FORWARD KINEMATICS SOLUTION FOR 5 DEGREE OF FREEDOM OF RHINO XR-3 ROBOT

Mohd Norhisham Bin Che Soh

Bachelor Degree's of Mechatronic Engineering

May 2009

"I hereby declared that I have read through this report and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Mechatronic Engineering"

Signature	: ……………………………….
Supervisor"s Name	: Pn Irma Wani Binti Jamaludin
Date	: 22 April 2009

FORWARD KINEMATICS SOLUTION FOR 6 DEGREE OF FREEDOM OF RHINO ROBOT

MOHD NORHISHAM BIN CHE SOH

B 010510101

This Report is Submitted in Partial Fulfillment of Requirements for the Degree of Bachelor in Mechatronic Engineering

FACULTY OF ELECTRICAL ENGINEERING

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

MAY 2009

"I hereby declared that this report is a result of my own work except for the excerpts that have been cited clearly in the references."

Signature : ………………………………

Name : Mohd Norhisham Bin Che Soh

Date : 22 April 2009

ACKNOWLEDGEMENT

Alhamdulillah, finally the Final Year Project (FYP) is complete. I take a great pleasure in this opportunity to thank many persons who generously advice and assists me while I was doing my FYP which is compulsory to all Universiti Teknikal Malaysia Melaka (UTeM) students to pass before awarded the degree.

First of all, I would express my deepest thanks and gratitude to my project supervisor, Pn Irma Wani bt Jamaludin for undivided support morally and physically, assistance, guidance, tolerance, which proved to be invaluable as to the completion of my FYP.

I also would like to thank the panel, Mr. Ahmad Zaki bin Shukor, Mr. Farez bin Ali@Ibrahim, Pn Nur Ilyana Anwar Apandi and Mr. Farhan Bin Hanaffi whose give me a good comment during my presentation. I also would like to take this opportunity to express my appreciation to my family and friends for their patients, understanding and also for their undivided support that they had gave me throughout the completion of my project.

Last, but not least I also would like to thank all those helping and supporting me directly and indirectly during my Final Year Project.

ABSTRACT

This project develops an algorithm for solving the forward kinematics equation. The forward kinematics equation is solving to determine the gripper position and orientation by using suitable programming software for 5 Degree of Freedom (DOF) of Rhino XR-3 robot. The process involves the control of parameter known as angles (θ) of rotation linkages of the three main axes of the robot. This parameter is determined by user before the forward kinematics equation is solved. Its forward kinematics problem will be solved by using conventional method and simulation. Result from the conventional method is used as guidelines for developing an algorithm to ensure the algorithm is correct and can be use for solving the forward kinematics equation. The graphical user interface will be design where user needs to setting up the value of control parameter and then solve the forward kinematics equation. The correct output of interface prove that this command can be used for solves forward kinematics equation. Overall, this project will describe the technique, calculation, method and algorithm on how the forward kinematics is solved.

ABSTRAK

Projek ini membangunkan algoritma untuk menyelesaikan permasalahan kinematik hadapan. Permasalahan kinematik hadapan ini diselesaikan untuk menentukan kedudukan dan pergerakan cekam dengan menggunakan perisian yang sesuai untuk robot Rhino XR-3. Proses yang terlibat ialah kawalan terhadap parameter iaitu sudut putaran bagi 3 paksi. Parameter ini akan ditentukan oleh pengguna bagi menyelesaikan permasalahan kinematik hadapan tersebut. Permasalahan kinematik hadapan akan diselesaikan menggunakan kaedah konvensional dan simulasi. Hasil dari penyelesaian menerusi kaedah konvensional akan digunakan sebagai rujukan untuk membangunkan algoritma supaya algoritma tersebut sesuai dan boleh digunakan untuk menyelesaikan permasalahan kinematik hadapan. Seterusnya, pengantara muka akan dibangunkan di mana pengguna perlu memasukkan nilai parameter sebelum permasalahan tersebut diselesaikan. Hasil yang tepat dari pengantara muka ini membuktikan bahawa pengantara muka ini boleh digunakan untuk menyelesaikan permasalahan kinematik hadapan. Secara keseluruhannya, projek ini menerangkan tentang teknik, pengiraan, kaedah dan algoritma bagaimana permasalahan kinematik hadapan diselesaikan.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DH	-	Denavit Hartenberg
DK	-	Direct Kinematics
DOF	-	Degree of Freedom
GUI	-	Graphical User Interface
KAP	-	Kinematics Analysis Program

