

Kontrol Kecepatan Motor DC Dengan Metode PID Menggunakan Visual Basic 6.0 Dan Mikrokontroler ATmega 16

Muhammad Rizki Setiawan, M. Aziz Muslim dan Goegoes Dwi Nusantara

Abstrak—Dalam penelitian ini telah diimplementasikan suatu sistem kontrol motor DC dengan kontroler PID menggunakan mikrokontroler ATmega16. *Interface* sistem kontrol ini menggunakan Visual Basic 6.0 sehingga memudahkan untuk pengembangan lebih lanjut. Proses identifikasi sistem dilakukan dengan menggunakan sinyal input *Pseudo Random Binary Sequence* (PRBS) 10 bit dengan variasi sinyal acak sebanyak 1024 data. Dari hasil identifikasi diperoleh fungsi alih model system adalah
$$F(s) = \frac{3,492 s^2 + 24,4 s + 0,5313}{s^3 + 9,173 s^2 + 32,05 s + 0,5376}$$
 Selanjutnya dilakukan proses penentuan parameter PID dengan menggunakan metode *root locus*, yang hasilnya menunjukkan bahwa semua akar berada disebelah kiri bidang *s*. Sehingga respon yang didapat dari semua pole stabil. Hasil perhitungan parameter PID dengan pole $s = -9,4 + j3,2$ didapatkan nilai parameter PID terbaik yaitu $K_p=4,6805$, $K_i=10$ dan $K_d=0,1026$.

Kata Kunci—identifikasi sistem, *root locus*, PID.

I. PENDAHULUAN

KEBUTUHAN akan sistem kontrol yang lebih efektif dan efisien di era modern ini semakin meningkat, mengingat bahwa jumlah *plant* yang akan dikontrol semakin banyak dan memiliki struktur yang semakin kompleks. Maka dari itu sistem kontrol yang dapat dikendalikan melalui perangkat komputer mutlak diperlukan. Selain dapat melakukan sistem monitoring secara real time dengan menggunakan penyajian data yang lebih bagus, perangkat komputer dapat dengan mudah melakukan proses pengiriman set point yang dikehendaki..

Salah satu jenis kontroler yang banyak digunakan saat ini adalah kontroler PID (Proporsional Integral Diferensial) karena kontroler ini sederhana dan relatif mudah dalam pengaplikasiannya. Pada umumnya PID

Muhammad Rizki Setiawan adalah mahasiswa Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang Indonesia (Korespondensi penulis melalui HP 085736021066; email pen43x@yahoo.com)

M. Aziz Muslim adalah dosen Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang Indonesia.

Goegoes Dwi Nusantara adalah dosen Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang Indonesia (Korespondensi penulis melalui HP 081328199511; email goegoesdn@ub.ac.id, goegoesdn@yahoo.com)

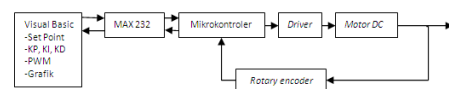
diimplementasikan menggunakan rangkaian analog bahkan ada yang menggunakan komponen mekanis dalam penentuan *set point*, *setting parameter*.

Berdasarkan permasalahan tersebut maka dalam skripsi ini akan dirancang sebuah alat dengan piranti lunak yang mampu mengendalikan kecepatan motor DC dengan sistem monitoring secara *real time* sehingga memudahkan untuk mengetahui perubahan data yang terjadi. Data yang disajikan langsung dirubah menjadi bentuk grafik supaya dapat dimengerti dengan baik, hal tersebut dapat dilakukan mengingat pembuatan perangkat lunak dalam skripsi ini dilakukan menggunakan program Visual Basic 6.0 yang dapat menyajikan data dalam bentuk grafik secara real time. Sedangkan untuk perangkat kerasnya menggunakan Mikrokontroler ATmega16 dan perantara port serial dalam komunikasi datanya.

Tujuan yang ingin dicapai dalam penelitian ini adalah terbuatnya perangkat lunak yang mampu melakukan monitoring, pengiriman *set point* dan parameter PID pada Mikrokontroler untuk menjalankan *driver*.

