

DESAIN DAN SIMULASI HMI SISTEM KENDALI POSISI MOTOR DC BERBASIS *PROGRAMMABLE LOGIC CONTROLLER*

Syaiful Rachman ¹⁾, Zuraidah ²⁾,
Nurkamalia ³⁾, Isma Sapitri ⁴⁾

Saifulrachman1@poliban.ac.id ¹⁾, zuraidah469@gmail.com ²⁾,
mia.nurkamalia@gmail.com ³⁾, safitriisma@gmail.com ⁴⁾

^{1, 2, 3)} Teknik Elektronika, Politeknik Negeri Banjarmasin

⁴⁾ LPK Elite course, Banjarmasin

Abstrak

Pada penelitian ini, menitikberatkan pada desain dan simulasi pada kendali otomatis yaitu *Proportional-Integral* (PI), *Proportional-Derivative* (PD) dan *Proportional-Integral-Derivative* (PID). Untuk mempermudah pemahaman konsep kontrol perlu dibuat sistem yang bisa menulis *set point* (SP) dan *variabel proses* (PV), dan dapat menginputkan konstanta K_p K_i dan K_d, dengan memanfaatkan software HMI yang disimulasikan dengan PLC untuk kendali posisi motor DC. Dalam merancang sistem kontrol, dilakukan dengan metoda trial & error. Untuk mendapatkan aksi kontrol yang baik diperlukan langkah dengan kombinasi antara P, I dan D sampai ditemukan nilai K_p, K_i dan K_d yang diinginkan. Kemudian hasil pengujian simulasi dengan dengan kontroler PI diperoleh nilai K_p = 2 dan K_i = 1. Dengan hasil menggunakan kontroler PI respon bisa mencapai setpoint. Meskipun sistem yang masih kurang dari target yang diinginkan dengan error pembacaan sudut pada busur derajat berkisar antara 1° sampai 5°. Selanjutnya pengujian menggunakan kontroler *proportional derivative*. Dengan nilai K_p = 2 dan K_d = 1, menggunakan kontroler PD respon bisa *mencapai* setpoint. Kemudian hasil dengan kontroler *Proportional-Integral-derivative*, Dengan menggunakan nilai K_p = 2, K_i = 1 dan K_d = 1, respon kontroler PID bisa mencapai *setpoint*. Dari pengujian beberapa kontroler didapatkan penggunaan kontroler PID dan PD lebih baik dibanding dengan PI dalam simulasi HMI pengendalian posisi motor DC dalam penelitian ini.

Kata Kunci : *Desain, HMI, Simulasi, Proportional-Integral-Derivative*

1. PENDAHULUAN

Kendali otomatis telah memegang peranan yang sangat penting dalam perkembangan ilmu dan teknologi. Penerapan sistem kendali mencakup bidang yang sangat luas termasuk kestabilan pesawat ruang angkasa, peluru kendali, sistem pengemudian pesawat, dan sebagainya. Karena kemajuan dalam teori dan praktek, kontrol otomatis memberikan

kemudahan dalam mendapatkan performansi dari sistem dinamik, meningkatkan kualitas dan menurunkan biaya produksi sehingga meningkatkan laju produksi. Sehingga pemahaman yang baik dan mendalam di bidang sistem kendali menjadi suatu hal yang wajib bagi ilmuwan sekarang [1].

Permasalahannya, perhitungan model matematika yang kompleks membuat mata kuliah sistem kendali justru menjadi momok

yang menakutkan bagi sebagian besar mahasiswa elektronika dan instrumentasi. Padahal sebagai ilmuwan maupun praktisi dibidang instrumentasi, tidak akan terlepas dari pengetahuan otomasi industri. Sehingga perlu dibuat terobosan yang mampu menjadikan pemodelan matematika kompleks menjadi lebih sederhana. Untuk menunjang pada penelitian ini, didiskripsikan dalam bentuk penelitian terdahulu tentang teknik kontrol dalam mengendalikan motor DC, yang telah dilakukan meliputi beberapa penelitian dari jurnal dengan judul “Kontrol Kecepatan Motor DC Dengan Metode PID Menggunakan Visual Basic 6.0 Dan Mikrokontroler ATMEGA 16”, dalam jurnal tersebut menjelaskan 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 [2].

