

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Di era globalisasi sekarang ini teknologi dan informasi semakin berkembang pesat, begitu juga teknologi robot. Robotika merupakan bidang teknologi yang mengalami banyak perkembangan perubahan bentuk dan pola gerak robot dari para ilmuwan yang mengembangkan teknologi robot sampai perlombaan robot yang diadakan dari beberapa perguruan tinggi.

Robot *Line Follower* merupakan robot yang dapat bergerak mengikuti lintasan berupa garis pandu berwarna hitam di atas permukaan berwarna putih yang dibuat dengan tingkat presisi tertentu. Pengendali *Proporsional Integral Derivative* (PID) merupakan pengendali konvensional yang digunakan dalam dunia industri untuk membuat nilai keluaran, sehingga menghasilkan *error* sekecil mungkin.

Perancangan robot *Line Follower* ini pada dasarnya dibagi menjadi 2 bagian, yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Perancangan perangkat keras berupa perancangan bentuk robot, sensor kombinasi LED dan photodiode, sistem minimum ATmega32, *driver* motor dan *Liquid Crystal Display* (LCD), sehingga menjadi sistem yang dapat bekerja mengikuti lintasan. Perancangan perangkat lunak menggunakan bahasa pemrograman C kemudian program dapat ditanamkan ke dalam mikrokontroler.

Adanya perlombaan robot *line follower* yang dilaksanakan oleh beberapa perguruan tinggi yang bertujuan untuk mengembangkan pemahaman dalam bidang robotika dan menambah kreatifitas serta meningkatkan kompetensi generasi muda Indonesia, maka penulis menggunakan robot *line follower* untuk menyelesaikan suatu lintasan berupa garis hitam di atas permukaan putih pada kondisi yang terukur. Berdasarkan penjelasan di atas, penulis menggunakan lintasan sesuai dengan Ebotec#3".

1.2 Rumusan Masalah

Terdapat beberapa perumusan masalah yang perlu diperhatikan dalam Proyek Akhir ini sebagai berikut:

1. Bagaimana perancangan kontrol gerak robot *Line Follower* menggunakan algoritma Proportional Integral Derivative (PID)?
2. Bagaimana implementasi dari perancangan PID terhadap pergerakan robot *Line Follower* untuk mendapatkan kondisi yang terukur?

1.3 Tujuan

Adapun tujuan dari Proyek Akhir ini adalah

1. Merancang kontrol gerak robot *Line Follower* menggunakan algoritma Proportional Integral Derivative (PID).
2. Melakukan implementasi dari perancangan PID terhadap pergerakan robot *Line Follower* untuk mendapatkan kondisi yang terukur.

1.4 Batasan Masalah

Adapun batasan masalah dalam Proyek Akhir ini adalah

1. Perangkat keras yang digunakan adalah mikrokontroler ATmega32 sebagai sistem kontrol untuk pemrosesan semua sistem, *Liquid Crystal Display* (LCD) sebagai penampil menu dan *driver motor* sebagai penggerak motor.
2. Perangkat lunak yang digunakan BASCOM AVR yaitu kompilasi untuk mikrokontroler.
3. Sensor yang digunakan LED putih dan photodiode.
4. Algoritma yang digunakan adalah Proportional Integral Derivative (PID)
5. Robot berjalan mengikuti garis hitam di atas permukaan putih pada lintasan ebotech#3.

1.5 Definisi Operasional

Robot line follower menggunakan algoritma PID merupakan salah satu bentuk aplikasi dari IC mikrokontroler ATmega32 yang menggunakan sensor dari kombinasi rangkaian LED dan photo

dioda. Spesifikasi robot *line follower* ini adalah menggunakan 8 sensor di depan via ADC, dan menggunakan Multiplexer (MUX), *high RPM* DC motor, gear motor 1: 4.8, diameter roda 33 mm, *battery* Lipo Turnigy Nano Tech 850 mAH, driver motor dual mosfet H-Bridge with *auto brake system* motor serta menggunakan bahasa pemrograman BASCOM – AVR versi 1.11.9.5. Lebar garis yang ideal untuk dilalui robot *line follower* ini sekitar 1,5 – 2 cm dengan kemungkinan 2- 3 sensor dapat mengenai garis.

1.6 Metode Pengerjaan

Metode yang digunakan dalam menyusun Proyek Akhir yang berjudul "Rancang Bangun Robot *Line Follower* Menggunakan Algoritma PID" menggunakan metode SDLC (*Systems Development Life Cycle*) yang terdiri dari beberapa tahapan, yaitu:

1.6.1 Metode Pengumpulan Data

- a. Pencarian referensi dan sumber-sumber yang berhubungan dengan perancangan *line follower* menggunakan algoritma PID.
- b. Mempelajari dan memahami proses dari masing- masing kontrol yaitu kontrol P, kontrol I dan kontrol D.

