

# Sistem *Pick and Place* Dua Derajat Kebebasan menggunakan Metoda Regresi

ERWANI MERRY SARTIKA, RUDI SARJONO, HAZEL XARIS CHRISOPHRAS

Program Studi Teknik Elektro Universitas Kristen Maranatha  
Email: [erwanimerry@gmail.com](mailto:erwanimerry@gmail.com)

*Received* 24 Juni 2019 | *Revised* 15 Juli 2019 | *Accepted* 18 Agustus 2019

## ABSTRAK

*Sistem pick and place merupakan suatu sistem mekanik yang digunakan untuk memanipulasi pergerakan mengangkat, memindahkan, dan meletakkan untuk meringankan kerja manusia. Dalam mempelajari cara kerja robot industri sederhana dibuat miniatur robot pick and place (sederhana). Perancangan yang dibuat yaitu sistem pick and place dengan dua derajat kebebasan dengan ukuran yang memiliki perbandingan 1:0.35 dari referensi ukuran desain robot. Aplikasi SolidWorks digunakan untuk mendesain robot Diamond. Metode Regresi digunakan untuk memprediksi posisi motor servo dalam mencapai posisi yang diinginkan. Metode regresi berhasil digunakan untuk mencari hubungan antara target posisi setpoint dengan posisi motor servo 1 dan 2 (persamaan orde 2 dan 3) untuk mengontrol motor servo. Performansi yang terbaik dari sistem pick and place yang dibuat menggunakan trayektori miring, dengan kecepatan 100 (11.1 rpm), menghasilkan error  $\pm 0.0729$  dan presisi 1.63%. Dalam penelitian ini, kecepatan end-effector yang lebih rendah menghasilkan keakurasian dan kepresisian yang lebih baik.*

**Kata kunci:** *Pick and Place, Robot Diamond, Dua derajat kebebasan, Regresi*

## ABSTRACT

*The pick and place system are a mechanic system used in manipulating the movements of lifting, moving, and laying to ease human work. In learning how to work a simple industrial robot, a miniature pick and place robot is created. The design made is a pick and place system with two degrees of freedom with a size that has a ratio of 1: 0.35 from the reference size of the robot design. Regression method is used to predict the position of the servo motor in reaching the desired position. Regression method was successfully used to find the relationship between the target setpoint position and the position of servo motors 1 and 2 (order equations 2 and 3) to control the servo motor. The best performance from the pick and place system that is made using an aslope trajectory, with a speed of 100 (11.1 rpm), produces an error  $\pm 0.0729$  and precision 1.63%. In this research, lower end-effector speeds result in better accuracy and precision.*

**Keywords:** *Pick and Place, Diamond Robot, 2-DOF, Regression*

## 1. PENDAHULUAN

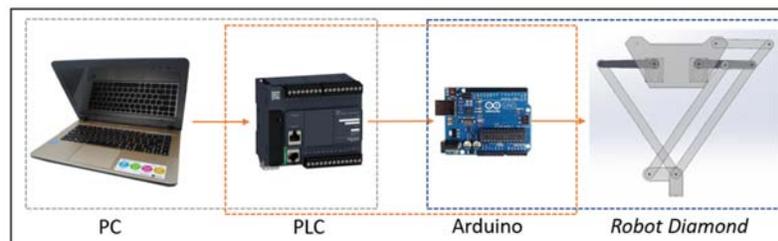
Penggunaan otomatisasi robot memungkinkan fleksibilitas dalam proses manufaktur sehingga robot mampu melakukan tugas secara efisien. Robot cenderung digunakan sebagai pengganti pekerja manual menjadi sistem yang otomatis, karena tugas yang berulang dan mengurangi kesalahan pekerja. Robot *pick and place* merupakan salah satu robot industri yang membantu menyelesaikan tugas tersebut (Li, Xiao, Bi, & Du, 2013). Robot *pick and place* banyak digunakan dalam industri, terutama di jalur produksi tempat membantu memilih dan menempatkan komponen di lokasi yang diinginkan (Kit & Venkatratnam, 2016). Perangkat lunak Solidworks dipilih karena memungkinkan analisis dan simulasi desain lengan robot industri *pick and place* (Sam, Arrifin, & Buniyamin, 2012).

Masalah perencanaan lintasan merupakan hal mendasar dalam robotika, sehingga persyaratan yang ditetapkan untuk lintasan dipenuhi. Tujuan perencanaan lintasan adalah untuk menghasilkan input untuk sistem kontrol manipulator untuk menjalankan satu aplikasi. Salah satu masalah utama dalam robotika adalah pengembangan sistem perencanaan lintasan otomatis (Li et al., 2013). Sebagian besar robot beroperasi dalam siklus yang telah ditentukan sebelumnya, dengan membandingkan cara manusia melakukan pekerjaan yang lebih banyak fleksibilitas, (Andhare, 2016) memberikan gagasan menemukan koordinat objek  $x, y$ . Hal ini juga dapat membantu pengembangan algoritma pergerakan robot *pick and place* untuk mengoptimalkan karakteristik kinematik saat menghadapi dua titik di antara ruang kerja robot (Li et al., 2013).

