

# VISUALISASI DAN PENGENDALIAN GERAK ROBOT LENGAN 4 DOF MENGUNAKAN VISUAL BASIC

<sup>[1]</sup>Uray Ristian, <sup>[2]</sup>Ferry Hadary, <sup>[3]</sup>Yulrio Brianorman

<sup>[1][3]</sup>Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura

<sup>[2]</sup>Jurusan Teknik Elektro, Fakultas Teknik Universitas Tanjungpura

Jl. Ahmad Yani, Pontianak

Telp./Fax.: (0561) 577963

e-mail:

<sup>[1]</sup>erish.ristian@gmail.com, <sup>[2]</sup>ferryhadary@yahoo.com, <sup>[3]</sup>rionorman@gmail.com

## Abstrak

*Pada umumnya robot lengan biasanya dikendalikan dengan menggunakan mikrokontroler yang terhubung ke tiap motor penggerakannya. Namun bentuk pergerakannya tidak dapat diketahui. Oleh karena itu, pada penelitian ini mencoba untuk membuat aplikasi pengendalian berbasis visualisasi yang dapat melakukan simulasi pergerakan robot lengan 4 DoF sesuai dengan gerak robot aslinya. Penelitian ini dilakukan dengan membuat perangkat lunak antarmuka menggunakan aplikasi Visual Basic. Aplikasi ini memiliki keunggulan yaitu dapat menganimasikan pergerakan robot lengan dengan menggunakan perhitungan kinematika. Selanjutnya data pergerakan visualisasi yang terbentuk dikirimkan ke mikrokontroler menggunakan komunikasi serial sehingga robot dapat bergerak sesuai dengan tampilan aplikasi antarmuka. Data pergerakan yang akan dikirimkan dikapsulasi sehingga hanya membutuhkan satu kali pengiriman data saja. Hasil dari penelitian ini adalah berhasilnya mengendalikan robot lengan menggunakan aplikasi pengendali berbasis visual sehingga pergerakan robot lengan dapat diketahui.*

Kata Kunci: *Visual Basic*, Robot Lengan 4 DoF, Kinematika, Mikrokontroler, Komunikasi Serial.

## 1. PENDAHULUAN

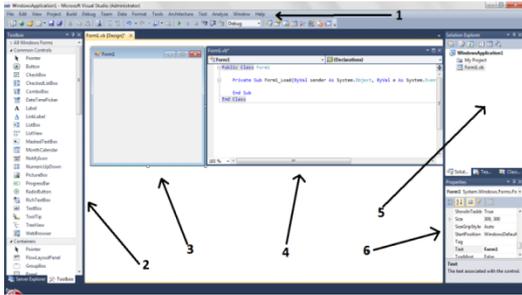
Pengendalian robot lengan merupakan pengendalian gerak robot lengan dengan memanipulasi putaran motor-motor penggerakannya sehingga robot dapat mengambil objek yang telah ditentukan posisinya. Namun, permasalahan yang dihadapi dalam pengendalian ini adalah bagaimana robot dapat mengambil objek dengan posisi yang tepat. Sedangkan jika menggunakan pengendali konvensional, arah pergerakan robot lengan masih belum bisa diprediksi. Ini disebabkan karena pengendali tersebut hanya bertugas menggerakkan motor-motor penggerakannya saja sehingga diperlukan pengujian secara manual terlebih dahulu untuk memprediksi pergerakan robot lengan. Oleh karena itu dibutuhkan suatu aplikasi pengendali yang dapat mensimulasikan pergerakan robot

lengan dan mengirimkan data kepada robot lengan agar bergerak sesuai dengan yang divisualkan. Dengan membuat aplikasi pengendali berbasis visualisasi, diharapkan dapat mempermudah dalam pembacaan pergerakan robot lengan sehingga pengendalian pergerakannya pun menjadi mudah.

## 2. TINJAUAN PUSTAKA

### 2.1 Visual Basic

*Visual Basic* adalah salah satu *development tools* untuk membangun aplikasi dalam lingkungan *Windows*. Dalam pengembangan aplikasinya, *Visual Basic* menggunakan pendekatan visual untuk merancang *user interface* dalam bentuk *form*, sedangkan untuk pemrogramannya menggunakan bahasa *Basic* yang cenderung mudah dipelajari. [1]



Gambar 1. Tampilan *Project Visual Basic 2010*

## 2.2 Akses Komunikasi Serial Menggunakan Visual Basic

Untuk pengaksesan port serial dapat dilakukan secara langsung menggunakan kontrol Port Serial yang telah disediakan Visual Basic. *MsComm* ini menyediakan fasilitas komunikasi antara program aplikasi yang dibuat dengan port serial untuk mengirim atau menerima data melalui port serial. Setiap *MsComm* hanya menangani satu port serial sehingga jika ingin menggunakan lebih dari satu port serial, harus digunakan *MsComm* yang lain. [2]

