

Makalah Seminar Tugas Akhir
Perancangan *Simulator* Pengendalian Posisi *Turret*
Pada Mobil Pemadam Kebakaran

Isma Candra Jati Kusuma*, Trias Andromeda, ST.MT**, Darjat, ST, MT.**

Abstrak - *Turret* merupakan salah satu perangkat utama pada mobil pemadam kebakaran. Posisinya berada di atas mobil pemadam kebakaran, bentuknya seperti *cannon* pada tank militer. Perangkat *Turret* digunakan untuk mengarahkan air pada posisi yang diinginkan. *Turret* dapat bergerak dalam dua sumbu, sumbu horisontal dan sumbu vertikal. Pergerakan *Turret* dikendalikan oleh seorang operator dari dalam mobil dengan kendali *joystick*.

Sistem kerja *Turret* mirip sistem kerja pada kendali posisi dengan posisi referensi dapat diubah-ubah melalui sebuah *joystick*. Kendali PI (*Proporsional-Integral*) merupakan kendali yang sudah sering dipakai dalam pengendalian posisi. Perubahan pada posisi referensi dari *joystick* mengakibatkan *Turret* akan mengikuti perubahan posisi tersebut. Dengan memanfaatkan mikrokontroler *ATMega8535* sebagai pusat pengendali dan motor DC sebagai penggerak *Turret* dirancang sebuah *simulator* untuk mempelajari proses kerja *Turret*.

Dalam Tugas Akhir ini dirancang sebuah *simulator* pengendalian posisi *Turret*. Motor DC diaplikasikan sebagai aktuator/penggerak. Penelitian dilakukan pada unjuk kerja *simulator* untuk mencapai posisi yang diinginkan.

Kata kunci: *Turret*, Pemadam kebakaran, Kendali PI

I. PENDAHULUAN

1.1 Latar Belakang

Turret merupakan salah satu perangkat utama pada mobil pemadam kebakaran. Posisinya berada di atas mobil, bentuknya seperti *cannon* pada tank militer. Perangkat *Turret* digunakan untuk mengarahkan air pada posisi yang diinginkan. *Turret* dapat bergerak dalam dua sumbu, sumbu horisontal dan sumbu vertikal. Pergerakan *Turret* dikendalikan oleh seorang operator dari dalam mobil dengan kendali *joystick*.

Sistem kerja *Turret* mirip sistem kerja pada kendali posisi dengan posisi referensi dapat diubah-ubah melalui sebuah *joystick*. Kendali PI (*Proporsional-Integral*) merupakan kendali yang sudah sering dipakai dalam pengendalian posisi. Perubahan pada posisi referensi dari *joystick* mengakibatkan *Turret* akan mengikuti perubahan posisi tersebut. Dengan memanfaatkan mikrokontroler *ATMega8535* sebagai pusat pengendali dan motor DC sebagai penggerak *Turret* dirancang sebuah *simulator* untuk mempelajari proses kerja *Turret*.

1.2 Tujuan

Tujuan yang hendak dicapai dalam Tugas Akhir ini adalah merancang sebuah *simulator* pengendalian posisi *Turret* yang digunakan untuk mempelajari sistem kerja *Turret* pada mobil pemadam kebakaran.

1.3 Pembatasan Masalah

Dalam pembuatan Tugas Akhir ini penulis membatasi permasalahan sebagai berikut :

1. *Plant* yang diuji adalah *plant simulator* pengendali posisi *Turret* pada mobil pemadam kebakaran.
2. Pengendalian hanya dilakukan pada pengendalian posisi *Turret*.
3. Tidak dilakukan pengendalian pada banyaknya air yang keluar lewat *Turret*.
4. Metode pengendalian menggunakan kendali PI (*proporsional-integral*).
5. Sistem dengan kendali ON-OFF digunakan sebagai pembanding.
6. Menggunakan motor DC sebagai penggerak (aktuator).
7. Tidak dilakukan pembahasan mendalam tentang mekanik *Turret*.
8. Mikrokontroler yang digunakan adalah mikrokontroler *ATMega 8535*.
9. Pembuatan perangkat lunak menggunakan bahasa *C-embedded*.

