

Rancang Bangun *GPS Robot Tracker* dan Penerapan Algoritma *Proportional Integral Derivative* pada Sistem Gerak Robot Beroda

Hastaman^{*1}, Yulianto², Eka Puji Widiyanto³

^{1,2,3}STMIK Global Informatika MDP Jl. Rajawali No.14 Palembang

^{1,2,3}PS Teknik Informatika STMIK Global Informatika MDP;

³PS Teknik Komputer AMIK MDP

e-mail: ^{*1}hastamanchan@gmail.com, ²yuliantomdp@ymail.com, ³ekapujiw2002@mdp.ac.id

Abstrak

Robot telah menjadi kebutuhan vital dalam masyarakat modern, sehingga diperlukan adanya peningkatan kemampuan robot dalam hal sistem kecerdasan dan kendalinya. Dalam perancangan dan pembuatan robot ini digunakan metode prototyping yaitu membuat bagian per bagian robot secara terpisah kemudian disatukan menjadi satu kesatuan sistem robot yang digerakan dengan motor dan roda. Adapun bagian-bagian yang dimaksud adalah penggerak, power supply, kompas, sensor jarak, GPS modul receiver, mikrokontroler, driver motor dan lain-lain. Hasil dari skripsi ini adalah terciptanya robot yang memiliki kemampuan untuk melakukan tracking atau bergerak ke titik tujuan secara otomatis dengan dipandu oleh data koordinat satelit yang didapat melalui GPS modul receiver yang dihitung secara matematis oleh robot sehingga menghasilkan nilai arah pergerakan yang menuju ke titik tujuan sehingga robot bisa mengetahui arah pergerakannya. Robot memiliki tingkat presisi sebesar 2,5 m, rata-rata error perhitungan bearing sebesar 5.60, rata-rata selisih dari titik tujuan sebesar 2.325 m dan rata-rata waktu yang dibutuhkan untuk sampai ketitik tujuan sebesar 81.75 detik pada jarak tempuh sekitar 20 m.

Kata kunci—Robot, GPS,Tracking, Prototyping

Abstract

In this paper designed and built a robot that can perform automatic tracking from point of origin to point of destination is autonomous. In the design and manufacture of robotic prototyping method that is used to make parts of the robot in parts separately and then combined into a single unit system with a motor-driven robot and the wheel. As for the parts in question is a motor, power supply, compass, proximity sensor, GPS receiver module, microcontroller, motor drivers and others. The results of this thesis is the creation of a robot that has the ability to track or move to a point of destination are automatically guided by the coordinate data obtained through GPS satellite receiver module is calculated mathematically by robots so as to produce the value of the direction of movement toward the destination point so that the robot could determine the direction of movement. The robot has a precision level of 2.5 m, an average error of 5.60 bearing calculations, the average difference from the point of destination at 2,325 m and the average time it takes to get to point the goal of 81.75 seconds at a distance of about 20 m.

Keywords—Robot, GPS,Tracking, Prototyping

1. PENDAHULUAN

Robot merupakan suatu mesin yang dapat melakukan pekerjaan yang telah diprogramkan dengan meniru gerakan-gerakan makhluk hidup, contohnya manusia. Robot juga terbagi menjadi beberapa kategori, berdasarkan cara robot bergerak robot menjadi 2 macam yaitu *wheeled robot* dan *legged robot*. Semakin hari tuntutan terhadap kinerja robot semakin banyak, robot semakin diharapkan bisa lebih dan terus lebih canggih, contohnya *wheel robot* yang pada awalnya digerakan manusia sekarang dituntut untuk dapat bergerak sendiri secara otomatis, berdasarkan tuntutan ini maka penulis tertarik untuk mengkaji, meneliti, merancang dan membangun sebuah robot beroda otomatis yang dapat bergerak secara otomatis ketempat-tempat yang telah ditentukan sebelumnya, untuk bisa melakukan hal tersebut robot beroda ini haruslah dipadukan dengan teknologi lain, yaitu GPS (*Global Positioning System*).

