

ANALISA *FORWARD KINEMATIC* PADA SIMULATOR *ARM ROBOT 5 DOF* YANG MENGINTEGRASIKAN MIKROKONTROLER ARDUINO-UNO DAN LABVIEW

Munadi

Laboratorium Komputasi, Jurusan Teknik Mesin, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Soedarto SH, Tembalang, Semarang 50272 Indonesia
E-mail: munadi@undip.ac.id

ABSTRACT

An arm robot simulator has been developed, that capable in simulating a 5 degree of freedom robot manipulator, in which it was equipped with two-finger gripper mechanism at end-effector. This simulator is designed for educational purposes so that many students can easily understand when learning about robot manipulator. The simulator was developed using Arduino Uno with LabVIEW through the Firmata interface for controlling the actuators (servo motors). Arduino Uno was chosen because it can interact with LabVIEW that will be able to control the angular position of servo motor easily. Angular position errors that occur on the servo motor can be solved by using a numerical program functions and numerical multiply divided on LabVIEW. For analysis, this paper presents the forward kinematics problem which is concerned with the relationship between the individual joints of the arm robot simulator and the position and orientation of the tool or end-effector. The analysis result is carried out in MATLAB.

Keywords: arm robot, robot manipulator, Arduino, LabView, forward kinematic

1. PENDAHULUAN

Perkembangan teknologi yang meningkat dari tahun ke tahun telah menuntut manusia untuk terus menciptakan alat-alat yang dapat membantu menyelesaikan pekerjaan demi perbaikan kualitas maupun kuantitas dari produk yang dihasilkan. Salah satu contoh alat tersebut adalah robot, dimana alat ini sering diaplikasikan pada bidang otomasi yang dirancang untuk membantu mengerjakan proses kerja yang umumnya memiliki ketelitian yang baik jika dibandingkan dengan manusia menyangkut konsistensi.

Ada banyak definisi yang dikemukakan oleh para ahli mengenai robot. Orang awam beranggapan bahwa robot mengandung pengertian suatu alat yang menyerupai manusia, dimana struktur tubuhnya terbuat dari logam (Novia, Leli, 2004). Dari berbagai definisi, robot dapat dinyatakan sebagai sebuah sistem mekanik yang dikendalikan oleh seperangkat sistem komputer menurut sequence program yang ditanam pada komputer tersebut untuk melakukan berbagai tugas sesuai dengan perintah baik secara semi ataupun otomatis penuh.

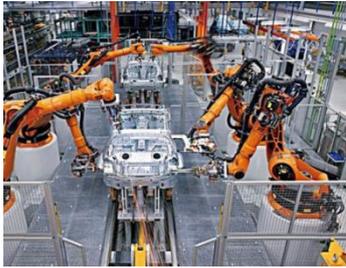
Pada awalnya aplikasi robot hampir tidak dapat dipisahkan dengan industri sehingga muncul istilah *industrial robot*. Definisi yang populer ketika itu, robot industri adalah suatu robot tangan (*arm robot*) yang diciptakan untuk berbagai keperluan dalam meningkatkan produksi, memiliki bentuk lengan-lengan kaku yang terhubung secara seri dan memiliki sendi yang dapat bergerak berputar (rotasi) atau memanjang/memendek (translasi atau prismatik). Satu sisi lengan yang disebut sebagai pangkal ditanam pada bidang atau meja yang statis (tidak bergerak), sedangkan sisi yang lain yang disebut sebagai ujung (*end effector*) dapat ditambah dengan tool tertentu sesuai dengan tugas robot, misalnya gripper.

Dengan pesatnya perkembangan robot khususnya di Indonesia saat ini, maka banyak siswa SD sampai dengan mahasiswa tertarik belajar tentang robot. Mereka belajar dari yang sederhana misalnya *line-follower* sampai dengan yang kompleks seperti misalnya *humanoid robot*. Sementara itu, untuk mempelajari robot dasar yang dipakai untuk industri tentunya butuh peralatan (*tools*) penunjang yang harganya tidak murah. Oleh sebab itu, maka penelitian ini akan membahas pembuatan simulator *arm robot* yang merupakan prototipe robot manipulator yang dibutuhkan di dunia industri dengan biaya yang terjangkau untuk kepentingan dunia pendidikan. Pembuatan simulator *arm robot* dirancang memiliki 5 *degree of freedom* (dof) dengan menggunakan bantuan software LabVIEW dan mikrokontroler Arduino Uno sehingga memudahkan komunikasi antara perangkat keras *arm robot* dengan perangkat lunak yang digunakan untuk mengontrol gerakan robot. Selanjutnya dilakukan analisa *forward kinematic* untuk mendeskripsikan orientasi dan posisi *end effector* terhadap *base*.