Selanjutnya penelitian berikutnya dengan judul “Implementasi Pengendali PID Untuk Kestabilan Posisi Terbang Wahana Tanpa Awak”, dalam artikelnya memaparkan tentang Pesawat tanpa awak atau UAV (*Unmanned Aerial Vehicle*) dengan pengendalian jarak jauh merupakan elektromekanik yang dapat melakukan misi-misi terprogram. Proses kontrol pesawat sepenuhnya dilakukan oleh sistem autopilot dengan mengacu pada parameter parameter yang telah ditentukan oleh pengguna sebelum terbang. Pada pesawat tanpa awak juga dibutuhkan kestabilan pada saat di udara. Stabilitas pesawat adalah kemampuan untuk kembali ke posisi tertentu dalam suatu penerbangan [3].

Pada penelitian ini, menitikberatkan pada aplikasi salah satu jenis kendali otomatis yaitu kendali PI (*proportional dan Integral*), PD (*proportional dan derivative*) dan PID (*proportional, Integral, derivative*). Untuk mempermudah pemahaman konsep kontrol perlu dibuat sistem yang bisa menulis set point. Yaitu dengan memanfaatkan software HMI yang disambung dengan PLC sebagai pengolah data. Sedangkan tujuan penelitian ini

membuat desain sistem dan mensimulasi kan untuk monitoring dan memberikan informasi salah satu jenis kendali otomatis ber basis *Human Machine Interface* (HMI), adapun manfaat penelitian ini dengan program aplikasi tersebut dapat menambah kontribusi dalam pembelajaran tentang kontrol pada posisi motor DC bagi mahasiswa teknik elektro dengan komunikasi HMI.

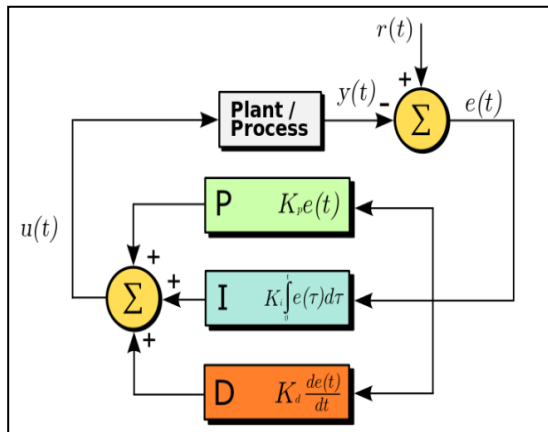
2. METODE PENELITIAN

2.1 Studi Literatur

2.1.1 Kontroler

Kontrol proporsional berfungsi untuk memperkuat sinyal kesalahan penggerak (sinyal *error*), sehingga akan mempercepat keluaran sistem mencapai titik referensi. Hubungan antara *input* kontroler $u(t)$ dengan sinyal *error* $e(t)$, K_p adalah konstanta proporsional. Kemudian Kontrol integral pada prinsipnya bertujuan untuk menghilangkan kesalahan keadaan tunak (*offset*) yang biasanya dihasilkan oleh kontrol proporsional. selanjutnya Kontrol derivatif dapat disebut pengendali laju, karena *output* kontroler sebanding dengan laju perubahan sinyal *error*.