2. DASAR TEORI

2.1 Algoritma PID (*Proportional Integral Derivate*)

Algoritma PID adalah algoritma kontrol dasar, "*PID adalah singkatan dari "proporsional, integral, turunan."Ketiga istilah menggambarkan elemen dasar dari sebuah kontroler PID. Masing-masing elemen ini melakukan tugas dan berbeda memiliki efek yang berbeda pada fungsi sistem*" [1]. Algoritma PID sendiri sebenarnya merupakan gabungan dari beberapa aksi kontrol yaitu aksi kontrol proporsional ditambah integral ditambah turunan hal ini dapat dilihat pada persamaan awal dari PID itu sendiri [2] :

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (1)$$

Dimana u adalah kontrol variable dan e adalah kontrol error. kontrol variabel merupakan hasil penambahan dari 3 bagian yaitu P (*Propotional error*), I (*integral Error*) dan D (*derivative error*), dan parameter-parameter adalah K berfungsi sebagai parameter propotional, $\frac{K}{T_i}$ sebagai parameter integral dan KT_d sebagai parameter *derivative*[2].

2.2 GPS (*Global Positioning System*)

GPS adalah sistem untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan (*synchronization*) sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke bumi. Sinyal ini diterima oleh alat penerima di permukaan dan digunakan untuk menentukan letak, kecepatan, arah dan waktu. Sistem yang serupa dengan GPS antara lain GLONASS Rusia, Galileo Uni Eropa, IRNSS India [3].

GPS *Tracker* atau sering disebut GPS *Tracking* adalah teknologi AVL (*Automated Vehicle Locater*) yang memungkinkan pengguna untuk melacak posisi kendaraan, armada ataupun mobil dalam keadaan *Real-Time*.GPS *Tracking* memanfaatkan kombinasi teknologi GSM dan GPS untuk menentukan koordinat sebuah objek, lalu menerjemahkannya dalam bentuk peta digital [3].

Pada tiap-tiap baris memiliki beragam informasi yang tersedia mulai dari posisi, kecepatan, ketinggian, waktu, tanggal dan lain lain. Baris kode yang dipakai dalam pengambilan informasi dalam skripsi ini adalah baris kode dengan kode depan \$GPRMC, yang kemudian didekoding dengan cara sebagai berikut :

\$GP	First code for Global Positioning
RMC	Recommended Minimum sentence C
123519	Fix taken at 12:35:19 UTC
A	Status A=active or V=Void.

4807.038,N	Latitude 48 deg 07.038' N
01131.000,E	Longitude 11 deg 31.000' E
022.4	Speed over the ground in knots
084.4	Track angle in degrees True
230394	Date - 23rd of March 1994
003.1,W	Magnetic Variation
*6A	The checksum data, always begins with *

Data-data tersebut dipisah kan oleh tanda koma ",", yang akan dijadikan sebagai pembatas tiap-tiap data. Data-data yang telah selesai didecoding, kemudian akan digunakan sebagai acuan pada pergerakan robot. Koneksi dari Modul *Receiver* ke mikrokontroler, penulis menggunakan komunikasi serial dengan *baudrate* 38400. *Baudrate* adalah kecepatan komunikasi data pada sistem komunikasi serial [4].



Gambar 1 Modul GPS U-BLOX LEA-6H [5]