II. RANCANGAN PENELITIAN

A. Blok Diagram Keseluruhan Sistem



Gambar 1. Blok Diagram Keseluruhan Sistem

Blok diagram keseluruhan sistem dari penelitian ini seperti pada Gambar 1, dimana Max232 sebagai penghubung antara visual basic dan mikrokontroler sedangkan *driver* merupakan penghubung antara mikrokontroler dengan motor DC.

B. Visual Basic 6.0

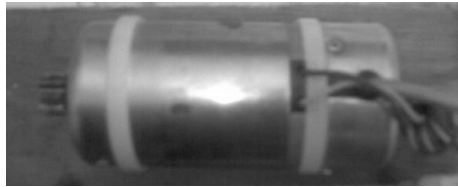
Merupakan bahasa pemrograman berbasis visual dengan dukungan berbagai fitur yang bisa digunakan dalam pembuatan perangkat lunak yang dinamis.

Dalam penelitian ini perangkat lunak yang dibuat harus mampu mengirim dan menerima data dari serial *port* dan menggambarkan grafik kecepatan motor secara

real time. Hal tersebut dapat dilakukan mengingat Visul Basic 6.0 memiliki fitur-fitur tersebut.

C. Motor DC

Motor yang digunakan dalam penelitian kali ini merupakan motor DC magnet permanen seperti pada Gambar 2. Untuk melakukan perubahan kecepatan dari motor ini adalah dengan diaturnya tegangan yang masuk ke dalam motor.



Gambar 2. Motor DC

D. Rotary Encoder

Sensor *rotary encoder* yang digunakan dalam penelitian ini terhubung langsung dengan motor DC dapat dilihat pada Gambar 3, memiliki 2 keluaran 5 V. akan tetapi dalam penelitian kali ini yang digunakan hanya 1 keluaran karena dalam penelitian ini lebih ditekankan untuk pembacaan kecepatan motor. Sensor *rotary encoder* ini memiliki resolusi sebesar 200 pulsa setiap 1 kali putaran



Gambar 3. Sensor Rotary Encoder

E. Mikrokontroler ATmega16 [1]

Mikrokontroler berfungsi sebagai pusat pengolahan data dan pengendali bagi perangkat lain seperti sensor SHT-11 dan modul GPS EG-T10. Untuk memenuhi kebutuhan memori program yang cukup besar, maka digunakan mikrokontroler ATmega16.

ATmega16 memiliki 32 pin yang digunakan untuk *input/output*, pin-pin tersebut terdiri dari 8 pin sebagai port A. 8 pin sebagai port B. 8 pin sebagai port C. 8 pin sebagai port D. Dalam komunikasi serial, maka hanya port D yang dapat digunakan karena fungsi khusus yang dimilikinya.

F. Pengambilan Data Input-Output[2]

Pengujian ini tentu memerlukan sinyal uji tertentu yang akan diberikan kepada sistem fisik yang akan diidentifikasi. Agar diperoleh model yang tepat maka dalam pemilihan sinyal uji ini tidak boleh sembarangan. Syarat pemilihannya adalah suatu sinyal uji harus memiliki cakupan frekuensi yang lebar dan standard yang digunakan adalah sinyal *Pseudo Random Binary Sequences* (PRBS)[3].

Pseudo Random Binary Sequence (PRBS) adalah

sinyal kotak yang termodulasi pada lebarnya dan berlangsung secara sekuensial. Sinyal ini biasanya dibangkitkan menggunakan *Linear Feedback Shift Register* (LFSR). Pada LFSR memiliki 2 parameter dasar yang menentukan sifat sekuensial yang dihasilkan, yaitu: panjang dari *shift register* dan susunan umpan balik. PRBS memiliki variasi panjang sekuensialnya, tergantung dari panjangnya *shift register* seperti ditunjukkan Tabel 1

Panjang dari *shift register* menentukan periode maksimum yang dapat dihasilkan dari sekuensial PRBS dan tidak berulang yang dapat dinyatakan dengan persamaan:

$$LPRBS = 2^n - 1 \quad (1)$$

Dimana n adalah panjang dari register LFSR (jumlah bit). Panjang maksimum dari PRBS disebut *M-sequence*.

