

Sistem Pengendalian Level Cairan Tinta Printer Epson C90 Sebagai Simulasi Pada Industri Percetakan Menggunakan Kontroler PID

Firda Ardyani, Erni Yudaningtyas, dan M. Aziz Muslim.

Abstrak— Pengendalian ketinggian cairan tinta yang dilakukan dalam penelitian ini memberikan keuntungan pada perusahaan percetakan yang memiliki skala besar untuk mengatasi kerumitan saat mengisi ulang cairan tinta serta untuk menghindari kelalaian pengisian cairan tinta karena terbatasnya waktu, dan kurangnya pengawasan tersebut dibutuhkan sebuah sistem otomatisasi pengisian cairan tinta dapat mengurangi terjadinya *maintenance* pada mesin cetak. Metode yang digunakan untuk pengendalian ketinggian tinta adalah Proporsional Integral Differensial (PID). Hasil identifikasi sistem dengan menggunakan MATLAB R2010A didapatkan fungsi alih sistem $\frac{3.185s+0.0579}{s^2+7.752s+0.11343}$.

Proses perancangan parameter PID digunakan metode *root locus* dan didapatkan bahwa semua akar persamaan berada disebelah kiri bidang s , sehingga memenuhi syarat kestabilan. Hasil perhitungan parameter PID dengan pole $s = -38$ didapatkan nilai parameter PID terbaik yaitu $K_p=19.14$, $K_i=2.5$ dan $K_d=0.25$ dengan *time settling* (ts) waktu yang dibutuhkan untuk mempertahankan ketinggian tinta sebear 15.34 detik.

Kata Kunci—Pengendalian Ketinggian, PID, *root locus*.

I. PENDAHULUAN

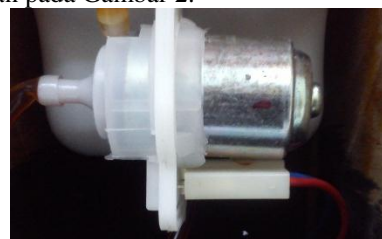
Kemajuan di bidang percetakan dewasa ini semakin meningkat dengan banyaknya permintaan akan spanduk, banner, kartu nama, brosur untuk mempromosikan sebuah event, keperluan promosi sebuah perusahaan, keperluan pelajar bahkan sampai keperluan politik dan masih banyak lagi. Pengisian cairan tinta pada tabung printer memerlukan waktu yang lama dan harus dilakukan sedikit demi sedikit, serta ketika cairan tinta habis dan pengawasan manusia yang terbatas tentunya dapat menghambat hasil serta tidak dapat mengefisisensikan waktu dalam pemenuhan kebutuhan konsumen. Oleh karena itu diperlukan sebuah sistem pengendalian ketinggian cairan tinta, agar dapat mengefisiensi waktu, menghindari kelalaian ketika cairan tinta habis dan agar tidak sering dilakukan *maintenance*. Pengendalian cairan tinta dilakukan agar pemilik tidak perlu melakukan pengisian tinta secara manual dan ketinggian tetap dipertahankan pada ketinggian yang diinginkan. Salah satu sistem pengendalian yang dapat dilakukan adalah PID. Keunggulan dari sistem kontrol PID adalah respon sistem lebih cepat dalam mengambil keputusan dan mudah dalam analisis.

Tujuan dari penelitian ini adalah untuk membuat sistem pengendalian ketinggian cairan tinta dengan menggunakan kontroler PID.

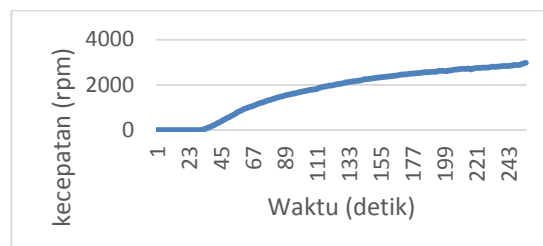
II. IDENTIFIKASI SISTEM

A. Motor Pompa

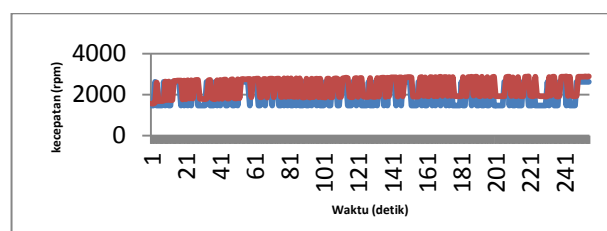
Motor pompa yang digunakan pada perancangan ini adalah Motor DC magnet permanen yang digunakan untuk memompa cairan tinta dari tandon besar ketabung infus. Karakteristik motor DC pompa yang digunakan ditunjukkan pada Gambar 2.



