

## DESIGN AND IMPLEMENTATION WHEELCHAIR FOR THE BLIND AND DISABLE BASED PID METHOD

Muhammad Alip Wijaksono<sup>1)</sup> Harianto<sup>2)</sup> Madha Christian Wibowo<sup>3)</sup>

S1 Sistem Komputer

STIMIK STIKOM Surabaya

Jl. Raya Kedung Baruk 93 Surabaya, 60298

Email : 1) [magic76man@gmail.com](mailto:magic76man@gmail.com), 2) [harianto@stikom.edu](mailto:harianto@stikom.edu), 3) [madha@stikom.edu](mailto:madha@stikom.edu)

**Abstract:** Various ways can be taken to get a location with long path or the short path. A system that can determine the best path that has a good navigation system as well as sensors to detect any supporting hitch so that the wheelchair does not hit a snag around and toward the location of the safe.

This research focuses on a wheelchair to avoid obstacles and this research develops previous research with title design and implementation wheelchair for the blind and disable based pid method toward the point where the goal of this research to develop a system using a wheelchair or implemented in real objects in real life and can move detect the presence of a hitch around the wheelchair. This compares wheelchair the distance between the obstacles to a wheelchair, and swerves to avoid an obstacle at a certain distance without bumping into an obstacle. Wheelchair systems are support systems or sub-systems that support the navigation system. The data from this study were taken from the articles from the internet and this research using the method of literature study analysis of the theories in the book. Based on data taken from literature and articles supporters.

Results from this study is the obstacle avoidance system on a wheelchair managed to avoid the obstacle and reach the desired location without bumping into objects or obstacles are nearby. The greater burden of the increasing rate of oscillation error and visible response to the slower motor to the load when approaching an obstacle.

**Keyword:** microcontroller, PID, wheelchair, sensor, ultrasound, obstacle avoidance, control system.

Kursi roda adalah alat bantu yang digunakan oleh orang yang mengalami kesulitan berjalan menggunakan kaki, baik dikarenakan oleh penyakit, cedera, maupun cacat. Biasa kursi roda didorong menggunakan bantuan orang lain, digerakan dengan menggunakan tangan, tapi kursi roda ini otomatis masih jarang ditemukan. Seiring dengan perkembangan teknologi yang sangat pesat, berbagai kursi roda telah di kembangkan khususnya dalam era informasi ini. Kursi roda dapat dikendalikan secara otomatis maupun

manual disesuaikan dengan kebutuhan manusia yang tentunya kursi roda bertujuan agar mempermudah kerja manusia sehari-hari dan membantu penyandang cacat.

Pada perkembangannya kursi roda penghindar halangan untuk penyandang tuna netra ini memiliki kemampuan untuk bernavigasi dengan baik dan presisi agar kursi roda ini dapat sampai pada tujuan yang diinginkan.

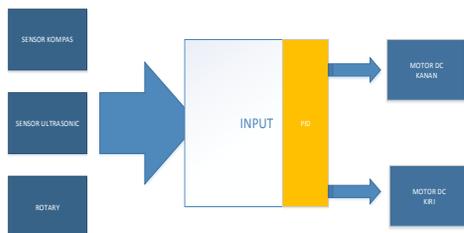
## METODE PENELITIAN

Studi kepustakaan berupa data-data literatur dari masing-masing komponen, informasi dari *internet*, dan konsep-konsep teoretis dari buku-buku penunjang.

Dari data-data yang diperoleh maka dilakukan perencanaan rangkaian perangkat keras. Dalam perangkat keras ini, dilakukan pengujian perangkat keras dengan program-program yang telah dibuat, pembuatan perangkat lunak adalah tahap selanjutnya. Terakhir adalah penggabungan perangkat keras dengan kerja perangkat lunak yang telah selesai dibuat.

## BLOK DIAGRAM SISTEM

Dari penelitian ini terdapat dua proses utama yang dijalankan, yaitu proses pada pendeteksian benda atau halangan disekitar kursi roda dan penentuan arah menghindari kursi roda untuk menuju titik tujuan. Diagram blok keseluruhan sistem dapat ditunjukkan seperti pada gambar 1.



Gambar 1. Blok diagram keseluruhan dari sistem.