II. DASAR TEORI

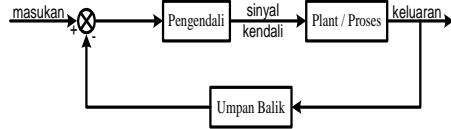
2.1 Sistem Pengendalian *Loop* Tertutup^[4]

Sistem pengendalian *loop* tertutup adalah sistem pengendalian dimana sinyal

* Mahasiswa Teknik Elektro UNDIP

** Dosen Teknik Elektro UNDIP

keluaran mempunyai pengaruh langsung terhadap aksi pengendalian. Sistem pengendalian *loop* tertutup melibatkan umpan balik negatif. Secara umum, diagram blok sistem pengendalian ini terlihat pada Gambar 2.1.



Gambar 2.1 Diagram blok sistem pengaturan *loop* tertutup

2.2 Pengendali PI (Proporsional-Integral)

Pengendali PI adalah suatu sistem pengendali yang merupakan gabungan antara pengendali *proporsional* dan *integral*. Dalam waktu kontinu, sinyal keluaran pengendali PI ditunjukkan pada Persamaan (2.1)^[4] sebagai berikut.

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt \right) \quad (2.1)$$

atau

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt \quad (2.2)$$

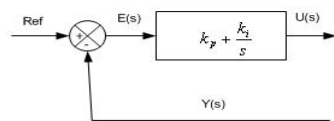
dengan

- $u(t)$ = sinyal keluaran kendali PI
- K_p = konstanta proporsional
- T_i = waktu integral
- K_i = konstanta integral
- $e(t)$ = sinyal kesalahan ($e(t)$)

Fungsi alih pengendali PI (dalam domain s) ditunjukkan pada Persamaan (2.3)^[4].

$$G_c(s) = K_p + \frac{K_i}{s} \quad (2.3)$$

Diagram blok pengendali PI^[4] dapat dilihat pada Gambar 2.2.



Gambar 2.2 Diagram blok pengendali PI

2.2.1 Pengendali Proporsional^[4]

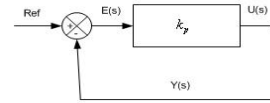
Pengendali proporsional berfungsi untuk memperkuat sinyal *error*, sehingga akan mempercepat keluaran sistem mencapai referensi. Hubungan antara keluaran pengendali proporsional dengan sinyal *error* adalah

$$u(t) = K_p e(t) \quad (2.4)$$

dengan transformasi Laplace, diperoleh fungsi alih

$$\frac{U(s)}{E(s)} = K_p \quad (2.5)$$

dengan K_p adalah konstanta proporsional. Diagram blok pengendali proporsional ditunjukkan pada Gambar 2.3.



Gambar 2.3 Diagram blok pengendali proporsional.

2.2.2 Pengendali Integral^[4]

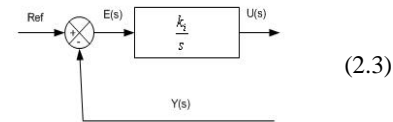
Pengendali integral pada prinsipnya bertujuan untuk menghilangkan kesalahan keadaan tunak atau *offset*. *Offset* biasanya terjadi pada *plant* yang tidak mempunyai faktor integrasi. Hubungan antara keluaran pengendali integral dengan sinyal *error* adalah

$$U(t) = K_i \int_0^t e(t) dt \quad (2.6)$$

dengan transformasi Laplace, diperoleh fungsi alih

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (2.7)$$

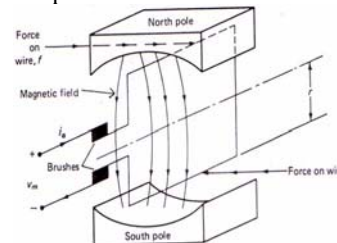
Jika harga $e(t)$ diduakalikan, maka harga $u(t)$ berubah dengan laju perubahan menjadi dua kali semua. Jika *error* nol, maka harga $u(t)$ tetap stasioner. Gambar 2.4 menunjukkan diagram blok kontroler integral.