2. ROBOT MANIPULATOR

Industrial robots adalah robot-robot yang digunakan di dalam industri. Robot-robot industri ini dapat digunakan untuk proses otomasi dalam produksi karena memiliki keakuratan yang tinggi dalam menjalankan tugasnya, misalkan untuk proses *welding* pada industri otomotif yang ditunjukkan pada Gambar 1. Contoh robot industri yang populer digunakan adalah robot manipulator yang ditunjukkan pada Gambar 2. Robot manipulator memiliki sebuah *end effector* seperti tangan manusia, diantaranya adalah *gripper* yang berfungsi untuk memegang atau memindahkan barang.

Robot manipulator tersebut merupakan sebuah rangkaian benda kaku (*rigid bodies*) terbuka yang terdiri atas sendi (*joint*) dan terhubung dengan *link* dimana setiap posisi *joint* ditentukan dengan variabel tunggal sehingga jumlah *joint* sama dengan nilai derajat kebebasan [1].



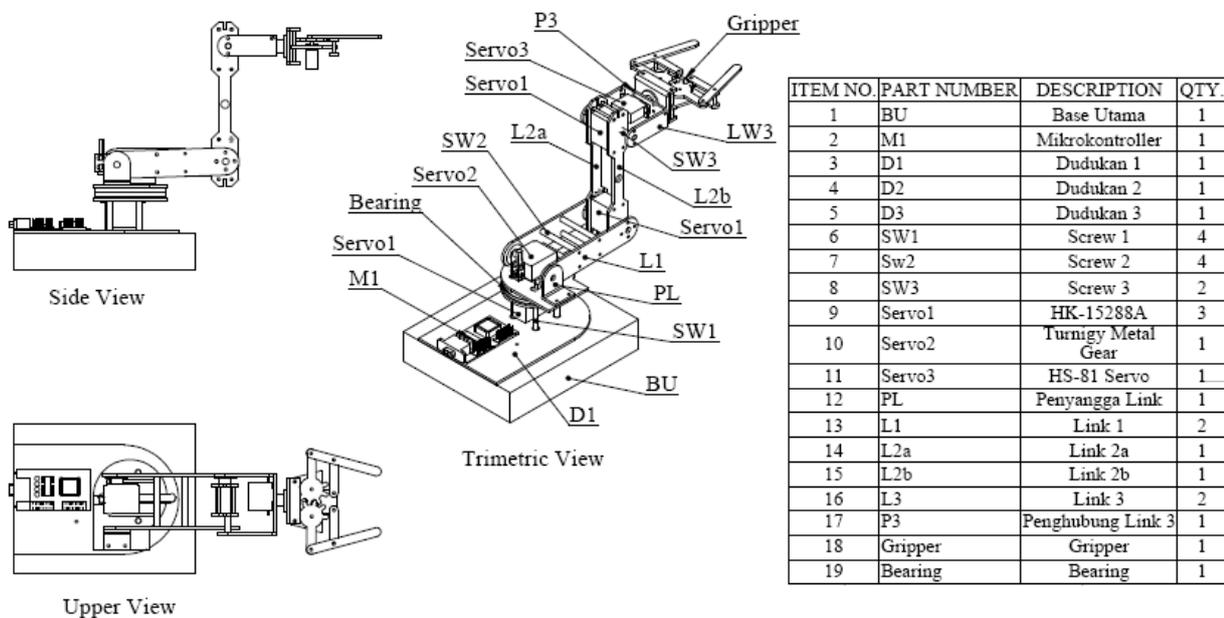
Gambar 1. Robot industri



Gambar 2. Robot manipulator

3. SIMULATOR ARM ROBOT

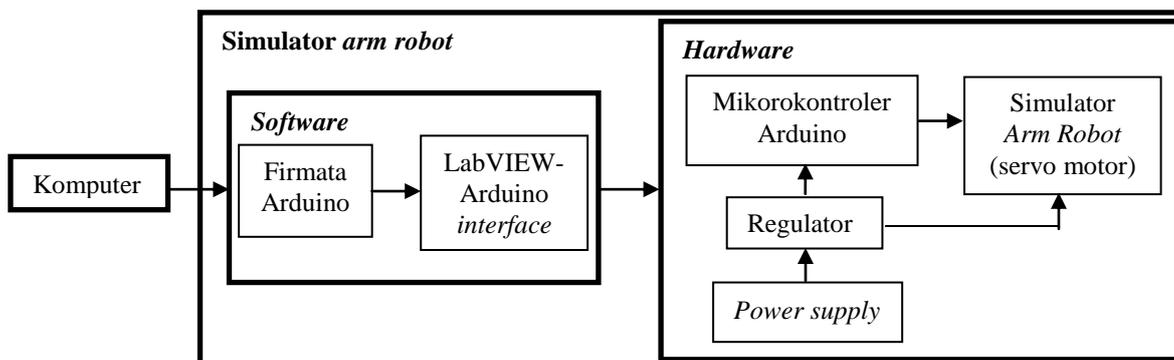
Menjadi pertimbangan bagi peneliti untuk membuat alat bantu yang dapat memudahkan pelajar dan mahasiswa dalam mempelajari robot dengan mudah dan murah. Oleh sebab itu, diciptakan simulator *arm robot* 5 dof sebagai prototipe dari robot manipulator yang sering digunakan oleh industri manufaktur. Gambar 3 menunjukkan desain simulator *arm robot* 5 dof yang akan dibuat.



Gambar 3. Desain simulator *arm robot* 5 dof.

3.1 Perangkat Keras

Simulator *arm robot* tersusun atas perangkat keras (*hardware*) dan perangkat lunak (*software*). Gambar 4 menunjukkan komponen perangkat keras dan lunak untuk simulator *arm robot*.



Gambar 4. Diagram *hardware* dan *software arm robot* 5 dof

Selanjutnya, untuk perangkat keras utama yang dibutuhkan terdiri atas:

a. Mikrokontroler