TABEL 1
TABEL VARIASI PANJANG SEKUENSIAL PRBS

Panjang Register (N)	Panjang Sekuensial L=2N-1	Posisi Tap Umpan Balik
2	3	1 dan 2
3	7	1 dan 3
4	15	3 dan 4
5	31	3 dan 5
6	63	5 dan 6
7	127	4 dan 7
8	255	2, 3, 4, dan 8
9	511	5 dan 9
10	1023	7 dan 10

G. Perancangan PID dengan Root Locus[4]

Rancangan sistem kendali loop tertutup dengan metode *root locus* dimungkinkan untuk mengatur sekurang-kurangnya beberapa letak pole sistem loop tertutup sehingga dapat diatur tanggapan *transient* pada tingkat tertentu dan pengaruhnya terhadap tanggapan keadaan mantap [5]. Prosedur analitis perancangan kontroler PID menggunakan metode *root locus* yang dijelaskan dalam *Feedback Control System* oleh Charles L. Philips dan Royce D. Harbour dapat dilihat dalam Gambar 4 berikut.



Gambar 4. Sistem Kendali

Untuk sistem tersebut, persamaan karakteristik diberikan oleh

$$1 + G_C(s)G_p(s) = 0 \quad (2)$$

Misalkan diinginkan *root locus* melalui

$s = s_1$, maka

$$G_C(s_1)G_p(s_1) = -1$$

$$G_C(s_1)|G_p(s_1)|e^{j\psi} = 1e^{j\pi}$$

Fungsi alih kontroler PID setelah ditransformasi laplace dinyatakan oleh

$$G_C(s) = K_p + \frac{K_i}{s} + K_d s$$

Sehingga dari persamaan diatas didapatkan

$$Gc(s_1) = \frac{1}{[Gp \zeta_1]} e^{j(\Pi - \psi)}$$

atau

$$Kd s_1^2 + Kp s_1 + Ki = \frac{e^{j(\Pi - \psi)}}{[Gp \zeta_1]} \quad (3)$$

Dengan

$$s_1 = |s_1| e^{j\beta}$$

Maka,

$$\begin{aligned} Kd |s_1|^2 [\cos 2\beta + j \sin 2\beta] + Kp |s_1| [\cos \beta + j \sin \beta] + Ki \\ = \frac{|s_1|}{[Gp \zeta_1]} [\cos(\beta + \Pi - \psi) + j \sin(\beta + \Pi - \psi)] \end{aligned}$$

Menyamakan real dengan real dan imajiner dengan imajiner, didapat

$$\begin{bmatrix} |s_1|^2 & |s_1| \cos \beta \\ |s_1|^2 & |s_1| \sin \beta \end{bmatrix} \begin{bmatrix} Kd \\ Kp \end{bmatrix} = \begin{bmatrix} \frac{|s_1|}{Gp \zeta_1} \cos(\beta + \Pi + \psi) - Ki \\ \frac{|s_1|}{Gp \zeta_1} \sin(\beta + \Pi + \psi) \end{bmatrix}$$

atau

$$\begin{bmatrix} |s_1|^2 & |s_1| \cos \beta \\ |s_1|^2 & |s_1| \sin \beta \end{bmatrix} \begin{bmatrix} Kd \\ Kp \end{bmatrix} = \begin{bmatrix} -\frac{|s_1|}{Gp \zeta_1} \cos(\psi - \beta) - Ki \\ \frac{|s_1|}{Gp \zeta_1} \sin(\psi - \beta) \end{bmatrix} \quad (4)$$

Dari persamaan diatas dapat dilihat bahwa untuk perancangan kontroler PID, satu dari tiga penguatan Kp, Ki, Kd, harus ditentukan dahulu. Sedangkan untuk perancangan PI atau PD, penguatan yang sesuai pada persamaan diatas dibuat sama dengan nol.