2.3 AVRATMega32

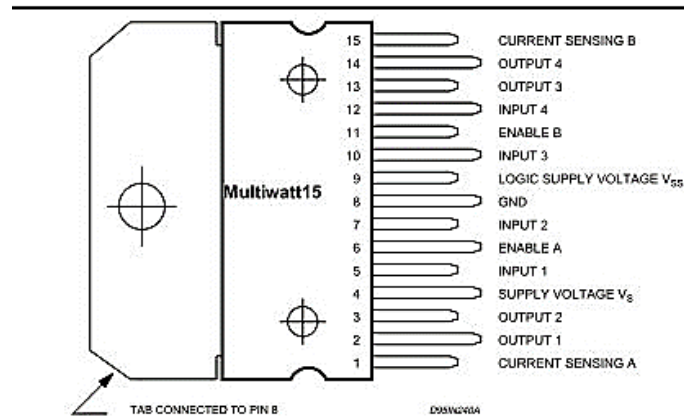
Setiap pin pada ATMega32 memiliki fungsi masing-masing, pada dasar nya dalam 40 pin pada ATMega32 terdapat 32 pin standar I/O yang bisa digunakan untuk fungsi-fungsi yang lain nya seperti pengatur PWM untuk *speed* motor dc, kemudian untuk membaca data *analog* yang disebut sebagai ADC (*Analog to Digital Converter*) dan masih banyak lagi kegunaan yang lebih spesifik.

2.4 BASCOM AVR

BASCOM AVR adalah *tools* yang digunakan untuk *programming* robot. Dengan menggunakan bahasa *basic* penulis akan menyetikkan kode-kode tertentu yang bertujuan untuk memerintahkan robot untuk melakukan sesuatu, seperti membaca sensor, bergerak kearah tertentu. Setiap bahasa pemrograman mempunyai standar penulisan program. Menurut[7] konstruksi dari program bahasa *BASIC* memiliki stuktur dan aturan yang sudah baku.

2.5 Driver Motor L298

Untuk pengendali kecepatan dan arah putaran motor digunakan *Driver Motor* L298, *Driver Motor* ini memiliki 2 masukan tegangan yaitu masukan tegangan ke IC dan masukan tegangan ke motor. *Driver Motor* L298 juga dapat digunakan untuk mengatur 2 Motor DC, Gambar 2 adalah gambar dan data pin *Driver Motor* L298.



Gambar 2 Pin Driver Motor L298 [8]

3. METODE PENELITIAN

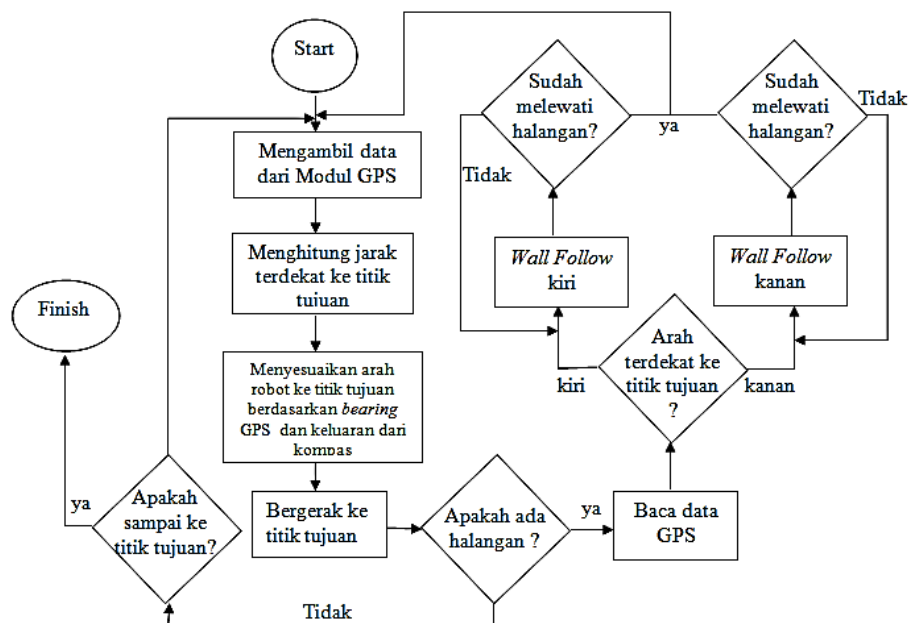
Pada bab ini akan dijelaskan tentang lingkungan pembangunan aplikasi, tahap-tahap metodologi yang digunakan penulis dalam membangun aplikasi, struktur data yang digunakan, dan *flowchart* program.