Kursi roda akan mendeteksi benda di depan kursi roda ketika berjalan menuju arah tujuan, kursi roda juga akan mengukur jarak antara kursi roda dengan benda atau halangan yang berada di rute kursi roda dalam menuju arah tujuan. Setelah halangan atau benda tersebut dekat atau jaraknya sesuai dengan jarak hindar kursi roda yang aman kursi roda akan secara otomatis akan menghindarinya jika benda tersebut berada di rute kearah tujuan sehingga kursi roda tidak akan menabrak halangan tersebut. Sedang PID yang terdapat pada blok diagram keseluruhan ini untuk mengatur kecepatan roda kiri dan kanan untuk

menghindari halangan yang berada di sekitar kursi roda tersebut agar kursi roda tersebut berjalan sesuai dengan arah tujuan yang ditentukan.

## PERANCANGAN MEKANIKA

Mekanik kursi roda yang digunakan adalah kursi roda yang telah dimodifikasi. Rangkaian elektronika mulai dari rangkaian *microcontroller*, *driver motor*, sensor ultrasonik, sensor kompas digital, dan *rotary encoder* akan diletakkan pada *chassis* kursi roda dan memiliki dimensi panjang 114.5 cm (*centimeter*), lebar 73 cm, dan tinggi 85 cm seperti yang terdapat pada gambar 2.



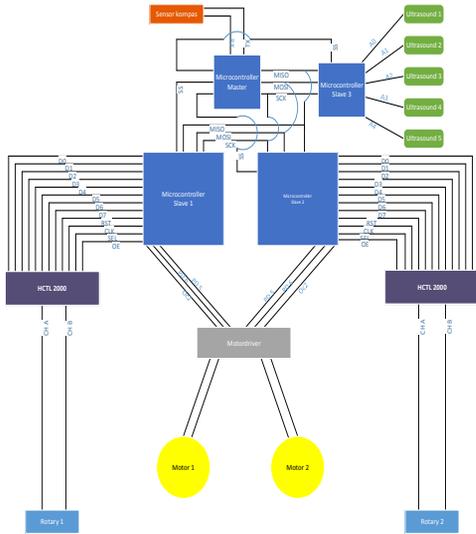
Gambar 2. Tampilan Keseluruhan Kursi roda.

## PERANCANGAN PERANGKAT KERAS

Perancangan perangkat keras pada sistem ini dilakukan berdasarkan blok diagram sistem keseluruhan yang terdapat pada gambar 1.

Dalam blok diagram pada gambar 1, *microcontroller* bertugas sebagai pemroses *data input* dari sensor ultrasonik, sensor kompas digital, dan *rotary encoder*. Sensor

kompass akan memberikan data berupa sudut arah mata angin dalam bentuk digital sebagai penunjuk arah tujuan. Sensor ultrasonik sebagai pengukur jarak antara kursi roda dengan benda disekitarnya. Rotary encoder akan memberikan data berupa *pulse* sebagai penghitung jarak tempuh kursi roda. Rangkaian perangkat keras dapat ditunjukkan seperti pada gambar 3.



Gambar 3. Rangkaian perangkat keras secara keseluruhan.

### KONFIGURASI PIN I/O (INPUT/OUTPUT) PADAPERANGKAT KERAS

Pada perancangan perangkat keras pada tugas akhir ini terdiri dari empat buah *microcontroller* ATmega8535 yaitu satu buah *master microcontroller* dan tiga buah *slave microcontroller*. Konfigurasi pin I/O pada *master microcontroller* dapat ditunjukkan seperti pada tabel 1.

Tabel 1. Konfigurasi pin I/O pada *master microcontroller*.

Pin I/O	Fungsi
Pin A6(ADC6)	Tombol start
Pin A7(ADC7)	Tombol <i>select</i> ruangan
Pin B0(XCK/T0)	<i>Slaveselect</i> motor kiri
Pin B1(T1)	<i>Slaveselect</i> motor kanan
Pin B2(INT2/AIN0)	<i>Slaveselect</i> ultrasound

Pin B5(OC0/AIN1)	SPI
Pin B6(MISO)	SPI
Pin B7(SCK)	SPI
Pin C0(SCL)	LCD
Pin C1(SDA)	LCD
Pin C2	LCD
Pin C3	LCD
Pin C4	LCD
Pin C5	LCD
Pin C6(TOSC1)	LCD
Pin C7	LCD
Pin D0(RXD) serial	<i>Reciver</i>
Pin D1(TXD) serial	<i>Transmitter</i>

Konfigurasi pin I/O pada *slave microcontroller* ke-1 dan ke-2 dapat ditunjukkan seperti pada tabel 2.