Gambar 2.4 Diagram blok pengendali integral

2.3 Motor DC^[3]

Motor DC adalah suatu mesin yang berfungsi untuk mengubah tenaga listrik arus searah (DC) menjadi tenaga mekanik (putaran). Motor bekerja berdasarkan prinsip induksi elektromagnetik. Dasar konstruksi motor DC ditunjukkan pada Gambar 2.5.



Gambar 2.5 Dasar konstruksi motor DC

Jika diberikan sumber tegangan V_m maka arus i_a akan mengalir pada kawat dengan arah seperti yang ditunjukkan pada Gambar 2.5. Karena kawat penghantar l berada didalam medan magnet B maka akan timbul gaya f pada kawat. Pada kawat sisi atas akan timbul gaya kekanan dan kawat sisi bawah timbul gaya kekiri. Hasilnya akan timbul perputaran searah jarum jam pada lilitan jangkar atau *armature*. Jika polaritas dibalik maka timbul perputaran melawan arah jarum jam. Prinsip ini sesuai dengan Persamaan (2.8)

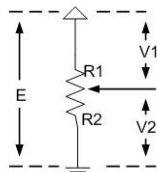
$$f = B \cdot i_a \cdot l \sin \omega t \quad (2.8)$$

di mana:

f = Gaya Lorentz
 B = Medan magnet
 i_a = Arus pada kawat penghantar
 l = Panjang kawat penghantar

2.4 Potensiometer^[9]

Potensiometer digunakan untuk merubah besarnya pergerakan *linear* maupun *angular*. Prinsip kerja potensiometer sebagai sensor posisi ditunjukkan pada Gambar 2.6.



Gambar 2.6 Prinsip kerja potensiometer
 Pada Gambar 2.6 berlaku persamaan:

$$V2 = \frac{R2}{R1 + R2} E \quad (2.9)$$

di mana:

E = sumber tegangan, volt.
 $V1$ = tegangan pada R1, volt.
 $V2$ = tegangan pada R2, volt.
 $R1$ = tahanan resistor atas, ohm.
 $R2$ = tahanan resistor bawah, ohm.

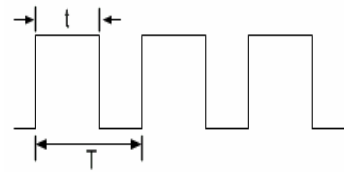
2.5 PWM (Pulse Width Modulation)^[11]

PWM adalah salah satu cara mengatur kecepatan motor DC dengan menggunakan sumber tegangan DC yang tetap, yaitu dengan men-*switch* secara cepat antara kondisi ON dan OFF pada frekuensi tertentu. Kondisi ON (logika *high*) disebut sebagai lebar pulsa yang akan diatur t , dan periode sinyal adalah T , dimana:

$$T = \frac{1}{f} \quad (2.10)$$

di mana:

T = periode gelombang kotak
 f = frekuensi gelombang kotak



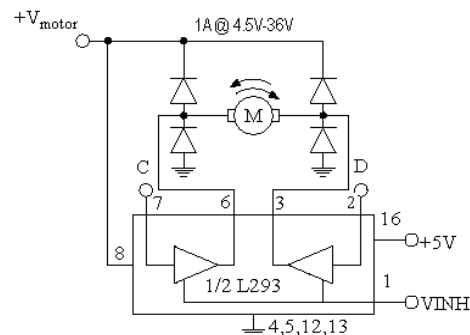
Gambar 2.7 Sinyal PWM

Rasio perbandingan antara kondisi ON dengan periode sinyal disebut sebagai *duty cycle*, dinyatakan dengan:

