sehingga akan memudahkan sistem ini menjadi suatu sistem pengatur yang semakin baik[6], [7].



Gambar 2.1 Kendali PID

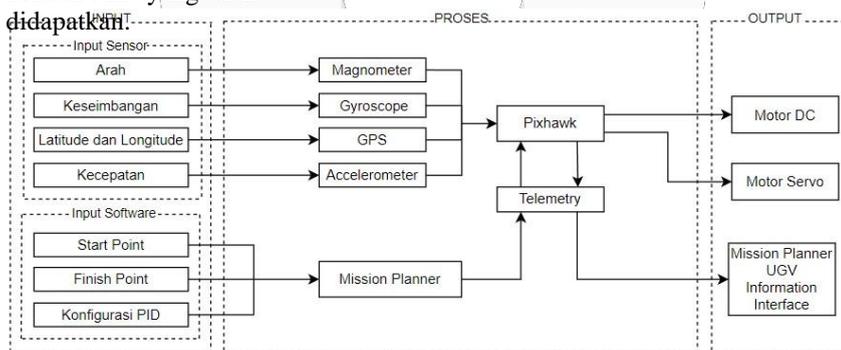
Dimana :

1. *Set Point* adalah nilai masukan dari sebuah sistem.
2. *Error* adalah selisih nilai dari sensor umpan balik dan nilai masukan.
3. *Sinyal Kontrol* adalah nilai ekuivalensi dari kendali P, kendali I dan kendali D.
4. *Output* adalah nilai yang dihasilkan dari digunakannya kendali PID yang telah didapatkan.

Kontroler PID terdiri dari 3 jenis cara pengaturan yang saling dikombinasikan, yaitu P—(Proportional) Controller, D (*Derivative*) Controller, dan I (*Integral*) Controller. Setiap jenis, memiliki kelebihan dan kekurangan masing-masing.

## 3. Perancangan Sistem

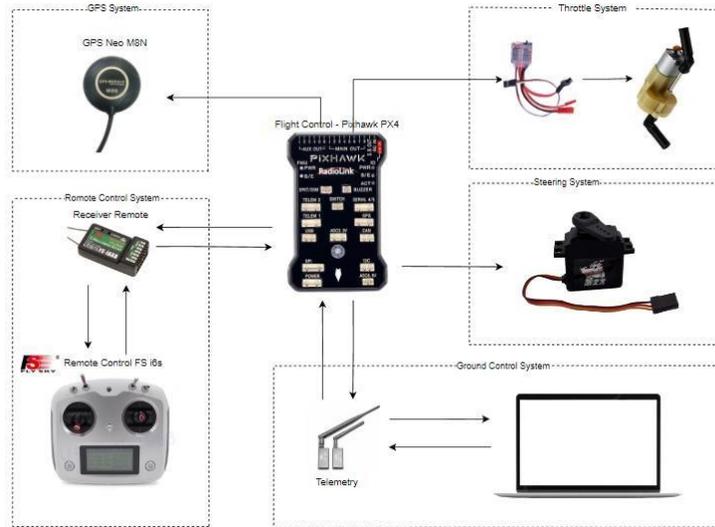
### 3.1 Desain Sistem



Pada Gambar 3.1. terdapat diagram blok sistem secara keseluruhan mengenai sistem otomasi pada sebuah UGV yang akan dikerjakan oleh penulis. Informasi yang didapatkan untuk menjalankan sistem otomasi waypoint diantaranya yaitu informasi yang berasal dari *software Mission Planner*, *Accelerometer*, *Gyroscope*, *Magnetometer* dan *Global Positioning System*

(GPS). Informasi tersebut akan diolah dan dipadukan oleh *controller* sehingga dapat memiliki output PWM yang sesuai untuk menggerakkan Motor DC dan Motor Servo agar berhasil menjalankan misi[7].

