

PERANCANGAN SELF-BALANCING TROLLEY DENGAN METODE PID

DESIGN OF SELF-BALANCING TROLLEY WITH PID CONTROLLER METHOD

Nashsharino Rudino¹, Porman Pangaribuan², Agung Surya Wibowo³

^{1,2,3}Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom

¹nashsharinorudino@gmail.com ²porman@telkomuniversity.ac.id ³agungsw@telkomuniversity.ac.id

Abstrak

Di dunia industri saat ini, untuk mempermudah beberapa pekerjaan, beberapa industri memanfaatkan troli sebagai salah satu alat untuk mengantar dan membawa barang dari satu tempat ke tempat lainnya. Sayangnya, penggunaan troli masih juga menghadapi kendala seperti jatuh atau tumpahnya barang bawaan ketika melewati jalan menanjak ataupun menurun. Maka dari itu, tujuan dari tugas akhir ini yaitu merancang *Self-balancing Trolley* yang merupakan troli pengangkut barang dimana troli ini dapat menyeimbangkan barang yang diletakkan di atas meja troli dalam kondisi jalan yang menurun maupun menanjak.

Sistem ini dirancang menggunakan sensor IMU sebagai sensor pembaca sudut, Arduino Uno sebagai mikrokontroler, dan motor dc sebagai aktuator. Sedangkan metode kontrol yang digunakan dalam merancang sistem ini adalah metode kontrol PID. *Output* kontrol PID merupakan nilai PWM yang akan berpengaruh pada pergerakan motor dc yang terhubung dengan meja troli.

Berdasarkan hasil implementasi kontrol yang telah dirancang, dibutuhkan konfigurasi nilai K_p , K_i , dan K_d yang berbeda-beda bergantung pada jalan yang dilalui dan beban yang dibawa. Pada kondisi statis, waktu yang dibutuhkan sistem untuk menggerakkan meja troli menuju keadaan *settle* (0°) dari kemiringan 10° adalah 4 detik. Sedangkan pada saat kondisi dinamis, waktu yang dibutuhkan sistem untuk menggerakkan meja troli ketika melewati tanjakan atau turunan dengan kemiringan 20° dan membawa beban 6 kg menuju keadaan *settle* (0°) adalah 8-9 detik. Dari hasil implementasi alat yang dibuat, sistem kerja troli sudah dapat mempertahankan kedataran dari meja troli saat melewati jalan menanjak atau menurun dengan sudut kemiringan -30° hingga 30° , walaupun masih terdapat *error steady state* pada sistem.

Kata kunci: *accelerometer*, *gyroscope*, K_d , K_i , K_p , PID, PWM, *self-balancing*.

Abstract

In today's industrial world, to facilitate some work, some industries use trolleys as one of the tools to deliver and carry goods from one place to another. However, the use of trolleys still face obstacles such as falling or spilling when passing uphill or downhill roads. Therefore, the purpose of this final project is to design a Self-balancing Trolley which is a freight trolley where it can balance the items placed on the trolley table when the trolley is on a uphill or downhill road.

The system designed using IMU sensors as angular reader sensors, Arduino Uno as microcontroller, and dc motor as an actuator. The control method that used in designing this system is the PID control method, the PID control output is the PWM value which will affect the movement of the dc motor that connected to the trolley table.

Based on the results of the implementation of the controls that have been designed, it is necessary to configure a different values of K_p , K_i , and K_d which depending on the road and the carried load. In static conditions, the time it takes for system to go to a stable state (0°) from 10° slopes is 4 seconds. While in dynamic conditions, the time it takes for system to move the trolley table when it passes a uphill or a downhill roads with a slope of 20° and carrying a 6 kg load toward the settle state (0°) is 8-9 seconds. From the results of the implementation made, the trolley's system has been able to maintain the angel of trolley's table as it passes uphill or downhill road with a slope angle of -30° to 30° , although there is still a steady state error in the system.

Keywords: *accelerometer*, *gyroscope*, K_d , K_i , K_p , PID, PWM, *self-balancing*.

1. Pendahuluan

Troli merupakan alat yang dapat memudahkan kita dalam membawa atau memindahkan barang dari suatu tempat ke tempat lainnya. Setiap kita mengunjungi Rumah makan, Hotel, Supermarket Rumah Sakit, bahkan Industri

sekalipun pasti akan menemukan troli dengan bentuk dan kegunaannya masing – masing. Troli dapat digunakan sebagai pengangkut makanan, koper, atau barang yang banyak ataupun berat.