1.6.2 Metode Pengembangan Sistem

a. Tahap Analisis

Pada tahap analisis dalam perancangan robot *line follower* menggunakan algoritma PID berdasarkan tinjauan pustaka, yaitu mencari referensi atau materi perancangan robot, referensi tentang perancangan perangkat keras robot *line follower*, pemrograman bahasa C pada pengendali PID.

b. Tahap Perancangan dan Implementasi

Pada tahap ini robot *line follower* yang dirancang terdiri dari perangkat keras dan perangkat lunak. Pada perancangan perangkat keras yang di rancang yaitu bentuk robot, rangkaian sistem minimum yang menggunakan ATmega 32, rangkaian sensor, rangkaian *driver* motor, dan LCD. Pada perancangan perangkat lunak yang dirancang yaitu pemrograman untuk sensor dan pemrograman untuk kontrol P, I, dan D dengan mengatur parameter P, I, dan D agar mendapatkan sinyal keluaran sistem yang diinginkan terhadap masukan tertentu. Dalam implementasi masing-masing rangkaian dari perangkat keras maupun pemrograman dapat bekerja sesuai fungsinya. Dengan menggunakan algoritma PID, robot *line follower* dapat bergerak dengan kondisi yang terukur.

c. Tahap Pengujian

1. Pengujian Mikrokontroler ATmega32

Pengujian mikrokontroler dilakukan untuk memastikan IC mikrokontroler dapat diisi program dan mengeluarkan tegangan output. Pengujian Mikrokontroler ATmega32 dilakukan dengan *mendownload* program dan memfungsikan semua kaki mikrokontroler sebagai *output* berlogika *high* untuk menguji tegangan di setiap kakinya. Hasil pengujian menunjukkan bahwa script program dari *BASCOM AVR* bisa diunduh, sehingga dari hasil ini dapat dipastikan bahwa mikrokontroler masih bisa digunakan.

2. Pengujian LCD

Pengujian LCD dilakukan dengan cara mengunduh program tes LCD untuk menampilkan script program tes LCD. Tegangan yang masuk terukur kerangkaian LCD adalah 3,2 V.

3. Pengujian *Driver* Motor DC

Pengujian driver motor DC dilakukan dengan cara memberikan variasi input pada driver mosfet *H-Bridge*, sehingga motor kanan dan kiri bisa bergerak maju, mundur, kanan, kiri dan berhenti.

4. Pengujian Sensor

Lampu LED mengenai permukaan terang, sehingga ada pantulan cahaya yang masuk mengenai *photodiode*, maka resistansi photodiode akan mengecil. Keadaan ini mengakibatkan adanya arus yang mengalir ke dalam photodiode, sehingga arus yang mengalir ke ADC menjadi kecil. Keadaan sebaliknya, apabila photodiode berada pada garis hitam sehingga sangat sedikit cahaya yang dipantulkan, maka resistansi photodiode akan naik. Hal ini mengakibatkan arus tidak bisa mengalir ke photodiode, aliran arus ini diteruskan menuju ADC. Keadaan saat photodiode terkena cahaya dan tidak terkena cahaya mengakibatkan adanya perbedaan tegangan yang masuk ADC. Perbedaan tegangan ini akan di konversi menjadi logika biner "1" dan "0" dengan menentukan batas tegangan tengah (*threshold*).

5. Pengujian PID

Algoritma PID terdiri dari tiga kontrol, yaitu kontrol *proportional*, kontrol *integral* dan kontrol *derivative*.

a. Kontrol Proporsional

Kontrol P didefinisikan sebagai kontrol dasar yang berfungsi menanggapi *error* yang terjadi pada sistem dan mengukur seberapa jauh robot keluar dari garis. Kondisi ideal pada robot adalah bergerak maju lurus mengikuti garis, dengan kata lain $PV = 0$ (nilai

sensor =00011000). *Process Variable* (PV) merupakan nilai sensor yang dibaca oleh sensor atau nilai aktual pembacaan.

b. Kontrol Derivative

Kontrol D digunakan untuk mengukur seberapa cepat robot bergerak dari kiri ke kanan atau dari kanan ke kiri. Semakin cepat robot bergerak dari satu sisi ke sisi lainnya, maka semakin besar nilai D. Konstanta D (Kd) juga digunakan untuk menambah atau mengurangi imbas dari derivative. Dengan mendapatkan nilai Kd yang tepat, maka pergerakan robot dari sisi ke sisi yang bergelombang bisa di minimalisasi dari proporsional PWM .

c. Kontrol Integral

Kontrol I digunakan untuk mengakumulasi *error* dan mengetahui durasi *error*. Dengan menjumlahkan *error* di setiap pembacaan PV akan memberikan akumulasi *offset* yang harus diperbaiki sebelumnya. Saat robot bergerak menjauhi garis, maka nilai *error* akan bertambah. Semakin lama tidak mendapatkan SP, maka semakin besar nilai I. Untuk mendapatkan nilai Ki yang tepat, imbas dari Integral bisa dikurangi.

1.7 Jadwal Pengerjaan

Tabel 1 menunjukkan jadwal pengerjaan proyek akhir

Tabel 0
Jadwal Pengerjaan Proyek Akhir

NO	Kegiatan	Tahun 2014																			
		Februari				Maret				April				Mei				Juni			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pengumpulan Data	■	■	■	■																
2	Analisis					■	■	■	■												
3	Perancangan					■	■	■	■	■	■	■	■	■	■	■	■				
4	Implementasi									■	■	■	■	■	■	■	■				
5	Pengujian													■	■	■	■	■	■	■	■
6	Pembuatan Laporan													■	■	■	■	■	■	■	■