3.1 Spesifikasi Kebutuhan Software dan Hardware

Spesifikasi *software* dan *hardware* sangat penting ketika kita ingin menggunakan atau membeli barang-barang yang berkaitan dengan dua hal tersebut, karena dengan *software* dan *sardware* yang mendukung, maka kinerja dari alat yang dirancang dapat optimal.

3.2 Rancang Bangun Software

Dalam sub bab ini akan dijelaskan kerangka cara berpikir yang telah ditanamkan ke mikrokontroler robot. Secara garis besar diagram alir dari kegiatan robot dapat dilihat pada Gambar 3.



Gambar 3 Diagram Alir Robot

3.3 Penerapan Algoritma PID

Setelah dilakukan rasionalisasi persamaan ke kode program *basic*, pada bagian ini akan dijelaskan bagaimana implementasi dari perhitungan PID itu sendiri. Berikut kutipan dari kode program yang diimplementasikan.

```

Flagsensor = Get_pingkanan()

Kp = Kpeprom
Ki = Kieprom
Kd = Kdeprom
Kecepatan = Kecepataneprom
Setpoint = Setpointeprom

Pv = Pingkanan
Error = Setpoint - Pv
P = Kp * Error
D1 = Kd * 10
I = Error / 10
I = I * 10           'pengali 10 adalah dt'
I1 = Ki * I
D2 = Error - Last_error
D = D1 * D2
'D = D / 10'           'persepuluh adalah dt'
Last_error = Error

Pd = P + D
Pd = Pd + I1

Kecepatanakan = Kecepatan + Pd
Kecepatankiri = Kecepatan - Pd

If Kecepatanakan >= 120 Then
  Kecepatanakan = 100
  Rpwm = Kecepatanakan
  Set Kananmaju
  Reset Kananmundur
Elseif Kecepatanakan <= 0 Then
  Kecepatanakan = 255
  Rpwm = Kecepatanakan
  Reset Kananmaju
  Reset Kananmundur
Else
  Rpwm = Kecepatanakan
  Set Kananmaju
  Reset Kananmundur
End If

If Kecepatankiri >= 120 Then
  Kecepatankiri = 100
  Lpwm = Kecepatankiri
  Set Kirimaju
  Reset Kirimundur

Elseif Kecepatankiri <= 0 Then
  Kecepatankiri = 255

```

```

    Lpwm = Kecepatankiri
    Reset Kirimaju
    Reset Kirimundur
Else
    Lpwm = Kecepatankiri
    Set Kirimaju
    Reset Kirimundur
End If

```

Hasil dari perhitungan PID penulis implementasikan dalam penyesuaian kecepatan kanan dan kecepatan kiri, hal ini dilakukan dalam proses melewati halangan berupa *wall-follower* dan mengikuti sudut atau *degree follower*. *Set point* ditentukan sebagai nilai konstan yang menjadi batasan atau acuan pada proses. Masukan didapat dari sensor jarak HC-SR04 dan kompas CMPS03, *error* didapatkan dari selisih *set point* dan masukan dari sensor, hasil dari perhitungan dapat berupa nilai *negatif* dan *positif* yang akan menjadi panduan robot dalam meningkatkan atau mengurangi kecepatan baik pada motor dc kiri atau motor dc kanan. Batasan kecepatan pada robot adalah 90, dengan batasan kecepatan adalah 0-120, hal ini dilakukan agar tidak terjadi respon berlebihan dalam pengurangan atau penambahan kecepatan. Sedangkan untuk pembatasan nilai I, P dan D, tidak dibatasi oleh penulis, hanya dalam penerapannya dilakukan penyesuaian masukan nilai Kp, Ki dan Kd, terutama pada nilai Ki, di karenakan dalam penerapannya nilai yang akan di masukan berupa bilangan bulat positif, maka penulis membagi nilai Ki tersebut dengan konstanta 10 agar didapat nilai Ki yang lebih kecil sehingga pengaruh *error* nilai dari I dapat diminimalisir sesuai dengan asumsi rasionalisasi persamaan PID oleh penulis.