Gambar 1. Motor Pompa



Gambar 2. Karakteristik motor DC



Gambar 3. Sinyal prbs dan rpm



Gambar 4. Identifikasi dengan ident pada MATLAB

Fungsi alih motor DC diperoleh dengan cara membangkitkan sinyal prbs dari mikrokontroler yang

ditunjukkan pada Gambar 3. Setelah didapatkan sinyal prbs dilakukan identifikasi dengan *ident* pada MATLAB yang ditunjukkan pada Gambar 4.

Data identifikasi yang digunakan, PRBS sebagai input sedangkan kecepatan sebagai output. struktur model yang digunakan adalah *Auto Regresive with Exogenous input* (ARX) dengan estimasi parameter 1 1 2 dan didapatkan fungsi alih motor $\frac{3.64s+0.06623}{s^2+4.567s+0.05553}$.

B. Sensor Ultrasonik PING))

Sensor ultrasonik PING)) yang digunakan pada perancangan alat ini memiliki 3 pin keluaran yaitu pin SIG untuk data, pin VCC untuk masukan sebesar 5V dan pin GND. Fungsi alih dari sensor adalah sebuah konstanta dengan perhitungan :

$$K = \frac{\text{span output}}{\text{span input}} \quad [1]$$

$$\frac{30-0}{40 \times 115} = 0.875 \text{ mA/Hz}$$



Gambar 5. Sensor Ultrasonik PING))

C. Mikrokontroler ATmega32

Mikrokontroler ATmega32 adalah mikrokontroler yang diproduksi oleh Atmel. mikrokontroler ini memiliki clock dan kerjanya tinggi sampai 16 MHz, ukuran flash memorinya cukup besar, kapasistas SRAM sebesar 2 KiloByte, 32 buah port I/O yang sangat memadai untuk berinteraksi dengan LCD dan keypad.

Pada perancangan alat ini mikrokontroler ATmega32 digunakan sebagai Mikrokontroler Utama.

D. Mikrokontroler ATmega8

Mikrokontroler ATmega8 merupakan mikrokontroler keluaran AVR yang merupakan mikrokontroler AVR CMOS 8 bit berdaya rendah. Mikrokontroler ATmega8 dapat mengeksekusi instruksi hingga 1 MIPS per MHz dalam satu siklus waktu.

Pada perancangan alat ini mikrokontroler ATmega8 digunakan sebagai mikrokontroler *slave* sebagai pengendali sensor ultrasonik PING)).

E. Perancangan PID dengan metode Root Locus [2]

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 berikut.



Gambar 6. Sistem Kendali Loop Tertutup

Persamaan karakteristik dari sistem kendali diatas adalah: $1 + Gc(s)Gp(s) = 0$ (1)

Untuk mendapatkan nilai-nilai dari parameter kontroler, harus ditentukan dahulu spesifikasi desain sistem yang diinginkan sehingga nilai pole simpul tertutup (dimisalkan = s_1) yang berkesesuaian dapat dicari. [3]