Tabel 2. Konfigurasi pin I/O pada *slave microcontroller* ke-1 dan ke-2.

Pin I/O	Fungsi
Pin A0 (ADC0)	Pin D0 HTCL
Pin A1(ADC1)	Pin D1 HTCL
Pin A2 (ADC2)	Pin D2 HTCL
Pin A3 (ADC3)	Pin D3 HTCL
Pin A4 (ADC4)	Pin D4 HTCL
Pin A5(ADC5)	Pin D5 HTCL
Pin A6 (ADC6)	Pin D6 HTCL
Pin A7 (ADC7)	Pin D7 HTCL
Pin B0 (XCK/T0)	<i>Reset</i> HTCL
Pin B1(T1)	<i>Clock</i> HTCL
Pin B2 (INT2/AIN0)	<i>Select</i> HTCL
Pin B3 (OC0/AIN1)	<i>Out enable</i> HTCL
Pin B4 (SS)	<i>SelectSlave</i> SPI
Pin B5 (MOSI)	SPI
Pin B6 (MISO)	SPI
Pin B7 (SCK)	SPI
Pin D5 (OC1A)	Maju motor driver
Pin D6 (ICP1)	Mundur motor driver
Pin D7 (OC2)	PWM motor driver

Konfigurasi pin I/O pada *slave microcontroller* ke-3 dapat ditunjukkan seperti pada tabel 3.

Tabel 3. Konfigurasi pin I/O pada *slave microcontroller* ke-3.

Pin I/O	Fungsi
Pin A0	Data pin <i>ultrasound</i>
Pin A1	Data pin <i>ultrasound</i>
Pin A2	Data pin <i>ultrasound</i>
Pin A3	Data pin <i>ultrasound</i>
Pin A4	Data pin <i>ultrasound</i>
Pin B5	SPI
Pin B6	SPI

Pin B7	SPI
--------	-----

## MODUL KOMPAS DIGITALCMPS10

Modul kompas digital ini berfungsi sebagai penunjuk sudut arah mata angin pada *mobile robot* dalam mencari koordinat tujuan yang akan dicapai. Mode yang digunakan pada untuk komunikasi antara kompas dengan mikrokontroller adalah komunikasi serial.

## ROTARY ENCODER

*Rotary encoder* ini berfungsi sebagai penghitung jarak tempuh kursi roda dalam perjalanan menuju koordinat tujuan, sehingga kursi roda dapat berhenti tepat pada koordinat tujuan. Dalam tugas akhir ini menggunakan *rotary encoder quadrature optic* dengan *output* sejumlah 72000 *pulse* dari tiap putaran roda pada kursi roda.

## DRIVER MOTOR DC 24V

Motor DC memiliki dua kabel terhubung, kabel pertama untuk *ground*, kabel kedua untuk *power supply* dengan besar tegangan sampai 24 volt. Kecepatan putar motor dapat dikendalikan dengan mengatur besar – kecilnya tegangan yang di masukkan, atau dapat juga dengan menggunakan teknik PWM (*Pulse Width Modulation*).

*Microcontroller* akan mengirimkan gelombang pulsa ke *driver motor* untuk mengatur kecepatan motor.

## KONTROLLER PID (PROPORTIONAL-INTEGRAL- DERIVATIVE)

Saat menghindari halangan pengendalian motor menggunakan kontroler PID untuk memperhalus, meningkatkan performa kontroler, dan mendapat respon sistem yang baik pada motor yang diimplementasi melalui program pada *master microcontroller* ATmega8535 sebagai pemroses. Di bawah ini merupakan formula PID yang sebelumnya dibahas pada