4. HASIL DAN PEMBAHASAN

Pengujian sistem dilakukan bertujuan untuk mendapatkan hasil perhitungan secara sistematis yang akan dibandingkan dengan nilai dari hasil pengukuran langsung.

4.1 Pengujian Lokasi

Pada pengujian ini yang akan diuji adalah kemampuan jangkauan signal GPS pada lokasi-lokasi tertentu yang secara garis besar dibagi menjadi 4 kategori. Kriteria dari masing-masing tipe lokasi pengujian adalah sebagai berikut :

1. *Outdoor* terbuka : suatu tempat yang terletak diluar bangunan dan tidak memiliki atap dan dinding penutup.
2. *Outdoor* tertutup : suatu tempat yang terletak diluar bangunan, memiliki atap dan tidak memiliki dinding penutup.
3. *Indoor* tertutup : suatu tempat yang terletak didalam bangunan, memiliki atap dan dinding penutup untuk menutupi setiap arah.
4. *Ground Floor / Basement* : suatu tempat yang terletak dibawah bangunan.

Hasil pengujian dapat dilihat pada Tabel 1.

Tabel 1 Pengujian Lokasi

Tipe Lokasi	Hasil
<i>Outdoor</i> terbuka	Bisa
<i>Outdoor</i> beratap	Bisa
<i>Indoor</i> tertutup	Gagal
<i>Ground Floor / Basement</i>	Gagal

4.2 Pengujian Pembacaan Kompas CMPS03

Pengujian pembacaan kompas CMPS03 adalah pembacaan sudut dari kompas CMPS03 dengan sensor kompas pada HP Android, kemudian dilakukan pengujian dalam melewati

halangan, pengujian pembacaan koordinat, pengujian perhitungan Bearing, pengujian *tracking* robot dengan input sudut manual dan pengujian *tracking* robot dengan GPS. Hasil dari pengujian dapat dilihat pada Tabel 2.

Tabel 2 Pengujian Pembacaan Kompas CMPS03

Robot	Android
332 ⁰	332 ⁰
264 ⁰	265 ⁰
222 ⁰	224 ⁰
220 ⁰	221 ⁰
219 ⁰	220 ⁰
156 ⁰	160 ⁰
119 ⁰	125 ⁰
120 ⁰	123 ⁰
121 ⁰	120 ⁰
14 ⁰	5 ⁰
15 ⁰	5 ⁰

4.3 Pengujian dalam Melewati Halangan

Pada pengujian ini yang akan diuji adalah kemampuan *tracking* robot, apakah robot dapat mencapai titik tujuan dengan diletakkan beberapa halangan. Hasilnya dapat dilihat pada Tabel 3.

Tabel 3 Pengujian Melewati Halangan

Jumlah Halangan	Waktu	Selisih
0	1 menit	1,8m
1	1 menit 25 detik	3m
2	1 menit 30 detik	3m
Rata-rata	78,3 detik	2,6 m

4.4 Pengujian Pembacaan Koordinat

Pada pengujian ini yang akan diuji adalah sensitifitas dari GPS dengan pembandingan berupa GPS dari HP Android. Hasil dari pengujiannya dapat dilihat pada Tabel 4.

Tabel 4 Pengujian Sensitifitas GPS

Robot		Android	
<i>Longitude</i>	<i>Latitude</i>	<i>Longitude</i>	<i>Latitude</i>
104,78115	-3,0221	104,781	-3,022
104,78134	-3,0222	104,781	-3,022
104,78158	-3,0221	104,782	-3,022
104,78106	-3,0222	104,781	-3,022
104,78124	-3,0221	104,781	-3,022
104,78114	-3,0221	104,782	-3,022

4.5 Pengujian Perhitungan Bearing

Pada pengujian ini yang akan dibandingkan adalah nilai bearing yang didapat dari perhitungan robot dan perhitungan *bearing* yang dilakukan dengan menggunakan HP Android. Hasilnya dapat dilihat pada Tabel 5.