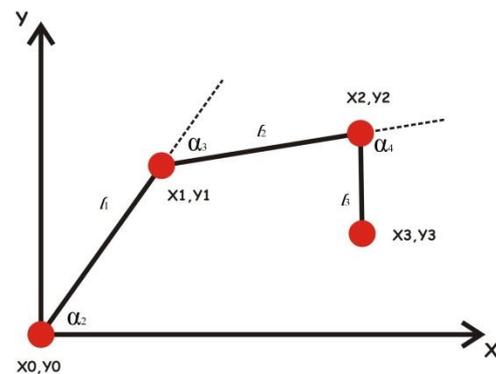
Jumlah properti pada *MsComm* sangat banyak dan hanya beberapa properti yang perlu diketahui. Adapun properti yang sering digunakan adalah *CommPort* yang digunakan untuk menentukan nomor port serial yang akan dipakai, *PortOpen* yang digunakan untuk membuka ataupun menutup port serial yang dihubungkan dengan *MsComm* ini, dan *Setting* yang digunakan untuk menset nilai *baud rate*, pariti, jumlah bit data, dan jumlah bit stop.

## 2.3 Kinematika Robot Lengan

Pada penelitian ini perhitungan kinematika robot lengan digunakan dalam pembuatan penganimasian visualisasi pergerakan robot lengan. Kinematika merupakan studi pergerakan robot tanpa memperhatikan gaya maupun faktor lain yang mempengaruhi pergerakan robot tersebut. [3]

Koordinat posisi dari tiap sendi robot lengan akan dianimasikan dengan memanipulasi besar sudut yang terbentuk.

Pada penggunaan rumus kinematika,  $\ell_1$ ,  $\ell_2$  dan  $\ell_3$  merupakan panjang tiap ruas lengan. Variabel  $x_0y_0$  adalah posisi sendi putar 1,  $x_1y_1$  adalah posisi yang dibentuk oleh sendi 2 ( $\alpha_2$ ),  $x_2y_2$  adalah posisi yang dibentuk oleh sendi 3 ( $\alpha_3$ ) serta  $x_3y_3$  adalah posisi yang dibentuk oleh sendi 4 ( $\alpha_4$ ). Sedangkan sendi 5 merupakan pencapit (*gripper*). Robot lengan 4 DoF jika divisualkan tampak dari samping secara 2 dimensi seolah-olah akan tampak seperti robot lengan 3 DoF seperti pada Gambar 2.

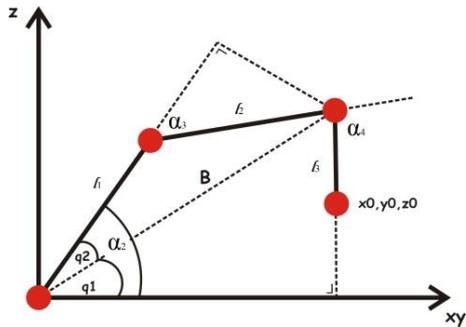


Gambar 2. Kinematika robot lengan

$$\begin{aligned}
 x_0 &= 0; \\
 x_1 &= (\ell_1) \cos(\alpha_2); \\
 x_2 &= (\ell_1) \cos(\alpha_2) + (\ell_2) \cos(\alpha_2 + \alpha_3); \\
 x_3 &= (\ell_1) \cos(\alpha_2) + (\ell_2) \cos(\alpha_2 + \alpha_3) + (\ell_3) \\
 &\quad \cos(\alpha_2 + \alpha_3 + \alpha_4); \\
 y_0 &= 0; \\
 y_1 &= (\ell_1) \sin(\alpha_2); \\
 y_2 &= (\ell_1) \sin(\alpha_2) + (\ell_2) \sin(\alpha_2 + \alpha_3); \\
 y_3 &= (\ell_1) \sin(\alpha_2) + (\ell_2) \sin(\alpha_2 + \alpha_3) + (\ell_3) \\
 &\quad \sin(\alpha_2 + \alpha_3 + \alpha_4); \tag{2.1}
 \end{aligned}$$

## 2.4 Invers Kinematika Robot Lengan

Invers Kinematika merupakan perhitungan berdasarkan posisi *end effector* untuk menentukan besar sudut joint yang akan terbentuk. Proses ini sangat berguna dalam robotika terutama pada proses simulasinya. Dengan menggunakan perhitungan invers kinematika, didapatkan besar pergerakan sudut yang dibentuk masing-masing joint secara langsung menggunakan tampilan simulasi robot lengan seperti pada Gambar 3. [3]



Gambar 3. Invers kinematika robot lengan

$$\alpha_3 = \cos^{-1} \frac{(B^2 - l_1^2 - l_2^2)}{2.l_1.l_2}$$

$$B = \sqrt{xy^2 + (z_0 + l_3)^2}$$

$$xy = \sqrt{x_0^2 + y_0^2}$$

$$\alpha_2 = \beta_1 + \beta_2$$

$$\beta_1 = \tan^{-1} \left( \frac{z_0 + l_3}{xy} \right)$$

$$\beta_2 = \tan^{-1} \left( \frac{l_2 \cdot \sin \alpha_3}{l_1 + (l_2 \cos \alpha_3)} \right)$$