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

The "Forward Kinematics Solution for 5 Degree of Freedom of Rhino XR-3 Robot" project is to identify and develope the programming software which is to design the graphical user interface (GUI). This GUI will be used to solve the forward kinematics problem for 5 DOF of Rhino XR-3 robot. This chapter will discuss about the background of the project, its concept, objective, scope of the project and the problem statement.

1.1 BACKGROUND OF THE PROJECT

Kinematics is the science of motion which treats motion without regard to the forces that causes it [1]. While the forward kinematics problem of Rhino XR-3 robot is to compute the position and orientation of the robot"s end-effectors relative to the base of the robot [8].

In order to solve the kinematics problem, a conventional method which has a lot of complexity equation and calculation must be applied. However, it is hard to be implemented in practice even many industrial robots are built with simple geometries to simplify the associated kinematics computations such as Rhino XR-3 robot.

It is now well accepted that the use a package of software is the best options to solve the forward kinematics problem. For this project, a graphical user interface will be designed to solve this problem which focuses only for Rhino XR-3 robot. The advantage of using this software is will minimize time for solving this problem.

1.2 OBJECTIVE

The main objective of this project is to solve a forward kinematics problem for 5 DOF of Rhino XR-3. In order to solve the problem, the suitable programming software will be identified and then the algorithm will be developed. After that, a user friendly GUI will be developed. Specific objectives of this project are:

i. To identify a suitable programming software for solving forward kinematics equation.
ii. To develop an algorithm based on the programming software.
iii. To design a graphical user interface by using the identified of programming software and solve the forward kinematics problem for Rhino XR-3.

1.3 SCOPE

The scope of this project is cleared where this GUI is developed for solving the forward kinematics problem only for Rhino XR-3 robot. The GUI is also developed only for calculation the final position without any simulation of the operation of this robot. Thus, this GUI is not suitable to other type of robot unless both algorithm and GUI is modified.

Besides, the algorithm used for solve the forward kinematics is only Denavit Hartenberg (DH) Algorithm. The DH algorithm provided a systematic matrix method, based on the homogenous transformation theory to describe the position and orientation of each link of the tool tip with respect to its neighboring link in a static situation [9]. This method is chosen because it is widely used in robotics for obtaining the kinematics solutions of robot manipulators.

1.4 PROBLEM STATEMENT

Many industrial robots are built with simple geometries such as intersecting or parallel joint axes to simplify the associated kinematics computations. The mathematical complexity of solving robots of general architecture detracts instructors and students from using robots with arbitrary structures in illustrative examples and assignments [2].

This is also one of the major problems for the 5 Degree of Freedom of an industrial robot which is to solve the forward kinematics. Solving the forward kinematics problem need a lot of calculation when using conventional method. The complexity of formulation and computational burden make this method hard to be implemented in practice which also will waste a lot of time [10]. Therefore, a new method of solution must be developed to solve the forward kinematics equation such as using programming software.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a review of previous research project that are related with this project will be discussed. The information about forward kinematics and Rhino XR-3 robot are also described in this chapter.

2.1 DK ANALYSIS OF 5 AXIS RHINO XR-3 ROBOT

DK Analysis of 5 Axis Rhino XR-3 Robot is C++ programs written without GUI by N.R. Narayana Murthy [3] for solving the forward kinematics for Rhino XR-3 robot. However, this program still useful for research where the output of the program is used as a comparison with the program of this project.

This program allowed user to enter the Denavit Hartenberg (DH) parameter table for the robot to perform the forward kinematics. The DH-parameters can be entered from the keyboard. Then, the calculation for the transformation matrix of base frame with respect to next frame is done one by one for 5 degree of freedom. Finally, the final position of the gripper for Rhino XR-3 robot is obtained.

The program and output for direct kinematics analysis of 5 axis Rhino XR-3 robot is as Appendix 2. Some editing should be done to this program since some error occurs when running the program.