Misalkan diinginkan lokus akar melalui, maka

$$s = s_1$$

$$Gc(s_1)Gp(s_1) = -1$$

$$Gc(s_1)|Gp(s_1)|e^{j\psi} = 1e^{j\pi}$$

Fungsi alih kontroler PID setelah ditransformasi laplace dinyatakan oleh

$$Gc(s) = Kp + \frac{Ki}{s} + Kd s \quad (2)$$

Sehingga dari persamaan diatas didapatkan

$$Gc(s_1) = \frac{1}{|Gp(s_1)|} e^{j(\pi-\psi)}$$

atau

$$Kd s_1^2 + Kp s_1 + Ki = \frac{e^{j(\pi-\psi)}}{|Gp(s_1)|} \quad (3)$$

Dengan

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

Maka

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

Menyamakan real dengan real dan imajiner dengan imajiner, didapat

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

atau

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

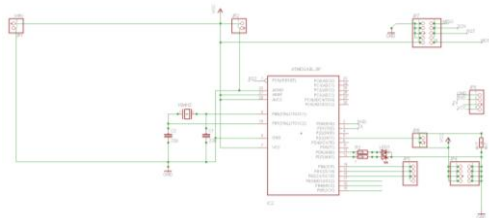
Dari persamaan diaatas dapat dilihat bahwa untuk perancangan kontroler PID, satu dari tiga penguatan Kp, Ki, Kd, harus ditentukan dahulu. Sedangkan untuk perancangan PI atau PD, penguatan yang sesuai pada persamaan diatas dibuat sama dengan nol. [3]

III. PERANCANGAN ALAT

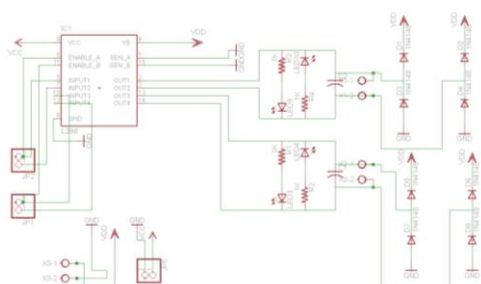
Perancangan ini meliputi pembuatan perangkat keras dan perangkat lunak, perangkat keras meliputi perancangan rangkaian *driver* motor L298N, perancangan rangkaian mikrokontroler utama dan mikrokontroler *slave*. Perancangan perangkat lunak meliputi pembuatan program untuk mikrokontroler utama dan mikrokontroler *slave* untuk pengendalian keseluruhan sistem dengan algoritma PID.

A. Rangkaian Mikrokontroler Utama (ATMega32) dan Mikrokontroler Slave (ATMega8)

Dalam penelitian ini mikrokontroler ATMega32 berfungsi sebagai pengendali utama dalam menjalankan program keseluruhan, sedangkan mikrokontroler ATMega8 berfungsi sebagai mikrokontroler *slave* untuk pengendali sensor (PING)) agar tidak mempengaruhi mikrokontroler utama. Komunikasi antara mikrokontroler utama dan mikrokontroler *slave* menggunakan komunikasi UART.



Gambar 7. Rangkaian Mikrokontroler ATMega8



Gambar 8. Rangkaian Mikrokontroler ATMega32

B. Rangkaian Driver Motor L298N

Pada penelitian ini modul pengendali motor DC menggunakan rangkaian driver dengan menggunakan IC L298N.



Gambar 9. Rangkaian driver motor DC L298N

C. Perancangan Perangkat Lunak

Pada penelitian ini pemrograman keseluruhan sistem menggunakan *software CodeVisionAVR V2.05.0*. Perancangan program dilakukan setelah didapat perhitungan seluruh algoritma dari hasil identifikasi.

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 7 dengan fasilitas ident yang tersedia.

A. Pengujian Driver Motor

Pengujian ini bertujuan untuk mengetahui kinerja dan respon dari rangkaian *driver* pengendali motor DC L298N dengan membandingkan dan menguji sinyal keluaran dari *driver* motor terhadap sinyal masukan

PWM yang diberikan oleh mikrokontroler. Hasil pengujian driver motor ditunjukkan pada tabel 1.

Berdasarkan tabel 1, terdapat selisih rata – rata *duty cycle* keluaran *driver* dengan mikrokontroler sebesar 4.73%. Hal ini disebabkan karena adanya akumulasi *delay* pada seluruh rangkaian. Selisih rata – rata sebesar 4.73% cukup bagus karena tidak memberikan pengaruh yang sangat besar terhadap kinerja sistem, sehingga dapat disimpulkan *driver* motor L298N dapat bekerja dengan baik.

Tabel 1. Hasil Pengujian Driver Motor

Duty Cycle (PWM MK)	Duty Cycle (keluaran driver)	Duty Cycle (delta error)
10.2%	15.6%	5.4%
20.3%	25.6%	5.3%
30.1%	35.5%	5.4%
40.3%	45.5%	5.2%
50%	55%	5%
60.1%	65.2%	5.1%
70%	74.9%	4.9%
80%	84.5%	4.5%
89.8%	93.7%	3.9%
95%	97.6%	2.6%
Selisih rata-rata duty cycle		4.73%

B. Pengujian sensor PING))

Pengujian ini dilakukan apakah pembacaan jarak oleh sensor (PING)) sesuai dengan jarak yang sesungguhnya. Hasil pengujian sensor (PING)) ditunjukkan pada tabel 2.