Pada penelitian ini dibuat robot *pick and place* dengan konsep paralel menggunakan sifat jajaran genjang dan digerakkan oleh persendian *revolute* (rotasi) proksimal berupa robot *Diamond* dengan 2-DOF. Desain robot *Diamond* dirancang dengan menggunakan aplikasi *SolidWorks* agar dapat menggerakkan posisi *movable platform* tegak lurus dengan bidang datar. Metoda regresi digunakan agar robot dapat mencapai koordinat yang diinginkan dengan mendapatkan hubungan variabel (posisi motor servo) dengan variabel yang lain (posisi robot) sehingga robot dapat bergerak menuju posisi yang diinginkan.

## 2. PERANCANGAN SISTEM PICK AND PLACE

Perangkat yang digunakan dalam merealisasi sistem *pick and place* secara garis besar terdiri dari PC, PLC, Arduino, dan dua motor servo yang terdapat pada robot *Diamond*. Diagram blok sistem *pick and place* ditunjukkan pada Gambar 1. PC digunakan untuk memprogram PLC dan Arduino, selain itu PC juga berfungsi sebagai media untuk memasukkan posisi dan kecepatan yang diinginkan. Proses *input* data melalui simulasi HMI yang terdapat pada program *Vijeo Designer*.



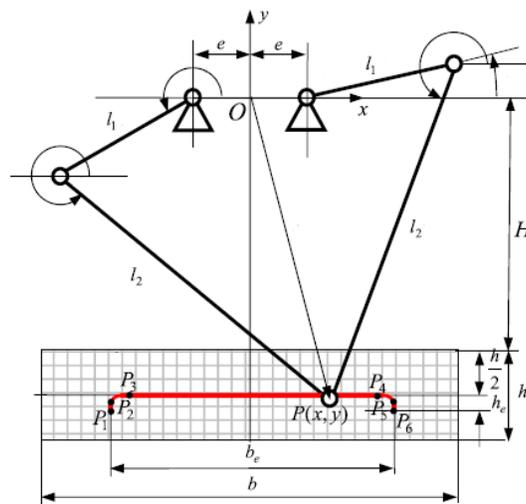
Gambar 1. Sistem *Pick and Place* yang dirancang

PLC akan mengontrol sistem *pick and place* dengan menggerakkan kedua motor servo agar mencapai posisi yang diinginkan. Motor servo yang digunakan adalah Dynamixel AX-12A,

karena keterbatasan aktuator yang tersedia (motor servo untuk robotika bukan motor servo dengan *standard* industri). Arduino digunakan untuk komunikasi dengan robot *Diamond* khususnya dengan dua motor servo *AX-12A*. Motor servo *AX-12A* memiliki sinyal kontrol berupa *Digital Packet*, sehingga tidak dapat langsung dikontrol oleh PLC yang hanya memiliki pin *output* PWM untuk mengontrol motor servo. Arduino digunakan sebagai *driver* agar dapat menggerakkan *Dynamixel AX-12A* sehingga sistem *pick and place* dapat terealisasi.

### 2.1. Perancangan Robot *Diamond*

Konsep perancangan robot *Diamond* mengacu pada konsep *without over constraints* (Chetwynd & Gosselin, 2016). Gambar skematik robot *Diamond* yang dilengkapi dengan Persamaan (1), (2), dan (3) yang menghubungkan antara ukuran robot dan daerah jangkauan sistem ditunjukkan pada Gambar 2. Pada perancangan dipilih panjang *distal link* (*inner* maupun *outer*) adalah 30 cm maka perbandingan yang diperoleh yaitu 1:0.35 dari ukuran desain pada jurnal (Huang, Liu, Mei, & Chetwynd, 2013). Ukuran miniatur robot *Diamond* yang dirancang ditunjukkan pada Tabel 1.



Gambar 2. Skematik Robot *Diamond* (Huang et al., 2013)

$$e \geq e_{min} \quad (1)$$

$$\frac{b}{2} = e + l1 \quad (2)$$

$$\sqrt{(H + h)^2 + \left(\frac{b}{2} + e\right)^2} - l2 - l1 < 0, l2 - l1 - H < 0 \quad (3)$$

Keterangan:

- $h$  = tinggi area yang dapat dijangkau sistem
- $b$  = area yang dapat dijangkau sistem
- $b_e$  = panjang trayektori sistem
- $e_{min}$  = jarak antara motor servo dan titik tengah robot
- $l1$  = panjang *proximal link*
- $l2$  = panjang *distal link*
- $H$  = tinggi miniatur sistem saat posisi *standby*

**Tabel 1. Perbandingan antara Ukuran Desain pada Jurnal dan Miniatur Robot *Diamond* yang dirancang**

Parameter	$e_{min}$	$l_1$	$l_2$	$H$
Ukuran Desain pada Jurnal	0.150 m	0.350 m	0.858 m	0.650 m
Miniatur Robot	5 cm	12 cm	30 cm	22.5 cm

Selanjutnya parameter lain yang harus ditentukan adalah parameter pada Persamaan (1), (2), dan (3). Parameter tersebut yaitu ketinggian dari robot *Diamond* yang dapat dijangkau ( $h$ ) dan area yang dapat dijangkau oleh sistem ( $b$ ). Nilai  $P(x,y)$  merupakan posisi target yang akan dicapai oleh *end-effector*. Nilai  $x$  merupakan titik dari panjang area yang ingin dicapai, sedangkan nilai  $y$  merupakan ketinggian yang ingin dicapai.