Sebuah robot membutuhkan kontroler (*controller*) yang berfungsi untuk mengontrol pergerakan robot yang menyimpan informasi berkaitan dengan data-data robot, seperti data sequence program gerakan robot dengan menggunakan bahasa pemrograman tertentu yang telah diinputkan, misalnya bahasa C atau C++. Pada penelitian ini, simulator *arm robot* menggunakan mikrokontroler Arduino Uno yang didalamnya terdapat ATmega328, dan ditunjukkan pada Gambar 5. Mikrokontroler ini mempunyai pin yang cukup banyak pin, memori penyimpanan yang besar serta harganya yang terjangkau. Mikrokontroler ini menggunakan data serial yang dapat dihubungkan dengan USB (*Universal Serial Bus*) sehingga sangat memudahkan untuk meng-*compile* program dari labtop/PC ke mikrokontroler. Adapun spesifikasi mikrokontroler ini ditunjukkan pada Tabel 1.



Gambar 5. Arduino Uno.

Tabel 1. Spesifikasi Arduino.

No	Hal	Spesifikasi
1	Mikrokontroler	ATmega328
2	Tegangan operasi	5 Volt
3	Tegangan masukan	7-12 Volt
4	Batasn tegangan masukan	6-20 Volt
5	Digital I/O pins	14 (6 keluaran PWM)
6	Analog input pins	6
7	Arus per I/O pin	40 mA
8	Arus untuk 3,3 volt pin	50 mA
9	Flash memory	32 kB
10	SRAM	2 kB
11	EEPROM	1 kB
12	Clock speed	16 Mhz

b. Aktuator

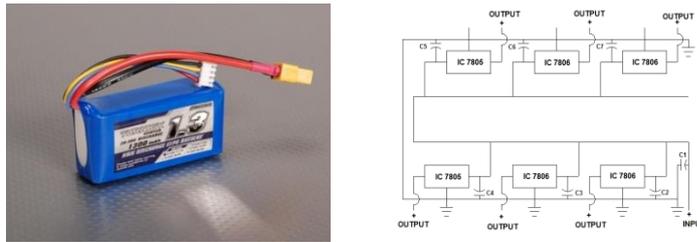
Aktuator yang dipilih adalah *servo motor* yang merupakan salah satu jenis motor dengan sistem *closed feedback*. *Servo motor* yang ditunjukkan pada Gambar 6 terdiri atas dari sebuah motor, serangkaian *gear*, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran, sedangkan sudut dari sumbu *servo motor* diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Pada pembuatan simulator *arm robot*, *servo motor* yang digunakan untuk joint 1, 3 dan 4 adalah servomotor Hobbyking HK-15288A, untuk joint 2 adalah Turnigy metal gear, dan untuk joint 5 sekaligus pada gripper digunakan *servo motor* dari Hitec HS-81.