$$\text{duty cycle} = \frac{t}{T} \times 100\% \quad (2.11)$$

2.6 Driver Motor DC L293D^[11]

Driver motor DC ini merupakan *driver* motor DC dua arah yang bisa menggerakkan motor untuk bergerak maju atau mundur sekaligus. Pada IC ini membutuhkan power suplai (V_{cc}) sebesar 4,5-36 Volt dan arus sebesar 1A. Pada IC ini juga disediakan pin khusus untuk mencatu motor secara langsung. V_{motor} yang bisa digunakan pada IC ini adalah 4,5-36 Volt. Setengah dari rangkaian *internal* L293D adalah seperti berikut:



Gambar 2.8 Driver motor DC L293D

Berdasarkan Gambar 2.8, logika pengendalian motornya^[10] seperti yang disajikan pada Tabel 2.1:

Tabel 2.1. Tabel Logika L293D

	Input	Fungsi
Vinh Tinggi	C = Tinggi ; D = Rendah	Putar kanan
	C = Rendah ; D = Tinggi	Putar kiri
	C = D	Berhenti
Vinh Rendah	C = X ; D = X	Berhenti

Keterangan:

- Rendah = 0 – 1,5 Volt
- Tinggi = 2,3 – 36 Volt
- X = *Don't Care*

2.7 Mikrokontroler ATmega 8535^[1]

Semua jenis AVR dilengkapi dengan *flash memory* sebagai memori program. Kapasitas dari *flash memory* ini berbeda antara *chip* yang satu dengan *chip* yang lainnya tergantung dari jenis IC-nya.

AVR mempunyai 2 jenis *timer* yang cara kerjanya berlainan, yakni *Timer* 8 bit dan *Timer* 16 bit. Frekuensi untuk kedua macam *timer* ini bisa diatur terpisah. Sebagai *timer*, frekuensi penggerak berasal dari osilator kristal yang frekuensinya diturunkan dengan faktor 1X, 8 X, 64X, 256X, atau 1024X. Sebagai *counter* sinyal yang dicacah diumpankan pada salah satu kaki AVR (kaki T0 atau T1).

Selain dipakai sebagai *timer/counter* selayaknya, *timer* AVR bisa dipakai untuk keperluan pembangkitan sinyal PWM secara perangkat keras, artinya cukup mengatur beberapa register untuk keperluan PWM, sinyal PWM akan disalurkan ke kaki OC1.

Pada dasarnya sinyal PWM merupakan semacam *digital to analog converter* dengan memasang rangkaian *low pass filter* sederhana pada kaki OC1, bisa diperoleh tegangan analog yang nilainya bisa ditentukan lewat program. Dengan menggunakan teknik pemrograman tertentu, bisa pula diperoleh tegangan sinus dengan berbagai frekuensi yang dikehendaki.

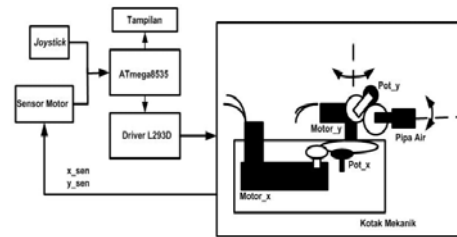
Sarana berharga lainnya adalah *Analog Digital Converter* dengan ketelitian 10 bit lebih baik dari *analog digital Converter* 0808 yang banyak dipakai yang ketelitiannya hanya 8 bit.

Sistem dengan RISC pada AVR mengurangi hampir semuanya, yaitu meliputi jumlah instruksi, mode pengalamanan, dan format. Hampir semua instruksi mempunyai ukuran yang sama yaitu 16 bit. Sebagian besar instruksi dieksekusi dalam satu siklus CPU.