Pada penelitian ini, nilai  $y$  tetap yaitu dengan ketinggian 1 cm. Untuk memenuhi Persamaan (1), nilai variabel  $e$  yang dipilih pada Persamaan (2) yaitu 5 cm sehingga nilai variabel  $b$  pada Persamaan (2) diperoleh sebesar 34 cm. Nilai variabel  $b_e$  dibatasi dari *range* 0 sampai 30 cm, maka nilai variabel  $h$  pada Persamaan (3) dapat diperoleh  $h < 13.27$  cm. Panjang kerangka penopang dibuat 35 cm, karena harus menyesuaikan dengan  $b$  yang didapatkan yaitu dibuat melebihi 34 cm. Tinggi kerangka penopang dipilih 40 cm, karena ketinggiannya harus melebihi ketinggian total  $(H + h) = 35.77$  cm.

Pada awalnya  $l_2$  didesain menggunakan *double rod end bearing*, tetapi dikarenakan sulit mendapatkan barang tersebut dengan spesifikasi yang diinginkan, maka dipilih bahan akrilik sebagai gantinya. *End-effector* yang akan digunakan yaitu laser dioda. Laser digunakan untuk melihat *movable platform* saat menuju ke posisi yang diinginkan.

Desain robot *Diamond* dirancang dengan menggunakan aplikasi *SolidWorks* (Rhakasywi, 2016). Pada aplikasi *SolidWorks* didesain bagian-bagian yang diperlukan dalam robot *Diamond*. Desain yang diharapkan yaitu desain agar robot *Diamond* yang dapat menggerakkan posisi *movable platform* tegak lurus dengan bidang datar. Bahan utama dalam pembuatan robot *Diamond* menggunakan akrilik. Gambar 3 menunjukkan realisasi robot *Diamond* yang telah dirancang.



**Gambar 3. Realisasi Robot *Diamond***

## 2.2. Sistem Kontrol *Pick and Place* menggunakan Metoda Regresi

Metoda regresi adalah salah satu analisis yang paling populer dan luas pemakaiannya, diantaranya dapat digunakan untuk melakukan prediksi. Metoda regresi dalam statistika adalah salah satu metode untuk menentukan hubungan antara satu variabel dengan variabel yang lain. Variabel tersebut terdiri dari variabel bebas dan variabel terikat (**Luknanto, 2001**).

Regresi polinomial digunakan untuk menentukan fungsi polinomial yang paling sesuai dengan kumpulan data (X,Y) yang diketahui. Persamaan (4) merupakan persamaan polinomial orde-n. Rumus umum metoda regresi untuk persamaan orde dua ditunjukkan pada Persamaan (5) (**Chapra & Canale, 2015**).

$$y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \quad (4)$$

$$\begin{aligned} Na_0 &+ (\sum_{i=1}^n xi)a_1 + (\sum_{i=0}^n xi^2)a_2 = \sum_{i=1}^n yi \\ (\sum_{i=1}^n xi)a_0 &+ (\sum_{i=0}^n xi^2)a_1 + (\sum_{i=0}^n xi^3)a_2 = \sum_{i=1}^n yixi \\ (\sum_{i=0}^n xi^2)a_0 &+ (\sum_{i=0}^n xi^3)a_1 + (\sum_{i=0}^n xi^4)a_2 = (\sum_{i=0}^n xi^2yi) \end{aligned} \quad (5)$$

Keterangan:

y = variabel terikat

N = jumlah data

x = variabel bebas

Korelasi merupakan salah satu analisis dalam statistik yang dipakai untuk mencari hubungan antara dua variabel yang bersifat kuantitatif. Korelasi bertujuan untuk memperlihatkan atau menentukan erat atau tidaknya hubungan antara dua variabel yaitu x dan y. Bentuk umum korelasi dapat dilihat dari Persamaan (6). Koefisien korelasi bernilai mendekati 1 berarti hubungan antara dua variabel (x dan y) sangat kuat. Koefisien korelasi bernilai 0 berarti tidak terdapat hubungan antara dua variabel (x dan y) yang diuji atau dapat dinyatakan tidak ada hubungan sama sekali (**Chapra & Canale, 2015**).