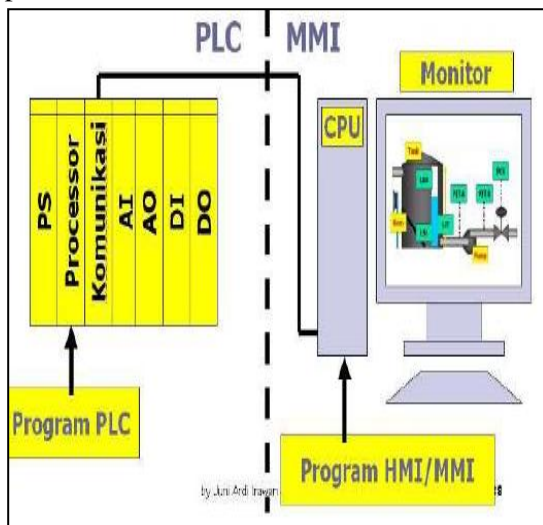
Kontrol derivatif tidak akan pernah digunakan sendirian, karena kontroler ini hanya akan aktif pada periode peralihan. Pada periode peralihan, kontrol derivatif menyebabkan adanya redaman pada sistem sehingga lebih memperkecil lonjakan. Seperti pada kontrol proporsional, kontrol derivatif juga tidak dapat menghilangkan *offset*. gabungan dari ketiga kontroler tersebut menjadi kontrol PID [1][4]. Diagram Blok dari kontrol PID ditunjukkan pada Gambar 1.



Gambar 1. Kontrol PID

2.1.2 HMI(*human machine interface*)

Human Machine Interface (HMI) merupakan sebuah media antarmuka yang memfasilitasi operator dengan memberikan kemampuan melakukan perubahan pada perangkat di lapangan Seperti ditunjukkan pada Gambar 2.



Gambar 2. Konfigurasi HMI

Data yang digunakan pada tampilan HMI dapat bersumber dari *Object Linking and Embedding for Process Control (OPC) server* maupun *Dynamic Data Exchange (DDE) server* dengan demikian HMI dapat bersifat sebagai *OPC client* ataupun *DDE client*. Dalam pengoperasinya, perangkat HMI dibagi menjadi dua kategori yaitu *Operator Workstation* dan *Engineering Workstation*. sistem konfigurasi antara perangkat PLC dan HMI [5].

2.1.3 *Programable Logic Controller (PLC)*

Di dalam dunia modern yang mengutamakan kenyamanan dan kecepatan, sistem yang bekerja secara otomatis akan semakin banyak. Otomatis sering kali diartikan sebagai “tidak menggunakan tenaga manusia”. Pada kenyataannya adalah sebuah kondisi, teknik, dan peralatan yang dioperasikan secara otomatis. Latar belakang tersebut yang mendorong dunia industri untuk meningkatkan sistem otomatis dalam membuat produk yang besar dan waktu yang sedikit. Salah satu pengendali yang paling populer dalam industri, khususnya yang bekerja secara sekuensial, ialah PLC merupakan perangkat elektronik yang di desain untuk digunakan pada industri yang mengontrol suatu sistem ataupun sekelompok sistem baik data I/O analog atau digital [6]. Seperti ditunjukkan pada Gambar 3.



Gambar 3. PLC Modicon TM221CEXXR

Pada awalnya, PLC digunakan untuk menggantikan fungsi relay yang banyak digunakan pada lingkungan industri. PLC (*Programable*, menunjukkan kemampuannya dapat diubah-ubah sesuai program yang dibuat dan kemampuannya dalam hal memori program yang telah dibuat. Logic, menunjukkan kemampuannya dalam memproses input secara aritmatik, yakni melakukan operasi negasi, mengurangi, membagi, mengalikan, menjumlahkan & membandingkan. Controller, menunjukkan kemampuannya dalam mengontrol dan mengatur proses sehingga menghasilkan keluaran yang diinginkan). keunggulan yang signifikan, karena sebuah perangkat pengontrol yang sama dapat digunakan dalam beraneka ragam sistem kontrol. PLC serupa dengan komputer

namun, bedanya : komputer dioptimalkan untuk tugas-tugas perhitungan dan penyimpanan data, sedangkan PLC dioptimalkan untuk tugas pengontrolan dan pengoperasian di dalam industri[7]. Oleh karena itu, PLC memiliki karakteristik berikut :

- Kokoh dan dirancang untuk tahan terhadap getaran, suhu, kelembaban, dan kebisingan.
- Antarmuka untuk masukan dan keluaran telah tersedia secara *built-in*. Agar dapat menjalankan fungsinya sebagai peralatan kontrol, PLC harus diprogram dengan fungsi kontrol yang diinginkan.