### 3.2 Desain Perangkat Keras

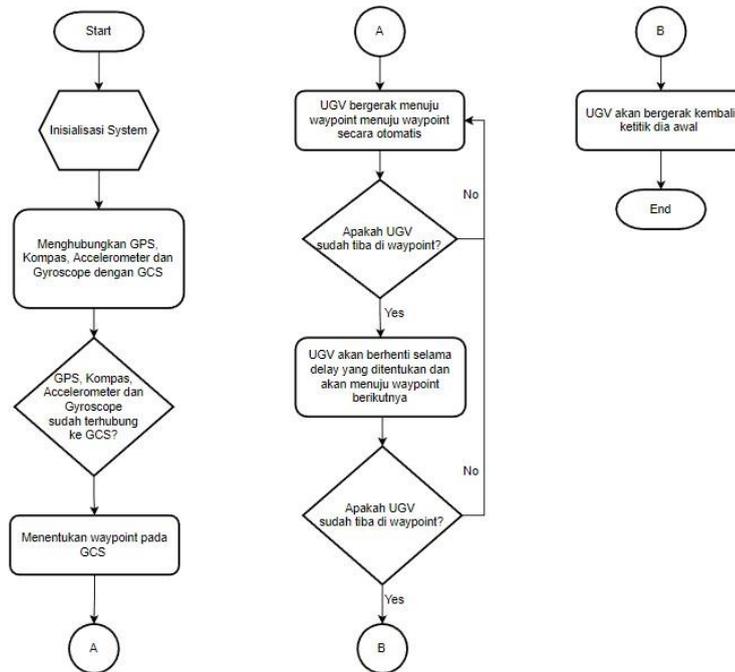


**Gambar 3.1** Desain Perangkat Keras

Desain perangkat keras pada penelitian ini menggunakan FlySky i6s sebagai *remote control* untuk memindah ke *mode auto* dan menggunakan *receiver* untuk menghubungkannya dengan *flight control*. *Flight control* yang digunakan adalah Pixhawk PX4 sebagai perangkat yang akan mengolah data dari sensor *Global Positioning System* (GPS), *Accelerometer*, *Gyroscope* dan *Manetometer* dan akan melakukan sistem kendali PID. Hasil dari data kendali PID akan mengatur Motor Servo sebagai arah belok

dan Motor DC sebagai motor penggerak UGV yang di kontrol menggunakan ESC Brushed pada saat berjalan otomatis. Pergerakan UGV juga dapat dilihat pada Ground Control Station dengan menggunakan aplikasi *Mission Planner* yang terhubung dengan flight control menggunakan telemetry sebagai sistem komunikasinya.

### 3.3 Desain Perangkat Lunak



**Gambar 3.2.** Diagram Alir

Perangkat lunak yang digunakan yang digunakan pada tugas akhir ini adalah *mission*

*planner*, *mission planner* merupakan aplikasi *open source* yang berfungsi sebagai *Ground Control Station* (GCS) yang biasanya

digunakan untuk *Unmanned Aerial Vehicle* (UAV), *Unmanned Surface Vehicle* (USV) dan dipakai juga untuk *Unmanned Ground Vehicle* (UGV). *Mission Planner* memiliki banyak fitur didalamnya yang bisa memberikan perintah pada UGV, selain itu

*mission planner* juga bisa memantau UGV, konfigurasi dan kalibrasi. Ada beberapa sensor yang harus dikalibrasi pada *mission planner* sebelum UGV dijalankan seperti

*Accelerometer*, *Compass* dan *Global Positioning System* (GPS). Selain itu *mission planner* memiliki sistem kendali PID untuk mengendalikan steering dan throttle agar pergerakan UGV lebih baik dan dapat menjalankan misi hingga selesai.

Pada Gambar 3.2. merupakan cara kerja sistem navigasi pada UGV yang akan berkomunikasi dengan *Ground Control*

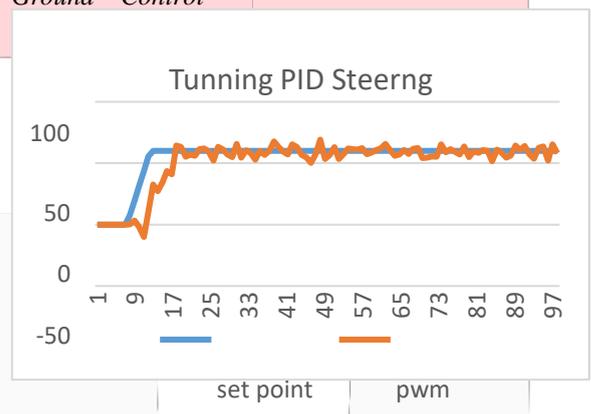
*Vehicle* (GCS) dengan menggunakan *telemetry* 433 MHz agar dapat mengontrol pergerakan rover saat bergerak secara otomatis.