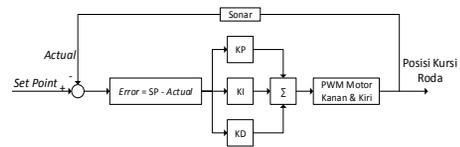
landasan teori PID pada BAB II untuk kemudian dijadikan algoritma PID yaitu sebagai berikut:

$$r_{mot} = r_{old} + SONAR\_KP * (sonar\_err - sonar\_e\_old) + SONAR\_KI * (sonar\_err - sonar\_e\_old) / 2 + SONAR\_KD * (sonar\_err - sonar\_e\_old + sonar\_e\_old2);$$

Dimana,

- $r_{mot}$  : hasil perhitungan PID
- $r_{old}$  : hasil perhitungan PID sebelumnya
- SONAR\_KP : konstanta P
- SONAR\_KI : konstanta I
- SONAR\_KD : konstanta D
- sonar\_err : error sekarang
- sonar\_e\_old : error sebelumnya
- sonar\_e\_old2 : error dua kali sebelumnya

Diagram blok sistem kontrol PID yang akan diimplementasikan pada tugas akhir ini dapat dilihat pada Gambar 4.

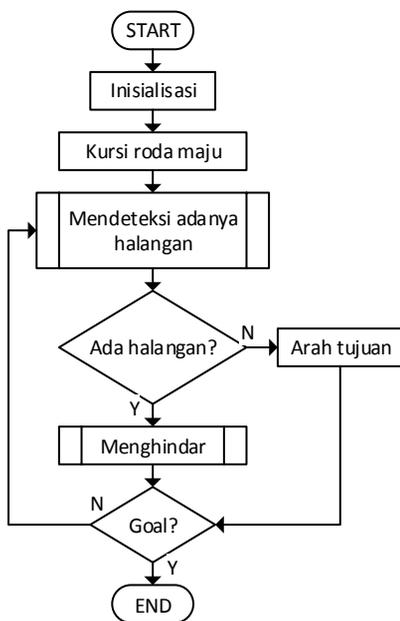


Gambar 4. Diagram blok implementasi PID.

Dari gambar 4 terlihat bahwa nilai *error* merupakan *set point* dikurangi *actual value*. *Set point* merupakan nilai jarak kursi roda dengan halangan yang diinginkan dan *actual value* merupakan nilai jarak kursi roda terdekat dengan halangan diantara nilai-nilai yang berasal dari kelima sensor ultrasonik. Setelah didapat nilai *error* kemudian *master microcontroller* melakukan proses perhitungan menggunakan kontroler PID untuk menghasilkan *output* berupa kecepatan roda kanan dan kiri agar menjauh dari halangan sehingga memperkecil *error* jarak kursi roda dengan halangan.

## PERANCANGAN PERANGKAT LUNAK

Perancangan perangkat lunak pada *microcontroller* ini bertujuan untuk mengirimkan data dari pembacaan sensor ultrasonik ke *minimum system* dan juga melakukan pengolahan data dengan melalui proses perhitungan mengubah data dari sensor ultrasonik menjadi jarak pada *minimum system*. *Minimum system* memperoleh data dari sensor ultrasonik yaitu berupa *pulse* yang dapat digunakan untuk penentuan jarak antara halangan dengan kursi roda, serta mengatur kecepatan putar roda melalui PWM yang dikirimkan ke *driver motor*. Perancangan perangkat lunak terbagi dalam beberapa program antara lain program motor DC, program membaca sensor, program perhitungan mengubah pulsa menjadi satuan jarak dalam centimeter, program penentuan arah menghindari kursi dan setelah kursi roda menghindari kursi roda akan menuju titik tujuan. Diagram alir perangkat lunak secara umum dapat dilihat pada gambar 5.



Gambar 5. Diagram alir program secara umum.

## MENGHITUNG JARAK TEMPUH KURSI RODA

Program untuk menghitung jarak tempuh kursi roda diimplementasikan pada *slave microcontroller* ke-1 dan ke-2. Menghitung jarak tempuh dilakukan untuk mengetahui posisi kursi roda pada saat itu, dengan diketahuinya posisi kursi roda maka jika kursi roda berada di titik *finish* maka kursi roda akan berhenti.