III. PERANCANGAN PERANGKAT KERAS DAN PERANGKAT LUNAK

3.1 Perancangan perangkat keras

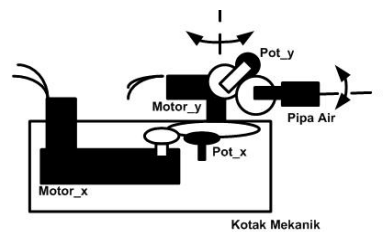
Perancangan *simulator* pengendalian posisi *Turret* pada mobil pemadam kebakaran ini digunakan sebuah mikrokontroler ATmega8535 sebagai pusat kendali, sarana masukan berupa sensor posisi motor DC, masukan referensi *joystick*, dan sebuah *switch*, sarana keluaran dan *driver* motor DC. Diagram blok perangkat keras secara keseluruhan ditunjukkan pada Gambar 3.1.



Gambar3.1 Diagram blok *simulator*

3.1.1 Mekanik Simulator

Mekanik *simulator* dibangun oleh dua buah motor DC, sensor-sensor posisi motor DC, dan rangka mekanik. Rancangan mekanik *simulator* ditunjukkan pada Gambar 3.2.



Gambar 3.2 Mekanik *simulator*

Motor_x adalah motor yang difungsikan untuk pergerakan pada sumbu horisontal. Motor_y adalah motor yang difungsikan untuk perputaran pada sumbu vertikal. *x_sen* digunakan untuk mengetahui posisi motor_x (dalam derajat). *y_sen* digunakan untuk mengetahui posisi motor_y (dalam derajat). Sebuah silinder pada bagian depan sebagai lubang selang air.

Perputaran pada sumbu horisontal berada pada *range* -50° sd 50° . Perputaran pada sumbu vertikal berada pada *range* -20° sd 40° .

3.1.2 Joystick

Bagian utama *joystick* dibangun oleh dua buah potensiometer $10k\ \Omega$.

3.1.3 Sensor Posisi Turret

Dalam perancangan *simulator* digunakan potensiometer *wirewound* $10k\ \Omega$.

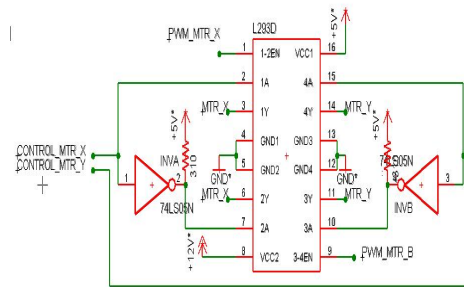
3.1.4 Mikrokontroler ATmega8535

Pada mikrokontroler tertanam algoritma pengendalian *simulator*. Semua sarana masukan dan keluaran sistem terhubung pada mikrokontroler.

3.1.5 Driver L293D

Sebuah IC L293D digunakan sebagai *driver* motor DC. Satu IC L293D mampu untuk

menggerakkan dua buah motor DC. Konfigurasi L293D dalam perancangan ini ditunjukkan pada Gambar 3.3.

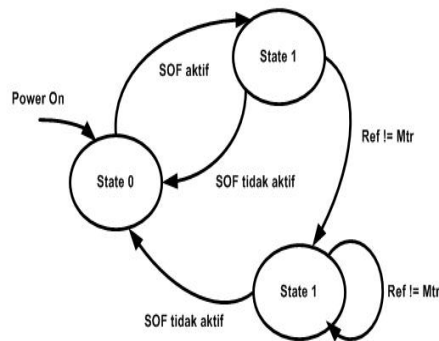


Gambar 3.3 Driver L293D

Pin 1A dan 2A sebagai *pin* masukan sinyal kendali arah putaran motor X, pin 1-2 EN sebagai *pin* masukan sinyal PWM dari mikrokontroler. Pin 3A dan 4A sebagai *pin* masukan sinyal kendali arah putaran motor Y, pin 3-4 EN sebagai *pin* masukan sinyal PWM dari mikrokontroler.