$$\alpha_4 = 90^\circ + \alpha_2 - \alpha_3 \quad (2.2)$$

## 2.5 Kapsulasi Data

Dalam istilah pemrosesan data, kapsulasi merupakan pengelompokan beberapa nilai yang dikirimkan dalam satu paket data sehingga menghemat proses pengiriman data. Data tersebut disatukan sesuai dengan urutannya sehingga membentuk suatu baris data yang dikelompokkan menjadi satu paket. Sudut-sudut yang akan dikirimkan dikapsulasi menggunakan aplikasi pengendali sehingga menjadi satu deret bilangan dan hanya memerlukan satu kali proses pengiriman data ke mikrokontroler. Setelah itu paket data tersebut nantinya dipecah oleh mikrokontroler dan dikirimkan ke masing-masing servo yang dituju. Adapun rumus yang digunakan untuk kapsulasi dan dekapsulasi adalah sebagai berikut:

$$S_1 = \alpha_1 \times 1000000$$

$$S_2 = \alpha_2 \times 10000$$

$$S_3 = \alpha_3 \times 100$$

$$S_4 = \alpha_4$$

$$\text{paket data} = S_1 + S_2 + S_3 + S_4 \quad (2.3)$$

## 2.6 Dekapsulasi Data

Dekapsulasi merupakan proses pemecahan paket data (parsing data) yang dikirimkan ke mikrokontroler sehingga data yang diinginkan dapat diterima sesuai dengan alamat data yang dituju. Paket data dipecah menjadi beberapa bagian yang digunakan sebagai pulsa tiap motor servo. Dengan menggunakan rumus dekapsulasi, maka didapatkan sudut-sudut yang akan digunakan. Adapun rumus dekapsulasi adalah sebagai berikut:

$$S_4 = \text{paket data mod } 100$$

$$\alpha_4 = S_4$$

$$S_3 = \text{paket data mod } 10000$$

$$\alpha_3 = (S_3 - S_4) / 100$$

$$S_2 = \text{paket data mod } 1000000$$

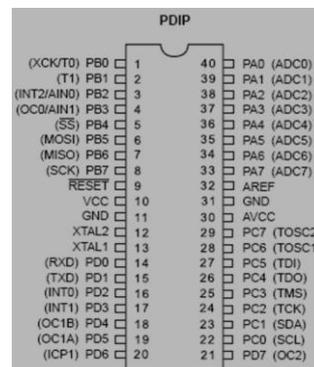
$$\alpha_2 = (S_2 - S_3) / 10000$$

$$S_1 = \text{paket data}$$

$$\alpha_1 = (S_1 - S_2) / 1000000 \quad (2.4)$$

## 2.7 Mikrokontroler ATmega16

Mikrokontroler ATmega16 merupakan mikrokontroler jenis AVR keluaran dari Atmel. Mikrokontroler ini mempunyai fasilitas lengkap dan didukung oleh *software BASCOM AVR* sebagai *compiler*. Mikrokontroler AVR ATmega16 memiliki saluran I/O 32 buah, yaitu PORT A, PORT B, PORT C, dan PORT D yang dapat digunakan. Selain itu mikrokontroler ini juga memiliki *ADC (Analog to Digital Converter)* 10 bit sebanyak 8 *channel*, 3 buah *timer/counter*, serta didukung port *USART programmable* untuk komunikasi serial. Lebih jelasnya dapat dilihat pada Gambar 4. [2]



Gambar 4. Susunan pin Mikrokontroler ATmega16

Pada penelitian ini, port pada mikrokontroler yang digunakan untuk mengendalikan motor servo adalah PORTA. PORTA (A.0-A.4) akan mengirimkan besar pulsa dari mikrokontroler ke motor servo sehingga servo dapat bergerak sesuai dengan besar pulsa yang diberikan.

## 2.8 Komunikasi Serial

Komunikasi serial adalah suatu proses pengiriman data secara sequential atau satu persatu melalui kanal informasi. Komunikasi ini mempunyai kecepatan transfer data yang rendah tetapi cocok untuk komunikasi jarak jauh. [4]

Untuk jalur komunikasi antar piranti, PORTD (D.0 dan D.1) pada mikrokontroler akan digunakan sebagai pengiriman dan penerimaan data secara serial. Data yang dikirimkan oleh *device* lain/komputer secara serial akan diterima oleh mikrokontroler melalui jalur Port D.0 *Receive Data (Rx)*, dan data yang akan dikirimkan secara serial oleh mikrokontroler melalui jalur Port D.1 *Transmite Data (Tx)*.