Troli yang umumnya digunakan biasanya memiliki satu atau lebih rak untuk menampung berbagai bahan dasar stainless steel dan memiliki empat roda sebagai penggerak. Troli tersebut juga memiliki gagang sebagai tempat tangan untuk menggenggam dan mendorong troli. Penggunaan troli dimaksudkan untuk membantu pekerjaan membawa barang – barang sehingga dapat lebih mudah membawanya dan meningkatkan efisiensi waktu.

Namun, terkadang pekerjaan tersebut memiliki beberapa kendala seperti posisi antar tempat memiliki beda ketinggian. Sehingga troli tersebut harus bergerak menanjak ataupun menurun untuk tetap dapat melanjutkan perjalanan. Kendala tersebut terkadang menyebabkan makanan yang dibawa oleh troli bergerak kedepan dan kebelakang, bahkan memungkinkan makanan tersebut terjatuh dari troli.

Permasalahan tersebut dapat diminimalisir dengan self-balancing trolley, dimana rak dari troli tersebut akan selalu datar walaupun troli melewati jalan menanjak maupun menurun. Troli tersebut akan mengikuti sudut dari jalanan dan akan terus menyesuaikan agar rak selalu dalam posisi datar dengan menggunakan sensor IMU sebagai pendeteksi sudut dan motor DC sebagai penggerak dari raknya.

Dengan adanya self-balancing trolley, bergeser dan jatuhnya makanan bisa sangat diminimalisir. Tentunya, troli ini juga dapat lebih meningkatkan efisiensi waktu karena pengguna tidak perlu lagi khawatir lagi akan makanan ketika menghadapi jalan turunan ataupun tanjakan.

2. Dasar Teori

2.1. Troli

Troli merupakan alat yang dapat memudahkan kita dalam membawa atau memindahkan barang dari suatu tempat ke tempat lainnya. Troli yang ada di pasaran saat ini adalah troli fix, biasanya berbahan dasar besi, *stainless steel*, maupun kayu. Seperti pada Gambar II-1 yaitu troli yang umumnya digunakan.



Gambar 1. Contoh Troli

2.2. Kontrol PID

Kontrol PID merupakan salah satu dari beragam jenis metode control yang dapat digunakan. Namun, pada tugas akhir kali ini penulis akan menggunakan kontroler PID karena memiliki beberapa fungsi penting. Kontroler ini dapat memberikan umpan balik, memiliki kemampuan untuk menghilangkan *offset steady state* melalui tindakan integral, dan dapat mengantisipasi masa depan melalui aksi derivatif. [2] Kontroler PID terdiri dari tiga parameter, yaitu: P (Propotional), I (Integral), dan D (Derivative). Parameter P, I, dan D merupakan parameter yang dapat diatur dalam sistem yang akan dikontrol agar respon output sistem sesuai terhadap input sistem yang diharapkan. Karena dengan metode PID, tingkat kestabilan dan waktu untuk mencapai kestabilan akan menjadi lebih baik. Untuk persamaan PID kontinyu bisa dilihat pada persamaan (II-1) dan persamaan PID diskrit dapat dilihat pada persamaan (II-3). sebelum mendapatkan persamaan PID distrik, Persamaan PID kontinyu harus diubah terlebih dahulu menjadi persamaan PID transformasi *laplace* yaitu pada persamaan (II-2).

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \tag{II-1}$$

$$\frac{U(s)}{E(s)} = K_p + \frac{1}{s} + K_d s \tag{II-2}$$

$$\frac{U(z)}{E(z)} = K_p + \frac{K_i}{(1-z^{-1})} + K_d(1 - z^{-1}) \tag{II-3}$$

2.3. Inertial Mesurement Unit (IMU)

Agar meja troli dapat selalu datar tentunya dibutuhkan sensor yang dapat mengukur kedataran dari meja troli. Sensor yang banyak digunakan salah satunya adalah sensor IMU. IMU adalah singkatan dari *Inertial Mesurement Unit* yang merupakan unit tunggal dalam modul elektronik yang dapat mengumpulkan data kecepatan sudut dan percepatan linier. Terdapat beberapa jenis IMU, salah satunya adalah IMU strap-down dimana IMU mempertahankan 6-degree-of-freedom (DOF) yang memperkirakan gerakan yaitu posisi (x, y, z) dan orientasi (roll, pitch, yaw) [4] Sensor ini merupakan kombinasi dari *accelerometer* dan *gyroscope*. *Accelerometer* merupakan sensor yang digunakan untuk mengukur percepatan suatu objek sesuai dengan tiga sumbu x, y, z. Sedangkan *gyroscope* merupakan sensor yang digunakan untuk mendeteksi rotasi atau perputaran suatu objek berdasarkan gerakan.