3.2 Perancangan Perangkat Lunak

Ada beberapa proses yang terjadi dalam perancangan *simulator* ini. Masing-masing proses terletak dalam blok-blok fungsi dimana tiap blok fungsi mempunyai tugas masing-masing.



Gambar 3.4 Diagram *state* keadaan *simulator*

State 0 menunjukkan keadaan *simulator* dalam keadaan parkir, *simulator* tidak memberikan tanggapan terhadap perubahan posisi referensi dari *joystick*. *State 1* menunjukkan *turret* pada posisi normal yaitu 0^0 pada sumbu horisontal dan sumbu vertikal. *State 2* menunjukkan *turret* pada posisi yang diinginkan sesuai posisi referensi dari *joystick*. Pada satu waktu, *simulator* hanya berada dalam satu *state*, *simulator* dalam *state 0* atau *simulator* dalam *state 1* atau *simulator* dalam *state 2*. SOF (*switch*

ON-OFF) adalah *switch* ON-OFF yang digunakan untuk mengaktifkan dan menonaktifkan *simulator*.

IV. PENGUJIAN DAN ANALISA

Setelah dilakukan perancangan *simulator* pengendalian posisi *Turret* pada mobil pemadam kebakaran, kemudian dilakukan beberapa pengujian

4.1 PENGUJIAN SENSOR POSISI MOTOR DC DAN JOYSTICK

Pengujian sensor posisi motor DC dilakukan dengan merubah posisi sudut motor maupun *joystick* secara manual, kemudian diukur tegangan keluaran dari sensor yang bersangkutan. Pengujian dilakukan setiap perubahan posisi 10^0 . Hasil pengujian sensor posisi motor DC dan *joystick* ditunjukkan pada Tabel 4.1.

Tabel 4.1 Hasil pengujian sensor posisi

	Motor X	Motor Y	Joystick x	Joystick y
Sudut Motor (Derajat)	Tegangan Sensor (Volt)	Tegangan Sensor (Volt)	Tegangan Sensor (Volt)	Tegangan Sensor (Volt)
50	0,97	-	2,30	-
40	1,31	0,68	2,09	0,82
30	1,68	1,04	1,90	1,11
20	2,02	1,41	1,74	1,41
10	2,36	1,72	1,57	1,77
0	2,76	2,12	1,34	2,07
-10	2,91	2,54	1,18	2,41
-20	3,36	2,91	0,97	2,72
-30	3,71	-	0,77	-
-40	4,13	-	0,57	-
-50	4,50	-	0,37	-

Pengujian sensor posisi motor DC dan *joystick* dilakukan pada jangkauan -50^0 sampai 50^0 . Hal ini berkaitan dengan batasan maksimum posisi motor DC dan *joystick* seperti dalam bagian perancangan.

4.2 PENGUJIAN DRIVER L293D

Pengujian ketiga dilakukan terhadap *driver* motor DC. Pengujian dilakukan dengan memberikan nilai *duty cycle* PWM dengan memasukkan ke nilai *duty cycle* ke dalam AVR secara *software*. Kemudian tegangan keluaran dari *driver* motor diambil. Nilai *duty cycle* dalam jangkauan 0% sampai 100% dengan frekuensi 1,064 kHz. Hasil pengujian diambil setiap kenaikan 10%. Adapun hasil pengujian ditunjukkan pada Tabel 4.2.