Menghitung jarak tempuh dilakukan dengan menghitung jumlah *pulse* yang dihasilkan oleh *rotary encoder* setelah dibandingkan dengan keliling roda pada kursi roda. Jika diketahui diameter roda pada kursi roda sepanjang 60 cm (*centimeter*) maka keliling roda dapat diketahui menggunakan rumus dasar keliling lingkaran seperti yang ditunjukkan sebagai berikut:

$$\text{Keliling} = \pi * d$$

$$\text{Keliling Roda} = \pi * 60 \text{ cm}$$

$$\text{Keliling Roda} = 188.496 \text{ cm}$$

Setelah diketahui keliling roda pada kursi roda, selanjutnya menghitung perbandingan antara jumlah *pulse* dan jarak tempuh. Diketahui dalam satu rotasi roda pada kursi roda akan menghasilkan sejumlah 72000 *pulse* maka dapat diketahui perbandingannya seperti yang ditunjukkan sebagai berikut:

$$72000 \text{ pulse} = 188.496 \text{ cm}$$

Jadi,

$$1 \text{ cm} = 381.972 \text{ pulse}$$

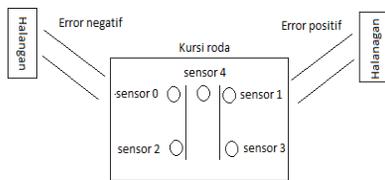
Pada pembahasan sebelumnya telah dijelaskan bahwa *pulse* yang dihasilkan oleh *rotary encoder* telah dihitung oleh HCTL-2000, adapun untuk mendapatkan jumlah *pulse* yang telah dihitung oleh HCTL-2000 dapat ditunjukkan melalui langkah-langkah sebagai berikut:

1. Set low pada pin SEL pada HCTL-2000
2. Atur delay 1  $\mu\text{s}$  (*micro second*)
3. Tunggu hingga pin CLK pada HCTL-2000 menjadi low
4. Dapatkan *high byte data* melalui pin D0 – D7 pada HCTL-2000
5. Set high pada pin SEL pada HCTL-2000
6. Atur delay 1  $\mu\text{s}$

7. Tunggu hingga *pin* CLK pada HCTL-2000 menjadi *low*
8. Dapatkan *low byte data* melalui *pin* D0 – D7 pada HCTL-2000

### MEDETEKSI HALANGAN DI DEPAN KURSI RODA

Program mendeteksi halangan di sekitar kursi roda diimplementasikan pada *slave microcontroller* ke-3. Ada lima sensor yang terdapat di kursi roda masing-masing sensor di hitung jaraknya dari kelima sensor tersebut, dihitung hingga menghasilkan *error* positif, *error* negatif, dan *error* 0 (nol). *error* positif adalah jika ada halangan di kanan kursi roda, *error* negatif adalah jika ada halangan di kiri kursi roda, dan *error* nol adalah jika tidak ada halangan sama sekali di kursi roda. Penempatan sensor *ultrasound* di kursi roda dan ilustrasi nilai *error* pada kursi roda dapat ditunjukkan seperti yang terdapat pada gambar 6.

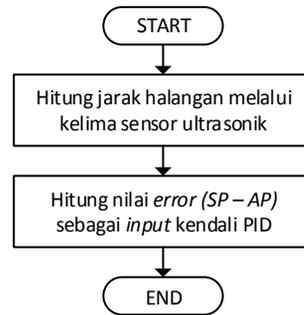


Gambar 6. Letak sensor pada kursi roda dan ilustrasi nilai *error* yang dihasilkan.

Jika diketahui *error* nol maka tidak terdapat halangan atau tidak ada halangan di depan kursi roda jadi kursi roda berjalan dengan lurus, jika diketahui *error* positif kursi roda akan berbelok ke kiri dikarenakan sensor yang di bagian kanan ada di kursi roda mengetahui ada halangan disebelah kanan, dan jika diketahui *error* negatif kursi roda akan berbelok ke kanan dikarenakan sensor yang ada di bagian kiri kursi roda mengetahui ada halangan di bagian kiri.

Diagram alir untuk mengetahui jarak antara kursi roda dan benda, setelah menghitung jarak kursi roda dengan benda, kursi roda akan pada titik tertentu dan menghindari yang menandakan kursi roda mendeteksi halangan dan akan menghitung dan menentukan arah menghindari kursi roda. Diagram alir pendeteksi halangan

pada kursi roda dapat ditunjukkan seperti yang terdapat pada gambar 7.



Gambar 7. Diagram alir mendeteksi halangan di depan kursi roda.