Untuk kasus S_1 adalah imajiner, persamaan diatas akan menghasilkan dua persamaan dalam Kp dan Kd serta besar Ki harus ditentukan terlebih dahulu [3].

III. PERANCANGAN MODUL

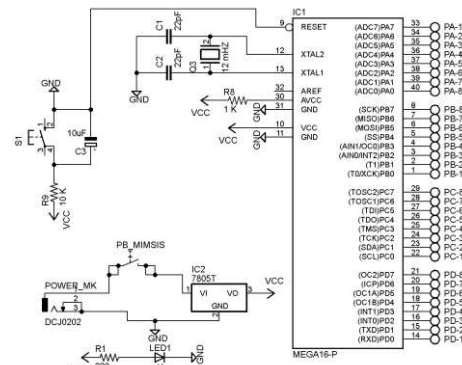
Perancangan ini meliputi pembuatan perangkat keras dan perangkat lunak, perangkat keras meliputi perancangan Minimum sistem ATmega16 dan rangkaian Max232 sebagai penghubung komputer dengan mikrokontroler, untuk perangkat lunak meliputi pembuatan program pada mikrokontroler untuk keperluan analisis sistem yaitu dengan membangkitkan sinyanya PRBS dan program kontrol PWM. Sedangkan pada visual basic 6.0 dibuat program untuk menerima dan mengirim data sekaligus membuatnya dalam bentuk grafik.

A. Minimum Sistem ATmega16

Dalam penelitian ini mikrokontroler yang digunakan adalah ATmega16 untuk penghitungan nilai RPM motor sekaligus pengiriman nilai RPM kepada program Visual Basic 6.0 dan pengiriman sinyal PWM kepada driver. Kristal yang digunakan sebesar 11.0592 MHz. Rangkaian minimum sistem ATmega16 seperti pada Gambar 5.

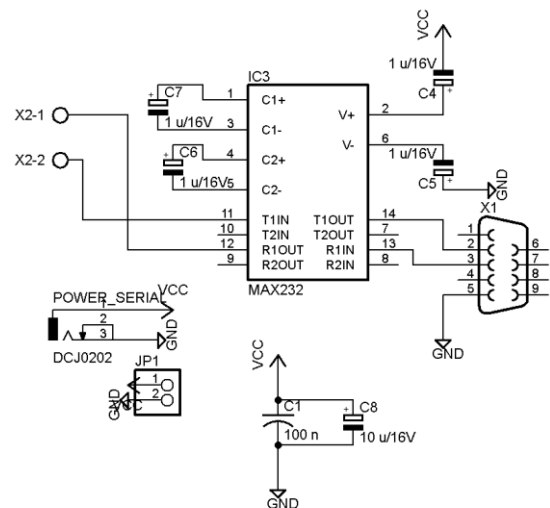
A. Rangkaian Max 232

IC MAX232 dari Maxim Incorporation adalah IC



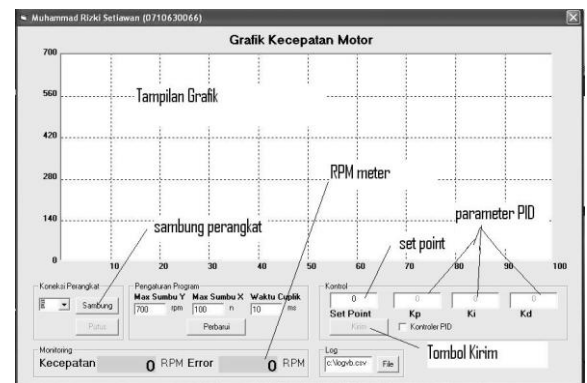
Gambar 5. Rangkaian Minimum Sistem ATmega16

pengubah level TTL menjadi RS-232 atau sebaliknya, yang memiliki sebuah *charge pump* tegangan yang dihasilkan +10V dan -10V dari tegangan catu daya 5V. Tegangan-tegangan ini dihasilkan dengan proses pengisian dan pembuangan empat kapasitor luar yang dihubungkan dengan rangkaian pengganda tegangan internal yang dimiliki IC ini. Rangkaian Max 232 seperti pada gambar 6.