PLC diprogram menggunakan *ladder* diagram pada perangkat lunak pemrograman yang dibutuhkan. Pada PLC M221, perangkat lunak yang digunakan untuk pemrograman adalah *SoMachine Basic*. Pada perangkat lunak ini terdapat tiga pilihan bahasa pemrograman berdasarkan IEC6113 1-3 programming languages, yaitu IL (*Instruction List*), dan LD (*Ladder Diagram*). Dalam pemrograman PLC ini, bahasa yang digunakan dapat dipilih salah satu baik IL ataupun LD. - *Instruction List* (IL) Sistem pemrograman ini bersifat tekstual [8].

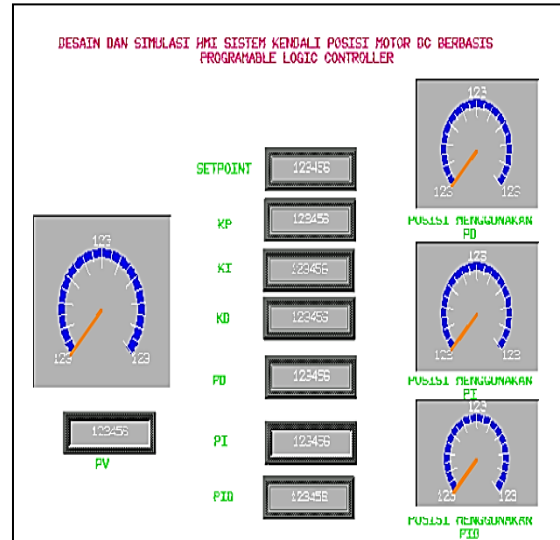
Singkatan-singkatan khusus yang disebut mnemonic digunakan untuk mengidentifikasi perintah yang berbeda yang sedang dijalankan ataupun tidak. Bahasa yang biasa digunakan adalah *OR*, *AND*, *NAND*, *XOR*, dan sebagainya.

2.2 Metode Penelitian

Berdasarkan pada tujuan yang ingin dicapai dalam penelitian ini, metode penelitian yang digunakan dalam penyusunan ini adalah sebagai berikut :

- a. Studi literatur, yaitu dengan cara mendapatkan data buku dan jurnal yang berkaitan dengan permasalahan yang dibahas dalam penelitian ini.
- b. Perancangan sistem *software vijeo designer basic*, yaitu tahap yang bertujuan untuk mencari model HMI yang akan dibuat. ditunjukkan pada Gambar 4.

- c. Perancangan sistem PLC ladder diagram menggunakan *function blok* .
- d. Ujicoba /pengujian berdasarkan hasil rancangan .
- e. Analisa hasil percobaan berdasarkan variabel yang diteliti
- f. Kesimpulan



Gambar 4. Model HMI

3. HASIL DAN PEMBAHASAN

3.1 Hasil

Hasil penelitian. pada pengujian sistem desain dan simulasi *human machine interface* (HMI) ini telah dibuat sistem plant antara lain

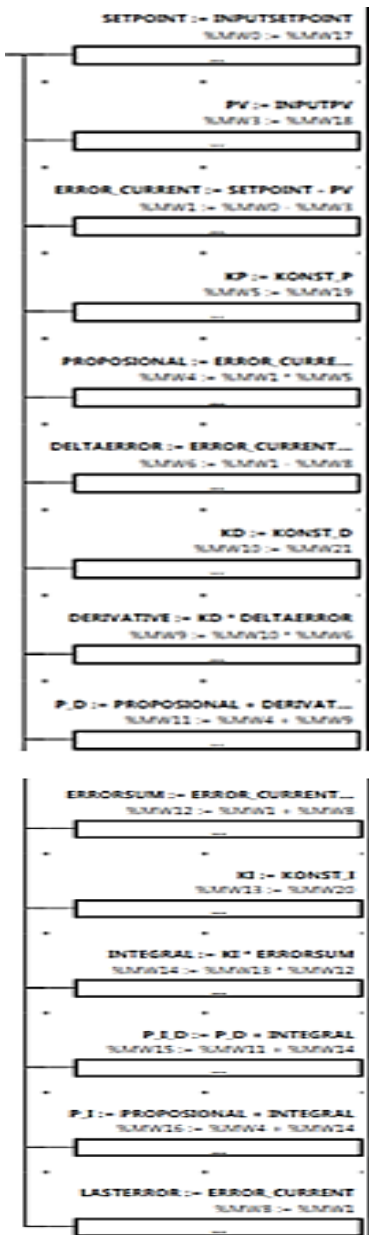
1. Dengan kontroler *Proportional-Integral*
2. Dengan kontroler *Proportional derivative*
3. Dengan kontroler PID