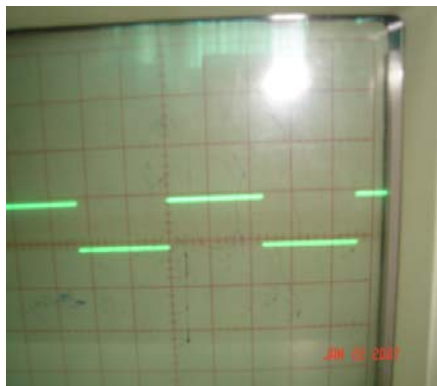
Tabel 4.2 Hasil pengujian *driver* L293D

Dutycycle (%)	Nilai OCR1X (desimal)	Tegangan keluaran <i>driver</i> (volt)
0	0	0,00
10	26	2,68
20	51	3,94
30	78	5,25
40	102	6,40
50	128	7,63
60	153	8,82
70	179	10,03
80	204	11,15
90	230	12,13
100	255	12,53

Dari Tabel 4.2 menunjukkan *driver* L293D telah bekerja. Namun terlihat ada perbedaan antara hasil perhitungan secara teori dengan hasil pengujian. Perbedaan hasil perhitungan teori dan hasil pengujian ini berkaitan dengan frekuensi yang digunakan pada sinyal PWM. Sebagai contoh perhitungan berikut,

$dutycycle = 20\%$, tegangan keluaran *driver* = 2,506 V

$dutycycle = 30\%$, tegangan keluaran *driver* = 3,759 V Frekuensi PWM yang digunakan ditunjukkan pada Gambar 4.1.



Gambar 4.1 Frekuensi PWM

Frekuensi yang digunakan 1,064 kHz. Berikut perhitungan frekuensi yang digunakan dalam pembangkitan sinyal PWM, $Time/Div = 0,2\text{ ms}$

Periode PWM = $4,7 \times 0,2\text{ ms} = 0,94\text{ ms}$

$$\text{Frekuensi} = \frac{1}{T} = \frac{1}{0,94 \times 10^{-3}} = 1,064\text{ kHz}$$

4.4 PENGUJIAN *SIMULATOR* SECARA KESELURUHAN

Pengujian *simulator* pengendalian posisi *Turret* secara keseluruhan dilakukan untuk menguji apakah *Turret* dapat mengikuti perubahan posisi dari *joystick*. Pengujian dilakukan dengan mengubah-ubah posisi *joystick*, baik untuk posisi sumbu horisontal maupun posisi sumbu vertikal.

Pada saat kondisi awal SOF (*switch* ON-OFF) berada dalam kondisi menutup. Secara otomatis posisi *Turret* menuju posisi parkir yaitu pada sumbu vertikal *Turret* berada pada posisi -20° , pada sumbu horisontal *Turret* berada pada posisi 50° . Selama SOF (*switch* ON-OFF) dalam keadaan menutup *Turret* tidak akan berubah posisi meskipun posisi *joystick* berubah-ubah. Kemudian saat SOF dalam kondisi terbuka *Turret* mulai bergerak. Posisi awal yang dituju *Turret* adalah posisi normal yaitu 0° pada kedua sumbu. Setelah mencapai posisi normal, *Turret* mengikuti posisi referensi dari *joystick*. *Turret* mampu bergerak mengikuti posisi referensi dari *joystick*. Perubahan nilai penguatan proporsional K_p dan penguatan integral K_i tidak terlihat pada pengujian karena pengamatan hanya dilakukan melalui pengamatan visual. Pengujian dilakukan dengan kendali PI (Proporsional-Integral).

Untuk mengetahui pengaruh kendali PI pada *simulator*, dilakukan juga pengujian dengan kendali ON-OFF sebagai sarana pembandingan. Pembandingan dilakukan karena unjuk kerja kendali PI hanya dilihat melalui visual, bukan melalui grafik respon sistem.

Pada pengujian dengan kendali ON-OFF, *Turret* mampu mengikuti posisi referensi sesuai posisi referensi dari *joystick*. Namun saat mendekati posisi referensi *Turret* terus berosilasi disekitar posisi referensi. Osilasi terjadi karena saat ada sinyal *error* (meskipun nilainya kecil) *simulator* langsung memberikan sinyal kendali maksimal sehingga perubahan posisi *Turret* besar. Adanya nilai sinyal kendali yang langsung maksimal merupakan karakteristik kendali ON-OFF yang hanya terdiri dari sinyal kendali nol dan maksimal.