2.2 Sirf Rhino (RhinoGui Interface)

The RhinoGui is an interface developed by Venkatraghavan Gourishankar, (May 2006) which can be used to experiment with the basic kinematics of the Rhino XR-3 robot [4]. The user interaction with the interface has been made as simple as possible. Three main files, the „RhinoForwardKinematics.m", „RhinoInverse.m" and „trajectory.m" work behind the interface in providing the user with the kinematics of the robot. The „Robotics toolbox for Matlab" is installed provides the visualization automatically [4].

The RhinoGui interface is shown in Figure 2.1. The forward Kinematics can be performed on the robot by setting up the required joint angles either by the edit box or by the sliders and then clicking the button named „Forward Kinematics". As soon as the forward kinematics button is clicked, the plot of the robot configuration changes to the user set configuration. The Arm matrix for the current configuration is also updated accordingly. The startup values for the joint angles and the arm matrix are the values for the Rhino XR-3 robot in its home position. The home position has the following joint angles and arm matrix Joint angles – theta1 = 0, theta2 = -90, theta3= 90, theta4 = 0 and theta5 = 0. All the joint angles are in degrees. All values are in meters. By default the last row of the arm matrix is not shown in the interface since it remains the same all the time [4].

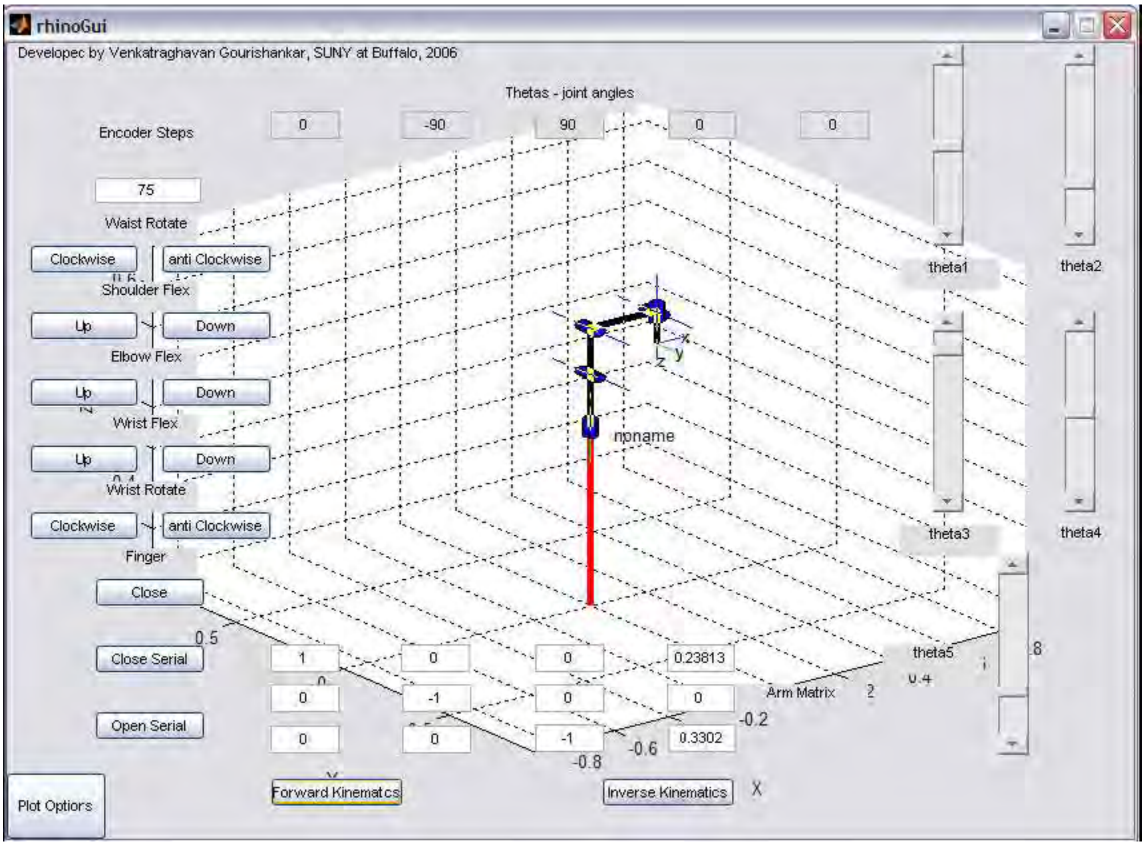

Figure 2.1: RhinoGui interface.