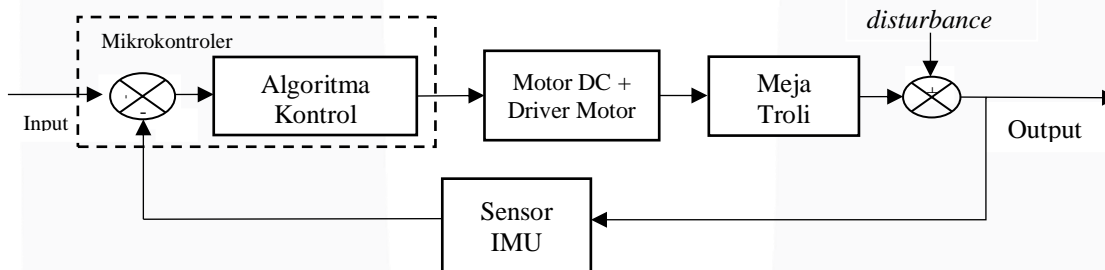
Gambar 2. Sensor IMU MPU6050

2.4. Motor DC

Dibutuhkan sebuah aktuator untuk dapat menggerakkan meja dari troli. Motor DC dapat digunakan sebagai penggerak dari meja troli agar meja dapat selalu datar. Motor listrik sendiri merupakan perangkat yang dapat merubah energi listrik menjadi energi mekanik. Sesuai dengan namanya, motor dc membutuhkan suplai tegangan arus DC pada kumparan medan.

3. Perancangan Sistem

Tugas akhir ini dirancang untuk dapat mempertahankan kedataran dari meja troli, apa bila posisi meja troli tidak datar maka sistem secara otomatis akan bekerja untuk mendatarkannya kembali.



Gambar 3. Diagram Blok Sistem

Gambar 3 di atas menunjukkan diagram blok sistem secara umum. *Input* dari sistem ini adalah *set point* yang akan ditentukan, nilainya merupakan keadaan meja troli saat datar yaitu 0° . Mikrokontroler bertugas sebagai otak dari sistem yang akan memproses semua input dan output dengan metode PID. *Aktuator* dari sistem ini adalah motor dc yang akan menggerakkan *plant* dari sistem. Terdapat *disturbance* yaitu keadaan jalan yang dilewati oleh troli. Meja troli merupakan *plant* dari sistem dan sensor IMU berfungsi sebagai *feedback* yang menghasilkan nilai sudut kemiringan dari meja troli. Pada perancangan ini, keluaran yang dihasilkan merupakan posisi meja troli yang datar.

3.1. Perancangan Perangkat Keras

self-balancing trolley dirancang menggunakan troli yang dijual di pasaran, dan di rubah bentuknya sedemikian rupa sehingga dapat mendukung sistem yang diinginkan.