Gambar 6. Servo motor.

c. Regulator

Agar robot dapat bekerja dengan baik, maka komponen yang tidak kalah penting adalah sebuah regulator yang berfungsi sebagai rangkaian regulasi atau pengatur tegangan keluaran dari sebuah catu daya (*power supply*) agar efek naik atau turunnya tegangan tidak mempengaruhi tegangan yang masuk ke aktuator atau dapat dikatakan agar tegangan input stabil. Catu daya yang dipilih adalah baterai dengan merk Turnigy yang memiliki kapasitas 1300 mAh, tegangan 11,1 volt. Gambar 7 menunjukkan baterai dan rangkaian regulator yang digunakan. Di sisi lain, simulator *arm robot* didesain menggunakan *servo motor* HK-15288A yang membutuhkan tegangan input 6 volt dan *servo motor* HS-81 yang membutuhkan tegangan input 5 volt. Oleh sebab itu, pada rangkaian regulator ditambah IC-7806 dan IC-7805 untuk menurunkan tegangan sehingga sesuai dengan input *servo motor* [2].



Gambar 7. Baterai dan rangkaian regulator.

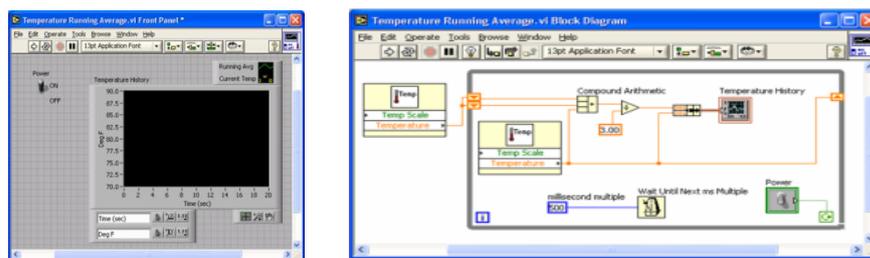
3.2 Perangkat Lunak

a. MATLAB

Merupakan bahasa pemrograman tingkat tinggi yang berbasis pada matriks. MATLAB secara umum digunakan untuk matematika dan komputasi, pengembangan visualisasi, pemodelan, simulasi dan pembuatan prototype, analisa data, eksplorasi, dan visualisasi, pembuatan aplikasi, termasuk pembuatan antarmuka grafis. MATLAB adalah sistem sistem interaktif dengan elemen dasar basis data *array* yang dimensinya tidak perlu dinyatakan secara khusus sehingga memungkinkan untuk memecahkan banyak masalah perhitungan teknik, khususnya yang melibatkan matriks dan vector dengan waktu yang lebih singkat dari waktu yang dibutuhkan untuk menulis program C.

b. LabVIEW

LabVIEW adalah produk dari National Instruments yang berupa *software* pengembangan program aplikasi dan *hardware input-output* untuk keperluan akusisi dan pengendalian. Sedangkan perangkat lunak atau software LabVIEW merupakan sebuah bahasa pemrograman *graphical* yang menggunakan simbol (ikon) untuk membuat aplikasi. Sementara itu, Visual Instruments (VIs) adalah program LabVIEW yang menirukan instrumen sebenarnya dalam bentuk simbol-simbol. Selanjutnya untuk membuat tampilan program aplikasi LabVIEW, digunakan *tools* dan objek. Tampilan aplikasi ini kemudian dikenal dengan jendela *front panel* yang berisikan kode representasi dari simbol sebagai fungsi untuk mengatur objek. Adapun *source code* simbol tersebut ada dalam tampilan jendela *block diagram*. Jadi pada dasarnya *software* LabVIEW terdiri dari 3 (tiga) komponen utama, yaitu *front panel*, *block diagram* dan tipe data [3]. *Front panel* merupakan penghubung (*interface*) antara pengguna (*user*) dengan program aplikasi. *Block diagram* merupakan jendela tempat menuliskan perintah dan fungsi, berisikan *source code* berupa simbol-simbol, *node* dan garis sebagai *data flow* untuk mengeksekusi program termasuk kode dari *front panel*. Gambar 8 (a) menunjukkan tampilan jendela *front panel* dan (b) menunjukkan tampilan jendela *block diagram* LabVIEW. Sedangkan dalam membuat aplikasi VIs, harus diperhatikan tipe data tiap simbol agar *data flow* dapat berjalan semestinya. Tipe data yang tersedia yaitu numerik, boolean dan *string*. Tipe data dari sebuah simbol dapat diketahui dari warna *node* atau warna kabel ketika dihubungkan ke simbol lainnya