Pada gambar 7, diagram alir mendeteksi halangan dimulai dari mengetahui jarak antara kursi roda melalui kelima sensor ultrasonik yang terpasang dengan halangan kemudian dilanjutkan dengan menghitung *error* yang merupakan *input* dari kontroler PID pada kursi roda. Untuk mengetahui jarak antara kursi roda melalui sensor ultrasonik dengan halangan yaitu dengan men-trigger sensor ultrasonik, kemudian fungsi *timer* dijalankan untuk menghitung waktu yang diperlukan oleh sensor ultrasonik mulai dari pemancaran gelombang sonar dari sensor ultrasonik, pemantulan gelombang sonar, dan pengembalian gelombang sonar ke sensor ultrasonik. Maka dari hasil pemancaran, pemantulan, dan pengembalian dapat dihitung jarak halangan dari kursi roda tersebut yang dapat ditunjukkan menggunakan rumus dasar kecepatan sebagai berikut:

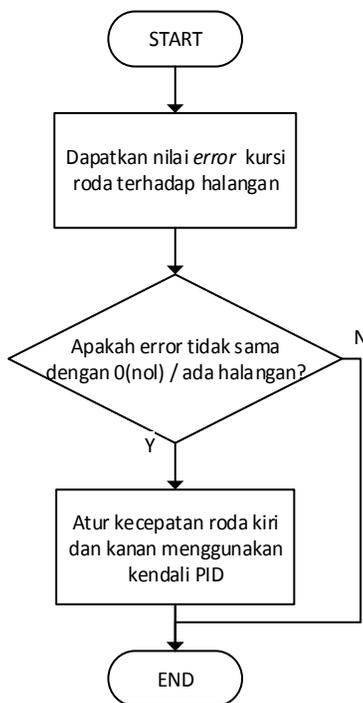
$$s_{kr} = v_{suara} * t_{sensor} / 2$$

Dimana,

- $s_{kr}$  : jarak kursi roda dengan halangan atau benda
- $v_{suara}$  : kecepatan suara (344 m/s)
- $t_{sensor}$  : waktu yang dibutuhkan gelombang suara mulai dari pemancaran oleh *transmitter* sensor ultrasonik hingga ditangkap kembali oleh *receiver* sensor ultrasonik

## MENENTUKAN ARAH MENGHINDAR KURSI RODA

Program penentu arah menghindar pada kursi roda diimplementasikan pada *master microcontroller*. Setelah diketahui nilai *error* kursi roda terhadap halangan, maka selanjutnya menentukan aksi kursi roda terhadap nilai *error* yang diketahui. Jika nilai *errors* sama dengan nol maka tidak terdapat halangan di sekitar kursi roda, aksi selanjutnya yaitu melanjutkan melaju menuju titik tujuan. Sebaliknya, jika nilai *error* tidak sama dengan nol maka terdapat halangan di sekitar kursi roda, aksi selanjutnya yaitu mengumpukan nilai *error* tersebut pada kendali PID yang akan menghasilkan *output* kecepatan roda kanan dan kiri untuk menghindari halangan tersebut. Diagram alir untuk menentukan arah menghindar kursi roda dapat ditunjukkan seperti yang terdapat pada gambar 8.



Gambar 8. Diagram alir penentuan menghindari halangan pada kursi roda.

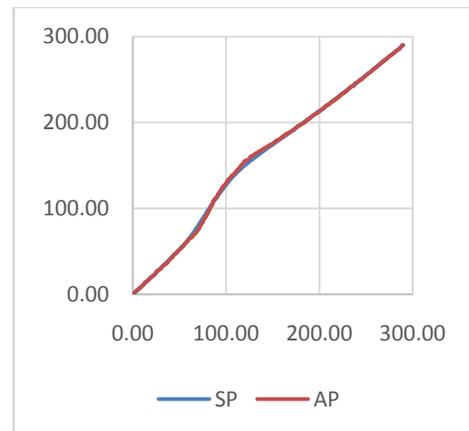
## PENGUJIAN SISTEM

Tujuan evaluasi sistem ini untuk mengetahui apakah sistem dapat berjalan sesuai dengan diharapkan yaitu kursi roda dapat melaju dari titik awal ke titik tujuan dan dapat menghindar jika terdapat halangan di jalur saat kursi roda sedang melaju.