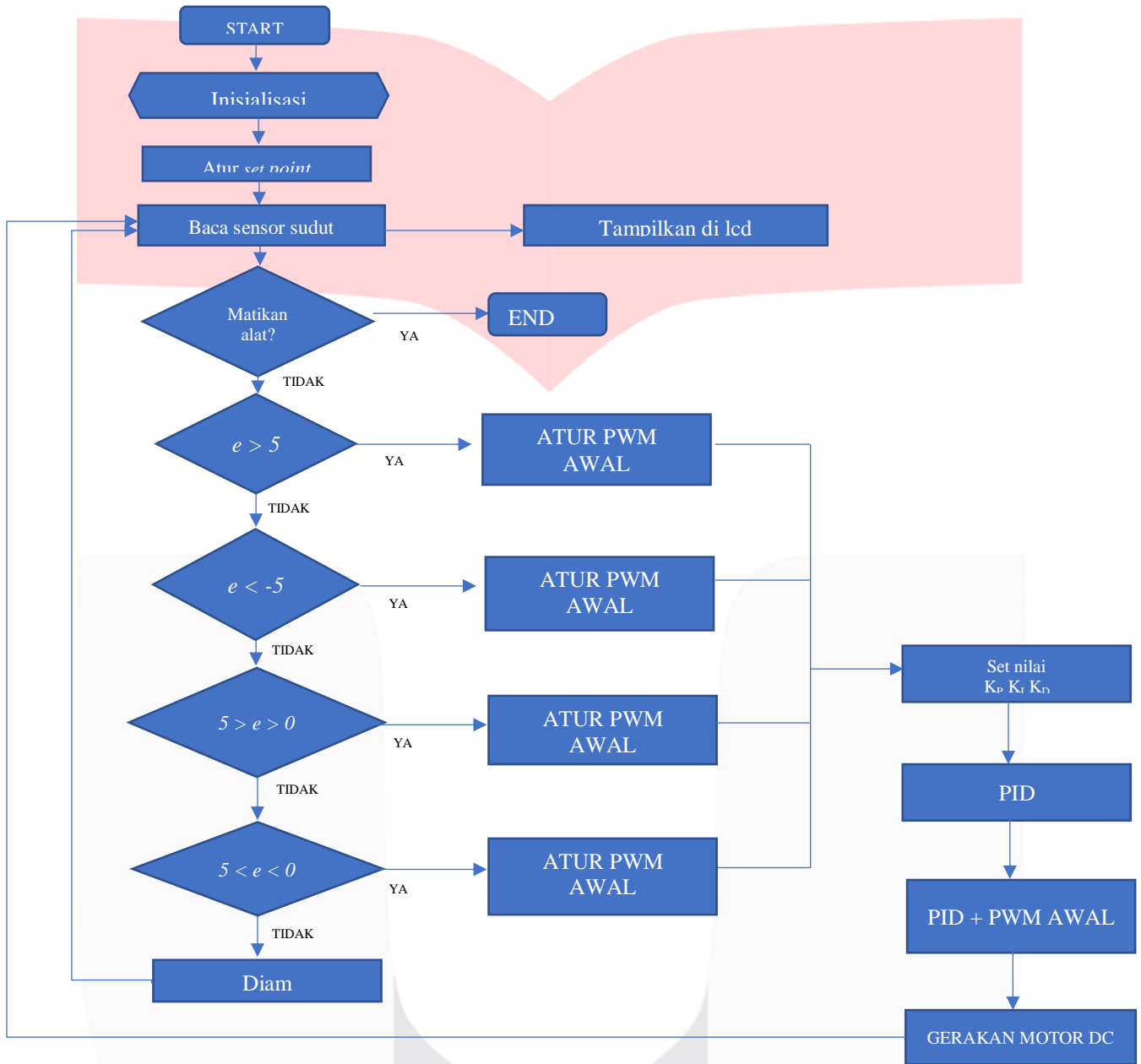
Gambar 4. *Self-balancing Trolley*

Berikut komponen yang digunakan untuk merancang *self-balancing trolley*:

- Mikrokontroler Arduino Uno
- Motor Driver H-Bridge Module IBT-2
- Motor DC 12v
- Sensor IMU (MPU6050)
- LCD 16X2
- Meja Troli
- Baterai 12v 12aH

Dari gambar 4 dapat kita lihat bahwa untuk mikrokontroler dan driver motor diletakan dalam kotak hitam yang berada pada pegangan dari troli. Sensor IMU diletakan pada bagian bawah dari troli sehingga dapat membaca sudut dari meja troli dengan baik serta tidak terganggu oleh barang yang dibawa pada meja troli. Untuk motor dc berada pada bagian bawah troli dan terhubung dengan meja troli oleh rantai dan gear.