(a)

(b)

Gambar 8. Front panel (a) dan block diagram (b) pada LabVIEW

c. Arduino Software (Firmata-Arduino)

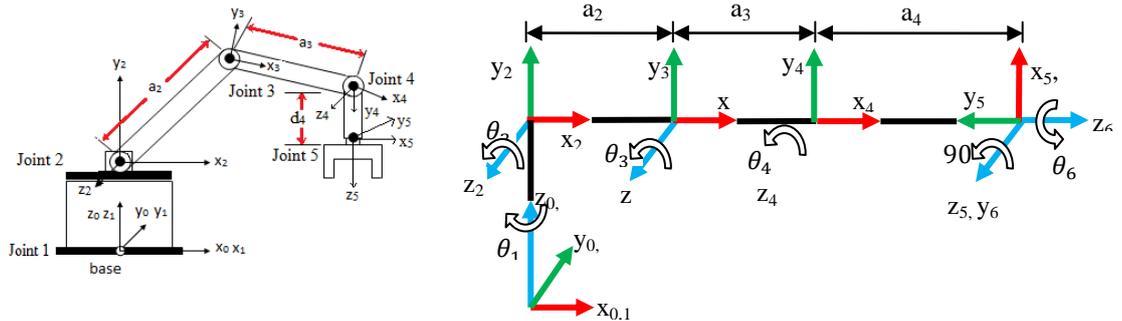
Arduino diciptakan untuk para pemula bahkan yang tidak memiliki dasar bahasa pemrograman sama sekali karena menggunakan bahasa C++ yang telah dipermudah melalui *library*. Arduino menggunakan *software processing* yang digunakan untuk menulis program kedalam Arduino. *Processing* sendiri merupakan penggabungan antara bahasa C++ dan Java. *Software IDE* Arduino terdiri dari 3 bagian yaitu editor program, *compiler*, dan *Uploader* [4]. Pada penelitian ini menggabungkan Arduino yang harganya murah dan LabVIEW yang memiliki aplikasi front panel yang tak terbatas dalam berbagai bidang aplikasi, maka perlu adanya komunikasi yang dapat menjembatani antara Arduino dengan LabVIEW, yaitu komunikasi serial Firmata yang merupakan sebuah protokol yang ditulis pada mikrokontroler (salah satunya adalah mikrokontroler Arduino). Firmata ini ditulis untuk memudahkan komunikasi Arduino dengan perangkat lunak yang lain termasuk LabView.

Firmata ini dapat memudahkan kita dalam membuat program karena tidak lagi dilakukan di kedua sisi, tetapi hanya di satu sisi yaitu sisi perangkat lunak komputer saja (LabVIEW).

4. HASIL DAN PEMBAHASAN

1.1 Forward Kinematik

Untuk dapat menentukan orientasi dan posisi *link* robot, maka dibutuhkan pemodelan kinematik. Dalam robotika, pemodelan kinematik merupakan bentuk pernyataan berisi deskripsi matematik geometri dari suatu struktur robot yang bergerak tanpa mempertimbangkan gaya penyebabnya. Dengan analisa *forward kinematic*, maka dapat ditentukan referensi input tiap aktuator yang berupa besaran sudut agar robot dapat melakukan gerakan untuk mencapai orientasi dan posisi yang dikehendaki [5]. Gambar 9 menunjukkan penempatan sumbu koordinat (*frame*) pada joint simulator arm robot berdasarkan kaidah tangan kanan.



Gambar 9. Koordinat frames untuk arm robot 5 dof.