### 3.4 Tuning Kendali PID

Dalam perancangan kendali PID diperlukan parameter  $K_p$ ,  $K_i$  dan  $K_d$ . untuk mendapatkan nilai masing masing parameter, penulis menggunakan metode Ziegler-Nichols 1 *tuning rules*.

Terdapat 2 nilai yang harus terpenuhi untuk melakukan *tuning* Ziegler-Nichols 1 *tuning rules* yaitu  $L$  (*delay time*) dan  $T$  (*time constant*). Setelah mendapatkan parameter  $L$  dan  $T$ , maka metode Ziegler-Nichols 1 *tuning rules* dapat digunakan untuk kendali PID.

#### 1. Tuning Steering



Gambar 3.3. Tuning PID Steering

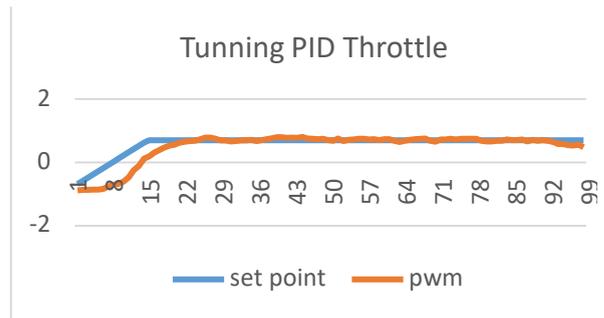
Saat melakukan tuning dapat dilihat pada Gambar 3.17. diperoleh nilai  $L = 1.1$  dan  $T=0.35$ . Setelah mendapatkan parameter  $L$  dan  $T$ , maka metode Ziegler-Nichols 1 tuning rules dapat digunakan, Dari perhitungan menggunakan rumus dari metode Ziegler-Nichols diperoleh nilai parameter PID yang dapat digunakan. Nilai parameter yang dihasilkan oleh rumus tersebut adalah sebagai berikut.

| Control Type | $K_p$ | $T_i$ | $T_d$ |
|--------------|-------|-------|-------|
| P            | 0.318 | 0     | 0     |
| PI           | 0.286 | 0.078 | 0     |
| PID          | 0.382 | 0.174 | 0.210 |

Nilai yang dihasilkan pada Tabel 3.1. akan diuji pada sistem untuk mengetahui nilai mana yang sesuai dan baik dalam merespon *setpoint* yang ditentukan.

Tabel 3.1. Tuning PID Steering

#### 2. Tuning Throttle



Gambar 3.4. Tuning PiD Throttle

Saat melakukan *tuning* dapat dilihat pada Gambar 3.18. diperoleh nilai  $L = 1.2$  dan  $T=1$ . Setelah mendapatkan parameter  $L$  dan  $T$ , maka metode Ziegler-Nichols 1 *tuning rules* dapat digunakan. Dari perhitungan menggunakan rumus dari metode Ziegler-Nichols diperoleh nilai parameter PID yang dapat digunakan. Nilai parameter yang dihasilkan oleh rumus tersebut adalah sebagai berikut.

|     |       |       |       |
|-----|-------|-------|-------|
| PI  | 0.750 | 0.188 | 0     |
| PID | 1.000 | 0.416 | 0.600 |

Nilai yang dihasilkan pada Tabel 3.2. akan diuji pada sistem untuk mengetahui nilai mana yang sesuai dan baik dalam merespon setpoint yang ditentukan.