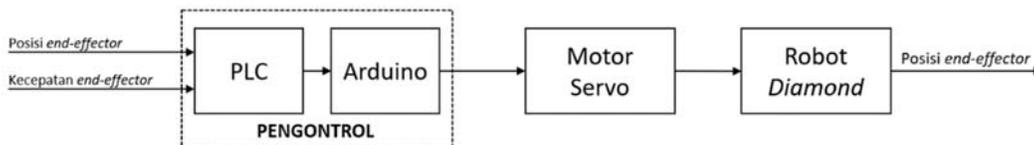
$$r_{xy} = \frac{N\sum xy - (\sum x)(\sum y)}{\sqrt{[N\sum(x)^2 - (\sum x)^2][N\sum(y)^2 - (\sum y)^2]}} \quad (6)$$

Keterangan:

$r_{xy}$  = Koefisien korelasi yang dicari

N = Jumlah responden

Pada sistem kontrol *pick and place* yang dirancang adalah sistem kontrol *open loop* (**Ogata, 2010**) seperti yang ditunjukkan pada Gambar 4. Metoda regresi akan digunakan untuk memprediksi hubungan antara posisi kedua motor servo dengan posisi robot (posisi *end-effector*). Posisi *end-effector* yang ingin dituju akan mempengaruhi posisi kedua motor servo yang terdapat dalam sistem *pick and place*, sehingga perlu dicari hubungan antara posisi *end-effector* dengan posisi masing-masing motor servo.



**Gambar 4. Diagram Blok Sistem Kontrol *Pick and Place***

Akibat keterbatasan disain robot *pick and place* (menggunakan akrilik), yang seharusnya menggunakan *double rod end bearing* mengakibatkan trayektori yang dilalui oleh *end-effector* tidak membentuk garis lurus. Trayektori yang dapat dilalui oleh *end-effektor* membentuk garis yang miring seperti ditunjukkan oleh penggaris pada Gambar 5. Area pergerakan yang ingin dicapai memiliki *range* 0 sampai 30 cm, sesuai dengan penggaris yang telah terpasang pada area trayektori. Pada Gambar 5 menunjukkan perbandingan antara trayektori lurus dan trayektori miring/diagonal yang dihasilkan oleh *end-effector* dari sistem *pick and place* yang dibuat. Maka trayektori yang digunakan adalah trayektori yang miring/diagonal karena tepat dengan yang dilalui *end-effector*.

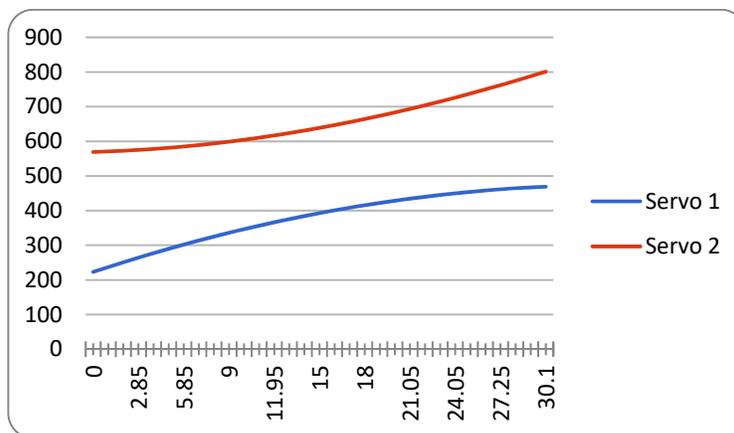


**Gambar 5. Area Trayektori Lurus dan Area Trayektori Miring/Diagonal**

**Tabel 2. Data Sampling Posisi Target terhadap Posisi Servo 1 dan Servo 2**

No	Posisi Target (cm)	Servo 1	Servo 2	No	Posisi Target (cm)	Servo 1	Servo 2
1	0	223	569	17	15.9	401	647
2	0.75	237	571	18	16.85	408	655
3	1.75	251	573	19	18	416	664
4	2.85	264	575	20	19	422	673
5	3.9	277	578	21	20.15	429	683
6	4.75	290	581	22	21.05	435	693
7	5.85	302	585	23	22.05	440	704
8	6.85	314	589	24	23.05	445	715
9	8	325	594	25	24.05	450	726
10	9	336	599	26	25.15	454	738
11	9.85	346	604	27	26.15	458	750
12	11	357	610	28	27.25	462	762
13	11.95	366	617	29	28.25	465	775
14	13.1	376	624	30	29.25	467	788
15	14	384	631	31	30.1	469	801
16	15	393	639				

Pengambilan data dilakukan untuk mendapatkan hubungan antara posisi *end-effector* dengan posisi motor servo 1 dan motor servo 2. Data diambil di sepanjang trayektori miring/diagonal dan diperoleh data seperti yang ditunjukkan pada Tabel 2. Dari pasangan data pada Tabel 2 digunakan metoda regresi untuk mendapatkan hubungan input1 (posisi motor servo 1 sebagai variabel x) dengan *output1* (posisi *end-effector* sebagai variabel Y) dan hubungan input2 (posisi motor servo 2 sebagai variabel X) dengan *output1* (posisi *end-effector* sebagai variabel Z). Terdapat 31 pasangan data sampling yang diambil untuk diolah agar diperoleh persamaan regresinya. Dari Tabel 2 tersebut, selanjutnya di plot gambar grafik dari hubungan *input* dan *output* tersebut seperti ditunjukkan pada Gambar 6.