Pada Gambar 5 merupakan hasil ujicoba menggunakan HMI desain dengan menggunakan kontroler *proportional-integral* dan *derivative*. Dengan menggunakan nilai K_p , K_i dan K_d dan dilengkapi setpoint dan Proses Variabel (PV).



Gambar 5. Hasil program tampilan HMI

Kemudian hasil program ladder mengguna-
 kan PLC dalam bentuk ladder diagram
 ditunjukkan pada Gambar 6



Gambar 6. Hasil Ladder Diagram PLC

3.2 Pembahasan

Dalam penelitian ini menggunakan *pseudo*
 untuk kontroller yang diimplementasikan
 pada pemograman ladder diagram di PLC
 adalah sebagai berikut:

Pid:

$$Error(current) = Setpoint - PV$$

$$P = Error(current) * Kp$$

$$Error(sum) = Error(current) + Error(last)$$

$$I = Ki * Error(sum)$$

$$Error(delta) = Error(current) - Error(last)$$

$$D = Kd * Error(delta)$$

$$Drive1 = P + I$$

$$Drive2 = P + D$$

$$Drive3 = P + I + D$$

$$Error(last) = Error(current)$$

Untuk merancang sistem kontrol,
 dilakukan dengan metoda *trial & error*. Untuk
 men dapatkan aksi kontrol yang baik
 diperlukan langkah dengan kombinasi antara
 P, I dan D sampai ditemukan nilai Kp, Ki dan
 Kd seperti yang diinginkan. Pengujian
 dilakukan pada 5 sudut awal yang berbeda
 yaitu sudut 20 derajat, sudut 40 derajat, sudut
 60 derajat, sudut 80 derajat dan 100 derajat.
 Kemudian target sudut yang diinginkan 40
 derajat, sudut 60 derajat, sudut 80 derajat,
 100 derajat dan 120 derajat.

Berdasarkan hasil dari pengujian maka
 akan di peroleh suatu kesimpulan pengaruh
 dari perubahan parameter pada yaitu
Proportional, Integral, dan Derivative [9].
 Pengaruh pada respon sistem menjadi suatu
 acuan untuk melakukan tuning, dengan
 melihat respon dari sistem maka secara teori
 maupun melihat dari percobaan sebelumnya
 maka dapat ditetapkan suatu parameter yang
 tepat. Kemudian hasil Tabel 1 adalah
 pengujian dengan menggunakan dengan
 kontroler *Proportional-Integral*

Tabel 1. Tabel Pengujian *Proportional-Integral*

Sudut awal	Target	Pembacaan pada busur derajat
20°	40°	41°
40°	60°	63°
60°	80°	85°
80°	100°	105°
100°	120°	125°

Pada Tabel 1 merupakan tabel pengujian
 dengan menggunakan kontroler proporti onal-
 integral. Dengan menggunakan nilai Kp
 sebesar 2 dan Ki bernilai 1. Saat mengguna

kan kontroler *PI* respon bisa mencapai *setpoint*. Meskipun dalam Tabel 1 sistem yang masih kurang dari target yang diinginkan dengan Error pembacaan sudut pada busur derajat berkisar antara 1° sampai 5°.

Selanjutnya Pada Tabel 2 merupakan tabel pengujian dengan menggunakan kontroler *proportional derivative*. Dengan menggunakan nilai K_p sebesar 2 dan K_d bernilai 1. Saat menggunakan kontroler *PD* respon bisa mencapai *setpoint*. Kemudian hasil pengujian pada Tabel 2 dengan kontroler *Proportional-derivative*.