Gambar 6. Rangkaian Max 232

B. Pemrograman Visual Basic 6.0



Gambar 7. Tampilan Program Pengontrol Kecepatan

Komunikasi antara mikrokontroler dengan komputer

dapat dilakukan dengan memanfaatkan fitur *active MSCOMM* pada visual basic 6.0 [6], sedangkan untuk membuat grafik dari data kecepatan maka fitur timer dapat digunakan, pemrograman ini juga dibuat untuk melakukan proses *log* data kecepatan dari waktu ke waktu. *Interface* dari program yang dibuat diperlihatkan pada Gambar 7.

IV. PENGUJIAN DAN ANALISIS DATA

Pengujian ini meliputi pengujian perangkat keras dalam hal ini *driver* dan hubungan PWM dengan kecepatan motor diteruskan dengan pengambilan data *input-output* dengan penggunaan sinyal PRBS, kemudian data tersebut dianalisa dengan MATLAB dengan fasilitas *ident* yang tersedia.

A. Pengujian Driver Motor

Pengujian ini dilakukan dengan diberikan nilai PWM dari 0%-100% *duty cycle*, pengujian ini dilakukan dengan 2 keadaan yaitu pengujian *driver* tanpa motor dan pengujian *driver* menggunakan motor. Pengujian ini dilakukan untuk mengetahui hubungan antara nilai PWM dengan tegangan keluaran *driver*.



Gambar 8. Grafik PWM dan Tegangan Keluaran Driver tanpa Motor

Dari Gambar 8 dapat diketahui bahwa keluar *driver* sudah memiliki nilai tegangan meskipun PWM masih 0. Grafik hubungan diatas menunjukkan grafik yang linier.



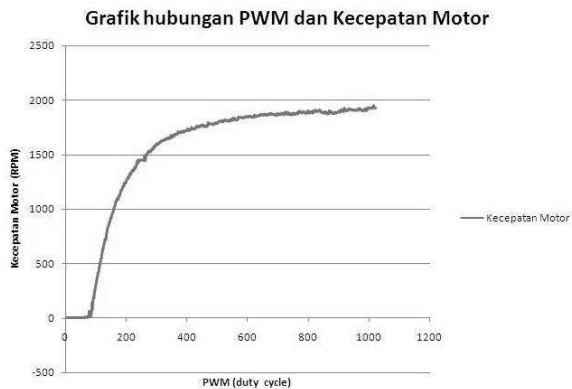
Gambar 9. Grafik PWM dan Tegangan Keluaran Driver Menggunakan Motor

Pada Gambar 9 dapat diketahui bahwa motor masih dalam keadaan berhenti meskipun sudah diberi nilai PWM sebesar 5% *duty cycle*, dan motor mulai bergerak pada kisaran 5%-10% *duty cycle*.

B. Pengujian Kecepatan Motor

Dengan memberikan nilai PWM sebesar 0% - 100%

duty cycle diperoleh respon kecepatan motor seperti Gambar 10.