3.2. Perancangan Perangkat Lunak



Gambar 5. Flowchart Perangkat Lunak Sistem

- a) Inisialisasi
Menginisialisasikan semua karakter yang akan digunakan dalam sistem.
- b) Set Point
Sistem menentukan *set point*. *Set point* digunakan sebagai titik acuan. Dalam hal ini *set point* bernilai nol, dalam kata lain meja troli dalam keadaan datar.
- c) Baca Sensor Sudut
Sensor sudut membaca sudut kemiringan dari meja troli. Data kemiringan selanjutnya akan langsung dikirim ke mikrokontroler

- d) Membandingkan dengan *Set Point*
Di sini sistem akan membandingkan data dari sensor sudut apakah sesuai dengan *set point* yang sudah ditentukan atau tidak. Jika sudah sesuai maka sistem tidak akan menggerakkan motor dc dan akan kembali membaca sudut dari troli.
- e) $-5 < \text{Sudut} < 5$
Sistem akan memeriksa apakah Sudut dari meja troli berada diantara -5 hingga 5 derajat. Jika tidak berarti sudut dari meja troli lebih besar dari 5 derajat atau lebih kecil dari -5 derajat
- f) $-5 \text{ Sudut} < 0$
Sistem akan memeriksa apakah sudut dari meja troli berada diantara -5 dan 0 derajat. Jika tidak berarti Sudut dari meja troli berada diantara 0 dan 5 derajat.
- g) $\text{Sudut} > 5$
Sistem akan memeriksa apakah sudut dari meja troli lebih besar dari 5 derajat. Jika tidak maka sudut dari meja troli lebih kecil dari -5 derajat.
- h) Atur PWM Awal
Sistem akan mengatur nilai dari pwm awal yang akan diberikan sesuai dengan sudut dari meja troli
- i) Set nilai K_p, K_i, K_d
Sistem akan memberikan nilai dari K_p, K_i, K_d untuk kontrol PID
- j) PID
Data yang diproses adalah nilai *error* sudut yang dihasilkan oleh sistem.. Nilai *error* sudut didapatkan dari perbedaan antara nilai *set point* dan pembacaan sudut dari sensor. Persamaan (3-1) menunjukkan *error* sudut.

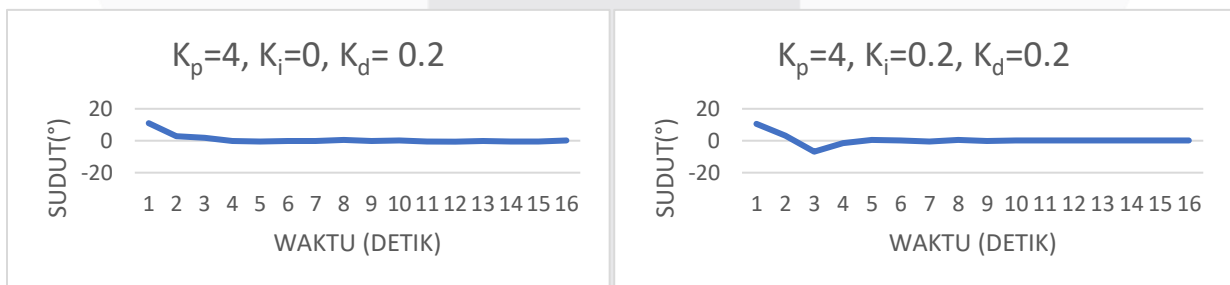
$$\text{error} = \text{set point} - \text{sudut sensor} \tag{3-1}$$

Setelah mendapatkan nilai *error* maka proses selanjutnya adalah proses kendai PID, nilai error tersebut digunakan untuk mencari nilai P, I, D. Setelah nilai dair P,I, dan D didapat, maka akan dilakukan pengecekan apakah nilai $\text{PID} < 0$ atau tidak, jika $\text{PID} < 0$ maka nilai dari PID akan dikalikan dengan -1 agar *output* dari PID bernilai *positif*.

- k) PID + PWM Awal
Nilai yang dihasilkan dari PID akan ditambahkan dengan PWM awal dan menjadi PWM total yang akan diberikan ke motor dc. Jika hasil dari penambahan tersebut bernilai lebih besar dari 255 maka nilainya akan tetap menjadi 255. Dan jika nilai dari penambahan jurang dari 0, maka nilainya akan tetap menjadi 0. Hal tersebut dikarenakan nilai dari PWM berbatas 0 sampai dengan 255.
- l) Gerakan Motor DC
Hasil dari proses sebelumnya yaitu PWM total akan digunakan sebagai input dari Driver Motor untuk menggerakkan posisi dari motor dc. Motor DC akan bergerak sesuai perintah dari sistem. Gerakkan dari motor dc akan mempengaruhi posisi dari meja troli. Selanjutnya sistem akan kembali membaca sensor sudut.
- m) Tampilkan di LCD
Nilai dari sudut yang dibaca oleh sensor akan langsung di tampilkan pada LCD.