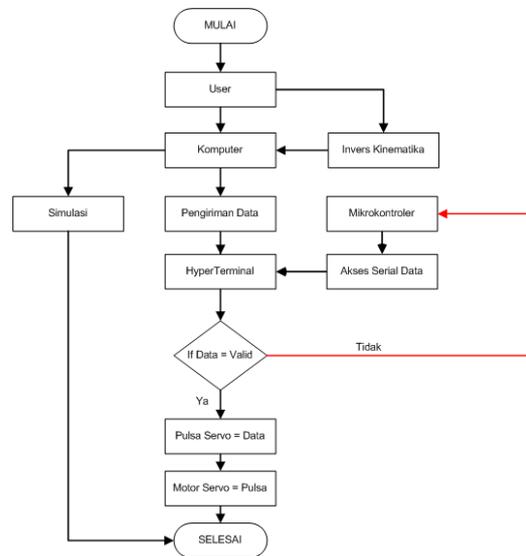
## 3. METODOLOGI

Penelitian ini melalui beberapa tahapan seperti pengenalan robot, perancangan dan pembuatan aplikasi serta uji coba. Pengenalan robot meliputi pengenalan kondisi fisik robot dan arah pergerakannya. Selanjutnya dilakukan perancangan aplikasi yang akan dibuat untuk mengendalikan robot lengan. Fitur-fitur yang dimiliki aplikasi pengendali berupa tampilan visualisasi robot lengan dan pengendali gerakan robot lengan yaitu *trackbar* sesuai dengan banyaknya sendi yang akan digerakkan. Setelah selesai dibuat, aplikasi diuji untuk memvisualisasi dan mengendalikan pergerakan robot lengan.

## 4. PERANCANGAN PENELITIAN

### 4.1 Diagram Alur Program

Adapun diagram alur program yang akan digunakan pada tugas akhir ini adalah sebagai berikut.

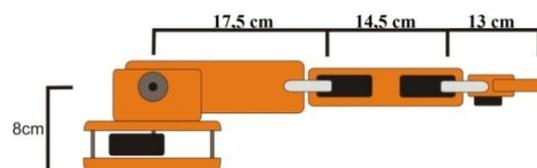


Gambar 5. Diagram alur program

Diagram ini merupakan gambaran umum dimana robot dapat dikendalikan menggunakan komputer berdasarkan perhitungan dan simulasi dengan mengirimkan data ke sistem minimum mikrokontroler menggunakan jalur komunikasi serial. Jalur serial ini nantinya diakses oleh mikrokontroler sebagai data untuk menginstruksikan pergerakan motor servo.

### 4.2 Konstruksi Robot Lengan

Robot lengan 4 DoF yang digunakan merupakan sebuah robot yang memiliki dua ruas lengan yang dihubungkan dengan sendi (*joint*) dengan pencapit (*gripper*) di ujung lengannya. Adapun dimensi dari robot lengan dapat dilihat pada Gambar 6.

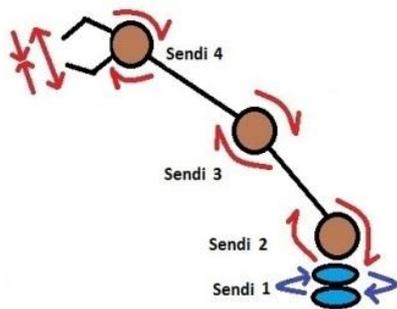


Gambar 6. Dimensi robot lengan 4 DoF tampak dari samping

Robot lengan 4 DoF ini terbuat dari papan kayu lempung dengan ketebalan 0,6 cm. Keunggulan dari kayu lempung ini adalah bahan ini sangat murah dan ringan sehingga mengurangi beban yang harus diterima motor servo dalam menggerakkan robot lengan.

### 4.3 Mekanika Robot Lengan

Robot Lengan 4 DoF adalah robot manipulator yang memiliki empat sendi yang dapat bergerak ditambah dengan satu pencapit (*gripper*) pada ujung lengan robot yang berfungsi sebagai pencengkram barang. Sendi-sendi pada robot lengan digerakkan menggunakan motor servo yang telah terhubung dengan rangkaian Mikrokontroler. Pergerakan lengan robot tersebut bekerja berdasarkan pulsa yang diberikan pada tiap motor servo sehingga pergerakan robot lengan dapat ditentukan.

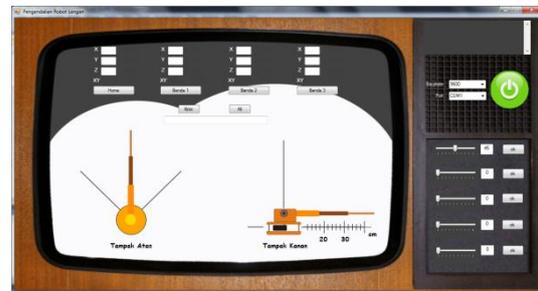


Gambar 7. Mekanika Robot Lengan 4 DoF

Pada Gambar 7, robot lengan memiliki 4 sendi dan *gripper* yang dapat digerakkan. Tiap-tiap sendi tersebut menggunakan motor servo sebagai penggerakannya. Servo 1 terletak di bagian poros bawah robot dan digunakan untuk menggerakkan robot lengan berputar ke kiri dan kanan. Servo 2 digunakan untuk menggerakkan lengan 1, lengan 2 dan *gripper* ke atas dan ke bawah. Servo 3 digunakan untuk menggerakkan lengan 2 dan *gripper* ke atas dan bawah. Servo 4 digunakan untuk menggerakkan *gripper* ke atas dan bawah, sedangkan servo 5 digunakan untuk menggerakkan *gripper*.