**Gambar 6. Grafik hubungan antara posisi target dan masing-masing motor servo pada trayektori diagonal**

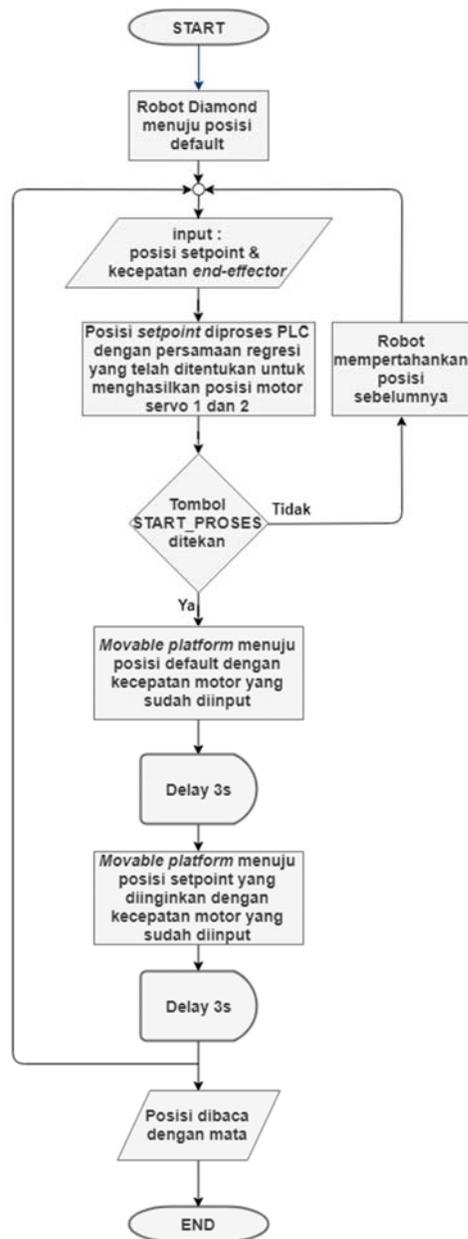
Selanjutnya dilakukan perhitungan menggunakan metoda regresi (Persamaan (4) dan Persamaan (5) berdasarkan data pada Tabel 2. Hasilnya diperoleh bahwa digunakan persamaan regresi orde 2 untuk mendapatkan posisi motor servo 1 dan persamaan regresi orde 3 untuk mendapatkan posisi motor servo 2 seperti yang ditunjukkan pada Persamaan (7) dan Persamaan (8). Persamaan tersebut dipilih setelah dihitung nilai koefisien korelasi yang paling mendekati 1 yaitu dipilih persamaan orde 2 pada posisi motor servo 1 dengan nilai koefisien korelasi sebesar 0.99995 sedangkan untuk posisi motor servo 2 dipilih persamaan orde 3 dengan nilai koefisien korelasi sebesar 0.99994.

Persamaan (7) merupakan hasil persamaan regresi (dipilih orde dua) dari posisi target dengan posisi motor servo 1, variabel X merupakan nilai posisi target (*end-effector*) sedangkan variabel Y merupakan nilai posisi motor servo 1. Persamaan (8) merupakan hasil persamaan regresi (dipilih orde tiga) dengan variabel X merupakan posisi target (*end-effector*) dan variabel Z merupakan posisi motor servo 2.

$$Y = 225.255 + 14.2531 X - 0.2051 X^2 \quad (7)$$

$$Z = 569.657 + 1.2129 X + 0.2391 X^2 - 0.0008 X^3 \quad (8)$$

*Flowchart* proses sistem *pick and place* ditunjukkan pada Gambar 7. Ketika sistem pertama kali dimulai, posisi *end-effector* pada robot *Diamond* akan bergerak pada posisi *default* dengan kecepatan motor *default*, yaitu 100. Selanjutnya, memasukkan posisi *setpoint* dan kecepatan *end-effector* yang diinginkan pada simulasi HMI yang terdapat pada program *Vijeo Designer*. PLC kemudian akan memproses data posisi tersebut menggunakan persamaan regresi yang telah ditentukan sehingga menghasilkan nilai posisi motor servo 1 dan 2.