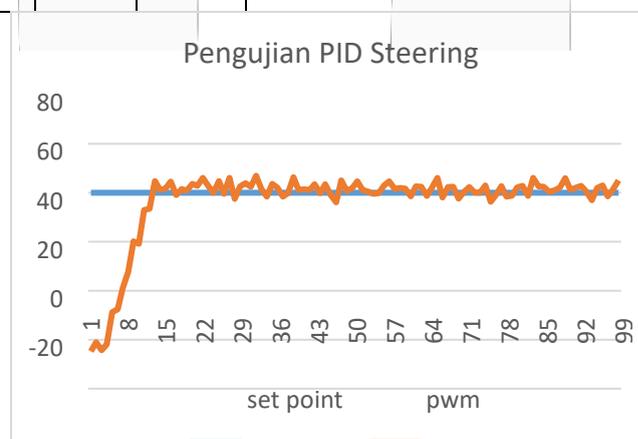
Tabel 3.2. Tuning PID Throttle

| Control Type | $K_p$ | $T_i$ | $T_d$ |
|--------------|-------|-------|-------|
| P            | 0.833 | 0     | 0     |

#### IV. HASIL DAN PEMBAHASAN

##### 4.1. Pengujian Parameter PID pada Steering

Pengujian ini dengan parameter PID Ziegler-Nichols 1 Tuning Rules. Metode ini menghasilkan parameter nilai  $P$  nilai  $P = 0.382$ ,  $I = 0.174$  dan  $D = 0.210$  respon motor servo untuk *steering* yang dikendalikan oleh kendali PID.

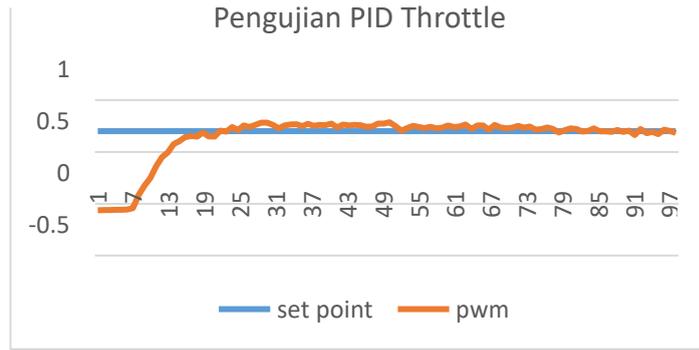


Gambar 4.1. Pengujian PID Steering

Hasil yang ditunjukkan pada Gambar 4.1. dapat dilihat bahwa respon motor servo merespon setpoint dengan lebih lama, ditunjukkan dengan risetime. Overshoot yang dihasilkan menjadi berkurang, Akan tetapi error steady statenya membesar.

##### 4.2 Pengujian Parameter PID pada Throttle

Pengujian ini dengan parameter PID Ziegler-Nichols 1 Tuning Rules. Metode ini menghasilkan parameter nilai  $P$  nilai  $P = 1.000$ ,  $I = 0.416$  dan  $D = 0.600$  respon Motor DC untuk throttle yang dikendalikan oleh kendali PID.



**Gambar 4.2.** Pengujian PID Throttle

Pada Gambar 4.2. dapat dilihat bahwa respon sistem menjadi lebih baik. Ditunjukkan dengan risetime dan overshoot yang menurun drastis dari nilai PI sebelumnya, Error steady statenya tidak terlalu tinggi dan sistem telah mendapat respon yang sesuai dengan setpoint yang telah ditentukan.

Positioning System (GPS). Persamaan ini menghitung jarak terdekat antar dua titik dalam permukaan lengkung dengan memanfaatkan nilai *latitude* (Nilai Y) dan *longitude* (Nilai X) dari titik tersebut, perhitungan ini penting untuk penggunaan navigasi. Persamaan Haversin adalah sebagai berikut :

**4.3 Pengujian Akurasi GPS terhadap GCS**  
Dalam Pengujian *Global Positioning*

System (GPS) penulis mengukur pergeseran jarak saat pada saat *Unmanned Ground Vehicle* diam penulis menggunakan *kennedy measuring wheel*, diukur ulang menggunakan *google earth* dan untuk mengetahui jarak antar satu koordinat dan koordinat lainnya pada sistem proyeksi geografis (*derajat decimal / decimal degrees/DD*) ini menggunakan persamaan Haversin dengan menggunakan pengambilan sample dari *DataFlash Log Global*

$$\begin{aligned}
 &= \sin^2\left(\frac{\Delta\phi}{2}\right) \\
 &+ \cos\phi_1 \cos\phi_2 \sin^2\left(\frac{\Delta\lambda}{2}\right) \\
 \phi &= 2 \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos\phi_1 \cos\phi_2 \sin^2\left(\frac{\Delta\lambda}{2}\right)}\right) \\
 \phi &= R \cdot \phi \\
 \phi &= \\
 \lambda &= \\
 R &= \frac{R \cdot \phi}{\phi}
 \end{aligned}
 \tag{3.1}$$