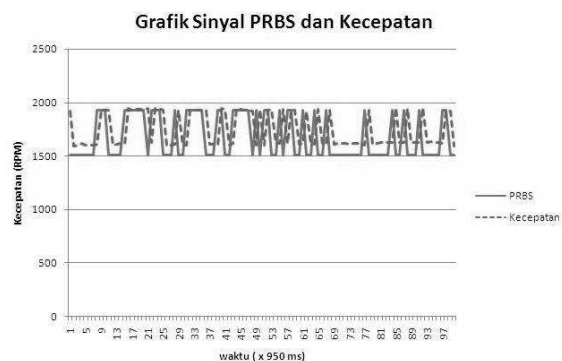


Gambar 10. Grafik Hubungan PWM dan Kecepatan Motor

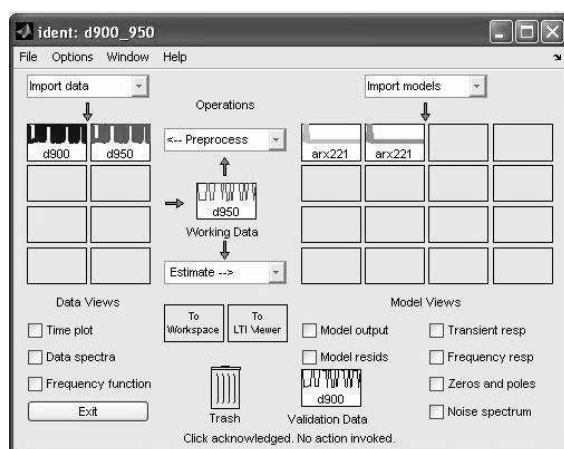
Dari grafik diatas motor mulai berputar di kisaran PWM 5%-10% *duty cycle*, kecepatan motor mengalami perubahan yang besar dalam kisaran 10%-30% *duty cycle* sedangkan 30%-100% *duty cycle* perubahan kecepatan tidak begitu besar.

C. Pengambilan data Input-Output

Pengambilan data *input-output* dengan cara dijalankannya program yang membangkitkan sinyal PRBS dari mikrokontroler. Hasil dari pengambilan data ditunjukkan pada Gambar 11.



Gambar 11. Grafik Hubungan Sinyal PRBS dengan Kecepatan

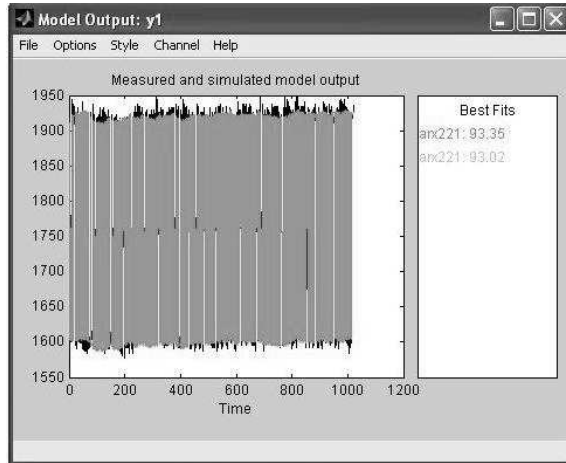


Gambar 12. Identifikasi Menggunakan Ident MATLAB

D. Identifikasi Menggunakan MATLAB

Identifikasi sistem bertujuan didapatkannya fungsi

alih dari keseluruhan sistem, pengujian dilakukan menggunakan software MATLAB dengan fungsi *ident* yang dimilikinya. Data identifikasi yang digunakan, PRBS sebagai *input* sedangkan kecepatan sebagai *output*. struktur model yang digunakan adalah *Auto Regressive with Exogenous input* (ARX) dengan estimasi parameter 2 2 1. Dapat dilihat pada Gambar 12 dan Gambar 13.



Gambar 13. Best Fit Sistem

Dari hasil diatas didapatkan best fit terbaik yaitu 93.35 % dengan fungsi alih diskrit

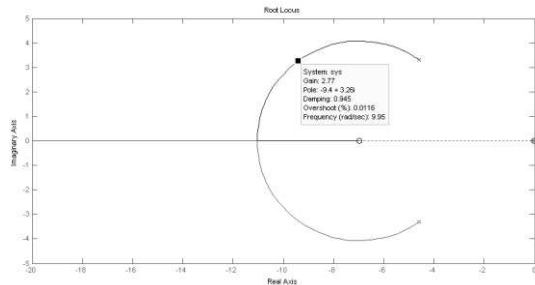
$$A(q) = 1 - 0.9717 (+0.009162) q^{-1} - 0.01271 (+0.00163) q^{-2}$$

$$B(q) = 0.7754 (+0.001338) q^{-1} - 0.7595 (+0.007813) q^{-2}$$

Dari data diatas bisa didapatkan fungsi alih dalam bentuk s nya:

$$F(s) = \frac{3.492 s^2 + 24.4 s + 0.5313}{s^3 + 9.173 s^2 + 32.05 s + 0.5376}$$