2.3 Kinematics Analysis Program (KAP)

Kinematics Analysis Program is a program developed by Rachid Manseur [2]. KAP is currently developed to compute the forward and inverse kinematics of robot manipulators with 5 or 6 revolute degrees of freedom. The functional block diagram of KAP is shown on Figure 2.3. There are two type of KAP:

i. KAP5 – to compute the inverse kinematics problem for 5 DOF of Rhino XR-3 robot (See Figure 2.2).
ii. KAP6 – to compute the forward kinematics of Fanuc robot.

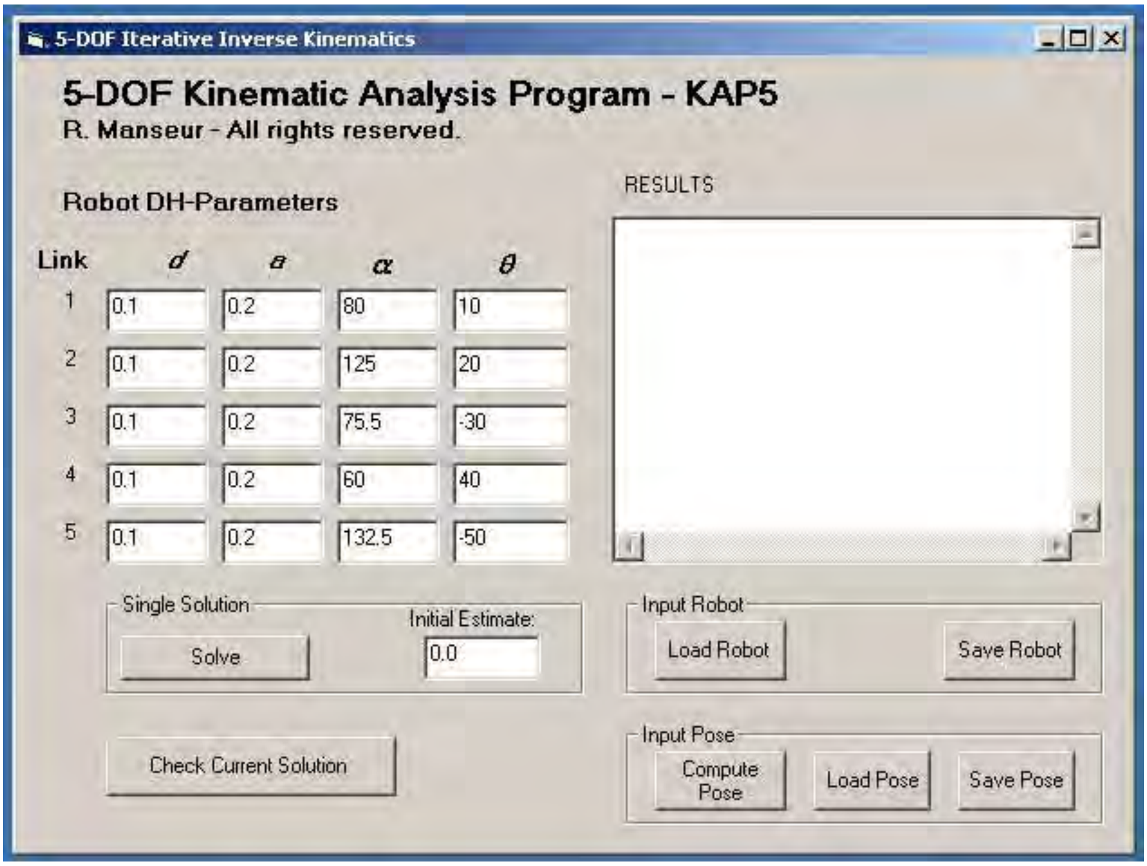

Figure 2.2: GUI of KAP5.