Simulator *arm robot* terdiri beberapa *link* dan *joint*. *Joint* digunakan untuk menghubungkan setiap *link-link* yang ada dimana setiap *joint* mewakili satu derajat kebebasan. Untuk mendeskripsikan hubungan translasional dan rotasional antara *link-link* yang berdekatan maka digunakan metode Denavit Hartenberg (DH) parameter sebagai sebuah metode matriks yang secara sistematis membangun sebuah sistem koordinat dari masing-masing *link*. Tabel 2 menunjukkan DH parameter simulator *arm robot*. Berdasarkan DH parameter tersebut, maka selanjutnya dapat ditentukan transformasi matriks tiap koordinat frame dari link *i* ke *i+1* dengan menggunakan persamaan berikut ini:

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Tabel 2. DH parameter untuk simulator *arm robot*

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0^0	0	0	θ_1
2	90^0	0	0	θ_2
3	0^0	a_2	0	θ_3
4	0^0	a_3	0	θ_4
5	0^0	a_4	0	90^0
6	90^0	0	0	θ_6

Untuk menghitung *forward kinematic*, kita tentukan nilai $a_2=119$ mm, $a_3=165$ mm dan $d_4=47,5$ mm berdasarkan panjang link simulator *arm robot* yang dirancang, dilanjutkan menentukan sampling nilai sudut θ_i , misalnya $\theta_1= 60^0$, $\theta_2= 60^0$, $\theta_3= 60^0$, $\theta_4= 60^0$, $\theta_6= 60^0$. Selanjutnya dengan menggunakan bantuan software MATLAB, maka diperoleh persamaan transformasi matriks yang menyatakan matrik orientasi dan vektor posisi dari *base* sampai dengan *end effector* sebagai berikut:

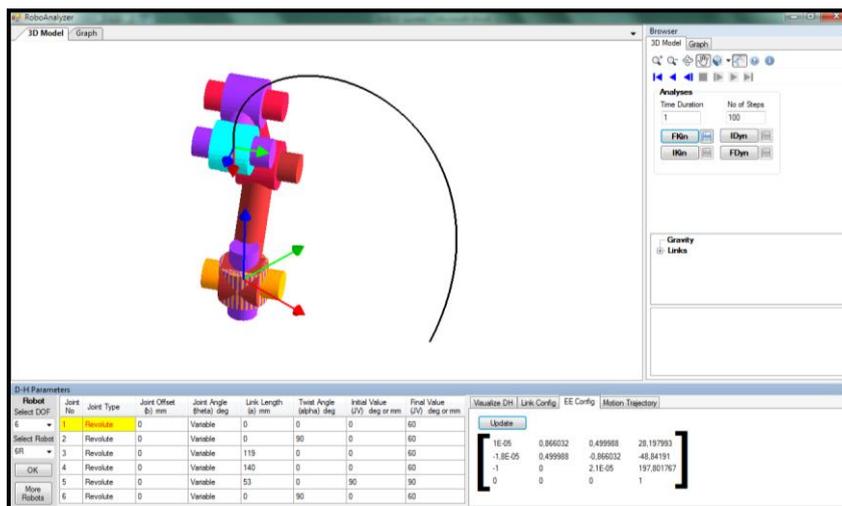
$${}^0T_5 = {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot {}^3T_4 \cdot {}^4T_5 \cdot {}^5T_6$$

$${}^0T_6 = \begin{bmatrix} \text{Matriks orientasi} & \text{Vektor posisi} \\ \hline -0,0000 & 0,8660 & 0,5 & 28,1997 \\ 0,0000 & 0,5000 & -0,8660 & -48,8433 \\ -1,0000 & 0 & -0,0000 & 197,8006 \\ 0 & 0 & 0 & 1,0000 \end{bmatrix}$$

Untuk memverifikasi hasil perhitungan *forward kinematic*, maka akan digunakan software *RoboAnalyzer* seperti yang ditunjukkan pada Gambar 10. Dengan memasukkan dimensi link dan besar sudut yang sama nilainya ketika menggunakan MATLAB, maka akan diperoleh hasil sebagai berikut:

$${}^0T_6 = \begin{bmatrix} 1E-05 & 0,8660 & 0,4999 & 28,1979 \\ -1,8E-05 & 0,4999 & -0,8660 & -48,8419 \\ -1,0000 & 0 & 2,1E-05 & 197,8018 \\ 0 & 0 & 0 & 1,0000 \end{bmatrix}$$