E. Penentuan Parameter PID



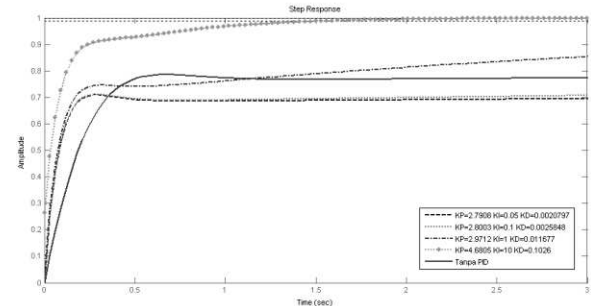
Gambar 14. Root Locus Fungsi Alih sistem dan Pemilihan Pole

Untuk mendapatkan parameter PID yang diinginkan terlebih dahulu ditentukan pole yang diinginkan berdasarkan grafik *root locus* dari sistem. Dari grafik *root locus* pada Gambar 14 dapat dilihat bahwa semua akar berada pada sisi kiri bidang s, dapat disimpulkan bahwa sistem stabil dalam nilai manapun, dalam penelitian ini dipilih pole $s = -9.4 + j3.26$.

Selanjutnya dilakukan perhitungan untuk menentukan parameter PID sesuai dengan kriteria yang diinginkan dalam hal ini menggunakan metode *root locus* yang diimplementasikan dalam program MATLAB. Sesuai dengan persamaan 4. [5] hasilnya seperti yang ditunjukkan pada Gambar 15.

Dari 4 jenis parameter PID yang didapat dipilih nilai PID yang memiliki respon terbaik yaitu :

- $K_P = 4.6805$
- $K_I = 10$
- $K_D = 0.1026$

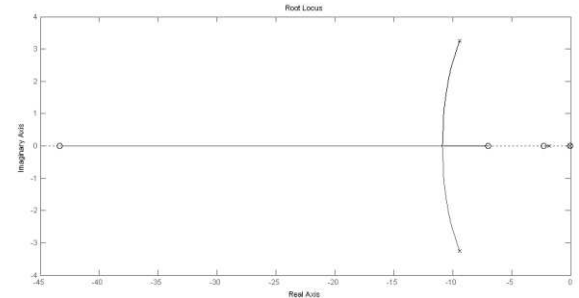


Gambar 15. Grafik Respon sistem tanpa PID dan Dengan PID

Dengan memasukkan nilai parameter PID kedalam fungsi alih sistem, maka fungsi alih sistem berubah menjadi:

$$F(s) = \frac{0.3583s^4 + 18.8478s^3 + 149.1795s^2 + 246.4868s + 5.313}{1.3583s^4 + 28.021s^3 + 181.23s^2 + 247.02s + 5.313}$$

Gambar *root locus* dari fungsi alih *plant* ditambah PID diperlihatkan pada Gambar 16.