In this program, the user is asked to enter the DH parameters table for the robot. The DH parameters can be entered from the keyboard or by specifying a pre-existing file name containing the robot DH parameters in the format shown on Table 2.1. The end effectors pose can also be entered through the keyboard, read from a pre-existing file named POSE.DAT, or computed from joint angles that the user enters through the forward kinematics block. The program determines the number of DOFs of the robot and applies the proper algorithm [2]. Even this program is to compute the inverse kinematics of Rhino XR-3 robot, it still useful as a sample for design a GUI for this project.

Table 2.1: Robot DH-parameter table

Link	d	a	α	θ
1	d_1	a_1	α_1	θ_1
2	d_2	a_2	α_2	θ_2
3	d_3	a_3	α_3	θ_3
4	d_4	a_4	α_4	θ_4
5	d_5	a_5	α_5	θ_5

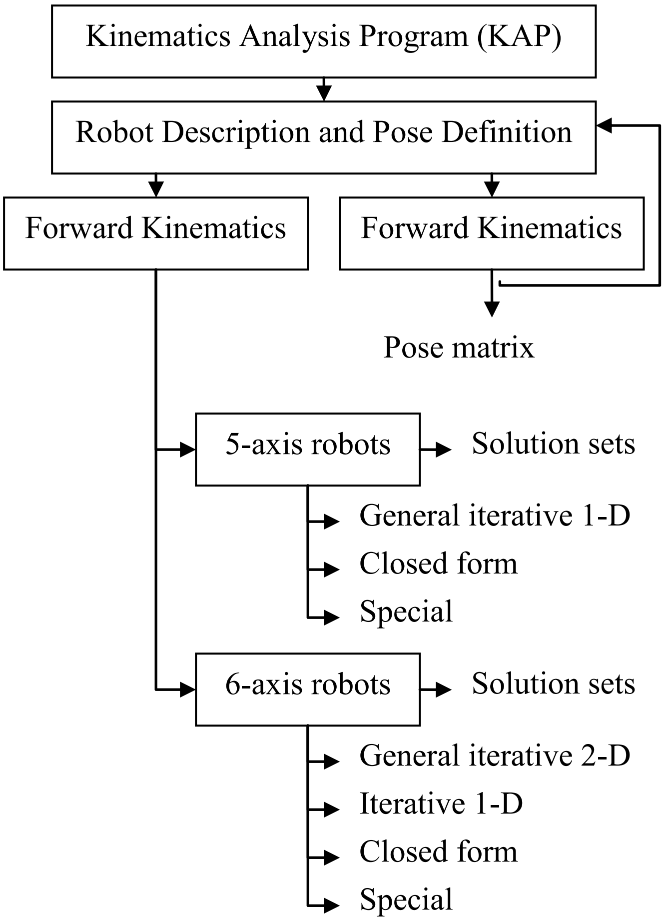

Figure 2.3: Functional Block Diagram of KAP

2.4 Forward Kinematics

The forward kinematics problem is concerned with the relationship between the individual joints of the robot manipulator and the position and orientation of the tool or end-effector. Stated more formally, the forward kinematics problem is to determine the position and orientation of the end-effector, given the values for the joint variables of the robot [5].

In order to describe the location of each link relatives to its neighbors we define a frame attached to each link. The link frames are named by number according to the link to which they are attached. That is, frame $\{i\}$ is attached rigidly to link i. To locate frames on the links, this convention is used: The Z-axis of frame $\{i\}$, called Z_i is coincident with the joint axis i. The origin of frame $\{i\}$ is located where a_i perpendicular intersects the joint i axis. X_i points along a_i in the direction from joint i to joint $i+1$ [1].

In the usual case of revolute joint, θ_i is called the joint variable, and the other three quantities would be fixed link parameters. For prismatic joints, d_i is the joint variable and the other three quantities are fixed link parameters. The definition of mechanism by means of these quantities is a convention usually called DH notation [1].