**Tabel 4.1.** Akurasi *Global Positioning System*

| Akurasi GPS |            |             |            |             |                  |
|-------------|------------|-------------|------------|-------------|------------------|
| No          | Lat GCS    | Long GCS    | Lat Asli   | Long Asli   | penyimpangan (m) |
| 1           | -6.9761188 | 107.6303508 | -6.9761316 | 107.6303569 | 1.574501153      |
| 2           | -6.9760813 | 107.6303376 | -6.9760859 | 107.6303576 | 2.265919015      |
| 3           | -6.9760374 | 107.6303416 | -6.9760308 | 107.6303514 | 1.307110774      |
| 4           | -6.9759642 | 107.6303302 | -6.9759579 | 107.6303443 | 1.706641089      |
| 5           | -6.9758865 | 107.6304321 | -6.9758727 | 107.6304318 | 1.534845893      |
| 6           | -6.9758647 | 107.6304013 | -6.9758628 | 107.6303931 | 0.929379854      |
| 7           | -6.9757868 | 107.6304623 | -6.9757831 | 107.6304782 | 1.802492048      |
| 8           | -6.9757848 | 107.6303262 | -6.9757769 | 107.6303168 | 1.35942988       |
| 9           | -6.9758734 | 107.6302638 | -6.9758607 | 107.6302529 | 1.855146765      |
| 10          | -6.9758154 | 107.6301941 | -6.9758303 | 107.6301945 | 1.657391116      |
| 11          | -6.9756717 | 107.6303168 | -6.9756658 | 107.6303217 | 0.850228755      |
| 12          | -6.9754673 | 107.6302852 | -6.9754714 | 107.6302752 | 1.194167942      |

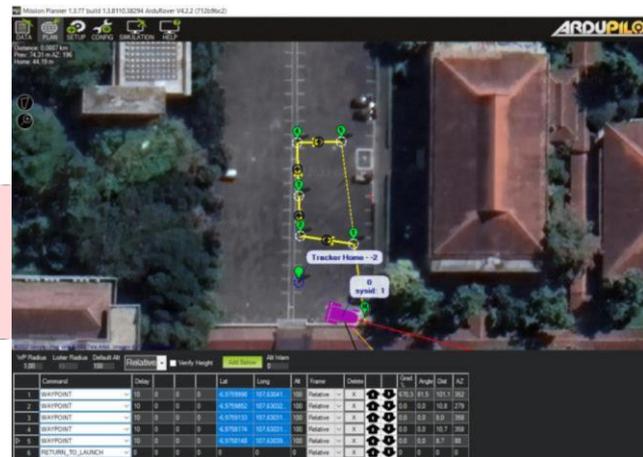
|    |            |             |            |             |             |
|----|------------|-------------|------------|-------------|-------------|
| 13 | -6.9754627 | 107.6303563 | -6.9754527 | 107.6303416 | 1.966932438 |
| 14 | -6.9755186 | 107.6304428 | -6.9755047 | 107.6304401 | 1.57407462  |
| 15 | -6.9755672 | 107.6303469 | -6.9755583 | 107.6303474 | 0.991171508 |

|                          |            |             |            |             |             |
|--------------------------|------------|-------------|------------|-------------|-------------|
| 16                       | -6.9756561 | 107.6303852 | -6.9756501 | 107.6303956 | 1.327671404 |
| 17                       | -6.9760391 | 107.6304562 | -6.9760272 | 107.6304648 | 1.628459656 |
| 18                       | -6.9761549 | 107.6304697 | -6.9761503 | 107.6304536 | 1.849134434 |
| 19                       | -6.9761442 | 107.6303892 | -6.9761576 | 107.6303811 | 1.737638163 |
| 20                       | -6.9761655 | 107.6303268 | -6.9761623 | 107.6303129 | 1.574888795 |
| 21                       | -6.9761396 | 107.6302457 | -6.9761456 | 107.6302559 | 1.30863256  |
| 22                       | -6.9761722 | 107.6302122 | -6.9761729 | 107.6302266 | 1.590153913 |
| Rata - Rata penyimpangan |            |             |            |             | 1.526636899 |

Dari Tabel 4.1. dapat dilihat hasil pengujian diatas dengan menggunakan GPS Ublox N 8M yang dipasang pada UGV memiliki rata – rata penyimpangan sebesar 1.52 m, nilai penyimpangan tersebut akan menentukan dalam penggambaran waypoint pada *Mision Planner* agar dapat menghindari kemungkinan tabrakan pada saat berjalan secara otomatis.