Tabel 5 Pengujian Perhitungan *Bearing*

<i>Range</i>	Robot	Android
3 m	266	266,3419
6 m	275	266,9612
12 m	247	253,8617
20 m	245	237,8392
Rata-rata	258,25	256,251

4.6 Pengujian Tracking Robot GPS dengan Input Sudut Manual

Pada pengujian ini yang akan diuji adalah kemampuan robot dalam melakukan tracking ke titik tujuan dengan sudut yang telah ditentukan sebelumnya. Hasilnya dapat dilihat pada Tabel 6.

Tabel 6 Pengujian Tracking Robot dengan Input Sudut Manual

Arah Awal	Selisih	Waktu
Ke arah titik tujuan	1.5 m	1 menit 19 detik
90 derajat kanan	1.5m	1 menit 29 detik
90 derajat kiri	1.6m	1 menit 13 detik
180 derajat	1.2m	1 menit 18 detik
Rata-rata	1.45 m	79.75 detik

4.7 Pengujian Tracking Robot dengan GPS

Pada pengujian ini yang akan diuji adalah kemampuan *tracking* robot ke titik tujuan dengan menggunakan GPS, perhitungan *bearing* otomatis yang telah ditanamkan sebelumnya ke otak robot.

Tabel 7 Pengujian Tracking Robot dengan GPS

Arah Awal	Selisih	Waktu
Kearah titik tujuan	1.5m	1 menit 15 detik
90 derajat kanan	1.5m	1 menit 15 detik
90 derajat kiri	2.3 m	1 menit 17 detik
180 derajat	4m	1 menit 40 detik
rata-rata	2.325m	81.75 detik

5. KESIMPULAN

Dari penelitian yang dilakukan diperoleh kesimpulan :

1. GPS Robot *Tracker* dapat melakukan *Tracking* ke titik tujuan dengan rata-rata *error* atau selisih dari titik tujuan 2,325 m, rata-rata waktu 81,75 detik pada kondisi *outdoor*/tidak beratap dan rata-rata *error* 2,6 m dari titik tujuan dan rata-rata waktu 78,3 detik saat dalam kondisi *outdoor* beratap dan diberikan halangan.
2. Rata-rata perhitungan *bearing* dari robot yang dibandingkan dengan perhitungan *bearing* di Android pun tidak memiliki perbedaan yang signifikan yaitu dengan rata-rata *error* 5,6 derajat.
3. Sensor kompas CMPS03 memiliki tingkat presisi sebesar $3,4^{\circ}$ dibandingkan dengan sensor kompas pada Android.

6. SARAN

Dalam pengembangan lebih lanjut diharapkan pengembang dapat menambahkan titik tujuan lebih dari 1 dan juga menambahkan pemancar radio pada robot agar pengguna dapat melihat alur pergerakan robot pada GUI sistem yang telah dibuat juga oleh pengembang.

DAFTAR PUSTAKA

- [1] Wescot, Tim 2000, *PID without a PhD*. Embedded Systems Programming : New Delhi
- [2] Astrom, Karl dan Tore Haggglund 2002, *PID Controllers: Theory, Design, ATuning 2nd*. Instrument Society of America: New York.
- [3] Drive, Stewart 2007 *GPS The First Global Satellite Navigation System*. Trimble Navigation Limited : California
- [4] Axelson, Jan 2007, *Serial Port Complete Second Edition*. Lakeview Research LLC : Chinook
- [5] Module U-BLOX LEA-6H Datasheet, <http://www.terraelectronica.ru>, diakses pada tanggal 14 Agustus 2013 jam 13:00 WIB.
- [6] Atmega32 Datasheet, <http://www.atmel.com> diakses pada tanggal 16 Agustus 2013 jam 14:16 WIB.
- [7] Cuhnel, Claus 2001, *BASCOM Programming Microcontroller with Ease*. Universal Publisher : Washington
- [8] L298NDual H-Bridge motor Driver Datasheet, <http://www.tech.dmu.ac.uk>, diakses pada tanggal 14 Agustus 2013 jam 15:00 WIB.