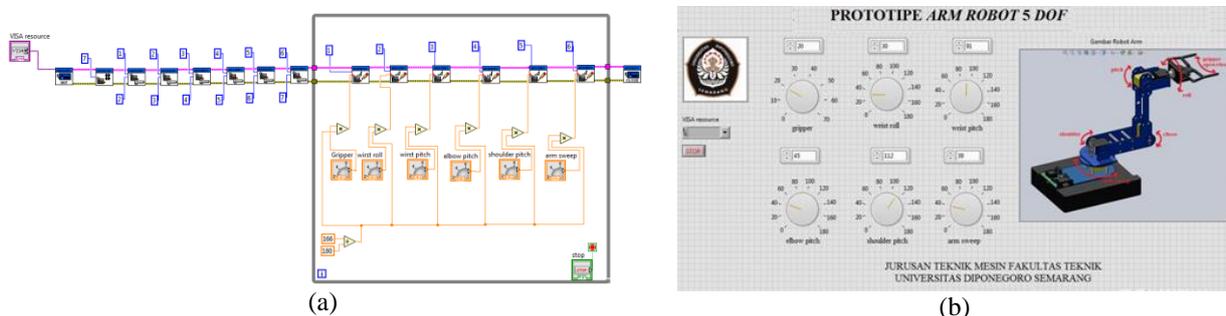
Terbukti bahwa hasil perhitungan menunjukkan bahwa nilai matriks transformasi *forward kinematics* antara MATLAB dan *RoboAnalyzer* bernilai sama sehingga hal ini menunjukkan bahwa transformasi yang dilakukan tipe link telah benar.



Gambar 10. Hasil transformasi *forward kinematic* dengan *RoboAnalyzer*

3.3 Desain software

Untuk perangkat lunak, Firmata pada Arduino-LabVIEW interface digunakan yang terdiri dari 7 file yaitu *LVIFA_Base*, *AFMotor.cpp*, *AFMotor.h*, *AccelStepper.cpp*, *AccelStepper.h*, *LabViewInterface.h*, dan *LabViewinterface*. Gambar 11 (a) menunjukkan block diagram pada LabVIEW yang terhubung dengan Arduino untuk mengontrol servo motor melalui front panel di LabVIEW yang ditunjukkan pada Gambar 11 (b).



Gambar 11. Blok diagram *arm robot* pada LabVIEW (a) dan *front panel arm robot 5 dof* (b)

Pada saat pengujian, sudut yang dimasukkan pada LabVIEW tidaklah sama dengan posisi sudut sebenarnya pada *servo motor*. Dengan demikian diperlukan suatu koreksi menggunakan *numerical divided* dan *numerical multiply*. *Numerical divided* berfungsi untuk membagi input nilai yang dimasukkan, sedangkan *numerical multiply* berfungsi untuk mengkalikan input nilai yang dimasukkan. Pada hasil akhir terdapat kesalahan/ketidaksesuaian (*error*) sebesar 1,67% setelah *input* 180° yang dihasilkan pada *servo motor* sebesar 183°.

4. KESIMPULAN

Pembuatan simulator *arm robot 5 dof* ini untuk mempresentasikan robot manipulator sebagai salah satu dari robot industri yang berguna dalam proses pembelajaran robotika pada dunia pendidikan. Simulator *arm robot* dibuat dengan biaya terjangkau berbasis mikrokontroler Arduino Uno bersama LabVIEW untuk pembuatan *front panel*. Firmata digunakan sebagai interface antara Arduino dengan LabVIEW. Hasil verifikasi menunjukkan tingkat *error* antara sudut pada *front panel* dengan sudut aktual pada *servo motor* sebesar $1,67^0$. Nilai ini masih dalam range toleransi. Selanjutnya analisa *forward kinematic* telah mendeskripsikan orientasi dan posisi simulator arm robot 5 dof dari *end effector* terhadap *base*.

5. REFERENSI

- [1] Craig, J., 1989, "*Introduction to Robotics, Mechanics and Control, 2nd Edition*," Addison Wiley, Inc, New York.
- [2] [Scherz](#), P., Monk, S., 2008, "*Practical Electronics for Inventors, Third Edition*," McGraw Hill Professional, New York.
- [3] LabVIEW, 1998, "*Data Acquisition Basics Manual*," National Instruments Corporation, Texas.
- [4] Waren, J.D., Adams, J., and Molle, H., 2011, "*Arduino Robotics*," Apress, New York.
- [5] Frankovsky, P., Hroncova, D., Delyova, I., Hudak, P., 2012, "*Inverse and Forward Dynamics Analysis of Two Link Manipulator*," *Procedia Engineering*, 48: 158-163.