## HASIL PENGUJIAN

Pengujian keseluruhan sistem penghindar halangan pada kursi roda ini menggunakan benda sebagai halangan yang diletakkan di rumah yang sudah disiapkan dan halangan diletakkan pada titik tertentu. Pada pengujian ini telah ditetapkan per 1 koordinat sama dengan 1 cm. Pada pengujian ini juga telah ditetapkan toleransi pencapaian titik tujuan sebesar 10 koordinat atau 10 cm ke arah x positif, x negatif, y positif, dan y negatif, maka saat mencapai daerah toleransi kursi roda akan berhenti melaju.

Pengujian kursi roda tanpa beban yang dimaksud yaitu pengujian pergerakan kursi roda dari titik awal (0,0) menuju titik tujuan (300,300) tanpa adanya penumpang. Hasil pengujian kursi roda tanpa beban berupa grafik dapat ditunjukkan seperti yang terdapat pada gambar 9.

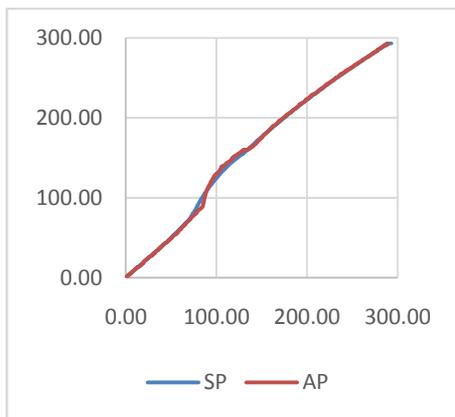


Gambar 9. Grafik hasil pengujian kursi roda tanpa beban.

Dari grafik pada gambar 9 dapat disimpulkan kursi roda dapat berhenti saat mencapai daerah tujuan. Dari keseluruhan data yang tercatat saat kursi roda melaju dari titik awal menuju titik tujuan tanpa adanya

beban memiliki rata-rata *error* terhadap sumbu  $x$  sebesar 16.11 koordinat atau 16.11 cm dan sumbu  $y$  sebesar 16.23 koordinat atau 16.23 cm. Penyimpangan terbesar terhadap sumbu  $x$  sebesar 22.95 koordinat atau 22.95 cm dan sumbu  $y$  sebesar 26.09 koordinat atau 26.09 cm.

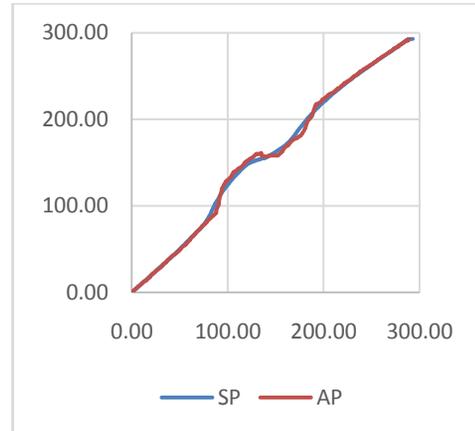
Pengujian kursi roda dengan beban 40 kg yang dimaksud yaitu pengujian pergerakan kursi roda dari titik awal (0,0) menuju titik tujuan (300,300) dengan adanya beban 40 kg yang diletakkan pada kursi penumpang. Hasil pengujian kursi roda tanpa beban berupa grafik dapat ditunjukkan seperti yang terdapat pada gambar 10.



Gambar 10. Grafik hasil pengujian kursi roda dengan beban 40 kg.

Dari grafik pada gambar 10 dapat disimpulkan kursi roda dapat berhenti saat mencapai daerah tujuan. Dari keseluruhan data yang tercatat saat kursi roda melaju dari titik awal menuju titik tujuan tanpa adanya beban memiliki rata-rata *error* terhadap sumbu  $x$  sebesar 13.64 koordinat atau 13.64 cm dan sumbu  $y$  sebesar 13.82 koordinat atau 13.82 cm. Penyimpangan terbesar terhadap sumbu  $x$  sebesar 21.46 koordinat atau 21.46 cm dan sumbu  $y$  sebesar 33.48 koordinat atau 33.48 cm.