Tabel 2. Pengujian *Proportional-Derivative*

Sudut awal	Target	Busur Derajat
20°	40°	40°
40°	60°	60°
60°	80°	80°
80°	100°	100°
100°	120°	120°

Kemudian hasil Tabel 3 dengan kontroler *Proportional-Integral-derivative*, dengan kontroler *PID*. Pada Tabel 3 merupakan tabel pengujian dengan menggunakan kontroler *Propor tional-Integral-derivative*. Dengan menggunakan nilai $K_p = 2$, $K_i = 1$ dan $K_d = 1$.

Saat menggunakan kontroler *PID* respon bisa mencapai *setpoint*. Dari pengujian beberapa kontroler didapatkan penggunaan kontroler *PID* dan *PD* lebih bagus dibanding dengan *PI* dalam simulasi pengendalian posisi motor *DC*. Dan respon *PD* sangat cepat, hal ini sesuai dengan pernyataan Pambudi,dkk menjelaskan bahwa tanpa *Derivative Controller* respon *system* menjadi semakin lambat dan tidak mencapai *steady state* [10].

Tabel 3. Pengujian Kontroler *PID*

Sudut awal	Target	Busur Derajat
20°	40°	40°
40°	60°	60°
60°	80°	80°
80°	100°	100°
100°	120°	120°

4. KESIMPULAN

Dengan metoda *trial & error System* yang memiliki respon terbaik adalah *PID* $K_p = 2$; $k_i = 1$; $k_d = 1$ dan *PD* untuk kontrol posisi dengan hasil $K_p = 2$; $k_d = 1$. dan *PI* untuk kontrol posisi dengan hasil $K_p = 2$; $k_i = 1$.

Pengembangan metode pembelajaran materi perancangan sistem kontrol pada mata kuliah sistem kontrol dengan simulasi komputer dapat membantu mahasiswa dalam memahami materi sistem kontrol.

5. DAFTAR PUSTAKA

- [1] Ogata, Katsuhiko. 1991. Teknik Kontrol Automatik—terjemahan: Ir. Edi Laksono, Jakarta: Erlangga
- [2] Muhammad Rizki Setiawan, M. Aziz Muslim dan Goegoes Dwi Nusantoro Kontrol Kecepatan Motor *DC* Dengan Metode *PID* Menggunakan Visual Basic 6.0 Dan Mikrokontroler *ATmega 16* Jurnal *EECCIS* Vol. 6, No. 2, Desember 2012
- [3] Christin P.R. Tuuk, Vecky C. Poekoel, Jane Litouw, Implementasi Pengendali *PID* Untuk Kestabilan Posisi Terbang Wahana Tanpa Awak, Jurnal Teknik Elektro dan Komputer Vol. 7 No. 1 (2018), ISSN : 2301 – 8402
- [4] Norman S. Nise, 2011, Control Systems Engineering, Sixth Edition, John Wiley & Sons, Inc California State Polytechnic University, Pomona.
- [5] Albert Sopurta, Parsaulian Siregar dan Estiyanti Ekawati, “Perancangan Sistem Simulasi *HYSYS & Integrasi* dengan *Programmable Logic Controller-Human Machine Interface* : Studi Kasus pada Plant Kolom Distilasi Etanol-Air,” J.Oto.Ktrl.Inst (J.Auto.Ctrl.Inst) Vol 6 (1), 2014 ISSN : 2085-2517.
- [6] Modicon M221 Logic Controller Hardware Guide, 2017.

- [7] Modicon M221 Logic Controller Industrial Automation and Control, 2017.
- [8] Modicon M221 Logic Controller Programming Guide, 2017.
- [9] Isnan Nur Rifai, Fathan Rozani, Pengaturan Sudut pada Modul Kendali Posisi Motor Secara Otomatis Menggunakan Kendali PID Descrete, Jurnal *Amplifier* Vol. 5 No. 1, Mei 2015, ISSN: 2089-2020
- [10] Pambudi, Wahyu & Putra, Jan & Pelawi. (2015). Simulasi Folding Machine Dengan PID, P, PI, PD Dan Fuzzy-PD . 1. 2460-173. *Jurnal Sains dan Teknologi ISSN: 2460-173X Volume 1, Nomor 1*