Tabel 2. Hasil Pengujian Sensor PING))

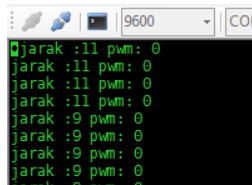
Pengujian ke-	Jarak Uji (cm)	Sensor PING))A	Sensor PING))B	Sensor PING))D
1	5	5	5	5
2	6	6	6	6
3	7	7	7	7
4	8	8	8	8
5	9	9	9	9
6	10	10	10	10

Sumber: Perancangan

Berdasarkan tabel 3 dari data hasil pengujian pembacaan sensor (PING)) dapat dilihat bahwa sensor (PING)) memiliki rata – rata kesalahan pembacaan sebesar 0% , sehingga dapat disimpulkan sensor (PING)) memiliki keakuratan pembacaan yang bagus.

C. Pengujian Komunikasi Serial UART ke PC

Pengujian ini dilakukan untuk mengetahui apakah rangkaian mikrokontroler dapat mengirim data melalui komunikasi serial UART. Hasil pengujian ditunjukkan pada gambar 10.

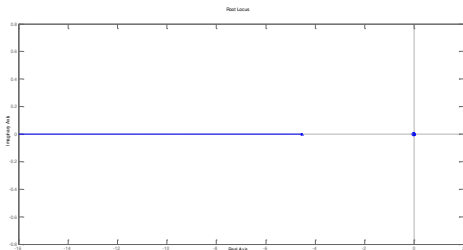


Gambar 10. Pengujian Komunikasi UART MK dan PC

Dari hasil pengujian komunikasi serial UART antara mikrokontroler dan PC dengan kabel serial RS-232, dapat disimpulkan bahwa data dapat dikirim dengan baik tanpa ada karakter yang hilang dan program pembacaan berjalan dengan benar

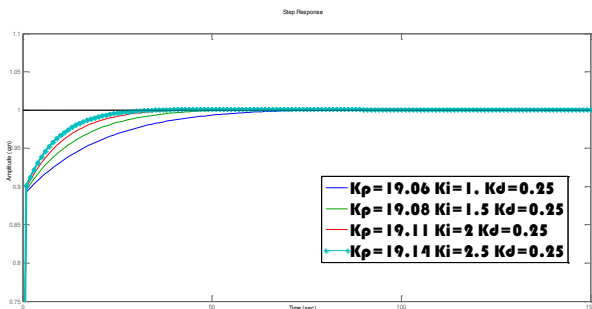
D. Penentuan Parameter PID

Untuk mendapatkan parameter PID yang diinginkan terlebih dahulu ditentukan nilai dari s_1 yang nilainya ada pada diagram root locus. nilai *pole* yang dipilih adalah yang terletak pada sebelah kiri bidang *s*, dan didapatkan nilai s_1 yang terbaik dan memenuhi karakteristik sistem adalah $s_1 = -38$.



Gambar 11. Diagram Root Locus Fungsi Alih Sistem

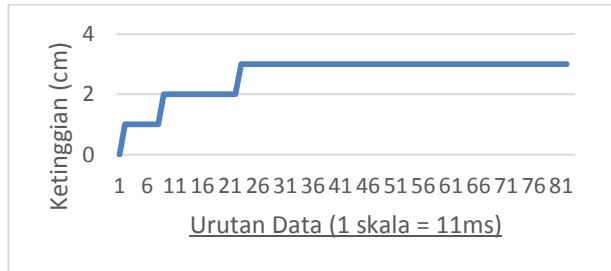
Selanjutnya nilai s_1 hasil perhitungan dimasukkan pada persamaan 4, dengan menggunakan listing program pada MATLAB didapatkan respon sistem dengan 4 variasi nilai K_p , K_i dan K_d yang ditunjukkan pada gambar 11.