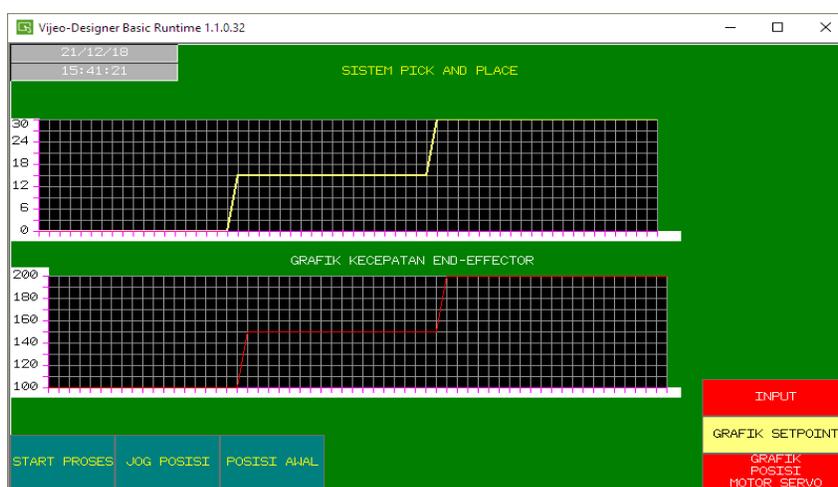
**Gambar 7. Flowchart Proses Pengujian Sistem *Pick And Place***

Setelah memasukkan data posisi *setpoint* dan kecepatan *end-effector* pada simulasi HMI, selanjutnya tombol *Start Proses* ditekan. Posisi *end-effector* akan bergerak ke posisi *default* terlebih dahulu, lalu akan bergerak ke posisi *setpoint* yang di-*input*. Kecepatan pergerakan kedua motor servo akan sesuai dengan kecepatan *end-effector* yang di-*input*. Delay 3 detik diberikan sebagai waktu maksimum pencapaian ke *end-effector*, kemudian sistem akan kembali memeriksa adanya data posisi dan kecepatan selanjutnya yang diinginkan. Bersamaan setelah delay 3 detik *end-effector* yang merupakan laser dioda yang terpasang pada *movable platform* digunakan untuk menunjukkan posisi *end-effector* yang tercapai, sehingga dapat dibaca oleh mata. Pembacaan posisi yang terlihat tersebut akan dicatat untuk dianalisis keakurasian dan kepresisian yang dihasilkan oleh sistem *pick and place*. Posisi *setpoint* yang

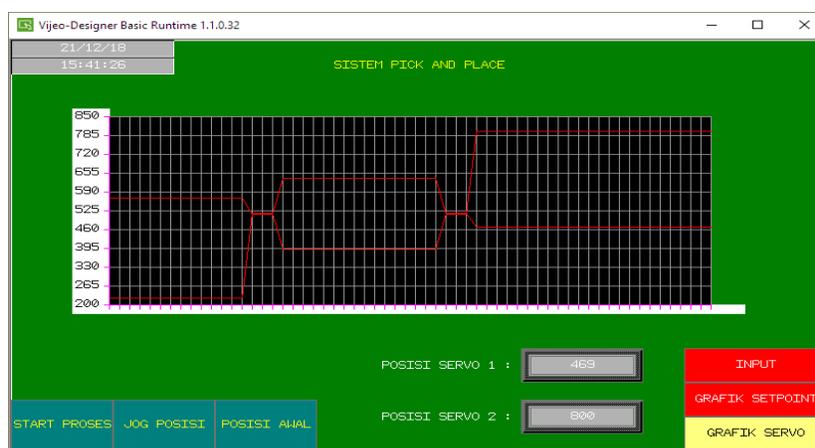
diuji pada area trayektori memiliki range 0 sampai 30 cm juga merupakan area pergerakan robot dengan dimensi arah horisontal.

### 3. HASIL DAN PEMBAHASAN

Untuk data pengamatan dibuat HMI menggunakan software *Vijeo Designer* (**Sartika, Sarjono, & Restianto, 2019**) agar dapat memasukan target (posisi *end-effector* dan kecepatan) yang diinginkan. Gambar 8 menunjukkan grafik *setpoint* ketika di-*input* posisi 0, 15, dan 30 secara berurutan dan kecepatan *end-effector* saat diberi *input* 100, 150 dan 200 pada saat ditekan tombol *Start Proses*. Sedangkan Gambar 9 menunjukkan grafik posisi motor servo yang terjadi saat di-*input* posisi.



Gambar 8. Panel Grafik *Setpoint* ketika diberi *Input* yang Berbeda



Gambar 9. Panel Grafik Saat Menunjukkan Posisi Motor Servo

Pada Tabel 3. menunjukkan hasil yang terbaca dari pengujian sistem *pick and place* pada trayektori miring/diagonal, dengan *setpoint* berupa koordinat (bukan jarak pergerakan) mulai dari 0-30 dalam satuan senti meter (cm). Pada Tabel 4 menunjukkan selisih antara posisi *setpoint* dan posisi *end-effector* yang terbaca pada trayektori miring. Jumlah posisi *setpoint*

yang diuji sebanyak 31 posisi. Kecepatan *end-effector* yang akan diuji yaitu 100, 150, dan 200. Pengontrolan sistem *pick and place* berhasil dilakukan. Data hasil pengujian dilakukan 5 kali pengulangan pada masing-masing kecepatan *end-effector* yang berbeda. Pengulangan ini dilakukan agar dapat melihat kepresisian yang dihasilkan oleh sistem.