4. Hasil Pengujian dan Analisis

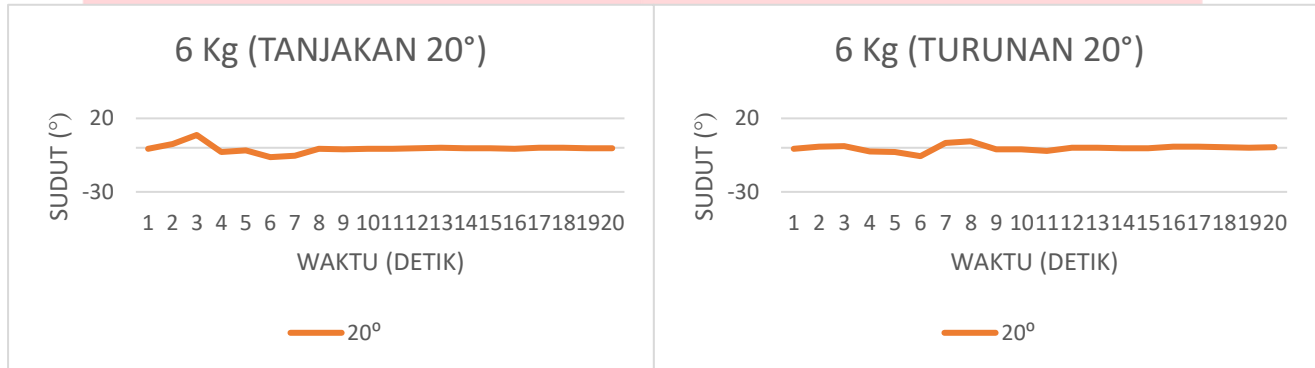
Pengujian sistem dilakukan pada kondisi statis dan dinamis. Untuk pengujian pada kondisi statis dengan cara, troli dimiringkan hinggna 10°. Lalu meja troli akan dilihat responnya hingga meja troli mendekati 0°. Pengujian pada kondisi statis dilakukan untuk mengetahui bagaimana respon dari sistem ketika diberi nilai dari $K_p, K_i,$ dan K_d yang berbeda-beda.



Gambar 6. Grafik Hasil Uji Statis

Dari gambar 6 dapat dilihat bahwa sudut meja troli dapat mendekati 0° dalam waktu 4 detik. Pemilihan nilai K_p , K_i , K_d dilakukan dengan metode coba-coba atau *trial and error*. Nilai yang dipilih adalah $K_p=4$, $K_i=0$, dan $K_d=0.2$. dari gambar diatas juga bisa dilihat bahwa penambahan nilai dari K_i dapat mengurangi *error steady state* namun juga dapat menambah *rise time* dari sistem.

Untuk pengujian saat kondisi dinamis, Pengujian ini dilakukan dengan cara troli didorong melalui tanjakan, turunan, dengan sudut kemiringan 20° , serta jalan datar dengan diberi beban di atas troli . Data yang diambil adalah nilai sudut yang terbaca oleh sensor dalam kurun waktu 20 detik.



Gambar 7. Grafik Hasil Pengujian Tanjakan dan Turunan

Gambar diatas merupakan grafik hasil pengujian saat troli melewati tanjakan dan turunan dengan sudut kemiringan 20° dengan keterangan sebagai berikut:

- Pada tanjakan 20°
 - Nilai $K_p = 5.5$, $K_i = 0$ dan $K_d = 0.2$
 - PWM Awal ketika $error < = 70$
 - PWM Awal ketika $error > 0 = 125$
 - PWM Awal ketika $0 > error > -4 = 65$
 - PWM Awal ketika $4 > error > 0 = 70$
 - Simpangan terjauh yang dihasilkan adalah 8.71° dan -6.36°
- Pada turunan 20°
 - Nilai $K_p = 5$, $K_i = 0$, dan $K_d = 0.2$
 - PWM Awal ketika $error < = 70$
 - PWM Awal ketika $error > 0 = 125$
 - PWM Awal ketika $0 > error > -4 = 65$
 - PWM Awal ketika $4 > error > 0 = 70$
 - Simpangan terjauh yang dihasilkan adalah 4.4° dan -5.7°