### 4.4 Desain Visualisasi Dan Pengendali Pergerakan Robot Lengan 4 Dof

Sebelum memulai membangun sebuah aplikasi simulasi, hal yang paling mendasar untuk dikerjakan adalah bagaimana aplikasi itu bekerja dan tampilan dari aplikasi tersebut. Disini tampilan simulasi akan terbagi menjadi 2 kendali, yaitu kendali sudut dan kendali tampilan. Kendali sudut berfungsi menggerakkan pergerakan robot lengan sesuai dengan sudut yang diberikan, sedangkan kendali tampilan akan menampilkan pergerakan manipulator robot lengan dari 2 sudut pandang yang berbeda, yaitu samping dan atas. Pergerakan ini nantinya menggunakan perhitungan kinematika robot lengan.



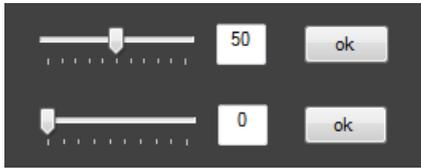
Gambar 8. Desain visualisasi robot lengan menggunakan *Visual Basic*

## 5. PENGUJIAN ROBOT LENGAN

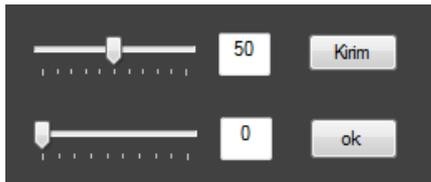
### 5.1 Pengujian Kendali Gerak Robot Lengan 4 Dof

Pada tahap ini pergerakan robot lengan diuji dengan menggunakan aplikasi pengendali sudut robot lengan. Pengendali dapat menentukan servo mana yang akan diaktifkan dengan menggunakan tombol "Ok". *Button* ini akan mengirimkan *Inkey* huruf yang menandakan bahwa servo mana yang harus digerakkan. Setelah itu, aplikasi pengendali dapat menentukan besar pulsa yang diberikan kepada tiap servo menggunakan *Trackbar*. *Trackbar* berfungsi sebagai pengendali besar sudut yang akan dibentuk oleh visualisasi robot lengan serta memberikan nilai kepada *hidden* tombol "Kirim" untuk dikirimkan melalui akses serial ke port D mikrokontroler sebagai nilai pulsa servo.

Tombol “Kirim” ini akan muncul jika tombol ‘Ok’ di klik seperti pada Gambar 9.



Gambar 9a. Tampilan Tombol “Ok” aktif



Gambar 9b. Tampilan Tombol “Kirim” aktif



Gambar 10. Pengendalian robot lengan menggunakan *Visual Basic*

## 5.2 Pemrograman Mikrokontroler Sebagai Pengendali

Mikrokontroler diprogram untuk menerima dua buah inputan, yaitu *Inkey* huruf dan *Inkey* angka. *Inkey* huruf berfungsi sebagai penentu servo mana yang harus diaktifkan. Setelah itu barulah *Inkey* angka dikirimkan yang berfungsi sebagai penentu seberapa besar pulsa yang harus diberikan ke motor servo yang diaktifkan. *Inkey* huruf terdiri dari 5 karakter yang berbeda, yaitu *Inkey* A, B, C, D dan E untuk mengaktifkan 5 servo, sedangkan *Inkey* huruf F digunakan untuk mengaktifkan semua motor servo (kecuali *gripper*) sesuai dengan urutan yang ditentukan. Untuk lebih jelasnya dapat dilihat pada Tabel 1 .

Tabel 1. Data Inputan ke Mikrokontroler

Inkey Huruf	Servo	Inkey Angka	Port
A	1	0-90	Port A.0
B	2	0-90	Port A.1
C	3	0-90	Port A.2
D	4	0-90	Port A.3
E	5	0-90	Port A.4
F	1-4	Baris <i>Inkey</i> Angka	Port A.0-A.3

Setiap *Inkey* huruf masing-masing memiliki *Inkey* angka sendiri (kecuali *Inkey* F) yang terdiri dari *Inkey* 0 hingga 90 sebagai acuan untuk menentukan besar sudut putaran motor servo yang diaktifkan. Namun, pada *Inkey* F, baris *Inkey* angkanya merupakan gabungan baris *Inkey* angka dari setiap sudut yang “dikapsulasi” menjadi suatu paket data. Paket data ini yang dikirimkan ke mikrokontroler untuk di “dekapsulasi”.

Pada pemrograman mikrokontroler ini, sebagai contoh jika komputer memberikan *Inkey* A, maka mikrokontroler akan mengirimkan tulisan “Respon OK” ke komputer dan memanggil perintah “Olah” untuk mengaktifkan servo 1. Selanjutnya komputer akan mengirimkan *Inkey* angka sebagai besar nilai pulsa untuk servo 1. Jika “Olah” dieksekusi, mikrokontroler akan mengirimkan besar pulsa yang diterima servo.