**Tabel 3. Data Posisi *End-Effector* yang Dihasilkan**

Set Point (cm)	Data Posisi Setpoint yang dihasilkan														
	Kecepatan End-Effector 100					Kecepatan End-Effector 150					Kecepatan End-Effector 200				
	0	0	0,05	0	0	0	-0,1	-0,1	-0,1	-0,1	-0,1	-0,1	-0,15	-0,15	-0,15
1	0,85	0,85	0,85	0,9	0,85	0,9	0,9	0,9	0,9	0,9	0,85	0,9	0,95	0,95	0,9
2	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,9	1,95	1,95	1,9
3	2,95	2,95	2,95	2,95	2,95	2,85	2,85	2,85	2,85	2,85	2,8	2,85	2,9	2,9	2,85
4	3,95	3,95	3,95	3,9	3,95	3,9	3,9	3,9	3,9	3,9	3,75	3,75	3,75	3,75	3,75
5	4,85	4,85	4,85	4,8	4,85	4,75	4,75	4,8	4,75	4,8	4,9	4,9	4,95	4,95	4,9
6	5,85	5,85	5,8	5,8	5,85	5,85	5,85	5,85	5,85	5,85	5,9	5,9	5,95	5,95	5,9
7	7	7	7	7	7	6,9	6,9	6,9	6,9	6,9	6,9	6,85	6,9	6,9	6,85
8	7,95	8	7,95	7,95	7,95	7,85	7,85	7,85	7,85	7,85	7,85	7,85	7,9	7,9	7,85
9	8,95	8,95	8,9	8,95	8,95	8,9	8,95	8,95	8,95	8,95	8,8	8,8	8,85	8,85	8,8
10	9,9	9,9	9,95	9,9	9,9	9,9	9,95	9,95	9,95	9,95	9,85	9,9	9,9	9,9	9,9
11	10,95	10,95	11	10,9	10,95	10,9	10,95	10,95	10,95	10,95	10,95	10,95	11	11	10,95
12	11,95	11,95	12	11,9	11,95	11,9	11,9	11,9	11,9	11,9	11,8	11,8	11,85	11,85	11,8
13	13	12,95	13	12,95	13	12,9	12,9	12,9	12,9	12,9	12,8	12,8	12,85	12,85	12,8
14	14	13,95	14	13,95	14	13,9	13,9	13,9	13,9	13,9	13,85	13,85	13,9	13,9	13,85
15	14,9	14,9	14,9	14,9	14,9	14,85	14,85	14,85	14,85	14,85	14,85	14,85	14,9	14,9	14,85
16	15,8	15,75	15,85	15,8	15,8	15,8	15,8	15,85	15,8	15,85	15,9	15,9	15,9	15,9	15,9
17	16,75	16,75	16,8	16,8	16,75	16,65	16,8	16,8	16,8	16,8	16,8	16,8	16,8	16,8	16,8
18	17,85	17,85	17,85	17,9	17,85	17,65	17,85	17,8	17,8	17,85	17,85	17,7	17,8	17,7	17,7
19	19	19	19	19	19	18,9	18,95	18,95	18,95	18,95	18,95	18,9	18,9	18,9	18,9
20	20	20	20	20	20	19,9	20	20	20	20	19,9	19,9	19,9	19,9	19,9
21	21	21	21,05	21,95	21	20,9	21	21	21	21	20,9	20,9	20,9	20,9	20,9
22	22,05	22	22,1	22	22	21,95	22	22	22	22	21,9	21,85	21,85	21,85	21,85
23	23	23	23	22,9	23	22,9	22,9	22,9	22,9	22,9	22,85	22,9	22,85	22,85	22,85
24	23,95	23,95	24	23,9	23,95	23,7	23,85	23,85	23,85	23,85	23,85	23,9	23,8	23,85	23,85
25	25	24,95	25	25	24,95	24,8	24,85	24,85	24,85	24,85	24,95	25	24,95	25	24,95
26	26	25,95	26	25,95	25,95	25,8	25,9	25,9	25,9	25,9	26	26,05	26	26,05	26
27	27,15	27,1	27,15	27,1	27,1	26,9	27	27,05	27,05	27	27,05	27,1	27,1	27,1	27,05
28	28,1	28,1	28,15	28,15	28,1	27,9	28	28	28	28	28,1	28,1	28,05	28,1	28,1
29	28,95	28,95	28,95	28,95	28,95	28,6	28,85	28,85	28,85	28,85	28,85	28,9	28,95	28,9	28,85
30	29,9	29,9	29,9	29,85	29,9	29,6	29,8	29,8	29,8	29,8	29,9	29,95	29,95	29,95	29,9