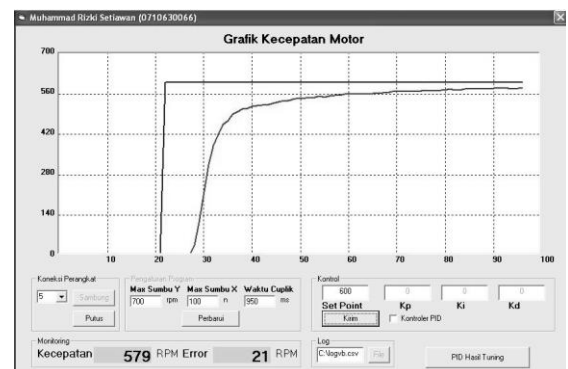


Gambar 16. Root Locus Sistem Keseluruhan

F. Pengujian Sistem keseluruhan

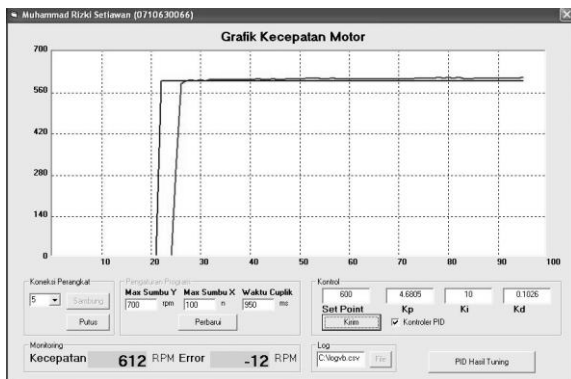
Pengujian sistem secara keseluruhan ini dilakukan untuk mengetahui kinerja perangkat keras dan perangkat lunak serta mengetahui respon motor tanpa PID dan dengan PID.

Dengan dilakukannya implementasi nilai parameter PID yang telah dihitung yaitu $K_P=4.680$, $K_I=10$ dan $K_D=0.1026$ ke dalam rangkaian keseluruhan sistem. Dari proses implentasi tersebut dihasilkan respon seperti pada Gambar 17 dan Gambar 18.



Gambar 17. Grafik Respon Motor Tanpa PID dengan Visual Basic

Dari Gambar 17 dapat diketahui bahwa respon motor tanpa menggunakan PID membutuhkan waktu yang lebih lama untuk mencapai *Steady State*.



Gambar 18. Grafik Respon Motor Menggunakan PID dengan Visual Basic

Dengan digunakannya parameter PID hasil tuning didapatkan respon yang lebih cepat dari pada respon tanpa menggunakan PID.

V. KESIMPULAN DAN PROSPEK

Berdasarkan dari serangkaian pengujian yang telah dilakukan didapatkan fungsi alih keseluruhan sistem dengan parameter PID adalah

$$F(s) = \frac{0.3583s^4 + 18.8478s^3 + 149.1795s^2 + 246.4868s + 5.313}{1.3583s^4 + 28.021s^3 + 181.23s^2 + 247.02s + 5.313}$$

hasil pengujian keakurasian menggunakan sinyal uji

PRBS sebesar 93.35 %, sedangkan perancangan parameter PID menggunakan metode *root locus* dengan nilai pole $s = -9.4 + j3.26$ didapatkan nilai $K_p=4.6805$, $K_i=10$ dan $K_d=0.1026$. Program pada visual basic dapat mengirim *set point*, parameter PID ke dalam mikrokontroler serta mampu membuat grafik dari nilai kecepatan secara *real time*.

Untuk membuat sistem kontrol kecepatan motor DC yang lebih fleksibel penggunaan USB lebih efektif karena port serial sudah jarang digunakan oleh masyarakat, sedang untuk penentuan parameter PID bisa menambahkan metode lain sebagai pembanding. Ditambah dengan modifikasi *hardware* dan *software* untuk mengetahui karakteristik ketika motor diberi beban.

DAFTAR PUSTAKA

- [1] ATMEL.2007.ATmega16/ATmega16L, 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash.
- [2] Wibowo, Teguh Budi. 2008. Pemodelan Sistem Plant Suhu Dengan Metode Identifikasi Recursive Least Square. Laporan Skripsi, Teknik Elektro Brawijaya
- [3] Landau, Ioan dan Gianluca Zito. 2006. Digital Control Systems Design, Identification and Implementation. Germany: Springer-Verlag London Limited
- [4] Ikrom, Hassanal. 2008. Perancangan Kontroler Pid-Kaskade Dengan Metode *Root locus* Untuk Kontrol Temperatur Dan Tekanan Pada Proses Evaporator. Laporan Skripsi, Teknik Elektro Brawijaya
- [5] Philip, C. L. & Harbor, R. D. 1996. *Feedback Control System*. Prentice Hall. New Jersey
- [6] Prasetya, Retna.dkk.2004. interfacing port paralel dan port serial computer dengan visual basic 6.0.Andi Yogyakarta