2.4.1 Denavit Hartenberg Representation

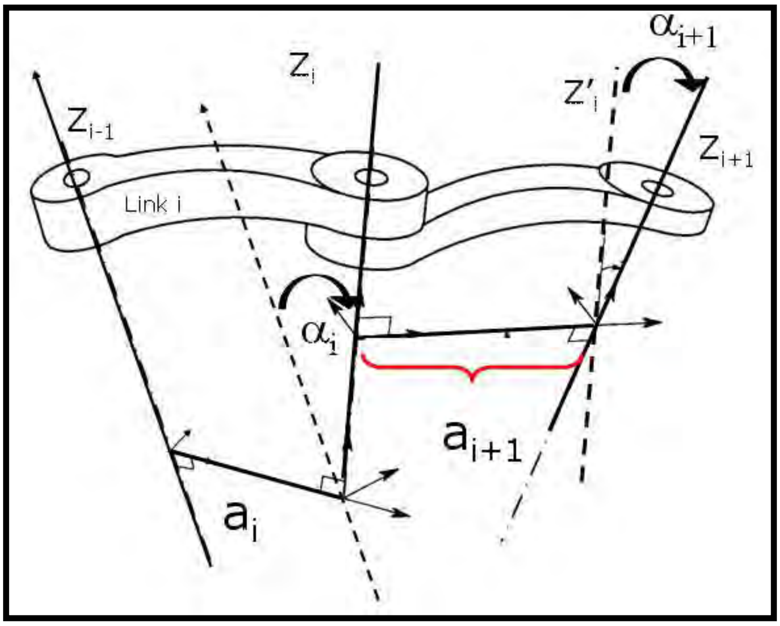

Figure 2.4: Link Parameter.

A commonly used convention for selecting frames of reference in robotic applications is the Denavit-Hartenberg, or DH convention. In this convention, each homogeneous transformation A_i is represented as a product of four basic transformations. The procedure based on the DH convention in the following algorithm for deriving the forward kinematics for any manipulator are summarize as below [5]:

Step 1 : Locate and label the joint axes $z_0, ..., z_{n-1}$.

Step 2 : Establish the base frame. Set the origin anywhere on the z_0-axis. The x_0 and y_0 axes are chosen conveniently to form a right-hand frame. For $i = 1, ..., n-1$, perform Steps 3 to 5.

Step 3 : Locate the origin O_i where the common normal to z_i and z_{i-1} intersects z_i. If z_i intersects z_{i-1} locate O_i at this intersection. If z_i and z_{i-1} are parallel, locate O_i in any convenient position along z_i.

Step 4: Establish x_i along the common normal between z_{i-1} and z_i through O_i, or in the direction normal to the $z_{i-1} - z_i$ plane if z_{i-1} and z_i intersect.

Step 5 : Establish y_i to complete a right-hand frame.

Step 6 : Establish the end-effector frame $o_n x_n y_n z_n$. Assuming the n[th] joint is revolute set $z_n = a$ along the direction z_{n-1}. Establish the origin o_n conveniently along z_n, preferably at the center of the gripper or at the tip of any tool that the manipulator may be carrying. Set $y_n = s$ in the direction of the gripper closure and set $x_n = n$ as $s \times a$. If the tool is not a simple gripper set x_n and y_n conveniently to form a right-hand frame.

Step 7 : Create a table of link parameters $a_i, d_i, \alpha_i, \theta_i$.

$a_i =$ Distance along xi from o_i to the intersection of the xi and z_{i-1} axes.

$d_i =$ Distance along z_{i-1} from O_{i-1} to the intersection of the x_i and z_{i-1} axes. d_i is variable if joint i is prismatic.

$\alpha_i =$ The angle between z_{i-1} and z_i measured about x_i (see Figure 2.4).

$\theta_i =$ The angle between x_{i-1} and x_i measured about z_{i-1} (see Figure 2.4). θ_i is variable if joint i is revolute.

Step 8 : Form the homogeneous transformation matrices A_i by substituting the above parameters into equation 2.1.

Step 9 : Form $T_n^0 = A_1 \ldots . A_n$. This then gives the position and orientation of the tool frame expressed in base coordinates.

The overall transformation is obtained by post multiplication of individual transformations:

$$T_i^{i+1} = Rot(z, \theta_i) Trans(0,0,d_i) Trans(0,0,a_i) Rot(x, \alpha_i) \quad (2.1)$$

$$T_i^{i+1} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & -s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$