Pengujian kursi roda dengan beban 80 kg yang dimaksud yaitu pengujian pergerakan kursi roda dari titik awal (0,0) menuju titik tujuan (300,300) dengan adanya beban 80 kg yang diletakkan pada kursi penumpang. Hasil pengujian kursi roda tanpa beban berupa grafik dapat ditunjukkan seperti yang terdapat pada gambar 11.



Gambar 11. Grafik hasil pengujian kursi roda dengan beban 80 kg.

Dari grafik pada gambar 11 dapat disimpulkan kursi roda dapat berhenti saat mencapai daerah tujuan. Dari keseluruhan data yang tercatat saat kursi roda melaju dari titik awal menuju titik tujuan tanpa adanya beban memiliki rata-rata *error* terhadap sumbu  $x$  sebesar 13.09 koordinat atau 13.09 cm dan sumbu  $y$  sebesar 13.26 koordinat atau 13.26 cm. Penyimpangan terbesar terhadap sumbu  $x$  sebesar 21.46 koordinat atau 21.46 cm dan sumbu  $y$  sebesar 30.90 koordinat atau 30.90 cm.

## KESIMPULAN

Dari hasil evaluasi keseluruhan kursi roda maka kursi roda mampu menuju ke koordinat tujuan dan mampu berhenti pada saat telah mencapai titik tujuan. Dengan menggunakan konstanta  $P=2$ , konstanta  $I=0.2$ , dan konstanta  $D=1.8$  pada pengendali PID untuk menghindari halangan pada kursi roda tanpa menggunakan beban didapat rata-rata *error* sumbu  $x$  sebesar 5.56 % dan sumbu  $y$  sebesar 5.60 %. Pada kursi roda dengan menggunakan beban 40 kg didapat rata-rata *error* sumbu  $x$  sebesar 4.66 % dan sumbu  $y$  sebesar 4.72 %. Pada kursi roda dengan menggunakan beban 80 kg didapat rata-rata *error* sumbu  $x$  sebesar 4.47 % dan sumbu  $y$  sebesar 4.52 %. Dari grafik hasil pengujian pergerakan kursi roda dapat disimpulkan semakin besar beban maka tingkat osilasi *error* semakin meningkat dan terlihat respon motor yang semakin lambat terhadap beban saat mendekati halangan.

## DAFTAR PUSTAKA

- ATMEL Corporation. 2005, ATmega8, (Online). (<http://www.atmel.com> , diakses 12 januari 2013 ).
- Braunl, Thomas. 2006. *Embedded Robotics Second Edition*. Perth, WA:Australia.
- Goge, Douglas W. ( 1995). *A Brief History of Unmanned Ground Vehicle (UGV) Development Efforts*, Unmanned System Magazine, United States of America.
- Hartanti, E. D. (2011). *Rancang Bangun Mobile Robot Penjejak Benda Bergerak Berbasis Pengendali PD (Propositional-Derivative) Menggunakan Mikrokontroler AVR Atmega8535*. Semarang: Jurusan Teknik Elektro, Fakultas Teknik Universitas diponegoro.
- INNOVATIVE ELECTRONICS. 2009. *AVR USB ISP*. (Online). ([http://www.innovativeelectronics.com/innovative\\_electronics/download\\_files/manual/Manual%20DT-HiQ%20AVR%20USB%20ISP.pdf](http://www.innovativeelectronics.com/innovative_electronics/download_files/manual/Manual%20DT-HiQ%20AVR%20USB%20ISP.pdf)). Diakses pada tanggal 21 januari 2013.
- Sanjaya, M. (2013, maret 17). robot penghindar halangan sensor ultrasonic ping))) (avoider) RA01. Retrieved from profesor bolabot:
- Vishay. (2004, 12 23). *16x2 Character LCD*. Diambil kembali dari [www.vishay.com](http://www.vishay.com).
- Winoto, Ardi.(2008). *Mikrokontroler AVR ATMEGA8/16/32/8535 dan Pemrogramannya dengan Bahasa C pada WinAVR*. Informatika. Bandung.
- Galih, Kusuma. (2014). *Rancang Bangun Sistem Navigasi Pada Differential Steering Mobile Robot*. Stikom. Surabaya
- Danang, Firmansyah. Achmad, Sugiharto. (2012). *Kendali PID Pada Robot Manual Menggunakan Komunikasi Nirkabel*. Stikom. Surabaya