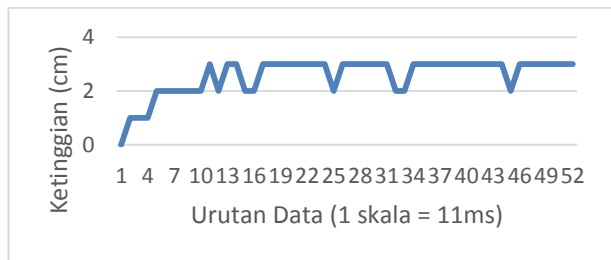
Gambar 11. Grafik Respon dengan Parameter PID Hasil Perhitungan

Berdasarkan Gambar 11 respon sistem dengan $K_p=19.14$, $K_i=2.5$ dan $K_d=0.25$ memiliki respon yang paling baik dengan t_s yang tercepat. Hasil parameter PID hasil perhitungan kemudian diimplementasikan pada sistem dan didapatkan pembacaan jarak terhadap ketinggian level dengan serial RS-232 pada Gambar 12 dan sistem dapat mempertahankan ketinggian dengan $t_s=15.34$ detik. Gambar 13 menunjukkan grafik sistem

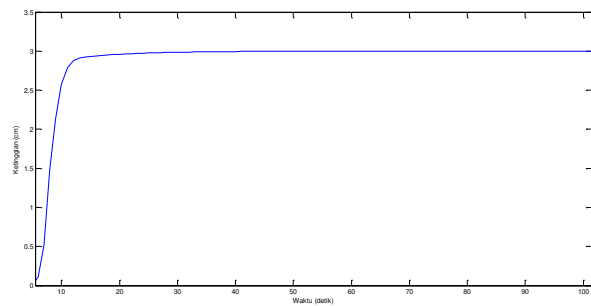
dengan diberikan gangguan dapat dilihat sesuai dengan grafik yang didapatkan sistem dapat mempertahankan ketinggian ketika level tinta mengalami pengurangan dan mencapai ketinggian yang ditentukan. Dapat disimpulkan dengan nilai parameter $K_p=19.14$ $K_i=2.5$ dan $K_d=0.25$ sistem dapat mempertahankan ketinggian level dengan $t_s=15.34$ detik. Gambar 14 menunjukkan keluaran respon dengan kontroler menggunakan simulasi pada MATLAB 2010, dari hasil simulasi respon yang dihasilkan sesuai dengan yang diinginkan dan memiliki e_{ss} (*error steady state*) yang kecil. Sehingga dapat disimpulkan hasil tuning parameter PID yang didapatkan sesuai dengan *plant*.



Gambar 12. Grafik Pembacaan Jarak Terhadap Urutan Data dengan Kontroler PID menggunakan serial RS-232 (Perancangan)



Gambar 13. Grafik Respon Sistem dengan Gangguan (Perancangan)



Gambar 14. Grafik Respon Sistem dengan $K_p=19.14$ $K_i=2.5$ $K_d=0.25$ pada MATLAB 2010 (Perancangan)

V. KESIMPULAN DAN PROSPEK

Dari hasil perancangan dan pengujian alat yang telah dilakukan, maka dapat disimpulkan bahwa dengan menggunakan metode root locus untuk menentukan nilai penguat kontroler PID, didapatkan parameter yang terbaik dengan $K_p=19.14, K_i=2.5, K_d=0.25$.

Setelah diimplementasikan pada sistem, sistem dapat mempertahankan ketinggian tinta dengan waktu atau *time settling (ts)* selama 15.34 detik. Dengan keterbatasan fisik yang ada pada sistem diperlukan adanya pemikiran lebih lanjut untuk menentukan parameter PID.

DAFTAR PUSTAKA

- [1] Ikrom, Hassanal. 2008. Perancangan Kontroler Pid-Kaskade Dengan Metode Root Locus Untuk Kontrol Temperatur Dan Tekanan Pada Proses Evaporator. Laporan Skripsi, Teknik Elektro Brawijaya
- [2] Philip, C. L. & Harbor, R. D. 1996. *Feedback Control System*. Prentice Hall. New Jersey
- [3] Ikrom, Hassanal. 2008. Perancangan Kontroler Pid-Kaskade Dengan Metode Root Locus Untuk Kontrol Temperatur Dan Tekanan Pada Proses Evaporator. Laporan Skripsi, Teknik Elektro Brawijaya