#### 4.4 Pengujian waypoint dengan menggunakan PID

Pengujian ini bertujuan untuk memperlihatkan seberapa tepat *Unmanned Ground Vehicle (UGV)* dapat mengikuti *waypoint* yang telah ditentukan. Pengujian ini dilakukan dilapangan Gedung Serba Guna (GSG) Telkom University, pengujian akan membanding *latitude* dan *longitude* yang sudah ditentukan pada *Mision Planner* dengan *longitude* dan *latitude* yang dilewati oleh UGV. Dari data yang diambil dapat dilihat seberapa besar penyimpangan yang terjadi pada UGV pada saat bergerak secara otomatis.



Gambar 4.3. Implementasi waypoint

Pada pengujian ini pengujian akan mengoperasikan UGV dengan parameter nilai  $P = 0.286$ ,  $I = 0.078$  dan  $D = 0$  pada *Steering* dan parameter nilai  $P = 1.000$ ,  $I = 0.416$  dan  $D = 0.600$  pada *Throttle*. Setelah desain *waypoint* upload misi akan dieksekusi, pengujian akan mengamati perilaku UGV pada tampilan *mission planner*.

Tabel 4.2. Pengujian PID *Steering* dan PID *Throttle*

| Lat target               | Long Target | Lat Dilalui | Long Dilalui | penyimpangan (m) |
|--------------------------|-------------|-------------|--------------|------------------|
| -6.9759998               | 107.6304173 | -6.9760017  | 107.6304164  | 0.233457563      |
| -6.9759852               | 107.6303208 | -6.9759831  | 107.6303208  | 0.233509149      |
| -6.9759133               | 107.6303181 | -6.9759156  | 107.6303141  | 0.51021335       |
| -6.9758174               | 107.6303154 | -6.9758208  | 107.630316   | 0.383818565      |
| -6.9758148               | 107.6303945 | -6.9758128  | 107.6303992  | 0.56440737       |
| Rata - rata penyimpangan |             |             |              | 0.385081199      |

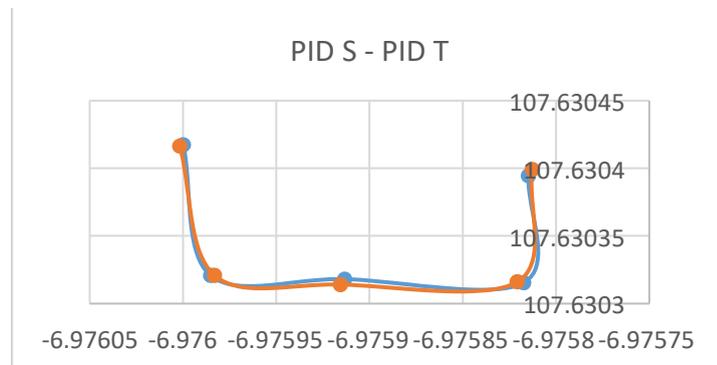
Pada saat menggunakan parameter PID *Steering* dan PID *Throttle* menyebabkan pergerakan UGV menjadi sangat agresif dan susah dikendalikan pada steering, hal ini disebabkan karena nilai parameter PID

*Steering* yang selalu beresilasi setelah melewati setpoint yang ditentukan dan mendapatkan rata-rata kecepatan 0,44 meter/s.

Dari Tabel 4.2. diatas dapat kita bandingkan seberapa besar simpangan yang

terjadi pada saat pengujian dilakukan. Pada pengujian ini UGV mendapatkan penyimpangan rata-rata paling kecil dari pengujian sebelumnya, UGV mendapatkan penyimpangan rata-rata sebanyak

0.000010035 derajat latitude dan 0.000000074 derajat longitude jika dikonversikan ke dalam satuan meter pergeseran yang terjadi sebanyak 0.385 meter atau 38.5 cm.