Pemilihan nilai K_p , K_i , K_d , dan PWM awal dari sistem menggunakan metode coba-coba. Dari gambar diatas dapat dilihat bahwa sistem dapat mulai menuju keadaan diam pada detik ke 8-9. Jika troli melewati tanjakan atau turunan dengan sudut kemiringan yang berbeda, maka nilai dari k_p , k_i , dan k_d nya pun akan berbeda lagi. Selain pengujian pada jalan tanjakan dan turunan, penulis juga menguji troli pada saat berjalan pada jalanan datar yang halus. Dengan hasil pengujian sebagai berikut:



Gambar 8. Grafik Hasil Pengujian pada Jalan Bertekstur Halus

Gambar 8 menunjukkan grafik respon sistem ketika berjalan pada jalan yang halus dengan nilai $k_p=1$, $k_i=0.2$, dan k_d , 0.2. dari grafik di atas, dapat disimpulkan bahwa sistem cenderung asih berosilasi. Hal tersebut dikarenakan sensor yang digunakan terlalu sensitif sehingga sangat berpengaruh pada respon sistem.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang sudah dibuat, didapatkan beberapa kesimpulan sebagai berikut:

1. Sistem kerja troli sudah dapat mempertahankan kedataran dari meja troli saat melewati jalan yang menanjak dan menurun dengan sudut kemiringan -30° hingga 30° walaupun masih terdapat *error steady state* pada sistem.
2. Sistem kerja troli saat melewati jalan yang datar cenderung masih mengalami *osilasi* karena sensor IMU selalu memiliki *error* dan pembacaan yang terlalu sensitif.
3. Pada pengujian, untuk tiap beban yang dibawa dibutuhkan konfigurasi nilai K_p , K_i , dan K_d yang berbeda-beda. Pada tiap pengujian PID, Penambahan nilai K_p dapat mempercepat *rise time* dari sistem, tetapi juga menambah *overshoot* dari sistem. Penambahan nilai K_i dapat mempercepat *rise time* dan memperbaiki *error steady state* dari sistem, tetapi juga menambahkan *overshoot* dari sistem. Sedangkan Penambahan nilai K_d dapat mengurangi *overshoot* dari sistem, tetapi juga mengurangi *rise time* dari sistem.
4. Pada pengujian sistem saat kondisi statis, waktu yang dibutuhkan sistem untuk menggerakkan meja troli dari kemiringan meja 10° menuju keadaan *settle* (0°) adalah 4 detik.
5. Pada pengujian sistem saat kondisi dinamis, waktu yang dibutuhkan sistem untuk menggerakkan meja troli ketika melewati tanjakan atau turunan dengan kemiringan 20° dan membawa beban 6 kg menuju keadaan *settle* (0°) adalah 8-9 detik.
6. Respon sudut yang dihasilkan dari meja troli dapat termonitor oleh lcd dan serial monitor pada komputer.

5.2. Saran

Berdasarkan hasil dari perancangan, pengujian dan analisis dari tugas akhir ini, maka penulis memberi saran:

1. Pembuatan desain dan konstruksi alat yang lebih baik agar tidak ada faktor lain yang dapat mempengaruhi kestabilan sistem.
2. Penentuan nilai K_p , K_i , dan K_d , yang lebih baik akan membuat respon sistem lebih cepat dan stabil.
3. Untuk pengerjaan tugas akhir selanjutnya, dapat menggunakan sensor dengan tingkat error lebih kecil.
4. Konstruksi dari troli dapat membawa beban yang lebih berat

Daftar Pustaka

- [1] OGATA, K. (1995). TEKNIK KONTROL AUTOMATIK (SISTEM PENGATURAN) Jilid 1. Erlangga.
- [2] Karl J. Astrom, T. H. (1934). PID Controllers, 2nd Edition. Instrument
- [3] Ali, M. (2004). PEMBELAJARAN PERANCANGAN SISTEM KONTROL PID DENGAN SOFTWARE MATLAB. Jurnal Edukasi.
- [4] Setyono, A., Wahyudi, & Iwan Setiawan. (2010). PERANCANGAN PERANGKAT LUNAK PENDETEKSI POSISI BENDA DALAM 6. Makalah Seminar Tugas Akhir.
- [5] Axelson, J. (1994). The Microcontroller Idea Book. Madison: Lakeview Research.
- [6] BERNDT, D. (n.d.). Electrochemical Energy Storage.