**Tabel 4. Akurasi dan Presisi yang Dihasilkan**

	Kecepatan <i>End-Effector</i> 100 (11.1 rpm)	Kecepatan <i>End-Effector</i> 150 (16.65 rpm)	Kecepatan <i>End-Effector</i> 200 (22.2 rpm)
<b>Akurasi</b>	± 0.0729 (7.29%)	± 0.1161 (11,61%)	± 0.12 (12%)
<b>Presisi</b>	± 0.0163 (1,63%)	± 0.0196 (1,96%)	± 0.0204 (2,04%)

Pada Tabel 4 dapat menunjukkan hasil keakurasian dan kepresisian yang dihasilkan oleh persamaan regresi yang diaplikasikan pada area trayektori miring. Toleransi kesalahan yang masuk dalam spesifikasi pada batasan masalah yaitu robot dengan kecepatan *end-effector* 100. Keakurasian yang dihasilkan robot dengan kecepatan *end-effector* 100 merupakan yang paling baik dibandingkan dengan robot dengan kecepatan *end-effector* 150 & 200. Kepresisian yang dihasilkan sistem sudah sangat baik. Pada sistem ini, keakurasian yang paling baik dimiliki oleh robot dengan kecepatan *end-effector* 100 dibandingkan dengan robot yang memiliki kecepatan *end-effector* 150 & 200. Dengan bertambahnya kecepatan *end-effector* pada sistem, maka semakin menurun keakurasian dan kepresisian yang dihasilkan.

#### 4. KESIMPULAN

Miniatur sistem *pick and place* berhasil dirancang dan direalisasi sehingga sistem dapat bergerak ke posisi yang diinginkan menggunakan metode regresi yang dikontrol oleh PLC dan Arduino. Metoda regresi berhasil digunakan untuk mencari hubungan antara target posisi *setpoint* dengan posisi motor servo 1 dan 2 (persamaan orde 2 dan 3) untuk mengontrol motor servo. Keterbatasan bahan yang digunakan menyebabkan pergerakan translasi yang dihasilkan robot *Diamond* menjadi miring/diagonal. Akurasi yang dihasilkan robot dengan kecepatan *end-effector*  $100 \pm 0.0729$  adalah yang terbaik. Penambahan kecepatan *end-effector* menyebabkan nilai akurasi yang dihasilkan semakin menurun. Demikian pula kepresisian 1.63% yang dihasilkan robot dengan kecepatan *end-effector* 100 (11.1 rpm) adalah yang terbaik. Penambahan kecepatan *end-effector*, maka nilai presisi yang dihasilkan semakin menurun.

#### DAFTAR RUJUKAN

- Andhare, P. (2016). Pick and Place Industrial Robot Controller with Computer Vision. In *Conference: 2016 International Conference on Computing Communication Control and automation (ICCUBEA)* (pp.1-4).
- Chapra, S. C., & Canale, R. P. (2015). *Numerical Methods for Engineers* (seventh Ed). McGraw-Hill Science/Engineering/Math.
- Chetwynd, D. G., & Gosselin, C. M. (2016). Conceptual Design and Dimensional Synthesis of a Novel 2-DOF Translational Parallel Robot for Pick-and-Place. *Transactions of the ASME, Elsevier, 126*, 449–455. <https://doi.org/10.1115/1.1711822>
- Huang, T., Liu, S., Mei, J., & Chetwynd, D. G. (2013). Optimal design of a 2-DOF pick-and-place parallel robot using dynamic performance indices and angular constraints. In *Mechanism and Machine Theory, Elsevier, 70*, 246–253. <https://doi.org/10.1016/j.mechmachtheory.2013.07.014>
- Kit, W. S., & Venkatratnam, C. (2016). Pick and Place Mobile Robot for the Disabled through Voice Commands. *2016 2nd IEEE International Symposium on Robotics and Manufacturing Automation (ROMA)*, (pp. 3–6).

<https://doi.org/10.1109/roma.2016.7847799>

Li, W., Xiao, Y., Bi, S., & Du, G. (2013). Automatic Elliptical Trajectory Planning Algorithm For Pick and Place Operation. In *Proceedings of the 2013 International Conference on Advanced Mechatronic Systems, Luoyang, China*. (pp. 36–39).

Luknanto, D. (2001). *Metoda Numerik* (pp. 16–21).

Ogata, K. (2010). *Modern Control Engineering*. New Jersey. Prentice Hall.

Rhakasywi, D. (2016). *Rancang Bangun Elemen Mesin 3D Dengan Solidworks*. Andi.

Sam, R., Arrifin, K., & Buniyamin, N. (2012). Simulation of Pick and Place Robotics System Using Solidworks Softmotion. In *2012 International Conference on System Engineering and Technology (ICSET)*.

Sartika, E. M., Sarjono, R., & Restianto, R. S. (2019). Simulasi Sistem Otomasi Load Shedding menggunakan Prediksi Beban, *ELKOMIKA*,  $\lambda(1)$ , 180-194.  
<http://doi.org/http://dx.doi.org/10.26760/elkomika.v7i1.180>