Program 1. Program untuk Mengaktifkan Servo 1

```

1) Awal:
2) Do
3) Char = Inkey()
4) If Char = "A"
   Then
5) Print "Respon OK"
6) Gosub Olaha
7) End If
8) Olaha:
9) Do
10) Servoin = Inkey()
11) Input Servoin Noecho
12) Servo(1) = Servoin
   +50
13) Print "Servo1 = " ;
   Servo(1)
14) Waitms 6
15) Goto Awal
16) Loop

```

Adapun tujuan dari penggunaan *Inkey* huruf pada pemrograman ini adalah sebagai *parsing data* yang akan dikirimkan ke mikrokontroler. *Parsing data* bermanfaat agar pada saat pengendalian servo tidak terjadi kesalahan dalam pengaktifan servo yang dituju sehingga pengendalian berhasil

dilaksanakan. Oleh karena itu, *Inkey* huruf dikirimkan terlebih dahulu yang bertujuan sebagai perintah ke mikrokontroler bahwa servo yang diperintahkan akan bergerak.

### 5.3 Penentuan Ruang Kerja Robot Lengan 4 Dof

Penentuan ruang kerja robot lengan dilakukan dengan cara menggerakkan robot lengan sesuai dengan besar *DoF* yang ditentukan pada masing-masing sendi (*joint*). Pengujian ini dilakukan untuk mendapatkan batas maksimal dari pergerakan robot yang dapat dilakukan. Berdasarkan besar *DoF* yang dimiliki oleh tiap sendi-sendi, didapatkan batasan-batasan daerah kerja (*work space*) dan daerah pengoperasian (*operational space*) yang dapat dilakukan oleh robot lengan saat bekerja.

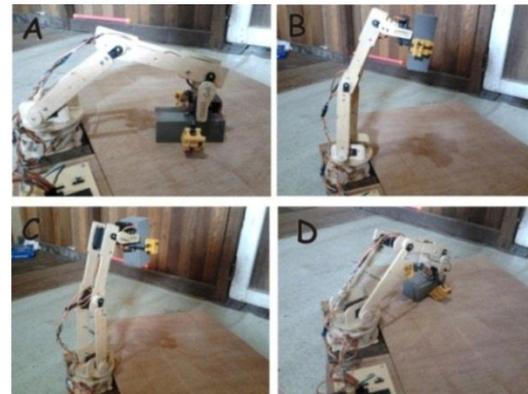
Dari hasil pengujian tersebut, didapatkan batasan-batasan daerah kerja dan daerah pengoperasian pergerakan robot lengan. Daerah kerja robot memiliki batas pergerakan robot lengan dari rentang 0-30 cm dengan daya jangkauan robot lengan dari rentang 19-26 cm sebesar  $90^\circ$ . Hal ini disebabkan karena terbatasnya masing-masing *DoF* yang dapat dilakukan yaitu  $\leq 90^\circ$  sehingga pergerakan yang dapat dilakukan pun terbatas.

### 5.4 Pengujian Pergerakan Robot Lengan 4 Dof

Pada tahapan ini dilakukan pengujian terhadap pergerakan robot lengan dalam mengambil suatu objek. Dengan memanfaatkan kendali sudut dan visualisasi aplikasi pengendali robot lengan, robot lengan diarahkan dan dikendalikan menuju posisi objek berada.

Pengujian pengendalian robot lengan ini terlebih dahulu melewati beberapa tahapan. Tahap pertama adalah robot lengan dikendalikan untuk mengambil suatu objek dengan koordinat xyz (26,0,0). Tahap kedua yaitu dengan perhitungan invers kinematika yang dimiliki aplikasi pengendali, didapatkan sudut yang harus dibentuk setiap sendi mulai dari sendi 1 yaitu  $45^\circ$ ,  $49^\circ$ ,  $50^\circ$  dan  $89^\circ$ . Sedangkan

*gripper* berada pada posisi terbuka. Tahap ketiga, *Trackbar* akan bergerak sesuai dengan sudut yang didapat sehingga visualisasi akan menggambarkan pergerakan yang akan dilakukan robot lengan. Tahap keempat, sudut yang didapat dari *Trackbar* “dikapsulasi” dan dikirimkan ke mikrokontroler. Setelah sampai ke mikrokontroler, sudut yang diterima di “dekapsulasi” dan data sudut-sudut tersebut dikirimkan sesuai dengan servo yang dituju sehingga robot akan bergerak sesuai dengan sudut yang dikirimkan. Tahap kelima, setelah robot bergerak dan sampai di tujuan, *gripper* diperintahkan untuk menjepit objek sehingga bisa diangkat. Kemudian tahap keenam, robot lengan pun diperintahkan untuk memindahkan objek ke tempat yang ingin dituju selanjutnya. Hasil yang didapat adalah robot lengan dapat bekerja sesuai dengan visualisasi dan instruksi sudut yang diberikan seperti pada Gambar 11 dan Gambar 12.