V. Kesimpulan dan Saran

5.1 Kesimpulan

Dari perancangan *simulator* pengendalian posisi *Turret* dapat diperoleh kesimpulan:

1. Potensiometer *wirewound* memiliki keluaran linear.
2. Nilai frekuensi PWM berpengaruh pada keluaran *driver* L293D. Semakin

besar frekuensi PWM semakin besar selisih antara teori dan realitas keluaran *driver* L293D.

3. Kendali PI mampu membawa *Turret* mencapai dan stabil pada posisi referensi.
4. Kendali ON-OFF mampu membawa *Turret* menuju posisi referensi, namun terjadi osilasi disekitar posisi referensi.

5.2 Saran

1. Untuk lebih menyempurnakan *simulator* pengendalian posisi *Turret* ditambahkan pengendalian kecepatan aliran air yang keluar.
2. Ditambahkan komunikasi dengan komputer untuk mengetahui respon *simulator* terhadap masukan posisi referensi yang diberikan melalui *joystick*.
3. Untuk memperbaiki unjuk kerja dari *simulator* dilakukan perbaikan pada mekanik.
4. Dicoba metode kendali lain, seperti kendali Fuzzy, kedali Jaringan Syaraf Tiruan, dan kendali Adaptif.

Daftar Pustaka

- [1] Bakti, Dheka, *Skripsi: Pengendalian Suhu Secara Cascade Control Menggunakan Proporsional-Integral Berbasis Mikrokontroler ATmega8535*, Teknik Elektro, Universitas Diponegoro, Semarang
- [2] Hartono, Jogiyanto, *Konsep Dasar Pemrograman Bahasa C*, ANDI, Yogyakarta, 2000.
- [3] Johnson, Curtis, *Process Control Instrumentation Technology*, John Wiley & Sons, New York, 1977.
- [4] Kattuhiko Ogata, Edi Laksono, *Teknik Kontrol Automatik*, jilid I, Erlangga, Jakarta, 1990.
- [5] Malik, Moh Ibnu, *Pengantar Membuat Robot*, Gava Media , Yogyakarta, 2006.
- [6] Malik, Mohammad Ibnu. Dan Anistardi, *Bereksperimen dengan Mikrokontroler 8031*, PT. Elek Media Komputindo, Jakarta, 2004.
- [7] Putra, Agfianto Eko, *Teori dan Aplikasi : Belajar Mikrokontroler AT89C51/52/55*, Gava Media , Yogyakarta, 2002.
- [8] Wardhana, Lingga, *Belajar Sendiri Mikrokontroler AVR Seri ATmega8535* : ANDI , Yogyakarta, 2006.
- [9] Wasito, S., *Vademakum Elektronika*, Gramedia Pustaka Utama, Jakarta, 2001.
- [10] ---, *LCD Module User Manual* , EL-TECH Electronics , Surabaya.
- [11] <http://www.alldatasheet.com/L293D.pdf>



Isma Candra JK.
(L2F002588) lahir di Sragen, 8 September 1984. Menempuh pendidikan dasar di SDN Slogo 2 lulus tahun 1996, kemudian melanjutkan ke SLTPN 2 Gemolong lulus tahun 1999, dilanjutkan lagi di SMUN 1 Gemolong lulus tahun 2002, dan sampai saat ini masih menyelesaikan studi S1 di Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang Konsentrasi Kontrol.

Email : drajat02@yahoo.co.id

Menyetujui dan Mengesahkan,

Pembimbing I,

Trias Andromeda, ST. MT.

NIP. 132 283 185

Tanggal

Pembimbing II,

Darjat, ST. MT.

NIP. 132 231 135

Tanggal