**Gambar 4.4.** Peta Penyebarat Pengujian PID *Steering* dan PID *Throttle*

Gambar 4.17. menunjukkan posisi koordinat pergerakan UGV pada saat menajalakan misi secara otomatis dengan menggunakan pameter PID *steering* dan PID *throttle*.

0.000010035% derajat *latitude* dan 0.000000074% derajat *longitude* jika dikonversikan ke dalam satuan meter menggunakan persamaan Harves pergeseran yang terjadi sebanyak 0.385 meter atau 38.5 cm.

## 5. Kesimpulan

Berdasarkan dari perancangan, implementasi, pengujian, dan analisa yang dilakukan pada *Unmanned Ground Vehicle* (UGV) dengan metode kendali PID, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Telah dilakukan perancangan kendali *steering* pada *Unmanned Ground Vehicle* (UGV) menggunakan metode kendali PID dengan nilai parameter  $K_p = 0.382$ ,  $K_i = 0.174$  dan  $K_d = 0.210$ .
2. Telah dilakukan perancangan kendali *throttle* pada *Unmanned Ground Vehicle* (UGV) menggunakan metode kendali PID dengan nilai parameter  $K_p = 1.000$ ,  $K_i = 0.416$  dan  $K_d = 0.600$ .
3. GPS yang digunakan pada *Unmanned Ground Vehicle* (UGV) menggunakan Ublox Neo 8M, mendapatkan rata-rata penyimpangan sebesar 1.52 meter setelah dikonversi menggunakan persamaan Harves pada saat *Unmanned Ground Vehicle* (UGV) bergerak secara *autonomous*.
4. Pengujian Waypoint dengan penerapan kendali PID *Steering* dan PID *Throttle* mengalami penyimpangan rata-rata penyimpangan rata-rata sebanyak

## Daftar Pustaka:

- [1] A. F. Anto and T. Sukardiyono, "Prototype Autonomous Rover Pembersih Sampah Pantai menggunakan ArduPilot," *Elinvo (Electronics, Informatics, and Vocational Education)*, vol. 4, no. 2, pp. 202–209, Dec. 2019, doi: 10.21831/elinvo.v4i2.28793.
- [2] J. Rahimatullah, N. Rachman, S. Muda, I. Fahmi, and Z. Akbari, "Rancang Bangun Autonomous Robot Tank dengan Metode Waypoint Berbasis Raspberry Pi The Design of Autonomous Robot Tank with Waypoint Method Based on Raspberry Pi," *TELKA*, vol. 6, no. 1, pp. 29–39, 2020.
- [3] S. M. Yusviva and I. N. Sutantra, "Analisis Pengaruh Parameter Operasional Terhadap Perilaku Belok dan Stabilitas Pada Panser Anoa 6X6 APC," *Jurnal Teknik ITS*, vol. 6, no. 2, Sep. 2017, doi: 10.12962/j23373539.v6i2.27948.

- [4] Alfiandy, M. T. Suprayogi, Nurwulan F, and M. Pfiis, "Iot (Internet Of Things) Navigasi Drone Berdasarkan Waypoint Iot (Internet Of Things) Navigation Drone Based On Waypoint."
- [5] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "The Pixhawk Open-Source Computer Vision Framework For Mavs," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXVIII-1/C22, pp. 13–18, Sep. 2012, doi: 10.5194/isprsarchives-XXXVIII-1-C22-13-2011.
- [6] B. Sampurno, A. Abdurrakhman, and H. S. Had, "Sistem Kendali PID pada Pengendalian Suhu untuk Kestabilan Proses Pemanasan Minuman Sari Jagung," in *Seminar Nasional Kontrol, Instrumentasi dan Otomasi (SNIKO) 2015*, 2016, p. 242. doi: 10.5614/sniko.2015.34.
- [7] Y. S. Bambang Setyono, "Rancang Bangun Sistem Transmisi, Kemudi, Dan Pengereman Mobil Listrik 'Semut Abang,'" *Institut Teknologi Adhi Tama Surabaya*.