Gambar 11. Robot lengan mengambil objek pada posisi yang ditentukan



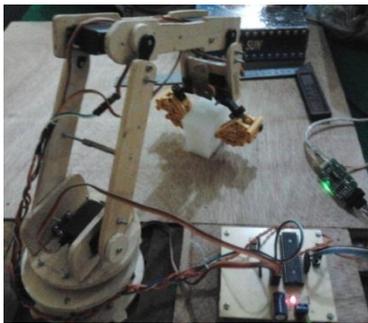
Gambar 12. Visualisasi robot lengan saat mengambil objek pada posisi yang ditentukan

### 5.5 Pengujian Kerja Robot Lengan 4 Dof

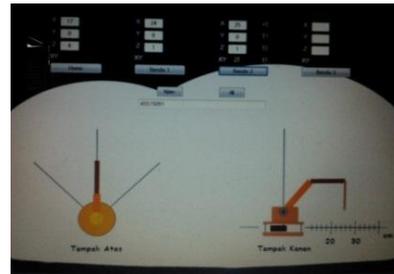
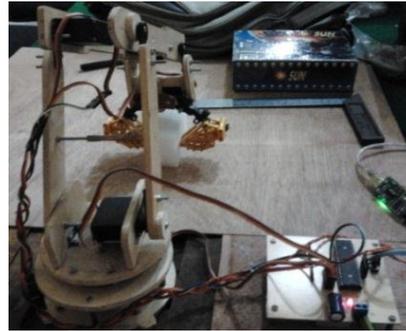
Pada tahapan ini, robot lengan ditugaskan untuk mengambil 3 objek di 3 titik yang berbeda untuk dipindahkan ke tempat pengumpulan objek (*home*). Penugasan yang pertama adalah ditentukan terlebih dahulu posisi *home* yang diinginkan, misalkan pada posisi (17,0,4) kemudian klik tombol “Home” dan “Kirim”. Setelah itu, ditentukan posisi objek 1, objek 2, dan objek 3, misalkan pada posisi (24,8,1), (25,0,1), dan (23,-7,1). Jika robot ingin diperintahkan mengambil objek 1 klik tombol “Benda 1” dan “Kirim”, maka robot akan mengambil objek ke posisi objek 1 dan membawanya ke posisi *home*. Begitu pula untuk mengambil objek 2 dan objek 3.



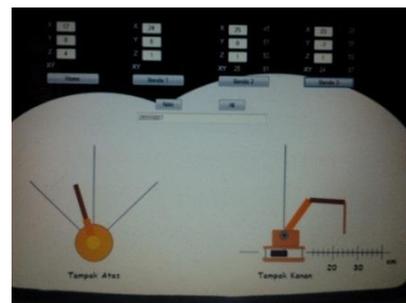
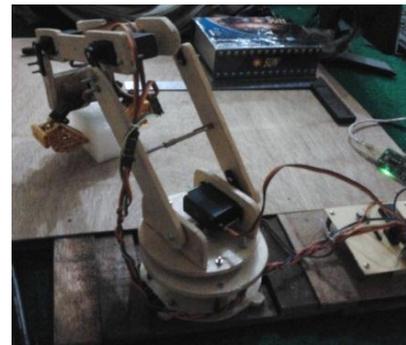
Gambar 13. Posisi *Home* robot lengan



Gambar 14. Posisi robot mengambil objek 1 dan visualisasinya



Gambar 15. Posisi robot mengambil objek 2 dan visualisasinya



Gambar 16 Posisi robot mengambil objek 3 dan visualisasinya

Pada percobaan ini, saat pengambilan objek dilakukan terkadang objek yang dituju tidak dapat diambil/ terlepas dikarenakan ukuran *gripper* yang besar. Oleh karena itu pemilihan *gripper* juga menjadi faktor utama dalam pengambilan objek. Adapun hasil pengujian pergerakan robot lengannya dapat dilihat pada Tabel 2.

Tabel 2. Tabel Hasil Pengujian Pergerakan Robot Lengan

Koordinat	Visual			Robot		
	x	y	Z	x	y	z
20,0,1	Ok	Ok	Ok	- 1 cm	pas	+ 0,5 cm
21,0,1	Ok	Ok	Ok	- 0,7 cm	pas	+ 0,5 cm
22,0,1	Ok	Ok	Ok	- 0,4 cm	pas	+ 0,5 cm
23,0,1	Ok	Ok	Ok	- 0,2 cm	pas	+ 0,5 cm
24,0,1	Ok	Ok	Ok	pas	pas	+ 0,5 cm
25,0,1	Ok	Ok	Ok	+ 0,2 cm	pas	+ 0,5 cm
26,0,1	Error			Diluar jangkauan		
20,3,1	Ok	Ok	Ok	- 1 cm	pas	+ 0,5 cm
21,3,1	Ok	Ok	Ok	- 1 cm	pas	+ 0,5 cm
22,3,1	Ok	Ok	Ok	- 0,8 cm	pas	+ 0,5 cm
23,3,1	Ok	Ok	Ok	- 0,5 cm	pas	+ 0,5 cm
24,3,1	Ok	Ok	Ok	- 0,3 cm	pas	+ 0,5 cm
25,3,1	Error			Diluar jangkauan		
20,7,1	Ok	Ok	Ok	- 0,2 cm	- 0,3 cm	+ 0,5 cm
21,7,1	Ok	Ok	Ok	- 0,3 cm	-0,5 cm	+ 0,5 cm
22,7,1	Ok	Ok	Ok	- 0,2 cm	- 1 cm	+ 0,5 cm
23,7,1	Ok	Ok	Ok	- 0,2 cm	- 1,3 cm	+ 0,5 cm
24,7,1	Error			Diluar jangkauan		
20,10,1	Ok	Ok	Ok	- 0,1 cm	- 1,5 cm	+ 0,5 cm
21,10,1	Ok	Ok	Ok	pas	- 2 cm	+ 0,5 cm
22,10,1	Ok	Ok	Ok	pas	- 2,7 cm	+ 0,5 cm
23,10,1	Error			Diluar jangkauan		

### 5.6 Analisa Pengujian

Dari hasil pengujian yang telah dilakukan, robot lengan berhasil digerakkan dan disimulasikan menggunakan *Visual Basic 2010* dengan memanfaatkan jalur komunikasi serial dan perhitungan kinematika. Adapun parameter yang menentukan keberhasilan pengujian ini adalah sebagai berikut:

Tabel 3. Tabel Parameter Keberhasilan

No	Pengujian	Parameter	Indikator	Ket.
1	Pemrograman Mikrokontroler sebagai Pengendali	Mikrokontroler menerima dan mengeksekusi data ( <i>Inkey</i> ) yang dikirimkan oleh komputer dengan mengirimkan tulisan "Respon Ok" dan besar pulsa yang diterima servo ke aplikasi.		Berhasil

2	Pengujian Visualisasi dan Pengendali Pergerakan Robot Lengan 4 DoF	Aplikasi dapat mensimulasikan pergerakan robot lengan berdasarkan pergerakan robot lengan sebenarnya menggunakan perhitungan kinematika, misalnya sudut-sudut yang dibentuk yaitu $0^\circ$ , $49^\circ$ , $50^\circ$ dan $89^\circ$ .		Berhasil
3	Pengujian Pergerakan Robot Lengan 4 DoF	Robot lengan dapat melakukan pergerakan sesuai dengan yang diperintahkan aplikasi pengendali.	Untuk pengujian pengendalian pergerakan robot lengan ini dapat dilihat pada alamat <i>website</i> berikut.  <a href="http://www.youtube.com/watch?v=B2r-YR_CJn8">http://www.youtube.com/watch?v=B2r-YR_CJn8</a>	Berhasil
4	Pengujian Kerja Robot Lengan 4 DoF	Robot lengan dapat mengambil objek dengan posisi yang ditentukan dan memindahkannya ke posisi <i>home</i> yang telah ditentukan pula serta divisualkan gerakannya.	Robot lengan bergerak menuju posisi objek dan membawa objek. Setelah itu kembali ke posisi <i>home</i> .	Berhasil

## 6. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa:

1. Pergerakan robot lengan dapat disimulasikan secara visual menggunakan *Visual Basic* dengan memanfaatkan perhitungan kinematika sehingga pergerakan robot lengan dapat diprediksi.
2. Dengan menggunakan visualisasi robot lengan, didapatkan besar sudut robot lengan yang harus dibentuk dengan urutan tiap sudutnya misalkan  $0^\circ$ ,  $49^\circ$ ,  $50^\circ$  dan  $89^\circ$ . Data sudut ini di "kapsulasi" dan dikirimkan ke mikrokontroler. Setelah itu mikrokontroler akan melakukan

“dekapsulasi” dan mengirimkan data sudut tersebut sesuai dengan servo yang dituju. Oleh karena itu pergerakan robot lengan dan simulasinya dapat bekerja secara sinkron karena data sudut yang dikirimkan ke simulasi dan robot lengannya adalah sama.

3. Robot lengan dapat diperintahkan bergerak menuju koordinat yang diberikan, misalkan koordinat (26,0,0). Kemudian robot lengan juga dapat melakukan gerakan seperti mengambil suatu objek di 3 tempat berbeda, misalkan pada posisi (24,8,1), (25,0,1), dan (23,-7,1) dan memindahkan objek tersebut ke posisi *home*, misalkan pada posisi (17,0,4) secara bergiliran.

## 7. SARAN

Adapun saran yang dapat diberikan untuk penelitian selanjutnya yaitu perlu dilakukan penelitian lebih lanjut mengenai penganimasian suatu objek secara *3D* serta pengendalian robot lengan menggunakan jalur komunikasi nirkabel (*wireless*).

## DAFTAR PUSTAKA

- [1] Supriyono, Rakhmat. 2010. *Desain Komunikasi Visual Teori dan Aplikasi*. Yogyakarta: Penerbit Andi
- [2] Handajadi, Wiwik., dkk. 2009. *Pembacaan Output Timbangan Digital Jarak Jauh Dengan Menggunakan Pemrograman Visual Basic 6.0*. Jurusan Teknik Elektro, Fakultas Teknologi Industri Institut Sains & Teknologi AKPRIND. Yogyakarta
- [3] Budiharto, Widodo. 2010. *Robotika Teori dan Implementasi*. Jakarta: Elex Media Komputindo.
- [4] Budiharto, Widodo. 2009. *Membuat Sendiri Robot Humanoid*. Jakarta: Elex Media Komputindo

